Experience: Automated Prediction of Experimental Metadata from Scientific Publications

Experience: Automated Prediction of Experimental Metadata from Scientific Publications

STUTI NAYAK, Institute of Data Science, Maastricht University, The Netherlands
AMRAPALI ZAVERI, Institute of Data Science, Maastricht University, The Netherlands
PEDRO HERNANDEZ SERRANO, Institute of Data Science, Maastricht University, The Netherlands
MICHEL DUMONTIER, Institute of Data Science, Maastricht University, The Netherlands

While there exists an abundance of open biomedical data, the lack of high-quality metadata makes it challenging for others to find relevant datasets and to reuse them for another purpose. In particular, metadata are useful to understand the nature and provenance of the data. A common approach to improving the quality of metadata relies on expensive human curation, which itself is time-consuming and also prone to error. Towards improving the quality of metadata, we use scientific publications to automatically predict metadata key: value pairs. For prediction, we use a Convolutional Neural Network (CNN) and a Bidirectional Long-short term memory network (BiLSTM). We focus our attention on the NCBI Disease Corpus, which is used for training the CNN and BiLSTM. We perform two different kinds of experiments with these two architectures (i) we predict the disease names by using their unique ID in the MeSH ontology and (ii) we use the tree structure of MeSH ontology to move up in the hierarchy of these disease terms which reduces the number of labels. We also perform various multi-label classification techniques for the above-mentioned experiments. We find that in both cases CNN achieves the best results in predicting the superclasses for disease with an accuracy of 83%.

CCS Concepts: • **Computing methodologies → Information extraction**.

Additional Key Words and Phrases: datasets, neural networks, metadata, quality, natural language processing

ACM Reference Format:
Stuti Nayak, Amrapali Zaveri, Pedro Hernandez Serrano, and Michel Dumontier. 2021. Experience: Automated Prediction of Experimental Metadata from Scientific Publications. *ACM J. Data Inform. Quality* 1, 1, Article 1 (January 2021), 11 pages. https://doi.org/10.1145/3451219

1 INTRODUCTION

Enormous amounts of biomedical data have been and are being produced at an unprecedented rate by researchers all over the world [11]. This is mainly due to advancements in molecular technologies that have enabled extensive profiling of biological samples and have unleashed a myriad of so-called *'omics* data such as gene expression, microRNA expression, DNA methylation, and DNA mutation data. During the last decade journals, investigators, funding agencies have realized that this data should be stored, shared with and used by other investigators. However, to enable reuse, there is an urgent need to understand the structure of datasets and the experimental

Authors' addresses: Stuti Nayak, stuti257@gmail.com, Institute of Data Science, Maastricht University, Universiteitssingel 60 (1st floor), Maastricht, The Netherlands, stuti257@gmail.com; Amrapali Zaveri, Institute of Data Science, Maastricht University, Universiteitssingel 60 (1st floor), Maastricht, The Netherlands, amrapali.zaveri@maastrichtuniversity.nl; Pedro Hernandez Serrano, Institute of Data Science, Maastricht University, Maastricht, The Netherlands; Michel Dumontier, Institute of Data Science, Maastricht University, Maastricht, The Netherlands.

1936-1955/2021/1-ART1 $15.00
https://doi.org/10.1145/3451219

conditions under which they were produced [3]. That is, there is an urgent need for an accurate, structured and complete description of the data – defined as *metadata*.

While there exists an abundance of open biomedical data, the lack of high-quality metadata makes it challenging for others to find relevant datasets and to reuse them for another purpose [9, 12]. This, in turn, can facilitate a data-driven approach by combining and analyzing similar data to uncover novel insights or even more subtle trends in the data. These insights can then be formed into a hypothesis that can be tested in the laboratory [1]. In particular, metadata are useful to understand the nature and provenance of the data. A common approach to improving the quality of metadata relies on expensive human curation, which itself is time-consuming and also prone to error. Poor metadata leads to the problems of (i) interpretability - can we understand what was done in the biomedical experiment?; (ii) findability - to find all studies that meet constraints (e.g. all studies for a particular disease); and (iii) re-usability - to use this data for discovery, validation, and reproducibility.

