BMI 707 Project - Predicting Clinical Trial Success using Deep Learning

Aishwarya Chander, Benedikt Geiger, Kezia Irene, Man Qing Liang, Thomas Smits

May 6, 2022

Abstract. Clinical trials for therapeutics are a time and resource-intensive process influenced by several trial, disease and drug properties. We aim to predict clinical trial success rates to accelerate healthcare research and enable the use of new therapeutics that will benefit patients. To this effect, we present a multilayer perception (MLP) model that uses a combination of drugs, drug targets, diseases and trial features to predict the success or failure of a phase III clinical trial. Our MLP model achieved a ROC-AUC of 0.65 and outperformed baseline logistic regression and random forest. Incorporating eligibility criteria and target data improves model performance while excluding drug structure embeddings also improved our model's performance, suggesting feature redundancy. Future work should develop better strategies to address data extraction and embedding challenges, analyze feature importance, and shift efforts to predicting different outcomes.

Introduction

Clinical trials, which are studies evaluating the effect of an intervention in humans, are a crucial step in the development of new drugs. A drug trial is considered successful when it meets its primary endpoint(s), for instance, a clinically and statistically significant improvement in a health outcome compared to a placebo or an existing drug. However, conducting clinical trials is extremely time-consuming and expensive. This challenge raises the question of whether it is possible to predict the outcome of a clinical trial *in silico* with machine learning approaches. A well-performing model can potentially be used to inform trial design, allocate resources more efficiently or even repurpose drugs.

Multiple factors such as the eligibility criteria or chemical properties of the drug and the disease influence the success probability in a complex manner, and the outcome of a clinical trial is by no means obvious. Fu et al.[6] recently demonstrated that deep learning approaches generally perform better in predicting clinical trial outcome when compared to simpler machine learning approaches. The authors also published the first standardized and publicly available Trial Outcome Prediction (TOP) dataset with benchmark performances of different AI approaches. Current approaches are often restricted to particular diseases or do not consider all complex relationships by using standard classification methods [6]. Hierarchical interaction network for clinical-trial-outcome predictions (HINT) [6] tries to overcome these limitations by building a graph neural network (GNN) with drug candidates, target diseases and trial eligibility criteria input.

Our primary research goal was to achieve a similar performance by including drug target information on the protein level and training a multilayer perceptron (MLP) model. Our approach is guided by the belief that current limitations are not caused by the deep learning models found in literature, but by the lack of tidy data and by ignoring all features influencing clinical trial outcomes. We focused our analysis on phase III clinical trials as they are the most expensive and time-consuming, but an expansion to phase I and II clinical trials is straightforward.

Materials

Problem Formulation

The machine learning problem is a binary classification task, and we first expand the problem formulation in [6] to include protein targets. Notice that our work is restricted to small molecule drug discovery clinical trials and does not include trials on biologics (which are large molecules with complex molecular structures), trials on medical devices or other procedures not involving drug testing.

Every clinical test has at least one drug candidate. Let

$$\mathbb{M} = \{m_1, \cdots, m_M\} \tag{1}$$

be the set of the M candidate drugs tested in a particular trial and let $i = 1, \cdots, M$. Each drug candidate m_i can be given in the form of the drug name or its Simplified Molecular Input Line Entry System (SMILES) string and has T_i corresponding protein drug targets

$$\mathbb{T}_i = \{t_1, \cdots, t_{T_i}\}. \tag{2}$$

The set of all drug targets is accordingly given by

$$\mathbb{T} = \bigcup_{i=1}^{M} \mathbb{T}_i. \tag{3}$$

In addition, there is at least one disease corresponding to a trial. Let

$$\mathbb{D} = \{d_1, \cdots, d_D\} \tag{4}$$

be the set of D target diseases, where disease information d_j for $j = 1, \cdots, D$ can be given in the form of a raw name or the International Classification of Diseases (ICD) code. Finally, every clinical trial has a set of I inclusion criteria $\mathbb{E}_I$ and a set of E exclusion criteria $\mathbb{E}_E$. We denote the set of all eligibility criteria by

$$\mathbb{E} = \mathbb{E}_I \cup \mathbb{E}_E = \{c_1, \cdots, c_I\} \cup \{\hat{c}_1, \cdots, \hat{c}_E\}. \tag{5}$$

