Incorporating biological knowledge into machine learning: application to gene expression data

Maja Trębacz
King's College

UNIVERSITY OF CAMBRIDGE

A dissertation submitted to the University of Cambridge in partial fulfilment of the requirements for the degree of Master of Philosophy in Advanced Computer Science

University of Cambridge
Computer Laboratory
William Gates Building
15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom

Email: mt675@cam.ac.uk

July 8, 2020

Declaration

I Maja Trębacz of King's College, being a candidate for the M.Phil in Advanced Computer Science, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

Signed:

Date:

Acknowledgements

First of all, I would like to thank my supervisors Zohreh Shams and Mateja Jamnik for their constant support and effort to put me in the best conditions to succeed. Moreover, I would like to thank the rest of the Cancer AI group (Paul Scherer, Nikola Simidjievski, Helena Andres, Pietro Liò) for offering me help whenever needed and brainstorming ideas.

I would also like to thank Robert Hoehndorf and other authors of ontology embeddings for providing me with the vector representations they obtained, as well as for answering all my questions. Moreover, I would like to acknowledge Lara Jerman who worked obtaining the embeddings, as an intern in the Computer Laboratory.

Finally, I would like to thank my friends and family who supported me along the way and made this master's degree a great experience.

Abstract

Incorporating biological knowledge into machine learning: application to gene expression data

The recent technical developments in cancer research often include applying machine learning techniques to gene expression data. Stratifying cancer patients based on the information contained in their gene expression levels allows to improve and personalise diagnosis, survival analysis and treatment planning. However, such data is extremely highly dimensional as it contains expression values for over 20’000 human genes per patient. Simultaneously, the number of samples (patients) in the datasets is very low, rarely exceeding few dozens. Hence, using gene expression data for predicting patient characteristics with statistical approaches can be challenging.

This project is enhancing the standard machine learning methods with the decades worth of biological knowledge about genes. It aims to verify if injecting biological knowledge into the system can improve performance of the techniques for stratifying cancer patients. To achieve this, it uses meaningful embedding for genes from ontologies and integrates them into deep learning pipeline. Similarity measure between the features (genes) is used to overcome the problems of high-dimensional low-sample data. The proposed approaches are twofold.

We first use the ontology embedding for the genes as a similarity measure for knowledge-informed feature selection. Given reference genes known as an important for prediction, one can use the ontology-based similarity to select a larger set of genes performing better on the patient classification task.

Second, we incorporate ontology-based knowledge about genes into deep learning model as a structural inductive bias. By using graph convolutional networks, we exploit the information about gene similarity to impose a biological bias on the model, in the same way as convolutional network impose a spatial bias on the image.

The results obtained by the proposed approaches demonstrate that incorporating the biological knowledge from ontologies into the machine learning pipeline can improve predictive abilities in the cancer classification task on the high-dimensional low-sample data of gene expression.

Word count: 14'612[1]

[1] This word count was computed using TeXcount (`https://app.uio.no/ifi/texcount/`), counting both words in text as well as captions, headers etc.

Contents

1 Introduction **1**

1.1 Contributions . . . 3

1.2 Project structure . . . 4

2 Background and related work **5**

2.1 Relevant machine learning background . . . 5

2.1.1 Random Forest . . . 6

2.1.2 Neural Networks . . . 6

2.1.3 Graph Neural Networks . . . 8

2.2 Understanding gene expression data . . . 9

2.3 Gene Ontology . . . 11

2.4 Ontology embeddings . . . 12

2.4.1 Onto2vec . . . 12

2.4.2 OPA2Vec . . . 13

2.4.3 EL . . . 14

2.4.4 DL2vec . . . 14

2.5 Related work on imposing inductive biases in biomedicine . . . 15

2.6 Data and cancer patient classification tasks . . . 17

3 Feature selection using ontology-based gene similarity **19**

3.1 Data-driven gene selection via recursive feature elimination . . 20

3.1.1 Method . . . 21

3.1.2 Results . . . 22

3.2 Hybrid ontology-based gene selection . . . 25

3.3 Evaluation of the ontology based gene selection on clinical tasks 26

3.3.1 Experimental setup . . . 27

3.3.2 Results . . . 28

3.4 Evaluation of the ontology-based gene selection on the single gene inference task . . . 33

3.4.1 Experimental setup . . . 34

3.4.2 Results 34

3.5 Summary 36

4 Imposing structural inductive bias capturing gene similarity 39

4.1 Methods 40

4.1.1 Graphs used for inductive bias 41

4.1.2 Graph convolution operation 44

4.1.3 Node embedding methods 46

4.2 Evaluation 48

4.2.1 Experimental setup 49

4.2.2 Results when using all 24'368 genes 50

4.2.3 Results when using pre-selected 1000 genes 54

4.3 Combining GCN with feature selection 55

4.4 Identifying high-weight genes for interpretability 57

4.5 Summary 60

5 Conclusions 63

5.1 Future work 64

List of Figures

1.1 Graphical abstract of the project. The project incorporates ontology-based knowledge about genes into a machine learning system that uses gene expression data to classify patients (e.g. into breast-cancer subtype). The contributed methods are: feature selection based on the ontology embeddings (Chapter 3) and gene expression convolutions using GCN on a prior-knowledge graph (Chapter 4). 2

2.1 Structure of a basic artificial neuron. 7
2.2 Graph convolutional operator. Diagram from [1]. 9
2.3 Conceptual view of gene expression data. The raw data from microarray experiments is quantified and transformed into matrices of gene expression levels [2]. Typically, the number of columns/genes is high (above 20'000) and number of rows/samples low (rarely exceeding few dozens). 10
2.4 Fragment of the Gene Ontology (visualised with QuickGO [3]) 11
2.5 Onto2Vec workflow (simplified). Diagram based on Smaili et al. [4]. 13

3.1 Genes ranked by their feature importance scores in the PAM50. The subset of these 10 genes was selected via recursive feature elimination. 22
3.2 Gene expression values of ESR1, FOXA1, GATA3, CEP55 plotted against a PAM50 breast cancer subtype class. 24
3.3 Framework of the ontology-based feature selection. Given a reference gene we use the ontology embedding to select K most similar genes. The selected subset of features is used in a Machine Learning model for patient classification task (e.g., PAM50 breast cancer subtype classification). 26

3.4 Results of the ontology-based feature selection on the patient classification tasks (PAM50, ER, iC10, DR). Each plot considers a different task and shows performance of RF trained on the varying number of genes either selected using the proposed ontology-based method (with EL embeddings [5]) or selected at random. The black lines show the results of the pre-selected 1000 genes. 30

3.5 Results of the ontology-based feature selection on the patient classification tasks (PAM50, ER, iC10, DR). Each plot considers a different task and shows performance of RF trained on the varying number of genes selected using the proposed ontology-based method with several embedding types (see Table 3.2) or selected at random. The black lines show the results of the pre-selected 1000 genes. 32

3.6 AUC scores of predicting if a single gene value is over/under-expressed given some genes closest to it in the ontology embedding space. The performance is compared to the random set of genes. For each plot, the target gene varies. The scores are plotted against the number of genes used for prediction. Three trials are used and error bars show standard error. . . . 35

3.7 Distribution of AUC improvements over genes from using the ontology-based gene selection rather than random. The mass on the right from the dashed line represents the genes for which the SGI performance was improved. 36

4.1 Overview of the Graph Convolutional Network applied to gene expression data. 41

4.2 Graph generated with EL embeddings, showing the two-hop neighbourhood of the ESR1 gene. The graph was generated with K=3. Note that for some of the edges we have more neighbours than K due to reciprocal connections. 42

4.3 Considered methods of obtaining input node embedding. . . . 46

4.4 Accuracy obtained on the PAM50 and iC10 breast cancer subtype classification tasks. The reported accuracy is an average over 5 trials (with different stratified split of data) and error bars represent standard error. 55

4.5 Accuracy obtained on the ER status classification task. The reported accuracy is an average over 5 trials (with different stratified split of data) and error bars represent standard error. 56

4.6 t-SNE visualisation of the ontology embedding space. The coloured points mark the top 30 high-weight genes in the PAM50 classification task, for each of the breast-cancer subtypes. 59

List of Tables

3.1 Comparison of the predictive performance (accuracy±std) of the model trained on varying subsets of genes selected with recursive feature elimination (Random Forest model, 5-fold cross validation). 23

3.2 Details of used ontology embeddings. 27

3.3 Comparison of the predictive performance on PAM50 (accuracy±std) task of the model trained on set of 1000, 100, or 10 genes selected with the proposed feature selection methodology, using different gene embeddings. The results are compared to the baselines of randomly selected set of genes (of the same size), set of genes including ESR1 and randomly selected genes, and a pre-selected set of 1000 genes from [6]. 28

3.4 Comparison of the predictive performance on PAM50 task (accuracy±std) of the model trained on set of 1000, 100, or 10 genes selected with the proposed feature selection methodology, but without including the reference gene (ESR1) in the set. 29

4.1 Comparison of graph sizes including number of nodes (and size of intersection of node names with genes occurring in the the METABRIC (MB) dataset [6]), and number of edges. In case of the graphs generated from ontologies the numbers of edges are given for a certain K parameters specifying number of nearest neighbours used to generate graph edges. 44

4.2 Five chosen hyperparameter settings for MLP. 49

4.3 Five chosen hyperparameter settings for GCN. 50

4.4 Performance comparison of the methods on the PAM50, ER and iC10 patient classification tasks (accuracy±std), using all genes as input features (in case of GCN, intersection of genes present in the dataset and graph). Bold font marks the cases when the model obtained statistically significantly better results than baselines. 50
4.5 Performance comparison of the graphs directing GCN structure on the PAM50 classification task (accuracy ± std). The GCN used expression values as the nodes input, without node embedding mechanism (A in the Figure 4.3). 52
4.6 Performance comparison of the node embeddings methods (see Figure 4.3) on the PAM50 classification task (accuracy ± std). The GCN in all cases operates on the graph produced from prior knowledge from ontologies. 53
4.7 Performance comparison of the methods on the ER, iC10 and DR classification task, using all genes as input features (in case of GCN, the intersection of genes present in the dataset and graph). Trained either on a training sample of 100 examples or 1500 examples. 54
4.8 High weight genes obtained by analyzing the connections in last linear layer of GCN for predicting the PAM50 breast cancer subtype. Bold font signifies genes present in the standard 50-gene subtype predictor [7]. 58

Chapter 1

Introduction

Gene expression data is commonly used within statistical learning in cancer research. It opened up a range of possibilities for analysing the genomic profiles of patients. Classifying cancer patients based on their gene expressions allows to personalise diagnosis, survival analysis and treatment planning.

Within typical machine learning contexts, a sample gene expression profile is represented as an unstructured feature vector of continuous values where each value corresponds to the expression level of a particular gene in the sample. Such data is extremely highly dimensional as it contains expression values for over 20'000 human genes per patient. At the same time the available datasets from the studies have a very low number of samples (patients), rarely exceeding a few dozen. Hence, using gene expression data for predicting patient characteristics using statistical approaches can be challenging.

The aim of the project is to incorporate prior biological knowledge about genes from ontologies (structured knowledge representation, Section 2.3) into the machine learning system classifying patients based on their gene expression data. The project uses ontology embeddings (Section 2.4) that capture the semantic similarity between the genes (features). The measure of similarity between genes is used to sparsify the features or network connections, and therefore overcome the problems of high-dimensional low-sample data.

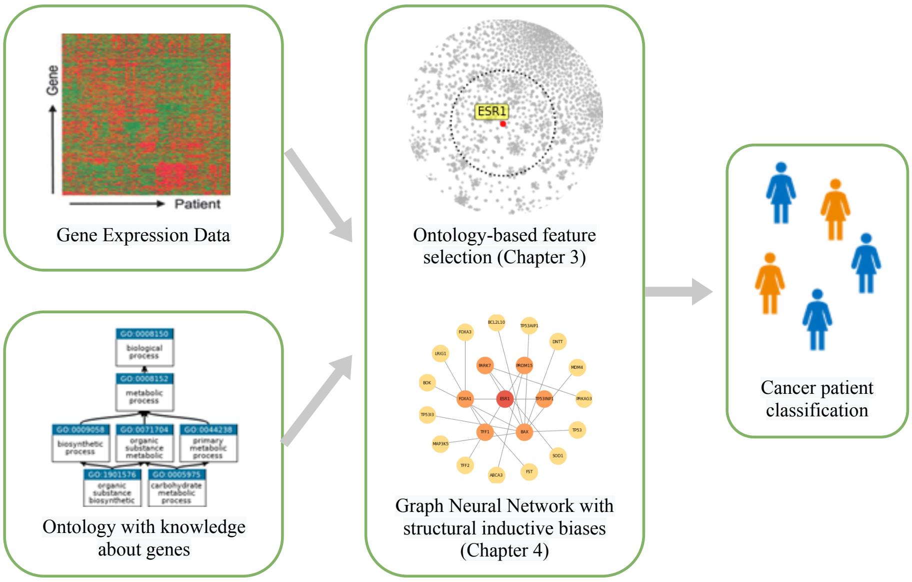
Gene
Patient
Gene Expression Data
GO:0008150
biological process
GO:0008152
metabolic process
GO:0009058
biosynthetic process
GO:0071704
organic substance metabolic
GO:0044238
primary metabolic process
GO:1901576
organic substance biosynthetic
GO:0005975
carbohydrate metabolic process
Ontology with knowledge about genes
ESR1
Ontology-based feature selection (Chapter 3)
BCL2L10
TP53AIP1
FOXA3
DNTT
LRIG1
PARK7
PRDM15
MDM4
BOK
FOXA1
ESR1
TP53INP1
PRKAG3
TP53I3
TFF1
BAX
TP53
MAP3K5
SOD1
TFF2
ABCA3
FST
Graph Neural Network with structural inductive biases (Chapter 4)
Cancer patient classification

Figure 1.1: Graphical abstract of the project. The project incorporates ontology-based knowledge about genes into a machine learning system that uses gene expression data to classify patients (e.g. into breast-cancer subtype). The contributed methods are: feature selection based on the ontology embeddings (Chapter 3) and gene expression convolutions using GCN on a prior-knowledge graph (Chapter 4).

Figure 1.1 gives an overview of the key methods of the project. The first part of the project (Chapter 3) focuses on using ontology embeddings to select biologically relevant subset of genes that have high predictive ability and biological relevance, avoiding processing all of the genes (>20'000). The further directions use graph topology capturing similarity of genes in ontology to inject structural inductive bias on the neural network (Chapter 4). The work uses graph convolutional network [8] to impose bias similar in nature to the spatial bias imposed by convolutional networks on images [9]. Such an approach can improve predictive performance on the highly-dimension data, simultaneously retaining high transparency in identifying driver genes.

The methods proposed have broad applicability and can be used on biomedical datasets consisting of various cancer classification tasks spanning multiple tissues and tumour types. In this project, the approach will be exemplified on the breast-cancer classification tasks from the gene expression data of the METABRIC study [6].

1.1 Contributions

The main contributions of this project are as follows:

1. Development of a novel **ontology-based feature selection** pipeline that exploits the semantic similarity of genes derived from ontology embeddings. The results have shown that such methodology outperforms the random baseline or pre-selected set of genes. See Chapter 3.

2. First work to use ontology embeddings of genes to impose **structural inductive bias** on the deep learning model by using **graph convolutional network (GCN)**. The method uses the ontology embeddings as a similarity measure to generate relations between the genes. See Chapter 4. The contributions in this method include:

   - Method of generating topology for the graph used for inductive bias, that captures the semantic similarity of the genes based on ontology embeddings (Section 4.1.1).

   - Proposal and implementation of novel node embeddings that integrate prior knowledge with the expression data (Section 4.1.3).

   - Demonstration of effectiveness of the proposed approach on the breast-cancer patient classification tasks (Section 4.2).

   - Combining GCN with the proposed feature selection method (Section 4.3).

3. Method for identifying high-weight genes influencing the prediction. Such genes were also visualised in the latent gene embedding space. This method shows transparency and explainability of the model. See Section 4.4.

4. Devised recommendation about which method should be applied for patient classification depending on the dataset size and dimensionality. See Section 4.3.

1.2 Project structure

The structure of this dissertation is outlined as follows:

- Chapter 2 provides background required for the project. It covers machine learning methods, as well as a brief description of gene expression data and ontologies. The chapter also discusses the related work and presents the dataset and tasks used in the project.
- Chapter 3 presents and evaluates our proposed method of ontology-based feature selection.
- Chapter 4 presents and evaluates our proposed method using GCN for imposing structured inductive bias capturing ontology-based semantic similarity of the genes.
- Chapter 5 summarises our research and suggests possible directions for future work.

Chapter 2

Background and related work

This chapter offers a relevant background for understanding the methods introduced in the project. It firstly briefly introduces machine learning models of random forest, neural networks and graph neural networks. To account for the multidisciplinary nature of the project, Section 2.2 introduces basic biological concepts about gene expression and Section 2.3 introduces the concept of ontologies, in particular the Gene Ontology [10].

The further sections cover the related work. Section 2.4 presents the four kinds of vector representations of genes derived from ontology, that will be used in the project. Section 2.5 provides a brief literature review on the existing approaches of imposing inductive biases from the biological knowledge about genes.

Finally, Section 2.6 presents the main dataset and classification tasks considered in the project.

2.1 Relevant machine learning background

This section introduces the machine learning models used throughout the dissertation.

2.1.1 Random Forest

Random forest (RF) [11] is an ensemble machine learning model that operates by constructing multiple decision trees. Each of the individual trees classifies the samples and the RF outputs the class that was the most common result.

