LM-DTA: Using Language models to predict Drug-Target Affinity

Research project by Flurin Hidber
supervised by Dr. F. Grisoni and M. Moret
in the Computer-Assisted Drug Design (CADD) Lab
of Prof. G. Schneider
March 9, 2020

Abstract

We implemented a flexible general framework to solve the drug-target affinity (DTA) problem with deep learning methods and challenged modern approaches with a network using LSTM blocks to learn deep representations of both drugs and targets. We compared our LM-DTA model with a selected baseline and extended our models methodology to a control setting showing that language models excel at handling pair-sequence data with very long sequences and applied and evaluated the use of transfer learning (TFL) in this framework, to alleviate the scarce-data problem met working with DTA values. Our optimized LM-DTA model managed to outperform the baseline approach and state-of-the-art method on the KIBA dataset.

A Introduction

The first step in drug discovery is the identification of novel drug-target interactions. Today databases serve as routine resources for the quantification of these interactions by providing drug-target affinity (DTA) values for drug-target pairs. Leveraging these databases and building computational models on top of this data allows us to predict target-interactions of novel drugs or interactions of known drugs with new targets. Now with the advances in the field of natural language processing (NLP) and by applying these novel methods to the sequence data of both drug and target their use could become a powerful tool in the modern drug discovery process. We challenge state-of-the-art models using sequence data of drug and target, by developing an approach using an NLP method, LSTM, and compare our model to previous works. We further explore a technique called transfer learning (TFL) attempting to improve our models performance and implement a control setting to validate our results.

A.1 Related Work

The problem of predicting interactions between a pair of drug and a target protein in contrast to the problem of predicting a chemical property associated with just a chemical compound requires processing an input pair of sequence data. DTA prediction is a regression problem, given input pairs we wish to predict a value representing their interaction, their affinity value.
There are various approaches in what form drug and target are represented. While structure-based approaches such as molecular docking [*K. Chaudhary et al.* 2016] have become a powerful tool in drug-discovery, the rise of deep learning and its success in many research fields, including NLP has opened new ways to approach the problem of DTA prediction. Today there is a vast abundance of sequence data of the human genome, which is ever increasing in the age of Next-Generation Sequencing (NGS) techniques and their high-throughput measurements. An approach using sequence data of target proteins in form of their amino-acid sequence predicting a drug's DTA with this target could serve as a valuable method in drug-discovery. Recent studies such as [*H. Öztürk et al.*, 2018], [*I. Lee et al.*, 2019] or [*M. Thafar et al.*, 2019] have attempted to leverage and compared deep learning methods to solve the DTA problem using only sequence data and shown success.

A.1.1 General Framework for Sequence-Based DTA Models

We observed a common pattern in the framework of deep learning models solving the DTA problem. The general deep learning architecture of a model used for DTA-predictions consists of three blocks (see Figure 1).

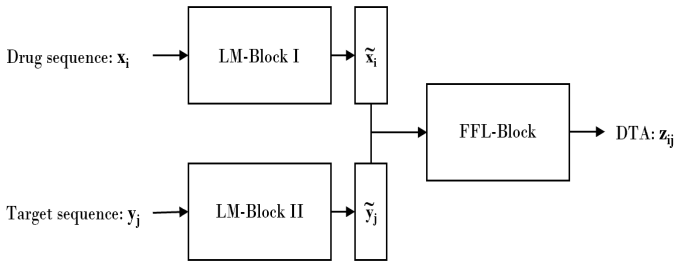


Figure 1: Architecture of the general framework of our proposed DTA model. Using only the drug's SMILES and target protein amino-acid sequence as input the DTA model predicts an affinity value for their interaction.

The binary input consists of a pair of drug and target is handled by first passing the drug and target separately through a part of the network that learns to abstract relevant information contained in the sequence data, i.e. we produce an intermediary representation, which then is combined and fed through a final network-block that maps the combined representations to the DTA value.

A.1.2 Convolutional Neural Networks

The separate blocks used to abstract a representation from sequence data termed language-model blocks (LM-Blocks) are what distinguishes the various proposed methods to a high degree. In our project we reference an approach applying LM-Blocks using convolutional layers as our baseline [*H. Öztürk et al.* 2018] and re-implemented their work using our general framework, extending the approaches applicability to our control experiment settings (see A.2).

