Using ontology embeddings with deep learning architectures to improve prediction of ontology concepts from literature

Pratik Devkota[1], Somya D. Mohanty[2] and Prashanti Manda[1]

[1]Informatics and Analytics, University of North Carolina at Greensboro
[2]United Healthcare

Abstract

Natural language processing methods powered by deep learning have been well-studied over the past years for the task of automated ontology-based annotation of scientific literature. Many of these approaches focus solely on learning associations between text and ontology concepts and use that to annotate new text. However, a great deal of information is embedded in the ontology structure and semantics. Here, we present deep learning architectures that learn not only associations between text and ontology concepts but also the structure of the ontology. Our experiments show that creating architectures that are capable of learning the structure of the ontology result in enhanced annotation performance.

Keywords

natural language processing, gene ontology, deep learning, ontology annotation, ontology embeddings

1. Introduction

Biological ontologies are widely used for representing biological knowledge across a wide range of sub-domains ranging from gene function to clinical diagnoses to evolutionary phenotypes [1, 2, 3]. While the ontologies provide the necessary structure and concepts, the real benefits of the ontologies can be reaped only when knowledge in scientific literature is represented using these ontologies through annotation. The scale and pace of scientific publishing demands sophisticated, fast, and most importantly, automated ways of processing scientific literature to annotate relevant pieces of text with ontology concepts [4].

Natural Language Processing (NLP) techniques beginning with lexical analysis, standard machine learning approaches, and of late, powered by deep learning models have made big strides in this area [5, 6, 7, 8, 9, 10]. Most NLP approaches for automated ontology annotation treat the task as that of named entity recognition where relevant entities are identified and associated with snippets of text. However, ontology based annotation is different from named entity recognition in that there is a great amount of information embedded in the structure and

Proceedings of the International Conference on Biomedical Ontologies 2023, August 28th-September 1st, 2023, Brasilia, Brazil

p_devkota@uncg.edu (P. Devkota); mohanty.somya@gmail.com (S. D. Mohanty); p_manda@uncg.edu (P. Manda)
0000-0001-5161-0798 (P. Devkota); 0000-0002-4253-5201 (S. D. Mohanty); 0000-0002-7162-7770 (P. Manda)

semantics of an ontology whereas generic entities can be independent objects. Knowledge of the ontological structure and relationships is a crucial part of biological annotation when performed by a human curator. It is therefore imperative to develop NLP models that are cognizant of the ontological hierarchy and can effectively incorporate it into the prediction mechanism for improved ontology concept recognition.

The automated annotation models previously developed by this team [11, 8, 12, 10, 9] have shown good accuracy in recognizing ontology concepts from text. In these studies our focus was to teach the models to learn associations between text and ontology concepts found in the gold standard corpus and use that knowledge to create new annotations. In a few studies, we experimented with different techniques of using the ontology structure as one of the inputs in a bid to improve annotation performance [10, 8, 9]. In some cases, these systems are able to predict the same ontology concept as the ground truth in the gold standard data achieving perfect accuracy. Incorporating ontology structure was a bid to improving partial accuracy in cases where the model does not achieve a perfect match to the actual annotation. Our hypothesis was that having knowledge of the ontology structure would enable the model to choose a closely related/semantically similar concept to the actual annotation thereby improving overall annotation performance as evaluated by semantic similarity.

Our goal in this study is to develop deep learning architectures that learn not only patterns in text but also the ontology structure. Our hypothesis is that the process of learning the ontology structure would in turn improve prediction of annotations. Deep learning models learn patterns in text and annotations from a gold standard corpus and similarly, we need to provide a gold standard representation of the ontology structure so the models can learn to predict the ontology structure.

In this study, we use graph embeddings for representing the ontology structure. These graph embeddings are used as a reference and reinforcement tool for the model as it learns to predicts the ontology structure. Semantic embedding of large knowledge graphs has been long used successfully for predictive tasks including natural language processing [13]. In recent years, these semantic embeddings have been extended to OWL ontologies resulting in approaches that can create embeddings for ontology concepts that effectively represent the structure and semantics of the ontology [13, 14]. These embedding algorithms translate ontologies represented as directed acyclic graphs into a vector space where the structure and the inherent semantics of the graph are preserved [15].

