Deep Learning meets Knowledge Graphs for Scholarly Data Classification

Fabian Hoppe
fabian.hoppe@fiz-karlsruhe.de
FIZ Karlsruhe – Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology
Karlsruhe, Germany

Danilo Dessì
danilo.dessi@fiz-karlsruhe.de
FIZ Karlsruhe – Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology
Karlsruhe, Germany

Harald Sack
harald.sack@fiz-karlsruhe.de
FIZ Karlsruhe – Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology
Karlsruhe, Germany

ABSTRACT

The amount of scientific literature continuously grows, which poses an increasing challenge for researchers to manage, find and explore research results. Therefore, the classification of scientific work is widely applied to enable the retrieval, support the search of suitable reviewers during the reviewing process, and in general to organize the existing literature according to a given schema. The automation of this classification process not only simplifies the submission process for authors, but also ensures the coherent assignment of classes. However, especially fine-grained classes and new research fields do not provide sufficient training data to automatize the process. Additionally, given the large number of not mutual exclusive classes, it is often difficult and computationally expensive to train models able to deal with multi-class multi-label settings. To overcome these issues, this work presents a preliminary Deep Learning framework as a solution for multi-label text classification for scholarly papers about Computer Science. The proposed model addresses the issue of insufficient data by utilizing the semantics of classes, which is explicitly provided by latent representations of class labels. This study uses Knowledge Graphs as a source of these required external class definitions by identifying corresponding entities in DBpedia to improve the overall classification.

CCS CONCEPTS

• **Information systems → Information systems applications**; **Clustering and classification**.

KEYWORDS

Multi-Label Classification, Deep Learning, Scholarly Data, Knowledge Graphs

ACM Reference Format:
Fabian Hoppe, Danilo Dessì, and Harald Sack. 2021. Deep Learning meets Knowledge Graphs for Scholarly Data Classification. In *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3442442.3451361

WWW '21 Companion, April 19–23, 2021, Ljubljana, Slovenia

ACM ISBN 978-1-4503-8313-4/21/04.
https://doi.org/10.1145/3442442.3451361

1 INTRODUCTION

The ever increasing amount of scholarly papers that are being published today brings unprecedented challenges to manage, find, and explore research outcomes. Researchers have been mobilized to investigate how scholarly papers can be classified according to various classes describing their content to provide a multitude of services. This may support the retrieval of research papers, the search of suitable reviewers for the reviewing process, and the organization of scientific literature according to a given schema. Given the large amount of new publications, automatic annotation systems are crucial for human annotators working at digital libraries to annotate the research papers into classes. Typically, this can be done either by discovering topics within the content of research papers [3], or by applying classification methods in predefined classes of an existing domain vocabulary, e.g., Medical Subject Headings[1] (MeSH), Mathematics Subject Classification[2] (MSC), and ACM Computing Classification System[3] (CCS).

The first class of methods usually relies on methodologies to detect scientific topics. However, they usually produce noisy results [12]. This depends on the fact that the discovered topics cannot easily be mapped to controlled vocabularies and ontologies. Conversely, the latter is based on standard text classification techniques with predefined classes which might suffer from the presence of fine-grained classes and new research fields which do not provide sufficient training data. The content of research papers is related to classes only by modelling the text data as a set of term frequencies or embedding representations, regardless a proper representation of the semantics behind classes. It follows that this information of classes is lost.

Recently, text classification approaches have been based on the Deep Learning paradigm, which has raised the interest of the scientific community because of its ability to achieve impressive results and to scale across different datasets [14, 19]. With this paradigm, models can be set to solve a classification problem by computing relatedness between the input resources and the classes. Common approaches for Deep Learning text classification use word embedding representations; however, while it is possible to represent the knowledge within text by combining word embeddings, the same does not apply to class labels which usually contain only few words. Thus, the use of the Deep Learning paradigm is crippled by a poor semantics associated to classes.

[1] https://www.nlm.nih.gov/mesh/meshhome.html
[2] https://msc2020.org/#
[3] https://www.acm.org/publications/class-2012

Recent studies have shown that Semantic Web resources and techniques can be leveraged to augment the results of Machine Learning methodologies through the exploitation of background knowledge provided by linked open data [16]. Especially in the last years, we are witnessing an increasing development of ontologies and Knowledge Graphs describing millions of entities and their relations. These can be leveraged to capture the semantics of classes to feed Deep Learning models. In fact, links of entities that represent classes can describe class interactions, providing explicit semantics between them (e.g., by traversing through their paths in a Knowledge Graph) and, therefore, enabling to create embedding representations [15] that supply more semantics than the simple embeddings associated to the words of class labels. For example, the entity *Artificial Intelligence*[4] in DBpedia [9] might provide much more information than the word embeddings of the class label words *Artificial* and *Intelligence*. Including this information in embedding representations can lead to more semantics that can be supplied to a Deep Learning model to better learn the relatedness between text data and target classes.

