Making Neural Networks FAIR

Anna Nguyen*, Tobias Weller*, and York Sure-Vetter

Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Abstract. Research on neural networks has gained significant momentum over the past few years. A plethora of neural networks is currently being trained on available data in research as well as in industry. Because training is a resource-intensive process and training data cannot always be made available to everyone, there has been a recent trend to attempt to re-use already-trained neural networks. As such, neural networks them- selves have become research data.
In this paper, we present the Neural Network Ontology, an ontology to make neural networks findable, accessible, interoperable and reusable as suggested by the well-established FAIR guiding principles for scientific data management and stewardship. We created the new FAIRnets Dataset that comprises about 2,000 neural networks openly accessible on the internet and uses the Neural Network Ontology to semantically annotate and represent the neural networks. For each of the neural networks in the FAIRnets Dataset, the relevant properties according to the Neural Network Ontology such as the description and the architecture are stored. Ultimately, the FAIRnets Dataset can be queried with a set of desired properties and responds with a set of neural networks that have these properties. We provide the service FAIRnets Search which is implemented on top of a SPARQL endpoint and allows for querying, searching and finding trained neural networks annotated with the Neural Network Onto- logy. The service is demonstrated by a browser-based frontend to the SPARQL endpoint.

Keywords: Neural Network · Ontology · Reusability · FAIR.

1 Introduction

The number of neural network architectures being proposed in research and the number of neural networks being trained on datasets have been on the rise in recent years.[1] Neural networks are not only applied for traditional information technology fields, but are also considered in finance [19,10], medicine [22,32], law [11,28] and public service [13,16] with increasing interest. Researchers of different sciences and data analysts are using neural networks and train neural network models on existing datasets. Providing scientific code, as well as trained

* These authors contributed equally to the work.

[1] http://scholarsight.org/ (keyword: neural networks) and https://trends.google.com/trends/explore?date=all&q=neural%20networks, last accessed: 2019-04-10.

models, is an important prerequisite for publishing work. Thus, the trained models are more and more uploaded to software repositories such as GitHub. The models provided serve not only to reproduce the results, but also to interpret them. In addition, the trained models can be reused or used via transfer learning in other domains.

To ensure a high-quality reuse of datasets and infrastructure, the 'FAIR Guiding Principles for scientific data management and stewardship' [34] have been proposed. These guidelines are designed to make digital assets **F**indable, **A**ccessible, **I**nteroperable, and **R**e-usable. They have been widely accepted by the scientific community nowadays.[2] Making digital assets FAIR is essential to deal with a data-driven world and thus keeping pace with an increasing volume, complexity and creation speed of data. We therefore aim for combining this concept with neural networks. We do not only consider datasets to be digital assets, but also the neural networks themselves which consist of architectures of neural networks and, subsequently, trained models. The key idea is that the information contained in these networks should also follow the FAIR guidelines. This enables to make neural networks and in particular trained models easy to find, accessible, integrable into an existing workflow, and describable via metadata.

Overall, we contribute the following two new resources:

1. We provide an ontology, called Neural Network Ontology, for representing neural networks according to the 5-star Linked Data Principles[3] of Tim-Berners Lee. The ontology is available online at the persistent URI `https://w3id.org/nno/ontology`.
2. We provide an RDF dataset, called FAIRnets Dataset, representing over 2,000 publicly available neural networks, following the FAIR Principles. The dataset is available online using a persistent URI by w3id,[4] and uploaded on Zenodo.[5]

The remainder of this paper is structured as follows. The following section gives an overview of related work. Section 3 describes the two resources, namely the Neural Network Ontology and the FAIRnets Dataset. In this context, we will address both the schema of the ontology and reused classes, and properties from well-established ontologies. Section 4 explains the reasons why the neural networks in the FAIRnets Dataset follow the FAIR Principles. Afterwards, in Section 5, we analyze the Neural Network Ontology, the FAIRnets Dataset and show an example how these resources interact with a provided web service called FAIRnets Search[6] for searching and analysing neural networks in the FAIRnets Dataset. Section 6 describes further ideas we want to follow in future. The last section summarizes the contributions and shows potential impact of the resources.

[2] `https://iswc2019.semanticweb.org/call-for-resources-track-papers/`, last accessed: 2019-04-10.
[3] `https://5stardata.info`, last accessed: 2019-04-06.
[4] `https://w3id.org/nno/data` last accessed: 2019-04-10.
[5] `https://doi.org/10.5281/zenodo.2634104`, last accessed: 2019-04-10.
[6] `http://km.aifb.kit.edu/services/fairnets/`, last accessed: 2019-04-10.

2 Related Work

Neural networks have been applied as a machine learning method to improve ontologies, in recent years. They are used to align [15,18,12], match [17,31] or map ontologies [14,24]. Furthermore, ontologies were combined with neural networks to solve different problems [33,29]. However, there is no standard ontology to describe neureal networks. Still, there exists an ontology which focuses on the description of weights [20] but does not fulfill the Linked-Data Principles.

