Domain adaptable language modeling of chemical compounds identifies potent pathoblockers for *Pseudomonas aeruginosa*

G. Kallergis[1,2], E. Asgari[1], B. Azarkhalili[3], M. Empting[4,6,7], A. Hirsch[5,6,7] and A. C. McHardy[1,2,6*]

[1*]Computational Biology of Infection Research, Helmholtz Centre for Infection Research, Braunschweig, Germany.
[2]Braunschweig Integrated Centre of Systems Biology (BRICS), Technische Universitat Braunschweig, Street, Braunschweig, Germany.
[3]AI VIVO, Bio-Innovation Centre, Cambridge, England.
[4]Antiviral & Antivirulence Drugs (AVID), Helmholtz-Institute for Pharmaceutical Research Saarland (HIPS) - Helmholtz Centre for Infection Research (HZI), Saarbrücken, Germany.
[5]Department of Drug Design and Optimization (DDOP), Helmholtz-Institute for Pharmaceutical Research Saarland (HIPS) - Helmholtz Centre for Infection Research (HZI), Saarbrücken, Germany.
[6]Deutsches Zentrum für Infektionsforschung (DZIF), Hannover - Braunschweig, Germany.
[7]Department of Pharmacy, Saarland University, Campus E8.1, 66123 Saarbrücken, Germany.

*Corresponding author(s). E-mail(s): alice.mchardy@helmholtz-hzi.de;
Contributing authors: georgios.kallergis@helmholtz-hzi.de; ehsaneddin.asgari@helmholtz-hzi.de; azarkhalili@aivio.co; martin.empting@helmholtz-hips.de; anna.hirsch@helmholtz-hips.de;

Abstract

Computational techniques for predicting molecular properties are emerging as pivotal components for streamlining drug development, optimizing time, and financial investments. Here, we introduce *ChemLM*, a transformer language model-based approach for this task. *ChemLM* further leverages self-supervised domain adaptation on chemical molecules to enhance its predictive performance across new domains of interest. Within the framework of *ChemLM*, chemical compounds are conceptualized as sentences composed of distinct chemical 'words', which are employed for training a specialized chemical language model. On the standard benchmark datasets, *ChemLM* has either matched or surpassed the performance of current state-of-the-art methods. Furthermore, we evaluated the effectiveness of *ChemLM* in identifying highly potent pathoblockers targeting *Pseudomonas aeruginosa* (PA), a pathogen that has shown an increased prevalence of multidrug-resistant strains and has been identified as a critical priority for the development of new medications. *ChemLM* demonstrated significantly higher accuracy in identifying highly potent pathoblockers against PA when compared to state-of-the-art approaches. An intrinsic evaluation demonstrated the consistency of the chemical language model's representation concerning chemical properties. Our results from benchmarking, experimental data, and intrinsic analysis of the *ChemLM* space confirm the wide applicability of *ChemLM* for enhancing molecular property prediction within the chemical domain.

Keywords: computational chemistry, deep learning, molecular property prediction, language processing of chemicals, chemical domain adaptation

1 Introduction

Approximately 12 years [1], and 1.8$ billion are typically required before a drug reaches the market [2], and there is an overall failure rate of 96% for candidate compounds [3]. The discovery and development of novel anti-infectives, especially against bacterial pathogens are challenging and prone to setbacks [4]. Despite unmet medical needs, and the steadily increasing threat of antimicrobial resistance (AMR), the lack of new antibiotics with novel, resistance-breaking modes of action has resulted in an 'innovation gap', potentially leading to a 'post-antibiotic era' [5]. In this scenario, the available treatment options for bacterial infections become ineffective, primarily due to the spread of multi- and pan-resistant strains. This is already evident with pathogens like *Pseudomonas aeruginosa*, frequently found with multiple drug resistances in clinical settings [6]. Consequently, the World Health Organization (WHO) has identified the need for new antibiotics targeting this bacterium as a critical priority.

To prevent unnecessary failures and help refill the development pipeline, improvements in the drug discovery and development process through the

implementation of cutting-edge methodologies and technologies are therefore paramount. *In silico* approaches, such as machine learning, and in particular, recent advancements in deep learning and deep language modeling, have shown the potential to accurately capture the structural properties of molecules and more accurately identify drug candidates [7, 8], facilitating drug development. However, currently, the value of these techniques has mainly been assessed on large benchmark datasets, including thousands of compounds, and it is unclear whether they can effectively detect drug candidate compounds from smaller experimental datasets generated within a drug discovery process.

In machine learning-based chemistry, the predictive models can be trained on chemical descriptors such as fingerprints representing the chemical characteristics of compounds [9, 10]. Their drawbacks, like sparsity, can be circumvented by representing chemical compounds either as natural graphs or as string representations that encode all the necessary chemical information. Such graphs are used as input to Graph Neural Networks (GNNs) [11–17]. Treating molecules as graphs maintains molecular topology, among other advantages. However, certain aspects in sequence representations, like chirality, cannot be conveyed using these approaches.

In a broad definition, languages consist of sequences generated from a finite set of elements [18]. From this perspective, many phenomena in the world can be regarded as languages. The analogy with language motivates the use of the distributional hypothesis in linguistics, which states: "a word is characterized by the company it keeps" [19]. Aligning with this theory, recent computational approaches have been developed. These approaches suggest that words sharing similar contextual usage demonstrate vector proximity in a high-dimensional space when trained on a large corpus [20, 21]. This is a useful property that makes language processing methodologies arise as potential solutions in various domains with extensive unlabeled data, e.g., in protein sequences [22–25], in DNA sequences [26] or even chemicals [27]. The prevailing sequence representation of compounds is SMILES, which stands for Simplified Molecular-Input Line-Entry System [28], a depth-first preorder spanning tree traversal of the molecular graph. Similar to proteins, SMILES meets the language definition, and their molecule representations can be processed with language models [29], such as Word2Vec [20, 30, 31], and Recurrent Neural Networks (RNNs) [29, 32]. Transformers models are a recent development [33] taking advantage of large amounts of sequence representations of chemical structures [34, 35]. Transformers employ transfer learning, where, briefly, a model is trained on a related or more general problem with abundant training data, to then be adapted or used for a target task with limited data available, resulting in improved performance, and accelerated convergence. Although transfer learning was initially developed for supervised machine learning tasks, its application has been expanded to self-supervised tasks [36–38], enabling model pre-training on large datasets with millions of records.

Here, we describe *ChemLM*, a language modeling-based approach for efficient transfer learning for chemical compounds. *ChemLM* utilizes the SMILES representation of molecules as sentences of the input language, and a three-stage training process for predicting a specific molecular property of chemical compounds. This includes pre-training of a self-supervised language model on large datasets, self-supervised training on further domain-specific data, and subsequent model optimization in a supervised setting. With this, we aimed for an approach that can be applied for real-world datasets of experimental compounds that comprise of limited training samples/compounds. We assessed whether language models' training using domain adaptation, which allows us to adapt the pre-trained model on further data from the target domain, enhances the model's predictive ability. We performed extensive performance comparisons to the state-of-the-art models. We furthermore investigated whether the model successfully captures the underlying chemical information, and reproduces the chemical space. Moreover, we predicted the potency of candidate pathoblocker compounds against *Pseudomonas aeruginosa* from an experimental dataset encompassing just 219 compounds, demonstrating the value of *ChemLM* for this application in the drug discovery process.

2 Results

The *ChemLM* method

ChemLM is a transformer-based method that processes molecules' SMILES as sentences representing the chemical structures. *ChemLM* has three training stages (Fig. 1a), consisting of (i) a self-supervised pre-training stage, (ii) a secondary domain-specific pre-training, and then, (iii) a fine-tuning stage for the supervised classification in molecular property prediction tasks. Initially, a language model is trained using transformers on a large corpus of chemical compounds, to learn the chemical language by unveiling the general relationships among the tokens, a step called pre-training. Then, the model is further trained in a self-supervised manner on domain-specific compounds. Optionally, the training instances are extended with a data augmentation algorithm to cover multiple views on the chemicals. In the last step the model is fine-tuned by supervised training on the domain-specific compounds for a given task. In all these stages, a workflow processes SMILES compound representations into a sequence of chemical 'words' that are then used as input for the *ChemLM* transformer models (Fig. 1b).

(i) Language-model pre-training: Pre-training is a part of transfer learning, where the model is trained on millions of samples before it gets fine-tuned on the specific task at hand. Masked language modeling (MLM) masks random tokens of the input sequence, and trains the model by predicting the masked token based on the surrounding ones. The model was initially trained on the large corpus of the ZINC database (10 million compounds) using MLM as introduced in BERT [36]. At this stage, we used unlabeled data consisting

of tokenized SMILES to learn the representations of the compounds. This created the *ChemLM* base model encoding the syntax, and semantics of the language of chemical compounds.