The FAIR principles specify desirable criteria that metadata and their corresponding datasets should meet to be Findable, Accessible, Interoperable, and Reusable [20]. For data to be FAIR, metadata needs to be accurate and uniform (relying on controlled terms where possible). However, currently, there is a large amount of biomedical metadata, which is of poor quality; that is, it is extremely heterogeneous and which makes data reuse extremely difficult [9]. One of the major challenges towards assessing and improving the quality of biomedical metadata is the size of data that is present. Delving further into the problem at hand – let's take an example of Gene Expression Omnibus (GEO) dataset [8] which is a widely used database for cross-species gene expression data. Currently users can submit data to GEO via three ways: (i) Spreadsheets, (ii) SOFT format (plain text), (iii) MINiML format[1] (XML). When users submit data to GEO via a spreadsheet, it requires them to fill out a metadata template that follows the guidelines set out by the Minimum Information About a Microarray Experiment (MIAME) guidelines [4]. GEO allows users to specify metadata in the form of textual key: value pairs (e.g. *sex: female*). However, since there is no structured vocabulary or format available, the 44,000,000+ *key: value* pairs suffer from numerous quality issues such as:

- minor spelling discrepancies (e.g. `age at diagnosis (years)`, `age at diagonosis (years)`; `genotype/varat`, `genotype/varaiation`, `genotype/variaion` `genotype/variataion`)
- having different syntactic representations (e.g. `age (years)`, `age(yrs)` and `age_year`)
- using different terms altogether to denote one concept (e.g. `disease` vs. `illness` vs. `condition`)
- using two different key terms in one (e.g. `disease/cell type`, `tissue/cell line`, `treatment age`).

Looking at these issues it can be seen that there is an urgent need to solve this research problem which would in-turn facilitate the re-usability of data.

Using domain experts for the curation of assessing the quality of metadata is not only time consuming, but also not scalable. Moreover, without a standardized set of terms with which to fill out the template fields (in the form when filling out the metadata), there are different versions of the same term without any (semantic) links between them, thus leading to several quality problems. Thus, there is a need for efficient methods for curating the metadata. In our previous projects [15, 16] we tried to cluster similar terms together using topic modeling and machine learning respectively. But when dealing with the values, it is very complex to map them to a concept because of a large amount of data. Whereas in scientific publication, we can identify values and then map them to a particular concept; this can be done per document. Therefore, we could exploit the information present in the scientific publication - where we aim to automatically predict metadata from scientific

[1] 'MIAME Notation in Markup Language' format

publications (unstructured text) using Machine Learning or Deep Learning, in other words, we want to build a **Metadata Wizard**.

We hypothesize that the scientific publications have in-depth descriptions of the experiments performed which facilitate predicting better quality metadata. This work assesses the extent up-to which experimental metadata *(key: value pairs - for example, disease: tuberculosis)* is predicted automatically using the scientific publications. More specifically, we want to automatically predict the values where the input is the abstract from a scientific publication. For instance, if a scientific publication reports information about myocardial infarction - we want to have the method to output the same automatically.

The contributions of this work are: (i) we develop a deep learning model for identifying metadata using scientific publications focusing on the metadata category *disease*; (ii) we perform empirical experiments on NCBI disease corpus [7] to identify disease categories and specific disease using different neural network approach; and lastly (iii) we present an analysis of results. In this manuscript, we discuss the methodology (Section 2), then we move on to results (Section 3), followed by conclusion, limitations and future work.

2 METHODOLOGY

The task of text classification or text categorization requires the following: assign a set of predefined categories to unstructured text that can be used to organize and structure the text appropriately, according to the use case. Text classification is a central problem with several applications in biomedicine. For instance, problems such as recognizing reportable cases of cancer from pathology reports, identifying certain phenotypes from clinical notes, performing word sense disambiguation (that is, given a context determine the semantic meaning for the usage of an ambiguous word), and associating medical subject headings (MeSH terms) to scientific articles, can all be reduced to instances of generic text classification problem. Further, text classification can be grouped into two categories namely, (i) multi-class classification (i.e. labels are mutually exclusive) and (ii) multi-label classification where each input can be assigned to more than one label. By definition, it is clear that multi-class classification is a special case of multi-label classification, and hence the latter problem becomes more harder to solve than the former.

