

Understanding Class Representations: An Intrinsic Evaluation of Zero-Shot Text Classification

Fabian Hoppe^{1,2}, Danilo Dessì^{1,2} and Harald Sack^{1,2}

¹FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, Germany

²Karlsruhe Institute of Technology, Institute AIFB, Germany

Abstract

Frequently, Text Classification is limited by insufficient training data. This problem is addressed by Zero-Shot Classification through the inclusion of external class definitions and then exploiting the relations between classes seen during training and unseen classes (Zero-shot). However, it requires a class embedding space capable of accurately representing the semantic relatedness between classes. This work defines an intrinsic evaluation based on greater-than constraints to provide a better understanding of this relatedness. The results imply that textual embeddings are able to capture more semantics than Knowledge Graph embeddings, but combining both modalities yields the best performance.

Keywords

Zero-Shot Learning, Text Classification, Class Representation, Embedding Model, Intrinsic Evaluation

1. Introduction

Managing, finding and exploring information from textual data is frequently done by applying Text Classification. As such classifying texts according to a predefined taxonomy is a key task in Natural Language Processing and Information Retrieval. For example within the scholarly domain classification supports researchers to retrieve relevant articles for their research by categorizing huge document collections according to a given schema. In order to achieve this goal Text Classification requires a certain understanding of the information presented in natural language texts. Typically, supervised classifiers obtain this understanding by statistically analyzing features of large training sets. In the last decade, the required amount of task-specific training data was to some extent reduced by pre-training large language models on the cloze task or similar self-supervised tasks on large unlabeled corpora. Nevertheless, this still requires a sufficient number of training examples for each class aside from the taxonomy itself. Moreover, collecting training data is costly, because human experts have to label each document and, therefore, this time consuming process is not feasible for many classification tasks. Classes might follow a long-tail distribution, i.e., there are many classes with only a few examples, or the taxonomy might get frequently extended by emerging new classes. One way of coping with insufficient training data is to exploit external knowledge about given classes, mimicking how


Workshop on Deep Learning for Knowledge Graphs (DL4KG@ISWC2021), October 25, 2021

✉ fabian.hoppe@fiz-karlsruhe.de (F. Hoppe); danilo.dessi@fiz-karlsruhe.de (D. Dessì); harald.sack@fiz-karlsruhe.de (H. Sack)

ORCID 0000-0002-7047-2770 (F. Hoppe); 0000-0003-3843-3285 (D. Dessì); 0000-0001-7069-9804 (H. Sack)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

humans learn to classify documents; this is what Zero-shot Text Classification aims to achieve. More precisely, instead of classifying documents by comparing them to other documents of a specific class, Zero-shot Classification exploits known relations between classes. Consequently, it does not require training data for all classes of a taxonomy and makes it possible to predict classes, which are not seen during training. This is achieved by transferring information from known classes to classes that are not seen or are not sufficiently represented. Practically, Zero-shot Classification aligns a class embedding space generated by using external knowledge with a document embedding space, and uses the interrelation between seen and unseen classes in their embedding space to obtain a representation for the unseen class in the aligned vector space. The actual classification is performed by applying a distance metric to find the classes most similar to the given documents. By definition the performance of such a classifier relies heavily on the utilized external knowledge of the classes and how it is represented in the class embedding space.

Many models focus on textual data as external knowledge and uses language models to create a vector space for these classes [1, 2]. More recently, efforts are made to include explicit knowledge by utilizing knowledge graphs, such as ConceptNet [3] or DBpedia [4]. However, improving the state of the art by providing the best suitable external knowledge using the best suitable embedding model requires a better understanding on how well the considered external knowledge is actually encoded in the embedding space. Certainly, this understanding is difficult to obtain by using extrinsic evaluation of the whole model, because these results are heavily influenced by other factors, e.g., the used document representations as well as the model used to perform the alignment of both spaces. Consequently, a sound judgment of the usefulness of the class representations on their own requires a more detailed review. Therefore, this paper proposes an intrinsic evaluation which investigates the class embedding space independently. The main requirement of class representations is an accurate encoding of the interrelations between classes, because these relations are exploited by Zero-shot Classification. It means that related or similar classes should be represented close to each other and unrelated or dissimilar classes should be further apart. Based on this intrinsic evaluation the most common external knowledge sources and the related embedding models are investigated on the example of the arXiv category taxonomy¹. This investigation uses a newly created gold standard which is made publicly available together with the source code for the detailed investigation of class representations². Thereby, this paper aims to facilitate further research into the used external knowledge for class representations and the applied embedding models and improve Zero-shot Text Classification.