(ii) Domain adaptation for the language modeling: before fine-tuning the *ChemLM* model in the chemical task, we introduce one more level of training on domain data. In this stage, the pre-trained model is further trained on domain-specific, unlabeled data, which improves the ultimate performance, as shown in [39, 40]. The goal is to fine-tune the language model to better capture the data structure specific to the final task. One main issue here is that there may only be little domain-specific data available to train the model, leading us to perform data augmentation on the task-specific compound dataset, which can be done using SMILES enumeration [41]. This technique performs atom reordering in the SMILES strings, resulting in multiple representations of a molecule, and is a fast, and computationally cheap way to augment the existing dataset by several factors. Data augmentation was used for the whole dataset. Since the model is trained in an unsupervised way using MLM, there is no leak of information to the model in the evaluation phase.

(iii) Supervised fine-tuning of the transformer language model network: In the final phase, the trained model undergoes supervised fine-tuning. To prevent overfitting, we deploy early stopping in addition to techniques in model development, e.g., L2 regularization. Instead of freezing the transformer's layers, and fine-tuning only the classification head, we choose to unfreeze all of them, and further fine-tune them to optimize performance. The attention maps, spread across various layers of a transformer model trained on chemical compounds, can be utilized to demonstrate how different chemical tokens interact in creating the final language model-based embedding of these compounds (Supplementary Fig. A1).

Architecture optimization

While hyperparameters play a significant role in influencing the effectiveness of deep learning models, their exploration within this domain has not been thoroughly investigated so far. Here, we assess the impact of key hyperparameters of the transformers architecture, and of our approach. We conducted a search using the Optuna framework, and we analyzed the importance of parameters including the augmentation number, the number of hidden layers, and attention heads, as well as the type of embeddings for the transformer model. The hyperparameters, and the range of their values for the optimizations can be found in Supplementary data (Supplementary Table A1). Furthermore, we evaluated each hyperparameter's impact on the final outcome through Optuna's f-ANOVA test (Fig. 2).

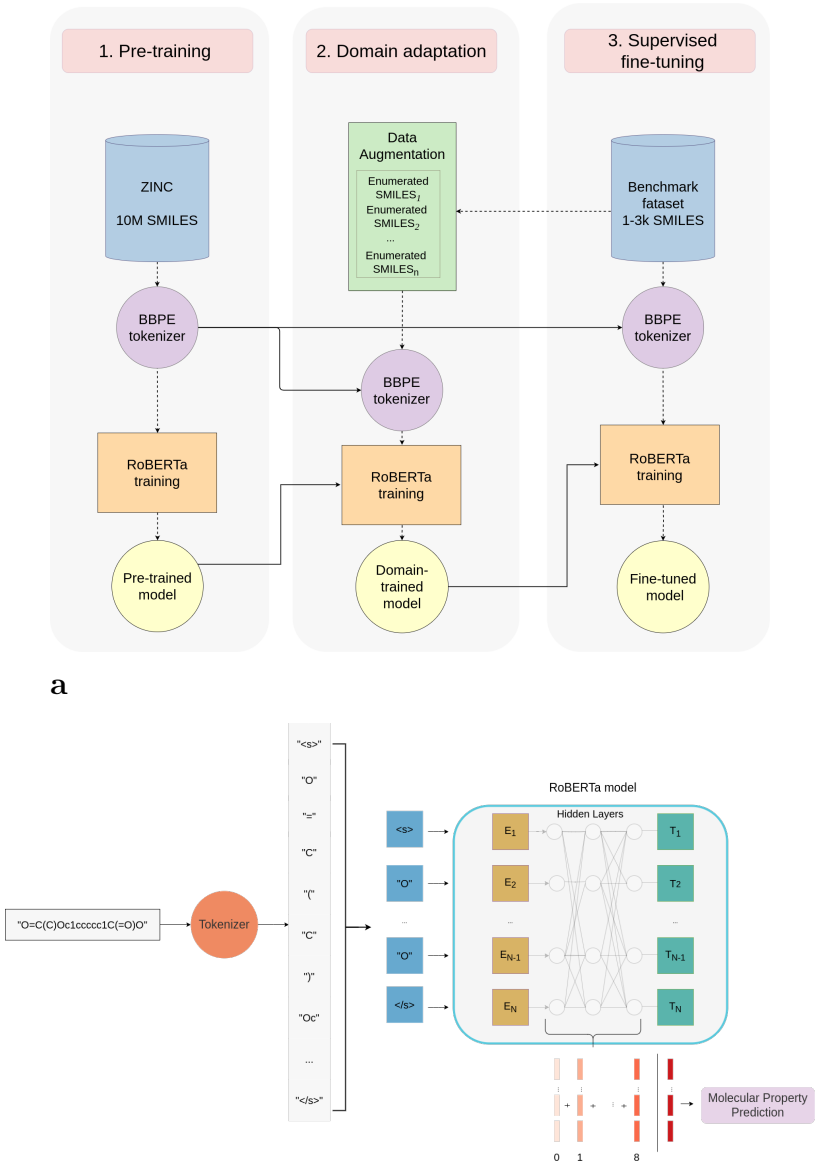


Fig. 1 The *ChemLM* approach. a) Training stages of the *ChemLM* model. All the trained models are represented by circular shapes, BBPE models are in purple, and RoBERTa is in yellow. Procedures like training, augmentation, and prediction are indicated with rectangles. The dashed line indicates the flow of information within a training stage, whereas the solid line describes the transfer of knowledge from one training stage to another. b) An example that indicates how a SMILES string is processed, and treated by the *ChemLM* transformer model. Firstly, it gets tokenized, and special tokens are added to the sequence. Then, these are fed into the model, and at the end, the sum of weights from the hidden layers is used to make predictions.

As a key part of the *ChemLM* method, we investigated the optimal augmentation number for domain adaptation training, i.e. the number of alternative molecule representations in SMILES. To examine this, we introduced a wide range of randomized SMILES representations during training, between 0, and 100. The augmentation number substantially affected the model (Fig. 2a), and high values (80 or 100 augmentations) were consistently selected in the optimization process. Data augmentation increased model training time, which rose linearly to the number of molecule augmentations provided (Supplementary Table A2).

Inspired by the authors of BERT [36], we also explored the optimal embeddings by combining weights from different layers in various ways, such as summation, and averaging in the last layer or across multiple layers. Notably, we examined whether using the weights of the first token of the sequence or a combination of all tokens yielded better results. The choice of focusing on the first token was grounded in the understanding that it encapsulates a description of the entire sentence, and receives the most attention from all the heads [36, 42]. The type of embeddings, substantially influenced performance (Fig. 2a). Contrasting this, the number of attention heads and the number of layers had the least impact on performance. The selected hyperparameter values during optimization are reported for each task (Supplementary Table A3).

ChemLM identifies potent pathoblockers for *P. aeruginosa*.

In drug discovery, oftentimes, a very limited number of compounds are available, substantially fewer than those included on commonly used benchmark datasets for chemical property prediction tasks. To assess the value of *ChemLM* for a real-world drug discovery problem, we employed it to identify potent pathoblockers compounds acting against *P. aeruginosa* (Fig. 3a), which is one of the priority pathogens identified by the World Health Organisation, often characterized by multidrug resistance [6]. The class of compounds that we focused on disrupts the quorum-sensing (QS) machinery of *P. aeruginosa* [43–49], using a compound library of 219 structures with varying potency. The drug target is the QS receptor, and transcription factor PqsR [50].

Small molecular compounds acting on PqsR via an inverse agonistic mode-of-action reduce the production of several virulence factors such as the toxin pyocyanin. The initial hit already impaired pyocyanin production with a potency in the double-digit micromolar range, and was characterized by a trifluoromethyl-pyridine fragment.[46] A lead generation campaign via structure-guided fragment growing was initiated, which yielded five QS inhibitor classes with substantially increased potency [43–45] (Fig. 2a), and retaining this fragment motif. We use the IC_{50} to measure drug potency, which is the inhibitor concentration needed to inhibit a biological process *in vitro* by 50%. Highly potent compounds have an IC_{50} of less than 500 nM. For the