We perform two experiments for the problem of text classification: (i) Multi-label classification and (ii) Deep learning - specifically Convolutional Neural Network (CNN) and Bidirectional Long Short Term Memory (BiLSTM). We illustrate a comparison of the performance of the methods mentioned above in Section 3. The motivation behind this comparison of employing two techniques is to answer the research question: which method performs better when we deal with the prediction of metadata (See Figure 1).

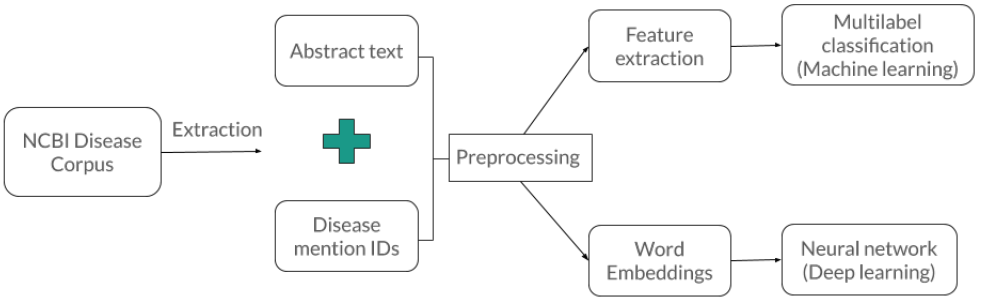

Fig. 1. Overview of the Metadata Wizard methodology.

2.0.0.1 Multi-label Classification. The dataset was tested with three types of problem transformations: (i) Binary Relevance (BR), (ii) Label Powerset (LP) and (iii) Classifier Chains (CC). The problem transformations were used from scikit-multilearn that is a library specific to multi-label classification built on top of the well-known scikit-learn. [19]. The classifiers that were tested using these transformations are listed as follows:

- Multi-layer Perceptron classifier (MLP classifier)
- Multi-label k Nearest Neighbour (MLkNN)
- Random Forest Classifier
- Decision Trees Classifier.

We selected these four classifiers as these are most widely used classifiers when performing a multi-label classification. For each combination, the evaluation metrics used were: accuracy and weighted F1 score. The classifiers and the metrics were calculated using scikit-learn [17]. The impurity measures for Decision trees and Random forest are kept as default (which is zero).

2.0.0.2 Deep Learning. The advent of deep neural networks (deep nets) in the last decade or so has led to a foundation for generic alternatives to supervised learning, especially for the task of object classification. Deep nets eliminate the laborious process of feature engineering and automatically learn high-level representations of input which suites best for the underlying classification problem. Although the resurgence of deep nets was initially meant for the field of computer vision, recently it has also been applied to natural language processing tasks (NLP)[2, 6, 14] primarily through learning distributed representations of words as vectors in high dimensional space. These vectors help the model in guiding elementary tasks such as part-of-speech (POS) tagging and parsing as well as abstract tasks such as text classification and machine translation. Typically, the novel deep learning approaches for text classification rely on architectures based on convolutional neural networks (CNNs) or recurrent neural networks (RNNs) [21].

In Figure 2, you can see a typical deep learning-based text classification architecture pipeline.

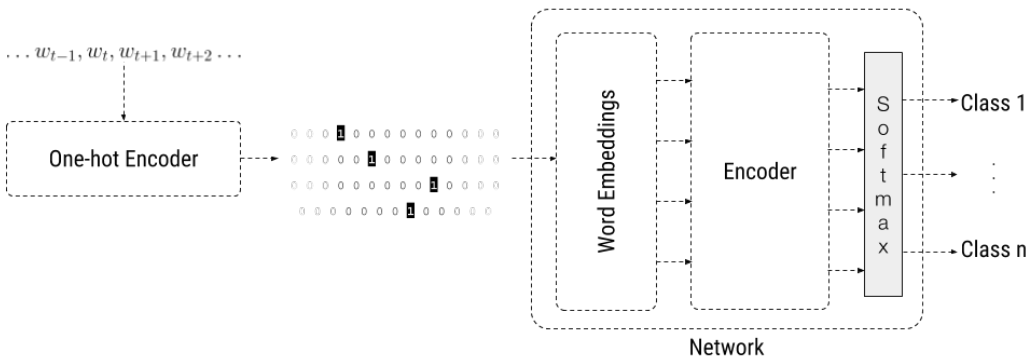

Fig. 2. Typical neural network based deep learning architecture for multiclass classification of texts.