A decision tree training process forms a tree where internal nodes represent tests on data and leaves represent the output classes. The learning process of each decision tree is performed inductively from the bootstrapped sample of training data. The most well-known decision tree learning algorithms are CART [12] and ID3 [13]. Decision trees are formed recursively. The rules at the nodes are selected to express the best split to differentiate samples. The CART analysis typically uses the Gini criterion [12], that intuitively measures how often a random element from the set would be incorrectly labelled if its label was selected at random according to the distribution of classes. Once a rule is selected and creates branching, the same process is recursively applied to each child node until a pre-set stopping rule is met.

RF is powerful, transparent and fast to train, provided that the number of estimator trees is not high [14]. They tend to perform well on low-sample training data [15, 16]. By fitting several decision trees on bootstrapped subsamples of the dataset, the method improves the predictive accuracy and controls over-fitting. However, the expressive power is lower than in neural networks. We will use it as a baseline in this project.

2.1.2 Neural Networks

Neural network is a machine learning model loosely inspired by the functioning of the human brain. They are computational graphs consisting of interconnected artificial neurons that perform operations on the input data [17]. Mathematically, they can be explained as universal function approximators that are trained to determine a function $f(x)$ that takes an input x and returns a predicted output $\hat{y}$ approximating the desired output y.

A neural network is built using a basic processing units called neurons. Similarly to a biological neuron, a neuron receives inputs from other sources, combines them, applies non-linear operation, and outputs a result [18]. Figure 2.1 shows the structure of an artificial neuron. The neuron takes an input vector of features $\vec{x} = [x_1 x_2 ... x_n]^T$, computes a weighted sum (with weights $\vec{w} = [w_1 w_2 ... w_n]^T$ learnt during the training) and applies an activation function σ to the result. The activation function applied by an artificial neuron is generally non-linear. This project uses mainly the rectified linear unit (ReLU) defined as $\sigma(x) = \max(0, x)$.

Multi-layer perceptron (MLP) [17] is a type of feed-forward neural network composed of multiple layers of neurons. It often uses fully-connected layer in which each neuron from a previous layer is connected to all of the neurons in the following layer. If the network has more than one intermediate (hidden) layer, then it is called a deep neural network.

According to Cybenko's *universal approximation theorem* [19], a neural network with one hidden layer with a finite number of neurons and sigmoidal activations can approximate any continuous real function defined on a closed and bound subset of real numbers. This highlights the powerful computational capabilities of neural models. Although, the deeper networks provide no further theoretical power, in practice, having more layers is often beneficial for network simplification and imposing hierarchical input processing.

Neural networks are typically trained via gradient descent algorithm, consisting of a sequence of small steps that adjust the parameters, starting from initially randomised values. An MLP learns how to approximate a function

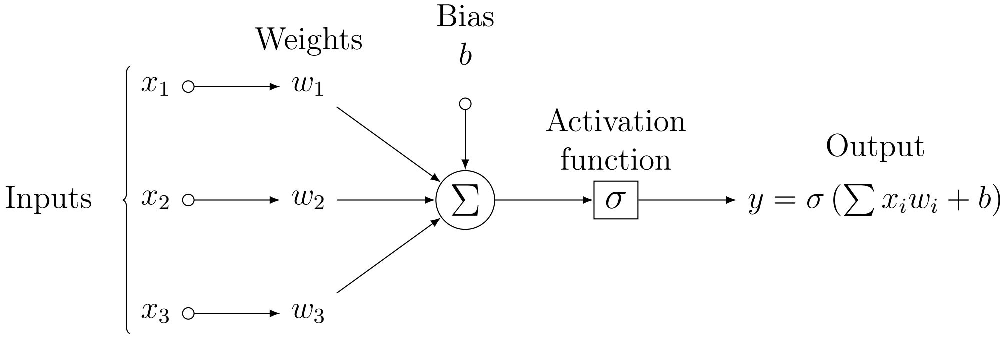

Figure 2.1: Structure of a basic artificial neuron.

by first producing a prediction $\hat{y}$ through a forward pass. The algorithm minimises the loss function that expresses a difference between the prediction and the expected output y. Gradient descent updates the network parameters in the direction of the steepest decrease in loss function value, that is the gradient of the error function with respect to the neural network's weights [17]. The gradients are calculated through the back-propagation method.

2.1.3 Graph Neural Networks

Despite its expressive power, the MLP was designed to process an unstructured data provided as a vector of features [1]. However, many inputs in the real world domains are best expressed with structure. The recent advances in the field of deep learning have made great progress in processing such data, by incorporating relational inductive biases in the architectures [20]. The application to computer vision has given rise to convolutional neural networks (CNNs) [21] that process a grid structured image data. The natural language processing [22] and speech recognition [23] benefit from recurrent neural networks (RNNs) [24, 25] exploiting the sequential nature of the input.

Part of this project considers imposing structural inductive biases on data by accompanying it with graph structure expressing prior knowledge. For this purpose, it uses Graph Neural Network (GNN) that is a type of Neural Network which gained high interest in recent years [20].

The model of GNN was originally presented [26, 27] as an extension of the recursive neural networks (RNN) that can operate on a more general class of inputs expressed with graphs. The GNN leverages the graph structure directly for directing the information flow between the nodes while learning intermediate representations. It can be used to classify each of the nodes or the entire graph.

Each of the GNN layers updates the node states based on the states of the direct neighbours (see Figure 2.2). After k iterations, the node state contains the information of its k-hop neighbourhood in the graph.

This project mainly uses the graph convolutional network (GCN) proposed

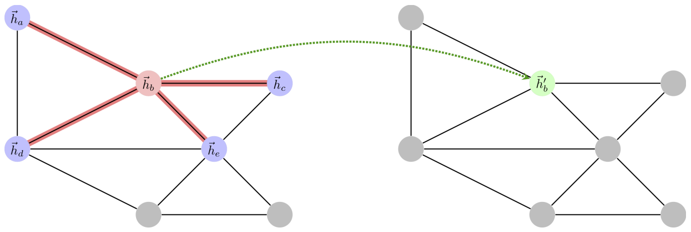

Figure 2.2: Graph convolutional operator. Diagram from [1].

by Kipf and Welling [8]. GCN is a scalable simplified graph neural network model, which achieved state-of-the-art classification results on several benchmark graph datasets. The exact graph convolutional operation is defined in Section 4.1.2. Other variants include a graph attention network (GAT) [28] which incorporates the attention mechanism into the propagation step or gated graph neural network (GGNN) [29] which includes a memory unit on the nodes. In general, all such architectures could be expressed in terms of a message-passing neural networks [30].

2.2 Understanding gene expression data

Gene is a sequence of nucleotides in DNA or RNA that contains the instructions to produce calls, which are the building blocks of the living organisms. Gene expression is the process of transforming the genetic information in the DNA into functional products such as proteins, RNA or other molecules which then dictate the cell function [31]. Thanks to the technology known as expression microarrays, it is possible to study nowadays in a single experiment the behaviour and expression of all the genes of an organism [32].

DNA microarray is a collection of microscopic spots attached to a solid surface. Microarrays quantify gene expression utilising fluorescence intensity that is captured as an image. Then, the images are transformed into numbers that represent expression levels of the genes in the sample [33].

This project considers the matrix representation of a multi-patient dataset, containing already transformed gene transcription levels (see Figure 2.3). Such matrix has thousands of columns representing genes and a much lower number of rows representing samples, i.e., various patients or various tissues. Each cell contains a real value characterising the expression level of a specific gene for the specific patient.

Recent years have seen a rapid growth of interest in an analysis of the gene expression data, which gained wide application in bioinformatics research [6, 34, 33]. DNA microarrays are useful in the identification and classification of various diseases like cancer, improving and personalising diagnosis and treatment of cancer patients, and in the discovery or development of drugs [35]. During an analysis of the gene expression levels, biologists may, for example, seek explanations about how the genes of patients relate to the types of cancers they had. This allows for better stratification of the cancer patients and also to identify novel cancer biomarkers, which are genes strongly correlated to a particular type of cancer.

The data generated by the microarray experiments consists of tens of thousands of variables (genes). Due to the high cost of acquiring data from multiple patients, most of the datasets with gene expressions have a low number of samples [9]. This curse of dimensionality combined with low-sample datasets poses challenges in applying machine learning approaches, as it is difficult to avoid overfitting [36]. Other difficulties include the presence of the background noise and need for appropriate normalisation of the data [37].

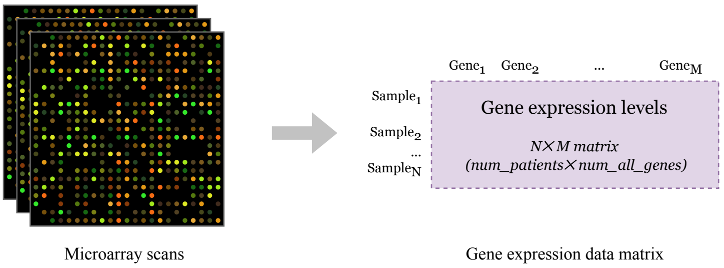

Figure 2.3: Conceptual view of gene expression data. The raw data from microarray experiments is quantified and transformed into matrices of gene expression levels [2]. Typically, the number of columns/genes is high (above 20'000) and number of rows/samples low (rarely exceeding few dozens).

2.3 Gene Ontology

Ontology is a formal and explicit representation of semantic knowledge. It contains definitions of the classes and axioms that comprise the overall theory. Ontological representations allow for collecting and unifying the knowledge in a given domain. It also facilitates validation of semantic relationships and derivation of conclusions from known facts (i.e., reasoning) [38]. Ontology could be visualised in a graph form and therefore it is related to the concept of a knowledge graph. However, there are certain differences as the ontologies model the general properties of classes, while, knowledge graphs include information about specific individuals in the domain [39]. Moreover, ontology axioms can capture more complex relationships between entities than knowledge graphs (including the Description Logic [40] operators of quantifiers, conjunction, disjunction and negation).

Ontologies are widely used to represent biological knowledge. The Gene Ontology (GO) [10] is a major bioinformatics project that aims to unify the representation of genes and their functions across all species (see Figure 2.4). It

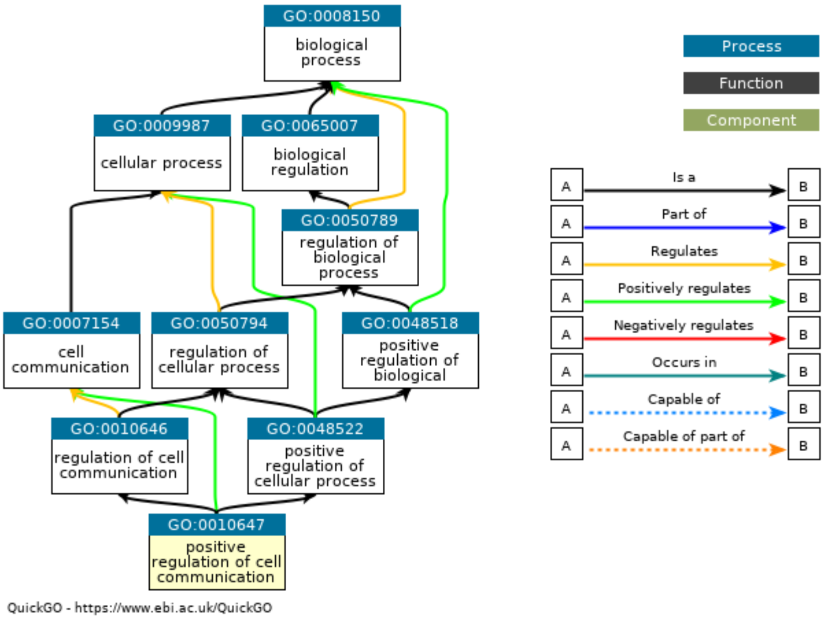

Figure 2.4: Fragment of the Gene Ontology (visualised with QuickGO [3])

describes the biological knowledge about genes with respect to three aspects that are root ontology terms (cellular component, biological process, and molecular function). All other GO classes (terms) are a further stratification of these aspects. They are composed of a definition, a label, a unique identifier, and several other elements including natural language descriptions. The genes are not terms in the GO, but they are connected to them via annotations [41]. The annotations are created based on observations and inferences from biological experiments.

The structure and information expressed in ontologies and their annotations make them essential in supporting biomedical research. They also prove useful for developing machine learning systems operating on biological entities [42, 43, 44]. In particular, this project uses prior knowledge from ontologies for a semantic similarity measure between the genes.

2.4 Ontology embeddings

The major part of the project uses ontology embeddings of the genes. Ontology embeddings are structure-preserving maps from ontologies into vector spaces [42]. This section introduces a series of methods that produce such feature vectors for biological entities from an ontology. The embeddings capture the semantic similarity between the genes.

2.4.1 Onto2vec

Onto2Vec [4] is a method of learning vector-space of terms in ontologies, and the biological entities (e.g., genes) annotated with these terms.

The process, shown in Figure 2.5, begins by collecting data from ontology-based annotations, axioms and ontology structures into a large corpus of “sentences”. Each axiom will constitute a sentence, and additionally, an ontology reasoner infers new logical axioms from the asserted ones. The words in the sentences represent ontology terms, biological entities (genes) and relations between them.

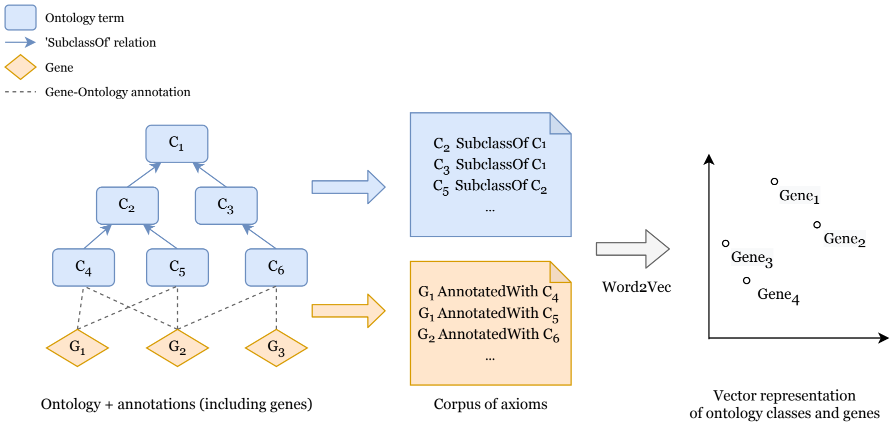

Figure 2.5: Onto2Vec workflow (simplified). Diagram based on Smaili et al. [4].

Given a large corpus of sentences expressing relations between classes and entities, they use Word2Vec skip-gram model [45] to learn the vector representation for each word. The vector for each word in the vocabulary (and therefore of an ontology term or gene) is predictive of words occurring within a context window. Intuitively, two entities will obtain similar embedding if they often co-occur with similar other terms.

2.4.2 OPA2Vec

OPA2Vec (Ontologies Plus Annotations to Vectors) [46] is an extension of Onto2Vec which combines both formal and informal content of ontologies. The biological ontologies often include descriptions in natural language that have valuable information. The OPA2vec vectors are generated not only from the formal axioms but also from these descriptions and meta-data. The learning process uses a Word2Vec model that was earlier pre-trained on PubMed abstracts. This allows for transfer learning of biological vocabulary.

The authors generate the embeddings from either just the Gene Ontology (GO) [10] or by merging three related ontologies: Gene Ontology [10], UBERON anatomy ontology [47], and Mammalian Phenotype Ontology (MP) [48]. In the case of merged ontologies, the set of embeddings is based on the union of all genes and their annotations. So the set includes genes that have annotations from one, two, or all three ontologies.

2.4.3 EL

EL embeddings [5] were designed to preserve the geometric structure in the ontology or other set of description logic statements. While Onto2vec and OPA2vec rely primarily on preserving syntactic properties from the co-occurrence of words in the knowledge corpus, the EL embeddings utilise prior knowledge about the semantics of operators during the search for an embedding function. To find such embeddings, Kulmanov et al. [5] define a collection of loss functions that preserve the semantics of the operators in the $\mathcal{EL}^{++}$ fragment of Description Logic [49] (e.g. inclusion or conjunction). The method was exemplified by generating the embeddings from the Gene Ontology (GO) [10] and testing on the prediction of protein-protein interactions.

2.4.4 DL2vec

DL2vec is a graph-based method for learning representations in biomedical ontologies. The process begins by converting ontology axioms in the Web Ontology Language (OWL) [50] format into a graph. Then it generates an embedding for each node and edge type by applying a process of a knowledge graph embedding [51]. The generation of knowledge graph embedding uses a series of random walks starting at nodes of the graph. The random walks generate a corpus from which Word2Vec [45] skip-gram model learns embeddings for nodes and edge labels. As an effect, node representation is predictive of the other nodes in its surrounding.

DL2Vec improves on the simpler Onto2vec and OPA2vec methods by an ability to encapsulate long-distance dependencies through the ontologies, capturing indirect associations between the terms that not directly co-occur. The method was exemplified by generating the embeddings from either just GO or from merged three related ontologies (GO, UBERON [47], and MP [48]), and then using embeddings for predicting gene-disease associations.

2.5 Related work on imposing inductive biases in biomedicine

This section gives a brief survey of previous work related to enhancing machine learning models with prior biological knowledge, that leverages decades of biology research [52]. Biological prior can help to automate feature importance and selection or act as a structural bias. For ML on the gene expression data, where the dimensionality is extremely high, the sample size is limited and the interpretability matters, incorporating prior knowledge can be particularly beneficial [43]. This lead to various publications integrating structured sources of domain knowledge with statistical models of gene expression data.

Several approaches use prior knowledge to aid model interpretation by applying "visible" machine learning approach [53]. This involves direct incorporation of the biological structure, such as modelling the deep neural structure from ontology [54] or biological networks [55]. Such models have a strict correspondence with biological entities, and therefore can be interpreted mechanistically and investigation of execution of the model can reveal the states of internal biological entities [53]. However, the current knowledge in genetics is very limited even with the recent advances in human cancer biology, so restricting processing to only known interactions and connections may hurt the predictive power of the model.