There exist standards for the exchange of neural network information. The Predictive Model Markup Language (PMML) [6] is an XML-based standard for analytic models developed by the Data Mining Group. The Open Neural Network eXchange format (ONNX) [5] is a project by Facebook and Microsoft which converts the neural networks into different frameworks. These formats serve as an exchange format for neural networks. However, PMML is an XML-based standard that does not follow the Linked-Data Principles. Nonetheless, PMML is a standard sponsored by the Data Mining Group (DMG) and supported by more than 30 organizations. We are less interested in the exchange of formats but the reusability of the neural networks. Our neural network ontology is inspired by PMML. Yet, it lifts the elements to a semantic level, i.e. to RDF/S, following the methodology in 'Reusing Ontologies' [30].

Keras has ten available trained neural networks for reuse [3]. The Berkeley Artificial Intelligence Research Lab has a deep learning framework called Caffe Model Zoo [4] which consists of about fifty neural networks. Wolfram Alpha has a repository with neural networks [9] which consists of approximately eighty networks. We collected more than 2,000 neural networks.

The paper 'Model Cards for Model Reporting' [25] suggests relevant information about neural networks that should be considered when saving information about them. Information such as description, date of the last modification, link to papers or other resources for further information, as well as the intended purpose of a neural network, are taken into account. Storing these information makes the neural networks more transparent.

3 Resources

In the following, we will introduce our resources, the Neural Network Ontology and the FAIRnets Dataset in more detail. The schema for linking the information, as well as necessary properties and classes are provided by the Neural Network Ontology. The ontology is presented in Section 3.1, followed by the FAIRnets Dataset in Section 3.2. The FAIRnets Dataset contains metadata of publicly available, trained neural network models in RDF*.

3.1 Neural Network Ontology

The Neural Network Ontology defines categories and properties, as well as a scheme for storing information from neural networks (NN). The ontology was developed by using Protégé [26] and with particular consideration of the existing standards PMML [6] and ONNX [5]. With the help of the Linked Data Principles we lift these standards on a semantic level. In addition to the consideration of PMML in the development of the ontology, findings from further work were also considered. In particular, model cards [25] were taken into account. Model cards encourage transparent model reports. These cards propose information on neural networks that are relevant to the intended application domains. We thus follow existing best practices and incorporate insights from existing works into the development of the ontology.

An overview of the structure of the Neural Network Ontology can be seen in Figure 1. The figure shows only a few selected classes and relationships. For an entire representation of the ontology WebVOWL [23] can be used as visualization.[7]

The ontology enables us to represent two different components of information. First, NN-independent characteristics can be represented, such as author information, a textual description of the network, and a link to the network implementation. Second, NN-dependent features can be modeled, such as type of layer and activation functions. In the following, we will describe these two components of the Neural Network Ontology in more detail. We use the prefix `nno` to represent the namespace URI of the Ontology (`https://w3id.org/nno/ontology#`).

General Components: The general components describe general information about the learned model, as well as the intended use of the developed neural network. Among other things, the author/owner of the trained neural network is saved by using the property `dc:creator`. Following the Linked Data Principles, the author is represented as URI. Thus, the authors are uniquely identified. In addition, a name (`rdfs:label`) and a description (`dc:description`) of the trained neural network can be saved. This information already gives a more detailed insight into the neural network as well as the intended use of it. Further information about the neural network is the timestamp of the last modification (`dc:modified`) and the license (`cc:license`) under which the neural network was published. This information will allow to assess the actuality of the neural network and the rights to modify and redistribute that network. In addition, the property `nno:hasRepositoryLink` allows to link to the repository in which the neural network is located. Likewise, references to published papers can be included using `dc:references`. Further links can be modeled with `rdfs:seeAlso`.

Model Components: Aside from modeling general information about the neural network, model-specific components of the neural network can also be modeled. The class of the neural network (FFNN, CNN, RNN) is available in the ontology, as well as the optimizer such as Adam, Adagrad or RMSPROP. The most common loss functions (e.g. binary crossentropy and mean squared error), are modeled in the ontology. Loss functions are subdivided into classification loss function and regression loss functions in the ontology. Likewise, neural network layers such as Core Layers, Recurrent Layers and Convolutional Layers are mod-

[7] `http://visualdataweb.de/webvowl/#iri=https://w3id.org/nno/ontology`, last accessed: 2019-04-10.

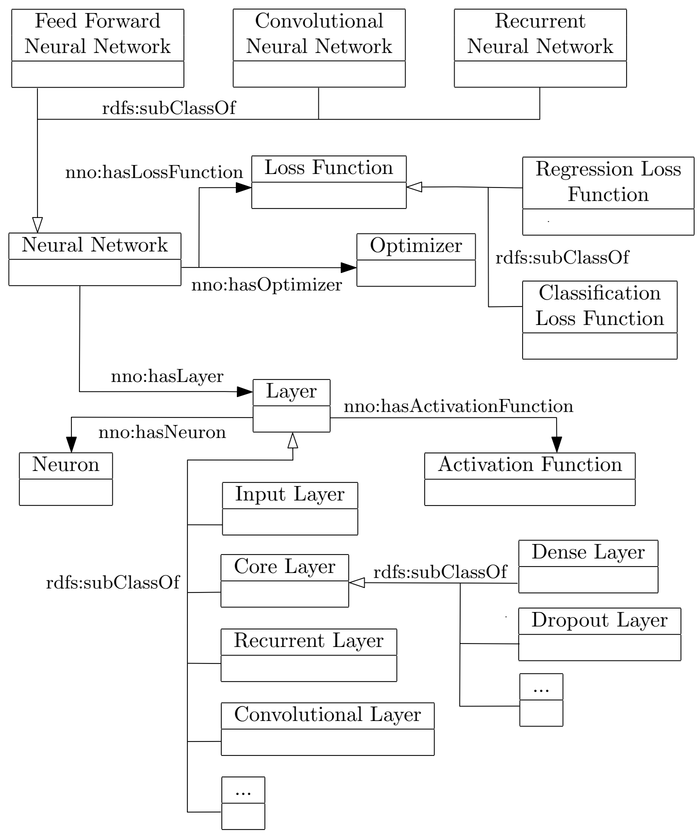

Fig. 1. The Neural Network Ontology as UML Class Diagram. We use the UML Class Diagram's class, inheritance and association to denote the RDFS ontology language's *rdfs:Class, rdfs:subClassOf* and *rdf:Property.*