Combining (1)-(5), clinical trial prediction can be formulated as inferring a prediction function f_θ with parameters θ such that the inferred success probability

$$\hat{y} = f_\theta(\mathbb{M}, \mathbb{T}, \mathbb{D}, \mathbb{E}) \in [0, 1]$$

of a clinical trial estimates the true binary success label $y \in \{0, 1\}$.

Datasets

Trial Outcomes Prediction Benchmark Dataset

The benchmark dataset TOP provides the tables `phase_III_train.csv`, `phase_III_valid.csv` and `phase_III_test.csv` in a GitHub repository [6, 5]. Every clinical trial in these datasets is represented by its National Clinical Trial Number (NCTID), its status (e.g., completed, terminated, active), a reason for the termination if applicable, the trial diseases in free text and their ICD-10 codes, the drugs in free text and their SMILES codes, the free text inclusion and exclusion criteria and a binary variable indicating clinical trial success. 'Success' of a trial is defined as having met the primary endpoints of the trial. The learning (training & validation) and testing phase III data was split on Jan 1, 2014 and the resulting database contains 3094 training, 344 validation and 1146 testing clinical trials. The data was curated from ClinicalTrials.gov and we refer to [6] for further details on the dataset curation process.

DrugBank Dataset

DrugBank is a publicly available dataset that contains information for over 500,000 drugs[13]. The database notably contains drug targets and molecule structure (SMILES string) information, which was leveraged as additional features to the TOP dataset.

AACT Dataset

The AACT dataset [11] contains 54 tables that provide information related to clinical trials from ClinicalTrials.gov. The main key used to link these tables is the `NCT_ID` which is common across HINT and AACT, allowing us to merge data from the two tables and obtain the number of participants per trial.

Methods

The publicly available benchmark dataset TOP is our main data source. However, we noted errors in their result of mapping drug free text descriptions to SMILES strings and decided to process the raw text descriptions on our own using DrugBank, which was also used to extract target data.

Every non-numeric input of our MLP model has to be embedded into Euclidean space $\mathbb{R}^n$ for an appropriately chosen dimension $n \in \mathbb{N}$. In the following, the data processing and embedding for each of the model inputs $\mathbb{D}, \mathbb{E}, \mathbb{M}$ and $\mathbb{T}$ is explained in detail. We also computed historical trial success rates corresponding to each disease group based on our training data and added the number of participants per trial as additional inputs of the model.

Disease Data

Disease information in TOP was extracted from the raw clinical trial in the form of plain text. We then mapped the plain disease text to its Concept Unique Identifier (CUI), a concept code used by the Unified Medical Language System (UMLS) [3]. In the case of no direct disease to CUI match, we first split the disease name into single words and mapped those. Whenever the non-matching disease was a single word or the word splitting approach still resulted in no match, we computed the Damerau-Levenshtein distance [4] to all CUI concepts and identified the closest match based on string similarity.

After manually removing unreasonable matches, we used the pre-trained *cui2vec* embeddings [2] to represent every disease concept as a vector in $\mathbb{R}^{500}$. Unfortunately, some CUIs were not included in the *cui2vec* embedding database.

For every clinical trial, all diseases $d_j \in \mathbb{D}$ for $j = 1, \cdots, D$ were mapped to a single or multiple CUIs and then to an embedding whenever possible. Let $\mathbb{D}_e$ be the set of embedded diseases. The final disease embedding $h_{\mathbb{D}} \in \mathbb{R}^{500}$ was then defined as

$$h_{\mathbb{D}} = \begin{cases} \frac{1}{|\mathbb{D}_e|} \sum_{d \in \mathbb{D}_e} d2cui2vec(d) & \mathbb{D}_e \neq \varnothing, \\ 0 & \text{otherwise.} \end{cases}$$

Taking the average is justified as diseases corresponding to a single clinical trial are often closely related, i.e. they have similar ICD codes and corresponding *cui2vec* embeddings. The issue of an empty embedded diseases set $\mathbb{D}_e$ occurred in 22.6% of all training, 25% of all validation and 21.8% of all testing clinical trials, respectively.