Development of novel deep learning models on graphs [56, 57, 8, 58] combined with biological motivations behind the network propagation model [59] has created an opportunity to leverage the prior knowledge without overly restricting the expressive power of the neural model. This has driven a number of publications that are closely related to this project.

Rhee et al. [60] proposed a combination of graph neural network (GNN) with a relation network [61]. The GNN uses a topology of protein-protein interaction (PPI) to perform convolutions on the gene expressions. The model was used for PAM50 subtype classification [7]. However, they use a set of 4,303 pre-selected genes as opposed to full set of over 20'000 we use in this project.

A similar method was proposed by Dutil et al. [9] who explore the usage of Graph Convolutional Network (GCN) with dropout and node embeddings to use the PPI networks [62] information on the single gene inference tasks (predicting the expression of the gene by using other genes). They explain that the bias imposed by GCN is similar in nature to the spatial bias imposed by convolutional networks on images. This work was further extended by Bertin et al. [63] who have studied and quantitatively evaluated the prior knowledge provided by multiple gene interaction graphs (PPI, gene regulatory, transcription regulation, gene co-expression etc.) comparing it to random or fully connected graphs. Instead of the graph neural network, they used simpler single-layer network approximating GNN, where the graphs were used as a masking measure on the connections in the fully-connected layer. Experiments were carried out both on single gene inference tasks and clinical tasks of cancer phenotype prediction. The works by Bertin et al. [63] and [64] concluded that expression dependencies can be predicted almost as well by using random networks as by using biological networks. One of the problems causing this result is that the degree of the nodes varies a lot and the predictions tend to be worse for the genes with a low number of neighbours (or no neighbours at worse). Such nodes with few neighbours get less signal for predictions and tend to get lower AUC. Crawford et al. [65] explored this observation and showed that when low-degree genes are removed, then the biological networks hold better predictors than random graphs.

This project takes a different approach and is the first one to use the ontology embeddings as a similarity measure between the genes for imposing the inductive bias for the patient classification tasks from gene expressions. The ontology-based feature selections (chapter 3) allows selecting a biologically relevant set of features. Graph convolution approach (chapter 4) extends the work of Dutil et al. [9] by proposing novel graphs automatically generated from ontology embeddings and a knowledge-based node embedding method. Automatic generation of the graphs expressing semantic similarity of genes allows overcoming the problems with varying degree of nodes described above.

2.6 Data and cancer patient classification tasks

The project demonstrates the approaches of integrating prior knowledge into machine learning on the gene expression data of cancer patients. Multiple international initiatives are collecting such genomic data, including The Cancer Genome Atlas (TCGA) [66, 67] and Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) [6]. They pose several cancer classification tasks which offer opportunity to improve disease stratification and develop personalised patient treatment [68].

In this project, the methods are mainly exemplified on the several breast-cancer classification tasks using data from the METABRIC cohort. The METABRIC project is a Canada–UK initiative that aims to stratify breast cancers based on heterogenous data types (genomic, transcriptomic, image). The dataset consists of 1,980 breast-cancer patients and it is one of the largest data sets with gene expressions.

In particular, the project uses the following patient classification tasks, treating the METABRIC grouping as the ground truth target variables:

- **ER:** two sub-types (ER+ and ER-) defined by immunohistochemistry (IHC) expression of estrogen receptor (ER), which is an important biomarkers in breast cancer [69]. Patients who are ER+ require different therapy that can reduce disease recurrence and mortality [70].

- **PAM50:** five intrinsic gene-expression sub-types [7] based on centroids from expressions of 50 classifier genes. The PAM50 subtypes have become a standardized model in diagnosis and it was shown to accurately predict recurrence and survival for patients [71].

- **iC10:** eleven integrative cluster sub-types [6]. The integrative cluster groups are defined by combining gene expression and DNA copy number alterations (CNAs), and are the most extensive molecular-based taxonomy of breast cancer [72]. Curtis et al. [6] firstly selected genes whose expression levels were significant in analising CNAs in proximal loci, and then used their expressions to define clustering into groups.

- **DR:** two *Distance Relapse* groups. The binary division signifies if cancer recurred in another organ after the initial treatment [73].

Experience with microarray data has repeatedly shown that normalisation is essential to ensure that expression estimates are more comparable across genes and samples [74]. Hence, the expression values are normalised by min-max scaling, before splitting the data into train-valid-test. For each of the genes, which represent the input features, range of gene expression values occurring across all the patients was scaled to $[0; 1]$. That is, the following transformation was applied to each component:

$$x_i' = \frac{x_i - min(x)}{max(x) - min(x)}$$

Where $min(x)$ is minimum expression of gene x across all samples. And x_i is the expression value of gene x for sample i.

Chapter 3

Feature selection using ontology-based gene similarity

The sequenced gene expression data is highly dimensional, with over 20'000 of human genes. This number of features combined with data scarcity can have a detrimental effect on some models. Limiting the number of features prior to building the model might be advisable. The common practice is to do feature (gene) selection with respect to the knowledge about the task and use it as an input to the trained models. A widely used example of such feature selection in analysing breast tumours is pre-selecting a set of 1000 genes as in Curtis et al. [6], where the genes are selected by considering the genes that are associated with DNA copy num-ber alterations (CNAs). Such subsets are designed to be task-specific and may not generalise well.

The chapter begins by introducing the commonly used data-driven method of feature selection and application of it on the gene expression data (Section 3.1). The further Sections 3.2 and 3.3 present the methodology and results of the experiments on a proposed ontology-driven selection of the human genes suitable for the input to the machine learning model. The method is hybrid in a sense that it combines prior knowledge of known cancer drivers with automatic selection based on semantic similarity of genes from ontology embeddings. Given at least one gene expression known as an important factor in the cancer-related classification task, one can select a set of (e.g., 1000) genes related with similar functions to the reference gene according to the ontology space. We show that they hold a better performance than using a set commonly used in literature or genes selected at random.

The methods developed are general and can be applied to any cancer characteristic classification task. The methodology is exemplified by studying the task of predicting PAM50 intrinsic breast cancer subtype and other clinical tasks (ER, iC10, DR) on a METABRIC cohort [6] (see Section 2.6).

Finally, Section 3.4 studies the approach on the Single Gene Inference tasks in line with the previous work experiments on biological network biased feature selection [75, 65].

3.1 Data-driven gene selection via recursive feature elimination

Due to the extremely high-dimensionality of the gene expression data, statistical analysis often requires finding a subset of most predictive genes for patient classification [76]. Although many unknown genes can have connections to a considered type of cancer, biologists prefer to focus on a small set of genes that highly correlate with the outcome before committing to expensive experiments with a larger set of genes.

Methods for automated feature selection are often data-driven. Such selection aims to maximise the predictive performance and achieves high accuracy scores [77]. However, there is no guarantee that the selected genes will have biological relevance to the considered task, especially since the importance scores on scarce data tend to have high variance and genes not relevant to the process may be enhanced by chance.

This section presents a data-driven recursive method of feature selection and uses it to analyse the data and set the baselines on predictive abilities of the subsets of genes. In recursive feature elimination, we train a Random

Forest on the full set of features and recursively eliminate half of the features with lower importance score. Such feature selection should provide an upper bound on the accuracy achieved by the subset of features of a given cardinality. Moreover, by identifying the genes that have the highest importance in prediction and analysing the performance of such subsets of genes, we can gain insight into the task.

3.1.1 Method

The feature importance score is a metric that reflects how significant is a feature for the prediction. The method for computing it depends on the ML model in use. In the case of a random forest, it can be derived during building the model. We can compute for each feature, how much it decreases the weighted impurity score (either Gini impurity [12] or information gain [13]) and average it among trees in the random forest. For other models we can use the permutation feature importance, that is applied to an already trained black-box model. It is based on observing how random re-shuffling of each predictor in turn (when keeping others intact) influences performance.

Although these metrics indicate the feature significance, taking the top-scoring features will often not result in a set of best predictors. When the variables are correlated, removing a single important variable may be compensated by using remaining variables. This effect causes the feature importance scores to be unstable. The gene expression data is often highly correlated and suffers from the problem of changing importance scores between the runs. Therefore, rather than using the importance scores directly, we use them to guide a recursive feature elimination process. The procedure is as follows:

- Train Random Forest on the set of features (starting with a full set of over 20k genes), using 5-fold cross-validation.
- Use the model's importance scores to rank the features. The values are averaged between the cross-validation folds.
- Discard half of the features with lower importance scores. Repeat.

To avoid overfitting bias when determining the performance of gene sets, the recursive feature elimination procedure is done on the subset of 80% of patients (cross-validating on 5 folds during the recursive feature elimination) and then performance is verified using the chosen features on the remaining test subset.

3.1.2 Results

To verify the predictive abilities of the recursive feature selection, we have used the method outlined above on four breast-cancer classification tasks (ER, PAM50, IC10 and DR described in Section 2.6) from the METABRIC dataset [6] and one tumour classification task (into BRCA, KIRC, COAD, LUAD and PRAD) from the TCGA dataset [66]. Moreover, the analysis provides insight into the gene expression data, identifying genes that are important for predictions of clinical targets.

The experiments on the task of predicting PAM50 breast cancer subtype from the gene expression data, result in identifying the most significant genes in the classification. They show that there is no single gene that allows perfect performance but rather a set of at least 100 genes is required to achieve reasonable test accuracies. The top 10 of the genes and their importance scores are shown in Figure 3.1. The findings of the high prognostic ability of

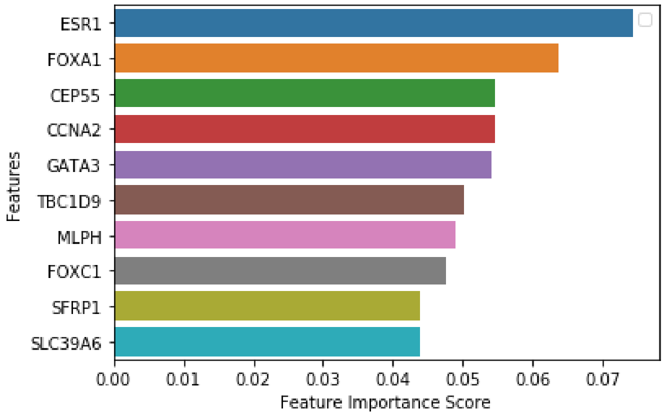

Figure 3.1: Genes ranked by their feature importance scores in the PAM50. The subset of these 10 genes was selected via recursive feature elimination.

	ER	PAM50	IC10	DR	TCGA
Full set of 24368 genes	93.14±1.2	76.28±1.2	71.26±1.2	69.70±1.9	99.75±0.41
Top 1000 features	94.07±1.3	81.77±1.6	**78.91±0.4**	**70.00±2.0**	**99.88±0.25**
Top 100 features	**94.33±1.**1	**82.00±1.7**	78.75±0.3	69.60±1.1	99.75±0.57
Top 10 feature	93.71±1.0	75.09±2.5	63.38±2.5	69.65±2.0	99.13±0.69
Top 1 feature	90.31±0.5	42.51±2.3	21.82±2.0	59.85±2.9	67.52±4.3

Table 3.1: Comparison of the predictive performance (accuracy±std) of the model trained on varying subsets of genes selected with recursive feature elimination (Random Forest model, 5-fold cross validation).

ESR1 gene expression is justified by the research in oncology [78, 79]. Using the best scoring gene alone (ESR1) gives 41.69%±0.027% accuracy on the task (where the majority class label covers 36.6% of the dataset).

Table 3.1 shows the results of the recursive feature elimination on all five classification tasks. The results show that the best performance is achieved when using a set of best 100-1000 genes and using all the available genes has a detrimental effect on the performance. Moreover, for tasks like ER status prediction, distance relapse or tumour type classification from TCGA, using already a few genes gives a near-optimal predictive performance. While for PAM50 or IC10 classification tasks more features are required.

We further explore the data and the discriminative abilities of the high importance genes. Figure 3.2 shows gene expression values against the PAM50 classes: LumA, LumB, Normal, Her2, Basal. For example, gene expression values of ESR1 (GE_ESR1) are in general higher and have a lower variance for patients classified as Luminal A or B rather than the other breast cancer subtypes. Hence, we can observe that different genes can discriminate between various subtypes of breast cancer. ESR1 and GATA3 gene expressions discriminate between the Luminal and other subtypes, while FOXA1 can identify Basal subtype. These results suggest that it is necessary to select a larger subset of genes as the features to stratify patients according to the cancer subtype. Having more features allows the model to compare expression values and successfully discriminate between multiple labels.

This section experimented with using a purely-data driven feature selection

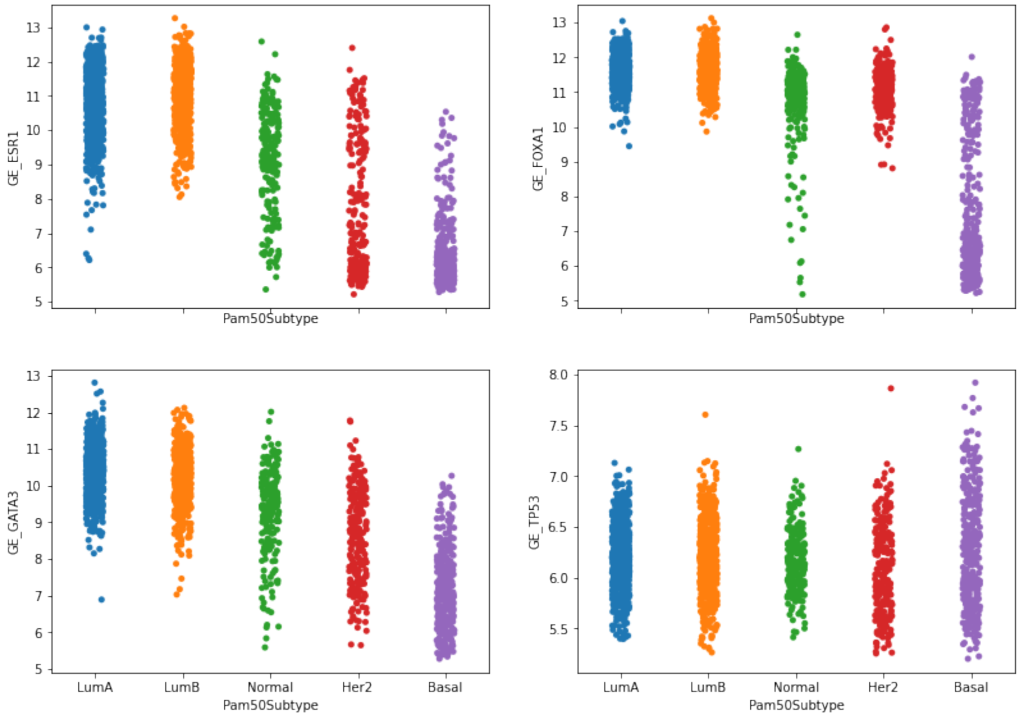

Figure 3.2: Gene expression values of ESR1, FOXA1, GATA3, CEP55 plotted against a PAM50 breast cancer subtype class.

approach. On the datasets as big as METABRIC [6], such an approach achieves great performance and can identify important genes for the considered task. However, in practice, large dataset of patient samples are not available. Then, the data-driven feature selection is unreliable as the method can pick up features that correlate with the target class by chance. It can also cause unexpected biases and overfitting towards the given patient cohort. Moreover, the selected features often lack biological relevance in a sense that little is known about selected genes and their associations with the task which makes it hard to analyse for an expert.

On the other hand, purely expert based selection can often identify only a few bio-marker genes that may not be enough for good classification performance, especially on the tasks that are not well-informed. Most of the genetics is completely unknown even to experts [80], what makes the expert based selection infeasible.

Hence, this project proposes a hybrid approach that combines expert knowledge with automated feature selection. The proposed methodology requires at least one reference gene known as an important factor in the biological process. Given that reference, we can select a set of features similar to that gene by using gene embedding derived from ontologies. This approach is often superior to data-driven feature selection in a way that it selects features (genes) that are biologically relevant to the considered task. The selected features from ontology space have known associations to the molecular processes and biological processes. Therefore they make better sense for an expert analysing the results.

3.2 Hybrid ontology-based gene selection

The proposed feature selection process is knowledge-driven and begins by choosing a single gene that is known for its expression levels being highly correlated with the task. For example, for the task of predicting PAM50 subtype, such gene is ESR1, which encodes an estrogen receptor [81, 78, 79]. This motivates using a feature subset of genes that are known to be similar to ESR1, in the sense of being close in the ontology space and associated with similar molecular functions, cellular components and biological processes expressed as gene ontology terms.

Given the set of gene embeddings produced using ontology (see Section 2.4) we use cosine similarity (Equation 3.1) to select the set of K most similar genes to the reference gene in the embedding space.

$$similarity = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}} \tag{3.1}$$

Figure 3.3 illustrates a framework which uses the feature selection process for the classification task. The selected set of genes is used to filer the features of the initial dataset of gene expressions used as an input for any machine learning (here we use Random Forest).