For the neural network approach, we utilized the pre-trained word embeddings - GloVe: Global Vectors for Word Representation [18] of dimension 300. For data preprocessing, we used the tokenizer API from Keras [5]. The CNN and BiLSTM architectures were used from PyTorch[2]. The dropout rate has been kept 0.2 in all the neural network models. The activation function used is Rectified Linear Unit (ReLU). The number of hidden layers is kept 256. The models are trained using

[2]https://pytorch.org/

the Adam optimizer [13]. For validating the results using the neural networks, we split the data into training and validation sets. The training set is 80% of the whole data, and the validation set is the rest 20%. A similar train-test split was done for the baseline multi-label classification 80%-20%.

3 RESULTS AND DISCUSSION

The source code of our pipeline and resulting data is available in GitHub for re-use and analysis at https://github.com/MaastrichtU-IDS/metadata-wizard. The results are presented using the neural network model, and they are compared with baseline multi-label classification methods. The comparison is divided into two parts (i) predicting the disease terms and (ii) predicting the superclass of the disease terms.

Furthermore, we would like to describe how these scores are calculated in a multi-label setting. Hence, for this purpose, statistical measures are used based on the confusion matrix shown in Table 1. In the confusion matrix, TP is the number of times the positive class (1) is classified correctly, FP means the number of times the negative class (0) is misclassified as positive (1), TN means the number of times the negative class is classified correctly and FN is the number of times the positive class (1) is classified incorrectly.

		Predicted	
		0	1
Actual	0	true negative (TN)	false positive (FP)
	1	false negative (FN)	true positive (TP)

Table 1. Confusion Matrix - table that is used to calculate evaluation metrics

The metrics that have been used in this project are defined as follows:

- Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$, which is the percentage of times where the model gives correct results.
- Precision = $\frac{TP}{TP+FP}$, this is also known as the positive predictive value. This metric tells the percentage of relevant results among the predicted results.
- Recall = $\frac{TP}{TP+FN}$, also known as sensitivity. This is the percentage of positive instances classified correctly among all the correctly classified instances.
- F1 score = $\frac{2*precision*recall}{precision+recall}$, this is used when we have a class which has a smaller number of occurrences. It helps in combining the trade-offs of precision and recall.

Now we define the metrics in terms of multi-label classification. For this purpose, we use an example demonstrated in Table 2:

Input	$y^{(i)}$ (Actual label)	$\hat{y}^{(i)}$ (Predicted Labels)
$\tilde{x}^{(1)}$	[1 0 1 0]	[1 0 0 1]
$\tilde{x}^{(2)}$	[0 1 0 1]	[0 1 0 1]
$\tilde{x}^{(3)}$	[1 0 0 1]	[1 0 0 1]
$\tilde{x}^{(4)}$	[0 1 1 0]	[0 1 0 0]
$\tilde{x}^{(5)}$	[1 0 0 0]	[1 0 0 1]

Table 2. An example of predictions in a multi-label classification to depict evaluation metrics

We start by defining the accuracy:

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N}\frac{|\hat{y}^{(i)} \wedge y^i|}{|\hat{y}^{(i)} \vee y^i|} \tag{1}$$

where $\wedge$ and $\vee$ are logical OR and AND operations, which are applied vector-wise. Then we define the F_1 measure for the multi-label classification:

$$F_1 = \frac{1}{N}\sum_{i=1}^{N}\frac{2|\hat{y}^{(i)} \wedge y^i|}{|\hat{y}^{(i)}| + |y^i|} \tag{2}$$

Overall, higher the value of accuracy and F_1 score the better the learning algorithm.