This paper investigates the use of a Deep Learning model for classifying scholarly research papers about the Computer Science domain, augmenting the classification model with external knowledge from DBpedia. In summary, this paper provides the following main contributions:

- A preliminary Deep Learning architecture for classifying scholarly research papers in a multi-label setting is proposed.
- Word and Knowledge Graph embeddings are explored for class representations to obtain better classification results.
- Insights about the use of Knowledge Graphs for the scholarly domain are provided.

The source code and all the resources are freely available through a GitHub repository[5].

The reminder of this paper is organized as follows. Section 2 discusses the related work. Section 3 presents the proposed approach, detailing the components, and justifying design decisions. Section 4 reports and discusses the evaluation. Finally, Section 5 concludes the paper and outlines future directions.

2 RELATED WORK

Previous studies for the semantic annotation of scientific literature have been already performed in the recent past. To start with, in [18] an approach based on Logistic Regression was proposed to classify Mathematics papers with 63 upper classes from MSC in a multi-class setting. In their work, the authors employed tf-idf features of the texts and made various experiments to understand if only the title, or the combination of title and paper abstract, allowed to achieve the best performance of the trained model. This problem was also explored recently by [10], where various Deep Learning models were evaluated with different amounts of text from research papers as input data. More precisely, since machine learning algorithms require a large amount of training data and the full-text of papers is not always available, the authors investigated whether using only the title of a vast amount of research papers is sufficient to train a better model than using the full-text of papers but with less examples. The authors discovered that the approach by using a large dataset with titles only achieved comparable, and in some cases, better results. Another attempt to classify research papers into classes was made in [4] where the authors applied a string-to-text approach for producing self-learned labels to annotate research papers according to the ACM CCS 2012 schema. As the reader may notice, these approaches did not consider any class representations, thus did not exploit all the available semantics.

A recent approach that leveraged latent semantics to annotate research papers was proposed by [17]. The authors used Word2Vec [11] word embeddings to represent Computer Science topics and n-grams from the text of research publications. They used a threshold method on the cosine similarity score between them to relate topics to a publication. However, their approach was limited to only topics from the Computer Science Ontology, and did not take into account the semantics behind the topic label words. A similar approach was also explored in other domains which share commonalities with the annotation of research papers (i.e., the amount of text data is limited to the title of documents and short descriptions) and require automatic annotations. To name a recent example, in [7], an unsupervised approach was applied to classify archival documents using their title against a set of class labels. The authors used word embeddings to detect the relatedness between the text data and the labels. However, the label representations via word embeddings limited the overall performance of the approach.

The same limitation also applies to Deep Learning methods previously designed for text classification tasks. In fact, in order to learn patterns from the data, a model is usually fed with word embeddings which are combined by different techniques. For example, in [14] semantics of text data was captured either by averaging the word embeddings or by using a Long-Short Term Memory layer. However, for the classes, only word embeddings were used and, therefore, their representation still remained limited. This motivated us to explore a Deep Learning approach combined with external knowledge originating from Knowledge Graphs to classify research papers according to a predefined schema. In contrast to previous works, the preliminary Deep Learning model learns the semantics of abstracts of research papers from their word embeddings, and the semantics of classes from their latent representations built on top of DBpedia.

3 KNOWLEDGE-AWARE DEEP MODEL

In this section, the preliminary proposed approach to classify research publications is described.

3.1 Problem Statement

The proposed approach has been developed to solve a binary classification problem with the goal to capture the relatedness between the semantics of text and the semantics of classes. In detail, given a scientific paper p and a class c from a set C, the model aims to learn a function $\gamma : (p, c) \rightarrow \{0, 1\}$ that computes 1 if p belongs to the class c, 0 otherwise. For doing so, it embeds the text from paper abstracts, the text from class label words, and the embedding representation from the corresponding DBpedia entities. Then it trains neural network layers to predict the label 0 or 1 for the input pair (p, c).

