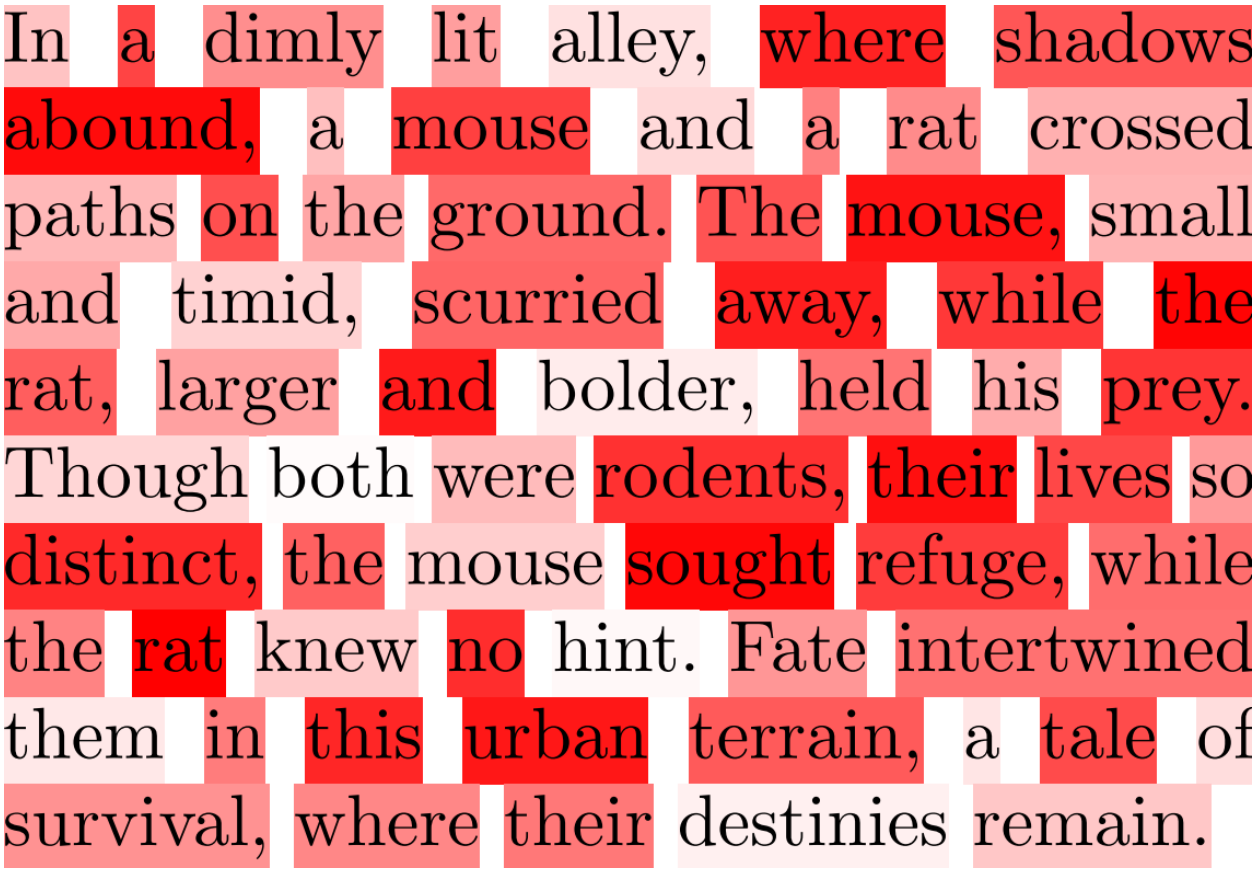


AI-Based Toxicity Prediction as an Alternative to Animal Testing

A Transformer-Based Deep Learning Approach to Toxicity Prediction

Master's thesis in Engineering Mathematics and Computational Science

Mercedes Dalman

DEPARTMENT OF MATHEMATICAL SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2023
www.chalmers.se

Master's thesis 2023

AI-Based Toxicity Prediction as an Alternative to Animal Testing

A Transformer-Based Deep Learning Approach to Toxicity Prediction

MERCEDES DALMAN

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mathematical Sciences
Systems Biology and Bioinformatics
Chalmers University of Technology
Gothenburg, Sweden 2023

AI-Based Toxicity Prediction as an Alternative to Animal Testing
A Transformer-Based Deep Learning Approach to Toxicity Prediction
MERCEDES DALMAN

Supervisors:
Erik Kristiansson, Department of Mathematical Sciences
Mikael Gustavsson, Department of Mathematical Sciences
Examiner:
Erik Kristiansson, Department of Mathematical Sciences

Master's Thesis 2023
Department of Mathematical Sciences
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Self-attention visualised on a poem of a rat and a mouse written by OpenAI ChatGPT [1].

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2023

AI-Based Toxicity Prediction as an Alternative to Animal Testing
A Transformer-Based Deep Learning Approach to Toxicity Prediction
MERCEDES DALMAN
Department of Mathematical Sciences
Chalmers University of Technology

Abstract

In recent years, there has been a significant increase in the use of chemicals in our environment due to growing demand and consumption. Consequently, large-scale chemical regulation based on toxicological assays has been implemented to prevent exposure-related consequences for nature and human health. Historically, animal-based assays have been used for this purpose. However, there is now an increasing demand to replace these animal-based assessment methods with computer-based alternatives. Despite previous attempts to develop computer-based models, these models have proven to be unreliable and inaccurate, leading to a decrease in interest. Therefore, there is a pressing need to develop new computer-based models for toxicity assessment. Here, the introduction of deep learning models, particularly transformer architecture, has the potential to revolutionise the field. Deep neural networks have demonstrated the ability to handle complex and high-dimensional problems, surpassing older modelling techniques. Moreover, as the transformer has shown promise in handling chemical structure information, there is growing interest in its usage in the field of environmental toxicity assessment. The aim of this project was hence to explore the potential of transformer-based deep neural network models for the purpose of toxicity assessment.

For this project, a subset of rat and mice *in vivo* toxicity assay data associated with EC_{50} and LOEC measurements, as well as different administration routes, were utilised. Here, three sets of data were analysed, each distinguished by the hazards: acute toxicity, carcinogenicity, or reproductive toxicity. The first type of model, the single-DNN model, was created for each data set separately. Subsequently, these models were expanded to the multiple-DNN model, able to handle all three data sets simultaneously. For all models, a pre-trained RoBERTa transformer was utilised to interpret canonicalised SMILES representation of chemical structures, with the performance then evaluated through repeated 10-fold cross-validation. Principal Component Analysis demonstrated that the transformer could identify patterns in chemical structures related to toxicity. Moreover, the study found that the single-DNN model outperformed the multiple-DNN model in all trials, likely due to the latter's increased complexity. All models exhibited leniency towards chemicals with low measured concentrations, and to mitigate this problem, a more stringent loss for lower concentrations was suggested. Overall, this project demonstrated the potential and effectiveness of transformer-based computer models for toxicity assessment, showcasing the versatility of this technology for addressing a broad range of toxic hazards.

Keywords: environmental risk assessment, SMILES, RoBERTa, deep learning, artifical intelligence, transformer, toxicity

Acknowledgements

This work would never have been possible without the help and support of the brilliant people around me. Here, I would first like to thank my main supervisor and examiner Erik Kristiansson, who has been a big contributor to the ideas and knowledge that has guided me along this journey. Moreover, I would like to thank my co-supervisor Mikael Gustavsson, who with his expertise in the world of ecotoxicology has helped provide the basis on which this work stands upon. Furthermore, I would especially like to thank Styrbjörn Käll for his help and engagement in this project, and for answering all my questions. Finally, I would like to thank my friends and family for their patience and support; without their love, it is safe to say that this work would never have seen the daylight.

Mercedes Dalman, Gothenburg, May 2023

Contents

List of Figures

List of Tables

1

Introduction

The swift progress of industrial and societal advancements has led to an escalated threat of chemical exposure to both nature and human health [2]. To mitigate this risk, chemicals are now mandated to undergo rigorous testing for potential hazardous effects, such as toxicity, prior to being introduced to the market [3], [4]. In relation to this, the European Union's Registration, Evaluation, Authorisation, and Restriction of Chemicals (REACH) regulation requires companies to register chemicals and provide associated risk assessments, which include information on toxicity and ecotoxicity [5]. Here, the foundation of these risk assessments lies within the Environmental Risk Assessment (ERA), which predominantly encompasses toxicological assays.

Toxicological assays are conventionally conducted through a variety of both *in vivo* and *in vitro* animal experiments [6]. However, these tests have been found to be both time and cost-intensive, in addition to being of ethical concern and questionable reliability. As a response to this, there has been a growing interest in developing alternative testing methods, with the EU advocating for the use of *in silico* models. However, despite this, there has been a decline in the number of computational models used in recent years, with only a small number currently in use [4]. Moreover, the primary cause of this reduced interest in computational models is the substantial variations in output and performance due to differences in chemical structural information handling [7].

Consequently, computational methods require further research and development to enhance their accuracy and reliability before they can completely supplant biological tests. Here, Artificial Intelligence (AI) has emerged as a promising prospect [3]. With its cost-effectiveness and ability to process vast amounts of intricate data, AI can outperformed traditional modelling methods and transformed the computational realm. Nowadays, AI is applied across various scientific fields, including biology and medicine [8], [3]. Moreover, the introduction of the revolutionising transformer architecture offers a potential solution to the challenge of handling chemical structural information [9]. Against this backdrop, this thesis aims to expand on previous endeavours to advance the exploration and development of AI-based models for toxicity evaluation.

1.1 Aims and Scope

The objective of this study is to create and evaluate AI-based models that can forecast mammalian chemical toxicity. Here, the data set used will include rat and mice toxicity assays for a diverse range of administration routes but be restricted to only EC_{50} and LOEC measurements. Initially, three distinct models will be developed, each exclusively focused on predicting one of the following toxic effects: acute toxicity, carcinogenicity, and reproductive toxicity. Subsequently, the goal is to merge these models into a single model that can manage all three toxic effects. Moreover, the models will be founded on a combination of transformer architecture and a Deep Neural Network (DNN). In this system, the transformer component will be responsible for processing and transforming chemical structural inputs in the form of SMILES. Furthermore, the DNN component will utilise the transformer's output, together with additional metadata, to perform the final toxicological assessment. In summary, the research aims addressed by this study are as follows:

- Develop and analyse a transformer-based model able to perform toxicological predictions for one specific toxic effect.
- Develop and analyse an expanded model able to handle and predict various toxic effects simultaneously.

This study had access to a substantial amount of mammalian data associated with different species and outcomes. However, due to time and feasibility constraints, the investigation was restricted to the categories: rats, mice, EC_{50}, LOEC and a few select administration routes. In addition, due to the same reasons, only one type of transformer, the RoBERTa transformer, was evaluated and integrated with a basic feed-forward neural network.

2

Theory

The next chapter aims to explain the theory necessary to comprehend the crucial procedures in the project, with specific details provided as necessary. Additional sources are cited for further information, and the methods for model building and training procedures used in the project will be covered in the next chapter.

2.1 Environmental Toxicology

In the scientific field of environmental toxicology, both the analysis of potential health risks, as well as the management and protection measures, associated with hazardous chemicals are covered [10], [11], [12]. Toxic chemicals can cause acute or chronic, such as cancer-related, health effects, with dosage being a significant factor in what outcome exposure will have. Laboratory toxicity assessments, made through a combination of *in vivo*, *in vitro*, and *in silico* testing procedures, are used to determine hazardous properties and dosage for different chemicals. Results are often presented as effect concentrations such as EC_x (such as 50% effect, EC_{50}) and Lowest Observed Effect Concentration, LOEC, which indicate the concentrations at which a certain percentage of the test population experiences health hazards. A dose-response curve is commonly used to visualize these results, with a general example of such a curve, where EC_{50} and LOEC values have been marked out, shown in Figure 2.1.

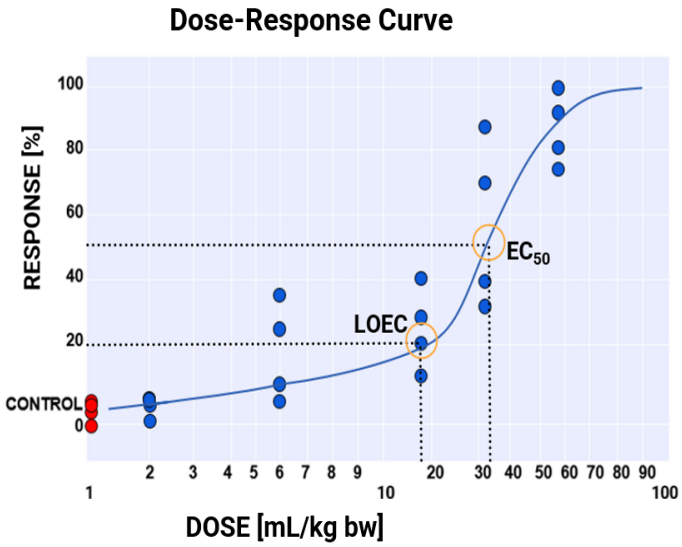


Figure 2.1: A general example of a dose-response curve, in which values for LOEC (Lowest Observed Effect Concentration) and EC_{50} (50% effect concentration) have been marked out.

Within the EU, the REACH Regulation requires companies to conduct chemical toxicity assessments before producing, importing or selling a chemical [13]. The European Chemicals Agency (ECHA) is responsible for registering and assessing the risks associated with the chemicals, as well as determining the need for restrictions or bans. Moreover, the regulation obligates companies to identify and manage potential hazards associated with the chemicals they market or produce by providing guidelines for data collection and toxicity assessment.

2.2 Deep Neural Networks

Deep Learning, based on Artificial Neural Networks (ANNs), has been shown to be a useful tool for chemical toxicity analysis [3], [14], [15]. ANNs, often in the form of Deep Neural Networks (DNNs), consist of multiple layers of interconnected nodes, forming a complex web of connections. Here, the connection between neurons is parameterised by weights and biases, making it essentially a weighted sum. Additionally, an activation function at each layer re-scales the signal and thereby determines to what degree the signal should be passed on to the next layer. The optimisation of this system, achieved through supervised learning with gradient descent, involves updating the weights and biases through backpropagation to minimise the error between the predicted outcomes and the measured values found in the data.

DNNs can handle large and complex data better than traditional regression techniques. However, making a DNN too large can lead to it becoming too fine-tuned to the training data, so-called overfitting, and poor performance on unseen data [16]. Hence, the performance of a DNN depends heavily on hyperparameters like the

number of hidden layers and neurons in each layer. To address this, dropout, where a percentage of neurons in the model are inactivated, and freezing some layers during training can reduce the model's sensitivity to training data. An example of a simple DNN with only an input and output layer can be seen in Figure 2.2.

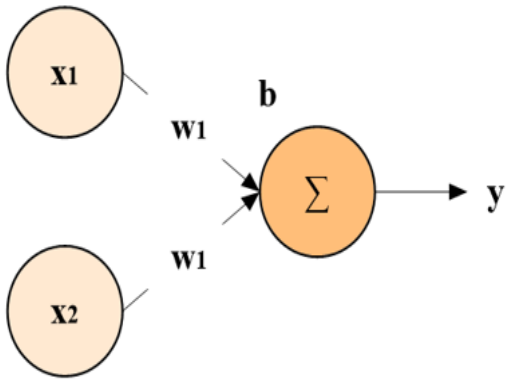


Figure 2.2: A simple illustration of a Deep Neural Network, consisting of only one input layer with two nodes, as well as an output layer with one node parameterised by one weight for each input node (w_1 and w_2), and a bias, b.

