Εθνικό Μετσόβιο Πολυτεχνείο

Εργαστήριο Βιοϊατρικών Συστημάτων

ΤΟΜΕΑΣ ΜΗΧΑΝΟΛΟΓΙΚΩΝ ΚΑΤΑΣΚΕΥΩΝ ΚΑΙ ΑΥΤΟΜΑΤΟΣ ΕΛΕΓΧΟΣ

Διπλωματική Εργασία

Analysis of important signaling sub-networks for drug mechanism of action identification

Φοιτήτρια: Κολέρη Χριστίνα

Επιβλέπων καθηγητής: Αλεξόπουλος Λεωνίδας

Αναπληρωτής Καθηγητής ΕΜΠ

ΑΘΗΝΑ, ΦΕΒΡΟΥΑΡΙΟΣ 2023

Acknowledgments

This project and the work that came with it wouldn't have been realized without the support and guidance of many great people.

More than an academic project

First of all I would like to thank professor Leonidas Alexopoulos for giving me the opportunity and the privilege to be part of his team and participate in one of the most up to date labs in NTUA. His advisory and guidance meant a lot for me and I highly appreciate it.

Moreover, I am more than grateful to meet and work with Chris Fotis. He is not only a mentor for me but also a friend. Without his leadership and help this project would have been a much harder task.

Finally, I am thankful to every single member of the lab that made this whole time a joyful period for me. They became more friends than lab mates to me.

The ones that were always there

Last but not least, I would like to thank my family and friends for being always there and providing me with love and support all these years. I wouldn't have been the person I am without them. I am grateful to them and I wish them all the best.

Περίληψη

Η βαθιά μάθηση έχει αναδειχθεί ως ένα ισχυρό εργαλείο στην εποχή της πληροφορίας, αντιμετωπίζοντας πολύπλοκες προκλήσεις σε διάφορους ερευνητικούς τομείς. Η παρούσα εργασία επικεντρώνεται στην εφαρμογή τεχνικών βαθιάς μάθησης για την αποκωδικοποίηση του μηχανισμού δράσης επιλεγμένων φαρμάκων χρησιμοποιώντας βιολογικά σηματοδοτικά δίκτυα. Στα πλαίσια αυτής αναπτύχθηκε ένας αλγόριθμος που εντοπίζει σημαντικά υπο-δίκτυα μέσα στα δίκτυα σήμανσης που προκαλούνται από τα συστατικά των φαρμάκων. Με τη χρήση ενός μοντέλου βαθιάς μάθησης για γράφους, που ονομάζεται deepSNEM, μετασχηματίσαμε τα δίκτυα σήμανσης που προκαλούνται από τα συστατικά των φαρμάκων σε υψηλής διάστασης αναπαραστάσεις και στην συνέχεια εντοπίσαμε διακριτές ομάδες αυτών που εμπλουτίζονται σημαντικά με συγκεκριμένους αναστολείς (mTOR, τοποϊσομεράση, HDAC και αναστολείς σύνθεσης πρωτεΐνης). Επιπλέον, η μεθοδολογία που αναπτύχθηκε περιλαμβάνει μια ανάλυση σημασίας υπο-δικτύων, αποκαλύπτοντας κρίσιμους κόμβους και υπο-δίκτυα που σχετίζονται άμεσα με τους πιο κυρίαρχους μηχανισμούς δράσης μέσα στην κάθε ομάδα. Αυτή η ανάλυση παρέχει ένα πλαίσιο για την κατανόηση της σημασίας των μεμονωμένων πρωτεϊνών στο μονοπάτι. Για να αποδείξουμε την πρακτική χρησιμότητα της προσέγγισής αυτής, το μοντέλο deepSNEM και η ανάλυση σημασίας υπο-δικτύων εφαρμόστηκε χρησιμοποιώντας προφίλ έκφρασης των γονιδίων των συστατικών των φαρμάκων από διάφορες πειραματικές πλατφόρμες, συμπεριλαμβανομένων των MicroArrays και της ακολουθιακής ανάλυσης RNA. Τα αποτελέσματα δείχνουν τη δυνατότητα δημιουργίας ακριβών υποθέσεων σχετικά με τους μηχανισμούς δράσης αυτών των συστατικών. Συνολικά, η έρευνα αυτή προσφέρει μια προηγμένη μεθοδολογία που συνδυάζει τεχνικές βαθιάς μάθησης με δεδομένα σηματοδοτικών δικτύων. Μέσω της ανάλυσης σημαντικών υπο-δικτύων, η μέθοδος που αναλύεται συντελεί στην αναγνώριση των μηχανισμών δράσης των φαρμάκων, παρέχοντας σημαντικές ενδείξεις για τις υποκείμενες διεργασίες που επηρεάζουν την επίδραση των συστατικών.

Abstract

Deep learning has emerged as a powerful tool in the era of Big Data, addressing complex challenges across various research domains. This thesis focuses on the application of deep learning techniques to unravel the mechanisms of action for selected drugs using biological signaling networks. We present a comprehensive pipeline that identifies significant sub-networks within compound-induced signaling networks. By employing an unsupervised graph deep learning pipeline called deepSNEM, we transform compound-induced signaling networks into high-dimensional representations. Utilizing the deepSNEM embeddings and clustering with the k-means algorithm, distinct clusters enriched for specific inhibitors (mTOR, topoisomerase, HDAC, and protein synthesis inhibitors) are identified. Additionally, our pipeline incorporates a subgraph importance analysis, revealing critical nodes and subgraphs directly associated with the most prevalent mechanisms of action within each cluster. This analysis provides an interpretable framework for understanding the significance of individual proteins in the pathway. To demonstrate the practical utility of our approach, we apply deepSNEM and the subgraph importance pipeline to compounds' gene expression profiles from various experimental platforms. The results indicate that accurate hypotheses can be generated regarding the mechanisms of action for these compounds. In summary, our research offers an advanced methodology that combines deep learning techniques with signaling pathway data. By analyzing important signaling sub-networks, our pipeline contributes to the identification of drug mechanisms of action, providing valuable insights into the underlying processes driving compound effects.

Table of Contents

1 Introduction

1.1 Biology

1.1.1 Systems Biology

Systems biology is an integrative discipline connecting the molecular components within a single biological scale and also among different scales (e.g. cells, tissues and organ systems) to physiological functions and organismal phenotypes through quantitative reasoning, computational models and high-throughput experimental technologies. The modeling and the revelation of the possible links between their components come with the potential of new discoveries regarding drugs, treatments and diseases' biological profiles.

Systems biology uses a wide range of quantitative experimental and computational methodologies to decode information flow from genes, proteins and other subcellular components of signaling, regulatory and functional pathways to control cell, tissue, organ and organismal level functions [1].

1.1.2 DNA

DNA, or deoxyribonucleic acid is the central information storage system of most organisms including a portion of viruses. The complete set of information in an organism's DNA is called its genome, and it carries the information for all the proteins the organism will ever synthesize. The information included in it is stored as a code consisting of four chemical bases. These bases are adenine (A), guanine (G), cytosine (C) and thymine (T). Human DNA is made up of approximately three billion bases. The way that these bases are ordered and their sequence determines the information which are at the cell's disposal for the survival of the organism. Each base A, C, T, or G can be considered as a letter in a four-letter alphabet that spells out biological messages in the chemical structure of the DNA. DNA's bases pair in a specified way. Adenine pairs with thymine and cytosine pairs with guanine forming some units that are known as the base pairs. All the bases are also attached to a phosphate molecule and a sugar molecule. These three compounds together form the nucleotide. The nucleotides form two long strands and these in their turn coil around each other held by hydrogen bonds and form a double helix. The base pairs are the linking parts of the helix's vertical sides. These sides are composed by the sugar and the phosphate molecules that were previously mentioned [2].

The DNA structure is presented below.

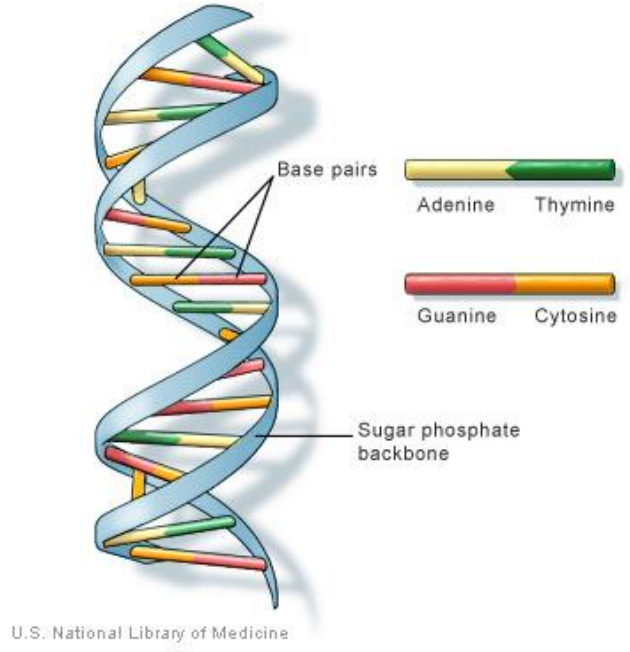


***Figure 1.1**: DNA structure, U.S. National Library of Medicine*

1.1.3 RNA

RNA, or ribonucleic acid, is one of the three major biological macromolecules that are essential for all known forms of life, with the remaining two being DNA and proteins. RNA is a nucleic acid similar in terms of structure and properties to DNA, but it consists of only one strand of bases. The nucleotides in RNA are ribonucleotides which means they contain the sugar ribose (hence the name ribonucleic acid) rather than deoxyribose. Moreover, its bases are the same with DNA's, except from one. Instead of the base thymine (T), RNA has another base called uracil (U). So, RNA's bases are adenine (A), guanine (G), cytosine (C) and uracil (U). There are many types of RNA, but the three most significant and well-known are the messenger RNA (mRNA), the transfer RNA (tRNA) and the ribosomal RNA (rRNA). One of the molecules that are useful for the explanation of gene expression process is the messenger RNA. The mRNA is a molecule in cells that carries codes from the DNA that is present in the cell's nucleus to the sites where protein synthesis takes place, in the cytoplasm (mainly ribosomes). The other molecule that is present in the processes is the transfer RNA (tRNA). This is a small molecule in cells which carries amino acids to ribosomes, where they are linked to form proteins [2].

With DNA and RNA defined, one can proceed to the explanation of the processes which compose the gene expression.

1.1.4 Gene Expression

In general, with the term gene expression (GEx) we refer to the natural process in which the aforementioned information that is stored inside the DNA is converted into functional

products like proteins or different types of RNA. The process of gene expression is being deployed by two basic operations, transcription and translation.

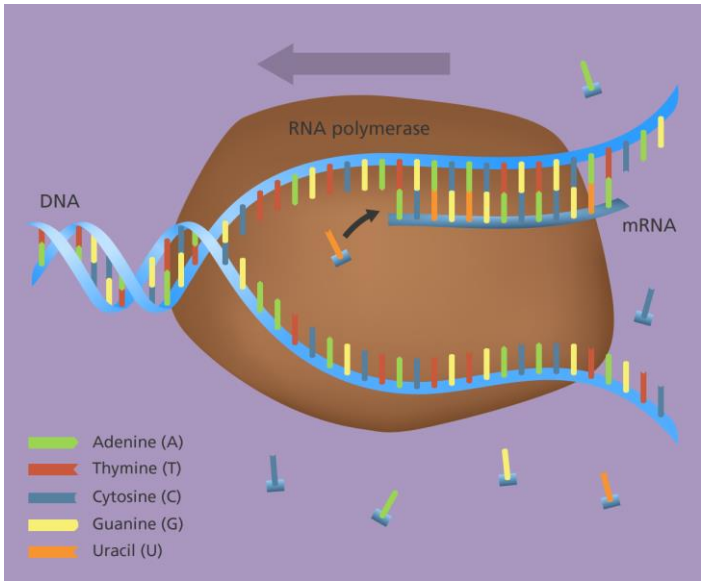


***Figure 1.2**: The process of transcription (Image Credit: Genome Research Limited)*

The first step a cell takes in reading out a needed part of its genetic instructions is to copy a particular portion of its DNA nucleotide sequence -a gene- into an RNA nucleotide sequence. The information in RNA, although copied into another chemical form, is still written in essentially the same language as it is in DNA – the language of a nucleotide sequence. Hence the name transcription. RNA in the cell is completely created by DNA transcription, which begins by opening and unwinding of a small portion of the DNA double helix to expose the bases on each DNA strand. One of the two strands of the DNA double helix then acts as a template for the synthesis of an RNA molecule. The enzymes that perform transcription are called RNA polymerases and the transcript is called messenger RNA (mRNA).

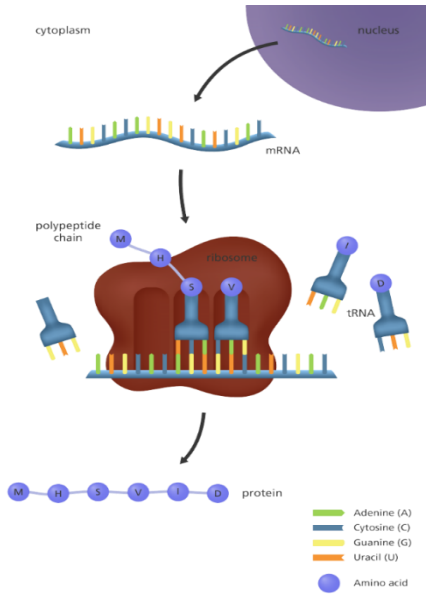


***Figure 1.2:** The process of transcription (Image Credit: Genome Research Limited)*

The second process, translation, occurs when the aforementioned messenger RNA has carried the transcribed the needed information from the DNA to the cells' ribosomes, in which proteins are being created. The translation of mRNA into protein depends on adaptor molecules that can recognize and bind both to the codon (three letters) and, at another site on their surface, to the amino acid. These adaptors consist of a set of small RNA molecules known as transfer RNAs (tRNAs), each about 80 nucleotides in length. Once the tRNA is bound, it releases its amino acid and the adjacent amino acids all join together into a long chain called a polypeptide, continuing the process above until the protein is formed.[2]

1.1.5 Proteins

Proteins are by far one of the most chemically complex and functionally sophisticated molecules known so far. A protein molecule is made from a long chain of amino acids (20 different types of amino acids) each linked to its neighbor through a covalent peptide bond, thus the alternate name polypeptides. Each type of protein has a unique sequence of amino acids.

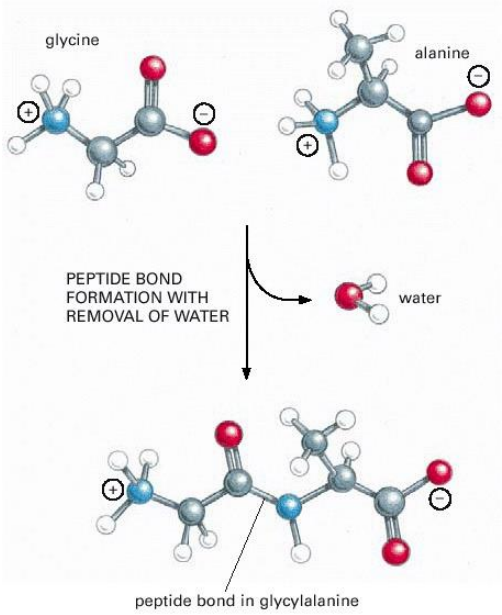


Figure 1.3: *Peptide bond [2]*

The biological properties of proteins depend entirely on their physical interaction with other molecules. For example, antibodies, Y shaped proteins that are produced by the immune system, bind to viruses or bacteria, actin molecules bind to each other to assemble into actin filaments, and so on. The substance that is bound by the protein is called a ligand. The region of a protein that associates with a ligand, known as the ligand's binding site, usually consists of a cavity in the protein surface formed by a particular arrangement of amino acids [2].