There are several approaches for learning ontology embeddings [16, 17] each with different strengths. The approaches differ based on whether the ontology is directed, weighted, if it dynamically changes over time, and the approach for learning the network [17]. In this study, we selected Node2Vec [14] for learning ontology embeddings from the Gene Ontology since it is widely used in literature for this task [17].

We use the Colorado Richly Annotated Full Text Corpus (CRAFT) as a gold standard for training and testing the performance of our architectures [18]. CRAFT is a widely used training resource for automated annotation approaches. The current version of the CRAFT corpus (v4.0.1) provides annotations for 97 biological/biomedical articles with concepts from 7 ontologies including the GO.

We hypothesize that the added information gained from ontology embeddings can improve model performance in recognizing ontology concepts from scientific literature. We persent

two deep learning architectures and explore how the different architectures combined with the inclusion of ontology embeddings impacts annotation performance.

2. Related Work

The rise of deep learning in the areas of image and speech recognition has translated into text-based problems as well. Preliminary research has shown that deep learning methods result in greater accuracy for text-based tasks including identifying ontology concepts in text [5, 19, 20, 21, 8]. These methods use vector representations that enable them to capture dependencies and relationships between words using enriched representations of character and word embeddings from training data [7].

Our initial foray into this area involved a feasibility study of using deep learning for the task of recognizing ontology concepts [11]. In a comparison of Gated Recurrent Units (GRUs), Long Short Term Memory (LSTM), Recurrent Neural Networks (RNNs), and Multi Layer Perceptrons (MLPs) along with a new deep learning model/architecture based on combining multiple GRUs, we found GRUs to outperform the rest. These findings indicated that deep learning algorithms are a promising avenue to be explored for automated ontology-based curation of data.

In 2020, we presented new architectures based on GRUs and LSTMs combined with different input encoding formats for automated annotation of ontology concepts [8]. We also created multi level deep learning models designed to incorporate ontology hierarchy into the prediction. Surprisingly, inclusion of ontology semantics via subsumption reasoning yielded modest performance improvement [8]. This result indicated that more sophisticated approaches to take advantage of the ontology hierarchy are needed.

Continuing this work, a 2022 study [12] presented state of the art deep learning architectures based on GRUs for annotating text with ontology concepts. We augmented the models with additional information sources including NCBI's BioThesauraus and Unified Medical Language System (UMLS) to augment information from CRAFT for increasing prediction accuracy. We demonstrated that augmenting the model with additional input pipelines can substantially enhance prediction performance.

Our next work explored a different approach to providing the ontology as input to the deep learning model [8]. Subsequently, we presented an intelligent annotation system [10] that uses the ontology hierarchy for training and predicting ontology concepts for pieces of text. Here, we used a vector of semantic similarity scores to the ground truth and all ancestors in the ontology to train the model. This representation allowed the model to identify the target GO term followed by "similar" GO terms that are partially accurate predictions. We showed that our ontology aware models can result in a 2% - 10% improvement over a baseline model that doesn't use ontology hierarchies.

Our most recent contribution presented a method called Ontology Boosting [9]. A key component of this approach is to combine the prediction of the deep learning architectures with the graph structure of ontological concepts. Boosting amplifies the predicted probabilities of certain concept predictions by combining them with the model predictions of the candidate's ancestors/subsumers. Results showed that the boosting step can result in a substantial bump in prediction accuracy.

3. Methods

3.1. Generating ontology embeddings

We used the Node2Vec approach for generating ontology embeddings from the Gene Ontology.

The Node2Vec algorithm consists of two steps:

1. Conduct random walks from a graph or ontology to generate sentences which are a list of ontology concepts. Once all random walks are conducted, the set of all sentences makes the corpus which is a representation of the ontology.
2. The Word2Vec [22] algorithm is applied on the corpus to learn and generate embeddings for each concept in the ontology. These embeddings are low dimensional vector representations of ontology concepts.