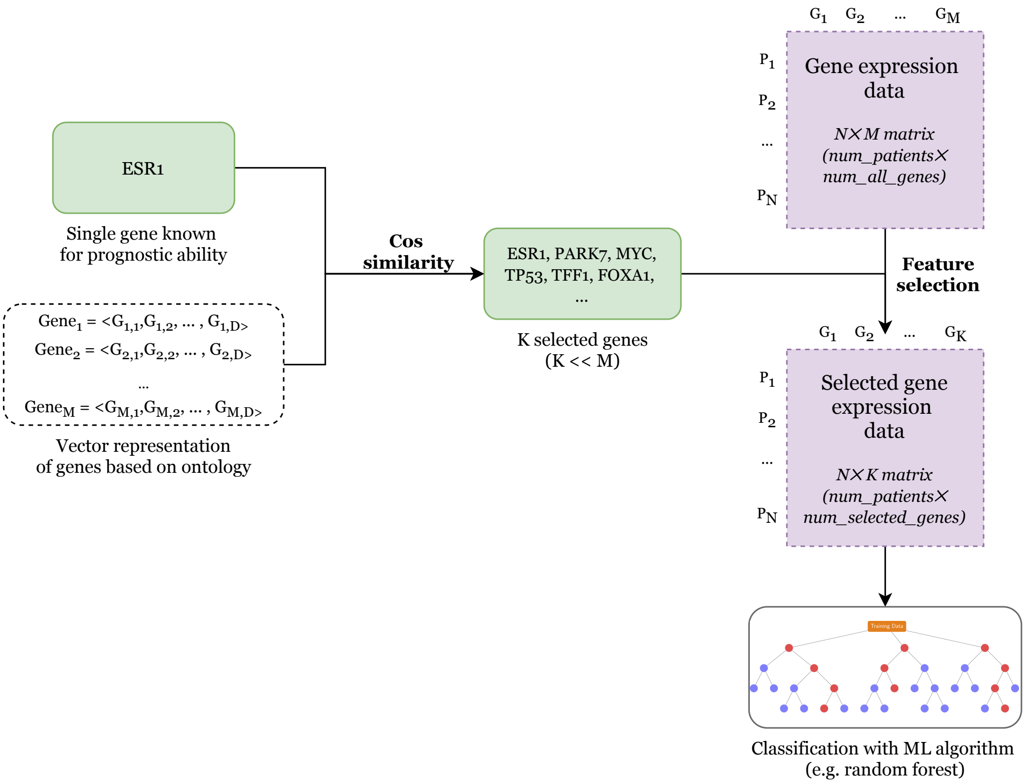
ESR1
Single gene known for prognostic ability
Gene$_1$ = <G$_{1,1}$,G$_{1,2}$, ... , G$_{1,D}$>
Gene$_2$ = <G$_{2,1}$,G$_{2,2}$, ... , G$_{2,D}$>
...
Gene$_M$ = <G$_{M,1}$,G$_{M,2}$, ... , G$_{M,D}$>
Vector representation of genes based on ontology
Cos similarity
ESR1, PARK7, MYC, TP53, TFF1, FOXA1, ...
K selected genes (K << M)
G$_1$ G$_2$ ... G$_M$
P$_1$ P$_2$ ... P$_N$
Gene expression data
N×M matrix (num_patients× num_all_genes)
Feature selection
G$_1$ G$_2$ ... G$_K$
P$_1$ P$_2$ ... P$_N$
Selected gene expression data
N×K matrix (num_patients× num_selected_genes)
Training Data
Classification with ML algorithm (e.g. random forest)

Figure 3.3: Framework of the ontology-based feature selection. Given a reference gene we use the ontology embedding to select K most similar genes. The selected subset of features is used in a Machine Learning model for patient classification task (e.g., PAM50 breast cancer subtype classification).

3.3 Evaluation of the ontology based gene selection on clinical tasks

This section describes the experiments and results evaluating the ontology-based gene selection methodology on the clinical tasks, focusing on the PAM50 breast cancer classification [7]. The evaluation compares the results of the proposed feature selection to the following baselines:

- **pre-selected 1000** genes - knowledge-driven selection of 1000 genes from Curtis et al. [6]. The genes are selected through the integration of genomic and transcriptomic data specifically on the METABRIC dataset [6]. They are the most significant cis-acting genes that are significantly associated with CNAs determined by a gene-centric ANOVA test.

- genes selected at **random** - 10, 100 or 1000 genes chosen at random from the gene columns occurring in the METABRIC dataset [6].
- genes selected by **recursive feature elimination** - data-driven approach described in Section 3.1. Note, that this is expected to be an upper bound of performance as the method is driven by maximising accuracy rather than exploiting biologically relevant knowledge.

3.3.1 Experimental setup

The approach of ontology-based feature selection is tested on the dataset of 1,980 breast cancer patient mRNA expressions from the METABRIC [6]. For the classification label of the patient in the majority of presented experiments, we use PAM50 molecular subtype (with five classes), iC10 (eleven integrative clusters), ER (two immunohistochemistry sub-types) and DR (Distance Relapse). See Section 2.6 for the descriptions. The expression values are normalised by min-max scaling to the range $[0; 1]$.

The feature selection process uses ontology embeddings that were released by the authors of Opa2vec [46], EL [5] and DL2vec [82] methods. We map the Entrez [83] gene identifiers (or protein identifiers) to the gene symbols using the mappings provided by the Enseml BioMart [84, 85] (downloaded on 21.01.2020). Due to the mismatch in naming conventions used though the decades of genetics research, some of the gene vectors were lost in the process. Table 3.2 shows the summary of the used vector representations and the number of the gene embeddings obtained after mapping of symbols.

Embedding method	Ontologies	Dimensions	Number of genes after name conversion
Opa2vec (GO)	GO	100	17'580
Opa2vec (merged)	GO, MP, UBERON	100	20'258
EL	GO	50	13'866
DL2Vec	GO, MP, UBERON	200	17'591

Table 3.2: Details of used ontology embeddings.

Selected features/genes are used as an input to the Random Forest model. The Random Forest classifier uses 100 trees, and the minimum number of samples required to split an internal node is set to 2. These parameters were chosen via initial hyper-tuning experiments on the pre-selected set of 1000 genes, using a grid-search selecting over 12 configurations. Another study was also performed using the MLP as a classifier, but the conclusive results were similar so they will not be presented.

3.3.2 Results

In the first set of experiments, we consider a scenario in which the model uses K chosen genes as input: the reference gene (ESR1) and K-1 genes closest to in the ontology embedding space.

Using the standard pre-selected 1000 genes as the input achieves 74.0%. Another baseline is selecting a set of K genes at random, which also holds strong performance as taking gene expression levels from diverse genes has a high predictive ability. To make the comparison fairer in case of the random baseline we also built a set of K genes by randomly sampling K-1 genes and include the reference gene (ESR1). Note, that the pre-selected 1000 genes

	Number of genes (including ESR1)		
Gene selection used	1000	100	10
Opa2vec (GO only) [46]	78.2±0.4	71.6±0.4	56.7±0.6
Opa2vec (merged ontologies) [46]	77.7±0.5	70.6±0.6	56.8±0.5
EL [5]	**78.7±0.5**	70.7±0.4	**59.2±0.5**
DL2vec [82]	76.5±0.1	**72.0±0.2**	57.3±0.1
Random	75.1±0.7	65.1±2.6	46.5±5.5
Random (+ ESR1 gene)	74.8±0.3	68.9±3.5	55.9±2.4
Pre-selected 1000 (Curtis et al. [6])	74.0±1.1	-	-
Recursive feature elimination	81.8±1.6	82.0±1.7	75.1±2.3

Table 3.3: Comparison of the predictive performance on PAM50 (accuracy±std) task of the model trained on set of 1000, 100, or 10 genes selected with the proposed feature selection methodology, using different gene embeddings. The results are compared to the baselines of randomly selected set of genes (of the same size), set of genes including ESR1 and randomly selected genes, and a pre-selected set of 1000 genes from [6].

from Curtis et al. [6] do not include ESR1 as it was not selected as one of the most significant cis-acting genes.

Table 3.3 shows that ontology-based gene selection methodology outperforms both the random baseline as well as the widely used set of 1000 genes derived based on CNA associations. The best performance for the sets of 1000 and 10 genes is achieved when using EL embedding. However, the differences between the different methods of generating ontology embeddings are minor.

As expected, the results for hybrid ontology-based feature selection are lower than the performance of data-driven recursive feature elimination. However, choosing the genes similar in the ontology space we gain the benefit of having biologically relevant features. By construction of embeddings, the genes with low distance are associated with the similar Gene Ontology terms and hence annotated with the similar biological processes and molecular functions. Such features should have more meaning for the expert analysing the results.

In the second set of experiments we verify to what extent, choosing a set of genes similar to the reference ESR1 gene can replace the performance of the original gene. That is, we choose K genes that are closest to the ESR1 gene in the ontology embedding space, but do not include the ESR1 gene itself as the input to the model. This task aims at gaining greater insight into the contribution of prior knowledge in the context of gene expression data. By comparison to random baseline, we analyse the gain of the ontology knowl-

	Number of genes (discluding ESR1)		
Gene selection used	1000	100	10
Opa2vec (GO only) [46]	77.8±0.3	69.0±0.3	45.4±0.7
Opa2vec (merged ontologies) [46]	77.2±0.4	68.4±0.5	47.3±0.3
EL [5]	**78.2±0.5**	**70.6±0.3**	**58.1±0.5**
DL2vec [82]	76.3±0.1	70.5±0.1	51.0±0.1
Random	75.1±0.7	65.1±2.6	46.5±5.5

Table 3.4: Comparison of the predictive performance on PAM50 task (accuracy±std) of the model trained on set of 1000, 100, or 10 genes selected with the proposed feature selection methodology, but without including the reference gene (ESR1) in the set.

edge in the setting where the selected genes should contribute predictions on their own rather than in a combination with a chosen reference gene.

Table 3.4 shows that replacing the ESR1 gene with similar genes from the latent ontology space is significantly better than choosing a random sample of genes, on the PAM50 classification. It also shows that the best performance is achieved by EL embeddings that aim to encode the geometric structure of the ontology. The highest gain in performance is observable when using just 10 closest genes.

The positive results in this task suggest a high level of correlation between the expression levels of the genes that are similar according to the embedding space produced with ontologies.

Figure 3.4 shows the effect on the performance depending on the number of genes used for predictions, which varies across the classification tasks.

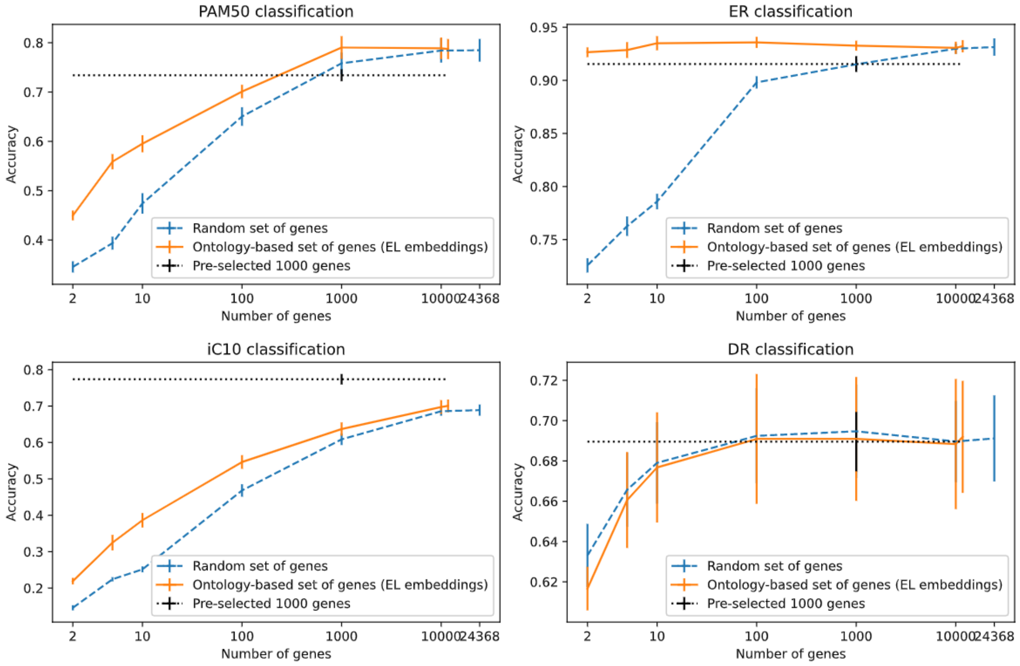

Figure 3.4: Results of the ontology-based feature selection on the patient classification tasks (PAM50, ER, iC10, DR). Each plot considers a different task and shows performance of RF trained on the varying number of genes either selected using the proposed ontology-based method (with EL embeddings [5]) or selected at random. The black lines show the results of the pre-selected 1000 genes.

The yellow line shows the results obtained by the gene selection using EL embeddings [5]. For comparison with other embedding types see Figure 3.5.

For the PAM50 and iC10, we observe that predictive performance increases as more genes are added, demonstrating that the relevant information about the cancer type is distributed across multiple genes, not only the ones related with similar terms in the ontology space.

Note, that pre-selected set of 1000 genes is outperforming other gene sets on the iC10 classification task. These 1000 genes were specifically selected by Curtis et al. [6] to define the integrative clustering of patients into groups on the METABRIC data. iC10 (or IntClust) is a classification of breast cancer into 11 subtypes based on molecular drivers identified through the integration of genomic and transcriptomic data, where for the genomic data only the pre-selected genes were taken into account. Hence, by definition, the pre-selected set of 1000 genes are the best set to decide this particular clustering (but not necessarily for other patient classification). Choosing different genes or adding other irrelevant features only impairs the performance.

The ER status classification is a much simpler, binary task, that achieves high results already with 2 or 5 genes selected with the proposed method. The performance is consistent as more genes are added.

Distance Relapse is a binary target that groups the patients depending on whether cancer metastasised to another organ. It is a very complex and difficult predictive task where gene expression analysis has limited ability in predicting this variable. One can achieve better results via integrating clinical and mRNA data [86]. Multiple experiments in this project have shown that the performance of ML models on DR from only mRNA data is relatively low no matter which feature selection or machine learning model is used. Hence, we not consider this task in the rest of the dissertation.

Figure 3.5 shows the results of the experiments with various embedding types used for the ontology-based feature selection. The differences in accuracy are minor, but one can observe a slight improvement when using EL [5] or DL2vec [82] embeddings when the smaller sets of 2-10 genes are selected.

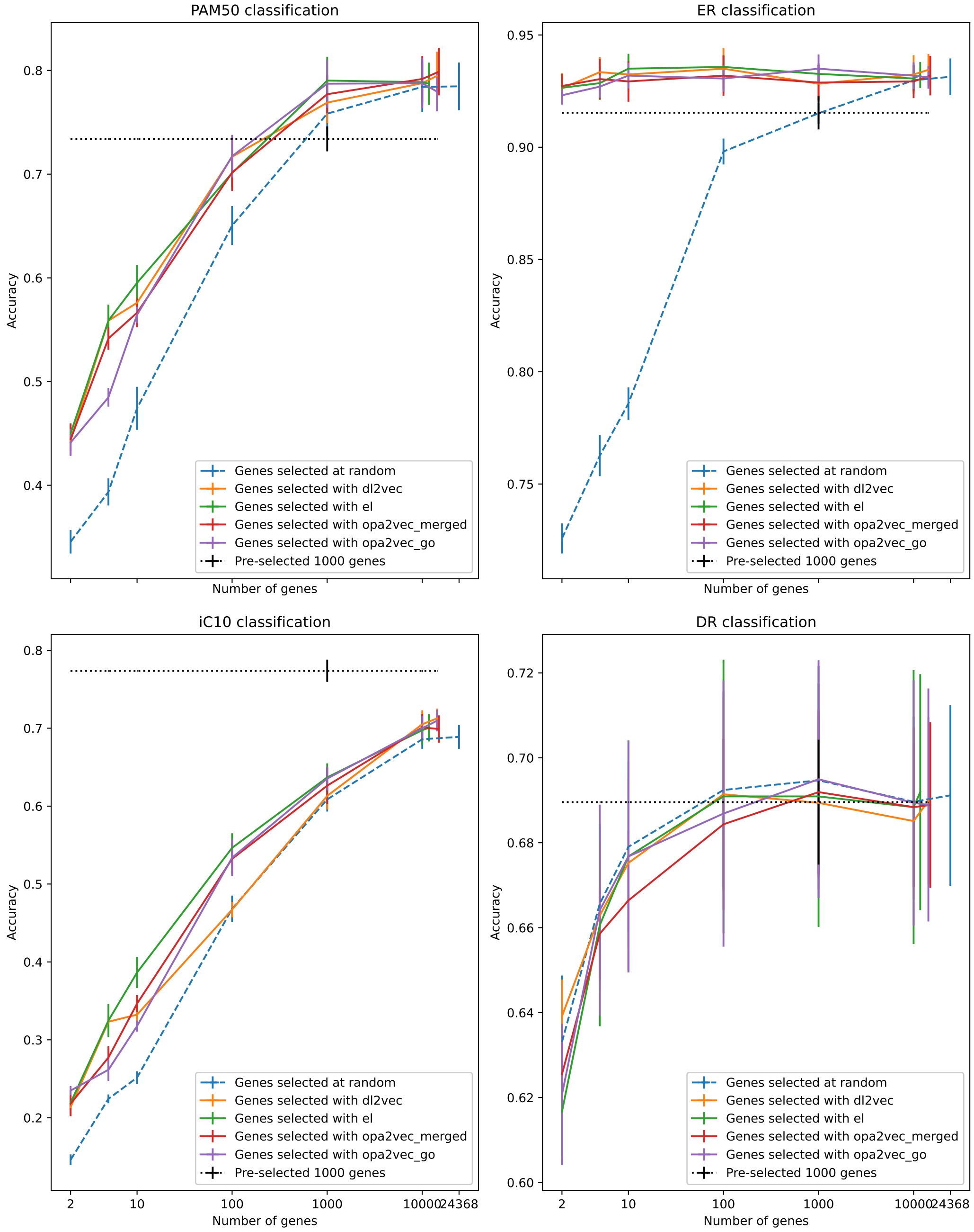

Figure 3.5: Results of the ontology-based feature selection on the patient classification tasks (PAM50, ER, iC10, DR). Each plot considers a different task and shows performance of RF trained on the varying number of genes selected using the proposed ontology-based method with several embedding types (see Table 3.2) or selected at random. The black lines show the results of the pre-selected 1000 genes.