1.1.6 Transcription Factors

Transcription factors or sequence-specific DNA-binding factors are proteins that control the rate of transcription of the genetic information included in the DNA to messenger RNA

(mRNA) [3]. In fact, transcription factors are proteins that help turn specific genes “on” or “off” by binding to nearby DNA. The transcription factors can be either activators, in case they boost a gene’s transcription, or repressors, in case they decrease a gene’s transcription. Moreover, groups of transcription factor binding sites that are called enhancers and silencers can turn a gene on or off in specific parts of the body. The most significant property of the transcription factors is that they allow cells to perform logic operations and in fact combine different sources of information to determine whether a gene will be expressed or not. As mentioned before, at the Transcription process analysis, RNA polymerase has to attach to the DNA of a gene in order to make an RNA molecule. While in some organisms this process doesn’t require anything else, in humans and other eukaryotes, an extra step exists. In these cases, the transcription can happen only with the help of some proteins called basal (general) transcription factors. These proteins are part of the cell’s core transcription tools and are crucial for the transcription of any gene.

1.1.7 Signaling Networks

One of the most important issues in biology is the study of the various interactions between cellular molecules which determine their biological properties. Such interaction networks are usually classified according to the type of the molecules involved, these usually being genes or proteins. Networks that involve cell signaling, i.e. the response of a cell to internal and external stimuli (chemical or even of mechanical and electrical nature) and coordinate the regulation of its activity are called Signaling Networks.

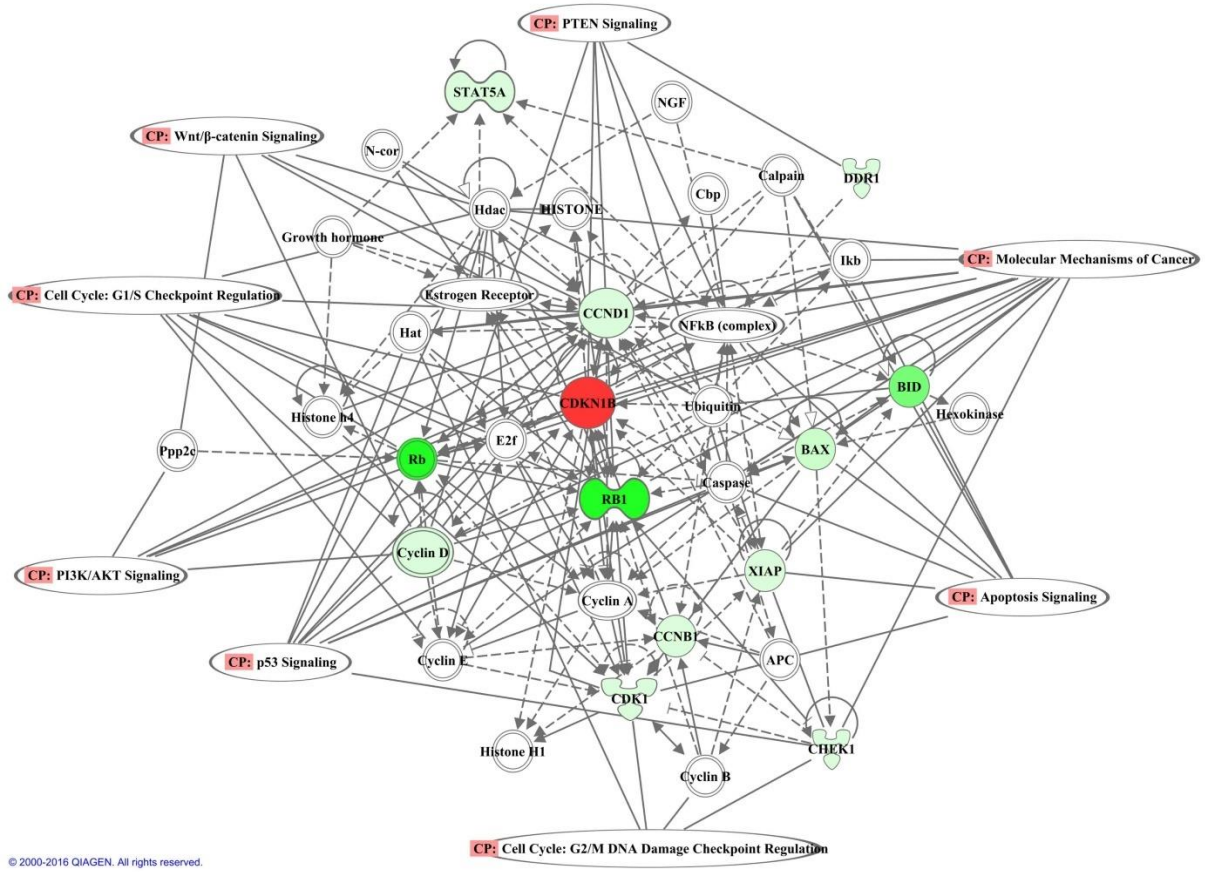


Figure 1.4: *An example signaling network*

Individual pathways transmit signals along linear tracts resulting in regulation of discrete cell functions. This type of information transfer is an important part of the cellular repertoire of regulatory mechanisms. Inside the cell, there exists a particular family of proteins called receptors that bind to signaling molecules and initiate an initial response. Such signaling molecules include Hormones which are the major signaling molecules of the endocrine system, Neurotransmitters, which are signaling molecules of the nervous system and Cytokines which are signaling molecules of the immune system. Essentially, carrying out complex biological processes requires the cooperation of several cells along with their specific functions, by the process of cell signaling [4][5][6].

A possible malfunction of a signaling pathway may result to disease or even cancer. Signaling pathways provide a more structured form of genes. In contrast to gene expression that takes account of every gene separately, signaling pathways group together genes that take part in specific functions in the cell, thus providing a wider and more causal perspective to the gene expression topic. This method can be also considered as a higher biological level method, since it uses prior knowledge regarding the signaling pathways of the cell to derive the ones that are activated or deactivated under certain circumstances.

1.1.8 Protein Networks

Protein-protein interactions (PPIs) [7] are crucial to the majority of a cell's processes. According to this, in order to gain knowledge and understanding of the cell's processes, the study of PPIs is really important. Moreover, in a reverse way of thinking, one can use and analyze the activity in the PPI level to generate insights into the underlying processes of the cells. When the PPIs are linked, the protein-protein interaction networks (PPIN) are built. These are directed networks that visualize the way that the proteins interact with each other, which proteins are linked and as a result which is the effect of the proteins on the network.

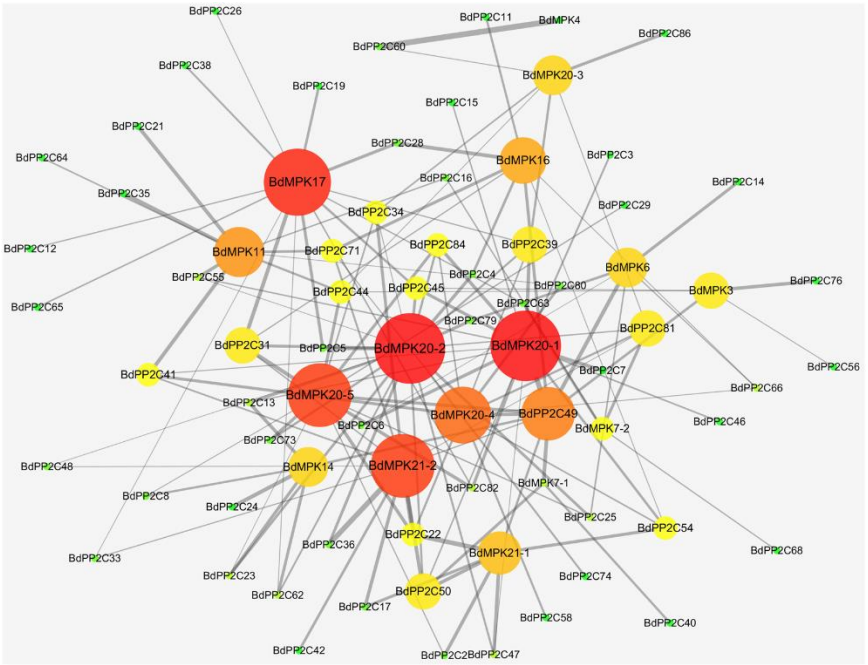


Figure 1.5: *An example Protein-Protein network (PPI).*

Knowing the above mentioned PPIN and the gene expression or the transcription factor activity of an experiment, the protein network of a specific cell can be constructed, by combining these two characteristics. This network describes sufficiently the activity that takes place in the cell and is unique for every experiment, provided that the gene expression varies each time that an experiment is carried out under different circumstances. Thus, PPIN are a higher level biological attribute that is defined using a known PPI as a prior knowledge factor and the gene expression or the transcription factor activity of an experiment. This attribute includes some topological knowledge, based on the PPI, and also the biological experimental results. Their combination can lead to characteristic protein networks for the cells that are subjected to research each time.

1.2 Artificial Intelligence – Machine Learning – Deep Learning

Artificial Intelligence (AI) is an area of computer science that simulates the structures and operating principles of the human brain. Machine learning (ML) belongs to the area of AI and endeavors to develop models from exposure to training data. Deep Learning (DL) is another subset of AI, where models represent geometric transformations over many different layers.

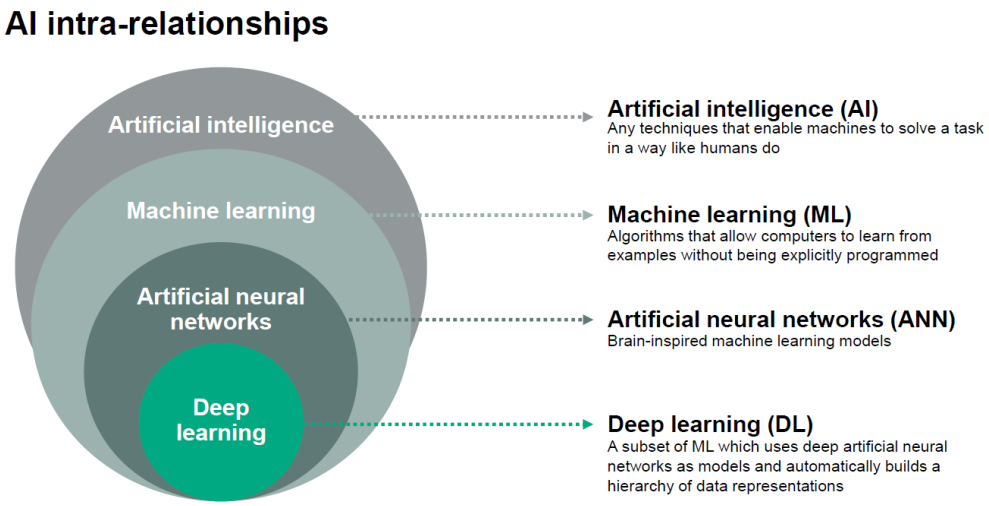


Figure 1.6: *AI intra-relationships*

1.2.1 Neural Networks

Deep Learning (DL) is a subfield within ML based on the use of different artificial neural network (ANN) algorithms that, through a sequence of layers with non-linear processing units, for modeling high-level abstractions contained in data. In DL architectures, each layer trains on a distinct set of features based on the previous layer's output. The further you advance into the neural net, the more complex the features your nodes can recognize, since they aggregate and recombine features from the previous layer. For example, a DL algorithm predicting whether an image contains a face or not extracts features such as the first layer perceiving edges, the second layer perceiving shapes such as noses and eyes, and the final layer perceiving face shapes or more complex structures. This is known as feature hierarchy, and it is a hierarchy of increasing complexity and abstraction.

Inspired by biological neural networks, the basic building block of ANNs are neurons which are essentially a form of mathematical entity that holds a real number. Each neuron accepts the value of previous connected neurons as input, and maps into a non-linear function, also called an activation function:

$$x_{new} = s \cdot (w \cdot x_{prev} + b)$$

where w, b are the trainable parameters of the node called weight and bias and s is the non-linear activation function. Neurons, in their turn form layers a series of whom describes the basic architecture of a Feedforward Neural Network. In other words each neuron can be looked at as a processing unit and an interlinking of such neurons leads to massive computing power that can solve complex operations.

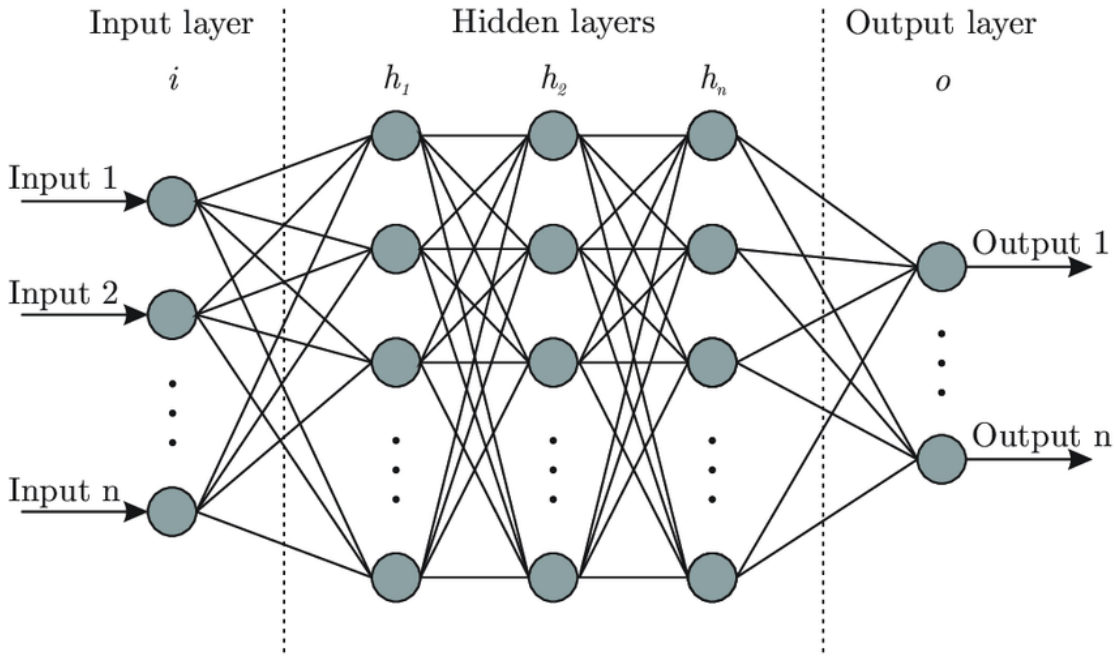


Figure 1.7: *An example Artificial Neural Network (ANN).*

A typical ANN consist of the input layer, the hidden layers and the output layer. Input, hidden and output layers can be linked in many fashions, which will have an impact on the performance of the network for different tasks. The final result of any input propagated through the network will depend on the pattern of connections in every layer, the functions controlling neuronal activation and the weights associated to each link between neurons. The architecture (pattern of connections) and the activation function play a critical role in the final result obtainable by the network; essentially they define the global model of the network, that has to be in line with the nature of the task to be performed. Neural networks are generally trained by optimizing the selected cost functions. These cost functions accept the networks output as well as the ground truth labels as input and are trained using optimization algorithms that revolve around Stochastic Gradient Descend (SGD).[8]

When the number of hidden layers is more than two or three, for example hundreds, then that is known as deep neural network (DNN).