These embeddings or feature vectors can be used in downstream tasks such as classification or natural language processing. in a downstream task such as node classification.

We set the weight of all edges to 1 for weighted random walks indicating that all edges are weighted equally. The length of random walk was set to 5 and the walk number set to 100. Dimensionality of embeddings was set to 128 and batch size is set to 50 and the model was trained for 2 epochs to learn the embeddings.

3.2. Deep Learning Architectures

Here, we present and test three sets of architectures:

1. Baseline
    - Tag only (*TO*)
    - Ontology Embedding only (*OEO*)
2. Cross-connected:
    - Tag to Ontology Embedding ($T- > OE$)
    - Ontology Embedding to Tag ($OE- > T$)
3. Multi-connected:
    - Embedding to Tag to Embedding

3.2.1. Baseline Architectures

We created two baseline architectures (Figure 1): Tag only (*TO*) and Ontology Embedding only (*OEO*). The *TO* architecture predicts tags/ontology IDs while the *OEO* architecture predicts ontology embeddings. The *TO* architecture has previously been presented in our prior work [10]. This architecture has been adjusted to create the *OEO* architecture that predicts ontology embeddings. Both baseline architectures consist of input pipelines, embedding/latent representations, and a deep learning model and produce either a probability vector of ontology IDs (*TO*) or an ontology embedding (*OEO*) as the output.

The baseline architectures use three inputs. Each word in a sentence from the CRAFT corpus is represented by three inputs - 1) token (X_{train}^{token}), 2) character sequence (X_{train}^{char}), 3) Parts-Of-Speech (POS) (X_{train}^{POS}).The token (X_{train}^{token}) input, is a sequential tensor consisting tokens, each

represented with a high dimensional one hot encoded vector. The character sequence (X_{train}^{char}) is also a sequential tensor consisting of character sequences present in a word/token. POS tags (X_{train}^{POS}) indicate the type of words in a sentence.

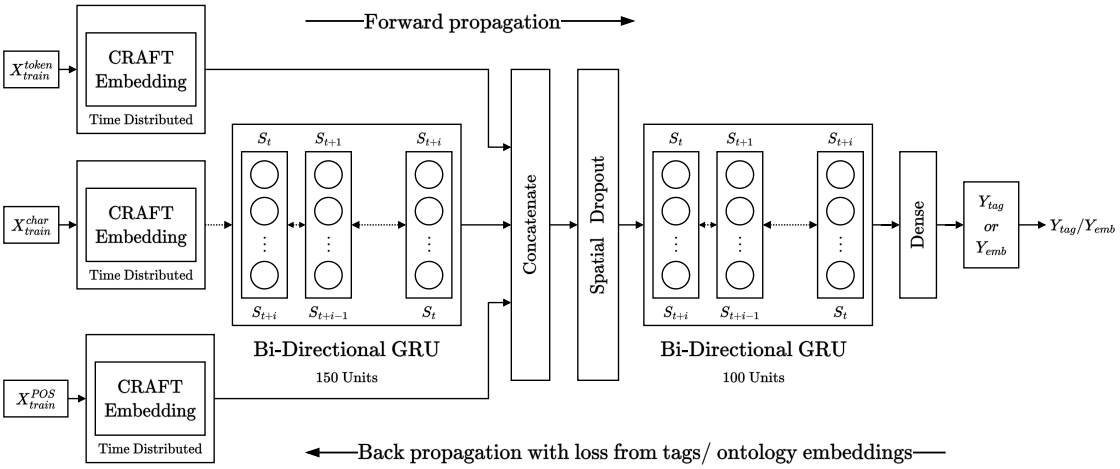

Figure 1: Tag-only (*TO*) and Ontology Embeddings only (*OEO*) baseline architectures. *TO* produces a tag/ontology ID (Y_{tag}) as output in the final block whereas *OEO* produces an ontology embedding (Y_{emb}). The architectures also differ in that is used during back propagation to compute gradient loss. *TO* uses tags while *OEO* uses Node2Vec ontology embeddings.