Eligibility Criteria

The eligibility criteria in the TOP dataset were represented as one string of free text for each trial, containing both inclusion and exclusion criteria. Inclusion and exclusion criteria were extracted by a simple regex split. We split each criteria by newline characters, since different trials varied with usage of numbering, bullet points, or other markings. We used a version of BERT specifically trained on a clinical corpus [1]. With this pre-trained clinical BERT model, we encoded each split criterion as one vector in $\mathbb{R}^{768}$. For a faster generation of embeddings, we created a file containing each possible criterion and its mapping.

For each trial, we then concatenated $\mathbb{R}^{768}$ embeddings of all inclusion and exclusion criteria separately. To ensure each embedding has the same dimension, we then decided on a cut-off. By visualised the distribution of criteria lengths in the training set, we decided on a cut-off of 32 sentences, or equivalently a 24576 dimensional vector, either truncating the concatenated vectors or padding 0's at the end, as this represented approximately the 80th percentile of both inclusion and exclusion criteria lengths, truncating criteria in 20% of trials. Padding at the end and therefore potentially decreasing the importance of the end of the vector is justified by the rationale that the first criteria in free text are likely more important than the 32^{nd}. This procedure therefore resulted in two embeddings, $h_{\mathbb{E}_I} \in \mathbb{R}^{24576}$ and $h_{\mathbb{E}_E} \in \mathbb{R}^{24576}$.

Drug Data

Drug information for each trial was available as a list of two or more drugs in unstructured free text format. To obtain drug embeddings for each trial, each free text drug was mapped to a standardized drug name format with the following steps:

1. Converting the drug salt form to generic drug name (e.g., 'metformin hydrochloride' to 'metformin')
2. Matching the string from the free text drug to the DrugBank name
3. For drugs that are unmatched, manually match experimental drug names to its generic drug name (e.g., 'bi 10773' to 'empagliflozin')

A drug list with matched drug names for each trial was then generated. If a drug was unmatched, the drug was marked as 'none'. If the list contained a placebo, it was marked as 'placebo'. 83% of the trials in the training data contained at least one drug that was mapped to a standardized drug name. This proportion was 86% and 78% for the validation and testing data, respectively. A SMILES string list was generated for each matched drug list using DrugBank. Since trials might compare drugs that may have very different structures but have similar effects, we selected the SMILES of the main drug of the trial (e.g., first matched drug in the list) to serve as input for generating the drug embedding instead of using all drugs in a trial and averaging the drug embeddings.

MolR[12], a representation learning model that incorporates chemical reaction information, was used to generate the drug structure embedding for each drug's SMILES. The resulting drug embedding for each trial was a vector in $\mathbb{R}^{1024}$. For trials that did not have a drug SMILES string, the drug embedding was stored as a zero vector.

Target Data

Similar to the drug data, drug targets were extracted from DrugBank by matching the unstructured drug names to the DrugBank drug names. We integrated drug targets because they are biologically relevant macromolecules (i.e, proteins or nucleic acids) that exhibit a pharmacological function by interacting with drug molecules. Often, target molecules are associated with diseases or disease pathways which, when perturbed, can have a pathophysiological effect on the body, which can be used to help inform our model.

For our study, we considered a drug-target as any protein macromolecule that interacts with a drug and is documented in DrugBank. Our data acquisition process is as follows:

1. Amino acid sequences for each target were extracted from DrugBank and a dataframe of DrugBank ID, drug SMILES, target ID and target amino acid sequences was generated.
2. The set $\mathbb{T}_i$ as described in (2) varies for each input of our network, depending on how many targets of known origin and structure are recorded.
3. The dataframe was reduced to a subset of drugs found in our test, train and validation datasets by removing unmapped rows based on the drug SMILES encoding.
4. A total of 4809 unique targets were used as the input to generate our target embeddings.