eled. These classes are further subdivided into more specific layer classes. The loss functions and layer types available in Keras served as basis to model available loss functions and layers. By means of the property `nno:hasActivation Function` the activation function which is used in the layers can be linked. There is also a large selection of already modeled activation functions available such as softmax, relu and sigmoid available in the ontology. The ontology can be extended with additional layers, loss functions and activation functions. Furthermore, the ontology contains a class `nno:Neuron` with which the individual neurons of the neural network can be described. With the property `nno:hasNeuron` neurons can be added to the individual layers.

Most of the categories, properties and instances are annotated with a label (`rdfs:label`), a description (`rdfs:comment`) and, if given, a link (`rdfs:seeAlso`) which make it easy for ontology users to identify the intended use of categories, properties and instances and thus supports the reusability.

The W3C Permanent Identifier Community Group service[8] is used to ensure secure and permanent URL forwarding to the ontology. The Neural Network Ontology is accessible under `https://w3id.org/nno/ontology`. In addition to ensuring a persistent URI, the ontology was sufficiently annotated, including information about the creators, version of the ontology, description, preferred prefix and a license specification. Hereby, among others the vocabularies Dublin Core [2], Creative Common Rights Expression Language (CC REL) [1], Vocabulary of a Friend (VOAF) [8], Vocabulary for Annotating Vocabulary Description (VANN) [7] and RDF* were used to describe the ontology. These documentations support the reusability of the ontology. The ontology uses CC BY 4.0[9] as license. which supports the availability of the ontology. Likewise, the availability is given by the usage of a w3id persistent URI. Moreover, the ontology has been registered on LOV.[10] By publishing the ontology using w3id and LOV, the ontology is easily findable and is available for a wider community, beyond the work presented here.

3.2 FAIRnets Dataset

The Neural Network Ontology provides a schema for storing information about neural networks. We will use this to crawl freely available neural networks from GitHub and transfer the information into this ontology schema. We use the GitHub REST API[11] to find neural networks. We have used the two search terms *Neural Network* and *h5*. The first search term is used to identify neural networks. The second search term aims at finding h5 files.[12] h5 is an open source technology for storing trained Machine Learning models. Neural networks which have been trained with Keras, for example, can be stored in this format. This makes it easier for us to identify the architecture of the neural network by downloading the h5 models and loading them.

We use the *full_name* of the GitHub repository as a unique identifier. The *full_name* consists of the GitHub username combined with the name of the repository. It is unique within GitHub. The owner of the repository is also the owner of the neural network. We thereby make the assumption that the owner of the repository has also published the neural networks or at least is aware of the content of his repository. In addition, we store the link and the time of the last update of the repository. Furthermore, license information about the neural networks is stored, if available. This information is used to describe the way in which neural networks can be reused. Additionally, if existent in the repository, http links are searched within the readme file. arXiv links are linked via `dc:references`. If these links are not arXiv links, then they are interlinked to the network using `rdfs:seeAlso`. As the FAIR Principles suggest, these links are used for interoperability to include other related (meta)data.

[8] `https://w3id.org`, last accessed: 2019-04-10.
[9] `https://creativecommons.org/licenses/by/4.0/`, last accessed: 2019-04-10.
[10] `https://lov.linkeddata.es/dataset/lov/vocabs/nno`, last accessed: 2019-04-10.
[11] `https://developer.GitHub.com/v3/search/`, last accessed: 2019-04-10.
[12] `https://www.hdfgroup.org/solutions/hdf5/`, last accessed: 2019-04-10.

The repository is searched for existing h5 files. However, not every repository contains trained neural networks in h5 files. The reason is that trained neural networks often take up a lot of storage space. In order to save this space only the source code is provided with which the neural networks can be trained on the own computer. If there is an h5 file with architecture information, then an attempt is made to open it and read out the neural network structure. We use Keras to open the h5 file. The individual layers, their activation functions, if used, and the number of neurons in the layers, are saved. Likewise, the used loss function and the optimization function of the neural network are stored.

Just like the Neural Network Ontology, FAIRnets Dataset is also based on the 5-Star Linked Data Principles. The dataset is accessible under a persistent URI from w3id.[13] In addition, it is provided on Zenodo.[14] By combining the FAIR Principles and the Linked Data Principles, it is possible to easily reference and use the FAIRnets Dataset. Machine-readable metadata allows to describe and search the dataset itself as well as the neural networks in it. Creative Commons Ontology and DublinCore are used to add further annotations like license information and description to the dataset. The FAIRnets Dataset, like the Neural Network Ontology, is published under the Creative Commons BY 4.0[15] license. The license specifies the scope in which the data may be used.