Embeddings are used to provide a compressed latent space representation for very high dimensional input components. For example, the one hot vectorization of an individual word has a dimensionality of 34,166 (vocabulary size). In order to represent them succinctly and with contextual representation, we use supervised embeddings created from the CRAFT corpus. Note that these embeddings are different from the ontology embeddings discussed above. These embeddings provide low dimensional representations of words in the training corpus and do not use the ontology in any way.

Both baseline architectures use a bi-directional gated recurrent model (Bi-GRU). The choice of Bi-GRU for the architectures was informed by several of our prior works where this model has consistently outperformed other models such as CNNs, RNNS, and LSTMs [8, 11]. Architecture hyper-parameters were evaluated using a grid search approach. We used Adam [23] as our optimiser for all of the experiments with a default learning rate of 0.001.

The two baseline architectures differ primarily because of what they produce as output and what they use during the propagation stages. In *TO*, the output is a tag/ontology ID where each word in the input data is mapped to either a GO annotation or a non-annotation. *TO* takes the hidden/learned representations of the input from the preceding layers of the network and applies softmax activation to produce a probability distribution over all possible ontology ids. The predicted vector output values and ground truth values are compared to compute sparse categorical cross entropy as loss, followed by backpropagation which involves computing the gradients of the loss with respect to the model's weights. The ontology ID with the highest probability is regarded as the prediction.

In contrast, *OEO* uses ground truth ontology embeddings generated using Node2Vec during back propagation and for computing the loss functions. The intuition is that providing ontology embeddings to the architecture during the propagation stages will enable it to get an understanding of the ontology structure and eventually enable it to make more accurate and intelligent predictions. The output of *OEO* is an ontology embedding. The predicted ontology embedding is compared to all ground truth ontology embeddings using cosine similarity calculation. The ground truth ontology embedding that is most similar to the predicted embedding is identified and the ontology ID associated with it is treated as the architecture's prediction. Accuracy metrics are then computed by comparing the predicted ontology ID to that in the CRAFT corpus.

3.3. Cross connected architectures

We developed two cross connected architectures: 1) Tag to Ontology Embedding ($T- > OE$) and 2) Ontology Embedding to Tag ($OE- > T$). Here we test if connecting the tag and ontology embedding architectures causing one to inform the prediction of the other would result in improved accuracy and if the direction of the connection matters. The $T- > OE$ architecture (Figure 2) has two different outputs, tags/ontology ids and ontology embeddings. The tag output (Y_{tag} in Figure 2) is concatenated with the output of the main Bi-GRU layer to give a higher dimensional vector output. The concatenation is then passed through dense layers to further learn the hierarchical representations of the ontology before generating an ontology embedding for each input token. This predicted ontology embedding is compared with the ground truth ontology embeddings learned using Node2Vec. Using cosine similarity as the loss function, loss is calculated and the gradients are backpropagated to adjust the model's weight for convergence.

In $OE- > T$, the ontology embedding output (Y_{emb} is concatenated with the output of the main Bi-GRU layer to give a higher dimensional vector output. The concatenation is then passed through dense layers before generating a tag for each input token. This predicted tag is compared with the ground truth tag in CRAFT. The loss is calculated and the gradients are backpropagated to adjust the model's weight for convergence. The $OE- > T$ architecture can be depicted by switching the Y_tag and Y_emb blocks as well as the two outputs in Figure 2.