3.1 Predicting Disease Terms

In the following paragraphs we present the results of both approaches.

3.1.1 Multi-label Classification. For baseline multi-label classification, two types of feature extraction techniques were used: (i) TF-IDF and (ii) Bag of words representation (BoW). The results for these two techniques are presented in Tables 3 and 4. For both the techniques, the maximum feature length is kept at 200. It can be seen that the Multilayer Perceptron (MLP) classifier performs the best in both the techniques of feature extraction.

There is a difference in the performance if we compare the different feature extraction methods; the TF-IDF technique performs better than BoW. The TF-IDF normalizes the outputs, which is why this technique is known to perform better when it comes to the feature selection; it can be observed from the results as well. Since the dataset has a large number of diseases - 588, and a limited number of abstracts, the accuracy of predictions is not very high. The highest accuracy that we get here is $\approx$ 19%.

Classifiers	Random forest			Decision Tree			MLPClassifier			MLkNN			
Problem Transformations	BR	LP	CC	BR	LP	CC	BR	LP	CC	k=20	k=30	k=10	k=5
Accuracy	**0.101**	0.075	0.088	0.044	**0.107**	0.082	0.170	**0.189**	0.164	0.164	0.138	**0.176**	**0.176**
F1 score	0.125	0.191	0.117	0.259	0.208	0.250	0.207	0.233	0.203	0.194	0.184	0.238	0.232

Table 3. Results for disease terms using TF-IDF feature selection.

Classifiers	Random forest			Decision Tree			MLPClassifier			MLkNN			
Problem transformations	BR	LP	CC	BR	LP	CC	BR	LP	CC	k=20	k=30	k=10	k=5
Accuracy	0.088	**0.126**	0.088	0.082	0.094	**0.113**	0.132	**0.157**	0.126	0.088	0.088	**0.107**	0.082
F1 score	0.106	0.160	0.090	0.256	0.191	0.257	0.175	0.217	0.171	0.098	0.091	0.128	0.163

Table 4. Results for disease terms using bag of words feature selection.

3.1.2 Neural Network Results. For the neural network method, two models were trained - (i) BiLSTM and (ii) CNN. The embedding dimension in both cases is 300, where the input sequences are kept as 200 for which the results are discussed. The models are also retrained for a sequence length of 150 and 250 to check whether the accuracy increases or decreases. The results of the models are presented in Table 5. The number of epochs is kept high (200) because of a large number of labels.

	Validation accuracy	Sequence Length
BiLSTM	***0.177***	200
	0.168	150
	0.099	250
CNN	0.31	200
	0.283	150
	0.335	250

Table 5. Results for deep neural networks for predicting disease terms.

From Table 6, it can be seen that the CNN outperforms all the baseline methods and the BiLSTM. The BiLSTM has a lower accuracy than the highest baseline MLP classifier. One explanation would be that since the RNN treats the input as a sequence, and its a very long sequence, as it goes ahead working on word after word, it forgets what happened in the words before. Even though we're using a bidirectional RNN, perhaps the left to right forgets what happened at the start, say by the time it reaches mid sequence, and the right to left forgets what it saw in the first (rightmost) terms by the time it reaches mid sequence too.

In CNNs, the convolution kernels are time-invariant, so they cannot distinguish between different parts of the sequence. As a disadvantage, they cannot easily make inferences which require using context and need to treat the input as a sequence. However, in this case, it might not be needed. Abstracts may somewhere mention a particular disease keyword which is enough on its own to detect which class it belongs to (and this detection requires no sequential treatment of input, which is similar to 'find a word').

Method	**Accuracy**
MLP Classifier (TF-IDF)	0.189
MLP Classifier (BoW)	0.157
CNN (length = 250)	**0.335**
BiLSTM (length = 200)	0.177

Table 6. Best performing methods when predicting disease terms.