1.2.2 Graph Deep Learning Models

There have been many studies for the development of deep learning models for graph data in a variety of fields. These models are usually neural networks that aim to learn new task-specific node and graph representations by using the graph's connectivity [9].

1.2.2.1 Graph Convolution Neural Networks

Originally introduced by Kipf and Welling [10] Graph Convolutional Networks (GCNs) have become the starting point when working with graphs using neural networks. The main idea of the graph convolutional model can be described as the utilization of a message passing algorithm to learn neighborhood-level representations of the input graph. Recently, the successful transformer architecture for natural language processing (NLP) problems has been modified and applied on graph data [8,11].

In their simplest form, GCNs operate on undirected graphs $G = (V, E)$ where G is the graph and (V, E) describe the set of vertices and edges of the graph respectively.

A simple propagation rule would be

$$f\left(H^{(l)}, A\right) = \sigma(AH^{(l)}W^{(l)})$$

when $W^{(l)}$ is the weight matrix on layer i, A is the adjacency matrix and $H^{(l)}$ is the feature matrix of the graph nodes in layer i. Therefore, this process imitates the way typical Convolutional Neural Networks work, in the sense that a filter propagates around the graph, reading both the information of each node as well as aggregating the features of their neighborhood.

1.2.2.2.Graph Transformer Networks

Graph transformers utilize an attention mechanism for each node that is a function of the neighborhood's connectivity, rather than a message passing algorithm.

Created as an alternative to complex recurrent or convolutional neural networks, transformers (Vaswani et. al) follow a much simpler architecture but exhibit more powerful representations. The main structure makes use of the typical encoder - decoder architecture, bound together with an attention mechanism. It is important to explain the vanilla transformer to the reader, since the basic model used in this thesis belong to this family of neural network architectures. We will be focusing on the encoder part of the transformer and especially the attention mechanism as the modified variation we employ is inspired by them. The encoder is composed of a stack of N identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise, fully connected feed-forward network. Each sublayer is also connected using residual connections [12].

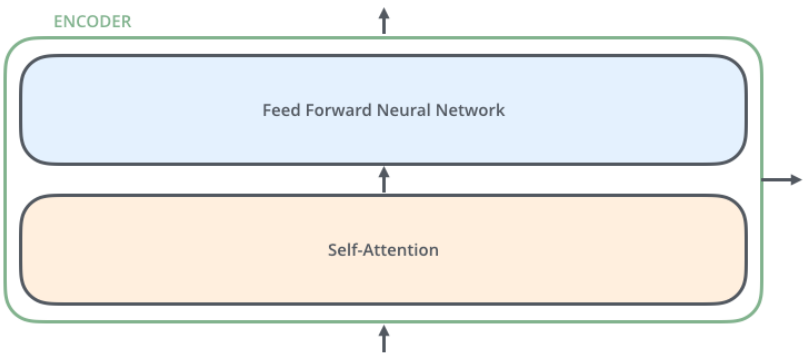


Figure 1.8: *Self-Attention Feed Forward Architecture*

In general, attention functions can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key.

Scaled Dot-Product Attention

The attention mechanism between the queries, keys and values in the original paper is called scaled dot-product attention. In practice, for each graph we have the three Q, K, V matrices that correspond to the total query, key and value vectors of the graph's node features. The attention is calculated as:

$$Attention(Q,K,V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{d_K}}\right)$$

where $\frac{1}{\sqrt{d_K}}$ is a scaling factor. Thus, nodes with features that are similar (due to their dot products) will be bound with a higher attention score between them than dissimilar nodes.

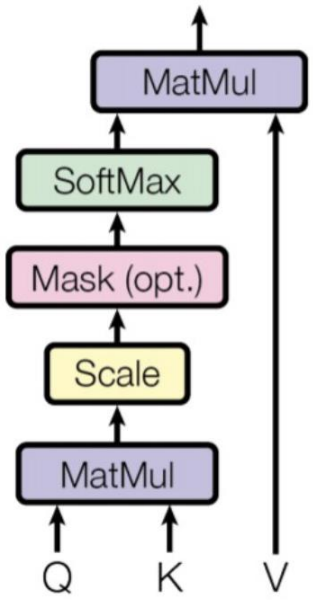


Figure 1.9: *Scaled Dot-Product Attention [13].*

Multi-Head Attention

Instead of single dot-product attention on the whole feature vectors q, k, v it was found that linearly projecting those vectors h times with different, learned linear projection was more beneficial. For example, if the dimensionality of the graph features d_{model} was 1024 and we selected 4 heads, then we would end up with 4 triplets of Q, K, V matrices with 256 dimensions, along with their corresponding weight matrices. In its general form, multi-head attention is described as:

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \dots, head_h) \cdot W^0$$

with each head corresponding to the previously calculated dot-product attention.

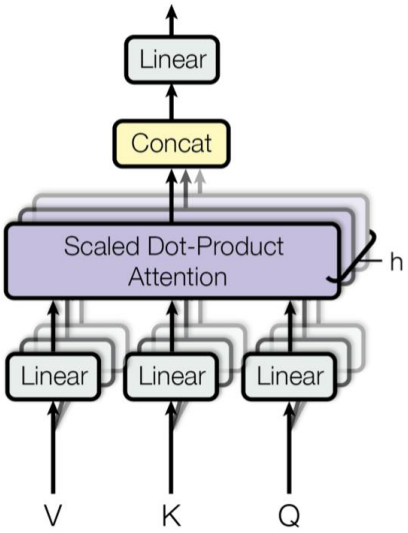


Figure 1.10: *Multi-Head Attention [13].*

Vanilla Encoder Architecture

The previously mentioned attentions mechanisms are the two most important parts concerning the architecture used in this thesis. Since it was originally designed to deal with NLP tasks, the modified Graph Transformer that we employ differs with the vanilla transformer in terms of attentional mechanisms but not in the general encoder architecture. Thus, we end up with *Figure 1.11* which, in this thesis, will be considered the universal architecture of the transformer encoder. As far as Positional Encoding is concerned, it is out of the scope of this thesis to thoroughly explain the math behind the original version. In general, it refers to a vector added to the initial node features that points each words location in the sentence. This positional encoding vector has a sinusoidal form, and the reader is encouraged to briefly study it in the original paper. Our modified positional encoding will be introduces in the Methods section of this thesis [13].

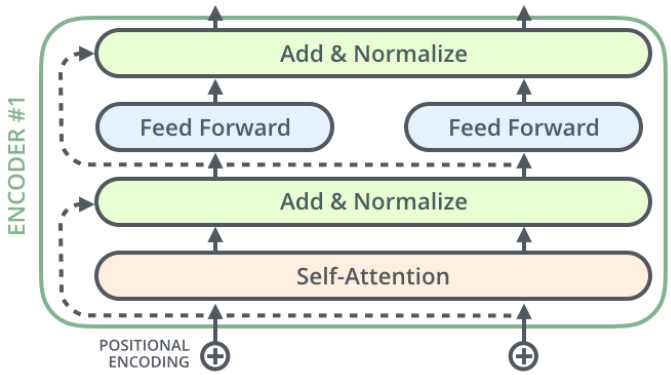


Figure 1.11: *Transformer Encoder [13].*

2 Material and Methods

2.1 Data

In general, vital for the accomplishment of a given task using AI algorithms is the model's feeding dataset. In Computer Science this concept is also known as Garbage in – Garbage out (GIGO), which indicates the importance to ensure that the used data are of the best available quality in order to attain robust predictive results.

The ways of gathering and preprocessing said data will be presented in the next sections of the thesis. Some of this data is available at the NTUA's System Biology Lab Github page, but due to the restrictive size of others such as the signaling network graphs, the option of uploading them to our repository was not available.

2.1.1 DeepSNEM Data

Compounds' signaling networks were created using the CARNIVAL pipeline and the transcriptomic signatures of the CMap dataset, resulting in a large scale dataset of signaling networks that can aid future studies.

2.1.2 CMap

CMap or the Connectivity Map project by the Broad Institute LINCS Center for Transcriptomics, have played a pivotal role in this field, by providing large datasets of compounds' transcriptomic signatures and methods for their analysis, comparison and interpretation.

The transcriptomic signatures needed to develop the appropriate signaling networks retrieved from CMap. The version of CMAP that was used was the GSE92742, with a level 5 transformed z-score. Note that only the differential expression of the 978 landmark genes in the L1000 assay was considered [14].

2.1.3 TAS Quality

For each gene expression signature, a quality score was derived, based on its transcriptional activity score (TAS), the number of biological replicates and whether the signature is considered an exemplar[14]. TAS is computed as the geometric mean of the Signature Strength (SS, the number of differentially expressed genes within a signature with absolute z-score greater than 2) and the Replicate Correlation (CC, 75th quantile of the spearman correlations between all pairwise combinations of replicate level 4 profiles on a given experiment) for a signature, scaled by the square root of the number of landmark genes. TAS ranges from 0 to 1 and the quality score ranges from 1 to 8, with the category of quality 1, which corresponds to a transcriptomic activity score greater than 0.4 and more than 2 replicates, containing the best quality signatures. For the selected experiments, only the 7788 available signatures of Quality Score 1 were selected to ensure the validity of our results.

Based on this quality score, only the signatures with the highest quality score were selected. An overview of the transcriptomic signatures used in this study can be found in *Table 1*. For each signature, TAS were inferred using the DoRothEA R package [9]. This method utilizes a knowledge base of signed TF-target interactions called Regulons and the VIPER enrichment algorithm to calculate TF activity scores [14].

Quality score	TAS	Number of replicates	Exemplar
Q1	> 0.4	> 2	True
Q2	0.2 – 0.4	> 2	True
Q3	0.2 – 0.4	≤ 2	True
Q4	0.2 – 0.4	> 2	True
Q5	0.2 – 0.4	≤ 2	True
Q6	< 0.1	> 2	True
Q7	< 0.1	≤ 2	True
Q8	< 0.1	< 2	False

Table 1: Signature quality Score

2.1.4 CARNIVAL

CARNIVAL (CAusal Reasoning pipeline for Network identification using Integer VALue programming) [16] is a causal network contextualization tool, that identifies upstream regulatory signaling pathways by using downstream gene expression data. It integrates various sources of prior knowledge, like signed and directed protein-protein interaction networks [5][17], transcription factor targets, as well as pathway signatures.

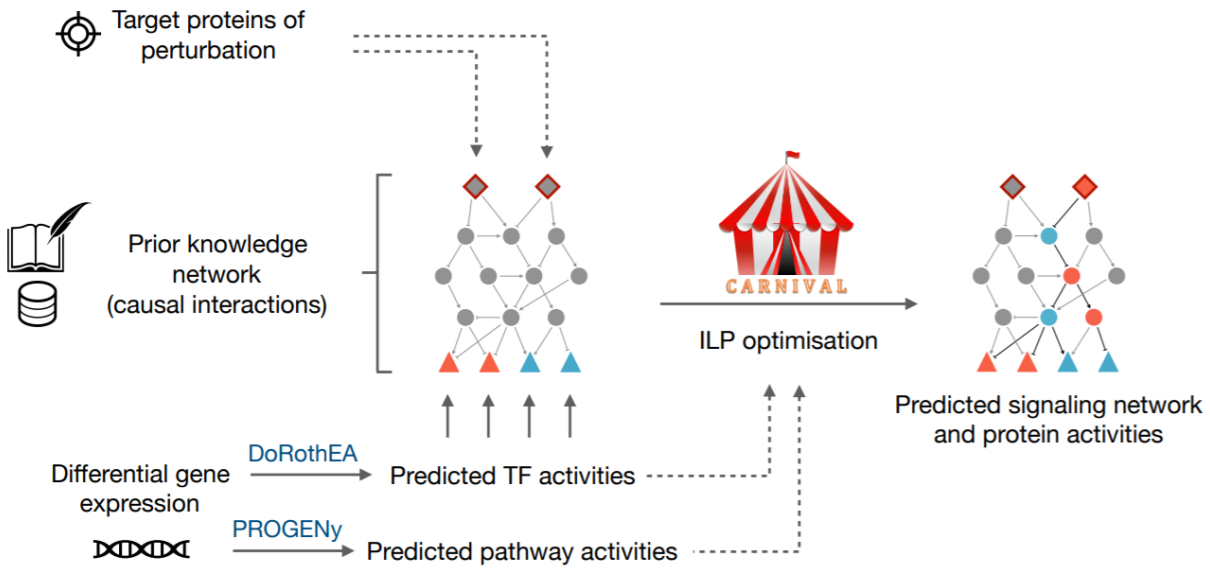


Figure 2.1: *Schematic overview of deepSNEM. For each compound-induced differential expression signature, a signaling network is created using the CARNIVAL framework. Then an unsupervised DL model is tasked to encode the created signaling network in a high dimensional embedding that best captures the input graph information.*

For each signature, transcription factor (TF) activity scores were inferred using the DoRothEA R package. This method utilizes a knowledge base of signed TF-target interactions called Regulons and the VIPER enrichment algorithm to calculate TF activity scores. In addition, for each compound perturbation, the discretized TF activities of DoRothEA were transformed into signaling networks using the CARNIVAL pipeline. CARNIVAL solves an ILP optimization problem to infer a family of highest scoring subgraphs, from a prior knowledge network of signed and directed protein-protein interactions, which best explain the TF activities, subject to specific constraints. In our approach the OmniPath network was used as the global prior knowledge network [18]. Furthermore, the CARNIVAL pipeline without using the perturbation targets as input was utilized (InvCARNIVAL method). Finally, the Integer Linear Programing (ILP) formulation of the problem was solved using the IBM ILOG CPLEX solver, which is freely available through the Academic Initiative.

The CARNIVAL pipeline was ran in parallel and without using the perturbation's known targets as input (InvCARNIVAL). The signaling network dataset was created with an older version of CARNIVAL in R version 3.6, but the same parameters can be used in the latest version of CARNIVAL. The main parameters, which can be found in *Table 2*, are the time limit until the optimization terminates (timelimit), the allowed number of solutions to be generated (limitPop), the allowed number of solution to be kept in the pool of solution (poolCap) and the external ILP Solver used. The rest parameters can be set to the default of each CARNIVAL version.

Execution mode	Parallel
inverseCR	TRUE
ILP Solver	Cplex
timelimit (in minutes)	1800
limitPop	500
poolCap	100

Table 2: CARNIVAL pipeline parameters

The end result, after using CARNIVAL, were 7788 weighted, signed and directed signaling networks, as well as their corresponding unweighted networks per signature, a number which varies from 5 to 100 per weighted signaling network. The weighted networks are produced by adding the unweighted ones, thus edge weights describe the percentage of times a certain edge appeared in the unweighted graphs.

2.2 Graph Features

In order to feed the CARNIVAL output graphs to the above described DeepSNEM model, node and edge features need to be properly established so that their mathematical equivalents provide useful representations.