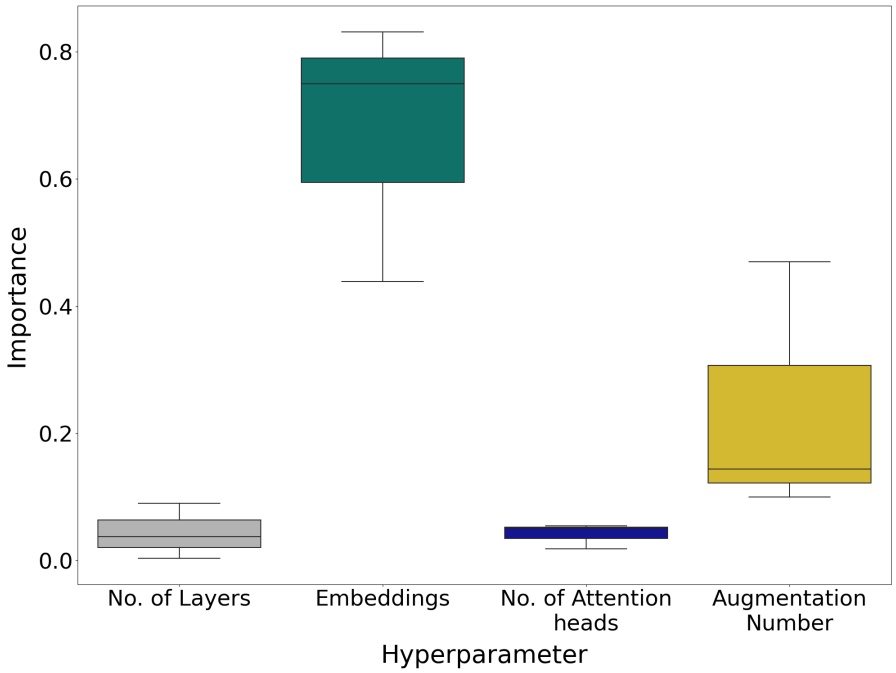
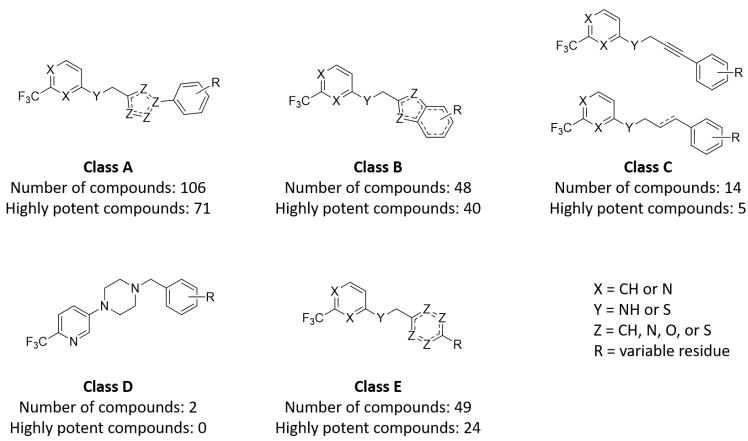


Fig. 2 Importance of hyperparameters in model's performance during hyperparameter optimization using the validation data of each dataset. The examined hyperparameters are: the embeddings type, the number of attention heads, and hidden layers, and the augmentation number.

five classes, the number of compounds, and their potencies vary considerably; from 2 to 107, and including between 0 and 71 highly potent compounds.

To rigorously evaluate the performance of *ChemLM*, we devised a challenging scenario. Given the substantial variation in the number of compounds per class in the compound library, we pursued an alternative approach to partition the data into more similarly-sized folds. We employed ward linkage hierarchical clustering on the *ChemLM* embeddings, and partitioned the library into five sets of chemically similar compounds, resulting in a more even distribution (Supplementary Data Table A4). Specifically, we organized the compound library by grouping compounds into these folds based on *ChemLM*'s embeddings similarity. This approach ensures that compounds with chemical similarity, even if they belong to different structural classes, are kept together within the same fold as opposed to using the initial structural classes. This strategy helps prevent information leakage during model training, and introduces a demanding challenge for the *ChemLM* model. Subsequently, we conducted the third stage of model training using the SMILES representations of compounds from four of the folds. The compounds from the remaining fold were then classified as highly potent or not. This process was repeated for each set of folds (Fig. 3b) and the same hyperparameters were used for all models (Supplementary Table A3).

a



b

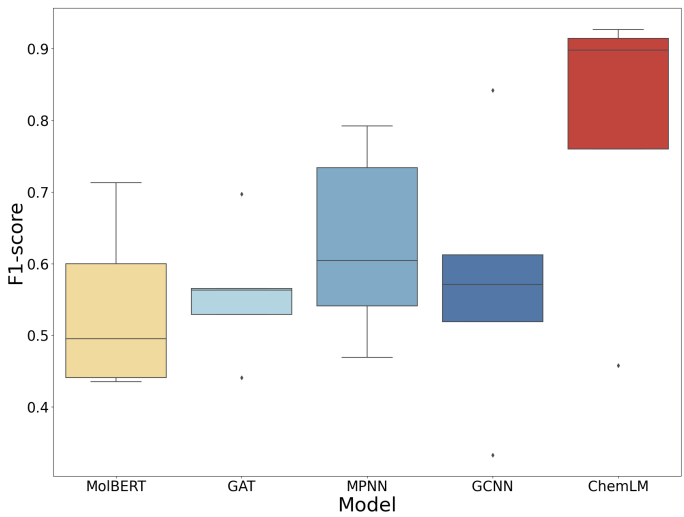


Fig. 3 Description of experimental data: (a) Chemical structures, and number of compounds per class. (b) Performance comparison of *ChemLM* with graph neural networks and MolBERT in 5-fold validation for experimental compounds on *Pseudomonas aeruginosa*.

We compared *ChemLM* to several state-of-the-art models on these data, including Graph Convolution Neural Networks (GCNN) [51], Graph Attention Transformers (GAT) [16] and Message Passing Neural Networks (MPNN) [14] using their implementations in DeepChem (version 2.6.0) with the default architecture (Fig. 3b). In addition, we compare our approach with MolBERT, a recent transformer-based approach [52]. *ChemLM* achieved the highest median of macro-averaged F1-scores (0.899), which is almost 30% more than that of the second-best model (MPNN; Fig. 3, Supplementary Data Tables A5). The same applies for all the evaluation metrics we examined. Moreover, its performance on identifying highly potent pathoblockers is quite high, as the F1-score for that class in each of the five folds consistently ranges from above 0.825 to a maximum of 0.92 in all folds (Supplementary Table A6). Most notably, *ChemLM* demonstrates consistency when compared to other models, which either fail or perform poorly on this task in certain folds. These results highlight the value of the optimized *ChemLM* for identifying highly potent compounds for an application with a very limited number of compounds available for a task-specific training scenario.

Optimizing *ChemLM* substantially improves performance

We assessed the performance of *ChemLM* by training models in different ways for binary classification tasks in molecular property prediction, and then again compared their performance to MolBERT, GCNN, GAT, and MPNN across the three benchmark datasets (Supplementary Table A7). The datasets were split in a stratified way using DeepChem's splitter [53]. We chose that way of splitting as it ensures that each class is represented in the training/validation/test sets, and reflects the actual class distribution in each set. All datasets were split proportionally into 70% training, 10% validation, and 20% test sets. Training parameters for grpah neural networks such as the epochs, and the learning rate were optimized using a grid search and deployed the DeepChem framework for that.

First, a *ChemLM* vanilla model was trained without using a domain adaptation phase or hyperparameter optimization. In its architecture, 12 layers, and attention heads were included, and pooling as the type of embeddings (Supplementary Table A3). A second model, *ChemLM* domain-adapted, was then trained on domain-specific data, with augmented SMILES representations, and no hyperparameter optimization took place, using the same architecture as *ChemLM* vanilla. Finally, for the *ChemLM* domain-adapted & optimized model, all the hyperparameters were optimized, and in addition, we unfroze the model's layers for fine-tuning in the task-specific training.

The optimized *ChemLM* was among the top performers in benchmark evaluation (Fig. 4). It performed substantially better than the graph-based models, with an improvement of up to 0.2 in F1-score on the ClinTox dataset relative to the second-best performing model (Supplementary Table A10). This makes

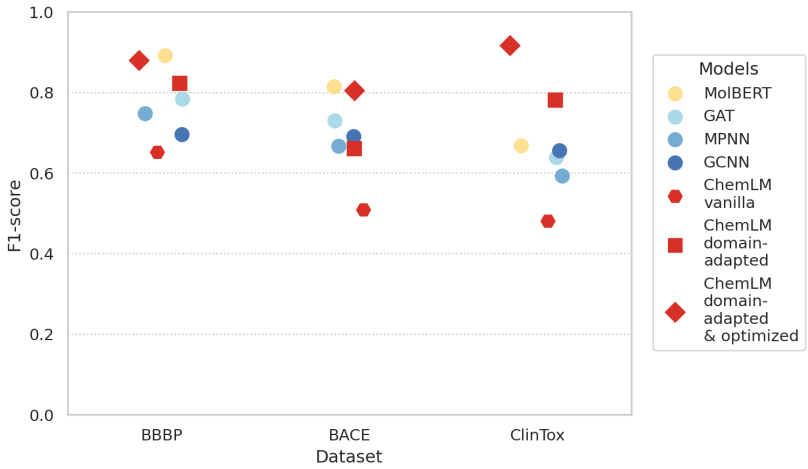


Fig. 4 Performance of *ChemLM* and state-of-the-art models with the macro averaged F1-score on the test data sets of the benchmark data. *ChemLM* and its variations are compared with state-of-the-art models. The graph neural networks (blue) are GAT (Graph Attention Transformers)[16], MPNN (Message Passing Neural Networks)[14], and GCNN (Graph Convolutional Neural Networks)[51]. MolBERT [54] (in yellow) is a transformer-based approach using the BERT model. *ChemLM* models are noted in red. *ChemLM* demostrates equal or better performance to the state-of-the-art models.

it a highly valuable innovation for computational chemistry. Compared to MolBERT, which also utilizes transformers, we observe a very similar performance on two of the datasets (BBBP and BACE); however, *ChemLM* substantially outperformed it on ClinTox by almost 25%. This performance gap on the ClinTox is caused by the poor results of these models on the positive class (Supplementary Table A11). We observed that even though they successfully perform this task on BACE and BBBP datasets, they do not do so on ClinTox dataset. Similarly to what we observed for the pathoblocker dataset, these models failed to identify the few positive samples on the dataset (Supplementary Table A11) as we also showed for the experimental pathoblockers (Supplementary Table A6).