3.2 Predicting Super Classes

In this experiment, we use the MeSH tree structure[3] to map the annotated disease names to their superclasses (or parent classes) of the disease terms as shown in Figure 3. Since the dataset we use involves the disease terms from the Medical Subject Headings (MeSH) ontology - which provides hierarchically-organized terminology for indexing diseases. For example, the disease term "Hemochromatosis" belongs to the superclass 'Nutritional and Metabolic Diseases [C18]' and 'Congenital, Hereditary, and Neonatal Diseases and Abnormalities [C16]'.

[3]https://meshb.nlm.nih.gov/treeView

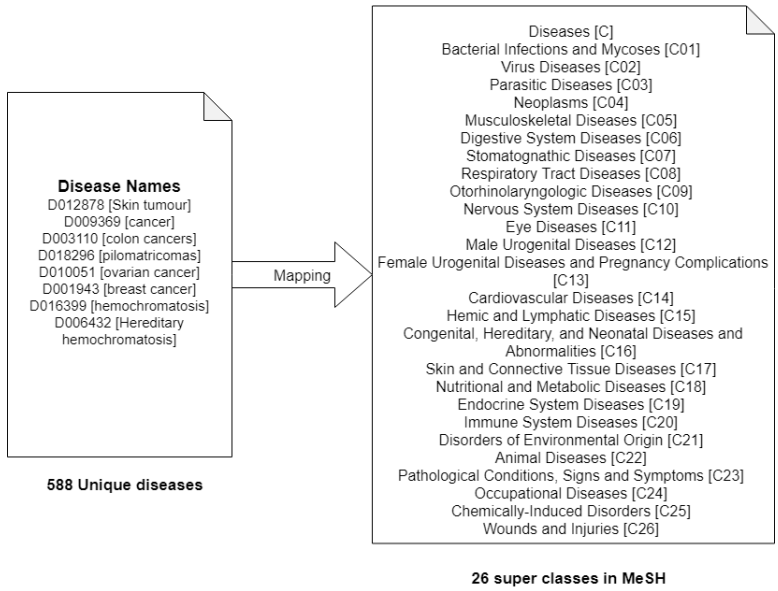

Fig. 3. Mapping superclasses in Medical Subject Headings.

We mapped the disease terms to the **26** superclasses, as depicted in Figure 3. However, the query returned only 48 classes. The additional 24 additional classes that are returned as a result of the query are mentioned below:

`'Physical Phenomena', 'Genetic Phenomena', 'Population Characteristics', 'Nonsyndromic sensorineural hearing loss', 'Diagnosis', 'Physiological Phenomena', 'Psychological Phenomena', 'Cell Physiological Phenomena', 'Investigative Techniques', 'Biological Phenomena', 'Behavior and Behavior Mechanisms', 'Immune System Phenomena', 'Reproductive and Urinary Physiological Phenomena', 'Fluids and Secretions', 'Cells', 'Health Occupations', 'Environment and Public Health', 'Tissues', 'Mental Disorders', 'Behavioral Disciplines and Activities', 'Musculoskeletal and Neural Physiological Phenomena', 'Therapeutics', 'Health Care Quality, Access, and Evaluation', 'Natural Science Disciplines'.`

This happens due to the fact that one disease term is sometimes present under more than one parent class. For example the terms 'Genetic Phenomena (ID: [G05])' and 'Pathological Conditions, Signs and Symptoms [C23]' are superclasses itself for the disease name 'Chromosome Aberrations' as can be seen in https://meshb.nlm.nih.gov/record/ui?ui=D002869.

As a result of extracting the super classes for disease terms, the number of labels is reduced to 48. In this experiment we perform text classification for these 48 labels.

3.2.1 Multi-label Classification. The results for the multi-label classification are present in Table 7. Here the features were extracted using TF-IDF. Another set of features were extracted using the bag of word representation; the results are shown in the Table 8. The maximum number of features for both the feature extraction techniques is kept to be 200. The Label Powerset transformation gives better performance for all three baseline classifiers. Overall, the Random forest classifier has the highest accuracy with problem transformation of Label Powerset.

In this setup, the feature extraction using the bag of words technique has a better performance when compared to TF-IDF. An explanation behind this would be that in a random forest classification method, when given a set of features and labels, it creates random subsets of features. Using these subsets, the algorithm builds decision trees, and then it makes predictions.