To generate embeddings for the target sequences, we used an ELMo model [10] which is trained on UniRef50 from the UniProt Knowledgebase, called *SeqVec* [8]. *SeqVec* weights are generated based on amino-acid sequence information independent of factors such as target-binding affinities, drug-biochemical properties or drug interactions.

A dictionary of target IDs and target embeddings was generated to map back to the original dataset, which was then used to link target embeddings to every drug and back to the clinical trials. For each trial in the study, a tensor of drug-target embeddings was generated by taking a mean of all drug-targets associated to the drugs in the study. ‘Placebo’ and ‘None’ values were given a tensor of 0.

Ultimately, we were able to map drug-target embeddings for a total of 35% of all clinical trial studies, specifically described below as (trials with targets/total trials):

- Training data: 1155/3094
- Testing data: 356/1146
- Validation Data: 125/344
- Total: 1636/4584

Metadata

In addition to the above described inputs, we decided to include basic trial data and key historical data as an input branch of our MLP model. This metadata includes the number of inclusion criteria, the number of exclusion criteria and the number of diseases studied in the clinical trial.

Additionally, we computed the historical success rate of the disease studied grouped by ICD chapter on the training data and merged our data with the AACT database to include the number of participants per trial. Because the raw number of participants varies between multiple orders of magnitudes and was skewed, we instead used the median normalized number of participants per ICD chapter studied across training trials. Missing number of participants were imputed with the same median per ICD chapter, which results in the value 0 after median normalization. Combining all gives us a 5 dimensional vector for metadata.

Deep Learning Approach and Evaluation Metrics

The machine learning task is a binary classification with multiple inputs. We therefore decided to implement an MLP model, which we guide by bringing together multiple input branches. For example, the first layers merge the related inclusion and exclusion criteria embeddings and learn a total eligibility criteria representation of the trial, to reduce the dimension. Similarly, we merge drug and target embeddings first and learn a drug-target embedding before connecting this branch to the rest of the model. Following this logic, the MLP learns a clinical trial embedding step by step. Metadata is added last. In the end, fully connected layers reduce the dimensionality of the clinical trial embedding and estimate a clinical trial success probability. Overfitting is avoided by including multiple dropout layers with the same dropout rate.

The last layer uses a sigmoid and all other layers use a ReLU activation. We chose the binary cross entropy loss and the Adam optimizer [9] for training our model. Regarding the hyperparameters, we performed a grid search on the dropout and learning rate on

$$\{0.2, 0.24, 0.28, 0.32, 0.36, 0.4\} \times \{10^{-2}, 5 \cdot 10^{-3}, 10^{-3}, 5 \cdot 10^{-4}, 10^{-4}\},$$

where the learning rates lie on a logarithmic scale [7, p. 434]. The hyperparameters with the lowest mean validation loss after five runs were a dropout rate of 0.32 and a learning rate of 0.01.

We use the classic ROC-AUC and accuracy metrics to assess the classifiers performance. Because the training dataset is mildly imbalanced with $\approx 75\%$ of successful clinical trials, we also include the PR-AUC and F1 scores. The model was run 30 times to evaluate the performance variation. The resulting set of evaluation metrics is the same as in [6].

Base models and ablation studies

Direct comparison of our models performance with [6, Table 2] would not show whether usage of an MLP is more beneficial than a simpler model, since we have created the embeddings in different ways, and added drug target data. Therefore, we created a logistic regression classifier and random forest model using `linear_model.LogisticRegression()` and `ensemble.RandomForestClassifier()` from the `scikit-learn` module in Python 3.7 with default parameters.

To understand the impact of our different inputs, we did ablation studies, omitting one of the features of our MLP, respectively. The performance evaluation of the ablation studies was the same as for our main model.

Results

The final module structure is shown in Figure 1.