Interestingly, we also observed a substantial improvement between the vanilla, and the domain-adapted models, which is a result of adding the domain adaptation stage and the data augmentation. That ranges from 15% for BACE dataset up to 30% for ClinTox. Most notably, the overall increase in performance from the vanilla version to the domain-adapted and optimized one, is up to 0.43 F1-score on the ClinTox dataset. That demonstrates the value of these steps to models' enhanced performance. Differences in performance between the different models were the least pronounced for the BACE dataset. The complete evaluation of the models for these datasets can be found in the Supplementary material (Supplementary Tables A8-A10).

ChemLM embeddings reflect molecular properties of chemical compounds

To assess whether the compound embeddings generated by *ChemLM* are reflective of the underlying molecular properties relevant for drug efficassy, we assessed the continuity of their representations in the embeddings using the Lipschitz constant (k), and compared it with a randomly created space, by randomly shuffling assigned molecular properties.

We applied this analysis to six relevant physicochemical properties: molecular weight, quantitative estimate of drug-likeness (QED), hydrogen-bond donors, and acceptors, polar surface area, and the number of aromatic rings. We used the chemical properties of compounds and their embeddings generated by *ChemLM* to calculate the k of 200 randomly selected chemical compounds for 100 rounds. Using the distributions of *ChemLM*'s and random's space (Supplementary Fig. A2), we calculated the median Lipschitz constant. The results of our analysis demonstrated that, for all properties, our space's median k exhibited significantly lower values compared to the random space (one-sided t-test, Table 1). This consistent behaviour suggests that *ChemLM* effectively maps molecules in an informative, and meaningful manner.

Table 1 Median Lipschitz constant values and its p-value for each molecular property.

Molecular Property	*ChemLM*	Random Space	Ratio	p -value
Molecular Weight	4.952	5.486	0.903	4.2e-34
QED	0.01	0.011	0.909	1.09e-59
Hydrogen donors	0.05	0.056	0.893	1.02e-40
Hydrogen acceptors	0.065	0.068	0.956	1.28e-07
Polar surface area	1.27	1.367	0.929	5.67e-13
Num. aromatic rings	0.034	0.035	0.971	2.66e-09

Low median values are observed for Lipschitz constant in most of the properties, and a relatively stable ratio of *ChemLM*, and the random space. P-values are calculated using one-tailed t-test.

To qualitatively assess our results, we visualized the embeddings of molecules in a two-dimensional space using UMAP. This approach allowed us to determine whether compounds are encoded in meaningful embeddings in the *ChemLM* model, aligning chemicals with similar physicochemical properties in close proximity, while maintaining the global structure of the data distribution. We applied this technique for several of the previously assessed molecular properties in our evaluation (Fig. 5). For all properties, we observed a gradual change of these properties in this space, indicating that molecules with similar properties tend to possess similar embedding values.

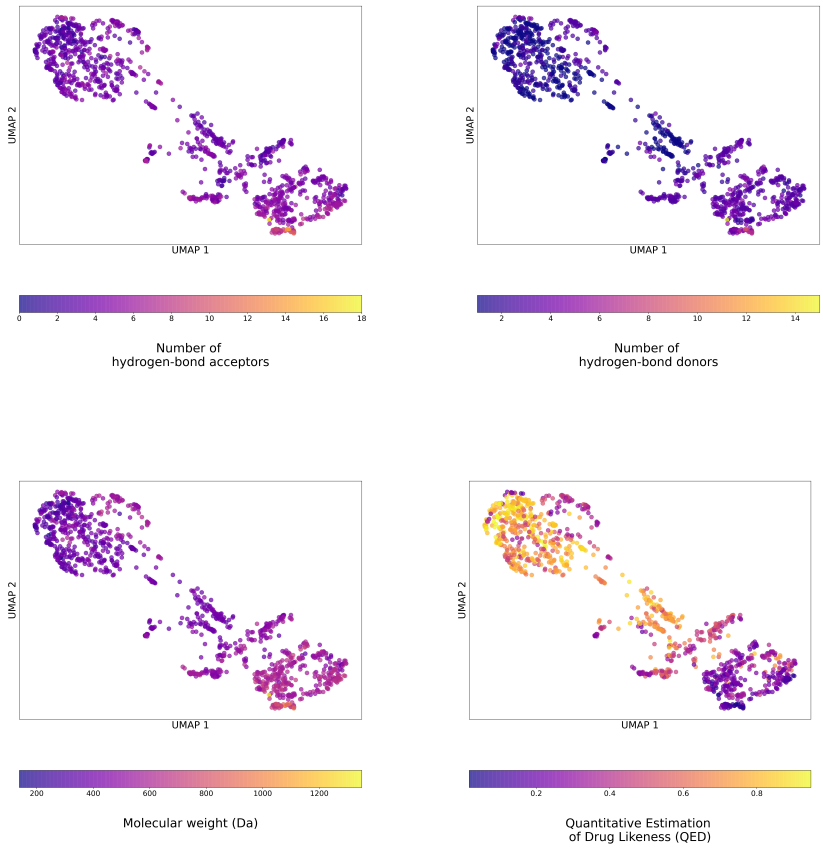


Fig. 5 UMAP plots of molecular properties. They demonstrate the distribution of molecular properties. Each dot represents a molecule in the BBBP dataset.

3 Discussion

In this study, we describe *ChemLM*, a language modeling-based approach for efficient transfer learning in the field of molecular property prediction for chemical compounds. *ChemLM* includes several methodological innovations for the chemical language modeling. The first novelty is the introduction of another training stage, in which the model is further trained on domain-specific compound representations in a self-supervised manner. That opposes to current methods that use only pre-training and fine-tuning on the prediction task [34, 35]. This domain adaptation training stage allows the model to learn the semantics of the chemical associations from task-specific data, and further improves the predictive power of the model for that task. Substantial improvement was noticed particularly on relevant tasks with little domain-specific data available. The second methodological novelty of the method lies in extending the domain-specific training data by data augmentation. Data augmentation is a technique for creating more representations of a sequence. It has been used to increase the number of instances in the pre-training or the fine-tuning stage, especially for chemical tasks with a few hundred samples. It is the first time that this technique has been used in the domain adaptation stage.

As a real-world test case, we evaluate our model on identifying compounds for *Pseudomonas aeruginosa*, a hospital-acquired pathogen that oftentimes exhibits multiple drug resistances. We observe substantial performance gains for the task of identifying potent pathoblocker compounds effective against *Pseudomonas aeruginosa* from a chemical compound library acting on the transcription factor PqsR. We partitioned a dataset of experimental compounds into training, and testing sets to assess the model's ability to identify structurally more distant candidate molecules. In this evaluation, the *ChemLM* model demonstrated a significant improvement, with a relative 30% enhancement over the second-ranking model on this task. The F1-score for the positive class (highly potent pathoblockers) was higher than 0.82 in all folds as well. This showcases the model's remarkable ability to generalize effectively and its consistency on the task compared to other assessed techniques. Thus, the performance gains provided by *ChemLM* can substantially facilitate the identification of relevant drug compounds for pharmacological applications. Further applications of the *ChemLM* framework extend from predicting active compounds to predicting activity levels, suggesting potential potent compound structures using generative models. We anticipate that it will find broader applications in experimental data analysis in the future.

We comprehensively assessed *ChemLM* on suitable benchmark datasets for molecular property prediction; the BACE (inhibition of the BACE-1 enzyme), BBBP (blood-brain barrier penetration) and the ClinTox dataset (clinical toxicity) deriving from MoleculeNet. On all of these, *ChemLM* demonstrated a substantial performance gain up to 20% relative to the graph neural networks. The results indicate that an optimized transformer-based approach

can outperform leading Graph Neural Network architectures. In addition, it performed similarly to MolBERT, another language processing approach; however, *ChemLM* substantially surpassed its performance on the ClinTox dataset, by 20% in F1-score. This further underscores the capability of *ChemLM* for excellent performance in discerning the positive class within imbalanced datasets when compared with state-of-the-art methods.

Moreover, we noticed substantial improvement in *ChemLM*'s performance due to our methodological improvements across all benchmark datasets, e.g., the addition of domain adaptation stage. Furthermore, via an extensive hyperparameter optimization, we demonstrated that certain parameters substantially impact the final performance. Among these parameters, embeddings proved to be the most influential, as indicated by our optimization results. Additionally, in the domain adaptation stage, we utilized multiple molecule representations. It's worth noting that a high number of these representations was selected leading to improved performance and proving its importance in this stage. Our optimization efforts provided valuable insights into the importance of hyperparameters in the model's architecture, ultimately enhancing its potential. That provides other researchers in the field with a useful guideline for hyperparameter tuning in future approaches.