Classifier	Random forest			Decision Tree			MLPClassifier			MLkNN			
Problem Transformation	BR	LP	CC	BR	LP	CC	BR	LP	CC	k=20	k=30	k=10	k=5
Accuracy	0.181	**0.348**	0.174	0.148	**0.271**	0.135	0.232	**0.348**	0.335	0.303	0.284	**0.316**	0.297
F1 score	0.605	0.642	0.572	0.676	0.632	0.662	0.669	0.712	0.668	0.670	0.669	0.681	0.683

Table 7. Results of predicting super class using TF-IDF feature selection.

	Random forest			Decision Tree			MLPClassifier			MLkNN			
	BR	LP	CC	BR	LP	CC	BR	LP	CC	k=20	k=30	k=10	k=5
Accuracy	0.168	**0.368**	0.200	0.116	**0.329**	0.168	0.258	**0.342**	0.303	**0.148**	0.097	0.142	0.142
F1 score	0.605	0.649	0.554	0.641	0.657	0.630	0.668	0.672	0.674	0.520	0.510	0.521	0.531

Table 8. Results of predicting super class using Bag of words feature selection.

3.2.2 Neural Network Results. For the neural network method, two models were trained - (i) BiLSTM and (ii) CNN. The dimension of the embedding matrix in both cases is 300, where the input sequences of the abstract are kept as 200 for which the results are discussed. The models are also retrained for a sequence length of 150 and 250 to check whether the accuracy increases or decreases. The models are trained using the Adam optimizer [13]. The results are shown in the Table 9. The number of epochs is kept to 65 as the number of labels is not high as in the previous setup.

	Validation Accuracy	**Sequence Length**
BiLSTM	0.623	200
	0.628	150
	0.697	250
CNN	***0.837***	200
	0.76	150
	0.803	250

Table 9. Results for deep neural networks for predicting super class.

Here both the BiLSTM and CNN outperform all the baseline classifiers. Overall, CNN has the highest performance accuracy with the sequence length of 200. Another interesting observation is that when the sequence length is increased to 250, the accuracy of the BiLSTM increases, whereas one would think it should decrease because LSTMs tend to forget what happened previously or ahead when the sequence length is increased.

Method	Accuracy
Random Forest and MLP Classifier (TF-IDF)	0.348
Random Forest (BoW)	0.368
CNN (length = 200)	**0.837**
BiLSTM (length = 250)	0.697

Table 10. Best performance among the different methods when predicting super classes.

4 CONCLUSION

In this manuscript, we focus on the research problem of automatically predicting experimental metadata using scientific publications. This is done to tackle the problem of poor quality metadata. To make data reusable, we need to assess the quality of metadata, and we hypothesize that using scientific publications for prediction of experimental metadata would help with increasing this quality.

In this work, we only focus on the prediction of metadata key type 'disease'. We performed two different types of experiments (i) predicting disease names and (ii) predicting superclasses of disease names (refer Section 3). In the first experiment, we found that CNN performed best with ≈30% accuracy, but when it comes to predicting the superclasses, CNN gave a very high accuracy of ≈80%. This is a promising result, which can be employed when we try to predict more than one metadata key type. Since we used a limited number (793) of abstracts; therefore, to predict the disease names, we did not get high accuracy in the first experiment. For predicting a large number of labels, we need a higher number of abstracts for the accuracy to improve.

We used baseline multi-label classification with two different feature extraction techniques - a bag of words and TF-IDF. We observed that the TF-IDF feature extraction technique worked better when predicting the disease terms, but the bag of words technique worked better when predicting superclasses (refer Section 3). From the results, we saw that rather than following the traditional methods of extracting features in the dataset and then select a machine learning model, if we use a pre-trained word embedding, we get better results. We use a pre-trained word embedding because it contains global information about a word and training a word embedding from scratch requires a large amount of data.