Figure 3 presents an explanation of the $T- > OE$ cross-connected architecture on three example tokens. Cross connected architectures differ from the baseline architectures by producing both tags and ontology embeddings instead of one or the other. Here, we show that the training/inference is done on a sequence of tokens "vesicle", "formation", and "in" (which are parts of a sentence in the CRAFT corpus) as it is evaluated by the network. Each token is preprocessed to obtain the representative tensors – X^{token}_{train}, X^{char}_{train}, X^{POS}_{train} which are passed through embedding layers learned from CRAFT. The embedding of X^{char}_{train} is also passed via a Bi-GRU layer. All of the resulting values are concatenated to be processed via the main Bi-GRU layer. The output from 'Tag Dense Layer' is concatenated with the output of main 'Bi-GRU layer' and passed as input to the 'Ontology Embedding Dense Layer' where the model generates ontology embeddings for each of the input tokens.

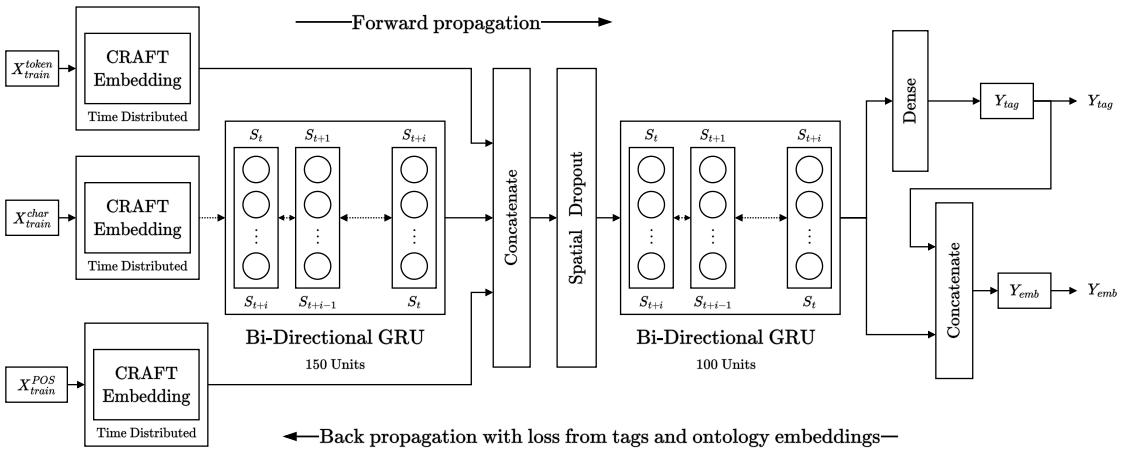

Figure 2: Tag to Embedding architecture ($T- > OE$). The tag output is further fed to the ontology embedding prediction block resulting in a better embedding prediction.

3.4. Multi-connected Architecture

The final architecture ($OE- > T- > OE$) explores if ontology embeddings can be improved iteratively by connecting a preliminary ontology embedding output to the tag output enabling improvements to the tag prediction. This predicted tag block is connected back to the ontology embedding block to urge further learning.

3.5. Performance Evaluation Metrics

We evaluate our architectures using a modified F1 score and semantic similarity [24]. Metrics such as F1 are designed for traditional information retrieval systems that either retrieve a piece of information or fail to do so (a binary evaluation). However, this is not a true indication of the performance of ontology-based retrieval or prediction systems where the notion of partial accuracy applies. A model might not predict the exact concept as a gold standard but might predict the parent or an ancestor of the ground truth as indicated by the ontology. Semantic similarity metrics [24] designed to measure different degrees of similarity between ontology concepts can be leveraged to measure the similarity between the predicted concept and the actual annotation to quantify the partial prediction accuracy. Here, we use Jaccard similarity [24] that measures the ontological distance between two concepts to assess partial similarity.

Since the majority of tags in the training corpus are non-annotations, the model predicts them with great accuracy. In order to avoid biasing the F1 score, we omit accurate predictions of non-annotations and focus instead on annotations only report a relatively conservative modified F1 score.