Finally, as an intrinsic evaluation of the chemical language model, we explored the distribution of the compound embeddings, i.e. their internal representations in the model that were generated by the *ChemLM* served as input, together with four molecular properties. Those properties are the number of hydrogen-bond (i) acceptors, and (ii) donors, (iii) the molecular weight, and (iv) quantitative estimate of drug likeness (QED) visualized in UMAP plots (Figure 4). UMAP[55] was preferred to tSNE for the visualization of property distribution as it is better in preserving the global structure of the data providing a more accurate representation of the space. There are distinct clusters with low/high values, and a gradual change in the molecules' properties. To quantify the relationship between the embeddings generated by ChemLM and the chemical properties, we calculated the Lipschitz constant. The results of our analysis demonstrated that, for all properties, the median Lipschitz constant (k) lower median values compared to the random space. The p-values of the t-test showed that this is statistically important. The intrinsic evaluation indicated a chemically menaingful encodingof the space.

In summary, we introduce an efficient modeling approach for accurately predicting the molecular properties of chemical compounds. We achieved this by leveraging transfer learning, and domain adaptation phases, with key insights drawn from the model's evaluation. The outcomes highlight the substantial improvements achievable through self-supervised training on domain data, and data augmentation, leading to enhanced accuracy in molecular property prediction. Hyperparameter optimization also played a pivotal role in enhancing performance by identifying critical parameters in the model's architecture. Together, these findings have the potential to significantly benefit

the deployment of transformer models in the chemical domain. Notably, our suggested architecture has demonstrated superior performance compared to state-of-the-art models in various chemical tasks. At the same time, *ChemLM* generates a chemically meaningful encoding space. However, the main achievement of this model lies in its successful application to real-world data and predictive challenges. Specifically, it excels in identifying potent pathoblockers against *P. aeruginosa* from a very limited amount of training data. This suggests that the approach holds substantial promise to facilitate drug discovery in the future.

4 Methods

Data Description

We used two types of datasets to train, and evaluate the model's performance. The first one is the ZINC (v15) database, a public collection of millions of chemical compounds[56]. We retrieved the SMILES representations of the molecules, and used them in the pre-training stage of the *ChemLM* model. The second ones were three benchmark datasets from MoleculeNet [57] for prediction tasks of the physicochemical properties of molecules (Supplementary Table A7). BACE's target class indicates binding results for a set of inhibitors to β-secretase 1. The Blood Brain Barrier Penetration dataset (BBBP) is a collection of compounds from a study about compounds' brain barrier permeability in which labels indicate penetration or non-penetration. ClinTox includes compounds that can be used for the tasks of FDA approval status, and clinical trial toxicity. We evaluate the models on the second task.

Tokenization using Byte Pair Encoding (BPE)

One of the most critical steps is the tokenisation of SMILES. We consider each representation string equivalent to a sentence consisting of many tokens. In our approach, we use a computational way for tokenisation, Byte-level Byte Pair Encoding (BBPE) [58] as it is suggested for the RoBERTa model.

BPE [59] was first used as a data compression method. Its function relies on assigning new symbols to the most common pair of characters. Hence, it can find those sets and let us consider them as tokens. It is ideal for establishing a hybrid of word-/character- tokenisation, thus there is a combination of single atoms with pairs of highly frequent atoms. Another advantage is the user-defined vocabulary size, which is equivalent to the total number of tokens at the end of the procedure. Hence, the larger it is, the more pairs will be included, leading to different tokenization. The vocabulary size we have chosen is 10000, following the suggestion of the authors [58].

To learn the underlying sequence of bytes, it is necessary to train a BBPE tokenizer in a large corpus of SMILES like the ZINC database. This tokenizer can be used in different applications or datasets.

Transformers

The model is based on transformers that have an encoder-decoder architecture [33]. At the core of multi-head attention lies the concept of self-attention, which focuses on generating improved representations of the sequence elements (tokens) by considering their interactions with neighbouring elements. This self-attention mechanism is utilized within multi-head attention to enable the model to attend to multiple views of the sequence interactions simultaneously, resulting in more expressive, and informative representations. Thus, each layer of the encoder includes a multi-headed attention sublayer, and a position-wise fully connected feed-forward network followed by normalization layers. In a broad definition of attention, each token of the sequence is associated with two real-valued vector representations: (i) a key vector (k) from the input embedding space, and (ii) a value vector (v) from the output embedding space. These vectors can be either randomly initialized or pre-trained. The query vector (q) represents the sequence element for which one wants to obtain a new representation, and must belong to the same space as the key vectors. To calculate a new representation for the entire sequence, the key, (k), query (q), and value (v) vectors are calculated using dot multiplication of the embedding with the corresponding learned weight matrices. Matrix multiplications are deployed to leverage efficiency, and parallelization. Embeddings, query, key and value vectors are packed to matrices, X, K, Q, and V. Attention is calculated as described in equation 1, in which d_k stands for the dimension of vector k.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

Instead of using a single attention mechanism, researchers introduced a multihead one. Its benefit to the model lies in the information that captures from different representation subspaces at different positions.

$$MultiHead(Q, K, V) = Concat(head^1, ..., head^h)W^0, \tag{2}$$

where each $head_i$ is equal to

$$head_i = Attention(QW_i{}^Q, KW_i{}^K, VW_i{}^V) \tag{3}$$

and W_i is the weight matrix.

$$Attention(Q, K, V) = softmax(\frac{QK^t}{\sqrt{d_k}})V \tag{4}$$

Instead of performing a single attention mechanism, researchers introduced a multihead one. Its benefit to the model lies in the information that is captured

of recurrent or convolutional elements.

$$PE_{(pos,2i)} = \sin(\frac{pos}{10000^{\frac{2i}{d_{model}}}}) \quad (7)$$

$$PE_{(pos,2_{i+1})} = \cos(\frac{pos}{10000^{\frac{2i}{d_{model}}}}) \quad (8)$$

In formulas 7 and 8, *pos* stands for the position in the sequence, d_{model} for the dimension of the output embedding space and i, for the embedding index. As stated earlier, this architecture comes with many advantages, overcoming many of the sequence models' limitations. At first, self-attention mechanism enables to modeling interactions between distant tokens in the sequence, and, thus, captures long-term dependencies among them. In addition to that, they are highly scalable as they can handle variable-length input sequences thanks to the self-attention mechanism, which operates independently on each position. Moreover, the transformer's architecture is parallelizable and makes more efficient computations, restricting the training, and inference time. In addition, this architecture enables transfer learning. Learning general representations by pre-training a model on a corpus of unlabeled data, and then fine-tuning it on a specific task leads to improved performance.

The RoBERTa model was selected from a pool of autoencoder models. Based on BERT model, it utilizes learnable position embeddings, as opposed to sinusuidal position encodings as seen in formulas 7 and 8. The mean number of tokens per SMILES sequence is about 45. RoBERTa is an appropriate model for that sequence length. In addition, there are many training tasks for language models, such as next-sentence prediction. RoBERTa masks tokens of the sequence, and gets trained on predicting which is the masked token based on the context, and that technique is called masked language modeling (MLM). This is appropriate in our application and can leverage the model to learn the syntax, and grammar of SMILES. Hence, RoBERTa's characteristics matched our needs.

ChemLM implementation

ChemLM utilized MLM for training in the first two training stages, pre-training and domain adaptation. The domain adaptation training stage used multiple SMILES representations for each molecule. These representations were generated using SMILES enumeration, a data augmentation technique for SMILES strings [41]. We experimented with different numbers of augmentations (Supplementary Table A2) per molecule to find the best-performing one during the hyperparameter optimization approach. All training stages took place on an NVIDIA t4 GPU.

To implement *ChemLM*, HuggingFace[60] (version 0.0.8) was used to configure and train the RoBERTa model for the first training stages. A combination of Huggingface, and PyTorch[61] (version 1.6) was used for the supervised fine-tuning. In addition, scikit[62] (0.24.1) was deployed for hierarchical clustering, and evaluation metrics, and RDKit[63] (v2020.09.1.0) to produce the molecular properties for the intrinsic evaluation.

Intrinsic Evaluation

Quantitative evaluation: Aiming for a model that efficiently maps the compounds, it was essential to perform an intrinsic evaluation of our model, and quantitavely evaluate the distribution of physicochemical properties in the computational space. The examined molecular properties are molecular weight, hydrogen-bond donors, hydrogen-bond acceptors, lipophilicity, polar surface area, and rotatable bonds. For that purpose, we calculated the properties for the compounds of BBBP dataset using RDKit, and examined Lipschitz continuity for them as (equation 6),

$$d_f(f(e_1), f(e_2)) \leq k * d_e(e_1, e_2) \tag{9}$$

where f is the property value, d_f the absolute difference of these values for the embeddings e_1, e_2, k is the Lipschitz constant, and d_e the Euclidean distance of the embeddings. The rationale behind utilizing the Lipschitz constant lies in our intention to understand how the values of properties change with variations in the embeddings of molecules. When we have a bounded Lipschitz constant, denoted as k, it signifies that the properties change predictably, and consistently for different input compounds. In both cases, Lipschitz, and UMAP, we used embeddings that come from the weights of the last layer. We compare *ChemLM* with the random Lipschitz constant that is generated by shuffling the property values for 200 randomly chosen chemical compounds of BBBP dataset. To assess whether these results are statistically important, we also perform a one-tailed t-test on the distributions of *ChemLM*'s, and random space's Lipschitz constant. The distributions were produced by randomly selecting chemical compounds for 100 rounds. Then, the one-tailed t-test is used to calculate the p-value to assess whether our null hypothesis, that the k of our space is lower than the random one's, is true. Scipy (v. 1.8.0) is used to perform the t-test, and calculate the p-value using as an alternative argument, the 'greatest'.