Across the clinical tasks of PAM50, ER, iC10, the proposed method of feature selection significantly outperforms the random baseline. The gain is the most significant for smaller sets of genes, and diminishes when more than a few thousand features are used (i.e., when almost all genes are used). This suggests that the relevant information is often present across the genes that are close in the ontology space to the reference gene.

To further explore this locality of the predictive signals, the next section considers the Single Gene Inference (SGI) task. By aiming to predict the value of one gene based on the other genes close in the ontology space, the experiments assess how well the similarity measure based on the ontology embedding captures the gene expression dependencies.

3.4 Evaluation of the ontology-based gene selection on the single gene inference task

Hashir et al. [75] in their paper asked the question *"Is graph-based feature selection of genes better than random?"*. The authors explored the use of gene networks for feature selection in machine learning on gene expression data, specifically considering the Single Gene Inference (SGI) task [9, 87]. Their analysis finds gene expression dependencies can be predicted almost as well by using random genes as by using curated interaction networks. Crawford and Greene [65] hypothesise that this effect is highly variable across genes.

We conducted a similar set of experiments to determine the impact of ontology-based feature selection in the single gene inference task. We compare the results achieved by predicting the gene value using the K nearest neighbours in the graph built from ontology embedding (here, EL embeddings [5]) versus using K randomly selected genes. The experiments aim to measure the true benefit of using biologically relevant ontology-based gene selection in the context of capturing the gene expression dependencies.

3.4.1 Experimental setup

For the consistency with the previous work [75, 65] the experiments use mRNA samples from the TCGA PANCAN database [66], prepared and released as a benchmark by Samei et al. [67]. It contains 10'459 samples spanning multiple tissues and has expression values for 20'530 genes for each sample. Most samples have been diagnosed with some form of cancer, but many healthy examples are also included. As the various tissues and various tumour types result in different expression level profiles, using the TCGA rather than breast cancer METABRIC cohort [6] allows avoiding the biasing the gene expression towards the breast cancer-related mutations.

In the single gene inference task, the model is predicting the expression of one gene given the expression values of other genes. We convert the real-valued expression level to a binary variable representing if it is over or under-expressed compared to the mean value of that gene across all patients. This allows a simple binary prediction that can be evaluated using AUC. Performance of the predictions using K genes closest in the ontology embedding space is compared to using K genes selected at random.

As the machine learning model, we follow Dutil et al. [9] and use MLP with 2 hidden layers, 512 channels, ReLU activation and 0.2 dropout.

In the second experiment, we select at random 1000 genes from the dataset (ensuring that the genes are present in the set of ontology embeddings). For each of the genes, we predict the binarised expression and then collectively analyse the added value of using the ontology-based gene selection method across a variety of genes.

3.4.2 Results

For some of the genes like HLA-B and S100-A9 (explored in [9]), the neighbouring 10 genes provide near-perfect predictors of its value, better than picking even thousands of genes at random (see top row of Figure 3.6). For others like ESR1 and TP53, knowledge informed selection brings only small improvement and sampling numerous genes is required to predict the value.

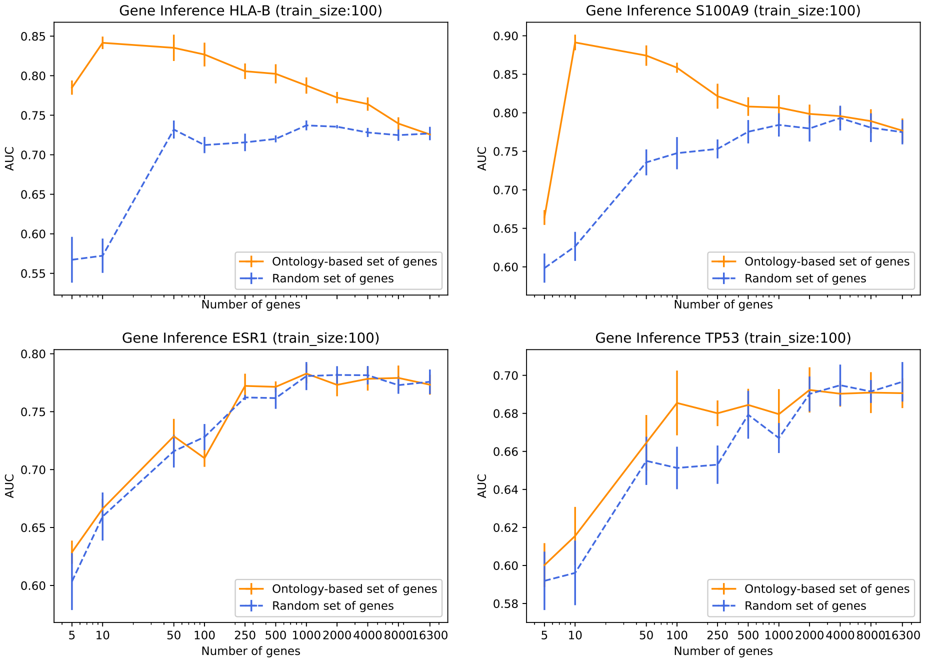

Figure 3.6: AUC scores of predicting if a single gene value is over/under-expressed given some genes closest to it in the ontology embedding space. The performance is compared to the random set of genes. For each plot, the target gene varies. The scores are plotted against the number of genes used for prediction. Three trials are used and error bars show standard error.

For these genes, the value is not highly correlated with other genes close in the ontology space.

We conduct experiments to determine for how many genes, using prior knowledge for gene selection is beneficial compared to the random selection. For 1000 genes selected at random (from the intersection of genes present in the dataset and the ontology embeddings) we want to predict whether the gene value is above or below mean for a particular patient, using as input the values of other K genes, for K = 10, 100 or 1000. For each such case, we train two MLP models, one predicting the gene value from the other K genes selected from the neighbourhood of the reference gene in the ontology embedding space, and the other predicting the value from K random genes. The Figure 3.7 shows a histogram of the difference in AUC between the models. Genes with an AUC improvement > 0 (right side) were better predicted when using the ontology informed selection.

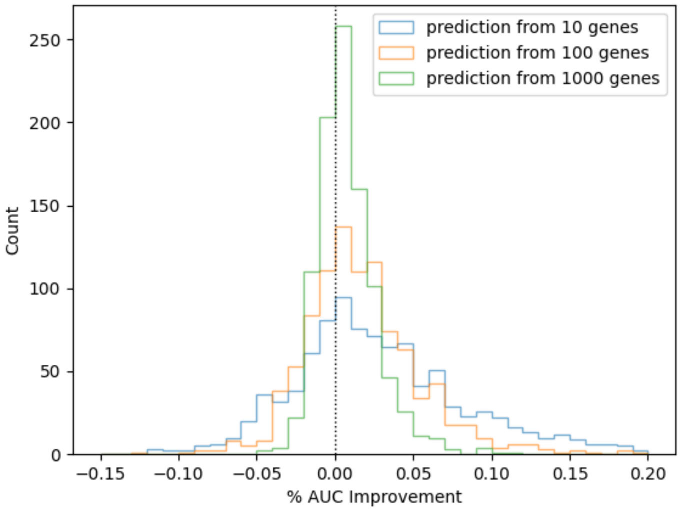

Figure 3.7: Distribution of AUC improvements over genes from using the ontology-based gene selection rather than random. The mass on the right from the dashed line represents the genes for which the SGI performance was improved.

In case of choosing a set of 10 predictors, for 68.5% genes, ontology-informed selection holds better AUC results than random selection. In the case of 100 predictors, for 66.5% ontology-based selection performs better. In the case of 1000 predictors, it is for 63.7% genes and the improvements are typically minor, improving AUC by 0.018 on average.

In conclusion, for the majority of the genes, the proposed ontology-based feature selection is better than a random selection, at capturing the gene expression dependencies. Hence, there is a benefit of using prior knowledge for the similarity measure of genes contributing features for predictive tasks.

3.5 Summary

This chapter presents the methodology and results of the experiments on an ontology-based selection of the human genes for the input to a machine learning model. First, we analysed the tasks and gene expression data by using a data-driven approach of recursive feature elimination. Although this method achieves great performance, it often lacks the biological relevance and can pick up features that correlate with the target class by chance.

We proposed a novel approach of using the ontology embeddings [46, 4, 5, 82] as a similarity measure between the genes, which are the features in the considered tasks. The evaluation shows that, on the clinical tasks of patient classification (PAM50, iC10, ER), they hold a better performance than using a set commonly used in the literature or selected at random. Further experiments and results on the Single Gene Inference task show that for many genes ontology-based feature selection of genes is better than random at capturing the gene expression dependencies.

Chapter 4

Imposing structural inductive bias capturing gene similarity

Ontologies are widely used in biology, expressing decades worth of knowledge about genes, their functions, processes and relations [10]. However, in the conventional models for stratifying cancer patients from the gene expression data, the input is provided as an array without leveraging the known relations or similarities between the genes.

The processes present in the human body are often of such intricacy that it is almost impossible to disentangle contributions of one gene from the other, as effects of one gene depend on interaction with many others [80]. Considering the values of gene expressions in isolation is not effective. So using neural network models for analysing the targets depending on complex relationships and relevant information present across many genes, necessitates the usage of multiple layers. Having fully connected layers in a deep learning model with high-dimensions creates a magnitude of spurious connections that can amplify the noise and hurt the performance and transparency. Instead, we can restrict the processing by using a neural network model operating on a graph based on prior knowledge about gene similarities derived from ontologies.

This chapter proposes a method for enhancing the neural model with a structural inductive bias directed by a graph generated from ontology embeddings. This is possible by using Graph Convolutional Neural Networks that restrict the message passing across the nodes based on their neighbourhood.

With high-dimensional low-sample data, known relationships between variables can help a model avoid learning spurious correlations and ignore noise which correlates with a target prediction by chance. The method uses Graph Convolutional Neural Network to impose structural bias, similar to the spatial bias imposed by convolutions on an image. For the topology of the graph, we use connections generated from ontology embeddings and compare it to using protein-protein interaction networks [9] and randomly generated graphs. Experiments also explore various ways of nodes embedding.

The results show that this approach provides an improvement for the considered clinical task of PAM50 classification, especially in a low data regime. The resulting model is also transparent, as by analysing weights of the classification layer we can identify most important genes used for the prediction, and plot them in the space of ontology embeddings.

4.1 Methods

In the proposed method, the prior knowledge graphs are complementary to the main task and impose a structural bias on the model. Known similarities/relationships between the features (genes) direct processing in the network and help the model to avoid learning spurious correlations [9]. Such sparsification of connections may be especially beneficial with low numbers of samples and high-dimensional data. Enforcing convolutions on the genes based on their similarity captures localised patterns of data, similarly as standard convolutional neural networks capture spatial relationships of pixels in the images [21]. The localised biases are very effective for processing image data because there is high covariance within local neghburhood of pixels [20]. In the proposed method we aim to exploit similar properties of genes by using ontology-based neighbourhood which also captures the data dependencies as shown in Section 3.4.

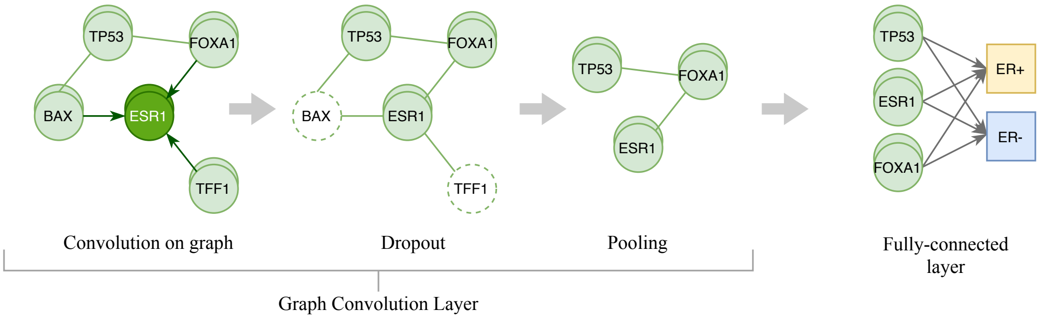

Figure 4.1: Overview of the Graph Convolutional Network applied to gene expression data.

Figure 4.1 illustrates the workflow of the neural model. Each gene contributes a node in a prior knowledge graph where edges are representing semantic similarity between the genes. The topology of the graph is determined from the ontology embeddings and is the same for all the patient samples in the dataset. Each patient spans a new instance of the graph with different expression values used as an input to initialise the set of (the same) nodes. At each convolution, the expressions (or embeddings) of neighbouring genes are aggregated together based on their connectivity (Figure 4.1 shows the update step for node representing ESR1, but similar message passing happens for every node). After convolution, to help with the data scarcity, the gene nodes are dropped at random and also undergo pooling determined by topology-based aggregation clustering. Finally, a prediction is made from the remaining nodes via a fully connected layer.

4.1.1 Graphs used for inductive bias

Prior knowledge about similarities and relations between genes can be expressed as a graph, where nodes represent genes. The gene nodes are connected with an edge if the similarity between them exceeds some threshold. The similarity values are derived from the latent space of the ontology embeddings. The experiments explore using 4 types of ontology embeddings (Opa2vec with merged ontologies [46], Opa2vec with GO only [46], EL [5] and DL2vec [82]) as described in Section 2.4.

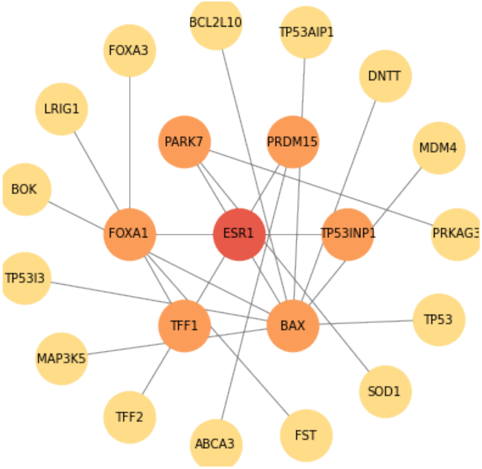

Figure 4.2: Graph generated with EL embeddings, showing the two-hop neighbourhood of the ESR1 gene. The graph was generated with K=3. Note that for some of the edges we have more neighbours than K due to reciprocal connections.

The process of generating ontology graphs creates edges that connect each gene to its K nearest neighbours in the ontology embedding space. We first generate ontology embeddings for all the genes. Then for each of the genes, we find K closest genes according to the cosine distance metric and add undirected edges between them. Figure 4.2 shows an example graph generated with EL embeddings, showing the neighbourhood of the ESR1 gene. Although the graph used K=3 for generating connections, for some of the edges we have more neighbours due to reciprocal connections. That is, the gene might have occurred as top 3 for more than 3 of other genes.

In relation with the previous work on imposing bias from biological networks via graph convolutions, our experiments are comparing the graph described above with two undirected graph datasets containing a mixture of protein-protein interaction and gene co-expression data [62, 88]. We also consider a baseline of a graph with randomly generated edges (with matching degree).

The possible advantage of using the graph generated based on similarity, rather than curated PPI network, is that one can freely control the sparsity or number of neighbours for each of the nodes. Therefore, it overcomes the problems regarding the genes with low degree [75, 63]. Moreover, we can generate a connected graph for any subset of genes present in the ontology embedding space, while taking a subset of genes for PPI network may result in a disconnected graph.

The evaluation compares the following graph topologies:

- Ontology graph - as described above. The graph is generated based on the ontology embeddings (Opa2vec [46], Onto2vec [4], EL [5] or DL2vec [82]) so that each of the genes is connected to its K nearest neighbours (according to cosine distance).
- Random-K graph - a baseline that allows determining if performance gains come from the model itself or the underlying prior knowledge. The method of generating the graph is analogous to the ontology graph. That is, for each node, connections are added to K neighbours (randomly chosen).
- GeneMANIA [62] - is a gene association network integrating numerous datasets. Many gene functions were predicted using ridge regression as well as a label propagation algorithm.
- STRINGdb [88] is a protein-protein interaction network. It has heterogenous types of interactions including co-citation in research papers, coexpression, gene neighbourhood, and co-occurence in metabolic pathways. Bertin et al. [63] replace proteins in the network with protein-coding genes, to create a gene-gene interaction network.

The considered graphs vary in nodes and edges number. The number of nodes determines the number of genes covered by the graph. The number of edges determines the density of the graph and therefore the number of neighbours considered in each message passing steps. GeneMANIA is a sparse graph in which each node has on average 32 neighbours, while STRINGdb is much denser with an average of 605 neighbours per node. For automatically generated ontology graphs (and random graphs) the number of edges depends on the parameter K.

Note, that not all of the genes present in the graph are features in the METABRIC (MB) dataset [6]. In the prepossessing of the data, we mapped

Graph		Ontology generated graphs				Strindb	Genemania
		Opa2vec	Opa2vec GO	EL	DL2Vec		
Nodes (genes)		20'258	17'580	13'866	17'591	18'851	16'300
Nodes occuring in MB		14'952	14'276	11'718	14'308	15'411	14'035
	K=500	8'401'561	7'320'451	4'600'773	7'976'452		
Edges	K=100	1'731'962	1'516'432	979'948	1'542'359	5'703'510	264'657
	K=10	179'935	157'359	103'104	150'068		

Table 4.1: Comparison of graph sizes including number of nodes (and size of intersection of node names with genes occurring in the the METABRIC (MB) dataset [6]), and number of edges. In case of the graphs generated from ontologies the numbers of edges are given for a certain K parameters specifying number of nearest neighbours used to generate graph edges.

all the gene or protein IDs (e.g., `ENSG00000091831`) to the gene symbols (e.g., `ESR1`). However, the mapping of protein and gene names did not succeed for all of the nodes. This is a consequence of plentiful naming schemes that emerged due to the nature of how genetic knowledge has developed over decades. Therefore, the experiments use the intersection of the genes such that have common names in graphs and MB dataset. This causes a loss in the number of genes considered, which could be improved given a better symbol mapping.