[4] http://dbpedia.org/page/Artificial_intelligence

[5] https://github.com/ISE-FIZKarlsruhe/DL-meets-KG-Scholarly-Classification

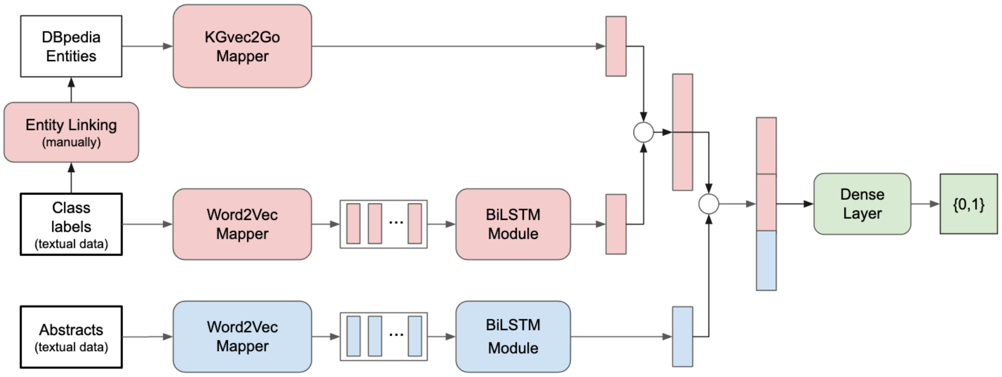

Figure 1: The proposed Deep Learning model for scholarly paper classification.

3.2 Dataset

The ArXiv corpus[6] is a collection of scholarly articles published on *arXiv.org*. During the publication process the authors assign their paper to at least one class of a taxonomy[7]. Therefore, the ArXiv corpus is a manually annotated multi-class multi-label classification dataset. However, due to the required mapping to KG entities for each class, this work considers only a subset of the ArXiv corpus. It is created by selecting all articles which are exclusively labelled with one or multiple of 21 fine-grained classes from the computer science domain. These classes are selected because they can be unambiguously mapped to DBpedia entities.

This dataset contains 92, 195 articles of which each article is on average labeled with 1.28 classes. The articles are distributed in a long-tail distribution over the classes. The largest class (*Computer Vision*, *cs.CV*) contains 35, 252 articles and the smallest class (*Mathematical Software*, *cs.MS*) only 685 articles. Each article is described by a title, an abstract and additional bibliographic information, such as authors. In order to keep the presented preliminary model simple, it only considers the abstract of all articles. The dataset is split into training and test set by random selection of documents with a train-test ration of 70% train set and 30% test set. Additionally, 10% of the training data are used as validation dataset. The multi-label dataset is transformed into a binary dataset to align with the input format of the proposed approach. For the training set a negative sub-sampling is applied. Instead of using all negative examples for each document 5 random negative classes are selected.

3.3 Embedding Representations

In order to represent the abstracts and classes of scientific papers, the pre-trained Word2Vec model[8] built on the Google News dataset is employed. This choice was made due to the assumption that words within a domain tend to hold their meaning across different research papers. Therefore, more advanced and contextualized representations (e.g., BERT [6]) might add additional and unnecessary overload in this preliminary study. This is more stressed as the classes are represented by word embeddings, too. In fact, since their labels have only few words, it is not possible to fine tune the model. Therefore, the combination of sub-word representations to represent classes might lead to embeddings that do not carry any additional and salient information compared to Word2Vec embeddings.

As an external knowledge source for our classes the Knowledge Graph embeddings provided by *KGvec2go*[9] [13] are employed. In particular, within this work Knowledge Graph embeddings trained on DBpedia, one of the hubs of the linked data cloud built on top of Wikipedia and other Wikimedia projects, are utilized. The required mapping has been manually done by a domain expert. In doing so, the class representations can benefit from a large amount of knowledge.

3.4 Deep Model

This section describes the components that are designed within the deep model depicted in Figure 1.

KGvec2Go Mapper This module takes as an input a class c and returns a Knowledge Graph embedding e_c by looking up the class from a hand-crafted map. The reader may notice that in the current setting only classes that have been mapped can be used.

Word2Vec Mapper The text of paper abstracts and class label words are preprocessed using a *Word2Vec Mapper* module. More specifically, given an input text $t = [w_0, ..., w_n]$ where w_i is the i-th token of t, and M the pretrained Word2Vec model, this module returns a list $[e_0, ..., e_n]$ where e_i is the word embedding of w_i in M. If a token w_j is not present in M, it is discarded.

BiLSTM Module This module uses a bidirectional Long-Short Term Memory neural network implemented by adopting the keras[10] framework. This neural network employs recurrent connections and memory blocks in the hidden layers, and is able to learn patterns from data sequences. Therefore, it is suitable to capture semantics from text by processing the sequence of its tokens. The use of a bidirectional neural network has the advantage to capture patterns in both forward and backward directions.