In summary, the contribution of this work is threefold:

- An evaluation of class embeddings for Zero-shot Classification is proposed.
- A new gold standard based on the arXiv category taxonomy is provided.
- Common class representations are evaluated and insights to obtain better classification results are discussed.

The reminder of this paper is organized as follows. Section 2 discusses the relevant related work in Zero-shot Text Classification and the evaluation of representations. Afterwards, Section 3

¹https://arxiv.org/category_taxonomy

²<https://github.com/ISE-FIZKarlsruhe/IntrinsicEvaluationOfClassRepresentations>

introduces common external knowledge sources and embedding models and describes the intrinsic evaluation. Section 4 reports and discusses the results of this intrinsic evaluation on arXiv class representations. Finally, Section 5 concludes the paper and outlines future directions.

2. Related Work

One of the first works exploring Zero-shot Text Classification is the dataless classification framework [5]. The authors use Explicit Semantic Analysis as a latent representation for documents and classes, which avoids the alignment of both spaces. The class representations are generated based on the class names as external knowledge resource. This research got extended by considering several neural network based word embeddings, such as Word2Vec [1]. Due to the omitted alignment step these approaches do not require any training data. Consequently, similar models are deployed, where no training data is available, like categorizing German archival documents [6]. Recently, contextualized embedding models are utilized by reformulating the classification task as a textual entailment problem [2]. This approach continues to represent classes by only using the class name in combination with large, self-supervised language models as external knowledge. However, as shown by [7] for Zero-shot Image Classification already a basic classifier using a linear transformation to align image and GloVe class embeddings can reach state-of-the-art performance by considering Wikipedia articles as additional external knowledge for the given classes. Recent studies in Zero-shot Text Classification also extend their model by including explicit knowledge sources, like Knowledge Graphs (KGs), to generate suitable class representations. For example, in [3] the ConceptNet KG is used to extract explicit relations between words. Another work investigates the usage of DBpedia RDF2Vec embeddings to represent arXiv categories [4]. The authors investigate whether text embedding combined with KG embeddings would provide better class representations for a Zero-shot Classification.

Regardless of the increased focus on class representations state-of-the-art Zero-shot models still rely on costly extrinsic evaluations to judge these representations, i.e., the quality of embedding models is assessed by only considering the performance of the task itself. Thus, it requires more computational power and is not suited to provide a better understanding on the representations. Other tasks utilizing embedding models apply intrinsic evaluation to gain a better understanding of the applied embedding model. Instead of considering the whole system, these evaluations focus on specific intermediate subtasks. Due to their widespread use especially the semantics of word embeddings is extensively investigated based on multiple intrinsic evaluations [8]. The most popular are word semantic similarity and the word analogy task. The word semantic similarity task evaluates the correlation between similarities of embedding pairs to human labelled semantic similarity. This provides a direct evaluation of the relations between words. However, the task is quite subjective and depends on the relations considered for the human labels, e.g. a football and a globe are both spheres and would be similar if the shape property is most important for the given task. The word analogy task (a is to \hat{a} as b is to \hat{b}) tries to reduce this influence by considering only a given relation which is defined by the first pair (a, \hat{a}) . Unfortunately, it requires a careful selection of word pairs so that the relation makes sense which increases the labelling effort. After all, both tasks provide a better understanding of the created embedding space by direct evaluation of semantic relatedness. The

importance of such an understanding is recently highlighted in [9]. The detailed investigation of KG embeddings by means of clustering and classification experiments raised doubt about the semantic capabilities of common used models. Nevertheless, the literature describes several pitfalls of intrinsic evaluation that need to be considered. Most commonly (i) it might fail to relate intrinsic and extrinsic evaluation [10], (ii) human annotators might introduce biases based on their background [11], (iii) low inter-annotator agreement [12]. However, intrinsic evaluation is crucial to understand the models and to know what to improve.

3. Class Representations

Class representations play a vital role for Zero-shot Text Classification. They encode external knowledge which is exploited to accomplish the classification of unseen or not sufficiently represented classes. Therefore, the understanding of the relevant external knowledge and the related embedding models is important. Before discussing how intrinsic evaluation provides a better understanding of these class representations, the most common models are briefly presented. Currently, two modalities of external knowledge are considered for the generation of class representations in Zero-shot Text Classification: textual data and Knowledge Graph-based data. Both modalities use their own embedding models.