The FAIRnets Dataset contains a high potential of insights into the use of neural networks in practice. In Section 5.2, we will present possible analysis. In addition, these networks can be used in a transfer learning approach. Further possible extensions and specified sustainable plans for the FAIRnets Dataset will be described in Section 6.

4 FAIR Principles for Neural Networks

In this section, we will identify the factors that make neural networks with our approach FAIR. The key aim of the FAIRnets Dataset is to treat neural networks as research data and as such should be provided according to the FAIR Principles. Hence, the research data on neural networks should be findable, accessible, interoperable and reusable.

Findable describes the property that metadata about digital assets is easy for both humans and machines to find. The neural networks we describe use unique identifiers and a persistent URI.[16] The neural networks are described with rich metadata. For the description we use existing standards such as DublinCore. The dataset is registered at Zenodo[17] and thus indexed and publicly accessible. By using RDF the data is machine-readable. Through the machine readability the dataset can be automatically detected and used by services. An exemplary

[13] `http://w3id.org/nno/data`, last accessed: 2019-04-10.

[14] `https://doi.org/10.5281/zenodo.2634104`, last accessed: 2019-04-10.

[15] `https://creativecommons.org/licenses/by/4.0/`, last accessed: 2019-04-10.

[16] `https://w3id.org/nno/data` and `https://doi.org/10.5281/zenodo.2634104`, last accessed:2019-04-10.

[17] `https://doi.org/10.5281/zenodo.2634104`, last accessed: 2019-04-10.

service for using and searching the FAIRnets Dataset, and thus supporting this statement, is presented in Section 5.3. Thus, the FAIRnets Dataset fulfills the findable property of the FAIR guideline.

Accessible describes that users can access the (meta)data using a standardised communications protocol. The protocol must be open, free, and universally implemented. The FAIRnets Dataset is located on a web server and can be accessed using the HTTPS protocol. The neural networks in the repositories can also be accessed using the HTTPS protocol. In addition to the open protocol, the accessible property requires that metadata can be retrieved even if the actual digital assets are no longer available. This property is also fulfilled by the FAIRnets Dataset because the metadata are still preserved even if they are deleted in the repositories. Thus, the FAIRnets Dataset also fulfills this property.

Interoperable refers to the capability of being integrated with other data as well as being available to applications for analysis, storage and further processing. We make use of Linked Data by applying the RDF* and SPARQL standards to represent the information. This makes the data machine-readable, even without the specification of an ad-hoc algorithm or mapping. The FAIRnets Dataset uses well-established and commonly used vocabularies to represent the information. Among others DublinCore, VOAF, CC and VANN are used for annotations and descriptions of the ontology and the FAIRnets Dataset. The Neural Network Ontology likewise follows the FAIR guidelines. The ontology is accessible via a persistent w3id URI, indexed in LOV, publicly accessible via standard protocols, sufficiently described using standard vocabularies and usable in other projects. License information specifies the intended reuse of the ontology. As a further requirement of the FAIR guideline, qualified references to further metadata are required. This requirement is fulfilled in the dataset by `rdfs:seeAlso` and `dc:references`. `dc:references` links provide scientific references between the neural networks and the scientific contributions. These references to the scientific contributions are provided via globally unique and persistent identifiers of the e-print service arXiv.[18] Accordingly, the FAIRnets Dataset also fulfills the intero- perability property required by the FAIR guidelines.

Reusable aims at achieving well-defined digital assets. This facilitates the replicability and usage in other contexts, as well as findability. The properties and classes of the Neural Network Ontology are provided with labels and descriptions so that these are self-explanatory. The FAIRnets Dataset contains a general description, the version number and the date of the last modification. Authors who created the content are listed. The information about the authors can among others be used for citation purposes. When this work is published, we will also provide the associated citation data. In addition, the FAIRnets Dataset contains the license used, which is CC BY 4.0.[19] The license information clearly indicates which rights of use are attached to the data. Thus, the FAIRnets Dataset also fulfills this requirement.

[18] `https://arxiv.org`, last accessed: 2019-04-10.

[19] `https://creativecommons.org/licenses/by/4.0/`, last accessed: 2019-04-10.

The FAIRnets Dataset thus fulfills all requirements of the FAIR Principles. By using standards, we emphasize machine-actionability. The use of RDF* to describe the information enables the integration of further information. Neural networks can also be easily found, called and reused by using SPARQL. The dataset can also be used for further analysis. We will make use of it in Section 5.2 by applying exemplary queries on the FAIRnets Dataset. This will help us to get a better understanding of the available information and its potential.

5 Analysis

In the following Section 5.1, we will discuss the metrics and properties of Neural Network Ontology in more detail. Afterwards, we will analyze the provided dataset in Section 5.2 and point out possibilities to what extent the dataset can be used to gain insights from the previously published neural networks. With these analysis, we want both to gain new insights and to identify interest in the FAIRnets Dataset. Section 5.3 presents the service FAIRnets Search. This service allows to search through the FAIRnets Dataset and find the suitable neural networks.

5.1 Neural Network Ontology

The ontology consists of a total of 598 axioms and uses a total of 80 classes whereof 71 are subclasses. It also consists of six object properties and 18 data properties. We reuse existing properties, among others cc, dc, vann and vs.