2.3 Simplified Molecular Input Line Entry System (SMILES)

Chemicals' properties and their potential health hazards are related to their molecular structures, making efficient utilization of structural information critical for *in silico* methods for toxicity prediction [9]. Moreover, to process molecular structures computationally, they must be represented in a 1D sequential format while retaining important structural information found in their 3D form. Here, Simplified Molecular Input Line Entry Systems (SMILES), a sequence of letters and symbols that represent a molecular structure, are commonly used for small structures, as they are designed in such a way that they contain information on the 3D aspects of the original molecular structure they represent. Figure 2.3 provides an example of a structure with its corresponding SMILES representation [17]. In the figure, colour has been used to demonstrate which parts of the original 3D structure correspond to a certain element in the SMILES string, as well as geometrically close pairs in the structure.

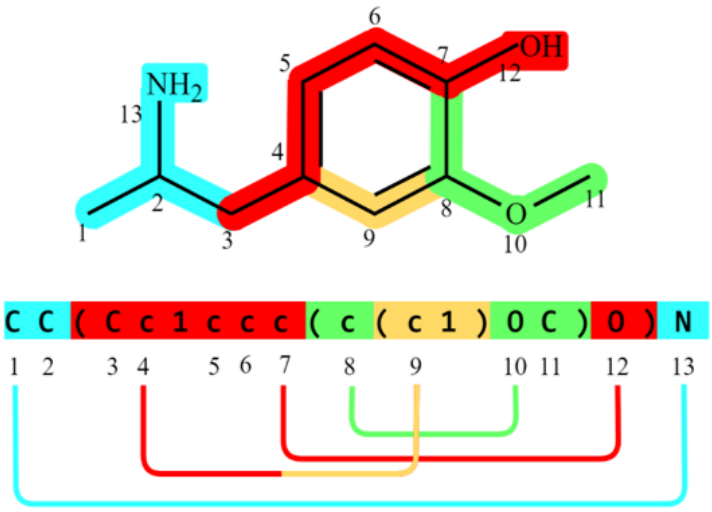


Figure 2.3: An example of a molecular structure together with its SMILES representation. Each part of the original structure corresponding to a certain part of the SMILES string has here been assigned the same colour, as well as geometrically close pairs in the SMILES string.

An issue is that SMILES representations can have multiple versions for the same chemical, depending on how the atoms have been numbered in the structure [8]. This problem is usually resolved by running SMILES through canonicalisation algorithms, which always follow the same specific rules for generating SMILES. However, different databases have their own versions of these algorithms, making it important to use the same algorithm to ensure the uniqueness of SMILES for a specific chemical.

2.4 Natural Language Processing and The Transformer Architecture

Computational scientists historically faced an issue with computers' inability to process textual inputs like SMILES [9]. To combat this, Natural Language Processing (NLP) was introduced, where recently the Transformer architecture has revolutionised the field with recent technological breakthroughs such as GPT [18]. Transformers rely on the self-attention mechanism to learn and process text into a high-dimensional numerical output processable in a neural network [9]. Contrary to older NLP algorithms where text was processed in a sequential manner, leading to issues both with speed and memory usage, transformers utilise the semantic meaning found in the geometrical distances between elements in an input string. This makes them suitable for processing SMILES, where the elements in the string correspond to actual 3D elements in molecular structures, and the distances between these elements are important.

More specifically, transformers are composed of encoders and decoders, with the encoder performing input encoding and the decoder predicting the most probable

translation [19], [20]. Input strings are pre-processed by a tokenizer, which split them into smaller elements, or tokens, according to some pre-existing vocabulary, and each token is then given corresponding input and positional embeddings before being transformed by the encoder. The task of the encoder is then to shift, or transform, these input-embeddings in a manner which takes context (such as if a certain word/token is more important for understanding the meaning of the sentence than others) into consideration. Here, the positional embeddings provide context for the encoder's interpretation, as input embeddings lack positional information. The output embeddings, unique for each token, are primed for the decoder's final translation.

The encoder in the transformer system uses self-attention to take context into consideration [19], [20], [9]. Self-attention computes the relative importance of input embeddings to one another based on their spatial relationship or distance. For example, for an input sentence, some words will be more closely related to each other than others (e.g. words describing nature, "tree", "river", "soil"). The closeness in the relationship between these words will translate to their input embeddings lying closer in high-dimensional space. Furthermore, the self-attention mechanism then essentially computes the dot product between each input embedding, meaning that the embeddings which lie close in high dimensional space will be associated with a high dot product. More specifically, each high-dimensional input token is divided into query, key, and value vectors to capture unique aspects of the input token. The dot product between each token's key, query, and value vector is computed to determine the final weight or attention between each token. To get these key, query, and value vectors, each input embedding is multiplied with the respective weight matrices W_K, W_Q, and W_V. Then the final weight (or attention) calculated between each token is described by a function, see Equation 2.1, of all three of these dot products [9]. Finally, the resulting output is passed through a softmax function whose task is to crush small and negative values to 0 (indicating distant or no relationship between tokens), whilst inflating large values (indicating a strong relationship between tokens).

$$[H]Attention(Q, K, V) = softmax(\frac{QK^2}{\sqrt{d_k}})V \tag{2.1}$$

Multi-head self-attention is a technique used to improve the efficiency of the self-attention mechanism in transformers [19], [20]. It involves splitting the input into multiple sets of key, query, and value vectors, with each set being associated with its own weight matrices that capture unique aspects of the input. These multiple heads work in parallel and each performs the self-attention process described before, before finally concatenating their outputs back into vectors of the same sizes as the original input embeddings. Moreover, this technique allows transformers to find extremely complex relationships in their inputs. Additionally, in modern transformers, several encoders are stacked together in encoder blocks to further enhance their ability to capture intricate relationships.

2.4.1 BERT-Based Transformers

Low-quality data can significantly affect the output and usefulness of both traditional and AI-based models. Deep Learning models often require labelled data, but large datasets in fields such as biology may have limited labelled data, reducing the performance of traditional transformers [21], [22], [23]. To address this issue, different classes of transformers have been developed for specific tasks, including those focused on handling unlabelled data. Here, the Bidirectional Encoder Representation from Transformers, or BERT, based on masked-language modelling, has become popular in text classification and language analysis due to its bidirectional pre-training and fine-tuning system, allowing it to handle a broad range of problems.

The BERT model's ability to handle large unlabelled datasets is due to its unsupervised, bidirectional pre-training [21], [22], [23]. Here, BERT uses Masked Language Modelling (MLM) to give the model a general understanding of the language of interest, through a percentage of tokens being randomly masked, and BERT then being tasked with predicting the masked word from surrounding words. In pre-training, Next Sentence Prediction (NSP) is also used to understand the relationship between sentences in the input data. Once pre-training is complete, BERT is ready for fine-tuning, where it becomes fine-tuned to a specific task/problem involving the language at hand. Fine-tuning is done through supervised learning, but BERT's understanding of the language from pre-training means that only a small set of labelled data is required. Through fine-tuning, BERT can perform a wide range of tasks, such as text prediction, summarization, and text generation. BERT also implements large-scale parallelization to make use of large amounts of data within an efficient time frame. When using BERT-based transformers, inputs are tokenized and a CLS (classification) token is added as the first token. Moreover, the tokens are then truncated or padded to a fixed size and matched to their associated embeddings. The transformer is then trained through multi-head self-attention, as mentioned previously, but without using decoders. Specific to BERT is the training of an additional CLS token that summarizes all the information BERT learns through fine-tuning, which can be used as a replacement for all the output tokens of a sentence.

Since the introduction of BERT, improvements have been made to the model, including the development of the Robustly Optimized BERT Approach, or RoBERTa, to address speed-related issues in pre-training [24]. RoBERTa eliminates the Next Sentence Prediction (NSP) technique used in BERT while outperforming it in terms of performance. RoBERTa is used in various NLP tasks, including the ChemBERTa model, which achieved a vocabulary size of about 8 thousand tokens through pre-training with large datasets of canonicalized SMILES, which were canonicalized using rdkit's canonicalization algorithm [25]. In this project, a pre-trained ChemBERTa will be used instead of training a new RoBERTa model. Finally, a general illustration of a RoBERTa transformer, including tokenization and CLS-token extraction, is shown in Figure 2.4.

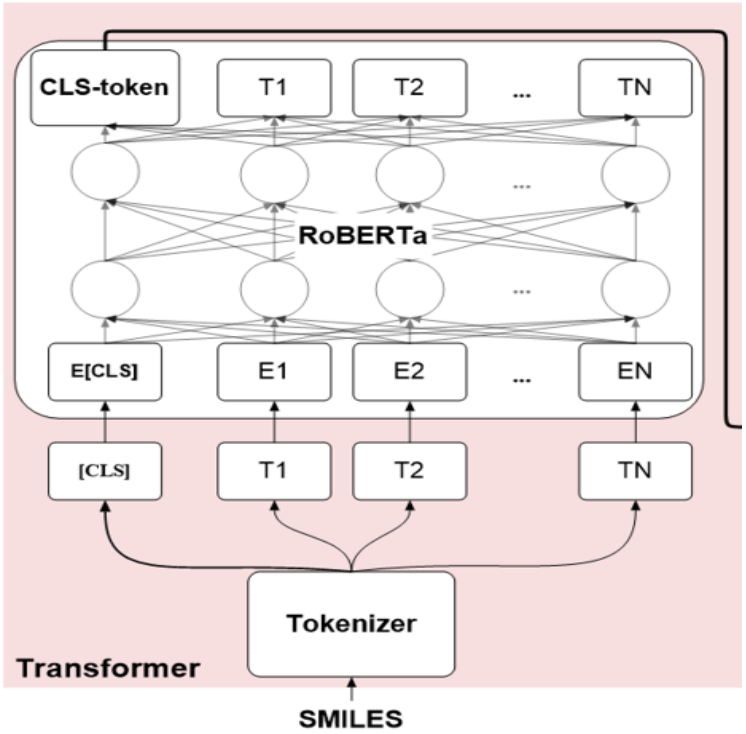


Figure 2.4: A simplified representation of a tokeniser connected to a RoBERTa transformer. The input string is first segmented into predetermined tokens (T_N) by the tokeniser, before being fed to the transformer. Then, each token is assigned an input and positional embedding (E_N). Using self-attention, the transformer adjusts these embeddings based on the problem at hand. During this process, the CLS token is trained, which then can be extracted as output from the transformer.

3
Methods

Presented in this section are the methods used together with information specific to the implementation of these methods. However, the purpose of this section is not to provide details on or understanding of specific concepts. For this, the reader is instead directed to the previous chapter.

3.1 *In Vivo* Toxicological Data

In this project, the *in vivo* data used, especially that from the RTECS dataset, is characterized by a large number of associated variables. These variables include various species, assay outcomes, and administration routes. hence, to simplify the analysis, only a limited subset of this data was used. Subsequently, the relevant variables and categories for this subset are listed in Table 3.1.

Table 3.1: Variables in *In Vivo* Data with Categories

Variable	Categories
Species	rat, mouse
Administration Routes	intraperitoneal, oral, intravenous, subcutaneous, dermal, intracerebral, intramuscular, parenteral, intratracheal, intraspinal, implant, other routes
Assay Outcomes	EC_{50}, LOEC
Data Sets	acute toxicity, carcinogenicity, reproductive toxicity

Notably, the "other routes" category in the administration route variable in the *in vivo* data used contains chemical routes of administration with only rare occurrences. The assay data, in its entirety, also includes a variety of concentrations in different units, however only those that can be converted to mL/kg body weight were kept during pre-processing. Moreover, although experiment duration was considered a potentially significant variable, it was not included due to the lack of available data. All data points in the *in vivo* data are associated with a canonicalised SMILES string obtained through the rdkit canonicalization function, as well as a CAS number. Furthermore, the distribution of data in the three different *in vivo* toxicological datasets used in this project varies significantly. The majority of the data, approximately 88%, corresponds to acute toxicity measurements, whereas reproductive toxicity and carcinogenicity tests only make up about 9% and 3% of the data, respectively. This information is visualized in Figure 3.1.

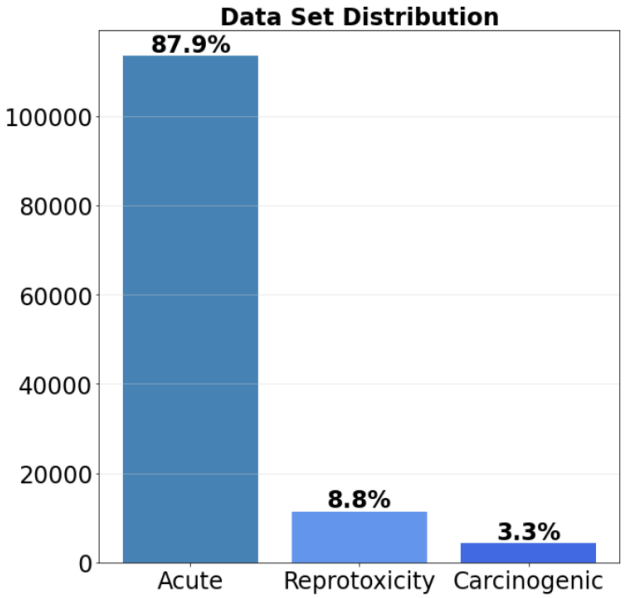


Figure 3.1: Relative amounts of the three different data sets in *in vivo* animal test data.

Moreover, the Venn diagram in Figure 3.2, depicting the number of unique SMILES in each data set, along with the overlap of unique SMILES between data sets, shows that as well as being the largest row-wise, the acute toxicity data set also dominates the other data sets in terms of unique SMILES. About 78,000 unique SMILES are found exclusively in the acute toxicity data set, whilst the other two data sets only have about 1,600 to 1,700 SMILES uniquely attributed to them.

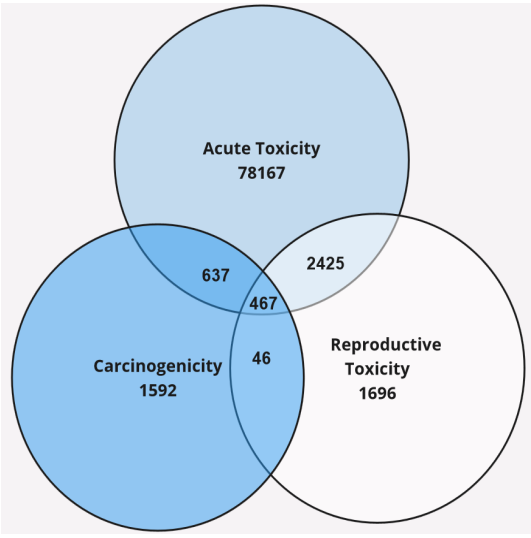


Figure 3.2: Venn diagram of unique SMILES, as well as their overlap, in and between *in vivo* animal test data sets.