3.1. Textual Representations

Typically, classes are described with natural language text to support the annotation by human experts as well as provide an easily understandable definition of these classes to the users. Therefore, this external knowledge is frequently available without any additional effort. Considering that this data informally defines the classes for the users it ranges from names of common, well-known classes to more detailed descriptions of specific classes depending on the classification task. In addition to these commonly available texts, classes can be linked to resources providing more background knowledge. If these links are not already part of the taxonomy, they can be retrieved by manual or (semi-)automatic entity-linking using the available texts. Even for large taxonomies this additional effort is dwarfed by the task of providing thousands of training examples for each class. One commonly used external resource is Wikipedia. As the largest encyclopedia it provides background knowledge for many classes in a broad set of domains.

The textual data is embedded into a vector space by using pre-trained language models. These models learn latent representations of words based on a word prediction task. As such they encode the usage of each word into a vector space. Words that frequently occur in the same context are represented as similar vectors. **Word2Vec** [13] is one example of such an embedding model. It provides two settings. The *skip gram* setting computes the embedding of a word given its surrounding context, whereas the *CBOW* setting computes the embeddings of a context, given a word. In the Word2Vec model a word is always represented by one single vector. This poses two challenges. First, a longer text sequence requires additional normalization steps and secondly, the ambiguity of a word considering the larger context cannot be encoded. Frequently, the normalization falls back to averaging over all word embeddings. However, especially for larger texts this results in representations containing less semantics. Both problems are

addressed by contextualized word embeddings such as **BERT** [14]. BERT uses masked language models and transformers to predict missing words. Due to its architecture these contextualized word embeddings provide also context embeddings which can be used as representations for larger word sequences. The semantics encoded into both models is based on the empirical usage of the considered words in a training corpus. One special kind of textual embedding model is the **Wikipedia2Vec** [15] approach. Instead of learning only word representations in a similar setting as Word2Vec it jointly learns representations for Wikipedia entities based on predicting neighbours in the Wikipedia link graph. The link graph connects entities that are mentioned in the corresponding articles of other entities and thereby provides additional external knowledge for the representations. Due to the exploited graph structure this is similar to the second modality which is frequently exploited: Knowledge Graph Representations.

3.2. Knowledge Graph Representations

Instead of implicitly retrieving semantics by considering the statistics of a large corpus, Knowledge Graphs provide explicit semantics by defining entities and relations between those entities. Providing explicit knowledge enables a more precise definition of the class representations and thereby improves the understanding. However, only a small number of taxonomies already maintain suitable references to KGs. This makes the entity-linking step described to retrieve additional textual data also necessary for KGs. Similar to the considered corpus for textual representations KG representations depend significantly on which KGs are utilized as external knowledge source. On one side the large nodes of the Linked Open Data cloud, like DBpedia or Wikidata, could be utilized or subject-specific KGs which provided more detailed domain knowledge, but are not available for many domains and tend to contain less knowledge.

The linked entities can be transformed into a low dimensional vector by a wide range of KG embedding models. In the scope of this paper three KG models are considered. **TransE** [16] represents entities of a KG by defining relations of a KG as translations in the embedding space from the head to the tail of triples. More specifically, given a triple $\langle h, r, t \rangle$ the model is trained to build a vector space where $t \approx h \oplus r$ holds. The training is performed by corrupting triples to generate negative samples i.e., the tail (or head) of the triple is substituted by another entity; the model is optimized to distinguish between corrupted and non-corrupted triples. In doing so, given a triple $\langle h, r, t \rangle$ and a corrupted triple $\langle h, r, t' \rangle$ from the generated embeddings space, $h \oplus r$ should be closer to t than t' . In opposition to that **TransR** [17] represents entities and relations in different embedding spaces. The assumption behind this model is that entities and relations in KGs are different objects and, thus, they need two distinct representations. Given a triple $\langle h, r, t \rangle$ the model uses a translation matrix M_r to move h and t from the entity space to the relation space. The score function used by the model is given in equation $f_r(h, t) = \|hM_r + r - tM_r\|_2^2$. **RDF2Vec** [18] is a model which adapts the *Word2Vec* algorithm to graph representations. First, it creates sequences of entities and relations by performing random walks on the graph in order to build sequences that can be used as text sentences. Then, the *skip gram* or the *CBOV* methodologies are applied to build embedding representations of entities and relations.