Qualitative evaluation: in addition to the quantitative evaluation of the trained space, we projected the 768-dimensional vectors of molecule embeddings to 2D space using the UMAP algorithm. In this way, we can visually inspect the distribution of the aforementioned properties.

5 Acknowledgements

G.K. gratefully acknowledges financial support by the Lower Saxony Ministry for Science and Culture for the doctoral program "Drug Discovery and Cheminformatics for New Anti-Infectives (iCA)".

6 Author contributions

G.K., E.A., and A.C.M. conceived the study. G.K. implemented the software. A.C.M. and E.A. supervised the work. B.A. provided feedback on the computational approach. G.K., E.A., and A.C.M. have written the article. A.H. and M.E. have shared the experimental dataset, advised and guided the work on the corresponding part. M.E. and B.A. have contributed to writing, too. All authors have reviewed the article.

7 Data and code availabilities

Code and data are available in https://github.com/hzi-bifo/ChemLM.
Models are available in https://huggingface.co/gkallergis.

8 Competing interests

The authors declare no competing interests.

Appendix A

Table A1 Values range that was utilized for the hyperparameter optimization.

Parameter	Values range
Augmentation size	0,5,10,15,20,25, 40, 60, 80, 100
Number of hidden layers	4, 8, 12
Number of attention heads	8, 12, 16
Embeddings	Pooling Last layer - mean of tokens Last layer -first token Sum of hidden layers - mean of tokens Sum of hidden layers - first token Mean of hidden layers - mean of tokens Mean of hidden layers - first token

Table A2 Training time with regard to the augmentation size.

Augmentation size	Training time(s)
0	668
20	11083
40	26256
60	34731
80	44800
100	55594

Our findings suggest that the training time is linear to the augmentation size. The required time is quite low, and is not discouraging from using more representations.

Table A3 Selected hyperparameters for our models.

Model	Augmentation number	Embeddings type	Number of hidden layers	Number of attention heads
vanilla	-	Pooling	12	12
domain-adapted	80/100	Pooling	12	12
BBBP	80	Last layer -first token	8	12
BACE	100	Sum of hidden layers - mean of tokens	8	12
ClinTox	80	Last layer -first token	12	16
PA	100	Sum of hidden layers - mean of tokens	4	12

In the vanilla model, default hyperparameters from HuggingFace were utilized. The domain-adapted model shares the same values with vanilla, except for the augmentation number, in which the optimal value for each dataset was used. We identified the best hyperparameters for benchmark datasets (BACE, BBBP, ClinTox) through optimization on the validation dataset. Regarding the model for *Pseudomonas aeruginosa* (PA), the lack of validation dataset did not allow us to follow a similar procedure. We selected the best hyperparameters according to the values derived from the successful configurations identified on benchmark datasets, except that we used fewer layers, because of the small training set size. In addition, we also investigated another setting for the embeddings type (embeddings of the first token of the last layer), a larger number of hidden layers (12 instead of 4) and a lower augmentation number (80). Results are shown for the model with the best performance, selecting the best of these models.

Table A4 Participation of each structural class in the 5-branch setting, and its percentage of highly potent compounds.

Hierarchical folds	A	B	C	D	E	Highly potent compounds	Number of compounds
1	36	29	0	0	13	88%	78
2	40	15	1	1	6	67%	63
3	20	4	2	1	22	40%	49
4	0	0	11	0	8	21%	19
5	10	0	0	0	0	50%	10

Table A5 Performance comparison of property prediction models over the test set of a 5-fold cross-validation setting over the experimental dataset.

Model	F1	AUC	Precision	Recall	Accuracy
MolBERT	0.495	0.5	0.553	0.6	0.714
MPNN	0.604	0.592	0.661	0.591	0.789
GAT	0.563	0.583	0.635	0.583	0.714
GCNN	0.571	0.567	0.575	0.568	0.651
ChemLM	**0.899**	**0.900**	**0.900**	**0.900**	**0.900**

The median metric value of each model is demonstrated.

Table A6 Predictive performance of *ChemLM* and state-of-the-art models on the positive class (highly potent pathoblockers).

Hierarchical Folds	ChemLM	MPNN	GAT	GCNN	MolBERT
1	0.917	**0.938**	0.915	0.906	0.869
2	**0.825**	0.796	0.820	0.771	0.775
3	**0.895**	0.723	0.618	0.627	0.695
4	**0.888**	0.333	0.000	0.750	0.000
5	**0.888**	0.750	0.727	0.000	0.600

The F1-score is reported as evaluation metric in this this table.

Table A7 Description of the evaluation datasets.

Datasets	Number of compounds	Percentage of positive class
BACE	1513	45.7%
BBBP	2039	76.5%
ClinTox	1478	7.6%

Table A8 Comparison of *ChemLM* on BBBP dataset with its simpler versions, and state-of-the-art models in more evaluation metrics.

Model	F1	AUC	Precision	Recall	Accuracy
MolBERT	**0.891**	**0.888**	**0.895**	**0.888**	**0.928**
MPNN	0.783	0.788	0.778	0.788	0.841
GAT	0.747	0.711	0.847	0.711	0.85
GCNN	0.695	0.664	0.820	0.664	0.828
ChemLM vanilla	0.689	0.674	0.72	0.674	0.799
ChemLM domain-adapted	0.823	0.811	0.837	0.81	0.87
ChemLM domain-adapted & optimized	0.879	0.885	0.874	0.884	0.912

Table A9 Comparison of *ChemLM* on BACE dataset with its simpler versions, and state-of-the-art models in more evaluation metrics.

Model	F1	AUC	Precision	Recall	Accuracy
MolBERT	**0.814**	**0.814**	**0.813**	**0.814**	**0.816**
MPNN	0.729	0.733	0.731	0.733	0.729
GAT	0.666	0.704	0.769	0.704	0.680
GCNN	0.69	0.692	0.731	0.692	0.71
ChemLM vanilla	0.508	0.554	0.603	0.553	0.584
ChemLM domain-adapted	0.661	0.662	0.674	0.662	0.673
ChemLM domain-adapted & optimized	0.804	0.803	0.804	0.803	0.805

Table A10 Comparison of *ChemLM* on ClinTox dataset with its simpler versions, and state-of-the-art models in more evaluation metrics.

Model	F1	AUC	Precision	Recall	Accuracy
MolBERT	0.667	0.671	0.662	0.671	0.915
MPNN	0.638	0.593	0.870	0.593	0.938
GAT	0.592	0.566	0.719	0.566	0.928
GCNN	0.655	0.614	0.784	0.614	0.935
ChemLM vanilla	0.480	0.500	0.462	0.500	0.925
ChemLM domain-adapted	0.823	0.750	0.980	0.750	0.962
ChemLM domain-adapted & optimized	**0.916**	**0.864**	**0.989**	**0.864**	**0.979**

Table A11 Comparison of *ChemLM* on benchmark datasets with state-of-the-art models in prediction of the positive class using F1-score as evaluation metric.

Model	ClinTox	BACE	BBBP
MolBERT	0.378	**0.796**	**0.955**
MPNN	0.308	0.721	0.895
GAT	0.222	0.734	0.909
GCNN	0.345	0.611	0.897
ChemLM domain-adapted & optimized	**0.842**	0.785	0.942

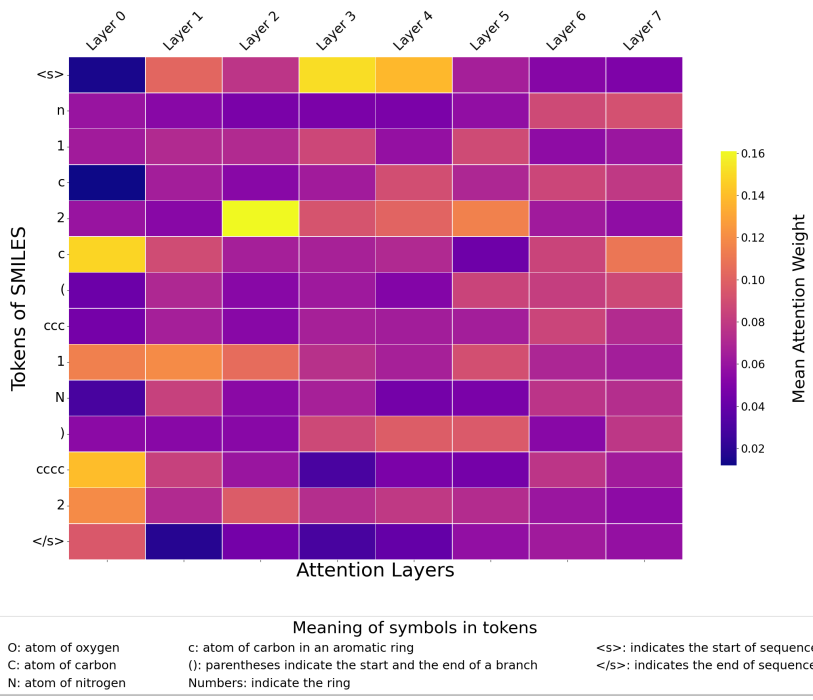


Fig. A1 Heatmap of the attention distribution in the tokens of a SMILES sequence. It depicts the sum of attention a token receives from all attention heads in each layer of the model.