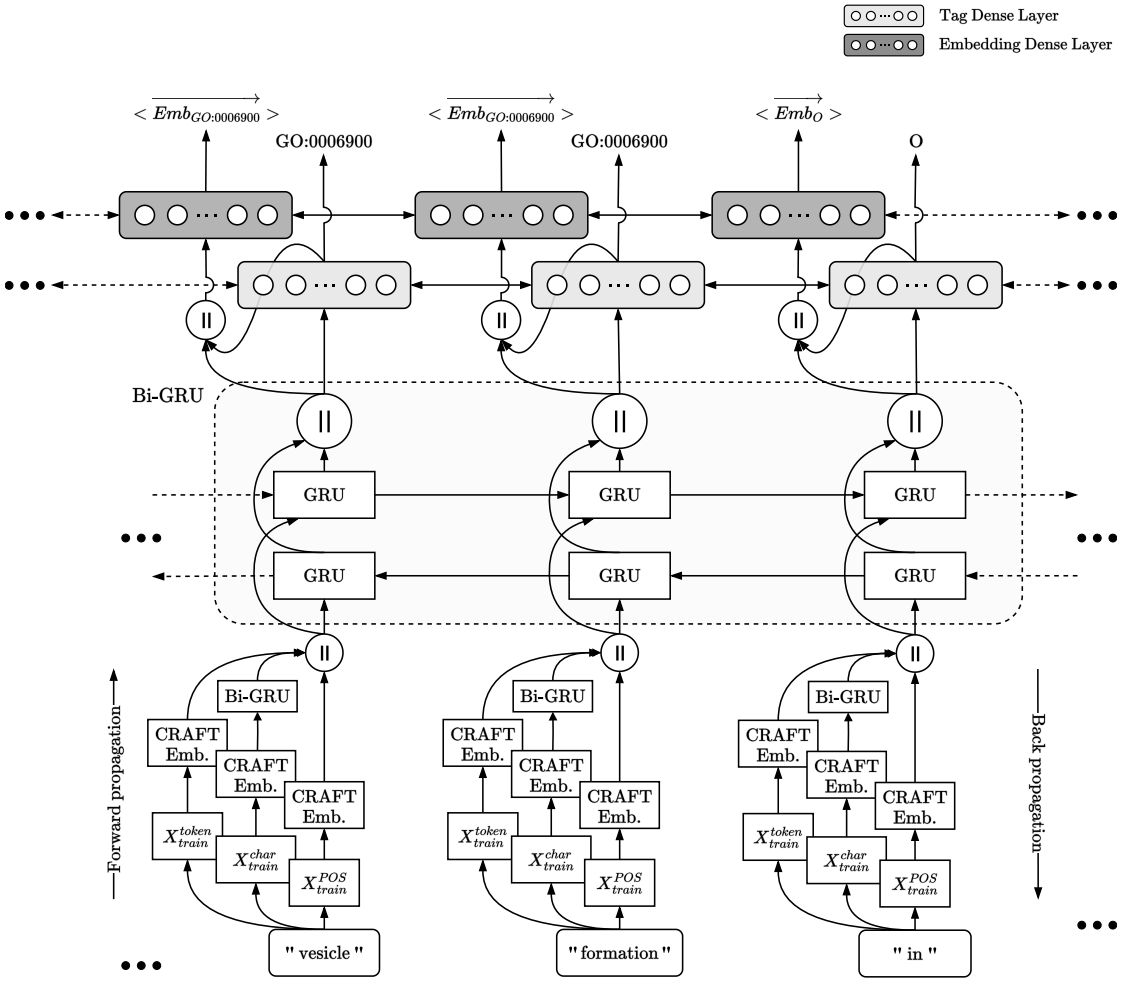

Figure 3: Illustration of the working of the $T- > OE$ architecture with an example sequence. The architecture produces two outputs - 1) a tag and 2) an ontology embedding.

4. Results and Discussion

The CRAFT v4.0.1 dataset contains 18689 annotations pertaining to 974 concepts from the three GO sub-ontologies across 97 articles. Table 1 provides further information of the coverage of GO terms in CRAFT.

The baseline tag-only architecture (*TO*) resulted in a 0.80 F1 and a 0.83 semantic similarity score. The baseline ontology embeddings only architecture (*OEO*) resulted in a 0.65 F1 and a 0.74 semantic similarity.