A.2 Control

To show that our model manages to extract a meaningful representation of a target sequence and is not merely clustering targets with a similar sequence we defined a control setting: Instead of training the proposed model with pairs of drug and target sequences, we only used the drug's sequence, but changed the target's input from its sequence to a target-specific identifier. Instead of using a LM-Block we then generated a target-specific representation by applying an embedding-layer to the identifier.

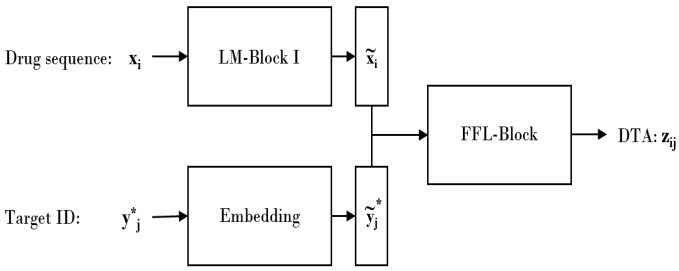


Figure 2: This is the control setting's architecture of the general framework using a unique identifier for each target instead of its sequence data. We use an embedding layer to generate a representation. Combined with the representation of the drug processed by an LM-Block we then process the combined representation through the FFL-Block to predict the drug-target affinity.

If indeed our model was clustering targets with similar associated drugs that would mean our model could not be applied outside of the known target domain. Outperforming this identifier-based control however would mean that we gained predictive power by inputting our model the whole target sequence.

A.3 Data

The data in the DTA problem setting consists of the drug's input sequence denoted as $x_i \in X$ where $\mathrm{X} = \{x_0, ..., x_n\}$ and the target's sequence denoted as $y_j \in Y$ where $\mathrm{Y} = \{y_0, ..., y_m\}$. The affinity values of a drug-target pair is given by z_{ij} with $i \in \{0, ..., n\}$ and $j \in \{0, ..., m\}$. A DTA model maps an input pair (x_i, y_j) to an affinity value z_{ij}. The dimensionality of the data $Z \in \mathbb{R}^{n \times m}$ depends on the choice of our dataset.

Each drug x_i is handled as a string sequence in the *Simplified Molecular Input Line Entry System*, the SMILES format [*D. Weininger et al.*, 1988], with the string length l_{x_i} dependent on the drug. The target protein is given by its amino acid sequence with length l_{y_j}.

We chose two benchmark datasets, Davis' dataset [*Davis et al.*, 2011] and the KIBA dataset [*J. Tang et al.*, 2014], containing this type of paired sequence data with corresponding drug-target affinity values, mapping pairs of drug- and target-sequences to affinities. Related deep learning approaches have also used these curated datasets when solving the DTA problem, so to guarantee the quality of the data during training of the model and to allow cross-comparison with previous state-of-the-art models we also applied our method on this data. Davis' dataset contains interactions for $n = 68$ drugs and $m = 442$ targets, with a total of 30056 specific drug-target affinity values. The KIBA dataset contains interactions for $n = 2111$ drugs and $m = 229$ targets, resulting in 118254 interaction-pairs. Most of the interactions are negative pairs (no interaction). Each interaction-pair represents one training sample. The prediction property here referred to as affinity values are dissociation constant values K_d in Davis' dataset, while KIBA uses a combination of K_i, K_d and IC_{50}.

A.4 Transfer Learning

While the number of training samples, that is the number of total interactions in a dataset, may seem quite large, the number of targets or drugs itself is rather small (see B).

In deep learning a common methodology to approach the lack of expensive or rare samples is transfer learning (TFL). When a specific type of data is scarce, but a very similar type of data exists we can try to obtain a better starting point of training for our network by first training a network on this secondary dataset. This approach can be extended to even a different type of optimization target as long as the objective and training lead to the network being trained to extract relevant information for our primary task.

For TFL on SMILES data we first had to chose a training regime where a network including our models LM-Block was optimized on a objective which would translate enough to our DTA-problem's domain to allow for the representation learned in this setting to be meaningful for our model. The objective is a step-wise next-token prediction, detailed in previous work on a generative design of drugs in SMILES format [*F.Grisoni et al.*, 2020] which we hypothesize leads to the LM-Block learning the underlying structure of this chemical language.

Further we apply the same training methodology on the target domain with amino-acid sequences. This attempt on TFL for proteins has been investigated [*E.Alley et al.*, 2019] termed *sequence-based deep representation learning* and shown state-of-the-art results in many target-property prediction domains.