The prevalence of certain species, administration routes, and assay outcomes in the *in vivo* data used in this project also varies a lot between data sets, where the number of SMILES associated with either the assay outcome EC_{50} or LOEC, as well as either

the species mouse or rat, is presented in Table 3.2. In the table, it can be seen that most of the data are of type EC_{50} and mouse. However, notably, the carcinogenicity and reproductive toxicity data sets are dominated by LOEC data, with the former data set only having data of this type, meaning that it is the acute toxicity data set that contributes to the majority of all data being EC_{50}. For the species variable, rats dominate in the reproductive toxicity data set, whilst the distribution of chemicals between the species is almost equal for carcinogenic data. Once again, it is therefore due to the acute toxicity data set, and its size, that the data overall are of mostly mouse assays.

Table 3.2: Distribution of Assay Outcomes and Species in *In Vivo* Data

Parameter	Acute Toxicity	Carcinogenicity	Reproductive Toxicity	Total
EC_{50}	92,823	0	20	92,843
LOEC	20,689	4,257	11,350	36,296
Mouse	81,143	2,135	3,252	86,530
Rat	32,369	2,122	8,118	42,609

Furthermore, the distribution of administration routes across the entire data is shown in Figure 3.3, where it can be seen that intraperitoneal, oral, and intravenous administration routes dominate, accounting for 34, 30, and 18% of the data, respectively.

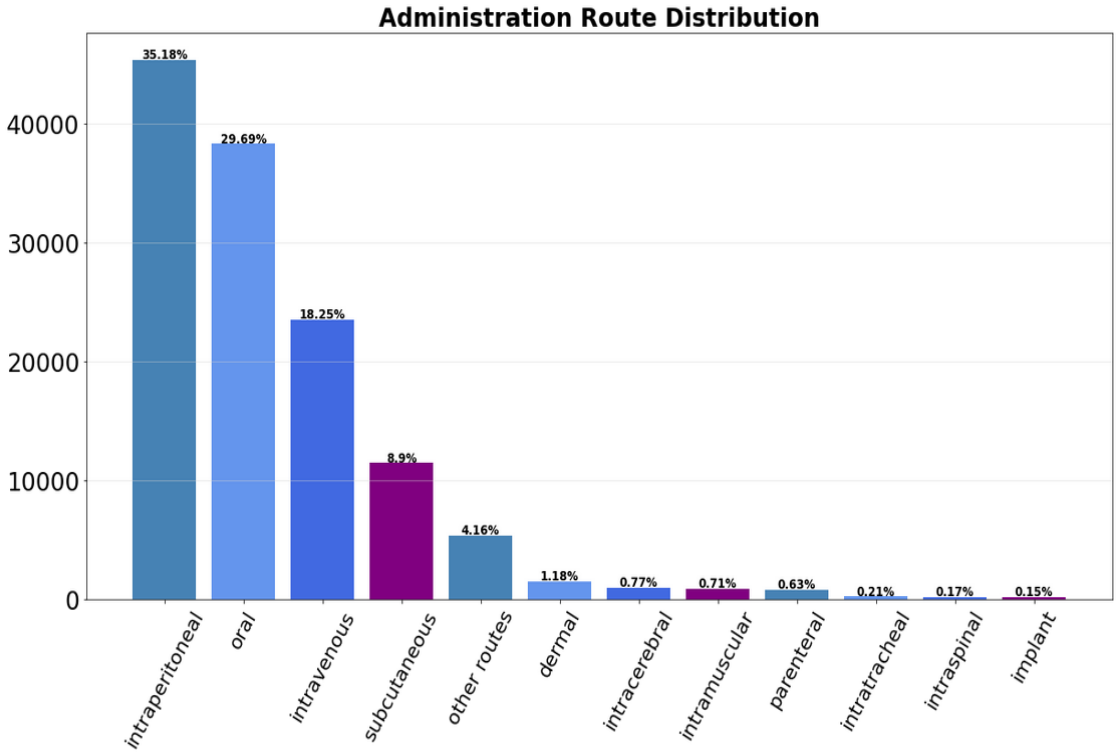


Figure 3.3: Administration route distribution in *in vivo* animal test data.

3.1.1 Pre-processing

For *in vivo* toxicological assays, all data used in this project comes from the RTECS and REACH databases, and to create the final data sets used in the project data from both databases were mixed randomly. Moreover, to reduce variance in concentration, the Log_{10} transformation was then applied to the measurements. To ascertain that there was not any variation in SMILES for the same chemicals, all SMILES were run through RDKit's canonicalization algorithm. Finally, all categorical variables (such as species, administration route and assay outcome) were one-hot encoded.

3.2 Architecture

The models made in this project employ two main components: a Natural Language Processing (NLP) transformer and Deep Neural Networks (DNNs). The first type of model which was built was the so-called single-DNN model is illustrated in Figure 3.4. Here, one ChemBERTa transformer is connected to one DNN associated with a specific data set, such as acute toxicity.

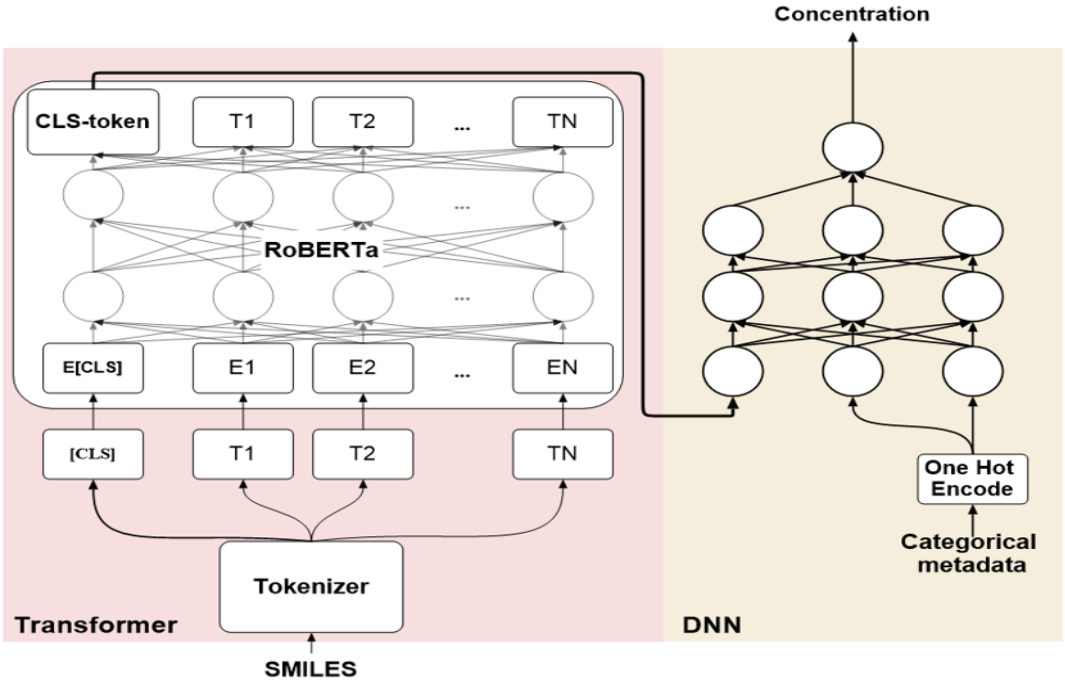


Figure 3.4: A simplified overview of the single-DNN model. The model consists of two parts: a ChemBERTa transformer and a Deep Neural Network (DNN). Here, the transformer receives the SMILES strings (textual representation) of chemicals' molecular structures as input and produces a numerical representation (CLS-token) that is then fed to the DNN. Then, the DNN utilises the transformer output, along with additional one-hot encoded metadata and measured $\log_{10}$ concentrations, to predict $\log_{10}$ concentrations.

Figure 3.5 depicts an overview of the second type of model, the so-called multiple-

DNN model, used in this project. Here, it can be seen that the ChemBERTa transformer instead is connected to several DNNs, each corresponding to their specific data set. Moreover, these DNNs work independently of one another, with the transformer being fine-tuned through backpropagation with the combined loss, or error, from each of these separate DNNs.

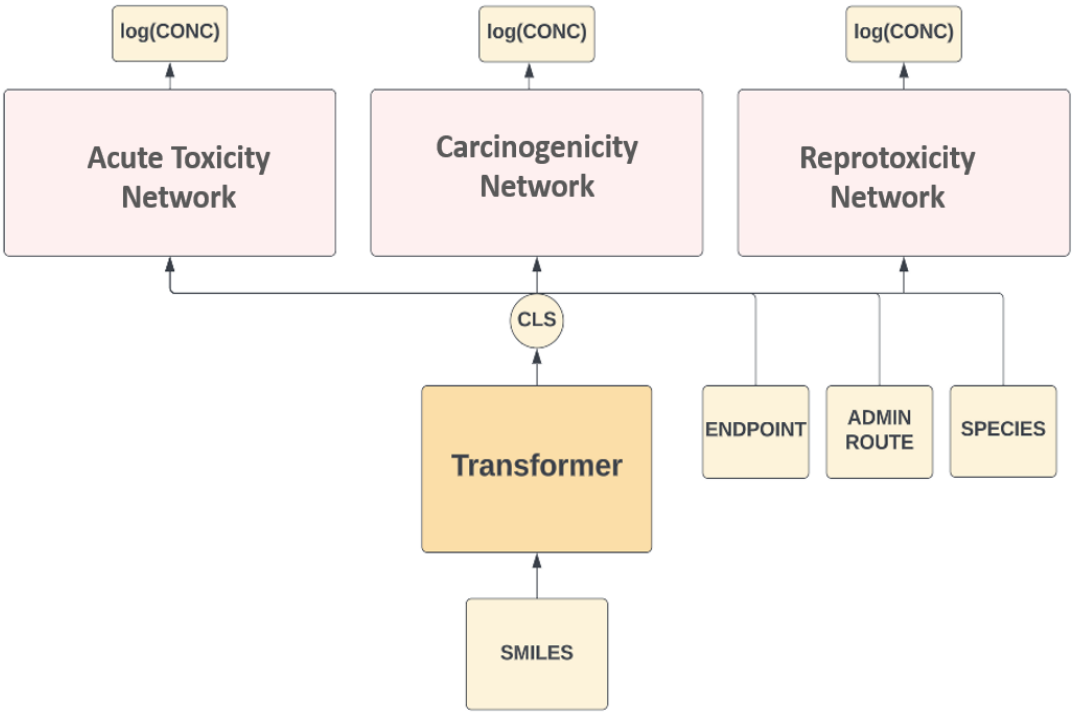


Figure 3.5: Structure of the second model used in the project. The ChemBERTa transformer is identical to what has previously been described for the first model, and each of the three networks connected to this transformer also has the same structure as the single models for each data set. Hence, the real difference between this model and the previous lies in the fact that the three data sets can in this case be analysed simultaneously through their own respective Deep Neural Networks (DNN). Moreover, these networks all work independently from one another, with the total loss (used for training) being the sum of the losses from each network.

In the case of both the single- and multiple-DNN model, the metadata consists of the categorical variables "species", "administration route", and "endpoint" (also called assay output), which has all been one-hot encoded, a CLS-token representation of a SMILES, and the 10th logarithm of the measured concentration for the specific assay. Moreover, the transformer used is a pre-trained ChemBERTa (see Theory), which task is to transform the molecular structure of a chemical, represented as a SMILES, into numerical output, the CLS token. Before being fed to the transformer, the SMILES are tokenised by a tokeniser provided by Huggingface [26]. The tokeniser used here has been pre-trained on SMILES, and is based on Huggingface's Byte-Pair. Finally, in the DNN, the input is passed through several hidden layers utilising ReLU activation functions, until it reaches a final single linear

output neuron, with the final output being the logarithm of the concentration for the desired toxicological outcome.

3.3 Training

In order to create the models used in the project, a pre-trained ChemBERTa transformer was downloaded from the Transformers library on Huggingface, whilst the DNN was created using PyTorch v1.9.0 [27]. Moreover, to train the DNN, the L1 loss function (or the Mean Absolute Error (MAE)), was used, together with the stochastic gradient descent optimiser AdamW. A number of hyperparameter sweeps were then used to set the DNN's learning rate, dropout probability, number of hidden layers and number of hidden neurons. These hyperparameter sweeps were performed using the so-called Bayesian hyperparameter search [28]. This optimisation method utilises a Bayesian approach, in which, compared to grid searches where all possible combinations of hyperparameter values are tested, the choice in parameter settings for a specific run is influenced by information gained in previous runs (that is, which settings that gave performance). Hence, the settings tested will be the ones which have the highest probability of being good choices. In the case of Bayesian searches, there is no obvious endpoint for the search (compared to a grid search which ends when all combinations have been tested), and in this case, the decision was made to end the search when around 400 runs had been performed. Moreover, all training was performed on Alvis OnDemand using A100 GPUs and logged on the machine learning platform Weights & Biases. Also, a linear warmup with 100 warmup steps, during which the learning rate was linearly increased from a low level to the set level, was implemented in each training session to reduce some early learning volatility [29].

Furthermore, in training the model, a 10-fold cross-validation method was employed, and a sequential sampler was used for both the training and test sets. A weighted random sampler was also considered for the training set due to some imbalance in the data, such as in terms of species and assay outcome. However, the use of a weighted random sampler did not significantly improve the model's performance, and a sequential sampler was necessary when creating a model with multiple DNNs. Therefore, the decision was made to use a sequential random sampler. Moreover, for the multiple-DNN model, all three data sets' test and training sets were concatenated after each data sets individual test-train split (where each data set was randomly split into 10 folds of equal size). As the acute toxicity data set is much larger than the other two, special care was needed here to make sure that the smaller sets were always represented in the training set. Specifically, the carcinogenicity and reproductive toxicity training sets were upsampled in such a way that they each would constitute 25% of the total, concatenated training set (before upsampling). This upsampling was performed by randomly drawing, with replacement, SMILES from the training sets. After the two training sets had been upsampled, they were concatenated with each other and the acute toxicity training set. Finally, the entire training set, as well as the test set, were shuffled.

3.3.1 K-Fold Cross-Validation

Figure 3.6 shows the k-fold cross-validation technique, which involves randomly dividing the data set into k equal folds [30]. For the case of this project, the data sets were split so that there was no SMILES overlap between the test and training sets. After the split, one fold is used as the test set, and the other k-1 folds are used for training. The model's final performance is evaluated by predicting the test set, resulting in a validation performance/loss for this fold. This process is repeated until all folds have been used as the test set once, resulting in k validation performance calculations. The number of folds used is typically 5-10, and in this project, 10 folds were used. Moreover, to reduce the potential for bias introduced during the initial data split and fold assignment, the k-fold cross-validation was repeated 10 times for each test, and the final model performance assessment was calculated as the average of these performances.

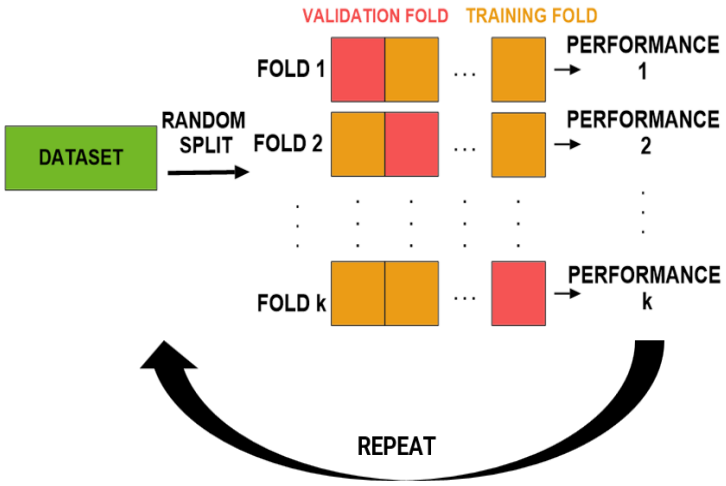


Figure 3.6: Illustration of K-fold cross-validation.