Node Features

Each node of the graphs that is used in this thesis, is an individual protein of a cell signaling network. Thus, when each graph is being processed, each node's features need to have a multi-dimensional distributed representation that mathematically describes the proteins' various modes of action and their biological significance i.e. their place in the biological map. To deal with this issue, embeddings that are concerned with the amino acid sequence of each individual protein, were used. In other words the aspects of protein function and structure are encoded based on each individual proteins' amino acid sequence.

A novel way to represent protein sequences as continuous vectors (distributed representations is presented in [19]. SeqVeq (Sequence-to-Vector)2 uses a bi-directional model inspired from NLP tasks called ELMo [20] to capture the biophysical properties of sequences from big unlabeled data, specifically the UniProt50 database. This method has been proved rather effective in terms of predictive results in various tasks, just by using protein sequence data, outperforming even some methods using evolutionary information.

Edge Features

In every graph, the edges represent the type of connection two nodes share. In our case, the connection between two neighboring nodes (in a directed fashion) depict three different relational entities:

Protein Interaction: In the cell signaling network, which is computationally formulated as a Directed Acyclic Graph (DAG), proteins either upregulate or downregulate the next protein in each branch of the DAG, originally being represented as a 1 or a -1 respectively. Since this formulation cannot be directly understood by neural networks, it was changed with one hot vectors, thus changing 1 to [1 0] and -1 to [0 1], enforcing a categorical attribute to this connection.

Edge Weight: In the case of weighted graphs, it quantifies the appearance of the edge in each of the unweighted graphs from which the original was produces. Ranges from 0 to 1, where 1 means this edge appeared in every unweighted graph.

PPR Weight: In the Methods section, Personalized Page Rank is presented as a method to enforce a relative positioning feature of each protein inside the graph. This attribute is represented as a single number ranging from 0 to 1 and signifies the ease of getting from protein A to protein B and is directly linked to the aforementioned edge weight.

2.3 Mechanisms of Action

Each signature ID corresponds to GEx data, after a drug has been administered to a cell line, where the corresponding cell signaling network exhibits the cells' response to each drug. For much less than half of the experiments, specifically 2733 signatures, we were able to acquire the administered drugs' possible mechanism of action(s). Since the original labels were not consistent, we had to group similar labels together, e.g. grouping all DNA or kinase inhibitors, or arbitrarily select one of the available mechanisms of action for each drug and if possible, grouping that as well. By following this procedure, we end up with 261 unique mechanism of action labels. It is very important to take into account the fact that each one of those labels is not definitive and unique for each drug. More that one labels may correspond to one specific drug, and this heavily undermines the training evaluation metrics. The goal of those labels is to provide us with a baseline evaluation procedure, in order to focus our attention in interpreting each signaling network pathways in terms of an attributed mechanism of action.

2.4 DeepSNEM Model

In this thesis DeepSNEM model was used in order to investigate the relationship between the signaling network effect of a compound perturbation in a cellular model and the compound's Mechanism of Action.

DeepSNEM is a novel unsupervised graph deep learning pipeline, developed by the NTUA's Biomedical Systems Laboratory. Its purpose is to encode the information in the compound-induced signaling networks in fixed-length high-dimensional representations. The core of DeepSNEM is a graph transformer network, trained to maximize the mutual information between whole-graph and sub-graph representations that belong to similar perturbations.

It is important, before the actual description of the used DeepSNEM model, to provide the reader with some additional and more detailed information about the methods and models on which the DeepSNEM model is based on.

In the previous chapter of this thesis, brief overview of the original Transformer [13] was presented. In order to process the cell signaling network graphs using a neural network, instead of the standard way of employing a Convolutional Graph Neural Network, a modified version of the Transformer for graphs was implemented. There are three different functional differences between our model and the original version, which will be presented in the following section.

Positional Encoding

As mentioned in the theoretical section of this thesis, the original Transformer used a method called Positional Encoding to enforce a sense of placement of each word inside the sentence. In similar fashion, in order for each protein to have an edge feature that depicts its relative position with child and parent proteins in the DAG, an alternative algorithm was used called Floyd-Warshal algorithm.

Floyd-Warshall Algorithm

The Floyd-Warshall algorithm is an algorithm used to calculate shortest path distances in a weighted graph with positive or negative edges, by comparing all possible paths through the graph between each pair of vertices. Let $dist(k, i, j)$ be the length of the shortest path from i to j that uses only the vertices $u_1, u_2, \dots, u_k$ as intermediate vertices. Then:

- k=0 is our base case as $dist(0, i, j$ is the length of each vertex i to vertex j if it exists, and its 1 otherwise.
- $dist(k, i, j) = \min(dist(k-1, i, k) + dist(k-1, j, k), dist(k-1, i, j)$

The pseudocode of the Floyd-Warshall algorithm is described bellow:

Algorithm 1 Floyd Warshall Shortest Path Distance Matrix

```
for i ← 1 to N do
    for j ← 1 to N do
        if there exists and edge from i to j then
            dist[0][i][j] ← length of edge from i to j
        else
            dist[0][i][j] ← ∞
        end if
    end for
end for
for k ← 1 to N do
    for i ← 1 to N do
        for j ← 1 to N do
            dist[k][i][j] ← min(dist[k − 1][i][j], dist[k − 1][i][k] + dist[k − 1][k][j])
        end for
    end for
end for
```

Then $dist[N][i][j]$ describes the shortest path distance between node i to node j.

Activity Embeddings

Apart from the activity edge features that were described in a previous section, each node (protein) is embedded with two distinct positive values that range from 0 to 1. Those two values refer to the frequency each node in the weighted graphs had an upregulated or downregulated activity value in the initial unweighted graphs from which it was created.

Instead of just plugging those values in the feature vector of each individual protein, thus assigning only two weight values to each activity, we decided to use a more appropriate method to enhance the presence of such important features. By projecting this 1×2 x_{act} vector into a $1 \times d_k$ vector, where d_k is the dimensionality of each protein feature vector, we end up with a trainable activity vector $u_{act} = x_{act}\, w_{act}$, with w_{act} being the $2 \times d_k$ weight matrix. The activity embedding is added with the trainable feature vector u_{prot} of each protein before they are processed by the Transformer encoder.

Modified Scaled Dot-Product Attention

The modified version of the scaled dot-product attention described in a previous section, has to account for both the new positional encoding method as well as the insertion of the edge features the DAG contains, something the word sentences that were processed with the original Transformer did not include.

Recall the query and key matrices Q, K from equation

$$Attention(Q,K,V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{d_K}}\right)$$

In the modified attention scheme, $U^h_{act} = X^h_{act} W^h_{act}$ and $U^h_{prot} = X^h_{prot} W^h_{prot}$

represent the matrices of each proteins activity and functional features for head h respectively. $U_{edge} U_{edge}$ represents the (3xN) edge matrix of the graph, with features described above in the Data section, with N being the number of proteins. So, the modified attention weights W_{attn} are calculated in the following scheme:

$$W^h_{attn} = softmax\left(\frac{\left(U^{Q_h}_{act} + U^{Q_h}_{prot}\right) * \left(U^{K_h}_{act} + U^{K_h}_{prot}\right)^T}{\sqrt{d_k}} + \beta * (U_{edge} \circledast W_e)\right)$$

where W_e is the (1x3) edge weight and b is the bias. Essentially, the process $U_{edge} \circledast W_e$ describes a pointwise convolution operation on the edge features. Also, β is a trainable parameter that weights the importance of the edge features.

Since the multi-head attention scheme remains the same, after the concatenation of each attention head, the value matrix V is weighted using the concatenated attention heads as

$$X = W_{attn}\left(U^V_{act} + U^V_{prot}\right) \quad (\boldsymbol{Eq\ 4.1})$$

where X is the $(N \times d_K)$ resulting node feature matrix before the pointwise feedforward network, following the typical Transformer architecture. Adding our relative positional embedding and the trainable parameter c, results in the final form of the self attention mechanism:

$$W^h_{attn} = softmax\left(C^h + \beta\left(U_{edge} \circledast W_e\right) + c * PD\right)$$

where PD is the relative positional encoding matrix and is shared among each head. D is defined as the pairwise protein embedding multiplication matrix for each head:

$$C^h = \left(U^{Q_h}_{act} + U^{Q_h}_{prot}\right) * \left(U^{K_h}_{act} + U^{K_h}_{prot}\right)^T$$

The Floyd-Warshall alternative exhibits a different attention scheme, which is as follows:

$$W_{attn}^{h} = softmax\left(C^{h} + \beta * \left(U_{edge} \circledast W_{e}\right) + \frac{\left(U_{act}^{Q_h} + U_{prot}^{Q_h}\right)R_{c}^{FW}}{\sqrt{d_{head}}}\right)$$

where R_c^{FW} is the corrected R^{FW} matrix in terms of dimensions, so that the equation above holds. We can now identify three different components that arise from the attention mechanism. C^h is the content attention component, $\beta\left(U_{edge} \circledast W_e\right)$ is the edge component and $\left(U_{act}^{Q_h} + U_{prot}^{Q_h}\right)R_c^{FW}$ is the position component of each attention head.

Deep Graph Infomax

Information Theoretic Definitions [22][23]

Entropy: Let X be a random variable on a (discrete) space $\boldsymbol{X}$ and x an element sampled from $\boldsymbol{X}$. For every positive integer d, we denote by $\boldsymbol{X}$ a $d-$dimentional random vector $(X_1, \dots, X_d) \in \boldsymbol{X^d}$ and by the letter **x** an element from $\boldsymbol{X^d}$. The Shannon entropy of a random variable X on a discrete space $\boldsymbol{X}$ measures its uncertainty during an experiment and is defined as:

$$H[X] = -\sum_{x \in X} P(X = x)\log\left[P(X = x)\right]$$

The joint entropy of a pair of random variables (X, Y) expresses the uncertainty one has about the combination of these variables:

$$H[X, Y] = -\sum_{x \in X, y \in Y} P(X = x, Y = y)\log[P(X = x, Y = y)]$$

Finally, the conditional entropy of a random variable X given another variable Y expresses the uncertainty on X which remains while Y is known:

$$H[X|Y] = -\sum_{x \in X, y \in Y} P(X = x, Y = y)\log[P(X = x|Y = y)]$$

Mutual Information It is a general measure of the dependence between two random variables X, Y. It expresses the quantity of information one has obtained on X by observing Y. The discrete mutual information between two random variables X and Y is defined as:

$$I(X;Y) = \sum_{x \in X, y \in Y} P(X = x, Y = y) \log \left[\frac{P(X = x, Y = y)}{P(X = x)P(Y = y)} \right]$$

If we recall the definition of the Kullback-Leibler divergence between the distributions P, Q

$$KL(P||Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)}$$

then Eq. 4.1 describes the KL divergence between the joint distribution $P(X)$ and the product distribution $P(X)P(Y)$. In terms of Shannon entropy, MI can be defined as:

$$I(X;Y) = H[X] - H[X|Y] = H[X] + H[Y[\ -H[X,Y] = H[X,Y] - H[X|Y] - H[Y|X]$$

The Deep Graph InfoMax [24][25] approach to learning a suitable encoder relies on maximizing local mutual information i.e. to obtain node representations that capture the global information content of the entire signaling network, which in this case is represented by a summary vector $\vec{s}$. We also refer to $\vec{z}$ as each node's patch representation after the DAG Transformer Encoder forward pass.

For the graph-level summary vectors $\vec{s}$, we use a readout function $\mathcal{R} : \mathbb{R}^{N \times d_K}$, which in our case is either an average pooling of the node features, or the Set2Set [30] global pooling method $(\vec{s} = Set2Set(X))$.

In order to maximize the local MI, a discriminator $D : \mathbb{R}^{d_K} \times \mathbb{R}^{d_K} \to \mathbb{R}$ is used so that for each node i of the graph $D(\vec{z}_i, \vec{s})$ it represents the probability scores assigned to the summary-patch pair.

The functionality of the discriminator depends on the existence of both positive and negative samples. For the graph G with a summary $\vec{s}$ negative samples are produced by pairing the summary with patch representations from another graph G namely $\vec{z}_j$. We use a combination of corrupted samples from each input graph as well as patch representations from graphs with a different signature id i.e. a different experiment, that should be different in nature. Corrupted representations are produced from an explicit, stochastic corruption function $C : \mathbb{R}^{N \times d_K} \times \mathbb{R}^{N \times N} \to \mathbb{R}^{M \times d_K} \times \mathbb{R}^{M \times M}$ such that $\vec{G} = C(G)$. In our case, C is a random permutation of both the node features as well as the edge positions of the whole graph. Note that we slightly abuse the notation of G. With this symbol, in our case, we represent the node and adjacency matrix (X, A).

Noise Contrastive Estimation (NCE) [26] is used as a lower bound on MI along with a standard Binary Cross Entropy (BCE) loss between the samples from the joint (positive examples) and the samples from the product of marginals (negative examples).Thus, the objective to be optimized is:

$$\mathcal{L} = \frac{1}{N+M}\left(\sum_{i=1}^{N}\mathbb{E}_{\mathcal{G}}[log\mathcal{D}(\vec{z}_i,\vec{s})] + \sum_{j=1}^{M}\mathbb{E}_{\tilde{\mathcal{G}}}\left[log\mathcal{D}(\vec{z_j},\vec{s})\right]\right) \quad (\boldsymbol{Eq\ 4.2})$$

As the discriminator probability estimation function we can choose the Fermi-Dirac distribution used in the Graph Autoencoder, as an alternative to the sigmoid function used in the original paper, due to the extra two degrees of freedom it includes. The loss function of Eq. 4.2 corresponds to the MI estimator used in the original Deep Graph Infomax paper.

We can use a variation of this loss function based on the Jensen-Shannon Mutual Information estimator as in InfoGraph [27] with the following notation, based on Nowozin et.al [28]. Let $I_{\varphi,\psi}$ be the mutual information estimator modeled by a discriminator D_ψ parametrised by a neural network with parameters ψ. As φ, we denote the parameters of our modified transformer neural network. The Jensen- Shannon MI estimator is:

$$I_{\varphi,\psi}\left(h_\varphi^i, s_\varphi\right) \coloneqq \mathbb{E}_{\mathbb{P}}\left[-sp\left(-D_{\psi,\varphi}(z_\varphi^i, s_\varphi)\right)\right] - \mathbb{E}_{\widetilde{\mathbb{P}}}\left[sp\left(-D_{\psi,\varphi}(\widetilde{z_\varphi^i}, s_\varphi)\right)\right]$$

where $\mathbb{P}$ is the empirical distribution of the input data set, $\widetilde{\mathbb{P}}$ is the negative distribution from which we sample from (either via the corruption function C or even better $\widetilde{\mathbb{P}} = \mathbb{P}$, i.e. masking networks from different signatures and sp is the softplus function $sp(z) = (1 + e^z)$. Thus, the new model's unsupervised loss is:

$$\mathcal{L} = -\frac{1}{N}\sum_{1}^{N} I_{\psi,\varphi}\left(z_\varphi^i, s_\varphi\right) + \gamma\, D_{\varphi,\psi}(\mathbb{V}||\mathbb{U}_{\varphi,\psi})$$

The second term is a regularization loss, which denotes matching the pushforward distribution of our summary vectors to a prior distribution, with the most successive being the uniform distribution.

2.5 Graph Transformer Model - DeepSNEM-GT-MI

During the model's research and development process, different models' variations were developed and tested. Based on the evaluation process's results, the model that was selected for this thesis to focus on, consists of a graph transformer trained to maximize the mutual information of nodes belonging to the same signature (termed deepSNEM-GT-MI).