Putting everything together The output of the model branches are concatenated and used to feed a *Dense* layer which should predict the probability of a paper p to belong to a class c.

[6] https://www.kaggle.com/Cornell-University/arxiv
[7] https://arxiv.org/category_taxonomy
[8] https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit?usp=sharing
[9] http://data.dws.informatik.uni-mannheim.de/kgvec2go/dbpedia/
[10] https://keras.io/

Table 1: Evaluation in terms of Hamming Loss, Precision, Recall, and F-measure of the proposed model and baselines.

Model	Hamming Loss	Precision micro; macro	Recall micro; macro	F-measure micro; macro
Random Classifier	0.499	0.061; 0.061	0.502; 0.380	0.109; 0.102
NLI ZSL	0.171	0.157; **0.293**	0.416; 0.408	0.229; **0.250**
BiLSTM (label)	0.52	0.066; 0.067	**0.578**; **0.438**	0.119; 0.112
BiLSTM (KG)	0.105	0.253; 0.234	0.373; 0.292	0.301; 0.236
BiLSTM (label + KG)	**0.096**	**0.275**; 0.248	0.348; 0.272	**0.307**; 0.237

4 EVALUATION

This section reports and discusses the preliminary evaluation of the proposed model.

4.1 Evaluation setting

The model has been trained using a *ReLU* activation function, a batch-size of 16, and an early stopping mechanism monitoring the binary cross-entropy on the validation set with a patience of 3 epochs. Both BiLSTM modules use a dropout of 0.17 and output 50-dimensional vector representations. All the experiments are run on a NVIDIA TITAN X GPU on a server with 1TB of RAM. To evaluate the proposed Deep Learning model, *Hamming Loss*, *Precision*, *Recall*, *F-measure* scores have been computed.

4.2 Baseline

The results of our model are compared against two baselines.

- A random selection which reflects how the classification tasks would be solved without any training and domain knowledge.
- A state-of-the-art Zero-shot Learning model[11] trained on the Natural language Inference (NLI) task. It poses a text sequence to be classified as a premise. The model has the goal to evaluate whether a hypothesis (i.e., the class label for the scope of this paper) on the premise holds, and returns a probability to score the entailment or contradiction. This model has obtained impressive results for many classification tasks [20].

4.3 Results and Insights

The results of the proposed model are reported in Table 1.
Random selection. As expected the random selection performs poorly, and although a good recall score has been achieved, the precision is very low, indicating a high error rate of its adoption.
NLI ZSL. The NLI ZLS model yields good scores in terms of all considered metrics. Indeed, this model performs well at being applied on any kind of classification scenarios due to its design which considers class semantics.
BiLSTM (label). The proposed model that uses only the word embeddings about class label words is not able to obtain satisfying results, and does not add improvements when compared with the random selection. This suggests that word embeddings alone are not sufficient to infer the semantics of classes for solving the classification task.
BiLSTM (KG). The proposed model trained only using the branch that considers Knowledge Graph embeddings is able to obtain good results which are comparable to those obtained by the NLI ZSL model. It is clear that the model benefits from the semantics contributed by the Knowledge Graph embeddings.
BiLSTM (label + KG). The full proposed model which tries to get benefits from both word and Knowledge Graph embeddings shows good results and is able to outperform the NLI ZSL model in some cases. More specifically, this model obtains the lowest hamming loss score, indicating its suitability to predict the correct classes with a small margin error. Although the improvement over the BiLSTM (KG) model is not remarkable, these results show that the path of combining different knowledge sources and representations may be worth to be investigated.

5 CONCLUSION

This paper describes how to combine Deep Learning methodologies with external knowledge resources with the aim to enhance the overall classification of scholarly papers. In particular, it shows the advantages of using Knowledge Graphs to better represent classes and come up with more accurate models. In addition, the model may further achieve better results if more classes will be employed since it might better learn patterns from the class representation space, thus creating more sophisticated deep models (e.g., Few-shot Learning and Zero-shot Learning models) targeting the scholarly domain. More precisely, the current setting does not limit the model to known and predefined classes, but may allow to scale to unseen classes, thus targeting more complicated scenarios where not all classes have sufficient training data such as extreme multi-label classification.

As future work, we are planning to employ domain-specific Knowledge Graphs, such as ORKG [8] for life and social sciences, AI-KG [5] for the Computer Science domain, HierClasSArt [1] for the Mathematics domain, and AIDA [2] for industry, to represent both classes and papers' content (e.g., by extracting the entities and using their Knowledge Graph embedding representations). Finally, we are also interested in investigating various deep neural layers that might better capture the patterns from the different latent representations and, consequently, improve the overall classification performance.