3.3.2 Median Loss, Best Average Loss and Loss$_{MEAN}$

During training, it was important to avoid bias introduced by having the same chemical appear multiple times in the test set. If a chemical appeared multiple times, it would have a larger impact on the average loss than a chemical that only appeared once. To address this issue, the concept of the best average loss was implemented. To calculate the best average loss, the median loss for each unique chemical in the validation set was calculated at each epoch. By taking the median for each unique chemical, the output was less influenced by certain chemicals appearing often in the data. The mean was then taken of all the median values to achieve the total average loss for the epoch, and the lowest total average loss achieved during the entire run was considered the best average loss.

Moreover, the lack of reference for losses was a problem during the project. Without a reference, it was difficult to determine what constituted a low value for losses. To address this issue, the concept of loss$_{MEAN}$-value was implemented. This value

represents the loss if the model were to predict the mean of the measured concentrations of the training set for all samples. If the model were to predict only this mean value, it would be equivalent to making random guesses for all predictions, indicating that the model has not learned anything from the input data.

4
Results

This section aims to provide an overview of the project results. The first part presents the final settings for the model architecture. The next section focuses on the model's performance results, presented in the order of acute toxicity, carcinogenicity, and reproductive toxicity. As two models were used, one with a single deep neural network and one with all deep neural networks connected to a transformer, a comparison of their performance is included in this section.

4.1 Architecture

To decide the project's model structures, hyperparameter sweeps were conducted, which tested combinations of different parameter values of interest. Here, initially, acute toxicity data were used to perform the hyperparameter sweeps in a single-DNN model setting. Later, the process was repeated using carcinogenicity data, yielding the same results as the first case. Time constraints prevented additional hyperparameter sweeps for the expanded (multiple-DNN) model or reproductive toxicity data, so the same parameter settings found in the initial sweeps were used for all models in the project. Finally, or more information on the methods used, please refer to the Methods section.

The table listed as Table 4.1 shows the parameters and values that were tested in the hyperparameter sweeps. Here, the number of hidden layers in the model was determined by testing for one, two, and three layers, and the case that resulted in the best outcome (three hidden layers) was chosen. Some parameters were tested using ranges of values, which was possible due to the Bayesian approach used in the sweeps (see Theory for more details). This allowed for the testing of combinations of parameters until a stopping point was reached (around 400 combinations tested for each sweep).

Table 4.1: Parameter Sweep Configuration Summary

Parameter	Tested Values	Chosen Values
Learning Rate	$1*10^{-2}$ - $2*10^{-6}$	$1*10^{-5}$
Dropout	0.2, 0.3	0.3
Hidden Layers Size 1	100-900	300
Hidden Layers Size 2	100-900	500
Hidden Layers Size 3	100-900	700

Instead of using hyperparameter sweeps, the batch size and epoch number parameters were determined based on previous knowledge and trial and error. Here, the epoch number was increased up to 80 since the loss continued to decrease until that point, while the batch size was set to 256, a commonly used value that provided good performance. Freezing some encoder layers of the transformer was also tested but didn't improve performance. To prevent early overfitting and hasten convergence, a warm-up period of 100 steps, selected through trial and error, was added through a linear scheduler (see Methods). Finally, the final parameter configurations used for all models in the project are available in Table 4.2.

Table 4.2: Final Parameter Settings

Parameter	Values
Learning Rate	$1*10^{-5}$
Batch Size	256
Number of Epochs	80
Dropout	0.3
Hidden Layers Size 1	300
Hidden Layers Size 2	500
Hidden Layers Size 3	700

4.2 Model Performance: 10-Fold Cross-Validation

Presented in this section are the results of the 10-fold cross-validations used to evaluate the model's performance in both single- and multiple-DNN forms. As mentioned in Methods, the cross-validations were repeated 10 times, and each split was associated with a unique random seed to reduce uncertainty. Subsequently, the tables and figures presented show the fold average over the 10 runs for each fold, and the values shown represent the final losses achieved at the end of a run after 80 epochs of training. The upcoming subsections provide a detailed analysis of the model's performance on and results for each *in vivo* data set separately, starting with acute toxicity, followed by carcinogenicity and reproductive toxicity. Moreover, the analysis will also compare the outputs of both single- and multiple-DNN models for each case. Here, the acute toxicity data set will be subjected to a more comprehensive analysis, while for the other two data sets, a detailed analysis will not be performed due to time and length constraints, assuming similar effects to those affecting acute toxicity data to affect their results (however, see Appendix 1 for some of the results related to the latter two data sets).

Table 4.3 lists the total fold average of the median, best average, and loss_{MEAN} (see Methods) for each of the data sets acute toxicity, carcinogenicity, and reproductive toxicity. Here, the results are shown separately for the single- and multiple-DNN models. It can be seen that the best average loss is consistently lower than the median loss, and the acute toxicity data set performs better overall than the other two data sets. Furthermore, the multiple-DNN model performs worse than the single-DNN model in all cases.

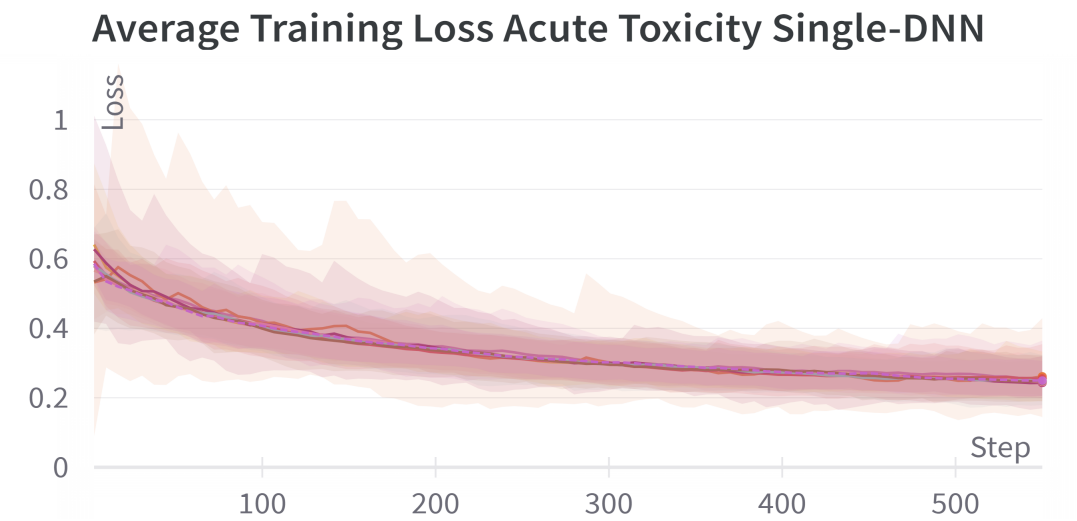
Table 4.3: Loss Comparison Between Analysed Data Sets in Single Model

Fold Average Measurement	Acute toxicity	Carcinogenicity	Reprotoxicity
Median Loss Single	0.273	0.543	0.688
Median Loss Multi	0.308	0.610	0.763
Best Average Loss Single	0.238	0.455	0.60
Best Average Loss Multi	0.260	0.513	0.627
Loss_{MEAN} Single	0.783	0.955	1.076
Loss_{MEAN} Multi	0.783	0.955	1.0882

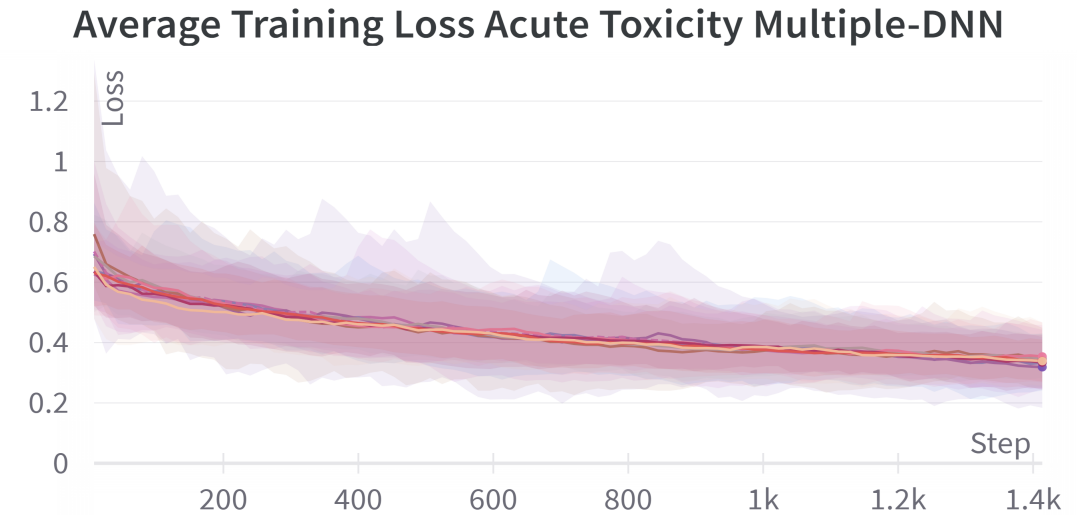
4.3 Acute Toxicity Data Set

4.3.1 Model Performance

Figure 4.1 shows the training loss curve (in Log_{10}[mg/kg bw] concentration) average for each of the 10 folds in the 10-fold cross-validation of the acute toxicity data set, plotted against the step count of the AdamW optimizer, for a) the single- and b) the multiple-DNN model. For both a) and b), a continuous decrease in loss is observed throughout each run, corresponding to the model learning. Overall, the two curves look similar, though the loss might be slightly higher for the multiple-DNN model. Potentially, in both cases, the loss is still decreasing at the end of each run, indicating that there is still more to learn in the training data at this point. The figure also indicates that there are some significant differences in output within the folds (seen by the large error margins), but that the averages are almost the same for all 10 cases. At the end of each fold's run, the lowest average training loss is achieved at about 0.3 in both a) and b).



a) Training loss average for each fold in 10-fold cross-validation for acute toxicity data in single-DNN model.



b) Training loss average for each fold in 10-fold cross-validation for acute toxicity data in multiple-DNN model.

Figure 4.1: Training loss average for a) single- and b) multiple-DNN models when analysing the acute toxicity data set.

In Figure 4.2, the median, in blue, and best average loss average, in yellow, for each fold (achieved at the end of each run) have been plotted together in a bar plot with the average loss_{MEAN}, in green, reference value for each of the 10-folds. Moreover, the results from both the single- and multiple-DNN model have been plotted together in such a way that the leftmost bar for each loss-value corresponds to the single-DNN model, whilst the rightmost bar corresponds to the multiple-DNN model. Here, it can be seen that the average values do not change much between folds, with the Loss_{MEAN} staying around 0.8 in both cases. Moreover, the median

and best average losses are much lower than the reference loss_{MEAN} value, being at around 0.3 and 0.25 respectively for both the single- and multiple-DNN case, indicating that the model is successfully finding patterns in the data. Notably, the best average loss values are lower than the median loss values for each fold.

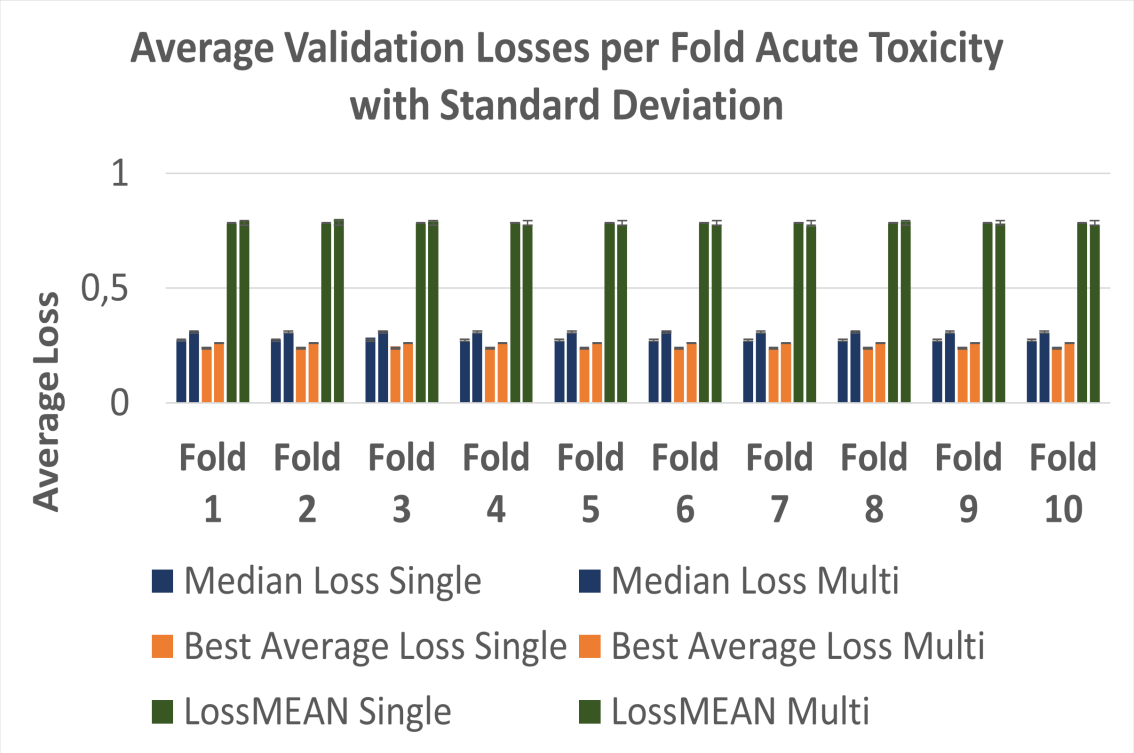
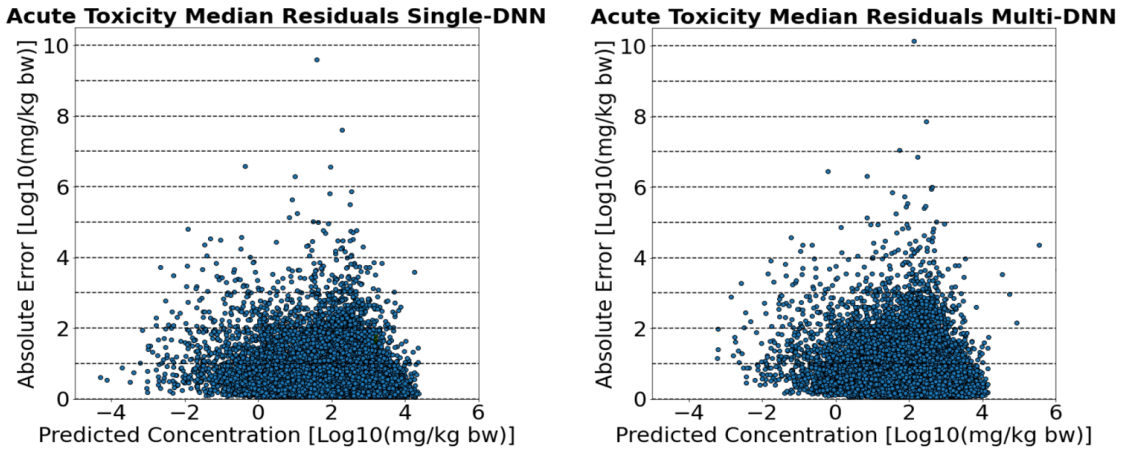