4.1.2 Graph convolution operation

In the message passing implementation we reused the codebase from Dutil et al. [9] who used standard GCN architecture introduced by Kipf & Welling [8]. The convolution over the first neighbouring nodes is defined as:

$$\mathbf{H}' = \tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{A}'\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}\mathbf{W} = \tilde{\mathbf{A}}\mathbf{H}\mathbf{W} \tag{4.1}$$

Where $\mathbf{H} \in \mathbb{R}^{n \times c}$ represents the input features (n is the number of nodes, and c is the input feature size). In [9] the input $\mathbf{H}$ of the model are gene embeddings, learned during training, and scaled by their corresponding expression level. $\mathbf{W} \in \mathbb{R}^{c \times o}$ is a weight matrix for the specific neural network layer, $\mathbf{A}' = \mathbf{A} + \mathbf{I}$ is the adjacency matrix (with added identity matrix to enforce inclusion of the node itself), and $\tilde{\mathbf{D}}$ is the diagonal node degree matrix that is applied for symmetric normalisation of the feature vectors. Each such layer contains only $c \times o$ learned parameters that define matrix $\mathbf{W}$. Same matrix $\mathbf{W}$ is applied for all the nodes. Hence this convolution operation can be regarded as firstly passing the node embeddings from all the node neighbours through the same transformation and then taking the mean of them.

We conducted another study using a different message passing mechanism of Graph Attention Network [28], but no significant improvement in performance was observed, so the results are not reported.

The model has a skip connection at each convolution layer, which preserves the node itself, as well as averaged messages from the neighbourhood. This allows the network to keep high weight for preserving the original node embedding rather than just averaging it together with the neighbours. The full convolution is then followed by an activation function (ReLU), 40% dropout, and an aggregation clustering method to reduce the number of nodes at each layer. The aggregation clustering uses hierarchical clustering based on the topology of the prior knowledge graph to reduce the number of nodes by half after each convolution.

$$\mathbf{H}'' = Aggregate(\sigma(\tilde{\mathbf{A}}\mathbf{H}\mathbf{W_1} + \mathbf{H}\mathbf{W_2})) \tag{4.2}$$

The model also may contain pre-pooling layers, that is, additional message propagation steps occurring within each layer before the aggregation clustering.

At the last layer of the network, the remaining nodes are concatenated together and fed into a fully-connected layer to make the final prediction stratifying the patients, for example into cancer subtypes (see Figure 4.1).

4.1.3 Node embedding methods

Dutil et al. [9] learn embeddings for the genes during the training of the graph neural network. The input for the nodes in the model is these embeddings (initialised to random) scaled by their corresponding expression level for a particular sample (B on the Figure 4.3).

Such method, initialising the gene embeddings to random values and scaling them via gene expression levels for each patient may potentially have a detrimental effect on the model as it introduces more noise in the process and may unnecessarily obfuscate the values of the input features (expressions).

Hence, we propose and compare other methods of obtaining the input features for nodes **H**.

The simplest method is to use the expression values directly as the model input without scaling them via node embeddings (A on the Figure 4.3). The potential problem with that approach is that the message-passing step in

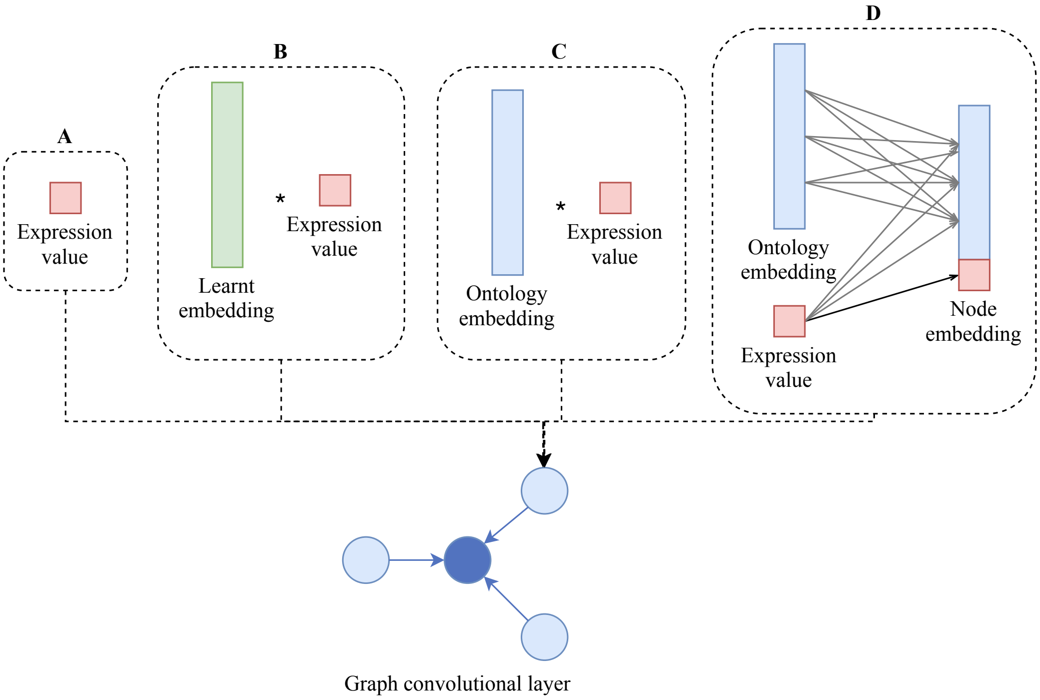

Figure 4.3: Considered methods of obtaining input node embedding.

GCN may be unable to extract enough information from the nodes' neighbourhoods, which in turn with limit the expressive power and learning abilities. Due to the simple nature of GCN operation, in which the convolution operation uses only one weight matrix for all the nodes and averages resulting values, nodes are not directly distinguishable from each other. For example, if we have a certain node with neighbours of gene expression values $G_1 = 0.8, G_2 = 0.2$, the result of the convolution will be the same as if the values were permuted $G_1 = 0.2, G_2 = 0.8$, or even simply had the same mean as original values $G_1 = 0.5, G_2 = 0.5$. Having the embeddings distinguishing between certain genes may be desirable, to avoid this ambiguity.

The embeddings do not have to be learnt during the training of the GCN starting from random values but could be static and knowledge-driven. Hence we propose using ontology embeddings (described in section 2.4) and combining them with the expression values for given patient via scaling (C on the Figure 4.3).

The scaling via gene expression introduces a risk of obfuscating the true values of gene expression. Another problem is that using the static ontology embeddings directly may lack enough distinguishing power between the genes, as the genes close in the ontology space will have similar embedding vectors.

To avoid such problems we propose another method of combining knowledge-based gene embeddings with the expression values (varying for each sample). First, the ontology embedding is concatenated with expression and passed through a fully connected layer. Then we concatenate the expression value to the produced representation. Such an approach may be potentially better than a simple scaling as it adjusts the node embeddings depending on the expression but also ensures the preservation of the expression value itself.

Having knowledge-driven node embeddings helps convolution kernels to differentiate better between the genes during the graph convolutions, simultaneously capturing the similarity between the genes related by biological knowledge.

4.2 Evaluation

The experiments assess the usefulness of the proposed approach of incorporating biological knowledge from ontologies, for the tasks of classifying cancer patients from their genomic data: PAM50 (5-class molecular cancer subtype), ER (binary classification into immunohistochemistry subtype) and iC10 (11 IntegrativeCluster subtype)[1].

The evaluation explores various scenarios of data scarcity, the set of input genes, topology of the prior knowledge graphs and node embeddings methods. The section begins by presenting implementation and hyperparamenter tuning details. Then it compares and discusses the results varying the settings across the following aspects:

- **data scarcity**: each experiment is performed with using the training sample of 1500 or 100 patients. These results are presented next to each other in the tables for all the other experiment.

- **dimensionality of the input**: the Subsection 4.2.2 considers using all genes available in the dataset (over 20'000) with no feature selection and hence explores the performance in the high-dimensional scenario. The Subsection 4.2.3 considers using pre-selected 1000 genes as in Curtis et al. [6].

- **topology of the prior knowledge graphs**: within the Subsection 4.2.2, we compare different graphs (ontology-based, random or PPI networks) used for inducing the structural bias, as described in Section 4.1.1.

- **node embedding methods**: within the Subsection 4.2.2, we also compare different methods for obtaining node embeddings as described in Section 4.1.3 and Figure 4.3.

[1] The task of DR (distance relapse) was also considered but due to the difficulty of the task, the results showed no significant difference between the models.

4.2.1 Experimental setup

For the bulk of the implementation, we reused the codebase from Dutil et al. [9] implementing the GCN and MLP models in PyTorch [89]. We extended the implementation by: introducing custom graph topology generated from four kinds of ontology embeddings, implementing novel methods of generating knowledge-driven node embedding, adjusting the models and further experiments verifying posed hypotheses.

As the baseline models, we use the Multi-Layer Perceptron (MLP) with dropout (implemented by Dutil et al. [9]) and Random Forest (implemented in scikit-learn [90]).

The models include a high number of hyperparameters. The parameter tuning used Bayesian Optimisation [91], with a Gaussian Process searching over discrete values for learning rate (from 1e-3 to 1e-6), number of layers (from 1 to 4), number of prepool layers (from 0 to 4), channel size (from 2^4 to 2^9), embedding size (from 2^4 to 2^9), and dropout (`True` or `False`). Based on the accuracy on PAM50 task validation split, we selected 5 best configurations for MLP (showed in Table 4.2) and GCN (showed in Table 4.3).

Both GCN and MLP models were trained using the Adam optimiser for the maximum of 100 epochs with early stopping criterion of patience 30.

For each task, for both GCN and MLP, we run five training runs for each of five configurations of chosen hyper-parameters, where each run uses a different stratified split of the dataset. We report the averaged scores (and standard deviation) on the test split of the best performing setting.

Number of layers	Dropout	Learning rate	Channel size
2	0	0.001	64
2	0	0.0001	64
2	0.2	0.001	64
2	0.2	0.001	16
1	0.2	0.001	64

Table 4.2: Five chosen hyperparameter settings for MLP.

Pooling layers	Prepooling layers	Dropout	Learning rate	Channel size	Embedding size
1	0	0.4	0.001	64	64
1	0	0.4	0.0005	64	64
1	2	0.4	0.001	16	16
1	2	0.4	0.0005	16	16
1	0	0	0.001	64	64

Table 4.3: Five chosen hyperparameter settings for GCN.

4.2.2 Results when using all 24'368 genes

The first set of results considers a setting in which all of the available genes are used as an input to the model with no feature selection. This allows verifying the performance in the high-dimensional low-sample data. For the standard models of RF and MLP, we use 24'368 gene expression as the features. For GCN we use the intersection of the gene names from the ontology embeddings and MB dataset (e.g., 14'308 genes for dl2vec embeddings [82]).

The summative results for PAM50 cancer subtype classification, presented in Table 4.4, demonstrate a significant improvement over the baseline models not using prior knowledge (with p-value of 0.035 between the results of GCN and MLP for the training sample of 1500 computed via two-tailed Student's t-distribution on the accuracy scores on cross-validation folds). Table 4.4 shows results for both GCN using pure expression values with no node embedding (A in the Figure 4.3) and GCN with the proposed method of obtaining node embeddings from combining the ontology embeddings with expression data

	PAM50		ER		IC10	
train size	100	1500	100	1500	100	1500
RF	69.3±1.1	78.4±1.0	88.7±1.2	93.1±1.2	66.7±0.9	71.3±2.1
MLP	60.1±5.0	77.9±2.5	88.8±1.8	90.9±5.0	40.4±4.5	68.9±1.6
GCN - no node emb (A)	**72.3±2.9**	**81.2±0.8**	**91.4±1.0**	93.7±1.0	48.2±3.1	**74.3±2.3**
GCN - onto node emb (D)	**72.7±3.7**	**81.6±2.2**	90.8±0.5	93.8±1.2	50.4±2.3	73.7±3.3

Table 4.4: Performance comparison of the methods on the PAM50, ER and iC10 patient classification tasks (accuracy±std), using all genes as input features (in case of GCN, intersection of genes present in the dataset and graph). Bold font marks the cases when the model obtained statistically significantly better results than baselines.

(D in the Figure 4.3), which are the two best of explored node embeddings (see Table 4.6). The proposed method of node embeddings achieves higher accuracy but the difference is not significant.

The improvement achieved by GCN over baselines is especially visible in the case of the low data scenario of 100 patient samples, therefore confirming the benefit of incorporating the biological knowledge about genes into the model operating on high-dimensional low-sample data. The MLP model is struggling to achieve high scores with the scarce data, due to the vast spurious connections, exhibiting accuracy lower than in case of Random Forest and also high standard deviation between the runs. Directing the neural connections via ontology-based graph convolutions seems to overcome these problems and perform better.

On the ER task, the GCN (with no node embeddings or the proposed ontology-based node embeddings) outperforms the baseline models in the low training data setting of 100 patients. However, for the set of 1500 patients, Random Forest baseline was too strong and the p-values obtained were too high to ascertain statistical significance for rejecting the null hypothesis at 0.05 significance level.

On the task of iC10 classification, the GCN outperforms baselines when there is sufficient training data used, but both neural network models of GCN and MLP fail in the situation of a scarce dataset. This is because the iC10 task has 11 classification labels, so the number of examples per class is less than 10. This is not enough to train a neural model.

Comparing graphs used for message passing

Another study compares the performance depending on the source of the prior knowledge, comparing various topologies of the graph used for the convolutions. It compares the proposed ontology graph (generated from DL2vec [82] gene embeddings) to the random baseline where the graph of the same degrees for nodes is created but with randomly chosen edges. Such comparison allows determining if the increased performance is driven by the prior

	PAM 50	
	train size=100	train size=1500
Proposed ontology graph	72.0±2.9	81.2±0.8
Random graph	68.9±3.9	78.0±4.7
GeneMANIA graph [62]	71.7±2.6	80.1±2.1
STRINGdb graph [88]	71.2±2.5	80.7±1.2
RF	69.3±1.1	78.4±1.0
MLP	60.1±5.0	77.9±2.5

Table 4.5: Performance comparison of the graphs directing GCN structure on the PAM50 classification task (accuracy ± std). The GCN used expression values as the nodes input, without node embedding mechanism (A in the Figure 4.3).

knowledge present in the topology of the graph or if it is a result of the effective GCN architecture itself.

The results in Table 4.5 show that performance of a randomly generated graph exceeds the accuracies achieved by the MLP baseline and compares with the performance of the random forest (RF). This suggests that the architecture of graph convolutions over the genes is powerful and useful in denoising the data even when the topology of the underlying connections has no biological relevance.

Using the ontology-based topology further improves the performance, showing that the incorporation of prior knowledge to perform the graph convolutions over the similar genes is beneficial for the predictions. Such operation benefits from the locality within the gene space, similarly to how convolutional networks exploit spatial locality when processing images.

To relate the results with previous work by Dutil et al. [9] and Bertin et al. [63], we compare the automatically generated ontology graph versus topology of gene interaction networks (see Section 4.1.1). The results suggest that curated biological networks are a good source of prior knowledge. The slightly improved performance in case of the automatically generated ontology graph is probably caused by the ability to freely tune the sparsity of the graph, which regularises sizes of elements in convolution and overcomes the problems caused by sparsely connected genes in the network [63, 92].

Another set of experiments was conducted to compare ontology graph generated from different ontology embeddings (Opa2vec [46], Onto2vec [4], EL [5] and DL2vec [82]) and varying parameter K specifying the number of neighbours in the generated graph (10, 30, 100 or 500). The topology that performed best is using DL2vec embeddings and K=30.

Comparing node embeddings methods

Table 4.6 presents the comparison of the performance achieved by the models with various proposed methods of node embeddings in the graph neural network (see Figure 4.3). The results show that the best accuracy is achieved by the novel method of generating node embedding via combining ontology embeddings with expression values using a linear layer. Comparing this method with the same model that was using randomised values for the embeddings, shows that there exists a benefit of using biologically relevant embeddings for the genes. Such a model can distinguish well between the different nodes, simultaneously capturing the similarity between the genes (nodes) that are close in the ontology space.

Second best score was achieved by the model that does not use node embeddings but instead takes the expression values directly as the input features. The models that used the scaling (multiplication) operation in the process of obtaining node embeddings performed worse. This can be due to the fact that the scaling operation via noisy gene embeddings can obfuscate the input expression data.

	PAM 50	
	train size=100	train size=1500
A - no embeddings (pure expression values)	72.3±2.9	81.2±0.8
B - learnt embeddings scaled by expression	71.8±3.2	81.0±0.6
C - ontology embeddings scaled by expression	70.3±3.5	80.2±3.6
D - ontology embeddings with linear layer	72.7±3.7	81.6±2.2
D' - random embeddings with linear layer	71.5±2.6	79.3±3.0

Table 4.6: Performance comparison of the node embeddings methods (see Figure 4.3) on the PAM50 classification task (accuracy ± std). The GCN in all cases operates on the graph produced from prior knowledge from ontologies.

4.2.3 Results when using pre-selected 1000 genes

To control the study and validate the impact of the dimensionality of the data on the performance, we evaluate the models on a pre-selected set of the input mRNA features from Curtis et al. [6]. The set uses the pre-selected 1000 genes that are most significantly associated with DNA copy number alterations (CNAs).