A major limitation to this problem is the availability of a large gold-standard dataset on which a model could be trained. We need a large annotated dataset which could be used to train a model which could then be used to predict metadata keys and values. The corpus that was used for this project had a limited amount of abstracts with only disease mentions that were annotated - which is the reason the poor results for predicting disease terms. As a part of future work, we plan to extend our model to predict for other metadata key types such as organism, tissue, cell line etc. A more context-specific word embedding could be used in the deep learning architecture that we used to check whether this would help improve the predictions of the metadata. Moreover, we plan to use transfer learning [10], to train the model on a large annotated corpus such as the data present in BioASQ challenge[4] and test in on the NCBI disease corpus.

[4]http://bioasq.org/participate/challenges

REFERENCES

[1] Tanya Barrett, Stephen E Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F Kim, Maxim Tomashevsky, Kimberly A Marshall, Katherine H Phillippy, Patti M Sherman, Michelle Holko, et al. 2012. NCBI GEO: archive for functional genomics data sets—update. *Nucleic acids research* 41, D1 (2012), D991–D995.

[2] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research* 3, Feb (2003), 1137–1155.

[3] Christine L Borgman. 2012. The conundrum of sharing research data. *Journal of the American Society for Information Science and Technology* 63, 6 (2012), 1059–1078.

[4] Alvis Brazma, Pascal Hingamp, John Quackenbush, Gavin Sherlock, Paul Spellman, Chris Stoeckert, John Aach, Wilhelm Ansorge, Catherine A Ball, Helen C Causton, et al. 2001. Minimum information about a microarray experiment (MIAME)—toward standards for microarray data. *Nature genetics* 29, 4 (2001), 365.

[5] François Chollet et al. 2015. Keras. https://keras.io.

[6] Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*. ACM, 160–167.

[7] Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics* 47 (2014), 1–10.

[8] Ron Edgar, Michael Domrachev, and Alex E Lash. 2002. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic acids research* 30, 1 (2002), 207–210.

[9] Rafael S Gonçalves, Martin J O'Connor, Marcos Martínez-Romero, John Graybeal, and Mark A Musen. 2017. Metadata in the BioSample Online Repository are Impaired by Numerous Anomalies. *arXiv preprint arXiv:1708.01286* (2017).

[10] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

[11] Sharona Hoffman and Andy Podgurski. 2013. The use and misuse of biomedical data: is bigger really better? *American journal of law & medicine* 39, 4 (2013), 497–538.

[12] Wei Hu, Amrapali Zaveri, Honglei Qiu, and Michel Dumontier. 2017. Cleaning by clustering: methodology for addressing data quality issues in biomedical metadata. *BMC Bioinformatics* 18, 1 (18 Sep 2017), 415. https://doi.org/10.1186/s12859-017-1832-4

[13] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[14] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[15] AZ Stuti Nayak, Amrapali Zaveri, and Michel Dumontier. 2018. Quality Assessment of Biomedical Metadata Using Topic Modeling. In *2nd workshop on Semantic Web solutions for large-scale biomedical data analytics (SeWeBMeDA)*.

[16] Stuti Nayak, Amrapali Zaveri, Pedro Hernandez Serrano, Rachel Cavill, and Michel Dumontier. [n.d.]. A Machine Learning approach towards Quality Assessment of Biomedical Metadata. *Journal of Biomedical Semantics* ([n. d.]). Submitted November 2018.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[18] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[19] V. Singh. 2017. Replace or Retrieve Keywords In Documents at Scale. *ArXiv e-prints* (Oct. 2017). arXiv:cs.DS/1711.00046

[20] Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J.G. Gray, Paul Groth andCarole Goble andJeffrey S. Grethe, Jaap Heringa, Peter A.C 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok, Joost Kok, Scott J. Lusher, Maryann E. Martone, Albert Mons, Abel L. Packer, Bengt Persson, Philippe Rocca-Serra, Marco Roos, Rene van Schaik, Susanna-Assunta Sansone, Erik Schultes, Thierry Sengstag, Ted Slater, George Strawn, Morris A. Swertz, Mark Thompson, Johan van der Lei, Erik van Mulligen, Jan Velterop, Andra Waagmeester, Peter Wittenburg, Katherine Wolstencroft, Jun Zhao, and Barend Mons. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data* 3 (2016). https://doi.org/10.1038/sdata.2016.18

[21] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. 2018. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine* 13, 3 (2018), 55–75.