**3DGT-DDI: 3D graph and text based neural network for drug–drug interaction prediction**

Drug-drug interactions (DDIs) occur when drugs are combined. Finding possible DDI allows the investigation of the mechanism underlying drug interactions or adverse responses in order to prevent the negative effects. Even though numerous artificial intelligence techniques are used to mine and forecast probable DDI, these techniques neglect the 3D structure data of pharmacological molecules and do not completely account for the role that molecular substructure plays in DDI. A novel artificial intelligence model called 3DGT-DDI is suggested to address the drawbacks of the earlier approaches. SCIBERT and a 3D graph constitute this model. For the purpose of predicting and maybe exploring DDI interactions, it can combine text features, 3D structures and position data of drug molecular entities. The model takes text and entity information into account. Instead of using conventional two-dimensional information for structural information, 3D structural information is utilized. CNN is used for effective fusion. For text information, the information in multiple hidden layers is considered.

A novel DDI prediction architecture called 3DGT-DDI is proposed in this paper. In order to obtain the three-dimensional structure, a 3D graph model is used. The text information in DDI is obtained using the feature extraction model of text scription based on BERT. The drug name's position embedding vector is extracted to improve the model's capacity to extract context-relevant features. A neural network model for DDI prediction is built using the Schnet as its framework. Input to the 3D GNN is obtained by creating three-dimensional conformations of medicinal compounds. The prediction of the DDI relationship is obtained after feature fusion. The Schnet framework constructs features with continuous distance. The atom distance is entered into a continuous mapping using the continuous filter operation. A multilayer one-dimensional convolutional neural network is utilized, and the number of node characteristic output channels of the Schnet network is modified. The two drugs' structural features were extracted. To extract the key information of the DDI relationship from the text description, the multi-head attention technique is introduced. To increase training effectiveness and more effectively extract text-related characteristics, the pretraining model SCIBERT is incorporated into the DDI model. The fused multi-hidden layer features are convoluted to produce high-dimensional features, which are then output through the full connection layer using a multilayer convolution neural network. An adaptive multiple fusion layers for hidden layer features is utilized to increase the feature extraction range and enhance feature extraction accuracy.

An attention-based intelligent framework called 3DGT-DDI is proposed in this paper. It consists of 3D graph models and pre-trained text models for the prediction and in-depth analysis of DDI interactions. On the DDI2013 dataset, the model outperforms other SOTA baselines. Experiments have demonstrated the superiority and interpretability of 3DGT-DDI. The approach introduced 3D information and combined it with textual drug description information, in contrast to prior methods that utilized molecular 2D information. The outcomes of the experiment demonstrated that the model's performance is enhanced by the efficient combination. Better interpretability was demonstrated when 3D information and text were combined. The performance of the model is greatly enhanced by the hidden layer feature fusion technique, which enhances the ability to extract DDI types from the text training model. More significantly, the suggested strategy incorporated the medicinal entity's 3D structural properties with location features. The training of the model is clearer and more usable when the structure of text and drug molecules are seen. 3DGT-DDI has acquired knowledge of the 3D structures of drug compounds and integrated it with text-based training of these molecules. It performs better and enhances the impact of DDI prediction. Medical researchers can more easily grasp probable DDI and the process underlying combination drugs or adverse effects by visualizing DDI.

**Learning to Branch for Multi-Task Learning**

Sharing some layers of a network, training several tasks simultaneously in a deep network result in decreased inference latency and superior performance to the single-task counterpart. However, excessive network sharing could inadvertently enforce over-generalization and result in unfavorable information transfer between tasks. Prior research for ad hoc branching architectures relied on human intuition or already computed task relatedness ratings. They frequently demand significant effort for the process of trial and error and deliver subpar outcomes. The automated multi-task learning algorithm presented in this study learns where to share or branch within a network, creating a useful network topology that is directly optimized for various task-related goals. A tree-structured network design space that can automatically figure out which branches to add to a network to reduce overall multi-task loss is proposed. Sampling from a categorical latent variable created by the gumbel-softmax distribution is used to carry out the branching procedure. Neither prior knowledge of the relationships between tasks nor human intuition for which layers record task-specific variables and should be divided are necessary for this data-driven network structure searching approach.

The proposed approach aims to learn a tree-structured network architecture and the weight values of the network that minimize the total loss over all tasks given a collection of tasks. Branching activities at specific layers create the network's tree structure. A branching layer may contain as many child (next) layers within the bounds of the given computational budget. In order to calculate the overall loss value during training, a network configuration is sampled from the distribution of the design space. The network's weight matrices and design space distribution are then backwards updated using the corresponding gradients. The final network configuration is sampled using the converged design space distribution after iterating through the process until the overall validation loss converges. A Directed Acyclic Graph (DAG) represents the topological space that has nodes for computational operations and edges for data flows. For each block, numerous parent nodes and child nodes are built, allowing a child node to choose a path from among all the paths connecting it to all its parent nodes. As a result, the tree structure is defined by the chosen connectivities through this sampling (branching) process. If each block only has one parent node and one child node, the suggested topological space degenerates into a standard single-path (convolutional) neural network. The backward pass then modifies the sampling distribution to make it more likely to produce network configurations toward the direction of minimizing the overall loss after the parameter of the sampling distribution is changed using the chain rule with respect to the final loss. The network's weight matrices and topology distribution are jointly optimized over the loss across all tasks during the training phase. When the validation loss converges, the final network configuration is chosen for each block in the network using the same category distribution but without noise. To achieve the desired performance, the final network architecture is retrained from scratch. The same process has also been proven successful in earlier literature, where such weight sharing network search schema shows good correlation between the intermediate network performance during the search phase and the final performance attained by completely retraining the network.

A novel end-to-end trainable algorithm is described in this paper that can automatically construct a hard parameter sharing multi-task network, sharing and dividing network branches based on the update gradients backpropagated from the total losses across all tasks. The suggested approach directly optimizes over the end outputs rather than using pre-calculated task relatedness scores, saving time-consuming computation and resulting in efficient network topologies. It is proposed to use a well-constructed topological space to enable direct optimization of the network's weights and branching distributions via gumbel-softmax sampling. On controlled synthetic data, real-world data, and a large Taskonomy dataset, the proposed method is validated