According to the results in Table 4.7, in the tasks of PAM50 and ER the model performs better when the set of all genes is provided rather than the pre-selected 1000 from Curtis et al. [6]. The opposite is happening for iC10 where the pre-selected 1000 genes perform significantly better. The reason for this is that the considered 1000 genes were explicitly selected to define the iC10 clusters on the METABRIC [6], so by definition they are the best set to decide this particular clustering. While for the other targets, they may not contain all the relevant data for the classification.

		PAM50		ER		IC10	
train size		100	1500	100	1500	100	1500
1000 genes	RF	66.3±1.5	74.0±2.0	88.5±1.2	91.2±1.0	64.5±1.6	77.4±2.4
	MLP	58.3±9.5	76.3±2.6	77.4±0.3	92.3±0.9	63.0±2.3	80.0±1.8
	GCN	66.2±1.2	75.5±1.5	90.4±1.0	91.9±1.2	66.4±2.9	82.9±1.6
all genes	GCN	72.3±2.9	81.2±0.8	91.4±1.0	93.7±1.0	48.2±3.1	74.3±2.3

Table 4.7: Performance comparison of the methods on the ER, iC10 and DR classification task, using all genes as input features (in case of GCN, the intersection of genes present in the dataset and graph). Trained either on a training sample of 100 examples or 1500 examples.

4.3 Combining GCN with feature selection

This section evaluates the pipeline that combines the proposed graph neural network model with our ontology-based feature selection strategy proposed in Chapter 3. The experiments compare the performance of the GCN and MLP models on the varying set of input features selected either by the proposed feature selection or at random.

The results in Figure 4.4 for PAM50 and iC10 classification show that the feature selection is important when using a smaller set of genes of cardinality

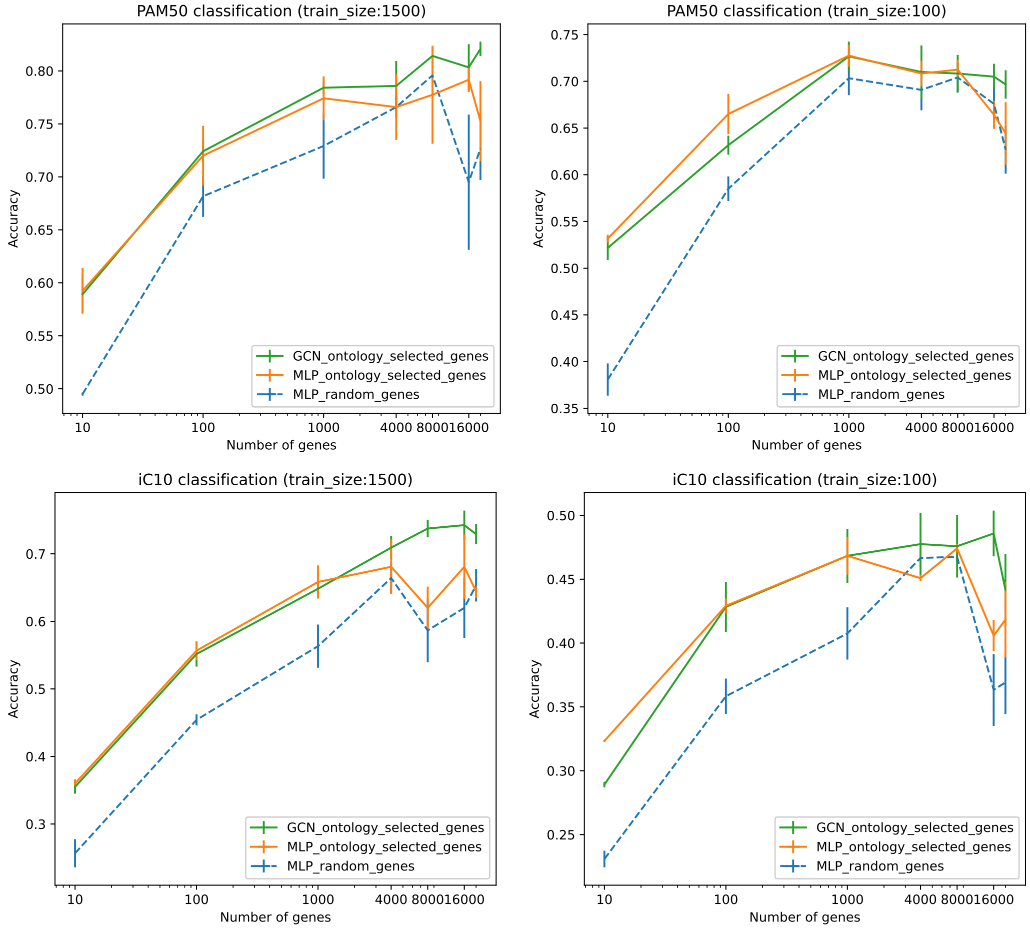

Figure 4.4: Accuracy obtained on the PAM50 and iC10 breast cancer subtype classification tasks. The reported accuracy is an average over 5 trials (with different stratified split of data) and error bars represent standard error.

below 4000. Given the genes chosen with the proposed feature selection strategy, both MLP and GCN outperform the baseline of MLP operating on a random set of genes. The differences are less significant on the larger gene set covering most of the genes as the models can then discover signals distributed among a variety of genes.

Our proposed GCN model with structural inductive bias based on prior knowledge performs on par with MLP on the smaller selected set of genes. However, it can outperform the MLP in the situation when the whole high-dimensional input of all 20000 genes is considered. In such a situation the MLP (green) performance drops while GCN (yellow) continues to improve with a larger gene set when given a large enough training sample. For PAM50 on scarce training samples, the performance of GCN and MLP reached its peak around an input set of 1000 genes. After that MLP performance declines while GCN is still performing well.

Such results suggest that imposing prior knowledge in the model in the form of structural inductive biases is beneficial when processing highly-dimensional data. It may potentially prove especially useful for the clinical tasks when little is known about which gene expressions can be used for predictions so

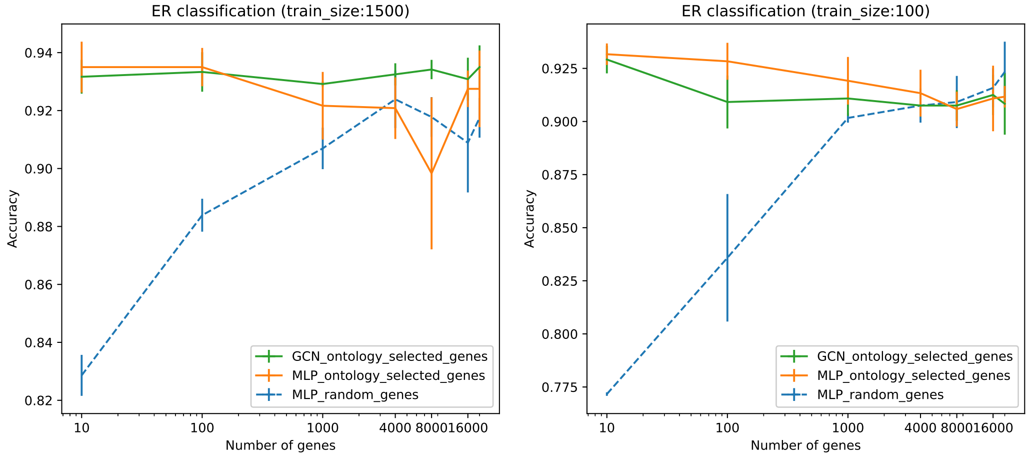

Figure 4.5: Accuracy obtained on the ER status classification task. The reported accuracy is an average over 5 trials (with different stratified split of data) and error bars represent standard error.

using all available data is desired.

ER is a task in which high accuracy can be achieved with already a small set of 10 genes. As seen in Figure 4.5, great improvements can be made by using proposed feature selection strategies. Due to the simplicity of this binary classification, the performance is not increasing as the dimensional of input is increased.

The results suggest a recommended treatment depending on the dataset size and dimensionality. GCN brings gains if the task requires using the set of all available genes. Then the structural inductive bias limiting spurious connections proves beneficial. But if there is enough knowledge about the task to do feature selection, then it is better to limit the number of inputs and apply a simpler machine learning model

4.4 Identifying high-weight genes for interpretability

The proposed model is interpretable in the sense that we can still understand how each gene is contributing to the prediction model. While in the MLP model, the hidden units do not have assigned meaning, in the proposed GCN model each node corresponds to some gene, making the model transparent, as the information from each node is only shared and mixed with the n-hop neighbouring genes (in case of the single layer as used in most of the evaluation, these are just direct neighbours).

By analysing the weights in the learnt parameters, one can gain better understanding of the behaviour of machine learning model. These high weight connections are identified using the weight matrices in the last layer that maps embedded genes to the output class. For each classification label, the genes are sorted by the absolute value of their weight. The high-weight genes are the genes with the highest of such weights. Note, that the node values are influenced by its neighbours, but one can also easily track the weights of the neighbour updates (which are orders of magnitude lower), especially in

PAM50 subtbe	High weight genes
Normal	**KRT17**, **KRT14**, KRT6B, **SFRP1**, CCNB1IP1, GDF6, **KRT5**, FOS, LAMB3, CX3CL1
LumA	SHROOM2, OR10A7, EPHA6, LCE4A, ZNF276, SHISA9, ZNF415, KLHDC1, MRGPRX2, SLC18A3
LumB	**KRT17**, CCNA2, SPC25, PBK, FEZF1, ASPM, **CEP55**, **BIRC5**, **ESR1**, EPHA6
Basal	DNAJB7, FOLR1, TXNDC8, USP51, NKX6-1, CRYAB, P2RY6, **SFRP1**, LIMD1, ZNF749
Her2	RB1CC1, **GRB7**, TFAP2B, SLC18A3, AKAP3, PGAP3, CELSR3, **FGFR4**, ARHGAP10, BCL7A

Table 4.8: High weight genes obtained by analyzing the connections in last linear layer of GCN for predicting the PAM50 breast cancer subtype. Bold font signifies genes present in the standard 50-gene subtype predictor [7].

the used model that has only one layer.

Performing such analysis of the single-layer GCN model trained to predict PAM50 subtype, resulted in identifying high-weight genes for each of the five breast-cancer subtypes, listed in Table 4.8. Multiple identified genes are present in the standard 50-gene breast cancer subtype predictor [7]. This observation confirms that the model retains its transparency and can successfully identify the driver genes.

Additional analysis was performed by visualising the identified high weight genes in the ontology embeddings. We plotted the genes in the 2-dimensional space mapped by the t-SNE technique [93] applied to the EL embeddings [5].

t-SNE is an algorithm to visualise high-dimensional data, that preserves the property that close points remain close and distant points remain distant. It assigns each point with coordinates in a 2-dimensional map. Firstly, t-SNE builds a probability distribution over the pairs of high dimensional data points, based on the similarity between them. Secondly, it builds an analogous probability distribution over the points in the 2-dimensional map. Then, the algorithm minimises the Kullback-Leibler divergence in the joint probabilities of the high dimensional points and low-dimensional embeddings.

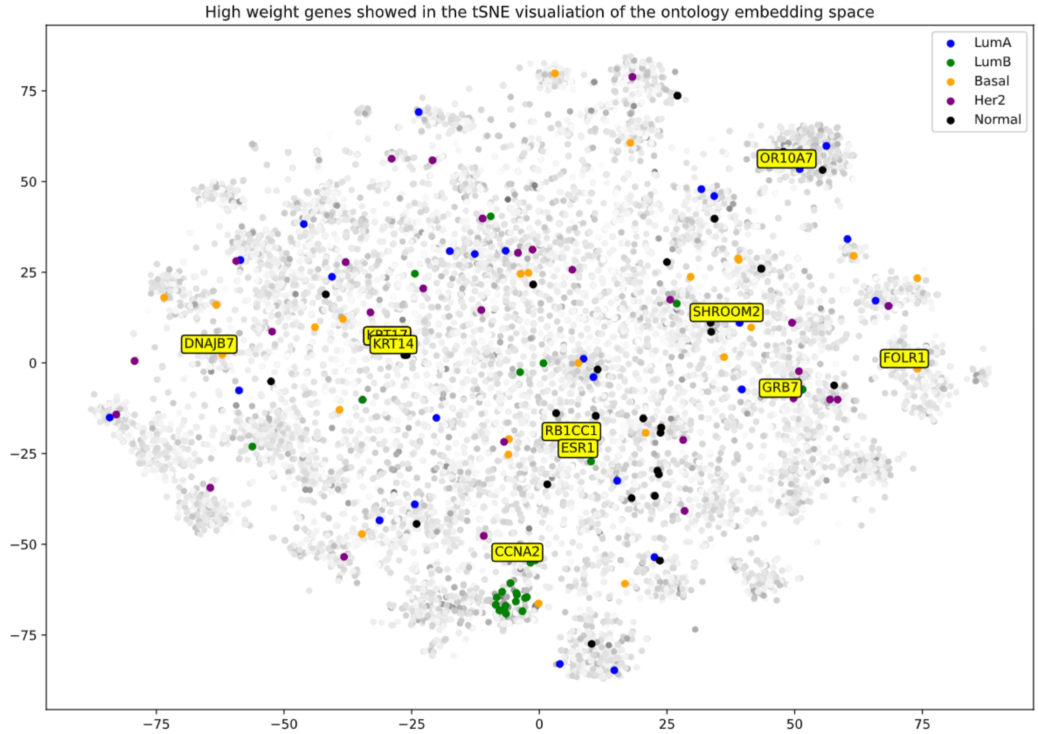

Figure 4.6: t-SNE visualisation of the ontology embedding space. The coloured points mark the top 30 high-weight genes in the PAM50 classification task, for each of the breast-cancer subtypes.

t-SNE is often used in visualising the gene expression data [94, 60] where points representing samples (patients or cells) are placed in a 2D plane according to their expression values. This analysis shows a different approach, as the points in the 2D plane represent genes and are placed according to their ontology embeddings.

The Figure 4.6 presents a t-SNE visualisation of such gene embedding space. It highlights the top 30 high-weight genes for each of the PAM50 subtypes labels (according to logits in the output layer). The annototations show the names of some of such top genes.

We can observe that most of the important genes were scattered around different regions but multiple of high-weight genes for LumB ended up clustering together. This means that genes that are good for classifying LumB subtype are close to each other in the gene ontology.

To investigate this finding we run the GO enrichment analysis [95, 96, 97] on the list of top 500 genes ranked by their respective weights in the trained model for LubB subtype. GO enrichment is a statistical analysis that identifies significant shared Gene Ontology terms (or parents of GO terms) used to describe the set of genes.

The GO enrichment identified `GO:1903047`-mitotic cell cycle and `GO:0022402`-cell cycle process process to have p-value below 10^{-10}. It possible that anomalies (or lack thereof) of expression values of these genes related to mitotic and nuclear division are predictors of high proliferation that often characterises LumB subtype [98].

Running the Gene-Disease Association analysis [96, 97] on this set of genes identified breast cancer as the related disorder, which confirms that the process of identification of high-weight genes in the model is capable of selecting cancer driver genes out of the set of over 20'000 genes.

Similar experiments for ER+ and ER- showed that the high-weight genes were not particularly clustered in the ontology space and they were clearly separated from each other.

The analogous analysis could be performed for any other clinical target, dataset and also a subset of genes, providing insight for cancer researchers. Having more interpretable models where relevant genes are placed withing the embedding space of ontologies could allow biologists to identify novel biomarkers or focus on specific subsets of genes. Interpreting the models and analysing the learnt connections could also help in generating new hypotheses that may be validated with experiments.

4.5 Summary

In this chapter, we presented a graph neural network model that exploits prior biological knowledge to analyse gene expression data and improves performance in the cancer patient classification tasks. The model uses ontology embeddings [46, 82] to generate a graph topology that directs the convolution operations in the network. By exploiting prior knowledge it aims to mitigate the difficulties of high-dimensionality, by limiting the number of connections in neural network architecture.

Then, we performed a comparative evaluation of the GCN model and its variants, against the baselines of MLP and Random Forest, on the cancer patient classification tasks of PAM50, ER and iC10. The achieved results suggest that the proposed GCN model can outperform the baselines and prove useful in the case of low-sample high-dimensional data. Combining the model with the previously presented feature selection methodology from the previous chapter showed the insight into the patient classification task depending on the number of genes used, and allowed to devise recommendations for the gene expression data analysis.

The recommendation is that the graph neural network approach improves performance when one wants to use the set of all available genes. Then the structural inductive bias limiting spurious connections proves beneficial. However, if there is enough knowledge about the task to do feature selection, then it may be better to limit the number of inputs and apply a simpler machine learning model.

Finally, the analysis of the weights learned by the model provided an insight into the beliefs learnt by the network. It has shown that the model is interpretable which is crucial in the domains of biomedicine and genetics where the predictions need to be explainable.

Chapter 5

Conclusions

This MPhil project had the overall goal of exploring how one could exploit the prior knowledge about genes from ontologies to improve machine learning on gene expression data. The project achieved its aim and resulted in proposing methods that use the knowledge entailed in ontologies to select the relevant genes for analysis or direct the processing inside a deep neural network. The proposed methods have the potential to help research on the biological processes and overcome the curse of high dimensionality causing difficulty in using deep learning models on gene expression data.

We proposed two frameworks exploiting ontology embeddings obtained from previous work, as a semantic similarity measure between the genes. To the best of our knowledge, this is the first work to use ontology-based gene representations to enhance learning on gene expression data. The developed methods were successfully applied on clinical tasks of breast-cancer patient classification, on a dataset from the METABRIC initiative [6].

In Chapter 3, we demonstrated that ontology embeddings can be used for knowledge-driven feature selection of the genes. Experiments, both on the clinical task and a single gene inference task, showed such selection performs better than randomly selected genes (or pre-selected set of genes), indicating the value of using the prior knowledge from vector representations of genes.