In a technical view the deepSNEM-GT-MI model encodes the input matrices of each signaling network using two multi-head attention layers. Each multi-head attention layer computes the attention score using the key, query and value matrices, which are later combined using a simple feed forward network. The output of this network is used to produce the whole-graph representations using the Set2Set LSTM model. The mutual information is approximated using simple discriminators in order to train the model. The final node embedding size is set to 128, while the whole-graph representation embedding size is set to 256.

In further detail, each compound-induced signaling network is represented as a labeled, signed and directed graph $G = (V, E)$, with nodes (V) being the proteins and edges (E) denoting the directed physical interaction between them.

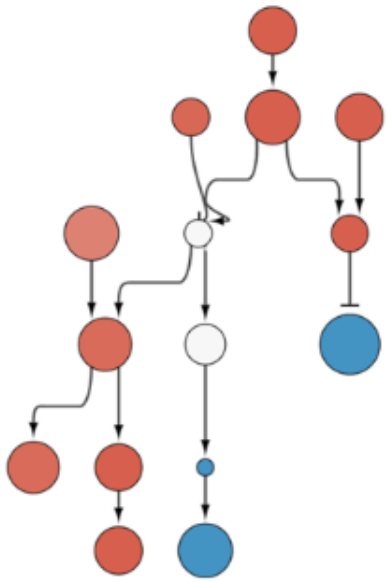


Figure 2.2: *An example input graph.*

Additionally, the activity of each protein is represented as a node attribute, while the inhibition or activation of each edge is represented as an edge attribute. Each input graph to the deepSNEM-GT-MI consists of a node feature matrix $X_{(prot)}$, a node activity embedding $X_{(act)}$ and a node proximity embedding $X_{(edge)}$. The node feature matrix contains the initial protein features of each graph, which were created using the SeqVeq protein sequence model [41]. For each protein, the node activity embedding is a projection of the node's activity to the dimensions of the SeqVeq features, using a single embedding layer.

The node feature and node activity matrices are added before being processed by the graph transformer. Finally, the node proximity embedding is a relative positional embedding, where each shortest path distance between nodes is calculated using the Floyd Warshall Algorithm [29]. Thus, the proximity embedding contains information about the relative distance of each node to all other nodes in the graph. The input matrices are then passed through the self-attention mechanism of the graph transformer, resulting in a final feature matrix X [30,31].

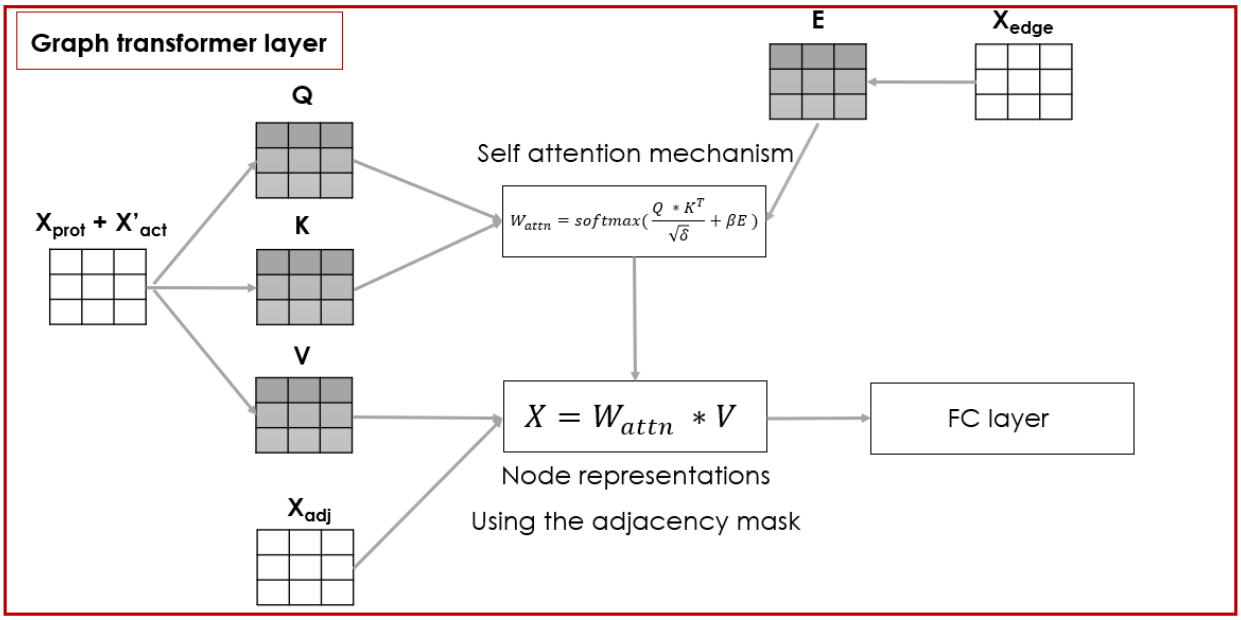
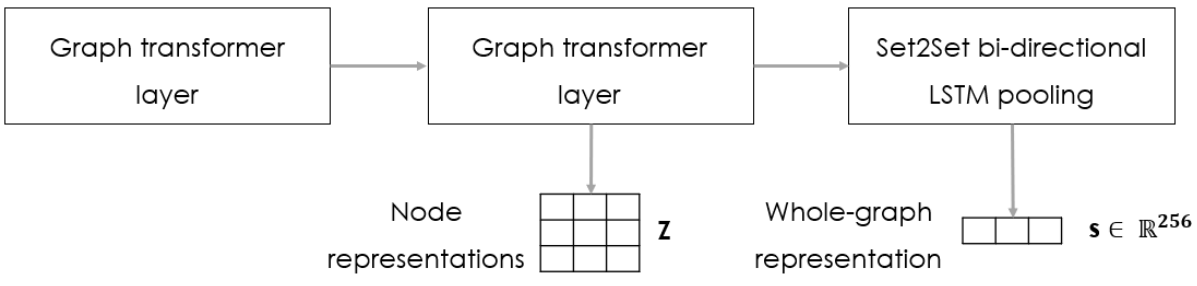


Figure 2.312: *How the graph transformer layer works.*

Finally, this feature matrix is summarized using the Set2Set global pooling method into a trainable whole-graph representation [32]. The model is trained fully unsupervised by maximizing the mutual information between node and whole-graph embeddings that are created from the same or duplicate transcriptomic signatures, using the CARNIVAL pipeline, thus resulting in similar graph representations for the same perturbation. Similar to the InfoGraph approach, the Jensen-Shannon Mutual Information estimator was used, while an additional term was added to the total loss function in order to force the embeddings to be uniformly distributed [27].



Unsupervised Loss function $\mathcal{L} = -\frac{1}{N}\sum_{1}^{N} I_{\psi,\varphi}\left(z^i, s\right) + \gamma D_{\varphi,\psi}(\mathrm{Z}||U)$

Figure 2.4: *Model's Flowchart: Outputs and Loss function*

2.6 DeepSNEM Clustering Analysis

The deepSNEM-GT-MI embeddings were clustered using the k-means algorithm. K-means algorithm is an iterative algorithm that tries to partition the dataset into K pre-defined distinct non-overlapping subgroups (clusters) where each data point belongs to only one group. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.

The optimal number of clusters were selected using the elbow method. The Elbow method is a visual method to test the consistency of the best number of clusters by comparing the difference of the sum of square error (SSE) of each cluster. The most extreme difference forming the angle of the elbow shows the best cluster number. The elbow plot of the clustering is presented in *Figure 2.5*. *Figure 2.5* shows the total within sum of squared distances between the centroids and the points of each cluster, for different values of k. We can see that the elbow starts to form around k=200.

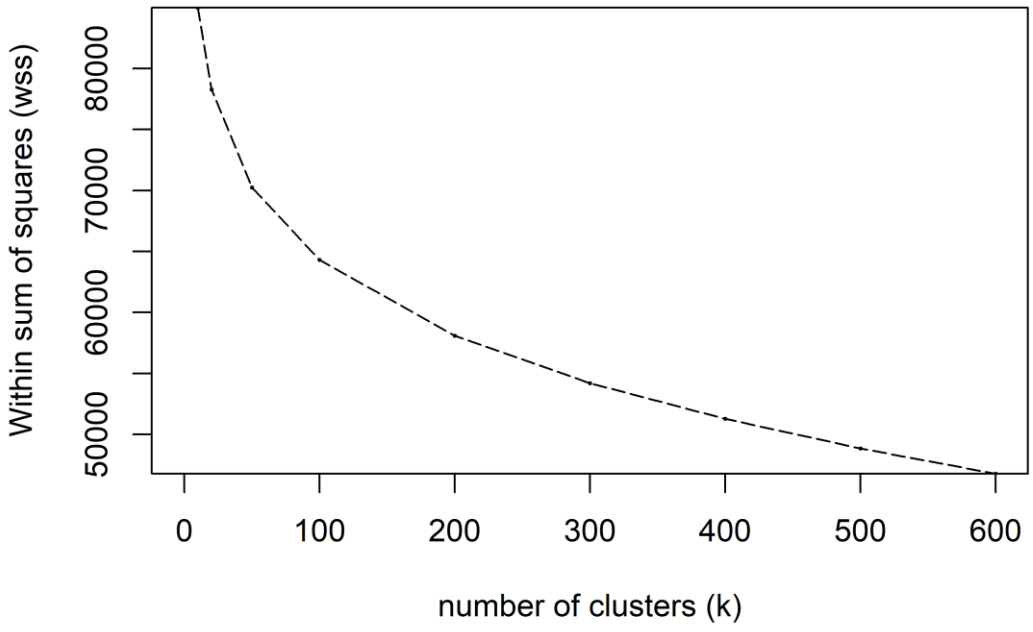


Figure 2.5: *Elbow plot of the k-means clustering of the deepSNEM-GT-MI embeddings.*

This comes in agreement with the internal diversity of the dataset, where we have 261 unique MoA labels assigned to 912 compounds. Based on the results of *Figure 2.5*, the number of clusters was set to 200.

2.7 Node and subgraph importance

It is well established until today, that ANNs have the capability to exhibit human-level performance on various datasets and tasks. This characteristic comes with the heavy price of reduced interpretability, i.e. the ability to provide meaningful and understandable explanations to human researchers. This reduced interpretability is an extremely important factor to account for when dealing with biological and medicinal cases as it reduces confidence to predictive results coming from ANNs. That is the main reason why it is crucial to develop methods and pipelines in order to improve the interpretability of the developing models and algorithms. For the DeepSNEM approach, a subgraph importance method was developed to identify the most important nodes for each graph-level representation and the subgraphs that cause the signaling networks to cluster together.

The average attribution of each node (protein) to the resulting signaling network embedding was calculated using the saliency map approach of the Captum library [33]. With the saliency approach the attributions are calculated based on the gradient with respect to the input [34]. This approach results in an attribution score for each node that shows the importance of the node to the model, when calculating the network embedding. Subsequently, a scoring function was designed in order to identify the important nodes in a specific cluster of signaling network embeddings. For each node, this scoring function calculates the product of the median rank of the node's attribution score in the cluster and the frequency that the node appears in the signaling networks of the cluster. The mathematical equivalent of the description for the nodes' score calculation is:

$$\mathbb{S}_i \sim S\left(\frac{\vartheta F}{\vartheta x_i}, f \right)$$

Finally, this score is normalized between 0 and 1. For visualization purposes, the 20 most important nodes of each cluster were connected using the shortest paths from the OmniPath PPI network that maximize the overall sum of importance scores in the connected graph.

2.8 Experimental Data – Use case

As a use case, deepSNEM was tasked to assign clusters to compounds' signaling networks generated using gene expression profiles from various experimental platforms (MicroArrays and RNA sequencing). Gene expression data from 7 additional compounds with known mechanism of action were retrieved from the GEO database.

The details regarding the experimental data used in the use case are presented in *Table 3*. Overall, the data were collected from 6 different studies, 4 cell lines and 3 different experimental platforms, i.e. Affymetrix/Agilent Microarrays and Illumina next generation sequencing. Following the deepSNEM pipeline, each differential gene expression signature was transformed into a compound induced signaling network with CARNIVAL and embedded using the deepSNEM-GT-MI model. Finally, each embedding was assigned to one of the already identified clusters.

Compound	MoA	Cell line	GSE	Platform
Sirolimus	mTOR inhibitor	MCF7	GSE116447	Affymetrix Microarray
CDK-887	CDK inhibitor	MCF7	GSE19638	Affymetrix Microarray
Panobinostat	HDAC inhibitor	A375	GSE145447	Illumina NextSeq
Sodium-Butyrate	HDAC inhibitor	HT29	GSE61429	Agilent Microarray
Belinostat	HDAC inhibitor	A549	GSE96649	Illumina NextSeq
SN38	Topoisomerase I inhibitor	MCF7	GSE18552	Affymetrix Microarray
Doxorubicin	Topoisomerase II inhibitor	MCF7	GSE19638	Affymetrix Microarray

Table 3: Information regarding the perturbations used in the use case data

3 Results

Throughout this project various tasks were assigned to our DeepSNEM model. These tasks include clustering analysis of the output embeddings using a large scale dataset of signaling networks created by NTUA's System Biology Lab, cluster assignment to a compound-induced gene expression signature as well as identification of the nodes and subgraphs which mostly influenced the proposed cluster assignment. In the section below the results of the model's performance on these tasks are presented.

3.1 Clustering analysis for MoA identification

As aforementioned a compound's perturbation Mechanism of Action is being revealed by its signaling network effect. The deeper examination of this relationship was accomplished by a two step process. First we identified groups of perturbations with similar network effect, by clustering the deepSNEM network embeddings, and then analyzed the resulting clusters based on the reported MoA of the compounds. On this front, the 256-dimensional deepSNEM-GT-MI embeddings were clustered using the k-means algorithm. The optimal number of clusters was found to be 200, according to the k-means elbow plot (link to method's section).

Additionally, in order to analyze and characterize the resulting clusters, the MoA labels provided by the Broad's Institute Repurposing Hub were utilized [35]. Out of the 3005 unique compounds, 912 were mapped to 261 unique MoA labels using the Repurposing Hub dataset. *Figure 3.1(A)* shows the 2-dimensional t-SNE projections of all available signaling network embeddings. It is important at this point, to note that t-SNE (t-Distributed Stochastic Neighbor Embedding) is a non-linear dimensionality reduction algorithm used for exploring high-dimensional data. It maps multi-dimensional data to two or more dimensions suitable for human observation or processing the output data using less computational power.

Additionally, the signaling network embeddings that belong to the top 9 most prevalent MoA labels in the dataset are presented with different colors (*Figure 3.1(A)*). In order to characterize the identified clusters, we focused on the subset of clusters that are significantly enriched for at least one mechanism (*Figure 3.1(B)*) since t-SNE wasn't able to separate the representations but revealed distinct areas where similar MoAs are grouped together. The selected clusters have at least 25% of their compound perturbations belonging to the same MoA, with a p-value lower than 10^{-6} compared to a random selection. *Figure 3.1(B)* shows the breakdown of the available MoA in the selected clusters. As it can be seen, the identified clusters are enriched for the same mechanisms that are most prevalent in the labeled dataset. As a result, DeepSNEM was able to identify 11 clusters that are significantly enriched for specific mechanisms, i.e. mTOR, HDAC, topoisomerase, protein and ATP synthesis inhibitors. We have to note that clusters that are enriched for mTOR inhibitors are also enriched for PI3K inhibitors, which is expected due to the PI3K/mTOR signaling pathway. However, the majority of the compounds in each cluster still do not have available labels regarding their MoA (represented with grey color in *Figure 3.1(B)*). Thus, due to the unknown

labels, the distribution of MoA between clusters that are enriched for the same MoA can still be quite different.