We use the w3id as a secure and persistent URI.[20] The preferred prefix is *nno*. The ontology is annotated with the authors, license, description, version and namespace uri and prefix. The used license is CC BY 4.0.[21] Thus the ontology may be changed and used under the same license. In addition, it may not be used for commercial purposes. The ontology follows the 5-Star Linked Data Principles and can therefore be easily reused. By using a license there is a clear description of how use it. Additionally, there is a human- and machine-readable DublinCore description using `dc:description`. The Neural Network Ontology is registered on Linked Open Vocabularies, is publicly findable and accessible, and follows the FAIR Principles by the described properties.

We followed best practices for the development of the ontology [27]. We lifted existing standards like PMML [6] and ONNX [5], as well as reused existing vocabularies like e.g. DublinCore [2] and Vocabulary of a Friend (VOAF) [8]. The CamelCase notation were used in naming the classes, properties and instances. Self-describing names were chosen for the names of the classes and properties. By using the w3id.org service for Permanent Identifiers for the Web, the URLs are both logical and short enough. Also the preferred namespace prefix of the Neural Network Ontology, which is `nno`, is below the proposed maximum of five

[20] `https://w3id.org/nno/ontology`, last accessed: 2019-04-10.
[21] `https://creativecommons.org/licenses/by/4.0/`, last accessed: 2019-04-10.

letters. In addition, we gave all concepts and properties a definition by using `dc:description` and a label by using `rdfs:label`. Multi-lingual capabilities are enabled, but currently not used. We assigned domains and ranges to the properties of the Neural Network Ontology, where appropriate. WebVOWL[22] is used to visualize the ontology in its complete aspect. In addition, Figure 1 shows an excerpt of the ontology to aid understanding it. The applicability of the ontology was extensively tested. The application of the ontology has an effect in the analysis of the FAIRnets Dataset, as well as in the FAIRnets Search presented in Section 5.3.

5.2 FAIRnets Dataset

By the representation in RDF structured information about the networks can be queried. We will take advantage of this in order to get a better understanding of the dataset and the neural networks used in practice. The FAIRnets Dataset contains 2,154 neural networks, provided by 2,096 users. All these networks have a link to the respective repository and owner. Description and license information are available, if existent in the repository. However, information about the exact architecture of the neural network is not available for all of these networks which includes the used layers, activation functions, optimization functions, etc. Information on the exact architecture is available for 116 (5.385%) neural networks. For the remaining 2,038 networks (94.615%) this information is currently not available. The information about the class of neural networks is inferred from the used layers. If a network uses a convolutional layer, then it is inferred that the network is a CNN. Likewise if a network contains a recurrent layer, then it is inferred that the network is an RNN. The neural network architecture is only available for 116 neural networks. That is why we can only make statements about the structure of these networks. The largest class containing 67 neural networks in the FAIRnets Dataset is CNN. This is more than half of the neural networks with available architecture information. It follows FFNN with 47 and RNN with four neural networks.

With the development and progress of neural networks, their distribution is increasing. To demonstrate this effect we displayed the number of neural networks over time in Figure 2. We used the information about the last modification as the time of the temporal course. This figure shows that in 2017, the slope of the CNN is stronger than that of the FFNN, i.e. that proportionally more CNNs have been provided than FFNNs. In 2018, however, this was aligned. Due to the small number of RNNs, no significant statement can be inferred here. Though, the small number of RNNs available means that only very few learned RNN models are made available and very little use is made of these network architectures.

Likewise interesting is the use of the activation functions in the neural networks. Often ReLu functions are recommended, because they converge very fast.

[22] http://visualdataweb.de/webvowl/#iri=https://w3id.org/nno/ontology, last accesed: 2019-04-10.

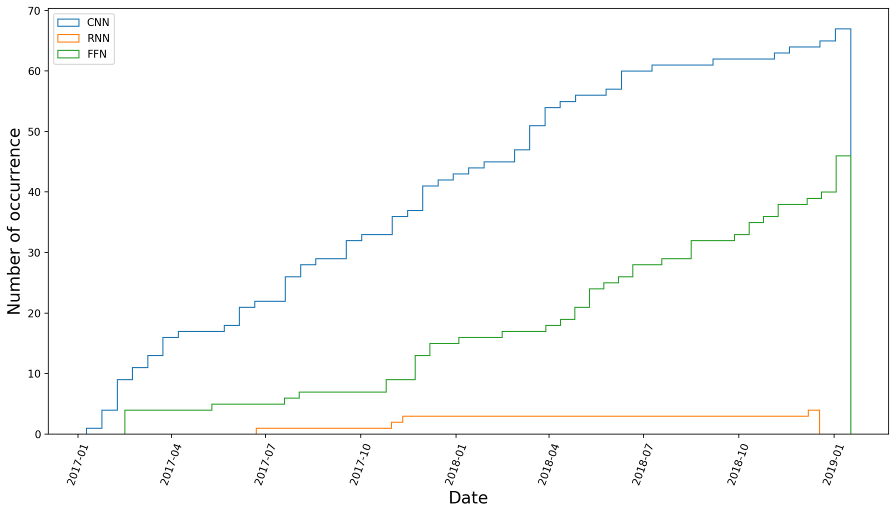

Fig. 2. Cumulative histogram of the number of Neural Networks provided over time in the FAIRnets Dataset.