[11] https://huggingface.co/transformers/main_classes/pipelines.html#transformers.ZeroShotClassificationPipeline

REFERENCES

[1] Mehwish Alam, Russa Biswas, Yiyi Chen, Danilo Dessì, Genet Asefa Gesese, Fabian Hoppe, and Harald Sack. 2021. HierClasSArt: Knowledge-Aware Hierarchical Classification of Scholarly Articles. In *Companion Proceedings of the Web Conference 2021*. https://doi.org/10.1145/3442442.3451365

[2] Simone Angioni, Francesco Osborne, Angelo Salatino, Diego Reforgiato Recupero, and Enrico Motta. 2019. Integrating Knowledge Graphs for Comparing the Scientific Output of Academia and Industry. (2019).

[3] Levent Bolelli, Şeyda Ertekin, and C Lee Giles. 2009. Topic and trend detection in text collections using latent dirichlet allocation. In *European Conference on Information Retrieval*. Springer, 776–780.

[4] Ekaterina Chernyak. 2015. An approach to the problem of annotation of research publications. In *Proceedings of the eighth ACM international conference on web search and data mining*. 429–434.

[5] Danilo Dessì, Francesco Osborne, Diego Reforgiato Recupero, Davide Buscaldi, Enrico Motta, and Harald Sack. 2020. AI-KG: An Automatically Generated Knowledge Graph of Artificial Intelligence. In *The Semantic Web - ISWC 2020 - 19th International Semantic Web Conference, Athens, Greece, November 2-6, 2020, Proceedings, Part II*. 127–143. https://doi.org/10.1007/978-3-030-62466-8_9

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Fabian Hoppe, Tabea Tietz, Danilo Dessì, Mirjam Sprau, Mehwish Alam, and Harald Sack. 2020. The Challenges of German Archival Document Categorization on Insufficient Labeled Data. In *Proceedings of the Third Workshop on Humanities in the Semantic Web (WHiSe 2020) (CEUR Workshop Proceedings, Vol. 2695)*. CEUR-WS.org, 15–20.

[8] Mohamad Yaser Jaradeh, Allard Oelen, Kheir Eddine Farfar, Manuel Prinz, Jennifer D'Souza, Gábor Kismihók, Markus Stocker, and Sören Auer. 2019. Open research knowledge graph: next generation infrastructure for semantic scholarly knowledge. In *Proceedings of the 10th International Conference on Knowledge Capture*. 243–246.

[9] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web* 6, 2 (2015), 167–195.

[10] Florian Mai, Lukas Galke, and Ansgar Scherp. 2018. Using deep learning for title-based semantic subject indexing to reach competitive performance to full-text. In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries*. 169–178.

[11] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[12] Francesco Osborne and Enrico Motta. 2012. Mining semantic relations between research areas. In *International Semantic Web Conference*. Springer, 410–426.

[13] Jan Portisch, Michael Hladik, and Heiko Paulheim. 2020. KGvec2go–Knowledge Graph Embeddings as a Service. *arXiv preprint arXiv:2003.05809* (2020).

[14] Pushpankar Kumar Pushp and Muktabh Mayank Srivastava. 2017. Train once, test anywhere: Zero-shot learning for text classification. *arXiv preprint arXiv:1712.05972* (2017).

[15] Petar Ristoski and Heiko Paulheim. 2016. Rdf2vec: Rdf graph embeddings for data mining. In *International Semantic Web Conference*. Springer, 498–514.

[16] Petar Ristoski and Heiko Paulheim. 2016. Semantic Web in data mining and knowledge discovery: A comprehensive survey. *Journal of Web Semantics* 36 (2016), 1–22.

[17] Angelo A Salatino, Francesco Osborne, Thiviyan Thanapalasingam, and Enrico Motta. 2019. The CSO classifier: Ontology-driven detection of research topics in scholarly articles. In *International Conference on Theory and Practice of Digital Libraries*. Springer, 296–311.

[18] Moritz Schubotz, Philipp Scharpf, Olaf Teschke, Andreas Kühnemund, Corinna Breitinger, and Bela Gipp. 2020. AutoMSC: Automatic Assignment of Mathematics Subject Classification Labels. In *International Conference on Intelligent Computer Mathematics*. Springer, 237–250.

[19] Rima Türker, Lei Zhang, Mehwish Alam, and Harald Sack. 2020. Weakly Supervised Short Text Categorization Using World Knowledge. In *International Semantic Web Conference*. Springer, 584–600.

[20] Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. *arXiv preprint arXiv:1909.00161* (2019).