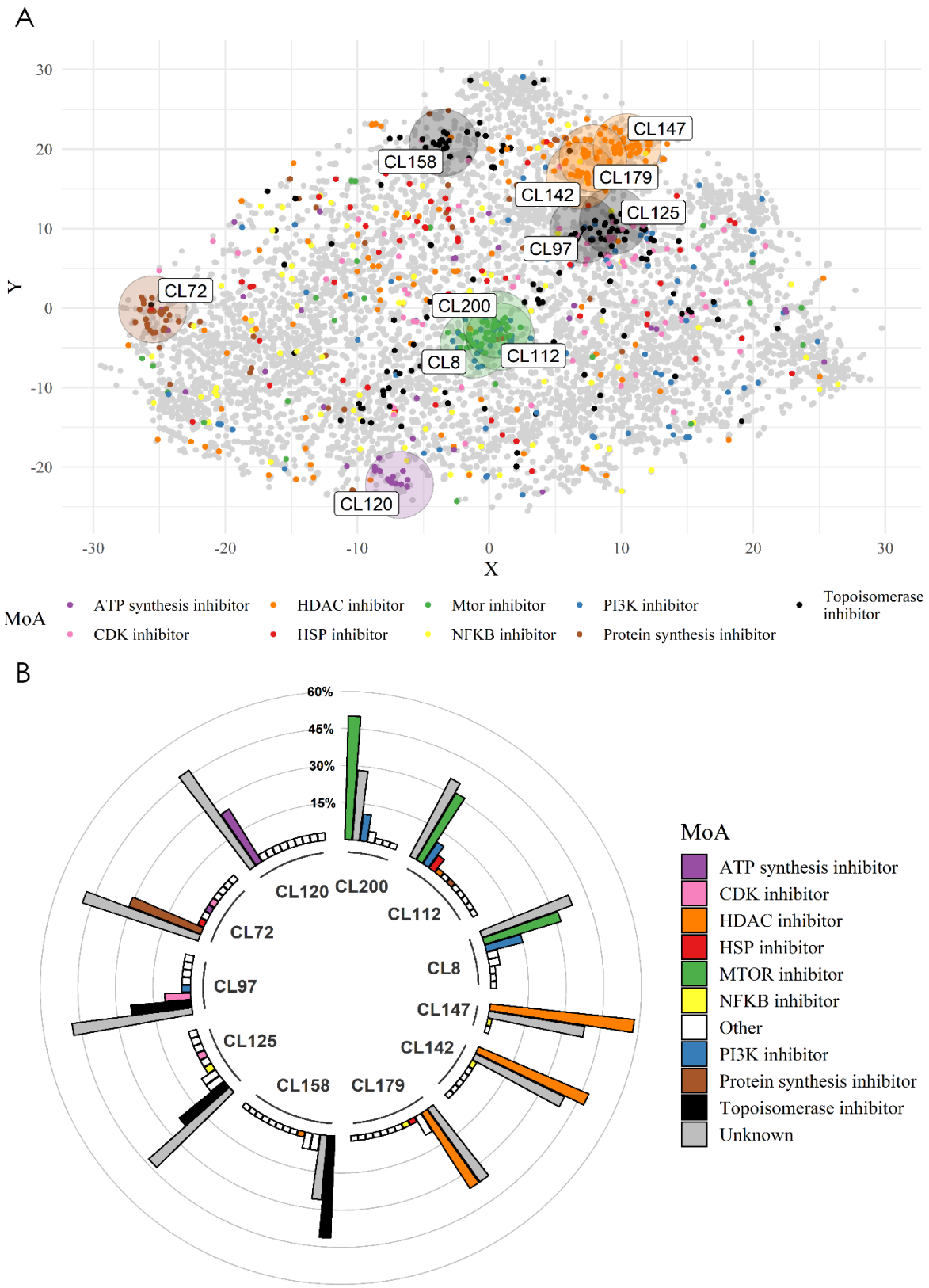


Figure 3.1: *Clustering analysis. (A) T-SNE projection of the 256-dimensional signaling network embeddings of deepSNEM-GT-MI. Different colors represent the 9 most prevalent MoA in the dataset, while the grey color represents perturbations with either unknown or other MoA. Additionally, the centers of the identified clusters are represented with circles (CL: cluster). (B) MoA composition of the analyzed clusters. The Y axis represents the frequency, as a percentage, of each MoA in the cluster (CL: cluster).*

3.2 Subgraph importance

Although ANNs have been proven to be a powerful and versatile technology used for machine learning, they also have several known disadvantages. Perhaps one of the most important shortcoming is to determine the cause of ANN decision or result. Making a specific decision is almost impossible. It's hard to believe in the reliability of ANN which is formed on real-world problems. In addition this shortcoming makes it difficult to transfer the information learned by a network to the solution of related problems. Therefore, it should be understood that it is desirable to form the extraction method or symbolic rules for the network.

In this case, the followed strategy, was the analysis of compound-induced signaling networks for MoA identification. In order to increase the interpretability and explainability of deepSNEM, we created a framework to identify the important subgraphs for the subset of clusters analyzed in the previous section. For each cluster, important nodes were identified using an aggregate score based on their importance to the embedding model and the nodes' prevalence in the cluster's graphs. *Figure 3.2(A)* shows the overlap, as a percentage, between the 20 most important nodes of the analyzed clusters. As it can be seen, clusters that are enriched for the same MoA, have a higher similarity between their most important nodes. Thus, the proposed importance framework can identify nodes of high importance in each cluster that show a connection to the cluster's most prevalent mechanism of action. For visualization purposes, the most important nodes in each cluster were connected by selecting the shortest paths between them, from the Omnipath PPI that also maximize the overall sum of importance scores in the path. *Figure 3.2(B)* shows an example of the important subgraphs for the clusters that are enriched for mTOR and PI3K inhibitors. The common most important nodes across the presented networks include the mTOR regulated transcription factors NRF1 and TFDP1 and the CSKNK2A1, RHOA, PRKACA and LCK proteins, which are involved in the PI3K-Akt-mTOR signaling pathway [36-40]. Finally, across all clusters, AKT1 and MAPK1 serve as central nodes that connect the most important nodes (*Figure 3.2(B)*). The important subgraphs for all analyzed clusters are presented in *Figure 3.3.*

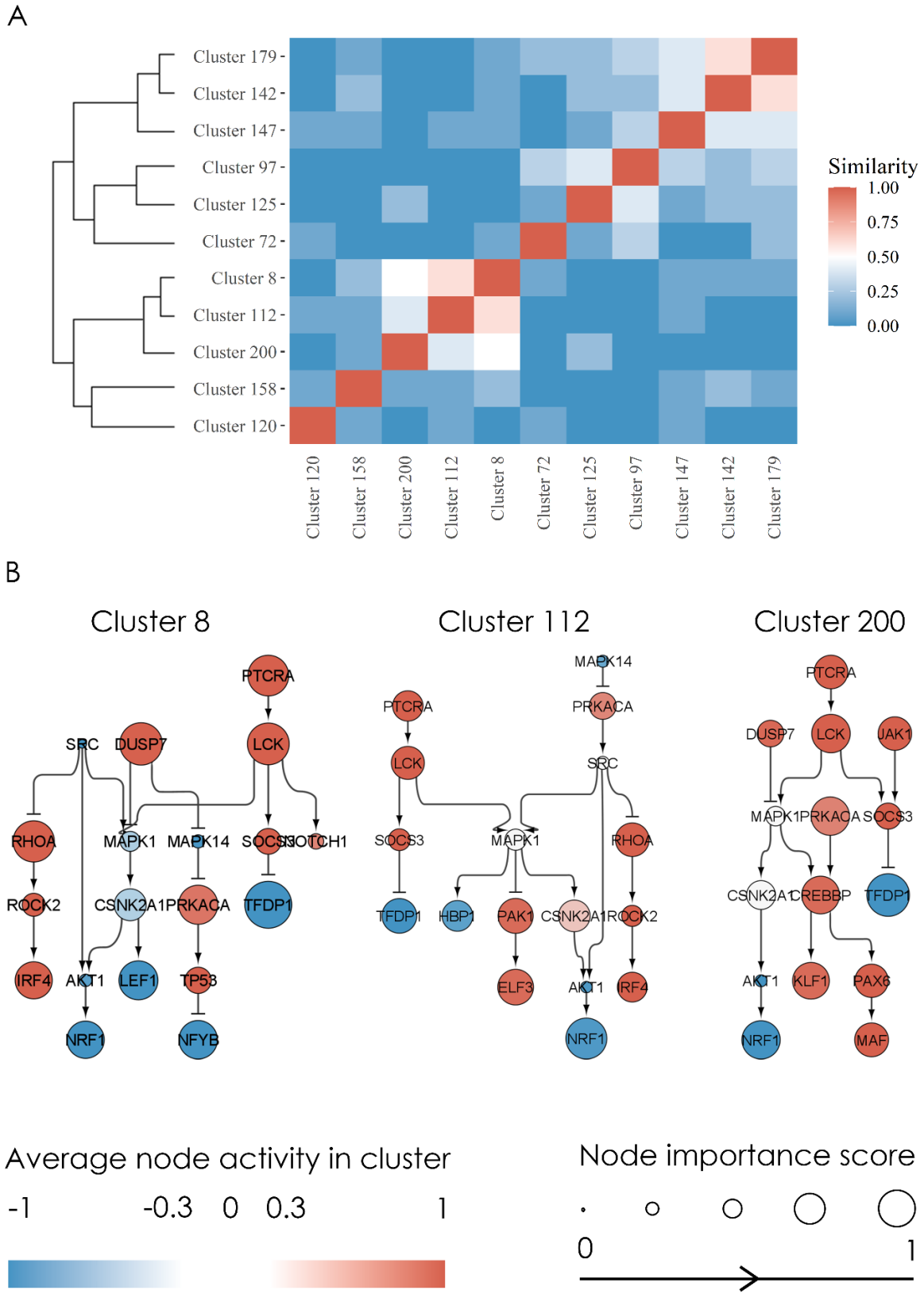


Figure 3.2: *Cluster subgraph importance. (A) Heatmap showing the similarity, as percentage overlap, between the 20 most important nodes of each cluster. (B) Important subgraphs identified for the clusters enriched for mTOR and PI3K inhibitors (Clusters 8, 112 and 200). The average activity of each node in the cluster is color coded from blue to red. Blue nodes are inhibited, while red are activated. Each node's importance score, ranging from 0 to 1, is represented by the size of the node's circle.*

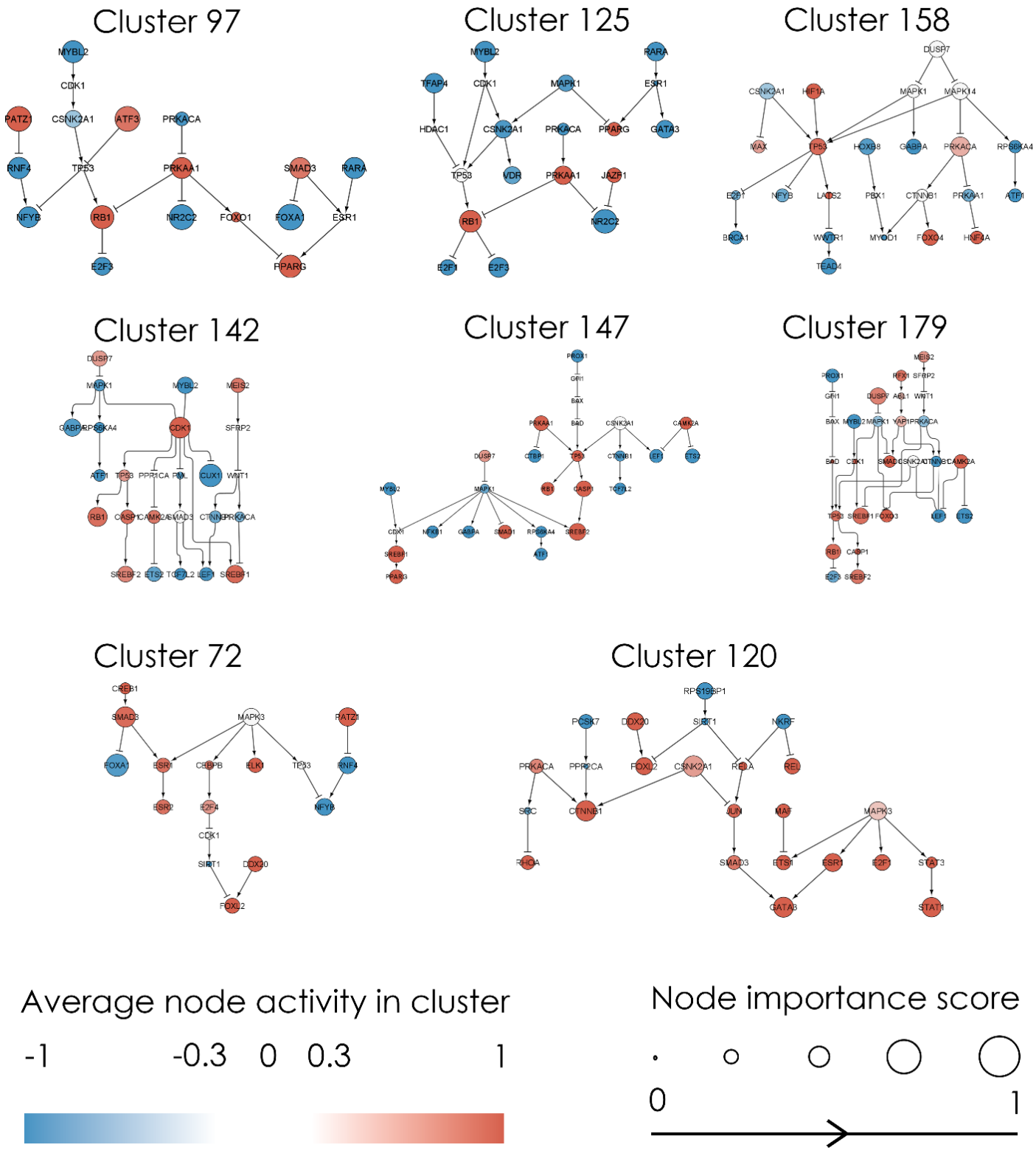


Figure 3.3 *: Important subgraphs identified for the clusters enriched for topoisomerase (Clusters 97, 125 and 158), HDAC (Clusters 142, 147 and 179), protein synthesis (Cluster 72) and ATP synthesis (Cluster 120) inhibitors. The average activity of each node in the cluster is color coded from blue to red. Blue nodes are inhibited, while red are activated. Each node's importance score, ranging from 0 to 1, is represented by the size of the node's circle.*

Use case: cluster assignment

Finally, the DeepSNEM model evaluated by its ability to assign clusters to compounds' signaling networks generated using gene expression profiles from various experiments. In other words GEx data collected by drug perturbation experiments, were used in our case to predict its prevalent MoA. It is important to note at this point that the MoA for the compounds used in the selected experiments were known and correspond to the clusters DeepSNEM was able to identify. The used embeddings for this final task were generated from GEx data retrieved by GEO platform, following the DeepSNEM pipeline as described in the Methods section.