For this reason, we wanted to investigate whether this best practice also applies to this dataset. As it can be seen in Figure 3, the most frequently used activation function is indeed ReLu. This function is followed by Merge and Bias Add. Tanh was not used at all. A similar result of the distribution of the activation functions can be seen when considering them with regard to their use in the individual network architectures.

In addition to the activation functions used, the layers used in practice are also of interest. Here, a distinction must be made between the network architectures. Figure 4 shows the used layers of CNNs. It is not surprising that Convolutional 2D layers are the most commonly used ones. Dense and Dropout layers follow. Dropout layers are probably often used to overcome the overfitting problem of increasing deep neural networks. An average CNN in FAIRnets Dataset has a depth of 18 layers. Based on Figure 4, it can be assumed that the layer types used are very simple and less complex layers, than e.g. Convolutional 3D layer. Even a GRU was used which suggests a RNN. In fact, CNN and RNN structures were combined to an RCNN [21]. A similar analysis of the most frequently used layers can be performed for FFNN and RNN. FFNNs have an average depth of six layers and whereas RNNs have nine layers, which are much flatter than CNNs.

In addition to the identification of frequently used activation functions and layer types, the deduction of an average neural network is of interest. We have counted the number of layer types per layer. We chose the layer type that appears most frequently per layer and selected it as the representative of an average network. Due to the different structures we distinguish between CNN, FFN and

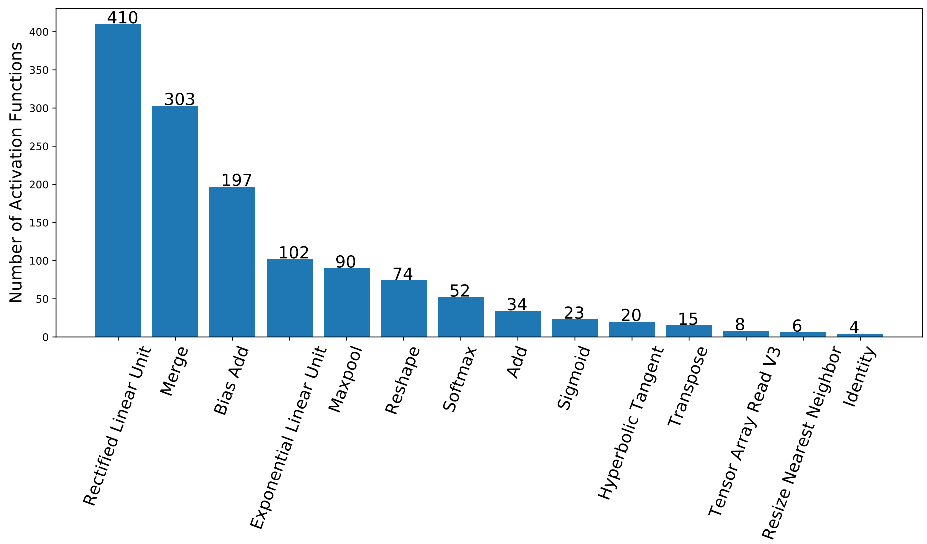

Fig. 3. Used activation functions in the neural networks.

RNN. As described above, a CNN has an average of 18 layers. Figure 5 represents the average CNN. It uses Convolutional 2D layers as first layers followed by a Dropout or Max Pooling 2D layer. Both of them occur with the same frequency. The remaining layers are filled with Dense layers.

The FAIR Principles also state that the digital assets are provided with clear and accessible data usage license. In total, 531 (24.652%) neural networks are provided with license information. With 446 times (83.992%), the MIT license was most frequently used for neural networks. This license allows the use, distribution and modification of the networks for commercial purposes. A liability or a warranty about the correctness is not guaranteed. The second most common license is the GNU GPLv3 license (38 usages). Compared to the MIT license, it states that if sources are used with the GNU GPLv3 license, the source code of the software must be published. The use of license information is essential for the correct use and distribution of digital assets. We hope that the use of license information for future neural networks will increase.

5.3 FAIRnets Search

In order to provide a further benefit to make the FAIRnets Dataset easily browsable and neural networks findable, we offer FAIRnets Search.[23] With FAIR- nets Search, search terms can be entered and corresponding neural networks can be

[23] http://km.aifb.kit.edu/services/fairnets/, last accessed: 2019-04-10.