3.3. Understanding Representation Spaces

In Zero-shot Text Classification the class representations are mapped into a shared class-document space to perform the classification by utilizing distance / similarity metrics. This mapping is learned by aligning the subset of classes with sufficient training data and then applied to unseen class representations. Consequently, the classification is based on the assumption that the relevant semantic relations between classes are encoded in the class embedding space. Therefore, a better understanding of the class representations is provided by evaluating the semantic relatedness between the classes. The straightforward way of analysing the semantic relatedness compares the vector similarity between a pair of embeddings to a human assigned label. This is the same process as for the word semantic similarity task. However, the human labels are highly subjective for class representations as well and require a detailed description on how specific relations should be quantified. These problems can be partially circumvented by comparing the similarity of two pairs. Such a comparison can be defined based on a triple of classes $\langle Anchor, A, B \rangle$ and a label indicating if class A or class B is more similar to the *Anchor* class. The intrinsic evaluation of an embedding space Θ predicts the label by checking if the constraint $cosine\ similarity(\Theta(Anchor), \Theta(A)) > cosine\ similarity(\Theta(Anchor), \Theta(B))$ is true and can be analyzed by precision, recall and f-measure in a binary classification setting.

Unfortunately, not all class combinations share some kind of relation among them, making it necessary to select only triples where such a relation exists at least for one pair. Additionally, in some cases the semantic similarity could be equal between the pairs. Consequently, the human labels need to include a third value to identify the cases where it is not possible to decide. The evaluation can either filter these cases which reduce the available labels but does not require any additional hyperparameter or introduce a threshold as well as a minimal similarity to predict this class as well and extend the binary classification setting to a multi-class setting.

4. Evaluation

The presented class representations are evaluated based on the proposed intrinsic evaluation on the example of the *arXiv.org* computer science classes. ArXiv manages its many published scholarly articles by categorizing them according to a taxonomy. Thereby, it represents a typical multi-class multi-label classification.

4.1. Gold Standard: ArXiv Classes

The arXiv taxonomy provides for all classes a descriptive name and a brief description. Additionally, the arXiv classes are manually mapped to the most suitable DBpedia entity, which enables retrieving the Wikipedia abstracts as textual description of the given classes. During this step classes like *General Literature* and classes without a suitable DBpedia mapping are filtered. This leaves overall 31 classes. In order to investigate the influence of explicit domain knowledge the classes are also mapped to AI-KG [19], a KG generated based on scholarly articles from the computer science domain. An overview of the amount of data available for the arXiv classes is given in Table 1.

Table 1

Overview of the external data with min, average and max number of tokens for text attributes and number of triples containing the mapped entity for the KGs.

Attribute	Size	Example
	min; average; max	
Name	1; 2.5; 5	Artificial Intelligence
Wikipedia abstract	22; 209.2; 494	Artificial intelligence (AI) is intelligence demonstrated by machines, as opposed to the natural intelligence displayed by humans and animals. Leading AI textbooks [...]
DBpedia	5; 1772.5; 7766	<i>dbr:Artificial_intelligence dcterms:subject</i> <i>dbr:Emerging_technologies</i> . [...]
AI-KG	3; 650.8; 4091	<i>aikg:artificial_intelligence rdf:type aikg-o:Task</i> . [...]

The evaluation utilize mostly pre-trained embeddings models. The textual embeddings use the skip-gram Word2Vec model³ built on the Google News dataset, the pre-trained BERT model⁴ built on BookCorpus and the English Wikipedia and finally skip-gram Wikipedia2Vec⁵. The KG entities for DBpedia use the models pre-trained for [20] which are online available⁶. Based on this code also the relevant AI-KG embeddings are trained. Additionally, the multi-class setting uses half of the standard derivation of all class similarities as threshold and the 10th percentile as minimum value.