Each embedding was assigned to one of the already identified clusters (*Table 4*).

Compound	MoA	Cell line	GSE	Platform	Cluster (CL)
Sirolimus	mTOR inhibitor	MCF7	GSE116447	Affymetrix Microarray	53
CDK-887	CDK inhibitor	MCF7	GSE19638	Affymetrix Microarray	163
Panobinostat	HDAC inhibitor	A375	GSE145447	Illumina NextSeq	22
Sodium-Butyrate	HDAC inhibitor	HT29	GSE61429	Agilent Microarray	22
Belinostat	HDAC inhibitor	A549	GSE96649	Illumina NextSeq	188
SN38	Topoisomerase I inhibitor	MCF7	GSE18552	Affymetrix Microarray	158
Doxorubicin	Topoisomerase II inhibitor	MCF7	GSE19638	Affymetrix Microarray	33

Table 4: Information regarding the perturbations used in the use case and their assigned clusters

Figure 3.6(B) shows the assigned clusters and the distribution of each cluster's available MoA. The topoisomerase inhibitor SN38 and the HDAC inhibitors Sodium-Butyrate, Panobinostat and Belinostat were assigned to clusters significantly enriched for topoisomerase and HDAC inhibitors respectively. Furthermore, the topoisomerase inhibitor Doxorubicin and the mTOR inhibitor Sirolimus were assigned to clusters enriched for their respective MoA, albeit having a large number of compounds with unknown MoA. Finally, the compound CDK-887 was assigned to a cluster that was not enriched for any particular MoA. Thus, the deepSNEM pipeline can be used to assign a cluster to a compound-induced gene expression signature, independent of the experimental platform, and provide insight into the compound's potential MoA.

For the compounds in the use case, we also compared the cluster assignment of deepSNEM to a clustering of the compounds' differential expression gene measurements. (*Figure 3.6(B)*). The MicroArray gene expression profiles following compound treatment were preprocessed

with the RMA algorithm, while the RNAseq data with the edgeR algorithm. The transcriptomic signatures of the CMap dataset were clustered with the k-means algorithm, similar to the signaling network embeddings. The elbow plot of the gene expression clustering is shown in *Figure 3.4*. Similar to the clustering of the deepSNEM embeddings, the number of clusters k was set to 200. Furthermore, *Figure 3.5* shows the t-SNE projections of the gene expression profiles, where the most prevalent MoA labels in the datasets are coded with different colors.

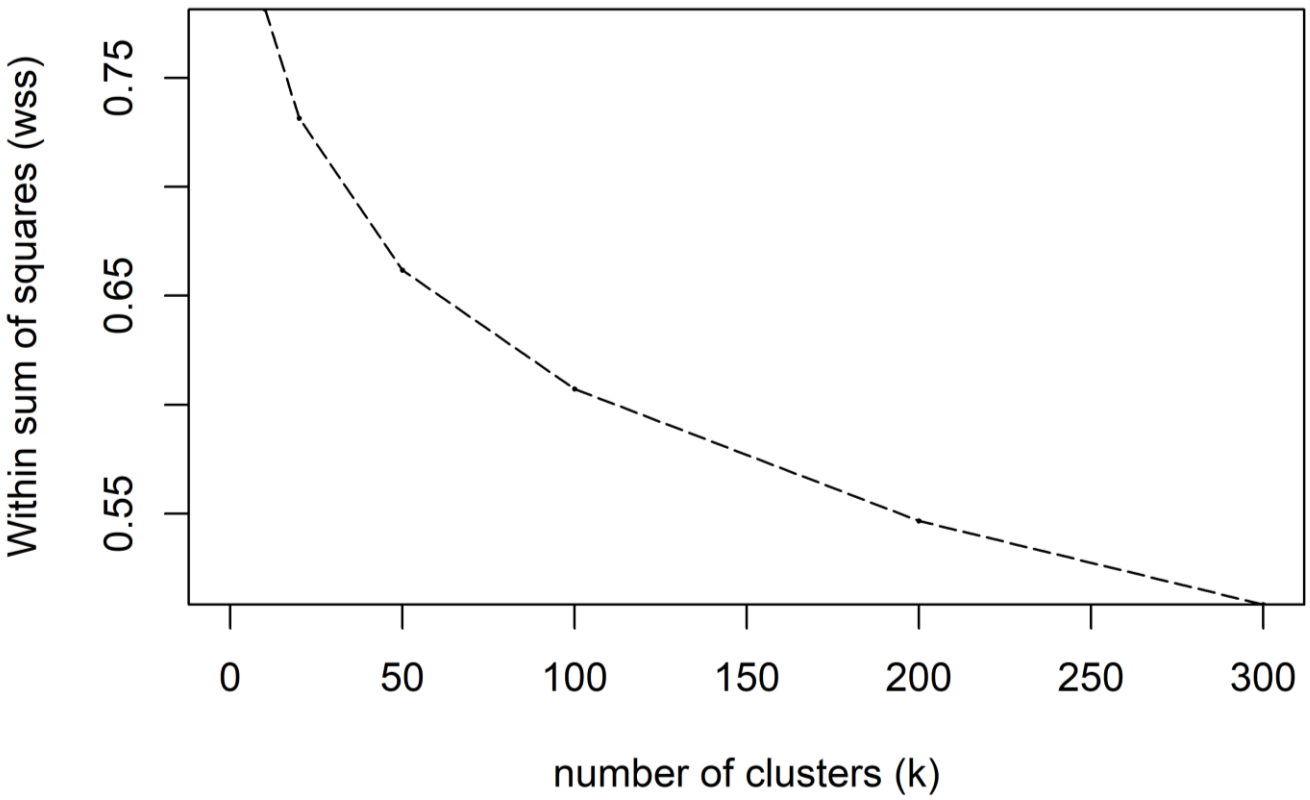


Figure 3.4: *Elbow plot of the k-means clustering of the differential gene expression profiles.*

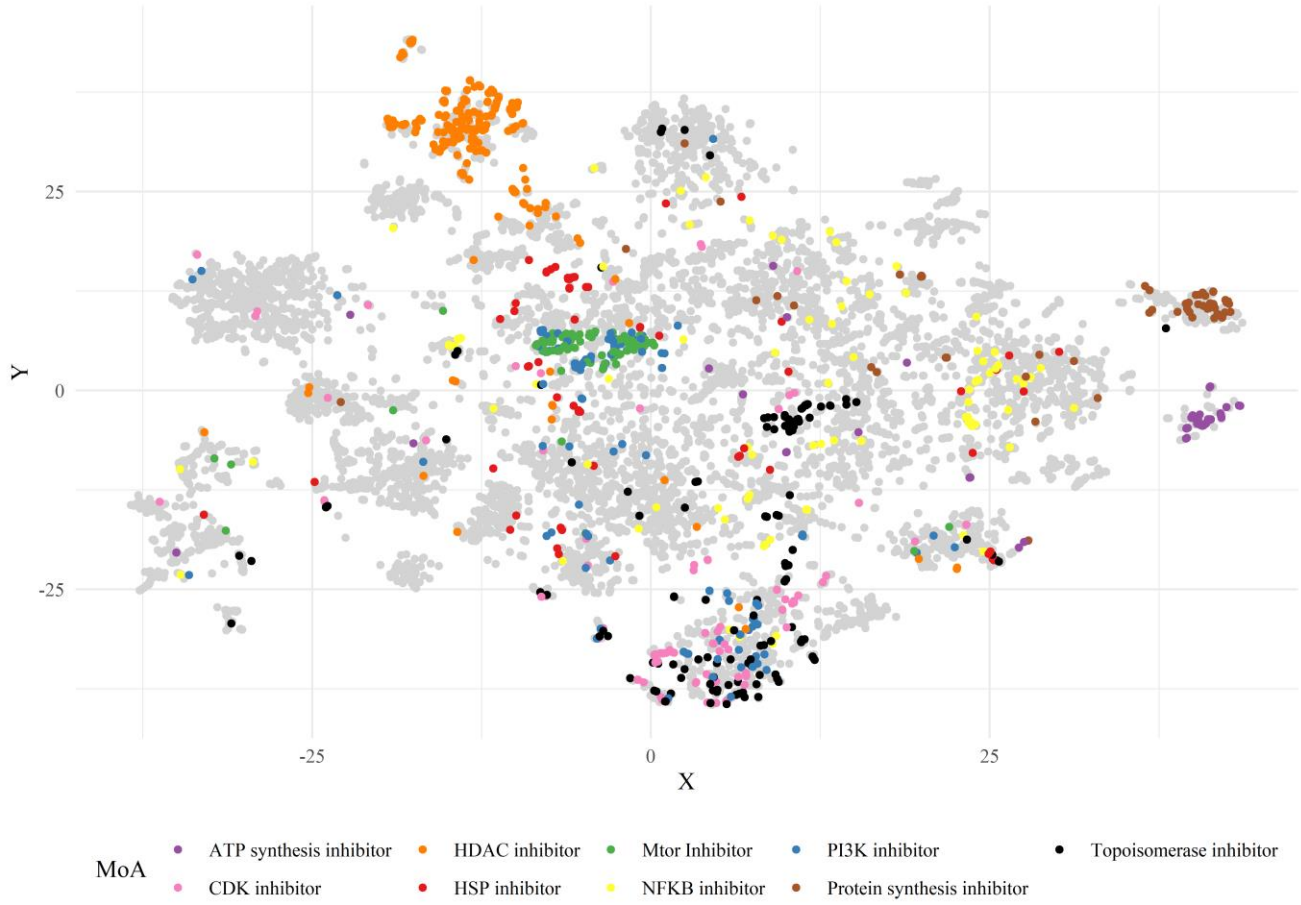


Figure 3.5: *T-SNE projection of the gene expression profiles. Different colors represent the 9 most prevalent MoA in the dataset, while the grey color represents perturbations with either unknown or other MoA.*

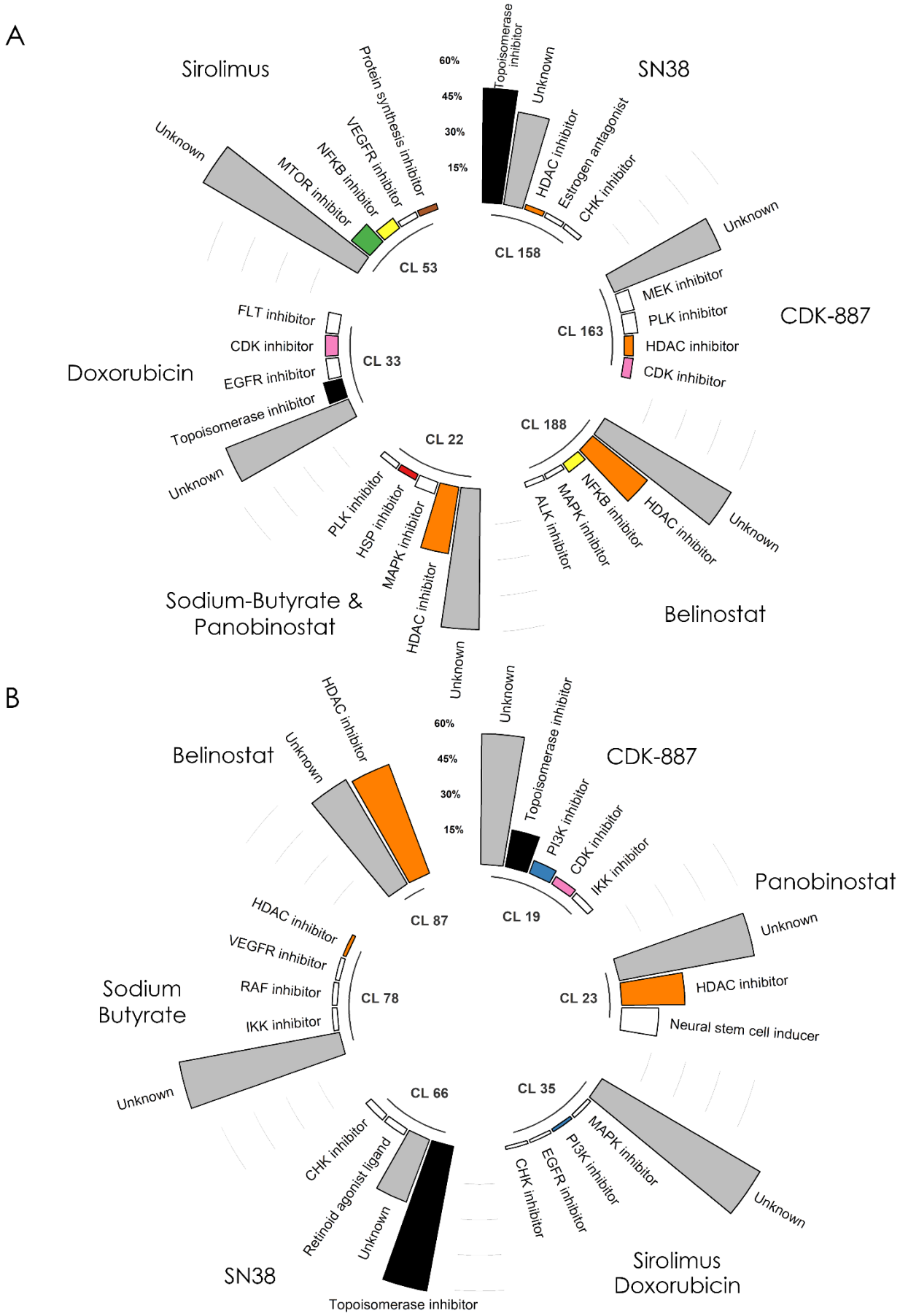


Figure 3.6 : *MoA composition of the compounds' clusters. (A) Bar plot of mechanism of action prevalence for the clusters that were assigned to the use case perturbations using the deepSNEM pipeline. (B) Similar bar plot for the assigned clusters using the gene-based clustering pipeline.*

Comparing the two approaches, SN38, Belinostat and Panobinostat were assigned to clusters composed of similar mechanisms. However, this is not the case for Sirolimus, Doxorubicin and Sodium Butyrate, which are assigned to clusters not enriched for any particular MoA, when the gene-clustering pipeline is used.

Finally, for each compound of the use case, we calculated the Jaccard similarity index between the perturbations of the identified clusters using the two methods (deepSNEM and gene-based clustering) *(Table 4*). As it can be seen in *Table 5*, across all compounds the similarity of the clusters is very low, with only the clusters assigned to the SN38 having a slightly higher Jaccard index. Thus, the deepSNEM and gene-based pipeline result in a different clustering of the perturbations, due to the different biological hierarchy of information provided by the compound-induced signaling networks and differential gene expression signatures.

Sirolimus	0.004
CDK-887	0
Panobinostat	0
Sodium-Butyrate	0.006
Belinostat	0.029
SN38	0.162
Doxorubicin	0.012

Table 5: Jaccard similarity index between the clusters that the use case compounds were assigned to, using the gene-based and deepSNEM pipelines.

4 Discussion

The changes in the protein signaling network caused by a compound perturbation can aid in studying the compound's mechanism of action in the cellular system. However, analyzing compound-induced signaling networks on a massive scale is a very complex problem, not only due to the limited availability of large datasets containing such networks but also due to the complex structure of the data. This complex structure of signaling networks limits their representation abilities and poses a challenge in identifying similarities or differences between them. In this study, we created a large dataset of compound-induced signaling networks from the CMap dataset, using the CARNIVAL network creation pipeline and developed an unsupervised deep learning model to transform them into high-dimensional and information-rich representations. This novel approach, called deepSNEM was used to identify clusters of perturbations with similar network representations and offer insight into the compounds' MoA by analyzing the distribution of MoA in the clusters.