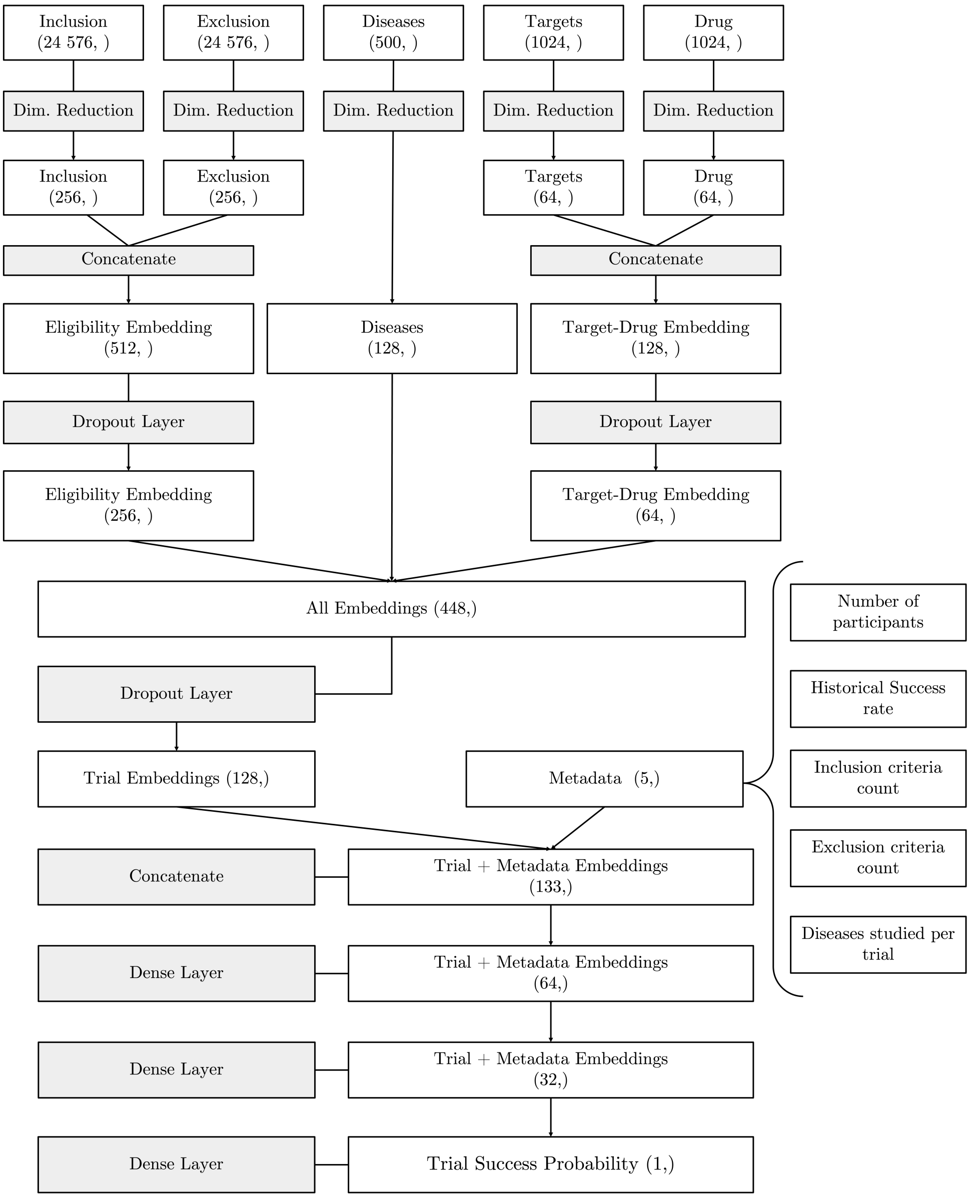

Figure 1: Our MLP model architecture. The dimension of criteria, disease, targets and drug input is first reduced. Then, the model computes a criteria and a drug-target embedding. The dimension of the concatenated trial embedding is then reduced and the metadata is concatenated. The top layers are three dense layers to finally estimate the success probability.

Table 1: Empirical results of various approaches for phase III trial outcome prediction. The metrics are reported using the mean and standard deviation of 30 runs.