Among the two cross-connected architectures, we found that the Tag to Ontology Embedding architecture ($T- > OE$) substantially outperformed the $OE- > T$ architecture according to F1 and was able to achieve similar performance as measured by semantic similarity. This indicates that $T- > OE$ is better at generating exactly matching predictions resulting in high F1 and

semantic similarity. In contrast, $OE- > T$ performs better are generating semantically similar matches rather than exact matches leading to lower F1 than semantic similarity scores.

The $T- > OE$ architecture was able to improve upon OEO's prediction of ontology embeddings by 23% (F1) and 9.4% (semantic similarity). We observed relatively modest improvements to TO's tag prediction with 3.8% (F1) and 1.2% (semantic similarity).

Connecting ontology embedding output to the tag output ($OE- > T$) either did not improve on the embedding prediction (F1) or resulted in a slight improvement (semantic similarity). $OE- > T$ did produce improvements for tag prediction over the TO model by 3.7% (F1) and 1.2% (semantic similarity). The multi-connected architecture did poorly in comparison to the cross-connected architectures.

Overall, the results suggest that architectures that use ontology embeddings only without learning associations between text and annotations perform poorly. The other takeaway is that connecting tag predictions to the ontology embedding block ($T- > OE$) and letting embedding prediction learn from the predicted tag iteratively results in more robust architectures. The $T- > OE$ cross-connected architecture results in improved performance in predicting both tags and ontology embeddings across both metrics.

Table 1
Coverage of GO ontology concepts and annotations in the CRAFT corpus

GO sub-ontology	Concepts in ontology	Total annotations in CRAFT	Unique occurences in CRAFT
Biological Process (BP)	30490	18392	710
Cellular Component (CC)	4463	6976	241
Molecular Function (MF)	12257	464	5

Table 2
Performance metrics of the three sets of architectures measured by F1 and Jaccard semantic similarity

Architecture	**Ontology Embedding F1 Score**	**Ontology Embedding Similarity Score**	**Tag F1 Score**	**Tag Similarity Score**
Baseline Architectures				
Tag-only (TO)	-	-	0.80	0.83
Ontology Embedding Only (OEO)	0.65	0.74	-	-
Cross-connected Architectures				
Tag to Ontology Embedding ($T- > OE$)	**0.80**	**0.81**	**0.83**	0.84
Ontology Embedding to Tag ($OE- > T$)	0.64	0.75	0.83	**0.84**
Multi-connected Architecture				
$OE- > T- > OE$	0.78	0.80	0.82	0.83

Acknowledgments

This work is funded by a CAREER grant to Manda from the Division of Biological Infrastructure at the National Science Foundation of United States of America (#1942727).

References

[1] T. R. Dalmer, R. D. Clugston, Gene ontology enrichment analysis of congenital diaphragmatic hernia-associated genes, Pediatric research 85 (2019) 13–19.

[2] D. Lee, N. de Keizer, F. Lau, R. Cornet, Literature review of snomed ct use, Journal of the American Medical Informatics Association 21 (2014) e11–e19.

[3] R. C. Edmunds, B. Su, J. P. Balhoff, B. F. Eames, W. M. Dahdul, H. Lapp, J. G. Lundberg, T. J. Vision, R. A. Dunham, P. M. Mabee, et al., Phenoscape: identifying candidate genes for evolutionary phenotypes, Molecular biology and evolution 33 (2015) 13–24.

[4] W. Dahdul, T. A. Dececchi, N. Ibrahim, H. Lapp, P. Mabee, Moving the mountain: analysis of the effort required to transform comparative anatomy into computable anatomy, Database 2015 (2015).

[5] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer, Neural architectures for named entity recognition, arXiv preprint arXiv:1603.01360 (2016).

[6] M. R. Boguslav, N. D. Hailu, M. Bada, W. A. Baumgartner, L. E. Hunter, Concept recognition as a machine translation problem, BMC bioinformatics 22 (2021) 1–39.