The prediction of a compound's MoA from biological response data has gained considerable attraction in the machine learning community [41,42]. This is evident by the recent release of the CTD2 Pancancer Drug Activity DREAM Challenge, which tasked the community to predict a compound's MoA based on post-transcriptional and cell viability data [42]. Even though the learning task of MoA prediction is frequently modeled as supervised, in our approach we decided to develop deepSNEM in a fully unsupervised fashion. This decision was based on the nature of the learning task and the compounds' MoA, wherein if a compound has a reported MoA based on binding affinity data, we can't know with absolute certainty that it doesn't have additional MoA labels due to other binding targets or interactions between the proteins in a pathway. Thus, for some compounds the negative labels for all possible MoA indications might not be truly negative, rather they might be simply unknown. Additionally, another important benefit of using an unsupervised approach, is that we can greatly increase the amount of available data by including transcriptomic signatures following treatment with compounds that have no reported MoA. In deepSNEM the learning model is tasked to produce meaningful representations that capture the information included solely in the compound-induced signaling networks without taking into account the compounds' reported MoA. However, this unsupervised task makes the evaluation of the different models and the resulting embeddings quite challenging.

For the task of mechanism of action identification, we decided to use the embeddings of the graph transformed trained to maximize the mutual information between nodes that belong to networks created from the same or duplicate gene expression signatures. We argue that this deepSNEM variation is better suited to capture the information of the signaling networks, due to the graph transformer architecture and due to the mutual information task that forces networks created from the same perturbation to have similar embeddings *(see Methods 2.4)*. Finally, we have to note that the resulting 256-dimensional graph embeddings contain all the information of the input signaling networks, which makes it difficult for the t-SNE algorithm to project them in 2 dimensions, as it can be seen in *Figure 3.1(A)*.

The clustering analysis and MoA identification using the deepSNEM-GT-MI embeddings was performed by analyzing the MoA labels provided by the Broad Institute in the drug repurposing hub. Using this dataset, 912 out of the 3005 total compounds were mapped to 261 unique labels. We argue that this diversity of mechanisms and large number of compounds with unknown MoA in the dataset resulted in the large number k (k = 200) of clusters that were identified using the elbow plot of the k-means algorithm. Additionally, due to the large number of unlabeled compounds, in order to analyze the resulting clusters, we focused on a specific subset that is significantly enriched for at least one specific MoA *(Figure 3.1(B))*. Using this approach, we identified 11 clusters that each were enriched for the most prevalent mechanisms in the dataset. However, even for the clusters enriched for the same MoA, the large number of unknown compounds could result in different cluster compositions, which potentially further signifies the importance of analyzing biological response from different points of view, e.g. genes, pathways, signaling networks.

There have been many studies for the identification of a compound's MoA using biological response data. The majority of these approaches utilize post-transcriptional data and have been utilized successfully in the fields of systems pharmacology and drug repurposing [34,35]. Since the initial part of deepSNEM relies on transcriptomic data, similarities between the results and clustering of gene signatures and signaling networks are expected. This effect is evident in the presented use case, where some of the compounds were assigned to clusters with similar MoA composition between the gene-based and network-based pipeline. However, some compounds were assigned to clusters enriched for different MoA between the two approaches (*Figure 3.6*). Most importantly, between the two methods, each compound was assigned to clusters that had a very low Jaccard similarity index, meaning that the transcriptomic signatures and signaling network embeddings of deepSNEM cluster in a different way (*Table 5*). Thus, even though transcriptomic signatures do provide meaningful insight into a compound's MoA, there are cases, where analyzing the signaling networks can reveal complex relationships that are hidden in the original expression data. We argue that this is because a compound's effect on a biological system is usually caused by changes in the expression of genes that interact with each other to form specific biological processes. By supplying deepSNEM with this required prior knowledge of interactions in the form of the Omnipath PPI, the compound-specific signaling networks can provide a mechanistic view of the compound's effect and translate to the identification of its MoA [18]. Additionally, deepSNEM's signaling network creation via the CARNIVAL pipeline can provide a robust normalization factor to analyze and incorporate data from different experimental platforms (*Table 4*). Finally, the analysis of compound-induced signaling networks has the inherent benefit of increasing the interpretability of results.

The interpretability and explainability of machine learning models is a concept that has gained considerable attraction since the creation and application of powerful and complex deep learning models in various fields [15]. This is especially true in the fields of drug discovery and systems pharmacology, where understanding why the model made specific decisions and predictions can not only validate and help interpret the results, but also generate new knowledge and hypotheses regarding the complex systems under study [21]. Here, we

developed a node and subgraph importance method to identify which nodes the model pays attention to when creating the embeddings and which nodes in the original networks cause the embeddings to cluster together. This resulted in the better understanding and interpretation of the novel representations that were extracted from the DL model. Using this approach, we showed that the models pay attention to similar nodes in order to cluster together compounds with similar MoA and were able to identify important signaling subgraphs that are characteristic of each cluster (*Figure 3.3*). For example, in the clusters enriched for mTOR inhibitors, even though mTOR as a node was not present in the input signaling networks of the cluster, deepSNEM was able to extract important subgraphs that are related to the mTOR signaling pathway.

5 Further Work

All of the findings presented are potential source for further work on the subject of elucidation a compound's Mechanism of Action (MoA) in a biological system by analyzing and comparing compounds' transcriptomic signatures.

As aforementioned the deepSNEM-GT-MI was selected to be used in this thesis' tasks, as we argued it was the most appropriate due to the nature of the problem. Although the results were satisfactory there is no evidence that another architecture (graph convolutions or graph2vec) couldn't have been equally effective. Thus, a lot of space for investigation exists regarding the architecture of the model.

Moreover, the modification of the embeddings' dimension combined with the t-SNE parameterization is an aspect of the problem it could be further investigated having as a prerequisite sufficient computing power.

Last but not least, it is noted in the section above, that even for the clusters enriched for the same MoA, the large number of unknown compounds could result in different cluster compositions. That potentially further signifies the importance of analyzing biological response from different points of view, e.g. genes, pathways, signaling networks.

Regardless the improvement of the model itself, the deepSNEM pipeline serves as proof of concept that compound-induced signaling networks can be analyzed on a massive scale, using deep learning and provide insight into the compound's effect. In a real-world application, deepSNEM would be used in combination with existing methods, utilizing transcriptomic data or pathway signatures, for a consensus-based assignment of compound perturbations into clusters that are enriched for specific MoA. Subsequently, deepSNEM could be used to identify which nodes and subgraphs mostly influenced the proposed cluster assignment, thus increasing its interpretability and help generate new hypotheses. We believe that our signaling network dataset and the proposed pipeline can help pave the way towards more studies that utilize the inherent knowledge of the changes in the signaling cascade of a system to better elucidate a compound's mechanism of action.

6 References

[1] Iman Tavassoly, Joseph Goldfarb, and Ravi Iyengar. “Systems biology primer: the basic methods and approaches”. In: Essays in Biochemistry 62.4 (Oct. 2018), pp. 487-500. DOI: 10.1042/ebc20180003.

[2] B. Alberts et al. Molecular Biology of the Cell 4th Edition: International Student Edition. Routledge, 2002. ISBN: 9780815332886. URL: https://books.google.gr/books?id=ozigkQEACAAJ.

[3] David S. Latchman. “Transcription factors: An overview”. In: The International Journal of Biochemistry & Cell Biology 29.12 (1997), pages 1305-1312,

[4] J. Dedrick Jordan, Emmanuel M Landau, and Ravi Iyengar. “Signaling Networks”. In: Cell 103.2 (Oct. 2000), pp. 193-200. DOI: 10.1016/s0092-8674(00)00112-4.

[5] F. Jordan, T.-P. Nguyen, and W.-c. Liu. “Studying protein-protein interaction networks: a systems view on diseases”. In: Briefings in Functional Genomics 11.6 (Aug. 2012), pp. 497-504. DOI: 10.1093/bfgp/els035.

[6] Valeria Fionda. “Networks in Biology”. In: Encyclopedia of Bioinformatics and Computational Biology. Elsevier, 2019, pp. 915-921. DOI: 10.1016/b978-0-12-809633-8.20420-2.

[7] Nahid Safari-Alighiarloo, Mohammad Taghizadeh, Mostafa Rezaei-Tavirani, Bahram Goliaei and Ali Asghar Peyvandi. Gastroenterol Hepatol Bed Bench. 7.1 (2014), pages 17-31, Protein-protein interaction networks (PPI) and complex diseases.

[8] Ian Goodfe4llow, Yoshua Bengio, and Aaron Courville. Deep Learning. http://www.deeplearningbook.org. MIT Press, 2016.

[9] Luz Garcia-Alonso et al. “Benchmark and integration of resources for the estimation of human transcription factor activities”. In: Genome Research 29.8 (July 2019), pp. 1363-1375. DOI: 10.1101/gr.240663.118.

[10] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. 2016. arXiv: 1609.02907 [cs.LG].

[11] Sorin Grigorescu et al. “A survey of deep learning techniques for autonomous driving”. In: Journal of Field Robotics 37.3 (Apr. 2020), pp. 362-386. DOI: 10.1002/rob.21918.

[12] Kaiming He et al. Deep Residual Learning for Image Recognition. 2015. arXiv: 1512.03385 [cs.CV].

[13] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems. 2017. arXiv: 1706.03762 [cs.CL].

[14] Aravind Subramanian et al. “A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles”. In: Cell 171.6 (Nov. 2017), 1437-1452.e17. DOI: 10.1016/j.cell.2017.10.049.

[15] Chakraborty, Supriyo, et al. "Interpretability of deep learning models: A survey of results." 2017 IEEE smartworld, ubiquitous intelligence & computing, advanced & trusted computed, scalable

computing & communications, cloud & big data computing, Internet of people and smart city innovation (smartworld/SCALCOM/UIC/ATC/CBDcom/IOP/SCI). IEEE, 2017.

[16] Anika Liu et al. “From expression footprints to causal pathways: contextualizing large signaling networks with CARNIVAL”. In: npj Systems Biology and Applications (2019). DOI: 10.1038/s41540-019-0118-z.

[17] Ian W. Taylor and Jeffrey L.Wrana. “Protein interaction networks in medicine and disease”. In: PROTEOMICS 12.10 (May 2012), pp. 1706-1716. DOI: 10.1002/pmic.201100594.

[18] Türei, Dénes, Tamás Korcsmáros, and Julio Saez-Rodriguez. "OmniPath: guidelines and gateway for literature-curated signaling pathway resources." Nature methods 13.12 (2016): 966-967. DOI: 10.1038/nmeth.4077

[19] Michael Heinzinger et al. “Modeling aspects of the language of life through transfer-learning protein sequences”. In: BMC Bioinformatics 20.1 (Dec. 2019). DOI: 10.1186/s12859-019-3220-8.

[20] Matthew E. Peters et al. Deep contextualized word representations. 2018. arXiv: 1802.05365 [cs.CL].

[21] Jiménez-Luna, José, Francesca Grisoni, and Gisbert Schneider. "Drug discovery with explainable artificial intelligence." Nature Machine Intelligence 2.10 (2020): 573-584. arXiv: 2007.00523 [cs.CL]

[22] Lejla Batina et al. “Mutual Information Analysis: a Comprehensive Study”. In: Journal of Cryptology 24.2 (2011), pp. 269-291.ISSN: 1432-1378. DOI: 10. 1007/s00145-010-9084-8.

[23] Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. 1st ed. Adaptive Computation and Machine Learning. The MIT Press, 2012.

[24] Petar Veličković et al. Deep Graph Infomax. 2018. arXiv: 1809.10341 [stat.ML].

[25] R Devon Hjelm et al. Learning deep representations by mutual information estimation and maximization. 2018. arXiv: 1808.06670 [stat.ML].

[26] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. 2018. arXiv: 1807.03748 [cs.LG].

[27] Fan-Yun Sun et al. InfoGraph: Unsupervised and Semi-supervised Graph Level Representation Learning via Mutual Information Maximization. 2020. arXiv: 1908.01000 [cs.LG].

[28] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. 2016. arXiv: 1606.00709 [stat.ML].

[29] Magzhan, Kairanbay, and Hajar Mat Jani. "A review and evaluations of shortest path algorithms." International journal of scientific & technology research 2.6 (2013): 99-104.

[30] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-Attention Graph Pooling. 2019. arXiv: 1904.08082 [cs.LG].

[31] Yun, Seongjun, et al. "Graph transformer networks." Advances in Neural Information Processing Systems 32 (2019): 11983-11993.

[32] Vinyals, Oriol, Samy Bengio, and Manjunath Kudlur. "Order matters: Sequence to sequence for sets." arXiv preprint arXiv: 1511.06391 (2015).

[33] Kokhlikyan, Narine, et al. "Captum: A unified and generic model interpretability library for pytorch." arXiv preprint arXiv:2009.07896 (2020).

[34] Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps." arXiv preprint arXiv:1312.6034 (2013).

[35] Corsello, Steven M., et al. "The Drug Repurposing Hub: a next-generation drug library and information resource." Nature medicine 23.4 (2017): 405-408.

[36] Wang, Chun-I., et al. "mTOR regulates proteasomal degradation and Dp1/E2F1-mediated transcription of KPNA2 in lung cancer cells." Oncotarget 7.18 (2016): 25432.

[37] Zhang, Yinan, and Brendan D. Manning. "mTORC1 signaling activates NRF1 to increase cellular proteasome levels." Cell cycle 14.13 (2015): 2011-2017.

[38] Jiang, Chao, et al. "CSNK2A1 promotes gastric cancer invasion through the PI3K-Akt-mTOR signaling pathway." Cancer Management and Research 11 (2019): 10135.

[39] Gordon, Bradley S., et al. "RhoA modulates signaling through the mechanistic target of rapamycin complex 1 (mTORC1) in mammalian cells." Cellular signalling 26.3 (2014): 461-467.

[40] Jewell, Jenna L., et al. "GPCR signaling inhibits mTORC1 via PKA phosphorylation of Raptor." Elife 8 (2019): e43038.

[41] Gao, Shengqiao, et al. "Modeling drug mechanism of action with large scale gene-expression profiles using GPAR, an artificial intelligence platform." BMC bioinformatics 22.1 (2021): 1-13.

[42] Menden, Michael P., et al. "Community assessment to advance computational prediction of cancer drug combinations in a pharmacogenomic screen." Nature communications 10.1 (2019): 1-17.

[43] Douglass, Eugene F., et al. "A community challenge for pancancer drug mechanism of action inference from perturbational profile data." (2020).

[44] Iorio, Francesco, et al. "A landscape of pharmacogenomic interactions in cancer." Cell 166.3 (2016): 740-754.

[45] Napolitano, Francesco, et al. "Drug-set enrichment analysis: a novel tool to investigate drug mode of action." Bioinformatics 32.2 (2016): 235-241.

7 Appendix

Some of the data is available at the NTUA's System Biology Lab Github page (https://github.com/BioSysLab/deepSNEM), but due to the restrictive size of others such as the signaling network graphs, the option of uploading them to our repository was not available

The code for the DeepSNEM model and its implementation, as well as the code regarding the clustering analysis and subgraph importance can be found at the same repository.