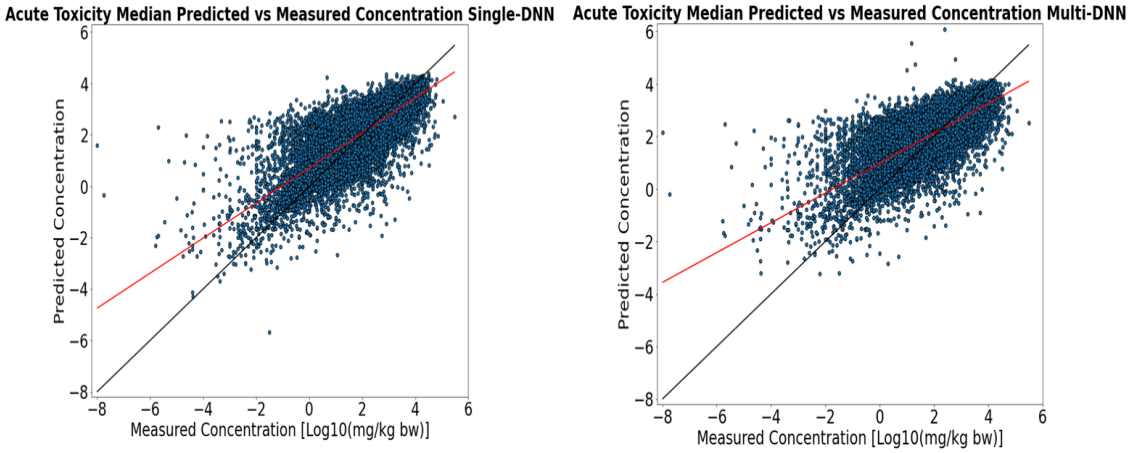
Figure 4.2: Validation best average loss, yellow, median loss, blue, and loss_{MEAN}, green, average over 10 folds for the acute toxicity data set, with values of single-DNN model to the left in each case, and multiple-DNN model to the right.

4.3.2 Model Results

Illustration a) in Figure 4.3 shows the residuals from the 10-fold (repeated 10 times) cross-validation of the acute toxicity data set, whilst illustration b) instead depicts the predicted concentrations plotted against the measured concentrations found in the data, all in Log_{10}[mg/kg bw]. In both cases, the values correspond to the median result for each SMILES in the data set, and in the second image, the red line corresponds to a linear fit to the data, whilst the black line is the perfect one-to-one fit. Moreover, the leftmost image in both cases corresponds to the single-DNN model, whilst the rightmost corresponds to the multiple-DNN model. For both the single- and multiple-DNN model, the residual plot, a), shows that most residuals lie relatively close to 0, except for a few notable chemicals with significantly larger residuals than the rest. Here, the result is strikingly similar between the two models, indicating that the models operate in much the same manner. For b), it can be seen that both models have a tendency to be too lenient on very low concentrations, predicting much higher concentrations for these cases than the measured data.



a) Median validation residuals in Log_{10}[mg/kg bw] for each SMILES.



b) Median validation predicted versus measured concentrations in Log_{10}[mg/kg bw] for each SMILES.

Figure 4.3: Median validation residuals and predictions versus measured concentrations for each SMILES in 10-fold cross-validation for acute toxicity data, with the left image corresponding to the single- and the right to the multiple-DNN model in both cases.

Table 4.4 contains the top five worst-performing chemicals in the acute toxicity data set, with their frequency of occurrence, mean concentration, predicted concentration, and absolute error in $\log_{10}$ scale. Here, the upper part of the table represents the results obtained from the single-DNN model, while the lower part represents those from the multiple-DNN model, with the same five chemicals appearing in both cases, albeit in a different order. The chemical names were identified by searching the corresponding SMILES on the PubChem database [31]. The table shows that the measured concentrations for these chemicals were much lower than the average, which may indicate that small doses of these chemicals were sufficient to produce the desired effect. However, the predicted concentrations for all these chemicals were much higher than the actual measured values, resulting in high absolute errors.

Table 4.4: Top Five Chemicals With Largest Median Residuals in Acute Toxicity Data Set

Single-DNN Model			
Name	Measured Conc.	Predicted Conc.	Absolute Error
rizatriptan	-8.00	1.59	9.59
JWH-015	-5.69	2.29	7.60
14-Methoxymetopon	-7.74	-0.36	6.59
SA4503	-4.64	1.97	6.57
Beta-Carotene	-5.30	0.997	6.30
Multiple-DNN Model			
Name	Measured Conc.	Predicted Conc.	Absolute Error
rizatriptan	-8.00	2.14	10.14
JWH-015	-5.69	2.47	7.85
Beta-Carotene	-5.30	1.75	7.05
SA4503	-4.64	2.23	6.85
14-Methoxymetopon	-7.74	-0.21	6.45

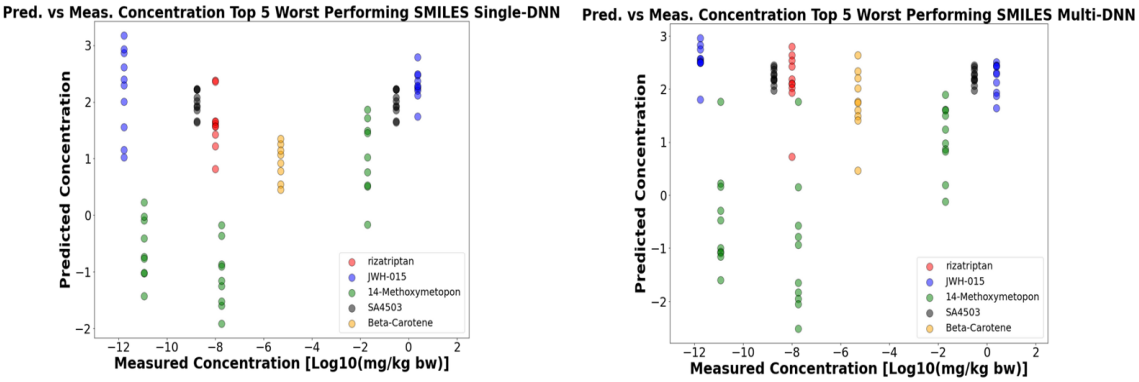
A corresponding table to the one above for the top five best-performing chemicals in the acute toxicity data set can be viewed in Table 4.5. In this table, notably, none of the chemicals except for carbon dioxide has common names and has therefore been addressed mostly by their CAS numbers. As in the previous table, the upper part of the table represents the results obtained from the single-DNN model, while the lower part represents those from the multiple-DNN model. Also notable, compared to the previous table, is that here all measured concentrations are much higher, and the same chemicals do not occur in the single- and multiple-DNN models.

Table 4.5: Top Five Chemicals With Smallest Median Residuals in Acute Toxicity Data Set

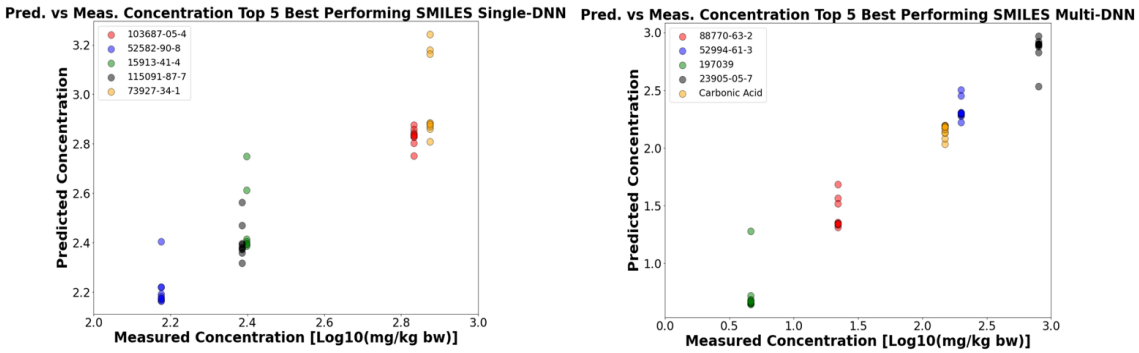
Single-DNN Model			
Name	Measured Conc.	Predicted Conc.	Absolute Error
103687-05-4	2.83	2.83	0.010
52582-90-8	2.18	2.18	0.011
15913-41-4	2.40	2.39	0.012
115091-87-7	2.39	2.38	0.014
73927-34-1	2.88	2.88	0.014
Multiple-DNN Model			
Name	Measured Conc.	Predicted Conc.	Absolute Error
88770-63-2	1.34	1.34	0.009
52994-61-3	2.30	2.30	0.012
197039	0.67	0.66	0.014
23905-05-7	2.90	2.89	0.016
Carbonic Acid	2.18	2.17	0.016

In Figure 4.4, all predictions in the entire 10-fold cross-validation (repeated 10 times) for a) the top five worst- and b) the top five best-performing chemicals have been

plotted against their measured concentrations in the data, as well as coloured according to which chemical the points belong, with the left plot in the figure corresponding to the single- and the right to the multiple-DNN model. For the worst-performing chemicals, in a), it is possible to see that there are large within-chemical-variation of the predictions, as well as between the different measured concentrations (which are all relatively low). Moreover, only 14-Methoxymetopon, SA4503 and JWH-015 have been measured more than once, with measured concentrations showing a large variation in value. On the contrary, the best-performing chemicals, in b), only occur once in the data set, and all have measured concentrations ranging between 0.5 to 3 Log_{10}[mg/kg bw].



a) Measured vs predicted Log_{10}[mg/kg bw] concentrations for top five worst performing chemicals.



b) Measured vs predicted $\log_{10}$(concentrations) for top five best performing chemicals.

Figure 4.4: Measured vs predicted Log_{10}[mg/kg bw] concentrations for a) the top five worst- and b) the top 5 best-performing chemicals, with the left plot corresponding to the single- and the right to the multiple-DNN model.

Finally, a Principle Component Analysis (PCA) was also performed on the CLS tokens (corresponding to unique SMILES) in the validation set data. Figure 4.5 showcases the first two components plotted against each other from a PCA done on one fold of the validation data, with the left plot in the figure corresponding to the single- and the right to the multiple-DNN model. In the case of both models, the first component (corresponding to the x-axis) has captured around 23% of the variation

in the data, whilst the second component (corresponding to the y-axis) has captured around 15% for the single- and around 13% of the variation in the multiple-DNN model. The PCA plots have been coloured by the median measured concentration for each SMILES. For both models, the measured concentration seems to vary along a gradient which largely corresponds to the separation captured by mostly the first but also partly the second PCA component. Here, it can be observed that SMILES associated with higher measured concentrations (indicated by yellow) lie further to the left in the plots, whilst SMILES associated with lower concentrations, in light blue/purple, instead lie further up and to the right of the plots.

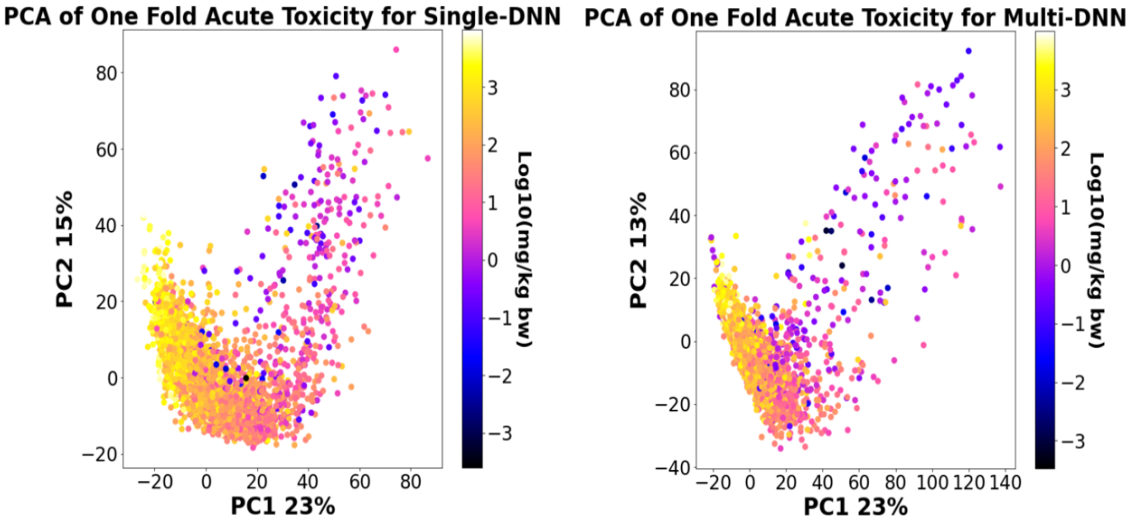
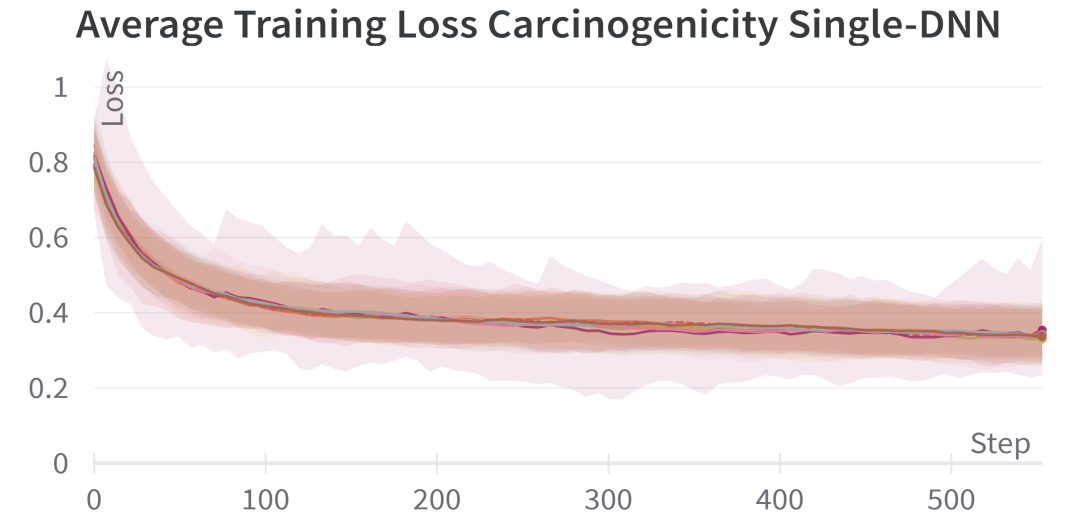


Figure 4.5: Principle component analysis of CLS tokens from one fold in 10-fold cross-validation of acute toxicity data set, with the left plot corresponding to the single- and the right to the multiple-DNN model.