The gold standard is created by random subsampling all combinations of the 31 arXiv classes and is manually annotated by 11 experts from the computer science domain. The annotators were instructed by a brief task description and were provided the available textual data from arXiv (descriptive name, brief description). Overall the experts labelled 3,000 triples with 5 votes for each triple. However, the analysis of intra- and inter-annotator agreement indicates only a small reliability of the whole dataset. The intra-annotator agreement is calculated based on 20 triples which were labeled twice by every annotator. The Cohen’s κ coefficient for this agreement ranges from 0.14 to 0.9 with an average of 0.49. The inter-annotator agreement is measured by the averaged Cohen’s κ coefficient of the 5 votes provided for each triple. It is 0.22. The Krippendorff’s α coefficient of 0.21 confirms this low reliability. Evidently, random subsampling includes many controversial triples, where the label depends on minor differences in the mental model of the individual annotators. In order to create a reliable gold standard these controversial triples are filtered out. A triple is considered as controversial if more then one of the votes disagrees with the other votes. After this step the gold standard contains 1,266 triples with 300 *A* labels, 354 *B* labels and 612 triples where no decision was possible.

4.2. Results and Discussion

The results are reported in Table 2. For the binary classification precision, recall and F-measure are presented and for the multi-class classification only the micro F-measure is considered,

³<https://code.google.com/archive/p/word2vec/>

⁴<https://huggingface.co/bert-base-cased>

⁵<https://wikipedia2vec.github.io/wikipedia2vec/pretrained/>

⁶<https://github.com/nheist/KBE-for-Data-Mining>

because the actual class distribution is arbitrary and by definition precision, recall and F-measure are equal in the micro setting. However, both classification settings provide similar results. This indicates that unrelated triples and equally similar pairs are not a special case, which suggests that binary classification with less hyperparameter is sufficient for the intrinsic evaluation.

Overall, the Wikipedia2Vec embeddings align best with the actual semantic similarities, followed by Word2Vec generated from names and BERT using Wikipedia abstracts. All KG embeddings could not reach this performance. Overall, no model is able to reach human-level performance. However, the human results presented are the average of the same annotations used for creating the gold standard and due to this biased. Independent labels would be below these results.

Table 2

Evaluation in terms of Precision, Recall, and F-measure.

Model	Attribute	Binary			Multi-Class
		Precision	Recall	F-measure	Micro F-measure
Word2Vec	Name	0.668	0.757	0.709	0.484
Word2Vec	Wiki abstract	0.682	0.65	0.666	0.478
BERT	Name	0.591	0.593	0.592	0.379
BERT	Wiki abstract	0.658	0.727	0.691	0.495
Wikipedia2Vec	Wiki entity	0.738	0.74	0.739	0.563
TransR	DBpedia	0.548	0.573	0.56	0.415
TransR	AI-KG	0.498	0.55	0.523	0.439
TransE	DBpedia	0.508	0.513	0.511	0.397
TransE	AI-KG	0.501	0.597	0.545	0.42
RDF2Vec	DBpedia	0.496	0.573	0.532	0.356
Human Annotator		0.947	0.853	0.881	0.86

The most apparent insight this analysis provides is the difference between both modalities. KG embeddings in general lack behind textual embeddings. With only small differences between the embedding models and considering that RDF2Vec exploits a similar methodology as Word2Vec this can be explained by the smaller amount of training data. Even large KGs like DBpedia only provide a few thousand triples for each class. A text corpus with significant more mentions of these classes is able to provide a better semantic model.

However, the comparison between Word2Vec and BERT, where BERT represents the larger model trained with more data, shows that the model size is not the only relevant factor. Especially, the BERT model relies on task-specific fine-tuning, which is not possible for unseen classes in Zero-shot Classification. These results are coherent with the extrinsic evaluation of Word2Vec and BERT in [2] where a fine-tuned BERT model performs on a similar level as untrained Word2Vec for unseen classes. Interestingly, the results show the main drawback of a Word2Vec model, too. If larger texts, like the Wikipedia abstracts, are normalized by basic averaging all embeddings get more similar and the performance decreases. On the other hand contextualized models as BERT perform worse without the context.

A less pronounced difference in the results is given between the KG embedding models and the used KG. TransE returns slightly better results on the domain specific AI-KG, but

TransR performs better on a larger dataset with more heterogeneous relations (DBpedia). That illustrates that TransE struggles with heterogeneous relations and therefore benefits from more domain-specific knowledge.

Overall, Wikipedia2Vec as a hybrid model using text and graph embeddings provides the best semantic relatedness. Such models seem to extract the semantics from large textual corpora and use the entity relations to emphasis or add important semantic relations. Thereby, it combines advantages of both modalities.