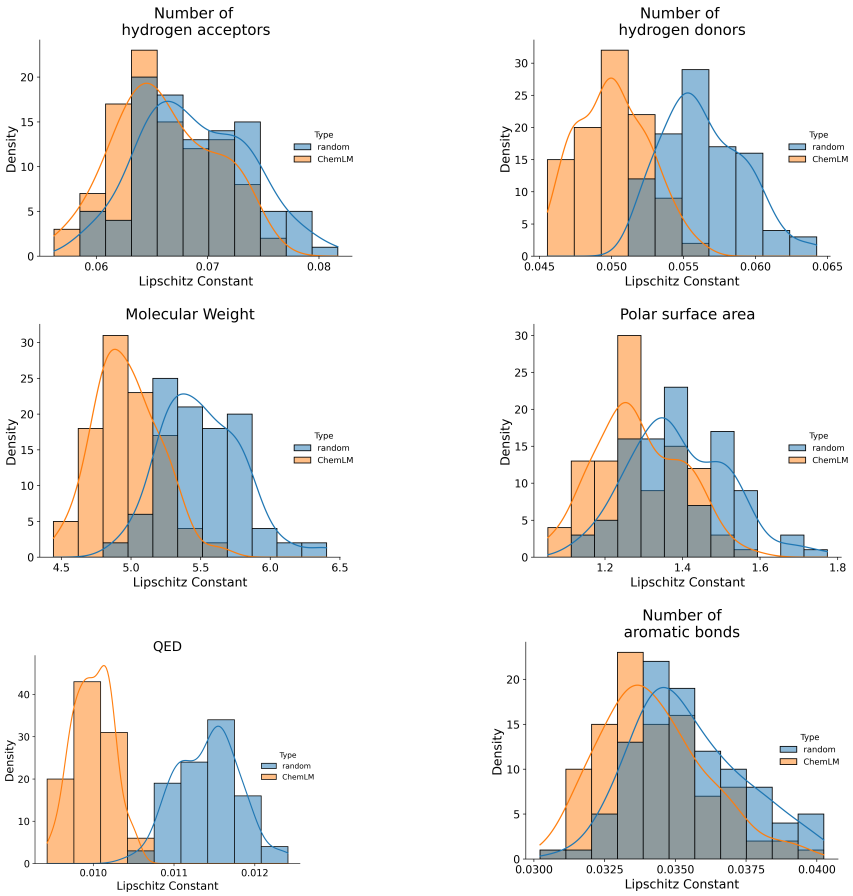


Fig. A2 Distribution plots of Lipschitz constant for ChemLM, and random space.

References

[1] Mohs, R.C., Greig, N.H.: Drug discovery and development: Role of basic biological research. Alzheimers. Dement. **3**(4), 651–657 (2017). https://doi.org/10.1016/j.trci.2017.10.005

[2] Paul, S.M., Mytelka, D.S., Dunwiddie, C.T., Persinger, C.C., Munos, B.H., Lindborg, S.R., Schacht, A.L.: How to improve RD productivity: The pharmaceutical industry's grand challenge **9**(3), 203–214 (2010). https://doi.org/10.1038/nrd3078

[3] Hingorani, A.D., Kuan, V., Finan, C., Kruger, F.A., Gaulton, A., Chopade, S., Sofat, R., MacAllister, R.J., Overington, J.P., Hemingway, H., Denaxas, S., Prieto, D., Casas, J.P.: Improving the odds of drug development success through human genomics: modelling study. Scientific Reports **9**(1), 18911 (2019). https://doi.org/10.1038/s41598-019-54849-w

[4] Silver, L.L.: Challenges of antibacterial discovery. Clinical Microbiology Reviews **24**(1), 71–109 (2011) https://journals.asm.org/doi/pdf/10.1128/CMR.00030-10. https://doi.org/10.1128/CMR.00030-10

[5] Kwon, J.H., Powderly, W.G.: The post-antibiotic era is here. Science **373**(6554), 471–471 (2021) https://www.science.org/doi/pdf/10.1126/science.abl5997. https://doi.org/10.1126/science.abl5997

[6] Karakonstantis, S., Kritsotakis, E.I., Gikas, A.: Pandrug-resistant gram-negative bacteria: a systematic review of current epidemiology, prognosis and treatment options. J. Antimicrob. Chemother. **75**(2), 271–282 (2020). https://doi.org/10.1093/jac/dkz401

[7] Ma, J., Sheridan, R.P., Liaw, A., Dahl, G.E., Svetnik, V.: Deep neural nets as a method for quantitative structure-activity relationships. Journal of Chemical Information and Modeling **55**(2), 263–274 (2015). https://doi.org/10.1021/ci500747n

[8] Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., Blaschke, T.: The rise of deep learning in drug discovery. Elsevier Ltd (2018). https://doi.org/10.1016/j.drudis.2018.01.039

[9] Zhang, Q.Y., Aires-de-Sousa, J.: Random forest prediction of mutagenicity from empirical physicochemical descriptors. Journal of Chemical Information and Modeling **47**(1), 1–8 (2007). https://doi.org/10.1021/ci050520j

[10] Zernov, V.V., Balakin, K.V., Ivaschenko, A.A., Savchuk, N.P., Pletnev, I.V.: Drug Discovery Using Support Vector Machines. The Case Studies of Drug-likeness, Agrochemical-likeness, and Enzyme Inhibition Predictions. Journal of Chemical Information and Computer Sciences **43**(6), 2048–2056 (2003). https://doi.org/10.1021/ci0340916

[11] D Duvenaud, D Maclaurin, J.A.-I.: Convolutional networks on graphs for learning molecular fingerprints. Adv Neural Inf Process Syst **2015**, 2224–2232 (2015)

[12] Y, X., J, P., L, L.: Deep Learning Based Regression and Multiclass Models for Acute Oral Toxicity Prediction with Automatic Chemical Feature Extraction. Journal of chemical information and modeling **57**(11), 2672–2685 (2017). https://doi.org/10.1021/ACS.JCIM.7B00244

[13] Wang, X., Li, Z., Jiang, M., Wang, S., Zhang, S., Wei, Z.: Molecule Property Prediction Based on Spatial Graph Embedding. Journal of Chemical Information and Modeling **59**(9), 3817–3828 (2019). https://doi.org/10.1021/ACS.JCIM.9B00410

[14] Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural Message Passing for Quantum Chemistry. 34th International Conference on Machine Learning, ICML 2017 **3**, 2053–2070 (2017) arXiv:1704.01212v2

[15] Withnall, M., Lindelöf, E., Engkvist, O., Chen, H.: Building attention and edge message passing neural networks for bioactivity and physical–chemical property prediction. Journal of Cheminformatics 2020 12:1 **12**(1), 1–18 (2020). https://doi.org/10.1186/S13321-019-0407-Y

[16] Veličković, P., Casanova, A., Liò, P., Cucurull, G., Romero, A., Bengio, Y.: Graph Attention Networks. 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings (2017) arXiv:1710.10903. https://doi.org/10.48550/arxiv.1710.10903

[17] Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., Palmer, A., Settels, V., Jaakkola, T., Jensen, K., Barzilay, R.: Analyzing Learned Molecular Representations for Property Prediction. Journal of Chemical Information and Modeling **59**(8), 3370–3388 (2019) arXiv:1904.01561. https://doi.org/10.1021/acs.jcim.9b00237

[18] Chomsky, N.: Syntactic structures. De Gruyter Mouton (2009)

[19] Harris, Z.S.: Distributional structure. Word World **10**(2-3), 146–162 (1954). https://doi.org/10.1080/00437956.1954.11659520

[20] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems **26** (2013)

[21] Rieger, B.B.: On distributed representation in word semantics. https://www.uni-trier.de/fileadmin/fb2/LDV/Rieger/Publikationen/Aufsaetze/91/icsi91.pdf. Accessed: 2023-11-24. https://www.uni-trier.de/fileadmin/fb2/LDV/Rieger/Publikationen/Aufsaetze/91/icsi91.pdf

[22] Asgari, E., Mofrad, M.R.K.: Continuous distributed representation of biological sequences for deep proteomics and genomics. PLoS ONE **10**(11) (2015)