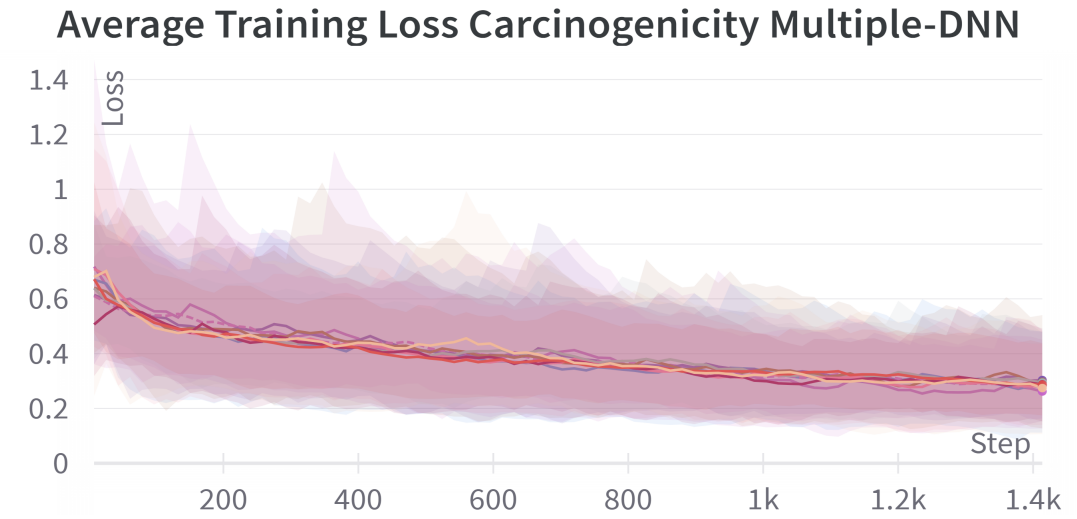
4.4 Carcinogenicity Data Set

4.4.1 Model Performance

In Figure 4.6, similarly to that of Figure 4.1 above, the average training loss (in $\log_{10}$ concentration) for each fold in the carcinogenicity data set has been plotted against the steps taken by the optimiser, with a) corresponding to the single- and b) to the multiple-DNN model. As in the case of acute toxicity, it seems that the fold average losses lie close to each other for both models, whilst the large shadowed areas indicate that there are some significant differences in output within the folds themselves. However, in this case, the second model seems to show more variation between folds, with some average fold losses even increasing before decreasing with the rest. Also, as observed for the acute toxicity case, it seems that the average losses, which reach an overall minimum of around 0.3-0.4 for both models, are still decreasing slightly at the end of the runs, indicating that there is still more to learn in the data.



a) Training loss average for each fold in 10-fold cross-validation for carcinogenicity data in single-DNN model.



b) Training loss average for each fold in 10-fold cross-validation for acute toxicity data in multiple-DNN model.

Figure 4.6: Training loss average for a) single- and b) multiple-DNN models when analysing the carcinogenicity data set.

Like in Figure 4.2 for acute toxicity, Figure 4.7 below depicts a bar plot of the average median loss, in blue, and best average loss, in yellow, plotted together with the average reference loss_{MEAN}, in green, and their respective standard deviations, for each fold in the carcinogenicity data. Here, the leftmost bar for each value corresponds to the single- and the rightmost to the multiple-DNN model. Compared to the acute toxicity data set, there here seems to be slightly more variation in output between the different folds for both models. As in the case of acute toxicity, however, the best average loss for each fold is in all cases lower than the median loss.

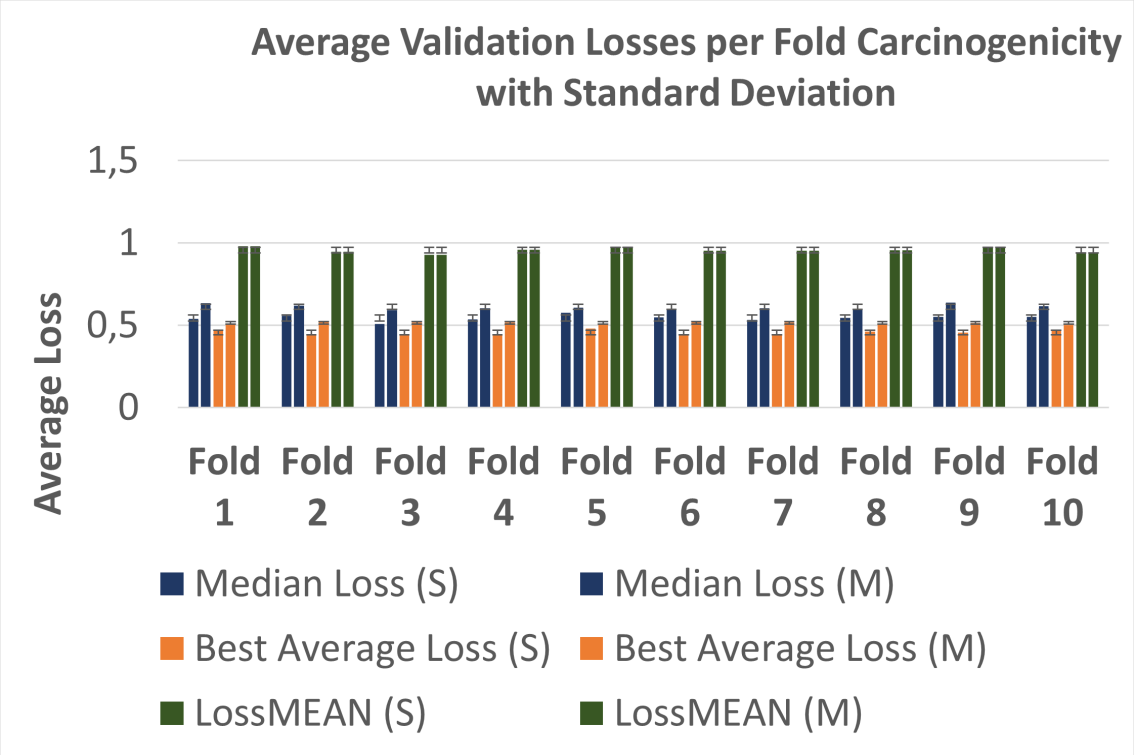
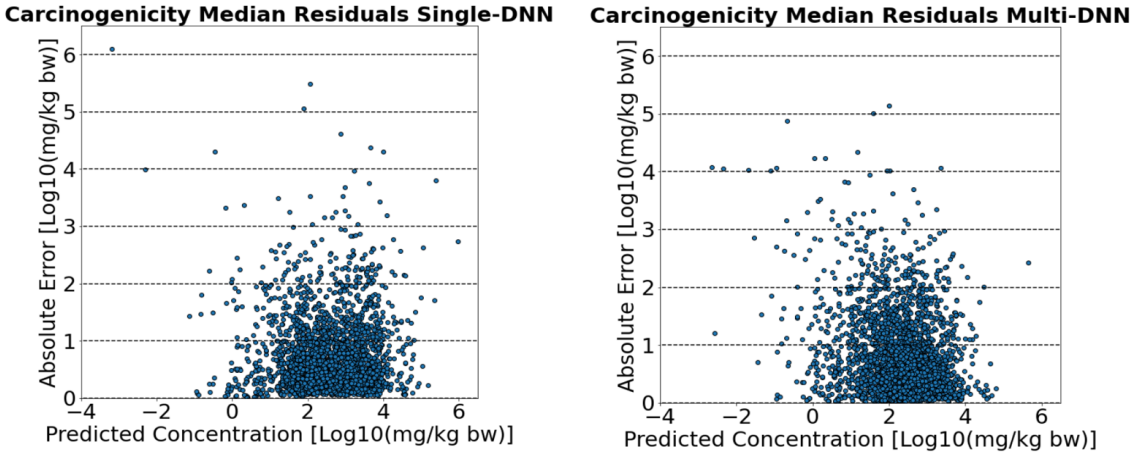


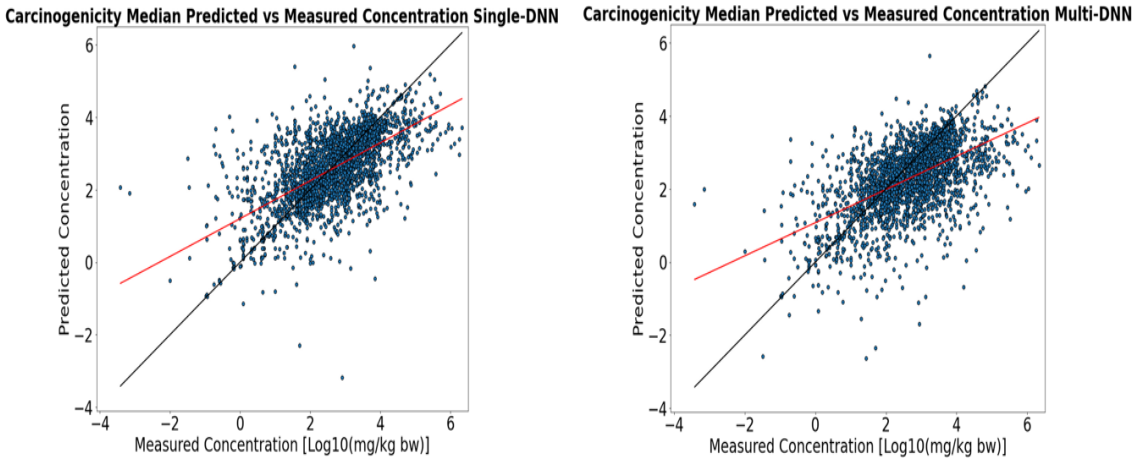
Figure 4.7: Validation best average loss, yellow, median loss, blue, and loss_{MEAN}, green, average with standard deviation over 10 folds for the carcinogenicity data set, with values of single-DNN model to the left in each case, and multiple-DNN model to the right.

4.4.2 Model Results

Figure 4.8 shows a) the residuals and b) the predictions plotted against measured concentration, in both cases taken as the median for each SMILES and in $\log_{10}$ concentration, of the single-DNN model, to the left, and the multiple, to the right. As in the case of the acute toxicity data set, it can be seen in a) that most residuals lie relatively close to 0, with some chemicals performing much worse than the rest. On the other hand, unlike the acute toxicity case, b) indicates that the carcinogenicity case has a less obvious tendency to predict high concentrations for chemicals with the lowest measured concentrations. However, there are still cases where the model has predicted much lower or higher concentrations than the measured values. In both a) and b), it can be seen that the single- and multiple-DNN models seem to perform very similarly.



a) Median validation residuals in Log_{10}[mg/kg bw] for each SMILES.



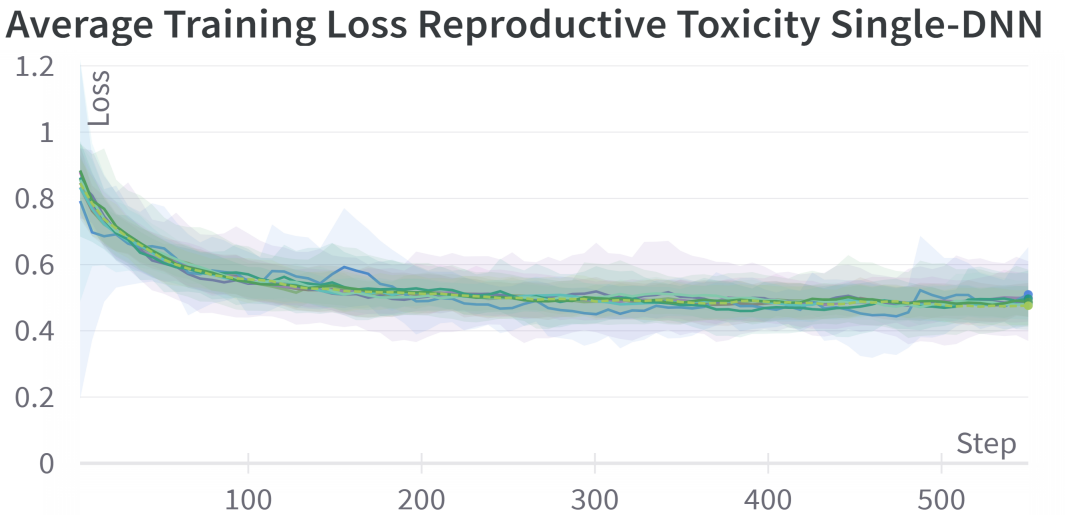
b) Median validation predicted versus measured concentrations in Log_{10}[mg/kg bw] for each SMILES.

Figure 4.8: Median validation residuals and predictions versus measured concentrations for each SMILES in 10-fold cross-validation for carcinogenicity data, with the left image corresponding to the single- and the right to the multiple-DNN model in both cases.

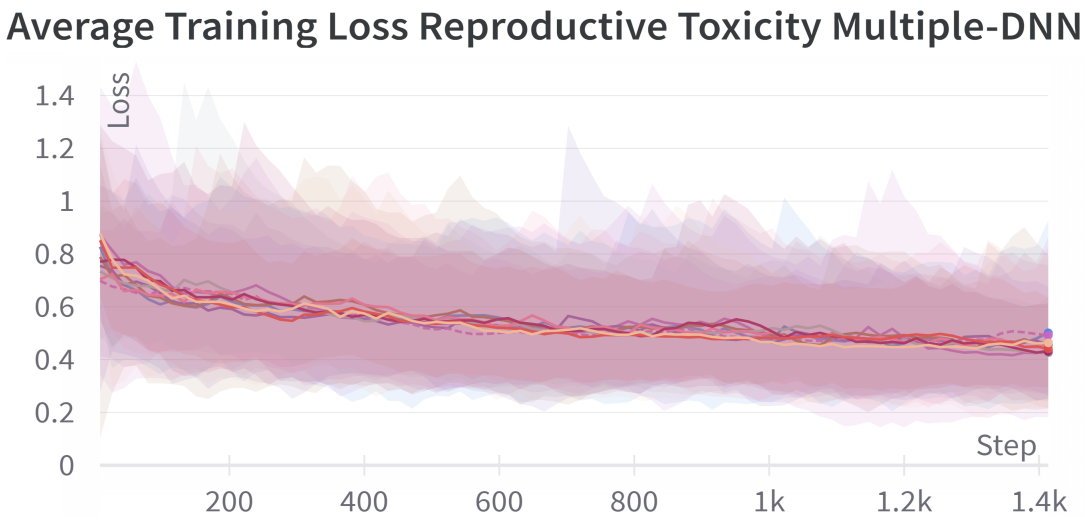
4.5 Reprotoxicity Data Set

4.5.1 Model Performance

The reproductive toxicity data set was analysed similarly to the previous two data sets, with the average training loss for the data set plotted against the AdamW optimiser step count in Figure 4.9 for a) the single- and b) the multiple-DNN model. Here, as for the other two data sets, the average fold training loss for all folds was similar, but with some large within-fold variations in the data, as indicated by the shaded areas. Unlike for the other two data sets, here the training loss did not show a significant decrease at the end of the runs, indicating that the training might have been more complete in this case. In the figure, it can be seen that the final average training losses for this data set were around 0.2-0.3 for a) and around 0.4 for b).



a) Average training loss over each fold in the single-DNN model.



b) Average training loss over each fold in the multiple-DNN model.

Figure 4.9: Average training loss over each fold in 10-fold cross-validation for *in vivo* reproductive toxicity data in a) the single- and b) the multiple-DNN model.