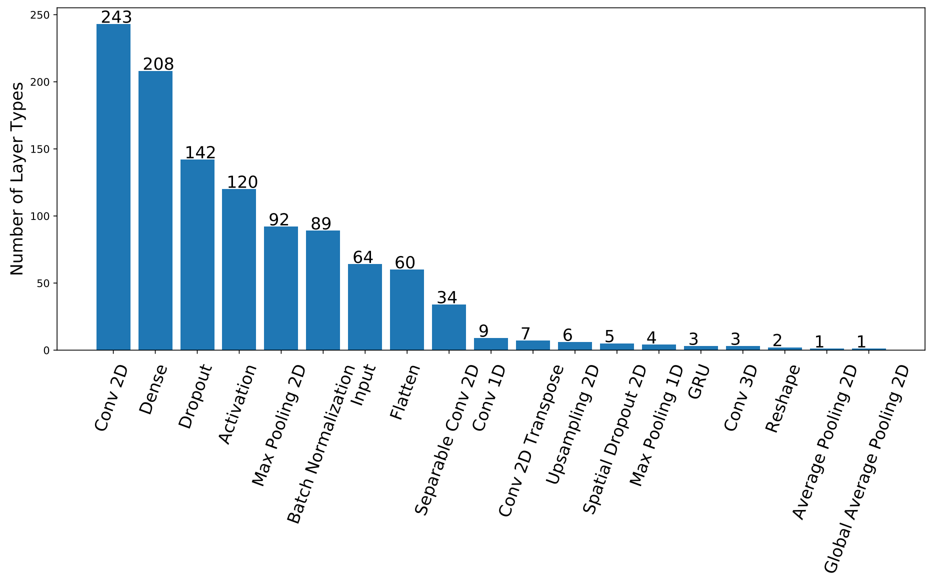

Fig. 4. Number of layer types used in CNNs.

found in the FAIRnets Dataset. The search makes use of SPARQL. FAIRnets Search uses django,[24] a high-level Python Web framework, and rdflib.[25]

6 Future Work

We see high potential in the areas explainability and reusability. Furthermore, we get insights into the use of neural networks in practice. Our framework gives basic information about neural network to start with. Given the structured data, it might be easier to find correlations and patterns within architectures, domains or types. Best practices could be computed based on a given problem or domain and then be inferred to generalize a neural network per problem or domain.

Currently the FAIRnets Dataset contains only architecture information for those neural networks, which also provide a trained h5 model. We want to address this in the future and provide architecture information even if they do not provide a trained h5 model. For this purpose, we will scan the repositories for structural information.

Furthermore, it is conceivable to use the available knowledge of the Neural Network Ontology and the FAIRnets Dataset to recommend suitable neural networks for a given problem to the users. This scenario can be extended step by step, all the way to the provision of a training dataset by the user. The system analyzes the user's requirements and the dataset provided, selects suitable

[24] https://www.djangoproject.com, last accessed: 2019-04-10.
[25] https://github.com/RDFLib/rdflib, last accessed: 2019-04-10.

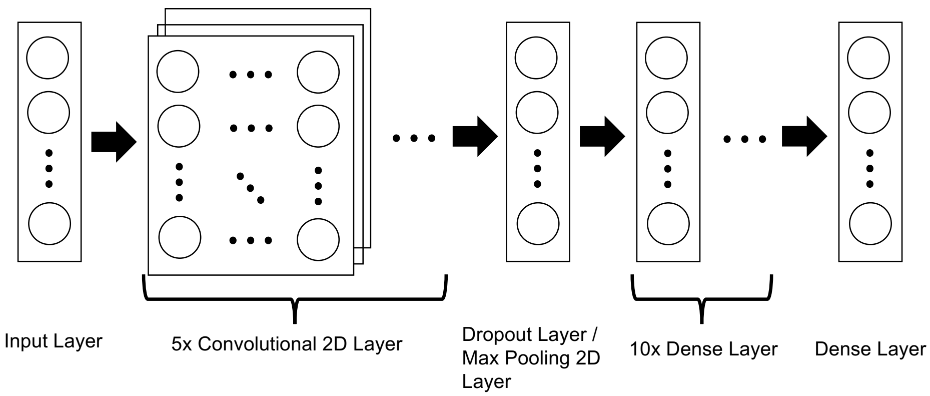

Fig. 5. Presentation of the deduced average CNN used in the neural networks in the FAIRnets Dataset.

trained models and adapts them to the dataset provided by means of transfer-learning. In addition, Neural Network Ontology can be used for the explainability of neural networks. This can range from providing simple descriptive texts on the structure of neural networks, up to explaining their results. Hereby, the Neural Network Ontology can be used among others to exploit the hierarchical structure of layers.

7 Conclusion

In this paper, we presented the Neural Network Ontology and the FAIRnets Dataset as new resources. We consider neural networks as research data and provide them according to the FAIR Principles. For this purpose, we crawled publicly available neural networks on GitHub and used the ontology to annotate these crawled neural networks. Through the FAIR Principles, we ensure that neural networks are now easily findable, accessible, integrable with other informa- tion, and therefore easy to reuse in different settings. Furthermore, by implemen- ting the Linked Data Principles, we bring a new perspective to the data view, additionally to the FAIR Principles, with which we combine these two worlds in a meaningful way.

To show potential impact, we analyzed the FAIRnets Dataset and highlighted the possibilities for gaining new insights from the FAIRnets Dataset. The findings help to deduce best practices in the field of neural net- works. Moreover, it helps to identify the extent to which scientific findings are implemented in practice. Additionally, we provide FAIRnets Search to easily search and find neural networks. The two resources presented here already show a high potential. As shown in Future Work, however, there are further ideas with which we would like to enhance the contributions in the future. Besides this, the insights gained already allow us to identify important points and help us to reuse neural networks.

We see a huge impact of the Neural Network Ontology in both research and industry sector. As reproducibility plays a major role in research and data retrieval in industry, our resources set a baseline to start with neural networks. Moreover, by reusing and collecting trained neural networks we do not only build a basis of data for further studies but also reflect the needed use cases of all the developers. We appeal to the community to share their networks and/or complete the information for already provided networks (e.g. on GitHub).