5. Conclusion

This paper describes an intrinsic evaluation, which provides a better understanding of the class vector space for Zero-shot Text Classification by checking human defined greater-than constraints between classes. Therefore, common embeddings models using textual data and KGs to define classes are generated for the computer science arXiv subset and evaluated with a created gold standard. This investigation shows that textual embeddings are able to extract more implicit knowledge compared to the explicit knowledge provide by KGs and that extending textual embeddings with (Knowledge) Graph-based information is able to capture semantic relatedness better than single modalities.

This line of research can be extended by an empirical study on the correlation between the performance of unseen classes and the semantic relatedness to confirm the theoretical argument for the intrinsic evaluation. Additionally, more embedding models using both modalities should be evaluated with respect to semantic relatedness. Another ongoing effort is the extension of the presented dataset considering more triples and a larger pool of domain experts. This way the intrinsic evaluation of class representations is able to facilitate further research, and thereby improve Zero-shot Text Classification.

References

- [1] Y. Song, D. Roth, On dataless hierarchical text classification, in: Proceedings of the 28th AAAI conference on Artificial Intelligence, 2014, p. 1579–1585.
- [2] W. Yin, J. Hay, D. Roth, Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 2019, pp. 3914–3923.
- [3] J. Zhang, P. Lertvittayakumjorn, Y. Guo, Integrating semantic knowledge to tackle zero-shot text classification, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 1031–1040.
- [4] F. Hoppe, D. Dessi, H. Sack, Deep learning meets knowledge graphs for scholarly data classification, in: Companion Proceedings of the Web Conference 2021, 2021, pp. 417–421.
- [5] M. W. Chang, L. Ratinov, D. Roth, V. Srikumar, Importance of semantic representation: Dataless classification, 27th AAAI conference on Artificial Intelligence (2008).

- [6] F. Hoppe, T. Tietz, D. Dessì, N. Meyer, M. Sprau, M. Alam, H. Sack, The challenges of German archival document categorization on insufficient labeled data, in: Proceedings of the Third Workshop on Humanities in the Semantic Web, co-located with 15th Extended Semantic Web Conference, 2020, pp. 15–20.
- [7] S. Bujwid, J. Sullivan, Large-scale zero-shot image classification from rich and diverse textual descriptions, in: Proceedings of the Third Workshop on Beyond Vision and LANguage: inTEgrating Real-world kNowledge (LANTERN), 2021, pp. 38–52.
- [8] A. Bakarov, A survey of word embeddings evaluation methods, arXiv (2018).
- [9] N. Jain, J.-C. Kalo, W.-T. Balke, R. Krestel, Do embeddings actually capture Knowledge Graph semantics?, in: 18th Extended Semantic Web Conference, 2021, pp. 143–159.
- [10] B. Chiu, A. Korhonen, S. Pyysalo, Intrinsic evaluation of word vectors fails to predict extrinsic performance, in: Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, 2016, pp. 1–6.
- [11] F. F. Liza, M. Grześ, An improved crowdsourcing based evaluation technique for word embedding methods, in: Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, 2016, pp. 55–61.
- [12] F. Hill, R. Reichart, A. Korhonen, Simlex-999: Evaluating semantic models with (genuine) similarity estimation, Computational Linguistics (2015).
- [13] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv (2013).
- [14] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, arXiv (2018).
- [15] I. Yamada, A. Asai, J. Sakuma, H. Shindo, H. Takeda, Y. Takefuji, Y. Matsumoto, Wikipedia2Vec: An efficient toolkit for learning and visualizing the embeddings of words and entities from Wikipedia, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, 2020, pp. 23–30.
- [16] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, Advances in neural information processing systems (2013).
- [17] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for Knowledge Graph completion, in: 29th AAAI conference on Artificial Intelligence, 2015, pp. 2181–2187.
- [18] P. Ristoski, H. Paulheim, Rdf2Vec: RDF graph embeddings for data mining, in: International Semantic Web Conference, 2016, pp. 498–514.
- [19] D. Dessì, F. Osborne, D. R. Recupero, D. Buscaldi, E. Motta, H. Sack, AI-KG: an automatically generated knowledge graph of artificial intelligence, in: International Semantic Web Conference, 2020, pp. 127–143.
- [20] J. Portisch, N. Heist, H. Paulheim, Knowledge Graph embedding for data mining vs. Knowledge Graph embedding for link prediction—two sides of the same coin?, Semantic Web Journal (2021).