[23] Asgari, E., McHardy, A., Mofrad, M.R.K.: Probabilistic variable-length segmentation of protein sequences for discriminative motif mining (dimotif) and sequence embedding (protvecx). Sci Rep (2019). https://doi.org/10.1101/345843

[24] Asgari, E.: Life language processing: deep learning-based language-agnostic processing of proteomics, genomics/metagenomics, and human languages. University of California, Berkeley (2019)

[25] Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., BHOWMIK, D., Rost, B.: Prottrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. bioRxiv (2020) https://www.biorxiv.org/content/early/2020/07/21/2020.07.12.199554.full.pdf. https://doi.org/10.1101/2020.07.12.199554

[26] Benegas, G., Albors, C., Aw, A.J., Ye, C., Song, Y.S.: GPN-MSA: an alignment-based DNA language model for genome-wide variant effect prediction (2023). https://doi.org/10.1101/2023.10.10.561776. https://www.biorxiv.org/content/10.1101/2023.10.10.561776v1

[27] Moret, M., Pachon Angona, I., Cotos, L., Yan, S., Atz, K., Brunner, C., Baumgartner, M., Grisoni, F., Schneider, G.: Leveraging molecular structure and bioactivity with chemical language models for de novo drug design. Nat. Commun. **14**(1), 114 (2023). https://doi.org/10.1038/s41467-022-35692-6

[28] Weininger, D.: SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules. Journal of Chemical Information and Computer Sciences **28**(1), 31–36 (1988). https://doi.org/10.1021/ci00057a005

[29] Schwaller, P., Gaudin, T., Lányi, D., Bekas, C., Laino, T.: "Found in

Translation": predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. Chemical Science **9**(28), 6091–6098 (2018) arXiv:1711.04810. https://doi.org/10.1039/C8SC02339E

[30] Öztürk, H., Ozkirimli, E., Özgür, A.: A novel methodology on distributed representations of proteins using their interacting ligands **34**, 295–303 (2018) arXiv:1801.10199. https://doi.org/10.1093/bioinformatics/bty287

[31] Chakravarti, S.K.: Distributed Representation of Chemical Fragments. ACS Omega **3**(3), 2825–2836 (2018). https://doi.org/10.1021/acsomega.7b02045

[32] Skinnider, M.A., Greg Stacey, R., Wishart, D.S., Foster, L.J.: Chemical language models enable navigation in sparsely populated chemical space. Nat Mach Intell 3, 759–770 (2021). https://doi.org/10.1038/s42256-021-00368-1

[33] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need **2017**, 5999–6009 (2017) 1706.03762

[34] Chithrananda, S., Grand, G., Ramsundar, B.: ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. http://arxiv.org/abs/2010.09885 (2020)

[35] Irwin, R., Dimitriadis, S., He, J., Bjerrum, E.J.: Chemformer: a pre-trained transformer for computational chemistry. Machine Learning: Science and Technology **3**(1), 015022 (2022). https://doi.org/10.1088/2632-2153/ac3ffb

[36] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1423. https://aclanthology.org/N19-1423

[37] Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., Hon, H.-W.: Unified Language Model Pre-Training for Natural Language Understanding and Generation, pp. 13063–13075. Curran Associates Inc., Red Hook, NY, USA (2019)

[38] Varnek, A., Gaudin, C., Marcou, G., Baskin, I., Pandey, A.K., Tetko, I.V.: Inductive transfer of knowledge: Application of multi-task learning and Feature Net approaches to model tissue-air partition coefficients. Journal of Chemical Information and Modeling **49**(1), 133–144 (2009). https://

doi.org/10.1021/ci8002914

[39] Beltagy, I., Lo, K., Cohan, A.: SciBERT: A Pretrained Language Model for Scientific Text. EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, 3615–3620 (2019) arXiv:1903.10676

[40] Howard, J., Ruder, S.: Universal Language Model Fine-tuning for Text Classification. ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers) **1**, 328–339 (2018) arXiv:1801.06146

[41] Bjerrum, E.J.: SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules (2017). http://arxiv.org/abs/1703.07076

[42] Vig, J., Belinkov, Y., John, H., Paulson, A.: Analyzing the Structure of Attention in a Transformer Language Model. In: Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pp. 63–76 (2019)

[43] Schütz, C., Ho, D.K., Hamed, M.M., Abdelsamie, A.S., Röhrig, T., Herr, C., Kany, A.M., Rox, K., Schmelz, S., Siebenbürger, L., Wirth, M., Börger, C., Yahiaoui, S., Bals, R., Scrima, A., Blankenfeldt, W., Horstmann, J.C., Christmann, R., Murgia, X., Koch, M., Berwanger, A., Loretz, B., Hirsch, A.K.H., Hartmann, R.W., Lehr, C.M., Empting, M.: A New PqsR Inverse Agonist Potentiates Tobramycin Efficacy to Eradicate Pseudomonas aeruginosa Biofilms. Advanced Science **8**(12) (2021). https://doi.org/10.1002/ADVS.202004369

[44] Schütz, C., Hodzic, A., Hamed, M., Abdelsamie, A.S., Kany, A.M., Bauer, M., Röhrig, T., Schmelz, S., Scrima, A., Blankenfeldt, W., Empting, M.: Divergent synthesis and biological evaluation of 2-(trifluoromethyl)pyridines as virulence-attenuating inverse agonists targeting PqsR. European journal of medicinal chemistry **226** (2021). https://doi.org/10.1016/J.EJMECH.2021.113797

[45] Hamed, M.M., Abdelsamie, A.S., Rox, K., Schütz, C., Kany, A.M., Röhrig, T., Schmelz, S., Blankenfeldt, W., Arce-Rodriguez, A., Borrero-de Acuña, J.M., Jahn, D., Rademacher, J., Ringshausen, F.C., Cramer, N., Tümmler, B., Hirsch, A.K.H., Hartmann, R.W., Empting, M.: Towards Translation of PqsR Inverse Agonists: From In Vitro Efficacy Optimization to In Vivo Proof-of-Principle. Advanced Science, 2204443 (2023). https://doi.org/10.1002/ADVS.202204443

[46] Zender, M., Witzgall, F., Kiefer, A., Kirsch, B., Maurer, C.K., Kany, A.M., Xu, N., Schmelz, S., Börger, C., Blankenfeldt, W., Empting, M.: Flexible

Fragment Growing Boosts Potency of Quorum-Sensing Inhibitors against Pseudomonas aeruginosa Virulence. Chemmedchem **15**(2), 188 (2020). https://doi.org/10.1002/CMDC.201900621

[47] PqsR Inverse Agonists 2018 Ref. No: WO2020007938A1 (EP18181475)

[48] New PqsR Inverse Agonist 2020 (EP20150104).

[49] Novel PqsR Inverse Agonists 2020 Ref. No: WO2021136805A1 (EP20150119).

[50] Schütz, C., Empting, M.: Targeting the Pseudomonas quinolone signal quorum sensing system for the discovery of novel anti-infective pathoblockers. M. Beilstein J. Org. Chem. **14**(14) (2018). https://doi.org/10.3762/bjoc.14.241

[51] Kipf, T.N., Welling, M.: Semi-Supervised Classification with Graph Convolutional Networks. 5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings (2016) arXiv:1609.02907

[52] Fabian, B., Edlich, T., Gaspar, H., Segler, M., Meyers, J., Fiscato, M., Ahmed, M.: Molecular representation learning with language models and domain-relevant auxiliary tasks (2020) arXiv:2011.13230 [cs.LG]

[53] Ramsundar, B., Eastman, P., Walters, P., Pande, V., Leswing, K., Wu, Z.: Deep learning for the life sciences. O'Reilly Media (2019)

[54] Fabian, B., Héléna, T.E., Segler, G.M., Meyers, J., Fiscato, M., Benevolentai, M.A.: Molecular representation learning with language models and domain-relevant auxiliary tasks arXiv:2011.13230v1

[55] McInnes, L., Healy, J., Melville, J.: UMAP: Uniform manifold approximation and projection for dimension reduction (2018) arXiv:1802.03426 [stat.ML]

[56] Sterling, T., Irwin, J.J.: ZINC 15 – Ligand Discovery for Everyone. Journal of Chemical Information and Modeling **55**(11), 2324–2337 (2015). https://doi.org/10.1021/ACS.JCIM.5B00559

[57] Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V.: MoleculeNet: A benchmark for molecular machine learning. Chemical Science **9**(2), 513–530 (2018) arXiv:1703.00564. https://doi.org/10.1039/c7sc02664a

[58] Wang, C., Cho, K., Gu, J.: Neural Machine Translation with Byte-Level Subwords. AAAI 2020 - 34th AAAI Conference on Artificial Intelligence,

9154–9160 (2019) arXiv:1909.03341

[59] Gage, P.: A new algorithm for data compression. C Users J. **12**(2), 23–38 (1994)

[60] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: HuggingFace's Transformers: State-of-the-art Natural Language Processing (2019). http://arxiv.org/abs/1910.03771

[61] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An imperative style, high-performance deep learning library (2019) arXiv:1912.01703 [cs.LG]

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

[63] G., L.: RDKit: Open-source cheminformatics. https://www.rdkit.org