In Chapter 4, we demonstrated that graphs representing the ontology-based similarity of genes can be utilised in a deep model with Graph Convolutional Networks and that they perform well on the clinical tasks when compared to MLP or random forest models. Moreover, we demonstrated that the model is transparent, as it allows for easy identification of the genes driving the prediction, and therefore could be potentially used to identify novel cancer biomarkers.

Moreover, we showed that these two approaches can be combined in a single system. The extensive experiments on the clinical tasks that considered various training set sizes and a number of input genes allowed to devise a recommendation that in case of a scarce dataset, one should restrict the number of genes used as an input, provided that there is some prior knowledge about potential driver genes. Otherwise, if the whole set of genes is to be used, it is worth using the structural inductive bias to limit the number of connections and overcome the problems of highly-dimensional noisy data.

The developed methods were exemplified on the tasks of breast-cancer patient classification, however they are general and could be applied to any patient classification task and gene expression dataset. The performance of the proposed GCN model is robust in the case of high-dimensional data using over 20'000 of genes as the input. Hence, it can be potentially applied for the tasks where little is known about which genes influence the target.

5.1 Future work

The ideas developed in this dissertation pave the way to further exploration of imposing biological biases for analysing gene expression data. Some of the possible directions include:

- Evaluating the performance of the described methods on other, possibly new, datasets of gene expressions. It would be particularly interesting to evaluate the method on clinical targets derived from heterogeneous data sources (e.g., imaging data as well as genetic data).

- Devising a custom message passing mechanism in the graph neural network that provides a proper separation of the expression value and node (gene) embedding. This could involve developing a custom attention mechanism over the knowledge-driven gene embeddings that would dictate the connection weights between the genes [28].

- Similar architectures could be used for an inverse task of deriving a new data-driven biological network based on the connection weights learned in graph neural network. One approach could use a densely connected graph structure between gene entities, where the connections between genes would be learned and reinforced during training, similarly as in the literature on self-attention transformers [99].

Bibliography

[1] Petar Veličković. *The resurgence of structure in deep neural networks.* PhD thesis, University of Cambridge, 2019.

[2] Alvis Brazma, Pascal Hingamp, John Quackenbush, Gavin Sherlock, Paul T. Spellman, Chris Stoeckert, John Aach, Wilhelm J. Ansorge, Catherine A. Ball, Helen C. Causton, et al. Minimum information about a microarray experiment (miame)—toward standards for microarray data. *Nature Genetics*, 29:365–371, 2001.

[3] David Binns, Emily Dimmer, Rachael Huntley, Daniel Barrell, Claire O'donovan, and Rolf Apweiler. Quickgo: a web-based tool for gene ontology searching. *Bioinformatics*, 25(22):3045–3046, 2009.

[4] Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Onto2vec: joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics*, 34(13):52–60, 2018.

[5] Maxat Kulmanov, Wang Liu-Wei, Yuan Yan, and Robert Hoehndorf. El embeddings: geometric construction of models for the description logic el++. *arXiv preprint arXiv:1902.10499*, 2019.

[6] Christina Curtis, Sohrab P Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M Rueda, Mark J Dunning, Doug Speed, Andy G Lynch, Shamith Samarajiwa, Yinyin Yuan, et al. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352, 2012.

[7] Joel S Parker, Michael Mullins, Maggie CU Cheang, Samuel Leung, David Voduc, Tammi Vickery, Sherri Davies, Christiane Fauron, Xiaping He, Zhiyuan Hu, et al. Supervised risk predictor of breast cancer based on intrinsic subtypes. *Journal of clinical oncology*, 27(8):1160, 2009.

[8] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the 5th International Conference on Learning Representations*, ICLR'17, 2017.

[9] Francis Dutil, Joseph Paul Cohen, Martin Weiss, Georgy Derevyanko, and Yoshua Bengio. Towards gene expression convolutions using gene interaction graphs. In *International Conference on Machine Learning (ICML) Workshop on Computational Biology (WCB)*, 2018.

[10] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[11] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

[12] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees.* CRC press, 1984.

[13] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[14] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification.* John Wiley & Sons, 2012.

[15] Necla Gunduz and Ernest Fokoué. Robust classification of high dimension low sample size data. *arXiv preprint arXiv:1501.00592*, 2015.

[16] Torgyn Shaikhina, Dave Lowe, Sunil Daga, David Briggs, Robert Higgins, and Natasha Khovanova. Decision tree and random forest models for outcome prediction in antibody incompatible kidney transplantation. *Biomedical Signal Processing and Control*, 52:456–462, 2019.

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning.* MIT press, 2016.

[18] Dave Anderson and George McNeill. Artificial neural networks technology. *Kaman Sciences Corporation*, 258(6):1–83, 1992.

[19] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

[20] Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinícius Flores Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Rela-

tional inductive biases, deep learning, and graph networks. *CoRR*, abs/1806.01261, 2018.

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[22] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[23] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

[24] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[25] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[26] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE, 2005.

[27] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

[28] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018.

[29] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard S. Zemel. Gated graph sequence neural networks. *CoRR*, abs/1511.05493, 2016.

[30] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1263–1272, 2017.

[31] Francis Crick. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[32] Alex Sánchez and MC de Villa. A tutorial review of microarray data analysis. *Universitat de Barcelona*, 2008.

[33] Alvis Brazma and Jaak Vilo. Gene expression data analysis. *FEBS letters*, 480(1):17–24, 2000.

[34] Ka Yee Yeung and Walter L. Ruzzo. Principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9):763–774, 2001.

[35] Shoba Ranganathan, Kenta Nakai, and Christian Schonbach. *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics.* Elsevier, 2018.

[36] Adriana Romero, Pierre Luc Carrier, Akram Erraqabi, Tristan Sylvain, Alex Auvolat, Etienne Dejoie, Marc-André Legault, Marie-Pierre Dube, Julie G. Hussin, and Yoshua Bengio. Diet networks: Thin parameters for fat genomics. In *International Conference on Learning Representations*, 2017.

[37] Indu Ravi, Mamta Baunthiyal, and Jyoti Saxena. *Advances in biotechnology.* Springer, 2014.

[38] Lisa Ehrlinger and Wolfram Wöß. Towards a definition of knowledge graphs. *SEMANTiCS (Posters, Demos, SuCCESS)*, 48:1–4, 2016.

[39] Bess Schrader. What's the difference between an ontology and a knowledge graph? `https://enterprise-knowledge.com/whats-the-difference-between-an-ontology-and-a-knowledge-graph/`.

[40] Franz Baader, Diego Calvanese, Deborah McGuinness, Peter Patel-Schneider, Daniele Nardi, et al. *The description logic handbook: Theory, implementation and applications.* Cambridge university press, 2003.

[41] David P Hill, Barry Smith, Monica S McAndrews-Hill, and Judith A Blake. Gene ontology annotations: what they mean and where they come from. In *BMC bioinformatics*, volume 9, pages 1–9. BioMed Central, 2008.

[42] Maxat Kulmanov, Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Machine learning with biomedical ontologies. *bioRxiv*, 2020.

[43] Jake Crawford and Casey S Greene. Incorporating biological structure into machine learning models in biomedicine. *Current Opinion in Biotechnology*, 63:126–134, 2020.

[44] Marinka Zitnik, Francis Nguyen, Bo Wang, Jure Leskovec, Anna Goldenberg, and Michael M Hoffman. Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities. *Information Fusion*, 50:71–91, 2019.

[45] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings*, 2013.

[46] Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Opa2vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics*, 35(12):2133–2140, 2019.

[47] Christopher J Mungall, Carlo Torniai, Georgios V Gkoutos, Suzanna E Lewis, and Melissa A Haendel. Uberon, an integrative multi-species anatomy ontology. *Genome biology*, 13(1):R5, 2012.

[48] Cynthia L Smith and Janan T Eppig. The mammalian phenotype ontology: enabling robust annotation and comparative analysis. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 1(3):390–399, 2009.

[49] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, Carsten Lutz, et al. Owl 2 web ontology language profiles. *W3C recommendation*, 27:61.

[50] Bernardo Cuenca Grau, Ian Horrocks, Boris Motik, Bijan Parsia, Peter Patel-Schneider, and Ulrike Sattler. Owl 2: The next step for owl. *Journal of Web Semantics*, 6(4):309–322, 2008.

[51] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017.

[52] Wei Zhang, Jeremy Chien, Jeongsik Yong, and Rui Kuang. Network-based machine learning and graph theory algorithms for precision oncology. *npj Precision Oncology*, 1(1):25, Aug 2017.

[53] K Yu Michael, Jianzhu Ma, Jasmin Fisher, Jason F Kreisberg, Benjamin J Raphael, and Trey Ideker. Visible machine learning for biomedicine. *Cell*, 173(7):1562–1565, 2018.

[54] Jianzhu Ma, Michael Ku Yu, Samson Fong, Keiichiro Ono, Eric Sage, Barry Demchak, Roded Sharan, and Trey Ideker. Using deep learning to model the hierarchical structure and function of a cell. *Nature methods*, 15(4):290, 2018.

[55] Nikolaus Fortelny and Christoph Bock. Knowledge-primed neural networks enable biologically interpretable deep learning on single-cell sequencing data. *bioRxiv*, page 794503, 2019.

[56] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NeurIPS'01, pages 585–591. MIT Press, 2001.

[57] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NeurIPS'16, pages 3844–3852. Curran Associates Inc., 2016.

[58] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1263–1272, 2017.

[59] Lenore Cowen, Trey Ideker, Benjamin J. Raphael, and Roded Sharan. Network propagation: a universal amplifier of genetic associations. *Nature Reviews Genetics*, 18(9):551–562, Sep 2017.

[60] Sungmin Rhee, Seokjun Seo, and Sun Kim. Hybrid approach of relation network and localized graph convolutional filtering for breast cancer subtype classification. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 3527–3534. AAAI Press, 2018.

[61] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems 30*.

[62] David Warde-Farley, Sylva L Donaldson, Ovi Comes, Khalid Zuberi, Rashad Badrawi, Pauline Chao, Max Franz, Chris Grouios, Farzana Kazi, Christian Tannus Lopes, et al. The genemania prediction server: biological network integration for gene prioritization and predicting gene function. *Nucleic acids research*, 38(suppl_2):W214–W220, 2010.

[63] Paul Bertin, Mohammad Hashir, Martin Weiß, Geneviève Boucher, Vincent Frappier, and Joseph Paul Cohen. Analysis of gene interaction graphs for biasing machine learning models. *arXiv: Genomics*, abs/1905.02295, 2019.

[64] Mohammad Hashir, Paul Bertin, Martin Weiss, Vincent Frappier, Theodore Perkins, Geneviève Boucher, and Joseph Paul Cohen. Is graph biased feature selection of genes better than random? *ArXiv*, abs/1910.09600, 2019.

[65] Jake Crawford and Casey S. Greene. Graph biased feature selection of genes is better than random for many genes. *bioRxiv*, 2020.

[66] John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart, Cancer Genome Atlas Research Network, et al. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113, 2013.

[67] Mandana Samiei, Tobias Würfl, Tristan Deleu, Martin Weiss, Francis Dutil, Thomas Fevens, Geneviève Boucher, Sebastien Lemieux, and Joseph Paul Cohen. The tcga meta-dataset clinical benchmark. *arXiv preprint arXiv:1910.08636*, 2019.

[68] Runpu Chen, Le Yang, Steve Goodison, and Yijun Sun. Deep-learning approach to identifying cancer subtypes using high-dimensional genomic data. *Bioinformatics*, 36(5):1476–1483, 10 2019.

[69] Simona Maria Fragomeni, Andrew Sciallis, and Jacqueline S Jeruss. Molecular subtypes and local-regional control of breast cancer. *Surgical Oncology Clinics*, 27(1):95–120, 2018.

[70] Early Breast Cancer Trialists' Collaborative Group et al. Effects of chemotherapy and hormonal therapy for early breast cancer on recurrence and 15-year survival: an overview of the randomised trials. *The Lancet*, 365(9472):1687–1717, 2005.

[71] Bette J Caan, Carol Sweeney, Laurel A Habel, Marilyn L Kwan, Candyce H Kroenke, Erin K Weltzien, Charles P Quesenberry, Adrienne Castillo, Rachel E Factor, Lawrence H Kushi, et al. Intrinsic subtypes from the pam50 gene expression assay in a population-based breast cancer survivor cohort: prognostication of short-and long-term outcomes. *Cancer Epidemiology and Prevention Biomarkers*, 23(5):725–734, 2014.

[72] Hege G Russnes, Ole Christian Lingjærde, Anne-Lise Børresen-Dale, and Carlos Caldas. Breast cancer molecular stratification: from intrinsic subtypes to integrative clusters. *The American journal of pathology*, 187(10):2152–2162, 2017.

[73] Bernd Holleczek, Christa Stegmaier, Julia C Radosa, Erich-Franz Solomayer, and Hermann Brenner. Risk of loco-regional recurrence and distant metastases of patients with invasive breast cancer up to ten years after diagnosis–results from a registry-based study from germany. *BMC cancer*, 19(1):520, 2019.

[74] J Zyprych-Walczak, A Szabelska, Luiza Handschuh, K Górczak, K Klamecka, Marek Figlerowicz, and I Siatkowski. The impact of normalization methods on rna-seq data analysis. *BioMed research international*, 2015, 2015.

[75] Mohammad Hashir, Paul Bertin, Martin Weiss, Vincent Frappier, Theodore Perkins, Geneviève Boucher, and Joseph Paul Cohen. Is graph biased feature selection of genes better than random? *arXiv preprint arXiv:1910.09600*, 2019.

[76] Fan Li and Yiming Yang. Analysis of recursive gene selection approaches from microarray data. *Bioinformatics*, 21(19):3741–3747, 2005.

[77] Jianping Hua, Waibhav D Tembe, and Edward R Dougherty. Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recognition*, 42(3):409–424, 2009.

[78] Frederik Holst, Phillip R Stahl, Christian Ruiz, Olaf Hellwinkel, Zeenath Jehan, Marc Wendland, Annette Lebeau, Luigi Terracciano, Khawla Al-Kuraya, Fritz Jänicke, et al. Estrogen receptor alpha (esr1) gene amplification is frequent in breast cancer. *Nature genetics*, 39(5):655–660, 2007.

[79] Steffi Oesterreich and Nancy E Davidson. The search for esr1 mutations in breast cancer. *Nature genetics*, 45(12):1415–1416, 2013.

[80] R Dawkins. *The Selfish Gene*. Oxford University Press, Oxford, UK, 1976.

[81] Roy RL Bastien, Álvaro Rodríguez-Lescure, Mark TW Ebbert, Aleix Prat, Blanca Munárriz, Leslie Rowe, Patricia Miller, Manuel Ruiz-Borrego, Daniel Anderson, Bradley Lyons, et al. Pam50 breast cancer

subtyping by rt-qpcr and concordance with standard clinical molecular markers. *BMC medical genomics*, 5(1):44, 2012.

[82] Jun Chen, Azza Th Althagafi, and Robert Hoehndorf. Predicting candidate genes from phenotypes, functions, and anatomical site of expression. *BioRxiv*, 2020.

[83] Donna Maglott, Jim Ostell, Kim D Pruitt, and Tatiana Tatusova. Entrez gene: gene-centered information at ncbi. *Nucleic acids research*, 39(suppl_1):D52–D57, 2010.

[84] Rhoda J Kinsella, Andreas Kähäri, Syed Haider, Jorge Zamora, Glenn Proctor, Giulietta Spudich, Jeff Almeida-King, Daniel Staines, Paul Derwent, Arnaud Kerhornou, et al. Ensembl biomarts: a hub for data retrieval across taxonomic space. *Database*, 2011, 2011.

[85] Andrew D Yates, Premanand Achuthan, Wasiu Akanni, James Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Andrey G Azov, Ruth Bennett, et al. Ensembl 2020. *Nucleic acids research*, 48(D1):D682–D688, 2020.

[86] Nikola Simidjievski, Cristian Bodnar, Ifrah Tariq, Paul Scherer, Helena Andres Terre, Zohreh Shams, Mateja Jamnik, and Pietro Liò. Variational autoencoders for cancer data integration: design principles and computational practice. *Frontiers in genetics*, 10:1205, 2019.

[87] Yifei Chen, Yi Li, Rajiv Narayan, Aravind Subramanian, and Xiaohui Xie. Gene expression inference with deep learning. *Bioinformatics*, 32(12):1832–1839, 2016.

[88] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1):D607–D613, 2019.

[89] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NeurIPS-W*, 2017.

[90] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al.

Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[91] Peter I Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

[92] Jake Crawford and Casey S Greene. Graph biased feature selection of genes is better than random for many genes. *BioRxiv*, 2020.

[93] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[94] Dmitry Kobak and Philipp Berens. The art of using t-sne for single-cell transcriptomics. *Nature communications*, 10(1):1–14, 2019.

[95] Eran Eden, Roy Navon, Israel Steinfeld, Doron Lipson, and Zohar Yakhini. Gorilla: a tool for discovery and visualization of enriched go terms in ranked gene lists. *BMC bioinformatics*, 10(1):48, 2009.

[96] Brad T Sherman, Richard A Lempicki, et al. Systematic and integrative analysis of large gene lists using david bioinformatics resources. *Nature protocols*, 4(1):44, 2009.

[97] Da Wei Huang, Brad T Sherman, and Richard A Lempicki. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic acids research*, 37(1):1–13, 2009.

[98] Claudette Falato, Nicholas P Tobin, Julie Lorent, Linda S Lindström, Jonas Bergh, and Theodoros Foukakis. Intrinsic subtypes and genomic signatures of primary breast cancer and prognosis after systemic relapse. *Molecular oncology*, 10(4):517–525, 2016.

[99] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[100] Ioana Bica. Unsupervised neural methods for modelling cell differentiation. Master's thesis, University of Cambridge, 2018.