Model	ROC-AUC	Accuracy	PR-AUC	F1
Logistic Regression (LR)	0.546 ± 0.0	0.715 ± 0.0	0.870 ± 0.0	0.823 ± 0.0
Random Forest (RF)	0.530 ± 0.006	**0.734** ±0.005	**0.873**±0.002	**0.841** ±0.003
Our model (MLP)	**0.650** ±0.032	0.728 ± 0.028	0.841 ± 0.009	0.831 ± 0.027

Table 2: Empirical results of ablation studies for phase III trial outcome prediction. The metrics are reported using the mean and standard deviation of 30 runs.

Excluded Feature	**ROC-AUC**	**Accuracy**	**PR-AUC**	**F1**
Eligibility Criteria	0.631 ± 0.022	0.710 ± 0.021	0.830 ± 0.011	0.817 ± 0.020
Drug	**0.667** ±0.015	**0.729** ±0.028	**0.842** ±0.007	0.830 ± 0.027
Drug Targets	0.649 ± 0.015	0.719 ± 0.040	0.839 ± 0.008	0.822 ± 0.042
Diseases	0.604 ± 0.0024	**0.729** ±0.023	0.814 ± 0.012	**0.838** ±0.019
Metadata	**0.654** ±0.019	0.721 ± 0.035	0.840 ± 0.008	0.824 ± 0.035

After training the baseline and MLP models, we evaluated the described metrics on the validation dataset. We ran each model 30 times independently to assess the variation, and denote the mean and standard deviation in Table 1. It shows that the MLP has a better ROC-AUC than the simple models, but the accuracy, PR-AUC and F1 was better for the Random Forest.

The resulting metrics of the ablation studies are denoted in Table 2. It shows that the removal of eligibility criteria decreases model performance on all metrics. Values above the MLP model are denoted in bold.

Discussion

We have observed that our model outperforms base logistic regression and random forest models, with 0.65 ± 0.032 performance on ROC-AUC metrics. In terms of accuracy, PR-AUC, and F-1 score, our model performs almost equally with the random forest model with the accuracy of 0.728 ± 0.028, PR-AUC of 0.841 ± 0.009, and F-1 score of 0.831 ± 0.027.

Furthermore, our model did not outperform the HINT model, which had a ROC-AUC of 0.723 ± 0.006. However, it is not possible to make a direct comparison between the two models since our dataset contained modified input data (e.g., remapped SMILES strings, addition of target data) and used different approaches to generate embeddings. Future work could focus on a HINT-based model performance when incorporating additional input data or using different representation learning methods.

The ablation studies showed that removal of the eligibility criteria decreases the model performance, a notion also observed by [6]. Furthermore, we observe that removal of the drug targets also decreases model performance, indicating that drug targets are a useful addition to the model. More interestingly, we observed that removal of metadata and drug information does not decrease model performance. For drug embeddings, this may be due to latent encoding in the drug targets embeddings. This may indicate that inclusion of drug targets only is better than inclusion of drugs only. We deem this reasonable, because the protein targets of a drug include more information than only the molecular structure of a drug. We can see that excluding metadata in the final model did not impact the performance, which is surprising since we believed the historical success rate and number of participants would hugely impact model performance.

This model and idea of predicting success rate of a clinical trial could highly impact the way we view clinical trials. It allows us to decide in advance whether a clinical trial would be worth the cost and effort. More interestingly, by analysing the variables that the outcome is most dependent on, specifically in a counterfactual way, we could pinpoint why a trial may have a lower probability for success. If this is due to the drugs used, then there may be little opportunity to increase the success rate, however, if this may be pinpointed to the eligibility criteria, then we could alter those and increase the success of a trial.

A main limitation of this study is the poor definition of 'success', which is defined as a binary variable (e.g., success or failure), which provides little actionable insights for conducting clinical trials. Having more informative labels (e.g., which indicates which endpoint was a success or failure) could further inform the design of clinical trials. Notably, trial failure as defined by high toxicity, which may be informed by drug properties, would be predicted differently than trial failure due to lack of statistical significance, which may be informed by the small number of participants in a trial.