[7] M. A. Casteleiro, G. Demetriou, W. Read, M. J. F. Prieto, N. Maroto, D. M. Fernandez, G. Nenadic, J. Klein, J. Keane, R. Stevens, Deep learning meets ontologies: experiments to anchor the cardiovascular disease ontology in the biomedical literature, Journal of biomedical semantics 9 (2018) 13.

[8] P. Manda, S. SayedAhmed, S. D. Mohanty, Automated ontology-based annotation of scientific literature using deep learning, in: Proceedings of The International Workshop on Semantic Big Data, SBD '20, Association for Computing Machinery, New York, NY, USA, 2020. URL: https://doi.org/10.1145/3391274.3393636. doi:10.1145/3391274.3393636.

[9] P. Devkota, S. D. Mohanty, P. Manda, Ontology-powered boosting for improved recognition of ontology concepts from biological literature (2023).

[10] P. Devkota, S. Mohanty, P. Manda, Knowledge of the ancestors: Intelligent ontology-aware annotation of biological literature using semantic similarity, Proceedings of the International Conference on Biomedical Ontology (2022).

[11] P. Manda, L. Beasley, S. Mohanty, Taking a dive: Experiments in deep learning for automatic ontology-based annotation of scientific literature, Proceedings of the International Conference on Biomedical Ontology (2018).

[12] P. Devkota, S. D. Mohanty, P. Manda, A gated recurrent unit based architecture for recognizing ontology concepts from biological literature, BioData Mining 15 (2022) 1–23.

[13] J. Chen, P. Hu, E. Jimenez-Ruiz, O. M. Holter, D. Antonyrajah, I. Horrocks, Owl2vec*: Embedding of owl ontologies, Machine Learning 110 (2021) 1813–1845.

[14] A. Grover, J. Leskovec, node2vec: Scalable feature learning for networks, in: Proceedings

of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016, pp. 855–864.
[15] M. Ou, P. Cui, J. Pei, Z. Zhang, W. Zhu, Asymmetric transitivity preserving graph embedding, in: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016, pp. 1105–1114.
[16] H. Cai, V. W. Zheng, K. C.-C. Chang, A comprehensive survey of graph embedding: Problems, techniques, and applications, IEEE transactions on knowledge and data engineering 30 (2018) 1616–1637.
[17] I. Makarov, D. Kiselev, N. Nikitinsky, L. Subelj, Survey on graph embeddings and their applications to machine learning problems on graphs, PeerJ Computer Science 7 (2021) e357.
[18] M. Bada, M. Eckert, D. Evans, K. Garcia, K. Shipley, D. Sitnikov, W. A. Baumgartner, K. B. Cohen, K. Verspoor, J. A. Blake, L. E. Hunter, Concept annotation in the craft corpus, BMC Bioinformatics 13 (2012) 161. URL: https://doi.org/10.1186/1471-2105-13-161. doi:10.1186/1471-2105-13-161.
[19] M. Habibi, L. Weber, M. Neves, D. L. Wiegandt, U. Leser, Deep learning with word embeddings improves biomedical named entity recognition, Bioinformatics 33 (2017) i37–i48.
[20] C. Lyu, B. Chen, Y. Ren, D. Ji, Long short-term memory rnn for biomedical named entity recognition, BMC bioinformatics 18 (2017) 462.
[21] X. Wang, Y. Zhang, X. Ren, Y. Zhang, M. Zitnik, J. Shang, C. Langlotz, J. Han, Cross-type biomedical named entity recognition with deep multi-task learning, arXiv preprint arXiv:1801.09851 (2018).
[22] K. W. Church, Word2vec, Natural Language Engineering 23 (2017) 155–162.
[23] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017. arXiv:1412.6980.
[24] C. Pesquita, D. Faria, A. O. Falcao, P. Lord, F. M. Couto, Semantic similarity in biomedical ontologies, PLoS computational biology 5 (2009).