References

1. Cc rel. https://wiki.creativecommons.org/wiki/CC_REL, last accessed 08 Apr 2019
2. Dublin core metadata initiative. http://dublincore.org, last accessed 08 Apr 2019
3. Keras applications. https://keras.io/applications/, last accessed 18 Mar 2019
4. Model zoo. https://github.com/BVLC/caffe/wiki/Model-Zoo, last accessed 18 Mar 2019
5. Onnx homepage. https://onnx.ai, last accessed 15 Mar 2019
6. Pmml 4.0 - neural network models. http://dmg.org/pmml/v4-0-1/NeuralNetwork.html, last accessed 15 Mar 2019
7. Vann: A vocabulary for annotating vocabulary descriptions. http://vocab.org/vann/, last accessed 08 Apr 2019
8. Vocabulary of a friend (voaf). https://lov.linkeddata.es/vocommons/voaf/v2.3/, last accessed 08 Apr 2019
9. Wolfram neural net repository. https://resources.wolframcloud.com/NeuralNetRepository, last accessed 18 Mar 2019
10. Bahrammirzaee, A.: A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. Neural Computing and Applications **19**(8), 1165–1195 (2010)
11. Cceda, R., James, G.A., Ely, T.D., et al.: Mode of Effective Connectivity within a Putative Neural Network Differentiates Moral Cognitions Related to Care and Justice Ethics. PLOS ONE **6**(2) (2011)
12. Chortaras, A., Stamou, G., Stafylopatis, A.: Learning ontology alignments using recursive neural networks. In: Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005. pp. 811–816. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)
13. lvaro Costa, Markellos, R.N.: Evaluating public transport efficiency with neural network models. Transportation Research Part C: Emerging Technologies **5**(5), 301 – 312 (1997)
14. Curino, C., Orsi, G., Tanca, L.: X-som: A flexible ontology mapper. In: 18th International Workshop on Database and Expert Systems Applications (DEXA 2007). pp. 424–428. IEEE, Regensburg, Germany (2007)
15. Djeddi, W.E., Khadir, M.T.: Ontology alignment using artificial neural network for large-scale ontologies. Int. J. Metadata Semant. Ontologies **8**(1), 75–92 (2013)
16. Garrido, C., de Oa, R., de Oa, J.: Neural networks for analyzing service quality in public transportation. Expert Systems with Applications **41**(15), 6830 – 6838 (2014)

17. Gracia, J., Bernad, J., Mena, E.: Ontology matching with cider: Evaluation report for oaei 2011. In: Proceedings of the 6th International Conference on Ontology Matching - Volume 814. pp. 126–133. Aachen, Germany, Germany (2011)
18. Huang, J., Dang, J., Huhns, M.N., et al.: Use artificial neural network to align biological ontologies. BMC Genomics **9**(2) (2008)
19. Huang, W., Lai, K.K., Nakamori, Y., et al.: Neural networks in finance and economic forecasting. International Journal of Information Technology & Decision Making **06**(01), 113–140 (2007)
20. Iqbal, M.R., Klawonn, M., Norris, S.: An ontology for automated neural network initialization with heuristic knowledge graph. Tech. rep. (2016), rensselaer Polytechnic Institute Technical Report, Ontology Engineering 2016
21. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4), 664–676 (2017)
22. Khan, J., Wei, J.S., Ringnr, M., et al.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nature Medicine **7**(6), 673–679 (2001)
23. Lohmann, S., Negru, S., Haag, F., et al.: Visualizing ontologies with VOWL. Semantic Web **7**(4), 399–419 (2016)
24. Mao, M., Peng, Y., Spring, M.: An adaptive ontology mapping approach with neural network based constraint satisfaction. Journal of Web Semantics **8**(1), 14–25 (2010)
25. Mitchell, M., Wu, S., Zaldivar, A., et al.: Model Cards for Model Reporting. In: Proceedings of the Conference on Fairness, Accountability, and Transparency - FAT* '19. pp. 220–229. ACM Press, Atlanta, GA, USA (2019)
26. Musen, M.A.: The ProtG Project: A Look Back and a Look Forward. AI Matters **1**(4), 4–12 (2015)
27. Noy, N.F., McGuinness, D.L.: Ontology development 101: A guide to creating your first ontology. Tech. rep. (2001), technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880
28. Palocsay, S.W., Wang, P., Brookshire, R.G.: Predicting criminal recidivism using neural networks. Socio-Economic Planning Sciences **34**(4), 271 – 284 (2000)
29. Peng, J., Wang, X., Shang, X.: Combining gene ontology with deep neural networks to enhance the clustering of single cell rna-seq data. bioRxiv (2018)
30. Pinto, H.S., Martins, J.P.: Reusing Ontologies. AAAI Technical Report SS-00-03 pp. 77–84 (2000)
31. Rubiolo, M., Caliusco, M.L., Stegmayer, G., et al.: Knowledge discovery through ontology matching: An approach based on an Artificial Neural Network model. Information Sciences **194**, 107–119 (2012)
32. Staudenmayer, J., Pober, D., Crouter, S., et al.: An artificial neural network to estimate physical activity energy expenditure and identify physical activity type from an accelerometer. Journal of Applied Physiology **107**(4), 1300–1307 (2009)
33. Trappey, A.J.C., Trappey, C.V., Chiang, T.A., et al.: Ontology-based neural network for patent knowledge management in design collaboration. International Journal of Production Research **51**(7), 1992–2005 (2013)
34. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., et al.: The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data **3** (2016)