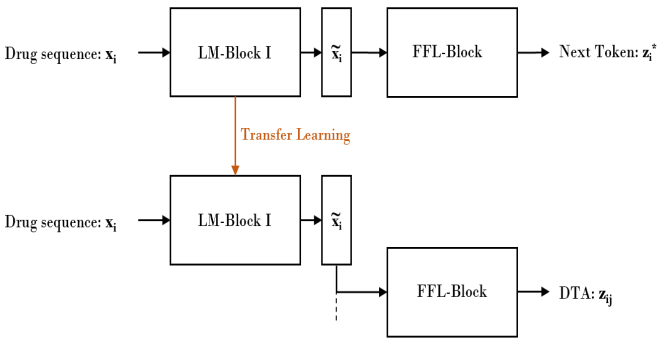


Figure 3: Visual representation of TFL. After training the network including the LM-Block generating in a representation learning setting we transfer the optimized weights to our LM-DTA model's LM-Block, shown here exemplary for the drug's representation learning.

We use TFL to pre-train both our LM-Blocks separately to augment our model with databanks on their respective chemical domain which we theorize leads to a general "understanding" of that domain, making a transfer of knowledge towards a combined DTA model feasible. This weight transfer of LM-Block-weights can be expressed as teaching the network the grammer of the chemical language of SMILES, respectively the grammer of the amino-acid sequence domain, called deep representation learning. For data we leverage the *ChemBl25 dataset* to mine drug compounds in SMILES format and the *Uniprot databank* to extract the human proteome as amino-acid sequences.

B Methods

Herein, we explain the architecture of our proposed network LM-DTA implemented in our framework using deep learning as an approach to solve the DTA problem and further outline the specifics of our approach using long short-term memory (LSTM) layers. The specifics of the build of the baseline model using convolutional layers in the LM-Blocks can be found in detail in their original description thereof.

The general framework consisted of three blocks, each of which can be thought of as its own network with an associated task. There were two language model blocks (LM-Block I and LM-Block II) that took as an input sequence data of a drug-target pair.

The basic idea of the presented framework was that the information of the sequence data of both drug and target was first passed through their respective LM-Block resulting in a vector representation $\tilde{x}_i$ and $\tilde{y}_j$. These representations were then combined by vector concatenation commonly denoted as $\tilde{x}_i \frown \tilde{y}_j$ and fed into the third network block, consisting of several feedforward-layers of decreasing size and intermediary non-linearities (FFL-Block), producing a predicted affinity value.

B.1 Data Processing

Using an upper length bound, $l_x = 85$ for SMILES and $l_y = 1200$ for proteins in the Davis dataset, and $l_x = 100$ for SMILES and $l_y = 1000$ for proteins in the KIBA dataset we excluded all sequences longer than our bounds. We padded the sequences to the upper bounds used to exclude outliers. These upper bounds were chosen arbitrarily by the authors of our baseline approach and reused here to allow for comparison of the model performances. Using common practice we trained our model using batch gradient descent with batches of size $b = 64$ for 200 epochs. Denoting $\mathbf{x} \in \mathbb{Z}^{b \times l_x}$ and $\mathbf{y} \in \mathbb{Z}^{b \times l_y}$ we defined $(\mathbf{x}, \mathbf{y})$ as input and mapped it to output $\mathbf{z} \in \mathbb{R}^b$. We scaled the affinity values, in Davis' case the negative log of K_d and in KIBA the KIBA score linearly to between 0 and 1.

B.2 LM-Blocks

While our architecture did permit flexibility in the design of each LM-Block, the models presented here always featured identical LM-Blocks.

Each LM-Block consisted of two types of layers, first the input sequences were passed through an embedding layer mapping with embedding size l_e s.t. $\mathbf{x} \in \mathbb{Z}^{b \times l_x} \rightarrow \mathbb{R}^{b \times l_e}$ and analogously $\mathbf{y} \in \mathbb{Z}^{b \times l_y} \rightarrow \mathbb{R}^{b \times l_e}$. Then we applied 2-layer LSTM-blocks with hidden-states of size $h = 256$ resulting in representations $\tilde{\mathbf{x}} \in \mathbb{R}^{b \times h}$ and $\tilde{\mathbf{y}} \in \mathbb{R}^{b \times h}$.