Exactly as was included for the previous two data sets, Figure 4.10 depicts the fold average median loss, in blue, and best average loss, in yellow, plotted together with the average reference loss_{MEAN}, in green, and respective standard deviation for each fold in the carcinogenicity data. As for the previous data sets, the leftmost bar for each value corresponds to the single- and the rightmost to the multiple-DNN model. Moreover, as noted for the carcinogenicity data above, even though the fold averages have similar values, there is more variation between the fold averages than in the acute toxicity data set. Also, as pointed out in the acute toxicity and carcinogenicity cases, the best average (average) value for each fold is much lower than the median loss.

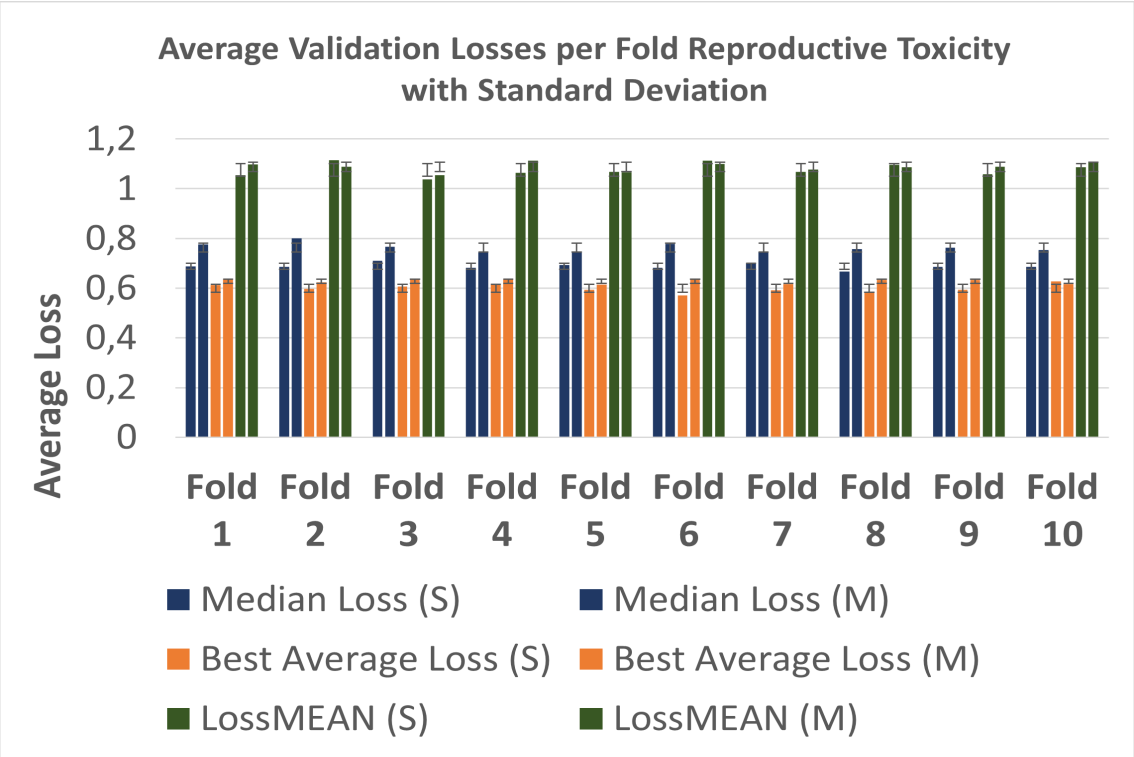
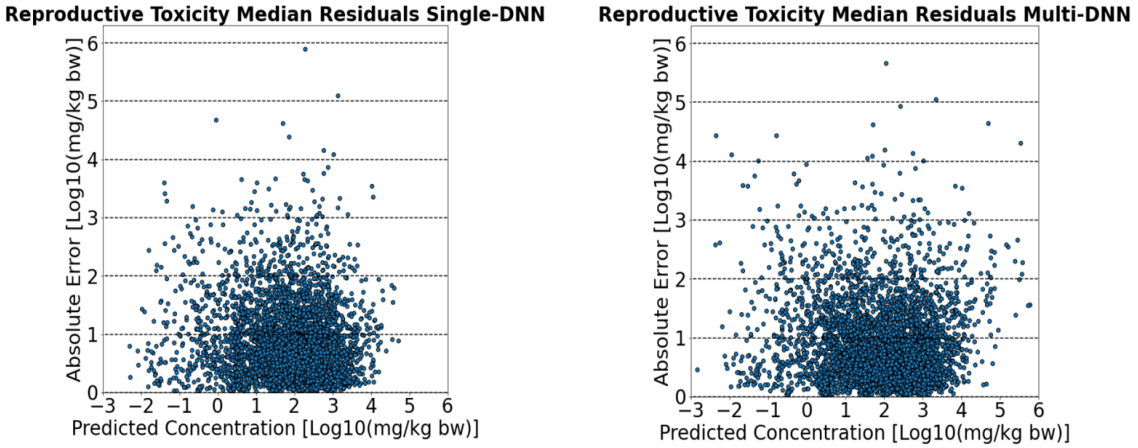


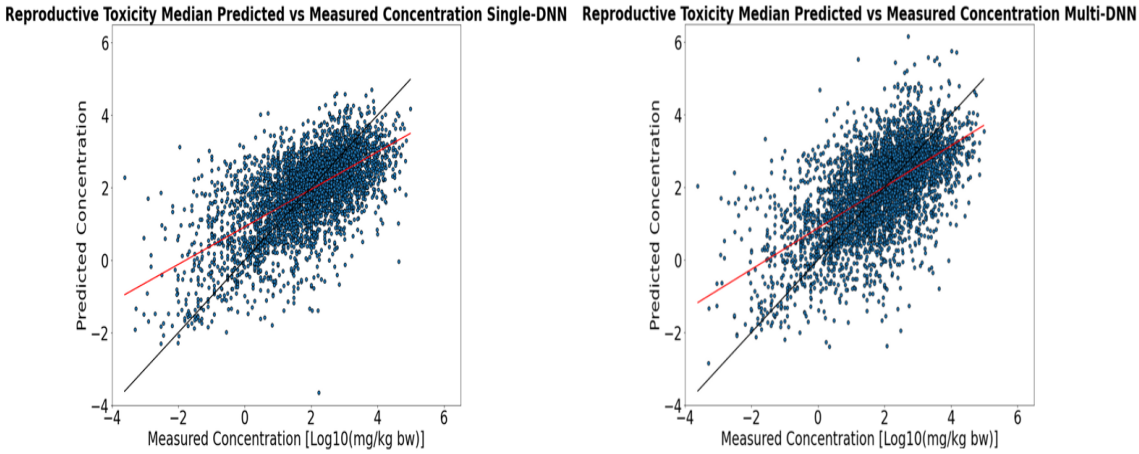
Figure 4.10: Validation best average loss, yellow, median loss, blue, and loss_{MEAN}, green, average with standard deviation over 10 folds for the reproductive toxicity data set, with values of single-DNN model to the left in each case, and multiple-DNN model to the right.

4.5.2 Model Results

For the reproductive toxicity data set, Figure 4.11 gives an overview of model performance, with a) the residuals and b) the predictions against measured concentrations, calculated as medians for each SMILES and in $\log_{10}$ concentration, for the single-DNN model, to the left, and the multiple, to the right. As for the previous two data sets, a) shows that a handful of chemicals in the data set have much higher residuals than the rest, with the results for the single- and multiple-DNN models looking very similar. Moreover, as in the case in the acute toxicity model (and to some degree in the carcinogenicity case), b) indicates that both of the models have a tendency to be too lenient on chemicals associated with very low measured doses, as indicated by the large number of chemicals lying far to the left in the plots. In b), it can also be seen that the multiple-DNN model seems to predict slightly higher values than the single.



a) Median validation residuals in Log_{10}[mg/kg bw] for each SMILES.



b) Median validation predicted versus measured concentrations in Log_{10}[mg/kg bw] for each SMILES.

Figure 4.11: Median validation residuals and predictions versus measured concentrations for each SMILES in 10-fold cross-validation for reproductive toxicity data, with the left image corresponding to the single- and the right to the multiple-DNN model in both cases.

5

Discussion

As hoped for, the single- and multiple-DNN models developed in this project demonstrate the ability to predict different toxic effects by utilizing the structural information of chemicals. Here, the first principal components of the transformer output indicate that the transformer is successful in identifying patterns in the chemicals' structures that correspond to their toxicity. However, both models tend to be too lenient with data associated with low concentrations, as seen in residual and prediction analysis. Moreover, the single-DNN model consistently performs better than the multiple-DNN model, possibly due to the added levels of complexity of the latter's task. To address all of these issues, one suggestion could be to make the loss more stringent for negative concentrations, as well as further hyperparameter sweeps to optimise the multiple-DNN model.

5.1 Model Performance

The training loss curves (Figures 4.1, 4.6, and 4.9) provide an initial indication of model performance by showing whether the model can learn from the data. Fortunately, the training loss decreases for all models and data sets analysed in the project, although some of the plots suggest that the model is still learning at the end of the runs. Here, it might be possible that longer training, say for 100 epochs instead of 80, may improve performance, but this depends on whether the training data accurately represents the test data. Notably, the average losses in each curve and fold are similar for all cases, indicating that 10-fold cross-validation successfully minimises fold bias, although there are large within-fold variations in the data, as shown by large error margins. Moreover, moving on to the validation data, Table 4.3 indicates that the average performance is best for the acute toxicity data set, as observed in Figures 4.2, 4.7, and 4.10. The reason behind this is definitely at least in part due to the fact that the acute toxicity data set accounts for around 90% of all *in vivo* data in the study and has the majority of unique chemicals, as illustrated in Figure 3.1 and 3.2 in the Methods section. Here, the fact that there is much more data in the acute toxicity data set compared to the other two translates to there being a lot more information for the model to train on (as the training set is made up of 90% of the data), meaning that the model will have many more opportunities to get familiar with different structures.

However, the size of the data sets cannot solely explain the differences in performance observed among the models, as while the reproductive toxicity data set is more than twice as large as the carcinogenicity data set, it still performs worse. The

final hyperparameter settings were determined after a set of sweeps, see Table 4.1 and 4.2, only performed on the acute toxicity and carcinogenicity data set, meaning that the reproductive toxicity data set is not necessarily represented by these settings. Potentially, this could at least partially explain the low performance of this data set. On the other hand, when the hyperparameter sweep was performed on the carcinogenicity data set, parameter settings were observed to have only a marginal effect on model performance. In lieu of this, there is no clear reason why hyperparameter settings would play a large role in model performance for the reproductive toxicity data set, even though this possibility cannot be dismissed. Instead, one possible reason why the reproductive toxicity data set performs worse than the carcinogenicity data set, despite being larger, could be that the former is inherently more difficult to predict due to the complexity of the data. For example, it can be challenging to define what constitutes a LOEC value for reproductive toxicity, whereas, for acute toxicity and carcinogenicity, the answer is clearer. Overall, effects on reproduction and offspring can have many explanations, making it difficult to define what reproductive toxicity really constitutes. This difficulty might, for example, result in noisier data, leading to difficulties in prediction. Hence, to improve the model for this type of data, it might be a good idea to separate the data into different groups and be more selective in the pre-processing step.

In analysing Table 4.3 and Figures 4.2, 4.7, and 4.10, it becomes evident that the multiple-DNN model consistently performs worse than the single-DNN models. Here, the main difference between the models is the number of DNNs connected to the transformer used for interpreting the SMILES. While, for the single-DNN cases, the transformer will prioritise parts of SMILES/chemical structures which will improve performance for the specific data set the model was built for, the idea behind the multiple-DNN model was to use one transformer to find patterns in the data that improve SMILES translation for all networks. However, by having three separate DNNs connected to the same transformer, it is possible that the transformer became worse at translating SMILES in a way that is beneficial for each separate data set due to being trained by conflicting messages from all three DNNs simultaneously. For example, a chemical which is cancerous might not be at all acutely toxic. As seen by the Venn diagram in Figure 3.2, only very few chemicals in the data overlap with each other in each data set. Notably, the diagram does not indicate how large the overlap is in structural elements of chemicals, which is potentially what might have a larger effect on the transformer performance. Here, possibly, the transformer embedding size used for the larger model could be increased to try and increase the transformer's ability to handle and adjust to different the different problems at hand.

Additionally, one reason why the multiple-DNN model performs worse than the single-DNN model could be the data set sizes and the sampling method used. In the implementation process of the multiple-DNN model, the difference in data set sizes between the concatenated acute toxicity, carcinogenicity, and reproductive toxicity data sets became an issue. As the acute toxicity data set is much larger than the other two, drawing random rows for training batches would exclude data from the smaller sets and make training for any network except the acute toxicity network

impossible, thereby breaking the model. To circumvent this problem, the smaller data sets were upsampled to ensure that a certain percentage of the training set consisted of the other two data sets in each training batch, see Methods for more details. During training, fold-dependence was observed in the outputs, see Figures 4.2, 4.7 and 4.10, particularly for the smaller data sets. As this fold-dependence was not that noticeable for the acute toxicity data set, but more pronounced for the smaller other data sets, the implication becomes that the latters' smaller sized training and validation sets could be a drawback for the performance of the multiple-DNN model (and make the output unstable for these cases). As the amount of upsampling required for these data sets seems to be a critical factor that affects the model's performance, the best way to determine the optimal upsampling percentage might be through a hyperparameter sweep.

Overall, the absence of a hyperparameter sweep on the multiple-DNN model could have a significant impact on its performance. Since the task for the multiple-DNN model is distinctly different from that of the single-DNN model, the latter's architecture may not be representative of the former's. However, the logistics of designing a hyperparameter sweep for the multiple-DNN model present a significant challenge as it includes the need to determine the architecture of three DNNs simultaneously. Here, the extensive parameter settings and combinations to be tested would make this a resource-intensive and time-consuming process. Consequently, future hyperparameter sweeps for this model must be planned carefully to ensure that no unnecessary steps are taken. As a last note, Table 4.3 and Figures 4.2, 4.7, and 4.10 reveal that the best average loss is consistently lower than the median loss average for each fold. This outcome is not unexpected, as the best average loss accounts for within-variation in SMILES (that is, the same SMILES occurring several times in the validation data), making it a more accurate reflection of the model's performance.

5.2 Result Analysis

Moving on to discussing the model outputs, that is residuals and predictions, see Figures 4.3, 4.8, and 4.11, it is first noted that the outputs from the single- and multiple-DNN models are very similar for each data set, indicating that the models work similarly. Furthermore, it seems that the models have a tendency to predict too high concentrations for chemicals with very low measured concentrations. The reason for this tendency is unclear, but it could be the result of an imbalance between the training and test data, where, for example, the training data contains very few chemicals with low concentrations. It is not unthinkable that chemicals associated with low measured concentrations represent more hazardous compounds, meaning that this leniency in the models could become an issue if they ever were to be used in practice. Hence, some effort should probably be taken to mitigate this leniency. Here, for example, one way of increasing the "importance" of low-concentration chemicals would be to make the loss harsher for these compounds. Possibly, increasing or weighing up the chemicals with low measured concentrations in the training set could also be considered. However, as a final note on this topic, it

can be pointed out that some of the chemicals with very low concentrations, outliers in the plots mention above, have very low measured concentrations relative to most other compounds in the data sets. Hence, it is not unlikely, as will be discussed again later, that these measurements are a result of errors in the data, reflecting the large uncertainty and reliability issues found when dealing with large sets of data.