We applied a number of strategies to handle missing data in the TOP dataset. We used strategies like imputation (i.e., number of participants data), or extracted from external datasets (i.e., missing SMILES values, drug-target information). In certain cases, missing values or embeddings were assigned 'none' or 0 values. For target data in particular, we had approximately 65% missing data. Given the limited time-frame, we were unable to resolve this issue but have considered several options. One approach could be to use a graph based system to generate highly specific drug-target interactions weighted by target binding affinities. Another strategy could be to use all possible biological targets as defined in the UniProt Database instead of DrugBank, which accounts for a limited subset of experimentally and pharmacologically relevant targets only.

A technical limitation we encountered involves our own set-up. Due to limited resources (i.e., absence of a shared cluster space, too little computing power on our local devices), we ended up using the Google Colab Pro computing resources. However, this made it impossible to connect with Git and Version Control. Furthermore, even with Google Colab Pro resources, we were limited in the number of times we could iterate our model and embeddings, with certain scripts running for more than 36 hours.

The future work we considered includes generating reliable embeddings, incorporating trial data from phase I and phase II, using counterfactual models to heighten model explainablity, incorporating molecular data such as drug-drug interactions, biochemical drug properties (solubility, structural data), and applying different strategies to generate model embeddings. We are also considering the modification of the input and output parameters of our model into a system that may take in the diseases, drugs and drug targets data, giving us the minimal number of participants and eligibility criteria, or target success metrics

Group Member Contribution

- Project ideation: Man Qing Liang
- Project proposal: Man Qing Liang, Aishwarya Chander, Kezia Irene
- Disease embeddings: Benedikt Geiger
- Eligibility criteria embeddings: Thomas Smits
- Drug data matching and embeddings: Man Qing Liang
- Target data curation and embeddings: Aishwarya Chander, Thomas Smits
- Metadata: Kezia Irene, Man Qing Liang, Benedikt Geiger, Thomas Smits
- MLP model & Ablation models : Benedikt Geiger
- LR and RF models: Thomas Smits
- Poster: Aishwarya Chander

References

[1] E. Alsentzer, J. Murphy, W. Boag, W. Weng, D. Jin, T. Naumann, and M. McDermott. Publicly available clinical BERT embeddings. *arXiv preprint arXiv:1904.03323*, 2019.

[2] A. Beam, B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, and I. Kohane. Clinical concept embeddings learned from massive sources of multimodal medical data. In *Pacific Symposium on Biocomputing 2020*, pages 295–306. World Scientific, 2019.

[3] O. Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

[4] F. Damerau. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7(3):171–176, 1964.

[5] T. Fu, K. Huang, C. Xiao, L. Glass, and J. Sun. TOP benchmark dataset. `https://github.com/futianfan/clinical-trial-outcome-prediction/tree/main/data`.

[6] T. Fu, K. Huang, C. Xiao, L. Glass, and J. Sun. HINT: Hierarchical interaction network for clinical-trial-outcome predictions. *Patterns*, 3(4):100445, 2022.

[7] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[8] M. Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, and B. Rost. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC bioinformatics*, 20(1):1–17, 2019.

[9] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] E. Matthew, N. Mark, I. Mohit, G. Matt, C. Christopher, L. Kenton, and Z. Luke. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018.

[11] A. Tasneem, L. Aberle, H. Ananth, S. Chakraborty, K. Chiswell, B. McCourt, and R. Pietrobon. The database for aggregate analysis of ClinicalTrials. gov (AACT) and subsequent regrouping by clinical specialty. *PloS one*, 7(3):e33677, 2012.

[12] H. Wang, W. Li, X. Jin, K. Cho, H. Ji, J. Han, and M. Burke. Chemical-Reaction-Aware Molecule Representation Learning. *arXiv preprint arXiv:2109.09888*, 2021.

[13] D. Wishart, C. Knox, A. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, and J. Woolsey. DrugBank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic acids research*, 34(suppl_1):D668–D672, 2006.