An advantage of using the hidden state of an LSTM as a representation of sequence data compared to convolutional layers is that the representation is not sequentially related. Meaning that the mapping from the input sequences x_i and y_j to our intermediary representations $\tilde{x}_i$ and $\tilde{y}_j$, i.e. their abstraction of the input where the position p in the representation vector e.g. $\tilde{x}_{i_p}$ shows no direct relation to any specific input sequence position k e.g. x_{i_k}.

Both representation vectors $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ were then concatenated into $\tilde{\mathbf{x}} \frown \tilde{\mathbf{y}} \in \mathbb{R}^{b \times 2h}$.

B.3 FFL-Block

The FFL-Block consisted of three linear layers with sizes of 1024, 1024 and 512 nodes, between each of which we applied a non-linearity function, ReLU, and used dropout. Finally an output node with a sigmoid activation function produced our models prediction.

B.4 Control

In our control experiment instead of using an LM-Block for the target, we tagged all targets with a unique identifier from $\mathrm{Y}^* \in \mathbb{Z}^m$. The identifier was passed through an embedding layer with output size l_{e*} where l_{e*} was equal to the size of the hidden state h, thus not changing the dimensionality of our model compared to the proposed settings.

B.5 TFL Pre-Training

The data used for TFL, the ChemBl25 and UniProt dataset, were processed in equal fassion to the DTA-data of Davis and KIBA. We produced valid input sequence data for the respective LM-Blocks.

The TFL regime used a surrogate loss for next token prediction to pre-train the weights of our network, then the FFL-Block returned an output vector representing the probabilities of the next token. The network architecture inspiring the TFL training with SMILES by [*Grisoni et al.*, 2019] was used for both the LM-Block I and II. Only the LM-Blocks weights were transferred to the LM-DTA model.

C Results

We report mean squared errors (MSE) and mean absolute errors (MAE) of our proposed LM-DTA model and the baseline model using convolutional layers was (we re-implemented the baseline approach to produce performance values in the Control experiment). The model with and without TFL are reported for both datasets, Davis' (Table 1) and KIBA (Table 2). The errors correspond to the respective loss of the predicted logarithmic affinity values. We include the standard deviation (see Table 1 and 2) of the error of the datasets.

	Control, with TFL ⋆		Control, no TFL ⋆		with TFL ⋆		no TFL ⋆	
	MSE (std)	MAE (std)	MSE (std)	MAE (std)	MSE (std)	MAE (std)	MSE (std)	MAE (std)
CNN			0.609 (1.851)	0.457 (0.633)			**0.261**	
LSTM	0.901 (1.805)	0.794 (0.521)	1.083 (1.577)	0.920 (0.486)	0.580 (1.688)	0.476 (0.594)	0.552 (1.712)	0.451 (0.590)

Table 1: Model performances on the Davis dataset.

	Control, with TFL ⋆		Control, no TFL ⋆		with TFL ⋆		no TFL **ours**	
	MSE (std)	MAE (std)	MSE (std)	MAE (std)	MSE (std)	MAE (std)	MSE (std)	MAE (std)
CNN			0.466 (1.130)	0.488 (0.477)			0.194	
LSTM	0.550 (1.272)	0.532 (0.516)	0.454 (1.015)	0.494 (0.458)	0.302 (0.878)	0.350 (0.423)	**0.181 (0.642)**	**0.261 (0.337)**

Table 2: Model performances on the KIBA dataset.

We report the best models performances, preliminary results are marked with ⋆ (see E.1). The hyperparameter settings reported in the section B only apply to our proposed LM-DTA model, without TFL or the use of control settings. Due to incomplete measurements and to provide our complete analysis table we opted to report preliminary results based on models using batches of size $b = 1$. This only applies to our LM-DTA model. The trends in the results of Davis for the control experiment and TFL are very similar to the results on KIBA: With both datasets we noticed the control experiment and application of TFL lead to inferior results, in case of TFL freezing the pre-trained layers' weights actually prevented the training completely (see D.3).

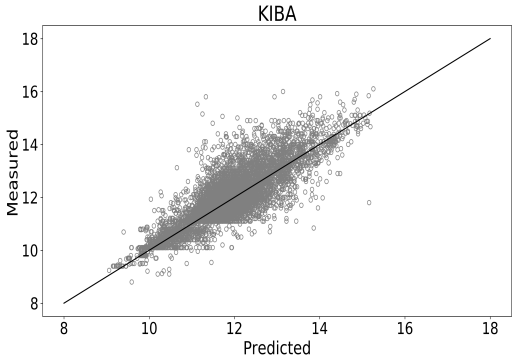


Figure 4: Our proposed LM-DTA models' re-scaled predictions plotted against the true KIBA scores.

D Discussion

We managed to outperform the baseline approach using convolutional network layers in place of our LSTMs on the KIBA dataset with an optimized LM-DTA model. The positive results could be interpreted that LM-Blocks with LSTMs thanks to their memory allowed for a more suitable approach to learn from protein and drug sequences, since compared to CNNs they managed to capture a better representation of these long sequences.

D.1 Control

The control models where only an embedding of the target protein identities was used as a target representation showed worse performance than our proposed LM-DTA network. This suggests our LM-DTA model where we combined learned representations from sequence data to predict affinity values, profited from inclusion of the target proteins sequence data.

D.2 LM-DTA

In the DTA problem setting using sequence data for both drug and target our LM-DTA model outperformed the baseline on the KIBA dataset. The optimized settings used were not however used with the Davis data, where the reported preliminary results did not beat the baseline. A complete model evaluation on both datasets using cross-validation to estimate the model's performance variance remains to be done.

D.3 Transfer Learning

When the language model blocks had frozen weights the training showed no learning, i.e. no decrease in prediction loss. When the pre-trained layer weights were unfrozen the models performance similarly to the untrained version learned to predict target affinities, however the models' performances were equal or worse than the models' where no TFL was applied. TFL did not show positive effects in our framework. There are a number of possible explanations (see D.4).

D.4 Outlook

Language models have evolved beyond the scope of LSTMs. Models using transformers [*Vaswani et al.* 2017] or their successor BERT [*Devlin et al.* 2018] show new state-of-the-art results on many natural language processing tasks and given our framework could serve as a basis for an extension thereof by exchanging them for the LSTM layers in the language model blocks. The modular framework we introduced is especially suited to easily accommodate such networks within our LM-Blocks to quickly investigate their capacity to solve the DTA problem.

While TFL failed to improve our LM-DTA model's prediction accuracy we did not intensively research its application. Taking a step back, by using only one pre-trained language model block, either of the drug or target, we could determine their individual pre-training's influence towards the LM-DTA model. There also is concern that since the representations of chemical compounds or proteins learned during their respective pre-training are dependent on the training regime (the chosen surrogate loss) they may not be suited for our DTA problem setting.

Neural networks are sometimes called *black boxes*, referring to our lack of understanding of how the weights of a network can be rationalized and extend our knowledge of understanding the mechanisms of the problems they are employed to solve. Work in the domain of explicability of neural networks could help us gain a better understanding of how the representations generated by the LM-Blocks are created.

E Appendix

E.1 Authors note

The work of this research project focused on creating a framework to allow for flexible support of new architectures to explore within the individual LM- and FFL-Blocks and enable TFL to augment the model's performance. The exploration of a variety of such networks instead of optimizing an early iteration and evaluate it lead to positive, but incomplete results by the time the project concluded. We provide the *github repository* with our framework implemented in python using *pytorch*. The Davis and KIBA datasets were downloaded from the *github repository* of the referenced baseline approach. We provide all models discussed here with their pre-trained weights. For an exact reproduction of a specific model, please contact us for the detailed hyperparameter configuration by *mail*.
One of the key steps during this project was the inclusion of a pytorch library named *pack-padded-sequences* which made running our LSTM-based DTA-model with batch-sizes greater than one possible and training feasible in reasonable time.

E.2 Acknowledgements

I would like to thank Prof. G. Schneider and my supervisors Dr. F. Grisoni and M. Moret for giving me the opportunity to explore a field where two of my main fields of interest during studies, maching/deep learning and chemistry meet. The project let me not only work in an interesting interdisciplinary field, but also meet some great minds in the CADD group.

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

LM-DTA: USING LANGUAGE MODELS TO PREDICT DRUG-TARGET AFFINITY

Authored by (in block letters):

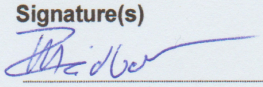
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
HIDBER	FLURIN

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zürich, 09.03.2020	

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.