A more detailed analysis was performed on the results of the acute toxicity data set. For both the single- and multiple-DNN model, the top five worst-performing chemicals were identified and listed in Table 4.4, where it was found that all of these chemicals had very low median measured concentrations. Interestingly, Beta-Carotene, which is known to occur in carrots and to be highly non-toxic, was one of the chemicals. Figure 4.4, which in a) shows all the predicted and measured concentrations of the worst-performing chemicals for both models, reveal that only three chemicals, 14-Methoxymetopon, SA4503 and JWH-015, are associated with several different measured concentrations. Here, it is evident that there is a large variation between the measured concentrations associated with these chemicals, which could potentially lead to difficulty in their prediction, as well as indicate some error in their measurements. As there is a lack of replicates for the other chemicals, the model is highly sensitive to the accuracy of the measurement for these chemicals. Hence, once again leading back to the data reliability issue, it is possible that an error in these measurements could be the root cause for the large residuals associated with these chemicals.

Finally, the principal component analysis of the first two components of the CLS tokens in the acute toxicity data, see Figure 4.5, shows some separation which seems to correspond to the measured concentrations of the data. This would indicate that the transformer itself is able to make the distinction between chemicals necessary to determine if a high or low concentration is associated with it. Notably, when comparing the single- and multiple-DNN model PCA plots with each other, the former seems to capture the variation slightly better, possibly as an effect of the former performing slightly better than the latter model.

6
Conclusion and Future Work

In this study, the aim was to develop and assess transformer-based AI models for predicting toxicity in mammalian *in vivo* assay data. To do this, two model types, a single-DNN model and a multiple-DNN model, were designed and evaluated. Here, the validation set used to evaluate the models contained SMILES not found in the training set, enabling the models' capacity to handle previously unseen data to be measured to some extent. However, the focus of the study was primarily on assessing the models' ability to predict different types of chemical hazards, both separately in the single-DNN model, and later in unison in the multiple-DNN model.

The results showed that both models achieved median losses ranging from 0.3 to 0.6 in logarithmic scale, translating to a median error of 2-4 in linear scale. Additionally, PCA visualisation showed that both models successfully identified patterns in SMILES structures related to measured concentrations of corresponding chemicals. However, throughout all experiments, it also became evident that the multiple-DNN model performed slightly worse than the single-DNN model. This was unexpected, as it was expected that the transformer would be able to accommodate several DNNs through fine-tuning. One explanation for this finding could be that, due to time limitations and logistical challenges, the multiple-DNN model's architecture never was evaluated using hyperparameter sweeps. For potential future implementations of the model, performing a hyperparameter sweep to determine its architecture could therefore be a priority. Moreover, factors such as the degree of upsampling needed for smaller data sets and the embedding size of the transformer could also be important factors to investigate more closely in the future. Both models tended to overestimate concentrations of compounds associated with lower measured concentrations, indicating a potential problem if the models were ever to be used in practice, as these probably correspond to more dangerous chemicals. Hence, in future adjustments of the models, mitigating this problem, by for example setting a harsher loss on these low-concentration chemicals, should be a priority. Finally, additional improvements for future versions of the model could be to incorporate additional data sets together with the inclusion of *in vitro* toxicological assay data to reduce dependence on *in vivo* animal test data.

To conclude this report, one can ask if this project has fulfilled the promise implied by its title. Here, undoubtedly, models have been introduced that possess significant potential to serve as alternatives to *in vivo* animal testing in the future. However, it is important to acknowledge that further research and development are necessary before these methods can attain widespread adoption and effectively compete with conventional testing approaches on a larger scale. Nonetheless, the numerous

challenges associated with *in vivo* animal testing, as highlighted in the report's introduction, make its eventual replacement inevitable. Furthermore, in light of the world's increasing digitalisation and the compelling advantages in terms of cost and time efficiency offered by these methods, it is not a matter of if, but rather when computer-based approaches will become the new standard. In this context, it is evident that artificial intelligence (AI), with its capacity to learn and process the vast amounts of data generated by contemporary society, will undoubtedly constitute the cornerstone of these computer-based methods. Moreover, the successful utilisation of transformer-based AI models in predicting chemical toxicity, as demonstrated by this project, establishes a crucial foundation for future testing methodologies. Given the intelligence, affordability, and reliability of this technology, it unquestionably represents an important initial step towards shaping the testing methods of tomorrow.

Bibliography

[1] Yang J, Zhang Y. NCRF++: An Open-source Neural Sequence Labeling Toolkit. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics; 2018. Available from: http://aclweb.org/anthology/P18-4013.

[2] European Environment Agency (EEA). *Chemicals for a sustainable future Copenhagen, 17 May 2017 Report of the EEA Scientific Committee Seminar.*; May 2017. URL: https://www.eea.europa.eu/about-us/governance/scientific-committee/reports/chemicals-for-a-sustainable-future.

[3] Mayr A, Klambauer G, Unterthiner T, Hochreiter S. *DeepTox: Toxicity Prediction using Deep Learning.* Front Environ Sci. 2016;80. Available from: 10.3389/fenvs.2015.00080.

[4] European Chemicals Agency (ECHA). *The use of alternatives to testing on animals for the REACH regulation*; 2021. https://op.europa.eu/en/publication-detail/-/publication/c53fbd08-7fbc-11eb-9ac9-01aa75ed71a1/language-en#.

[5] European Chemicals Agency (ECHA). *REACH Information requirements*; 2022. [Online; accessed 09-October-2022]. https://echa.europa.eu/regulations/reach/registration/information-requirements.

[6] Scholz, S and Sela, E and Blaha, L and Braunbeck, T and Galay-Burgos, M and García-Franco, M et al . *A European perspective on alternatives to animal testing for environmental hazard identification and risk assessment*; 2022. https://doi.org/10.1016/j.yrtph.2013.10.003.

[7] Cherkasov, A and Muratov, E N and Fourches, D and Varnek, A and Baskin, I I and Cronin, M et al . *QSAR Modeling: Where Have You Been? Where Are You Going To?*; 2014. https://doi.org/10.1021/jm4004285.

[8] David L, Thakkar A, Mercado R, Engkvist O. *Molecular representations in AI-driven drug discovery: a review and practical guide.* J Cheminform. 2020;39.

[9] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez Aea. *Attention is all you need.* Advances in neural information processing systems. 2017. Available from: https://www.sciencedirect.com/science/article/pii/S0169743997000610.

[10] Laws, EA. *Environmental Toxicology.[electronic resource]: Selected Entries from the Encyclopedia of Sustainability Science and Technology.* Springer New York. 2013:1-15. Available from: https://search.ebscohost.com/login.aspx?direct=true&db=cat07472a&AN=clec.SPRINGERLINK9781461457640&site=eds-live&scope=site.

[11] Fisher, MR. *Environmental Biology: 6.3v Environmental Toxicology*. Open Oregon Educational Resources. Available from: https://search.ebscohost.com/login.aspx?direct=true&db=cat07472a&AN=clec.SPRINGERLINK9781461457640&site=eds-live&scope=site.

[12] Agency SC. *Hazard and risk assessment of chemicals - an introduction*; 2020. https://www.kemi.se/download/18.32f4eb311753c0a67fe1cf6/1604653630900/Guidance-Hazard-and-risk-assessment-an-introduction.pdf.

[13] European Chemicals Agency (ECHA). *Guidance on registration*; 2021. [Online; accessed 11-April-2023]. https://echa.europa.eu/documents/10162/2324906/registration_en.pdf/de54853d-e19e-4528-9b34-8680944372f2?t=1629205524601.

[14] Jain AK, Mao J, Mohiuddin KM. *Artificial neural networks: A tutorial*. Computer; 1996.

[15] Schmidhuber J. Deep learning in neural networks: An overview. Neural Networks. 2015;61:85-117. Available from: https://www.sciencedirect.com/science/article/pii/S0893608014002135.

[16] Svozil D, Kcasnicka V, Pospichal J. *Introduction to multi-layer feedforward neural networks*. Chemometrics and Intelligent Laboratory Systems. 1997;39. Available from: https://www.sciencedirect.com/science/article/pii/S0169743997000610.

[17] Käll S. Predicting Chemical Ecotoxicity using Artificial Intelligence. 2022.

[18] He C. *Transformer in CV*; December 2021. https://towardsdatascience.com/transformer-in-cv-bbdb58bf335e.

[19] *Sequence Modeling with Neural Networks (Part 2): Attention Models*; 2016. https://indicodata.ai/blog/sequence-modeling-neural-networks-part2-attention-models/.

[20] Alammar J. *The Illustrated Transformer*; 2020. http://jalammar.github.io/illustrated-transformer/.

[21] Devil J, Chang W M, Lee KT. *Google, and A. I Language. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.*. Tech. rep.;. [Online; accessed 16-April-2023]. https://github.com/tensorflow/tensor2tensor.

[22] Devlin J, Chang M, Lee K, Toutanova K. *BERT: Pre-training of Deep Bidirectional Tranformers for Language Understanding*. Tech rep. 2019. Available from: https://doi.org/10.48550/arXiv.1810.04805.

[23] Muller B. *BERT 101 State of The Art NLP Model Explained*; 2022. [Online; accessed 16-April-2023]. https://huggingface.co/blog/bert-101.

[24] Liu Y, Ott M, Goyal N, Joshi M, Chen D, Levy Oea. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. Advances in neural information processing systems. July 2019. Available from: https://doi.org/10.48550/arXiv.1907.11692.

[25] Chithrananda S, Grand G, Ramsundar B. *ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. arXiv preprint arXiv:20100909885*. 2020.

[26] Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi Aea. *Transformers: State-of-the-art natural language processing. Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations.* 2020:38-45.

[27] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan Gea. *Pytorch: An imperative style, high-performance deep learning library.*. Advances in neural information processing systems 32; 2019.

[28] Paul, S. *Bayesian Hyperparameter Optimization - A Primer*; 2020. [Online; accessed 11-April-2023]. https://wandb.ai/site/articles/bayesian-hyperparameter-optimization-a-primer.

[29] *Sequence Modeling with Neural Networks (Part 2): Attention Models*;. https://paperswithcode.com/method/linear-warmup#:~:text=Linear%20Warmup%20is%20a%20learning,the%20early%20stages%20of%20training.

[30] Gaoxia, J and Wenjian, W. *Error estimation based on variance analysis of k-fold cross-validation.* Elsevier LTD. 2017. Available from: http://dx.doi.org/10.1016/j.patcog.2017.03.025.

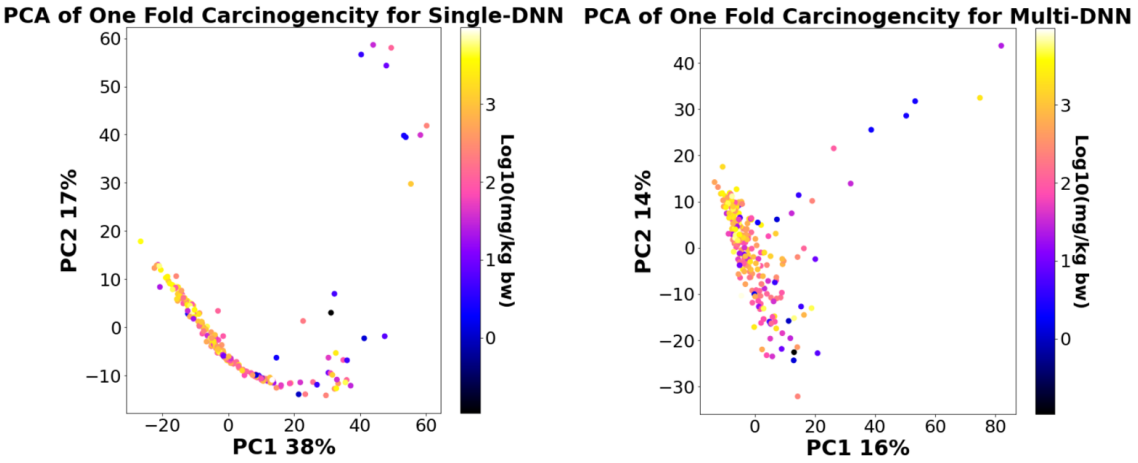
[31] Muller B. *BERT 101 State of The Art NLP Model Explained*; 2022. [Online; accessed 16-April-2023]. https://pubchem.ncbi.nlm.nih.gov/.

A

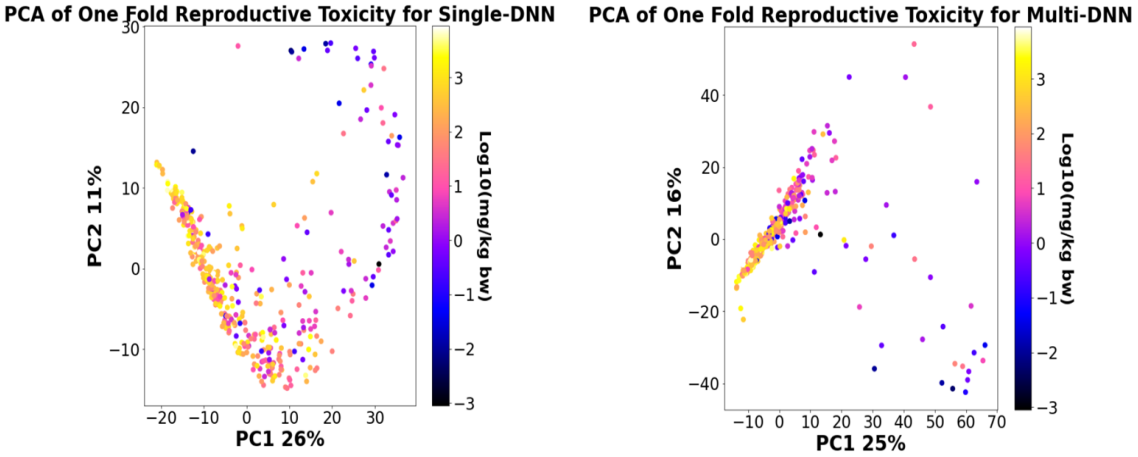
Appendix 1

A.1 PCA for Carcinogenicity and Reproductive Toxicity Data Sets

In this appendix, the principal component analysis plots for the CLS tokens from one fold of the carcinogenicity and the reproductive toxicity data sets can be found, see Figure A.1. In the figure, the individuals have been coloured according to the corresponding median measured concentration for the SMILES/CLS token, with lighter yellow corresponding to higher measured concentrations and blue/black to lower. Moreover, for both the carcinogencity and the reproductive toxicity data set, the leftmost plot corresponds to the PCA of the single-DNN model, and the rightmost for the PCA of the multiple-DNN model.



a) PCA of the CLS tokens from one fold in the carcinogenicity data set.



b) PCA of the CLS tokens from one fold in the reproductive toxicity data set.

Figure A.1: Principle component analysis of CLS tokens from one fold in 10-fold cross-validation of the a) the carcinogenicity data set, and b) the reproductive toxicity set, coloured by corresponding median concentration for each CLS, with the left plot corresponding to the single- and the right to the multiple-DNN model in each case.

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY