2019

NOVEL APPLICATIONS OF MACHINE LEARNING IN BIOINFORMATICS

Yi Zhang
University of Kentucky, zhangyimc@gmail.com
Digital Object Identifier: https://doi.org/10.13023/etd.2019.261

STUDENT AGREEMENT:

I represent that my thesis or dissertation and abstract are my original work. Proper attribution has been given to all outside sources. I understand that I am solely responsible for obtaining any needed copyright permissions. I have obtained needed written permission statement(s) from the owner(s) of each third-party copyrighted matter to be included in my work, allowing electronic distribution (if such use is not permitted by the fair use doctrine) which will be submitted to UKnowledge as Additional File.

I hereby grant to The University of Kentucky and its agents the irrevocable, non-exclusive, and royalty-free license to archive and make accessible my work in whole or in part in all forms of media, now or hereafter known. I agree that the document mentioned above may be made available immediately for worldwide access unless an embargo applies.

I retain all other ownership rights to the copyright of my work. I also retain the right to use in future works (such as articles or books) all or part of my work. I understand that I am free to register the copyright to my work.

REVIEW, APPROVAL AND ACCEPTANCE

The document mentioned above has been reviewed and accepted by the student's advisor, on behalf of the advisory committee, and by the Director of Graduate Studies (DGS), on behalf of the program; we verify that this is the final, approved version of the student's thesis including all changes required by the advisory committee. The undersigned agree to abide by the statements above.

Yi Zhang, Student

Dr. Jinze Liu, Major Professor

Dr. Miroslaw Truszczynski, Director of Graduate Studies

NOVEL APPLICATIONS OF MACHINE LEARNING IN BIOINFORMATICS

DISSERTATION

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy in the
College of Engineering
at the University of Kentucky

By
Yi Zhang
Lexington, Kentucky
Director: Dr. Jinze Liu, Associate Professor of Computer Science
Lexington, Kentucky
2019

ABSTRACT OF DISSERTATION

NOVEL APPLICATIONS OF MACHINE LEARNING IN BIOINFORMATICS

Technological advances in next-generation sequencing and biomedical imaging have led to a rapid increase in biomedical data dimension and acquisition rate, which is challenging the conventional data analysis strategies. Modern machine learning techniques promise to leverage large data sets for finding hidden patterns within them, and for making accurate predictions. This dissertation aims to design novel machine learning-based models to transform biomedical big data into valuable biological insights. The research presented in this dissertation focuses on three bioinformatics domains: splice junction classification, gene regulatory network reconstruction, and lesion detection in mammograms.

A critical step in defining gene structures and mRNA transcript variants is to accurately identify splice junctions. In the first work, we built the first deep learning-based splice junction classifier, DeepSplice. It outperforms the state-of-the-art classification tools in terms of both classification accuracy and computational efficiency. To uncover transcription factors governing metabolic reprogramming in non-small-cell lung cancer patients, we developed TFmeta, a machine learning approach to reconstruct relationships between transcription factors and their target genes in the second work. Our approach achieves the best performance on benchmark data sets. In the third work, we designed deep learning-based architectures to perform lesion detection in both 2D and 3D whole mammogram images.

KEYWORDS: Machine Learning, Deep Learning, Splice Junction, RNA-seq, Cancer, Biomedical Imaging

Yi Zhang

06/26/2019

Date

NOVEL APPLICATIONS OF MACHINE LEARNING IN BIOINFORMATICS

By
Yi Zhang

Jinze Liu
Director of Dissertation

Miroslaw Truszczynski
Director of Graduate Studies

06/26/2019
Date

ACKNOWLEDGMENTS

I would like to express my appreciation to the people who helped and encouraged me during my Ph.D. journey.

First of all, I would like to express my deepest gratitude to my advisor, Dr. Jinze Liu, for her continuous support and encouragement throughout my Ph.D. study. She is not only a great researcher with immense knowledge, but also an excellent advisor that drives students to the right direction. Without her patient guidance, I wouldn't have been able to finish this dissertation.

Next, I would like to thank Dr. James N. MacLeod. He not only taught me a lot of knowledge in biology but also helped me learn to work with researchers of other backgrounds. I am also grateful to Dr. Jerzy W. Jaromczyk, Dr. Zongming Fei, and Dr. Chi Wang, for their invaluable comments and constructive suggestions on my dissertation.

I would also like to thank my friends in Lexington for making my life enjoyable. Last but not least, I thank my family for their unconditional love and support all these years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

1.1 Machine learning

Machine learning can be broadly described as computational methods using previous experience to improve performance or to make precise inferences. Here, previous experience refers to the past information available to the learner. This data could be in the form of digitized human-labeled training sets, or other types of information obtained via interaction with the environment. In all cases, its quality and size are crucial to the success of the predictions made by the learner [1].

The processes involved in machine learning are similar to that of data mining and predictive modeling. Both require searching through data to look for patterns and adjusting program actions accordingly. Many people are familiar with machine learning from shopping on the internet and being served ads related to their purchase. This happens because recommendation engines use machine learning to personalize online ad delivery in almost real time. Beyond personalized marketing, other common machine learning use cases include fraud detection, spam filtering, network security threat detection, predictive maintenance and building news feeds.

Machine learning algorithms are often categorized as supervised or unsupervised. Supervised algorithms require both input and desired output, in addition to furnishing feedback about the accuracy of predictions during algorithm training. Once training is complete, the algorithm will apply what was learned to new data. Unsupervised algorithms do not need to be trained with desired outcome data.

Since the success of a learning algorithm depends on the data used, machine learning is inherently related to data analysis and statistics. More generally, machine learning techniques are data-driven methods combining fundamental concepts in computer science with ideas from statistics, probability and optimization [1].

The standard machine learning tasks which have been extensively studied are listed as follows:

- Classification: classification is used when the outputs are restricted to a limited set of values. For a classification task that filters emails, the input would be an incoming email, and the output would be the name of the folder in which to file the email [2].
- Regression: regression is adopted to predict continuous outputs, that is, real values within a range. Examples of a continuous value are the temperature, length, or price of an object [2].
- Ranking: ranking is to produce a permutation of items in unseen lists in a way which is similar to rankings in the training data in some sense. Ranking is a central part of many information retrieval problems, such as document retrieval, sentiment analysis, and online advertising [3].
- Clustering: clustering is the task of grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups. For instance, in social network analysis, clustering algorithms try to identify natural communities within a large group of people [4].
- Dimensionality reduction: dimensionality reduction is to transform an initial representation of items into a lower-dimensional representation while preserving some

properties of the initial representation. A common example involves preprocessing digital images in computer vision tasks [1].

1.2 Deep learning

Deep learning is a branch of machine learning, which employs numerous similar but distinct deep neural network architectures to solve various problems in natural language processing, computer vision, and bioinformatics, among other fields. Deep learning has experienced tremendous recent research resurgence, and has been shown to deliver state of the art results in numerous applications.

In essence, deep learning is the implementation of neural networks with more than a single hidden layer of neurons, as shown in Figure 1.1. However, this is a very simplistic view of deep learning, and not one that is unanimously agreed upon. These "deep" architectures also vary quite considerably, with different implementations being optimized for different tasks or goals. The vast research being produced at such a constant rate is revealing new and innovative deep learning models at an ever-increasing pace.

The successes of deep learning are built on a foundation of significant algorithmic details and generally can be understood in two parts: construction and training of deep learning architectures [5]. Deep learning architectures are basically artificial neural networks of multiple non-linear layers and several types have been proposed according to input data characteristics and research objectives. Here, we categorized deep learning architectures into four groups: deep neural networks (DNNs), convolutional neural networks (CNNs), recurrent neural networks (RNNs) and emergent architectures. DNNs include multilayer perceptron (MLP), stacked auto-encoder (SAE) and deep belief

networks (DBNs), which use perceptrons, auto-encoders and restricted Boltzmann machines as the building blocks of neural networks, respectively. CNNs are architectures that have succeeded particularly in image recognition and consist of convolution layers, non-linear layers and pooling layers. RNNs are designed to utilize sequential information of input data with cyclic connections among building blocks like perceptrons, long short-term memory units or gated recurrent units. In addition, many other emergent deep learning architectures have been suggested, such as deep spatio-temporal neural networks (DST-NNs), multidimensional recurrent neural networks (MD-RNNs) and convolutional auto-encoders (CAEs).

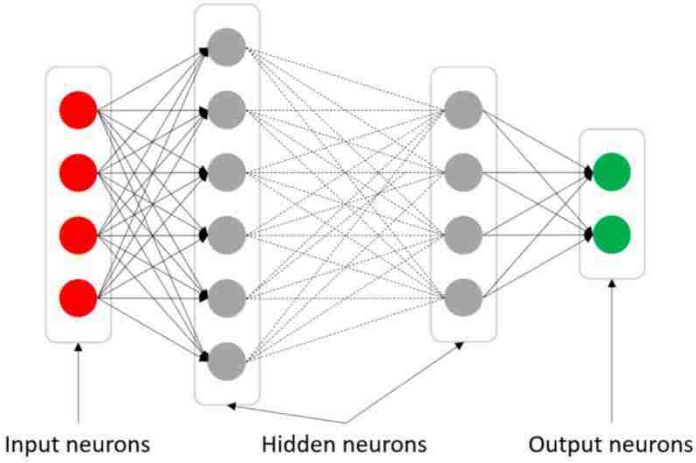

Figure 1.1 Illustration of the deep neural network layers.

The goal of training deep learning architectures is optimization of the weight parameters in each layer, which gradually combines simpler features into complex features so that the most suitable hierarchical representations can be learned from data. A single cycle of the optimization process is organized as follows. First, given a training dataset,

the forward pass sequentially computes the output in each layer and propagates the function signals forward through the network. In the final output layer, an objective loss function measures error between the inferred outputs and the given labels. To minimize the training error, the backward pass uses the chain rule to backpropagate error signals and compute gradients with respect to all weights throughout the neural network. Finally, the weight parameters are updated using optimization algorithms based on stochastic gradient descent (SGD). Whereas batch gradient descent performs parameter updates for each complete dataset, SGD provides stochastic approximations by performing the updates for each small set of data examples. Several optimization algorithms stem from SGD. For example, Adagrad and Adam perform SGD while adaptively modifying learning rates based on update frequency and moments of the gradients for each parameter, respectively.

Another core element in the training of deep learning architectures is regularization, which refers to strategies intended to avoid overfitting and thus achieve good generalization performance. For example, weight decay, a well-known conventional approach, adds a penalty term to the objective loss function so that weight parameters converge to smaller absolute values. Currently, the most widely used regularization approach is dropout. Dropout randomly removes hidden units from neural networks during training and can be considered an ensemble of possible subnetworks. Furthermore, recently proposed batch normalization provides a new regularization method through normalization of scalar features for each activation within a mini-batch and learning each mean and variance as parameters.

1.3 Machine learning in bioinformatics

In the era of big data, the rapid increase in biomedical data dimension and acquisition rate is challenging conventional analysis strategies. Modern machine learning methods, such as deep learning, promise to leverage large data sets for finding hidden structure within them, and for making accurate predictions.

The potential of machine learning in analyzing biomedical data sets is clear: in principle, it allows to better exploit the availability of increasingly large and high-dimensional data sets by training complex models that capture their internal structure. The learned models discover high-level features, increase interpretability and provide additional understanding about the structure of the biomedical data [6].

Recent papers are trying to apply machine learning to omics, biomedical imaging, electronic health record, and numerous other bioinformatics domains.

1.3.1 Omics

Improvements in technology have fueled the proliferation of omics applications. These techniques are often used to measure and study complex biological systems and their interactions. Omics includes a multitude of areas of focus such as genomics, transcriptomics, proteomics, interactomics, metabolomics, phenomics, and pharmacogenomics to name, but a few. Each one of these areas might also have many subdomains, each requiring further specialization in analytical and computational approaches [7].

Increasingly, the scale of omics data generation has been challenging researchers' abilities to integrate and model often noisy, complex, and high-dimensional data. Machine

learning has emerged as a powerful approach, which can both encode and model many forms of complex data both in supervised and unsupervised settings. DeepBind has been built to predict the sequence specificities of DNA- and RNA-binding proteins by deep learning [8]. Chen *et al.* [9] designed a machine learning-based method, D-GEX, to infer the expression of target genes from the expression of landmark genes. Arvaniti *et al.* proposed CellCnn [10], a representation learning approach to detect rare cell subsets associated with disease using high-dimensional single-cell measurements. Ma *et al.* developed DCell [11], a visible neural networks embedded in the hierarchical structure of 2,526 subsystems comprising a eukaryotic cell. Trained on several million genotypes, DCell simulates cellular growth nearly as accurately as laboratory observations. Altae-Tran *et al.* [12] introduced a new deep-learning architecture, the iterative refinement long short-term memory, a modification of the matching-networks architecture and the residual convolutional network. The architecture allows for the learning of sophisticated metrics which can trade information between evidence and query molecules. The authors demonstrated that their architecture offers significant boosts in predictive power for a variety of problems meaningful for low-data drug discovery.

1.3.2 Biomedical imaging

Over the recent years, machine learning has had a tremendous impact on various fields in science. It has led to significant improvements in speech recognition and image recognition, it is able to train artificial agents that beat human players in Go and ATARI games, and it creates artistic new images, and music. Many of these tasks were considered

to be impossible to be solved by computers [13]. Obviously this technology is also highly relevant for biomedical imaging.

The advantage of machine learning in an era of biomedical imaging big data is that significant hierarchal relationships within the data can be discovered algorithmically without laborious hand-crafting of features. The key machine learning applications in biomedical imaging include image classification, localization and detection, segmentation, and image reconstruction. Esteva *et al.* [14] demonstrated the effectiveness of deep learning in dermatology, a technique applied to both general skin conditions and specific cancers. Using a single convolutional neural network trained on general skin lesion classification, the authors matched the performance of at least 21 dermatologists tested across three critical diagnostic tasks: keratinocyte carcinoma classification, melanoma classification and melanoma classification using dermoscopy. Chlebus *et al.* [15] developed a fully automatic method for liver tumor segmentation in CT images based on a 2D fully convolutional neural network with an object-based postprocessing step. Inspired by the sharp, high texture-quality images retrieved by GANs, and the high contrast of MR images, Mardani *et al.* [16] employed GANs for modeling the low-dimensional manifold of high-quality MR images. This framework can leverage the historical data for rapid and high diagnostic-quality image reconstruction from highly undersampled MR measurements.

1.3.3 Electronic health record

Over the past 10 years, hospital adoption of electronic health record (EHR) systems has skyrocketed, in part due to the Health Information Technology for Economic and

Clinical Health (HITECH) Act of 2009, which provided $30 billion in incentives for hospitals and physician practices to adopt EHR systems [17]. According to the latest report from the Office of the National Coordinator for Health Information Technology (ONC), nearly 84% of hospitals have adopted at least a basic EHR system, a 9-fold increase since 2008 [18]. Given the increasingly vast amount of patient data and the rise in popularity of machine learning approaches, there has also been an increase in the number of publications applying machine learning to EHR data for clinical informatics tasks which yield better performance than traditional methods and require less time-consuming preprocessing and feature engineering.

Miotto *et al.* [19] presented a novel unsupervised deep feature learning method to derive a general-purpose patient representation from EHR data that facilitates clinical predictive modeling. In particular, a three-layer stack of denoising autoencoders was used to capture hierarchical regularities and dependencies in the aggregated EHRs. The authors reported that their findings indicate that deep learning applied to EHRs can derive patient representations that offer improved clinical predictions, and could provide a machine learning framework for augmenting clinical decision systems. Rajkomar *et al.* [20] proposed a representation of patients' entire raw EHR records based on the Fast Healthcare Interoperability Resources (FHIR) format. The authors demonstrated that deep learning methods using their representation are capable of accurately predicting multiple medical events from multiple centers without site-specific data harmonization, and their models outperformed traditional, clinically-used predictive models in all cases. Chen *et al.* [21] developed NoteAid, a natural language processing system that links medical terms in EHR

notes to lay definitions targeted at or below the average adult literacy level to support patient EHR comprehension.

1.4 Dissertation statement

This dissertation aims to design novel machine learning-based models to transform biomedical big data into valuable biological insights. It covers three different but closely related bioinformatics domains of great importance, including: classification of splice junction sequences using convolutional neural networks, reconstruction of gene regulatory networks using gradient boosted trees, and detection of lesions in mammogram images using deep learning.

1.5 Contributions of this dissertation

We have developed a series of novel machine learning-based approaches for analyzing various biomedical data, including genomics data, transcriptomics data, and biomedical imaging data. The performance of each of those approaches is assessed using a number of simulated and real datasets. The experiments demonstrate their advantages on accuracy and efficiency compared to other state-of-the-art approaches. This dissertation may contribute to the following three areas.

- **Accurate classification of novel splice junctions derived from RNA-seq alignment** A model inferred from the sequences of annotated exon junctions that can then classify splice junctions derived from primary RNA-seq data has been developed. Our DeepSplice model is the first deep learning-based splice junction classifier. The performance of the model was evaluated and compared through

comprehensive benchmarking and testing, indicating a reliable performance and gross usability for classifying novel splice junctions derived from RNA-seq alignment. Our findings further indicate that valuable information is present in the nucleotide sequence local to the splice junction, data that conventional splice site prediction techniques discard.

- **Efficient reconstruction of gene regulatory networks using multi-omics data sets** Leveraging gradient boosted trees, a multi-omics approach to uncover TFs governing cancer metabolic reprogramming and reconstruct their interactions with metabolic enzymes has been designed. We demonstrated that TFmeta achieved state-of-the-art performance in recovering TF-target gene interactions on public benchmark data sets. We applied our model to non-small-cell lung cancer patients' data sets to predict TFs modulating the dysregulation of glycolysis in lung cancer, leveraging the pairing information of the samples and TF DNA binding activities. Eventually, we predicted a list of key TFs that may motivate the upregulation of glycolysis observed in tumor cells, some of which have been supported by literature evidence, and some of which were predicted as novel putative TFs in lung cancer.
- **Precise detection of lesions in 2D and 3D mammography images** We conducted the first work that study both 2D and 3D mammography images for breast cancer classification through deep learning. We evaluated ten different convolutional neural network architectures and concluded that combining both data augmentation and transfer learning methods with a convolutional neural network is the most effective in improving classification performance. Our work sheds light on how each type of data sets performs when trained independently. 2D and 3D images are

complementary to each other, where 2D offers high resolution while 3D offers multiple views. Our work suggests the development of assembled classifiers that integrate the 2D and 3D data to achieve optimal performance.

The software packages for the algorithms developed in this dissertation are open source and publicly available to the research community.

CHAPTER 2. DISCERNING NOVEL SPLICE JUNCTIONS REVEALED BY RNA-SEQ WITH DEEPSPLICE

Exon splicing is a regulated cellular process in the transcription of protein-coding genes. Technological advancements and cost reductions in RNA sequencing have made quantitative and qualitative assessments of the transcriptome both possible and widely available. RNA-seq provides unprecedented resolution to identify gene structures and resolve the diversity of splicing variants. However, currently available *ab initio* aligners are vulnerable to spurious alignments due to random sequence matches and sample-reference genome discordance. As a consequence, a significant set of false positive exon junction predictions would be introduced, which will further confuse downstream analyses of splice variant discovery and abundance estimation.

In this chapter, we present a deep learning based splice junction sequence classifier, named DeepSplice [22], which employs convolutional neural networks to classify candidate splice junctions. We show (I) DeepSplice outperforms state-of-the-art methods for splice site classification when applied to the popular benchmark dataset HS3D, (II) DeepSplice shows high accuracy for splice junction classification with GENCODE annotation, and (III) the application of DeepSplice to classify putative splice junctions generated by Rail-RNA alignment of 21,504 human RNA-seq data significantly reduces 43 million candidates into around 3 million highly confident novel splice junctions.

2.1 Introduction

Technological improvements, reduced cost, and accessibility of RNA sequencing technologies have provided unprecedented visibility of the transcriptome through the deep sequencing of all mRNA transcripts present in a sample. Through analyses of mRNA-seq

data, researchers now believe that 92–94% of mammalian protein-coding genes undergo alternative splicing, with roughly 86% of these containing a minor transcript isoform frequency of at least 15% in certain cell types, developmental time points, physiological states, or other conditions [23]. This is an 87-89% increase from forty years ago when alternative exon structures from a single gene locus were first introduced and it was believed that only around 5% of genes in higher eukaryotes undergo alternative splicing [24].

The approach to defining exon junctions from RNA-seq data utilizes the subset of reads that have a gapped alignment to the reference genome. These reads can be aligned to two or more exons, indicating that there exist junctions joining adjacent exons. Whereas some mapping strategies [25-28] require pre-defined structural annotation of exon coordinates, more recently developed algorithms [29-33] can conduct *ab initio* alignment, which means that they do not rely on the existence of predetermined gene structure annotation and can potentially identify novel splice junctions between exons by the evidence of spliced alignments.

The accurate prediction of exon junctions is essential for defining gene structures and mRNA transcript variants. Splicing must be absolutely precise because the deletion or addition of even a single nucleotide at the splice junction would throw the subsequent three-base codon translation of the RNA out of frame [34]. However, novel splice junctions predicted by read alignments are not totally reliable, since the possibility of randomly mapping a short read up to 150 bases to the large reference genome is high [35], especially when gapped alignments with short anchoring sequences are permitted. In a recent report by Nellore et al [36] that investigated splicing variation, 21,504 RNA-seq samples from

the Sequenced Read Archive (SRA) were aligned to the human hg19 reference genome with Rail-RNA [37], identifying 42 million putative splice junctions in total. This value is 125 times the number of total annotated splice junctions in humans, making it impossible to admit that all of them actually exist. False positive splice junctions may lead to false edges in splice graphs, significantly increasing the complexity of the graphical structures [38]. Consequentially, this will impact the accuracy of splice variant inference algorithms as they often start from splice graphs derived from RNA-seq alignment [39].

Conventional strategies designed to filter out false positive exon splice junctions depend primarily on two properties: (1) the number and the diversity of reads mapped to the given splice junction [35]; and/or (2) the number of independent samples in which the specific exon splice junction is identified [35, 40]. In general, higher read support and sample reoccurrence rate both enlarge the likelihood of being a true splice junction. These criteria have a positive correlation with the number of read alignments, which are dependent on the sampling depth of the particular sample. Exact thresholds are difficult to set due to varying sampling depth across samples. Additionally, due to both sequencing and alignment errors, a splice junction with both high read support and high sample reoccurrence may still be the result of systematic bias. In contrast, a splice junction that exists in a transcript with relatively low expression may still be functionally important [41]. Thus, further classification of putative splice junctions revealed by RNA-seq data is still necessary but remains a challenging issue.

Since the 1980s, a number of bioinformatic approaches have been developed for splice site prediction. Neural networks [42-44], support vector machines [45-47], hidden Markov model [48-50], deep Boltzmann machines [51] and discriminant analysis [52, 53]

have been applied to recognize splice sites in the reference genome of many given species. Neural networks, support vector machines and deep Boltzmann machines learn the complex features of neighborhoods surrounding the consensus dinucleotide AG/GT by a non-linear transformation. Hidden Markov models estimate position specific probabilities of splice sites by computing the likelihoods of candidate signal sequences. The discriminant analysis uses several statistical measures to evaluate the presence of specific nucleotides, recognizing splice sites without explicitly determining the probability distributions [50]. However, all these work treat donor and acceptor sites as independent events, failing to leverage the inherent relationships between the donor and acceptor during splicing.

In this work, we develop a deep neural network-based approach to the classification of potential splice junctions. Our method is applicable to both splice site prediction and splice junction classification. First, instead of treating donor or acceptor splice sites individually, our method models the donor and acceptor splice sites as a functional pair. Thus, it is capable of capturing the remote relationships between features in both donor and acceptor sites that determine the splicing. Additionally, flanking subsequences from both exonic and intronic sides of the donor and acceptor splice sites will be used for learning and prediction, making it possible to understand the contribution of both coding and non-coding genomic sequences to the splicing. Our approach does not rely on sequencing read support or frequency of occurrence derived from experimental RNA-seq data sets, thus can be applied as an independent evidence for splice junction validation. Our experiments demonstrate that DeepSplice outperforms other state-of-the-art approaches [50, 54-58] when tested against a benchmarking dataset, Homo Sapiens Splice Sites Database (HS3D),

using a variety of evaluation metrics. Trained on an older version of the GENCODE project gene annotation data [59], we show that our algorithm can predict the newly annotated splice junctions with high accuracy and performs better than splice site-based approach. The application of DeepSplice to further classify putative intropolis human splice junction data by Nellore et al [36] is able to eliminate around 83% unannotated splice junctions. We discover that the combinational information from the functional pairing of donor and acceptor sites facilitates the recognition of splice junctions and demonstrate from large amounts of sequencing data that non-coding genomic sequences contribute much more than coding sequences to the location of splice junctions [47, 60].

2.2 DeepSplice method

DeepSplice employs a convolutional neural network (CNN, or ConvNet) to understand sequence features that characterize real splice junctions [61]. The overall architecture of DeepSplice is shown in Figure 2.1. In the supervised training step, CNN learns features that help to differentiate actual splice junctions from fake ones. In the inference step, the trained model uses the genomic sequence of the candidate splice junction and predicts the probability of it being a real splice junction. Deep Taylor decomposition [62] of the CNN is used to explain to what extent each nucleotide in the candidate splice junction has contributed to the inference.

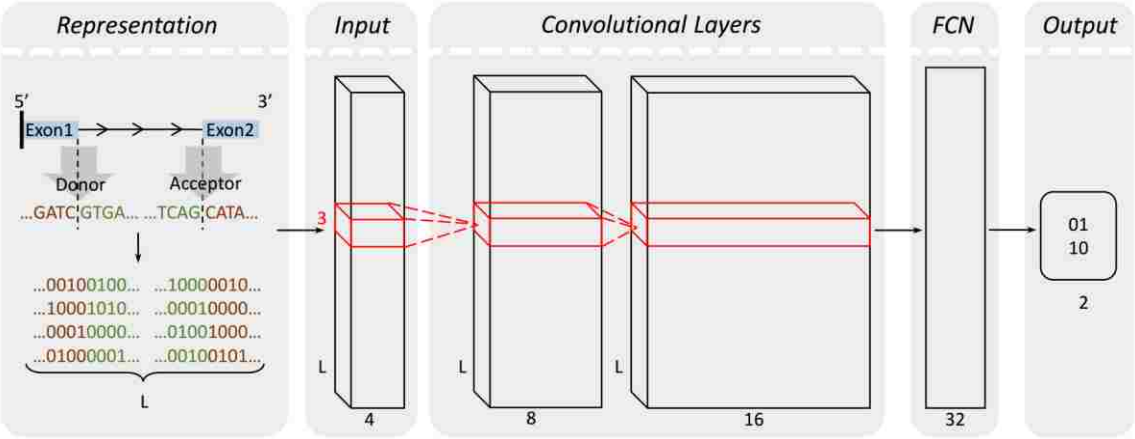

Figure 2.1 Visualization of splice junction sequence representation and deep convolutional neural network in DeepSplice. Each sequence is converted into a tensor through one-hot encoding in the pre-processing of the sequence representation. The tensor is fed as original input to the deep convolutional neural network, which contains one input layer, two convolutional layers, one fully connected layer (FCN) and one output layer. The convolutional neural network transforms the nucleotide signal in splice junction sequences to the final label of class.

2.2.1 Splice junction representation

A splice junction sequence is represented by four subsequences, the upstream exonic subsequence and downstream intronic subsequence at the donor site, and the upstream intronic subsequence and downstream exonic subsequence at the acceptor site, as shown in Figure 2.1. Each subsequence has the length of 30, which is believed to be optimal for splice site/junction prediction [41, 44, 48, 49, 63]. Nucleotides in each sequence are represented through one-hot encoding. In the proposed encoding system, the orthonormal sparse encoding is used for the four definite values (A, C, G and T) as it has been used widely in the numerical representations of biological sequences [64]. But for the ambiguous base N, instead of disregarding it or giving it the same importance as the definite values, the probability is used.

Each splice junction sequence is transformed into a 3-dimensional tensor. The first dimension 'height' is equal to one, and the second dimension 'width' indexes the sequence length, that is, the number of nucleotides in the sequence, and the third dimension 'channels' indexes the type of nucleotide. The tensors are fed as input to deep convolutional neural networks for downstream processing.

2.2.2 Deep convolutional neural network

DeepSplice contains a multi-layer feedforward neural network. We stack one input layer, two convolutional layers, one fully connected layer, and one output layer. The whole network architecture can be written as follows:

$$Label\ of\ class = f_{fcn}(f_{conv2}(f_{conv1}(Sequence\ nucleotide\ signal))).$$

In this way, the convolutional neural network transforms the nucleotide signal in splice junction sequences to the final label of class as shown in Figure 2.1.

In the first convolutional layer, the convolution will compute 8 features over the input tensor which represents splice junction sequence, which results in 8 feature maps of the input tensor. In order to reason the complex nonlinearity between inputs and outputs, we further stack the second convolutional layer computing 16 features over 8 feature maps from the first convolutional layer. In the convolutional layers, the filters have size 3x1. During the forward pass, we slide each filter along the splice junction sequence and compute dot products between the filter and the input tensor. As we slide the filter over the input splice junction sequence we will produce feature maps that give the responses of that filter at every spatial position. After two convolutional layers, the features are presented in 16 tensors. The output of the second convolutional layer is taken by a fully connected layer

with 32 feature maps for high-level reasoning. The fully connected layer is followed by the output layer indicating the final label of class. In the neural network, all parameters are learned during training to minimize a loss function which captures the difference between the true labels of class and predicted values.

Training the network follows the standard backpropagation and optimizes the loss function using Adam [65]. Advance deep learning techniques L2 regularization [66], dropout [67] and mini-batch gradient descent [68] are deployed to regularize the network to prevent over-fitting and to accelerate the training process.

In the reference step, testing splice junction sequences transformed by one-hot encoding are fed to the learned network for a binary classification, which outputs the predicted label of the class, true or false splice junction.

2.2.3 Deep Taylor decomposition of deep convolutional neural network

We propose to use deep Taylor decomposition [62] to explain the contribution of nucleotides in the splice junction sequence to the final decision function of the deep convolutional neural network, as shown in Figure 2.2. Taking image recognition task as an example, such decomposition results in a "heat map" that indicates what pixels of the image are important for a neural network classification. In our application, for testing splice junction sequence $\mathbf{S}$, we would like to associate to nucleotide n a contribution score $C_n(\mathbf{S})$ from which it is possible to judge which nucleotides are of importance to explain the predicted label of class from the deep convolutional neural network.

Deep Taylor decomposition operates by running a backward pass on the trained convolutional neural network using a predefined set of rules. Backpropagating from the

function output down to the input, it results in assigning a set of scores $\mathbf{C}(\mathbf{S}) = \{C_n(\mathbf{S})\}$ to the nucleotides in the input testing splice junction sequence **S** to quantify their contributions to the predicted label of class.

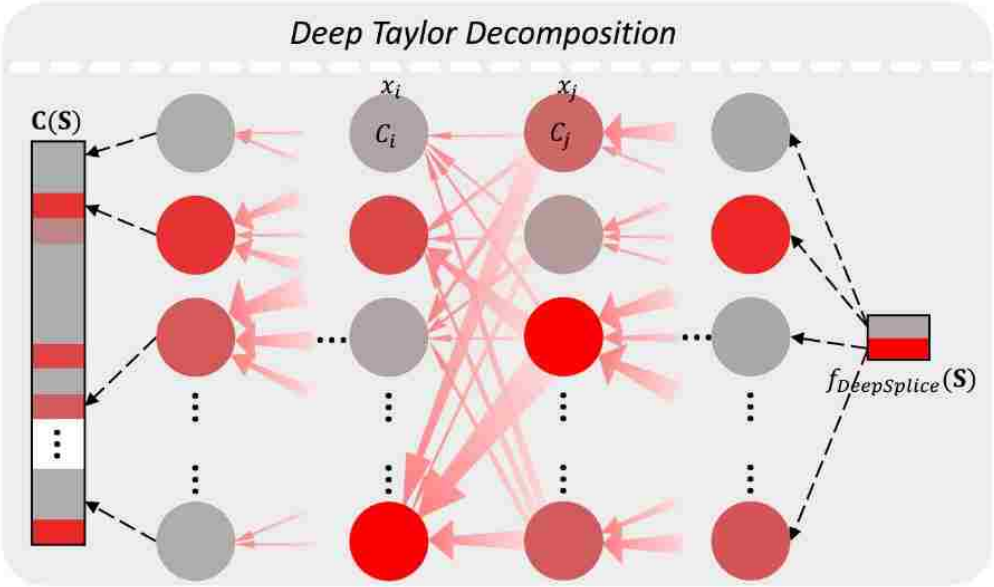

Figure 2.2 Visualization of deep Taylor decomposition in DeepSplice. Deep Taylor decomposition explains the contribution of each nucleotide in the splice junction sequence to the final decision function of the deep convolutional neural network. Deep Taylor decomposition operates by running a backward pass on the trained convolutional neural network using a predefined set of rules.

2.2.4 Other deep learning architectures

To decipher the abilities of different deep learning architectures in handling splice junction sequence data, we further build multilayer perceptron network (MLP) and long short-term memory network (LSTM) to compare with convolutional neural network. MLP is a feedforward artificial neural network with multiple hidden layers of units between input

and output layers. LSTM is a recurrent neural network architecture where connections between units form a directed cycle.

The multilayer perceptron network is composed of one input layer, four hidden layers and one output layer. Each layer is fully connected to next layer in the network. The number of neurons in each hidden layer is 64, 128, 128 and 256 respectively. In the long short-term memory network, we deploy one input layer, three hidden layers and one output layer. Each of the three hidden layers contains 16 LSTM cells. For both architectures, the inputs are splice junction sequences transformed by one-hot encoding, and the outputs are class labels. Advance deep learning techniques, dropout [67], regularization [66], mini-batch gradient descent [68] and Adam [65], are exploited in the supervised training steps in both networks.

2.2.5 Filtering of false splice junction as a result of repetitive sequences

One potential resource of false positive splice junction is the inability to align a sequence to the correct sites due to higher mismatches than the threshold set by aligners or small indels that cannot be detected by aligners. Before the classification of splice junctions, we first remove the splice junctions whose sequence at the acceptor (donor) site has high sequence similarity with the immediate flanking sequence next to the donor (acceptor) site or the sequence at any of its alternative acceptor (donor) sites, as shown in Figure 2.3. The edit distance between the alternative acceptor (donor) site sequences is computed using the Smith-Waterman algorithm [69]. This filtering strategy is independent of read coverage and enables the retention of correct splice junctions even with low read

coverage. The removal of these sequences is necessary as most of them are highly similar with one of the splice junctions remaining in the data set.

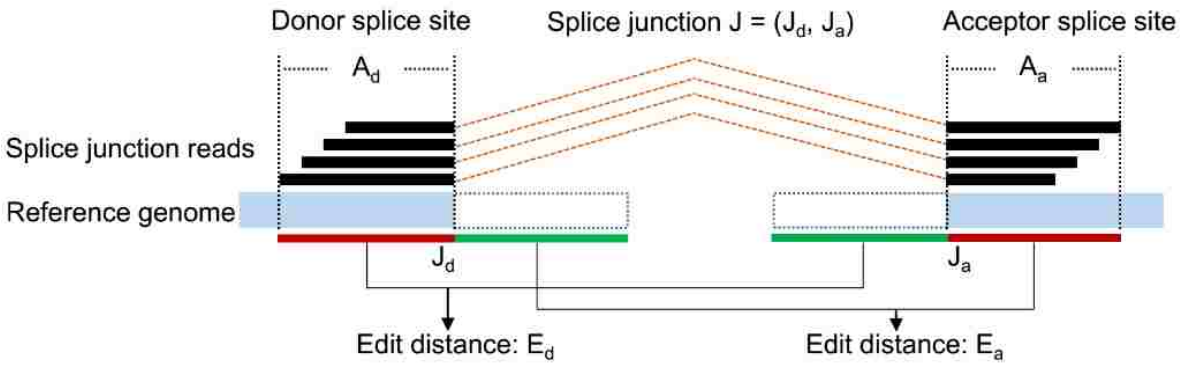

Figure 2.3 Illustration of splice junction filtering strategy. In this example, two edit distances are calculated. One (E_d) is between anchor sequence at donor site ($G[J_d\text{-}A_d\text{+}1\text{:}J_d]$) and intermediate flanking sequence next to acceptor site ($G[J_a\text{-}A_a\text{:}J_a\text{-}1]$). The other ($E_a$) is between anchor sequence at acceptor site ($G[J_a\text{:}J_a\text{+}A_a\text{-}1]$) and intermediate flanking sequence next to donor site ($G[J_d\text{+}1\text{:}J_d\text{+}A_d]$).

2.2.6 Implementation and performance measures

The deep learning architectures are implemented using TensorFlow [70]. Training and testing are deployed on Nvidia GeForce GTX 1080 graphics cards. DeepSplice is freely available for academic use and can be accessible at https://github.com/zhangyimc/DeepSplice.

We employ the following metrics: Area Under the ROC Curve (auROC), Area Under the Precision Recall Curve (auPRC), sensitivity, specificity, accuracy, F measure and Q^9. Q^9 is independent of the class distribution in the data set and is used to evaluate the classifier performance on splice site prediction [57].

2.3 Experimental results

We first applied our approach to a benchmark dataset HS3D [71] and compared the performance with other state-of-the-art approaches for donor and acceptor splice site classification. We then evaluated DeepSplice's performance by classifying annotated splice junctions from GENCODE gene annotation data [59]. Deep Taylor decomposition [62] was then applied for further interpretation of base level contribution of flanking splice sequence. Finally, we applied DeepSplice to intropolis [36], a newly published splice junction database with 42,882,032 splice junctions derived from 21,504 samples. The detailed results are described below.

2.3.1 DeepSplice outperforms state-of-the-art splice site prediction method

We utilized HS3D [71] (Homo Sapiens Splice Sites Data set, http://www.sci.unisannio.it/docenti/rampone/), a popular benchmark for measuring the quality of splice site classification methods. HS3D includes introns, exons and splice site sequences extracted from GeneBank Rel. 123. The splice site sequences in HS3D are with the length of 140 nucleotides. There are 2796 (2880) true donor (acceptor) splice sites and 271,937 (329,374) false donor (acceptor) splice sites which all contain conserved GT (AG) dinucleotides. We constructed the 1:10 data set, which contains all the true splice sites and 27,960 (28,800) randomly selected false donor (acceptor) splice sites. Binary classifications were conducted to identify the actual splice sites on donor and acceptor splice site data separately.

DeepSplice was trained on donor and acceptor splice site sequences separately in order to compare with state-of-the-art approaches of splice site classification. The exact

same number of training and testing splice site sequences from HS3D were used for all approaches. Table 2.1 summarizes the classification accuracies on the 1:10 data set by 10-fold cross-validation. To measure the quality of the classification results, we employed sensitivity, specificity, and Q^9 which is the global accuracy measure calculated from both sensitivity and specificity scores. Since the published splice site classification methods do not provide public tools for training and testing, the results of SVM+B [54], MM1-SVM [50], DM-SVM [55], MEM [56] and LVMM2 [57] were obtained from [55, 57]. As shown in Table 2.1, DeepSplice outperforms other methods in both sensitivity and specificity for both donor and acceptor splice site classification. For donor splice sites, there is a 95% likelihood that the confidence interval [0.0581, 0.0633] covers the true classification error of DeepSplice on the testing data. For acceptor splice sites, there is a 95% likelihood that the confidence interval [0.0814, 0.0872] covers the true classification error of DeepSplice on the testing data.

Table 2.1 Evaluation of DeepSplice and state-of-the-art approaches for donor (acceptor) site classification on HS3D data set

	Donor			*Acceptor*		
	Sensitivity	*Specificity*	Q^9	*Sensitivity*	*Specificity*	Q^9
LS-GKM	0.8679	0.8516	0.8595	0.8403	0.8319	0.8361
SVM+B	0.9406	0.9067	0.9212	0.9066	0.8797	0.8920
MM1-SVM	0.9256	0.9244	0.9247	0.8993	0.8869	0.8926
DM-SVM	0.9469	0.9339	0.9399	0.9215	0.9073	0.9136
MEM	0.9324	0.9275	0.9295	0.9153	0.8843	0.8978
LVMM2	0.9424	0.9242	0.9323	0.9122	0.8970	0.9039
DeepSplice	**0.9571**	**0.9376**	**0.9465**	**0.9337**	**0.9139**	**0.9232**

To deduce the most suitable architecture for learning the patterns in splice site/junction sequences, we then compared DeepSplice against two other prominent types of neural networks, multilayer perceptron network and long short-term memory network, in terms of classifying HS3D data set by 10-fold cross-validation. As shown in Figure 2.4, DeepSplice with convolutional neural network exceeds the other architectures, achieving an auROC score of 0.983 (0.974) on donor (acceptor) splice site classification and an auPRC score of 0.863 (0.800) on donor (acceptor) splice site classification. LSTM achieved an auROC score of 0.960 (0.942) on donor (acceptor) splice site classification and an auPRC score of 0.803 (0.721) on donor (acceptor) splice site classification. MLP achieved an auROC score of 0.931 (0.914) on donor (acceptor) splice site classification and an auPRC score of 0.650 (0.559) on donor (acceptor) splice site classification. In general, convolutional neural network is a well-studied architecture, which outperforms other deep learning architectures in almost all kinds of applications currently [72]. Even for speech recognition, convolutional neural networks recently beat recurrent neural networks. In our application, convolutional layers efficiently learned the complex information of nucleotide neighborhoods.

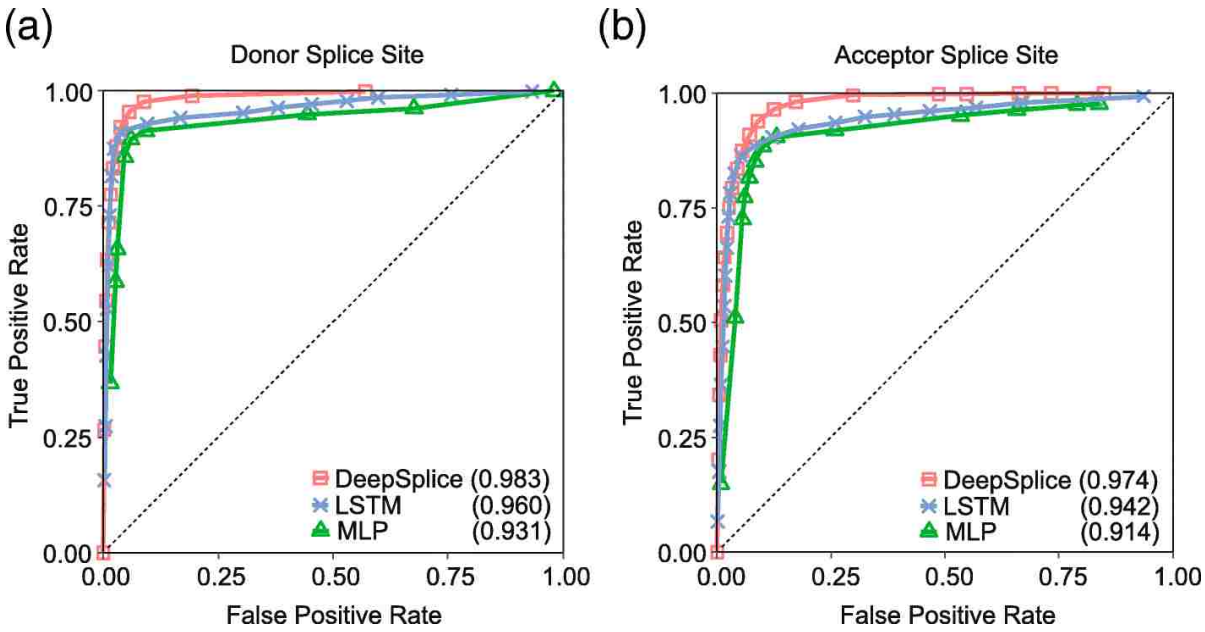

Figure 2.4 The ROC curves of DeepSplice, multilayer perceptron network (MLP) and long short-term memory network (LSTM) for donor (acceptor) splice site classification on the HS3D data set by 10-fold cross-validation. DeepSplice with convolutional neural network exceeds the other deep learning architectures, achieving an auROC score of 0.983 (0.974) on donor (acceptor) splice site classification.

2.3.2 DeepSplice predicts newly annotated splice junctions with high accuracy

Next, we evaluated the accuracy of DeepSplice in terms of splice junction classification. To achieve this, we trained DeepSplice using splice junctions extracted from the GENCODE annotation version 3c, and then tested the model on newly annotated splice junctions in the GENCODE annotation version 19. All GENCODE splice junctions used for training and testing are experimental validated by RT-PCR amplification. The training set contains 521,512 splice junctions, and the testing set contains 106,786 splice junctions. In both training and testing sets, half of the splice junctions are annotated, and the rest are false splice junctions randomly sampled from human reference genome (GRCh37/hg19).

We trained the first model by feeding the 521,512 training splice junction sequences to DeepSplice for a binary classification, splice junctions or not. In the meantime, we trained two other models separately by feeding the donor (acceptor) splice site sequences extracted from the 521,512 training splice junction sequences to DeepSplice for a binary classification, donor (acceptor) splice sites or not. This experiment was designed to determine whether making use of paired combinational information of donor and acceptor splice sites from a splice junction, instead of classifying donor or acceptor splice site individually, would ameliorate the quality of splice junction classification. In the first mode (Splice Junction Mode), the input splice junction sequences were with the length of 120 nucleotides, reflecting 30 nucleotides of upstream and downstream nucleotides for both donor and acceptor splice site. In the second mode (Donor+Acceptor Site Mode), the input splice junction sequences were split into two substrings with the length of 60 nucleotides and then fed to donor (acceptor) splice site classification model separately. For the second mode, we defined that the probability of a splice junction being classified as positive is the product of the probability of its donor splice site being classified as positive and the probability of its acceptor splice site being classified as positive, considering the two splice site classification events are statistically independent [73]. Figure 2.5 shows the ROC curves of the two modes. Splice Junction Mode achieved an auPRC score of 0.990, 0.987 for Donor+Acceptor Site Mode. Table 2.2 summarizes sensitivity, specificity, accuracy, and F1 score on the 106,786 testing splice junction sequences. Donor+Acceptor Site Mode acquires a higher specificity; however, Splice Junction Mode significantly outperforms Donor+Acceptor Site Mode in terms of sensitivity, accuracy, F1 score, auROC score, and auPRC score with substantially higher scores. In total, Splice Junction Mode predicted

50,340 out of 53,393 newly annotated splice junctions, which covered 9,806 genes, 98.01% of all newly annotated genes. Donor+Acceptor Site Mode detected 39,067 splice junctions from 9,185 genes. There is a 95% likelihood that the confidence interval [0.0432, 0.0456] covers the true classification error of DeepSplice on the testing splice junctions. These results indicate that the proposal splice junction classification in DeepSplice achieves high accuracy in identifying novel splice junctions in large data sets than conventional splice site classification.

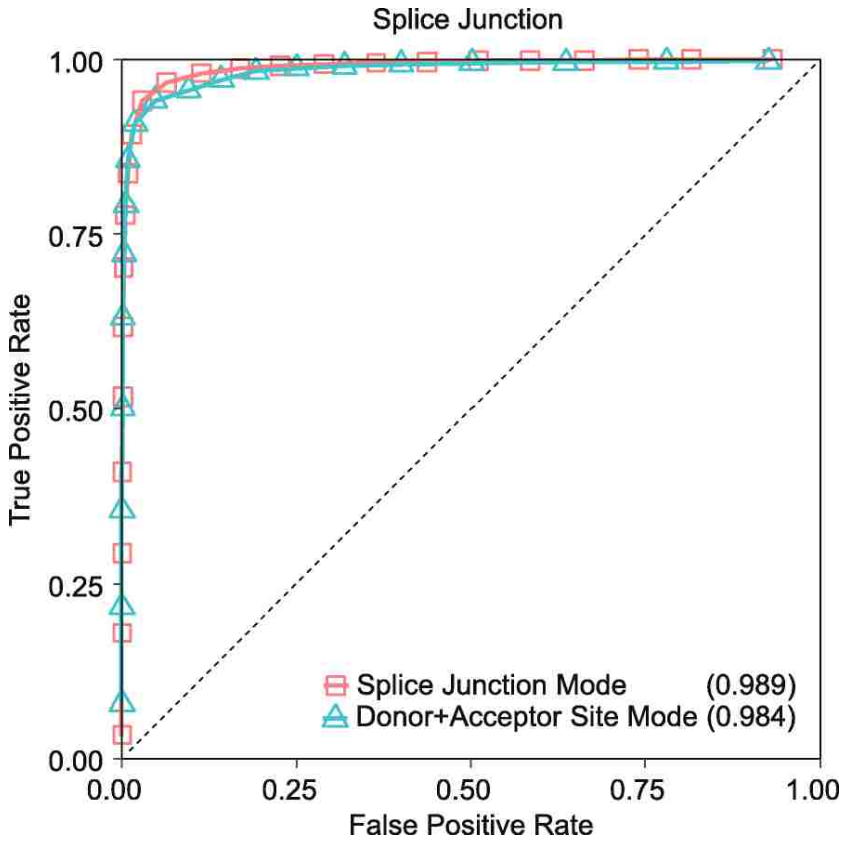

Figure 2.5 The ROC curves of DeepSplice Splice Junction Mode and Donor+Acceptor Site Mode for splice junction classification on the GENCODE data set. DeepSplice Splice Junction Mode achieves a higher auROC score of 0.989.

Table 2.2 Classification performance evaluation of different DeepSplice modes on GENCODE data set

		Sensitivity	*Specificity*	*Accuracy*	*F1 score*
Splice site classification	*Donor*	0.917	0.897	0.907	0.908
	Acceptor	0.873	0.913	0.893	0.891
Splice junction classification	*Splice Junction Mode*	0.943	0.968	0.956	0.955
	Donor+Acceptor Site Mode	0.732	0.997	0.864	0.844

2.3.3 Interpretation of sequence features captured by DeepSplice

There are highly conserved segments on splice junctions between exons and introns which help in the prediction of splice junctions by computational methods and decipher biological signals of splice junctions. We next further interpret which nucleotides contribute to the splicing process. This is achieved by the quantification of the contribution of nucleotides in splice junction sequences to the classification process using deep Taylor decomposition [62].

DeepSplice employs convolutional neural network with two convolutional layers. In the convolutional layer, we defined filters with a shape of 3x1, which means filters scan the input sequence with a window size of 3 to learn the information of nucleotide neighborhoods. DeepSplice fundamentally is not using a single base but rather 3-mers or subsequences of length 3 as its features. Then deep Taylor decomposition runs a backward pass on the convolutional neural network to sign contributions. The contribution score of each single base in DeepSplice reflects the aggregated importance of the three 3-mers it belongs to. We first used deep Taylor decomposition to decompose cross-validation results of the HS3D dataset in terms of input splice site sequences. For nucleotides in the testing splice site sequences, scores were assigned to present their contribution. We obtained a graphical representation from which it is possible to judge which region in the splice site

sequences is of importance. Figure 2.6 shows the contribution of nucleotides to the final decision function of DeepSplice. In general, intron sequences carry more discriminative information than exon sequences in this analysis. We then applied deep Taylor decomposition to the results of splice junction classification with the GENCODE data set. Figure 2.7 shows the contribution distribution of nucleotides in the testing splice junction sequences. Regions of increased importance in splice junction classification are consistent with the result from splice site classification.

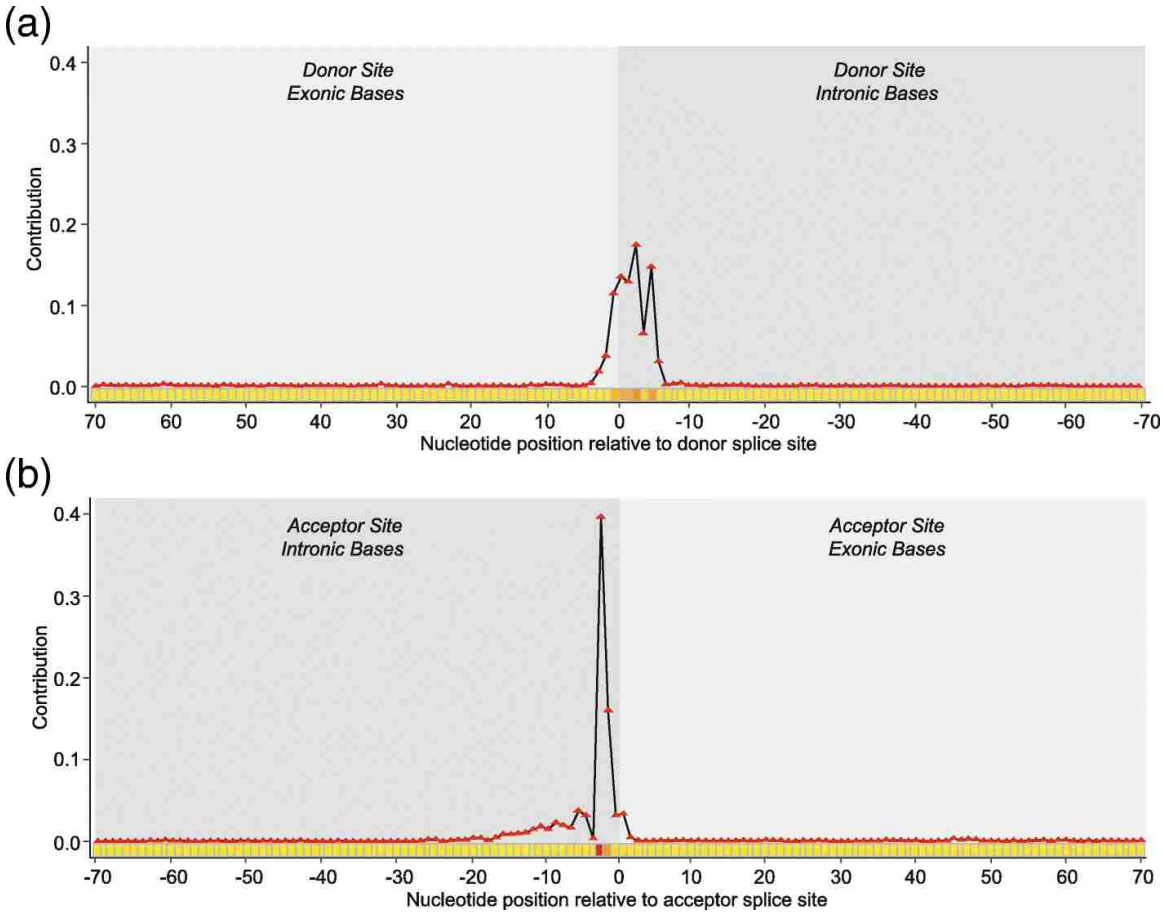

Figure 2.6 Visualization of the contribution of nucleotides in the flanking splice sequences to the final decision function of DeepSplice on the HS3D dataset for donor (acceptor) site classification. For both donor and acceptor site classifiers, intronic bases close to GT-AG di-nucleotides achieve the most importance in the classifiers. In general, intron sequences carry more discriminative information than exon sequences.

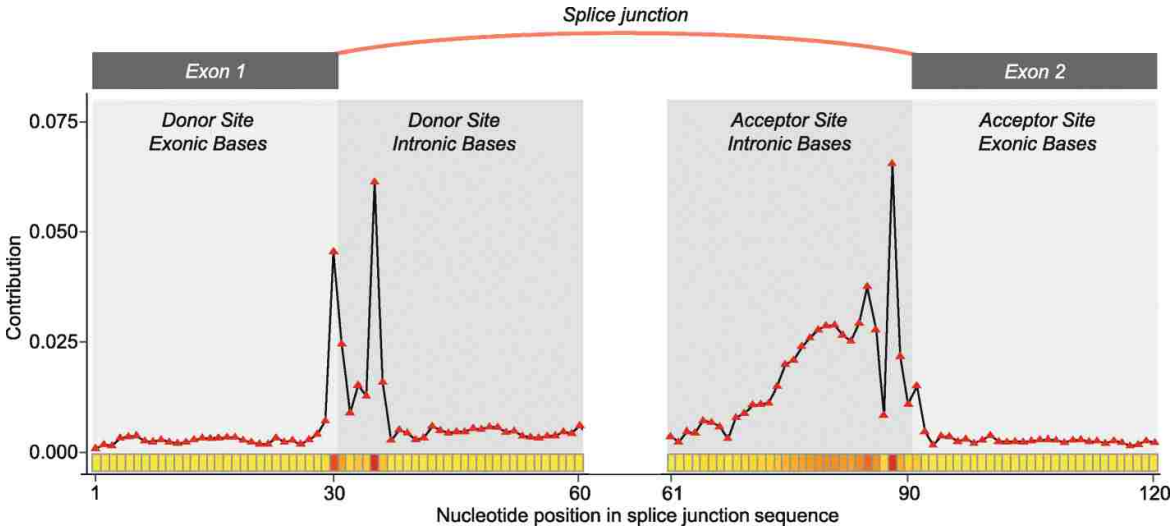

Figure 2.7 Visualization of the contribution of nucleotides in the flanking splice sequences to the final decision function of DeepSplice on the GENCODE dataset for splice junction classification. The nucleotides in the proximity of a splice junction have the highest impact on the classification outcome. As observed in the splice site classifiers, the contribution distribution of nucleotides in the flanking splice sequences indicates that intron nucleotides carry more discriminative information than exon nucleotides.

2.3.4 DeepSplice classification of intropolis

The intropolis v1 database [36] contains a large number of putative junctions found across 21,504 human RNA-seq samples in the Sequence Read Archive (SRA) from spliced read alignments to hg19 with Rail-RNA [37]. There are 42,882,032 putative splice junctions in total, including 18,856,578 canonical splice junctions containing flanking string GT-AG, 24,025,454 semi-canonical splice junctions containing flanking string AT-AC or GC-AG [74], and no non-canonical splice junctions which are not allowed by Rail-RNA. Table 2.3 lists the number of splice junctions in each category separated by the number of reoccurrence in samples and total read support across all samples in four scales: (a) equal to 1 {1}, (b) more than 1 and no greater than 10 (1, 10], (c) more than 10 and no greater than 1000 (10, 1000] and (d) more than 1000 (1000, $+\infty$). As listed in Table 2.3, for

our analysis, we only retain splice junctions in intropolis that are supported by more than one sample, followed by the filtering of false splice junction sequences due to repetitive sequences. After this pre-processing, 5,277,046 splice junctions were left for further classification.

Table 2.3 Distribution of splice junctions from intropolis given the reoccurrence in samples and total read support

Splice junction number		*Reoccurrence in samples*			
		{1}	*(1, 10]*	*(10, 1000]*	*(1000, +∞)*
Total reads	*{1}*	23M	-	-	-
	(1, 10]	3,331K	11M	-	-
	(10, 1000]	91K	936K	3,301K	-
	(1000, +∞)	38	187	124K	305K

"M" stands for "million".

"K" stands for "thousand".

The DeepSplice model was trained on 812,967 splice junctions including (1) 291,030 annotated splice junctions from GENCODE annotation version 19, (2) 271,937 false splice junctions generated from the HS3D data set, and (3) 250,000 randomly selected semi-canonical splice junctions with only one read support from intropolis. Overall, DeepSplice classified 3,063,698 splice junctions as positive. Figure 2.8 (a) lists the proportions of positive canonical splice junctions, positive semi-canonical splice junctions and negatives from the classification results at different levels of average read support per sample. Splice junctions with average read support per sample more than 15 achieve a positive rate around 88%. In contrast, for splice junctions with average read support per sample no more than 1, only 36% are identified as positives. There is a significant rise in the probability to obtain a positive splice junction with the increase of the average read

support per sample. Around 99% positive splice junctions contain the canonical flanking string. Figure 2.8 (b) illustrates the proportions of positive semi-canonical and canonical splice junctions cumulatively with the increase of the average read support per sample.

To further clarify characteristics of the positives, we categorized splice junctions in intropolis based on annotated splice sites in GENCODE annotation: (1) splice junctions with both splice sites annotated, (2) splice junctions with the donor splice site annotated, (3) splice junctions with the acceptor splice site annotated, and (4) splice junctions with neither the donor nor acceptor splice sites annotated. Figure 2.9 (a) shows the discrete proportions of negatives and positive splice junctions in each category above, given the average read support per sample. Results indicate that 97% of splice junctions with both sites annotated are classified as positives, while only 39% with both sites being novel are positive. Splice junctions connecting annotated splice sites also tend to be associated with higher read coverage. Figure 2.9 (b) illustrates the proportions of positive splice junctions in each category cumulatively with the increase of the average read support per sample. Figure 2.10 shows positive splice junctions in intropolis near known protein-coding junctions show a periodic pattern, such that splice sites which maintain the coding frame of the exon are observed more often than those which disrupt frame. This observation recapitulates patterns seen in studies of noisy splicing [41].

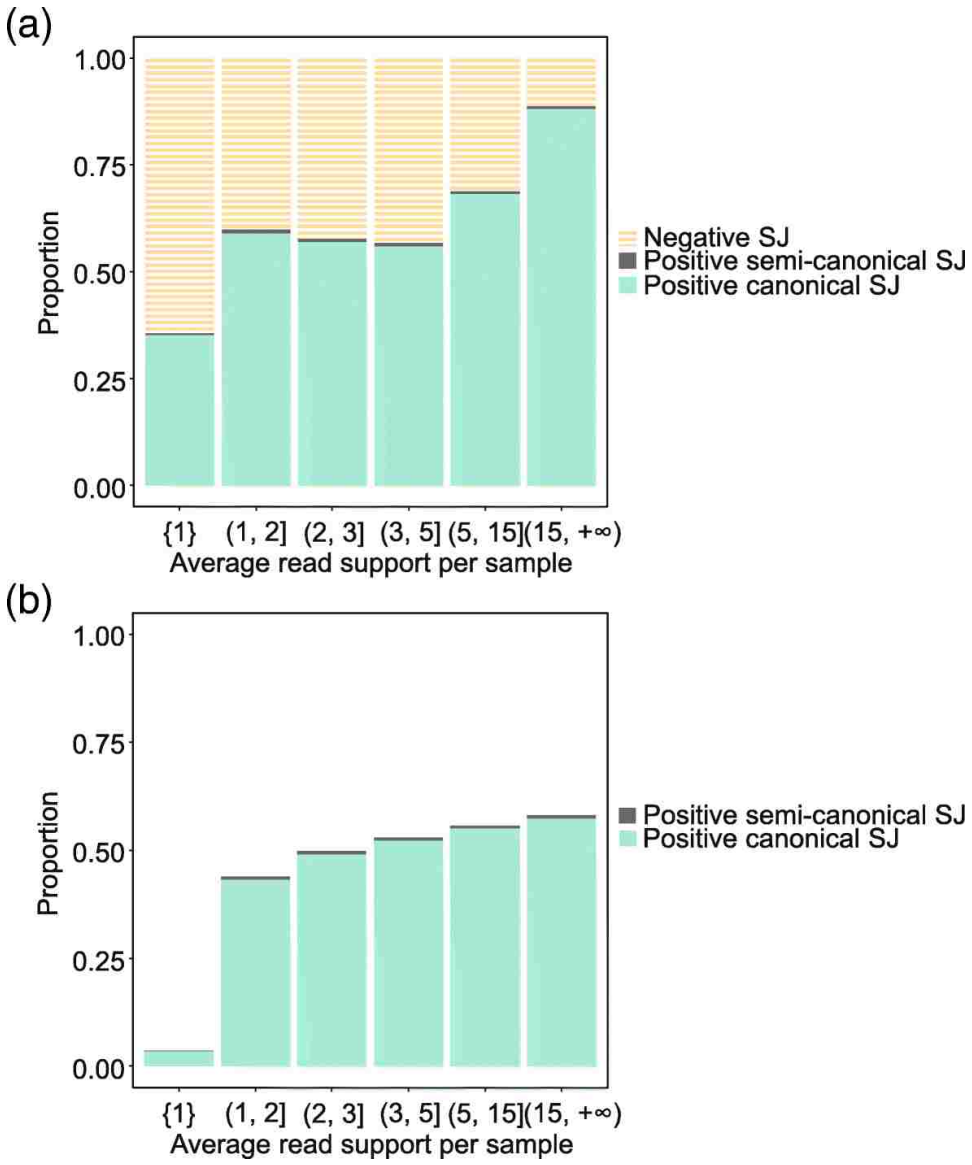

Figure 2.8 Positive splice junctions tend to have high read support and contain the canonical flanking string. (a) Discrete proportions of negatives, positive semi-canonical splice junctions and positive canonical splice junctions from the classification results, given the average read support per sample. Splice junctions with average read support per sample more than 15 achieve a positive rate of around 88%. In contrast, for splice junctions with average read support per sample no more than 1, only 36% are identified as positive. There is a significant rise in the probability to obtain a positive splice junction with the increase of the average read support per sample. Around 99% positive splice junctions contain the canonical flanking string. (b) Cumulative proportions of positive semi-canonical and canonical splice junctions with the increase of the average read support per sample.

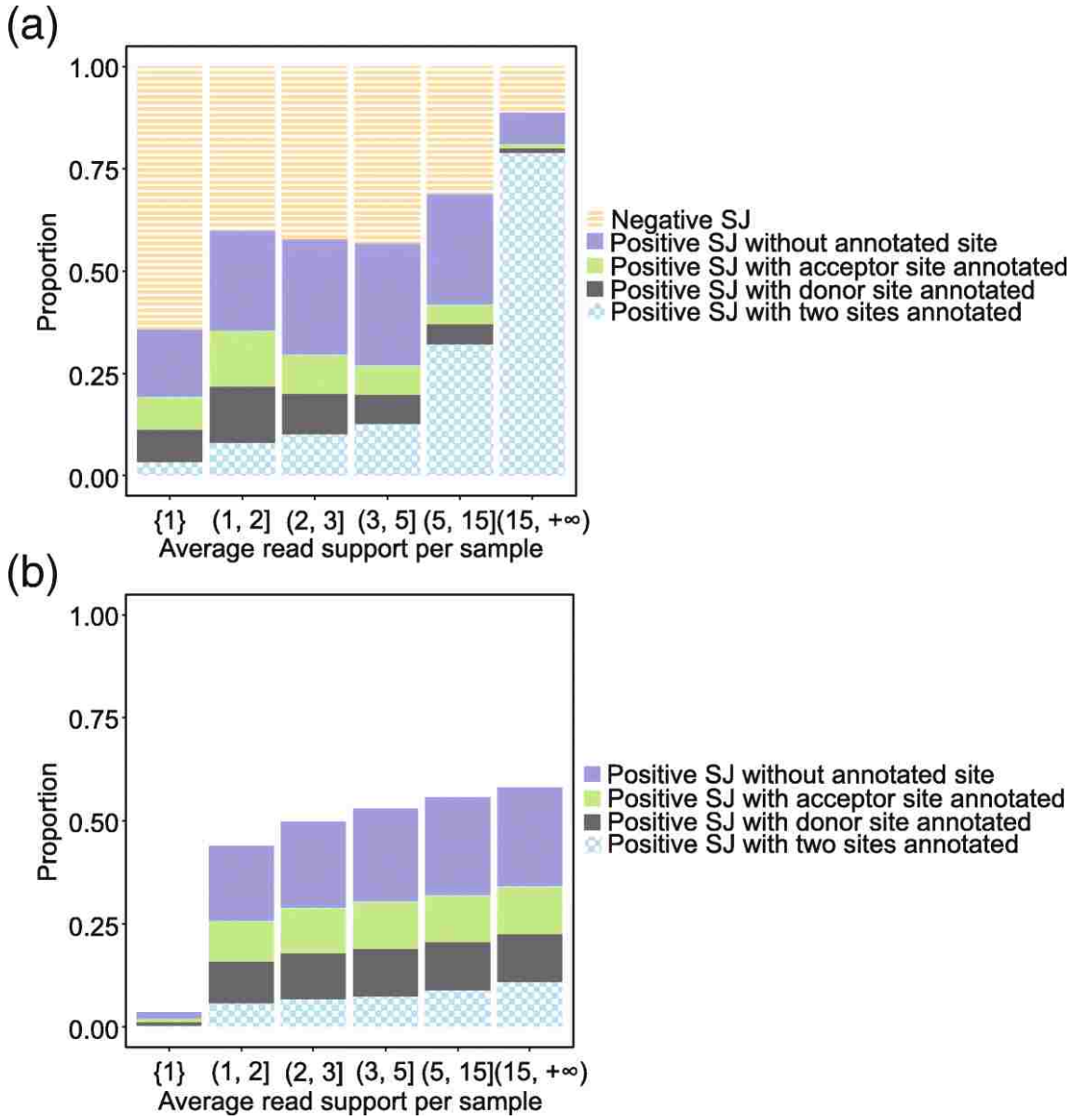

Figure 2.9 Positive splice junctions tend to have both donor and acceptor sites annotated. (a) Discrete proportions of negatives, positive splice junctions without annotated site, positive splice junctions with acceptor site annotated, positive splice junctions with donor site annotated and positive splice junctions with two sides annotated, given the average read support per sample. 97% of splice junctions with both sites annotated are classified as positives, while only 39% with both sites being novel are positive. Splice junctions connecting annotated splice sites also tend to be associated with higher read coverage. (b) Cumulative proportions of positive splice junctions in each category with the increase of the average read support per sample.

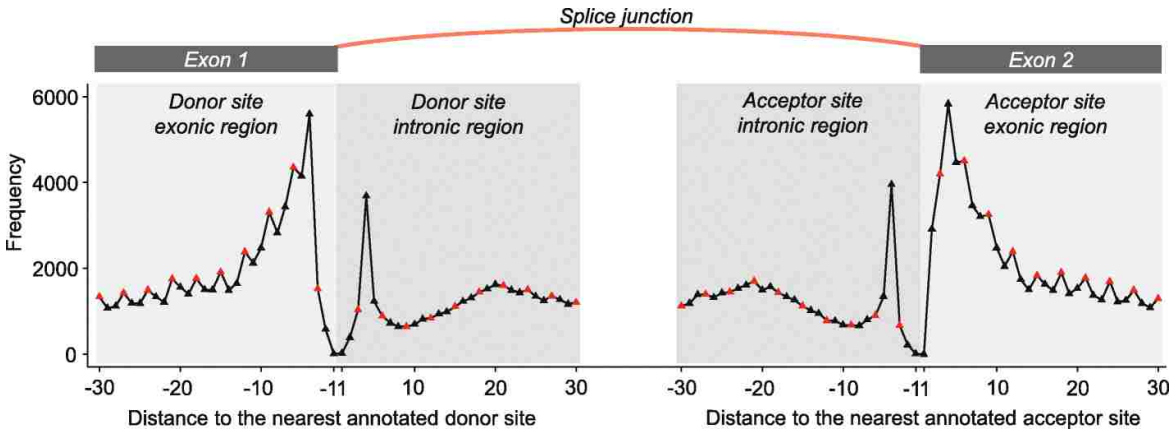

Figure 2.10 Splice sites which maintain the coding frame of the exon are observed more often than those which disrupt frame. Positive splice junctions in intropolis near known protein-coding junctions show a periodic pattern. For each donor (acceptor) site in the positive splice junctions, we calculated its distance to the nearest annotated donor (acceptor) site, and then counted the frequency for each position. The red points denote positions that are a multiple of three base pairs from the major splice form, and the black points those that are not.

2.4 Summary

Even though splice junctions with high read support and/or high reoccurrence are more likely to be classified as real, a significant portion of relatively low-expressed splice junctions also carry true splicing signals. DeepSplice does not rely on sequencing read support, frequency of occurrence, or sequencing read length derived from experimental RNA-seq data sets, thus can be applied as an independent evidence for splice junction validation. The accumulation of RNA-seq data especially in different cell types, tissues and disease conditions will further consolidate the cell type-specificity and tissue-specificity of some of these junctions and their corresponding isoforms. DeepSplice may provide the first round of filtering of RNA-seq derived splice junctions for further structural validation, and studies that assess functional annotation of these splice junctions are warranted. DeepSplice could also extend its functionality to discriminate splice junctions

that are highly or lowly supported by gene expression evidence and try to figure out what sequence patterns associate to this difference in future. For each input candidate splice junction, DeepSplice outputs a probability of being true, and the probability can be used as an input feature to the studies for learning the tissue-regulated splicing code [75] and the splicing in human tissues with a wide range of known diseases [76].

It is also well known that splicing can be changed due to mutations around the splice sites. Future studies that use subject-specific genomic sequences instead of reference genome sequences may further improve the accuracy of the DeepSplice model and classification performance. Additionally, DeepSplice can be further extended to the prediction of non-canonical splicing [77] that existing annotation has not captured, including not only exonic but also splicing involving Alu elements, small exons, and recursive splicing. Besides the classification of linear junctions, the identification of non-linear splice junctions, such as circRNA junctions will also expand the functionality of DeepSplice.

Employing deep convolutional neural network, we develop DeepSplice, a model inferred from the sequences of annotated exon junctions that can then classify splice junctions derived from primary RNA-seq data, which can be applied to all species with sufficient transcript annotation to use as training data. Results demonstrate that DeepSplice outperforms the state-of-the-art splice site classification tools in terms of both classification accuracy and computational efficiency. Our findings further indicate that valuable information is present in the nucleotide sequence local to the splice junction, data that conventional splice site prediction techniques discard. Nucleotide representations learned from the input sequences are meaningful and improve accuracy. The major application of

DeepSplice is the classification of splice junctions rather than individual donor or acceptor sites. For learning on large datasets of putative splice junctions, DeepSplice is orders of magnitude faster than the best performing existing alternatives, which becomes increasingly common considering the tremendous amount of new RNA-seq data being generated.

CHAPTER 3. INFERRING TRANSCRIPTION FACTORS GOVERNING METABOLIC REPROGRAMMING WITH TFMETA

Metabolic reprogramming is a hallmark of cancer. In cancer cells, transcription factors (TFs) govern metabolic reprogramming through abnormally increasing or decreasing the transcription rate of metabolic enzymes, which provides cancer cells growth advantages and concurrently leads to the altered metabolic phenotypes observed in many cancers. Consequently, targeting TFs that govern metabolic reprogramming can be highly effective for novel cancer therapeutics. In this chapter, we present TFmeta, a machine learning approach to uncover TFs that govern reprogramming of cancer metabolism. Our approach achieves state-of-the-art performance in reconstructing interactions between TFs and their target genes on public benchmark data sets. Leveraging TF binding profiles inferred from genome-wide ChIP-seq experiments and 150 RNA-seq samples from 75 paired cancerous (CA) and non-cancerous (NC) human lung tissues, our approach predicted 19 key TFs that may be the major regulators of the gene expression changes of metabolic enzymes of the central metabolic pathway glycolysis, which may underlie the dysregulation of glycolysis in non-small-cell lung cancer patients.

3.1 Introduction

Metabolism is collection of predominantly enzyme-catalyzed biochemical transformations that are needed for maintenance, growth and survival of an organism. For nearly a century, scientists have documented profound metabolic changes that occur in tumors [78]. Oncogenes and tumor suppressors are well-established regulators of metabolism, and dysregulated expression as well as mutations can lead to the altered metabolic phenotypes observed in many cancers [79, 80]. A high proportion of oncogenes

and tumor suppressor genes encode transcription factors (TFs) [81]. Most oncogenic pathways converge on sets of TFs that ultimately control gene expression patterns resulting in tumor formation and progression as well as metastasis [82]. Deregulated expression, activation or inactivation of TFs play critical roles in tumorigenesis. In cancer cells, TFs govern metabolic reprogramming by controlling the expression patterns of metabolic enzymes. For example, the transcription factor MYC is frequently overexpressed in human cancers and regulates the expression of many metabolic enzymes. In carcinomas, MYC drives increased Gln uptake and conversion to Glu by upregulating glutamine transporters and inducing the expression of metabolic enzyme GLS at the mRNA and protein level, leading to increased anaplerotic input via glutaminolysis into the Krebs cycle and increased Gln incorporation into lactate [79, 83, 84].

Comprehensive characterization of TF-metabolic enzyme interactions in cancer cells can help uncover potential TFs governing cancer metabolic reprogramming and prioritize targets for novel cancer therapeutics. Reconstructing interactions between TFs and their target genes from transcriptomic data is a long-standing and well-studied challenge in molecular and computational biology. Some interaction reconstruction methods [85-88] exploiting co-expression in gene expression patterns have successfully identified the interactions in the gene pairs whose expression vary sufficiently and correlate globally across a large set of samples. Other methods [89, 90] take advantage of differential gene expression to predict interactions between each TF and all the genes that are differentially expressed when the TF is deleted, overexpressed or perturbed. These methods, however, have at least two major drawbacks for reconstructing TF-target gene interactions. First, a fundamental assumption of current interaction reconstruction methods

using transcriptomic data is that mRNA levels of TFs and their target genes are strongly correlated; however, this assumption may not be true for all the data sets, especially for those containing complex TF-target gene interactions. The Dialogue on Reverse Engineering Assessment and Methods (DREAM) project performed an assessment of 35 TF-target gene interaction reconstruction methods on both synthetic and real transcriptomic data sets [85]. The competing methods achieved an average AUROC score of 0.69 on the synthetic data set, but 0.55 on the real data sets. The poor performance on the real data sets was due to the low correlation at the mRNA level in the data, which would suggest that reliable reconstruction of complex TF-target gene interactions requires additional inputs besides transcriptomic data, for example, TF binding profiles. Second, current interaction reconstruction methods disregard the valuable pairing information of the samples in transcriptomic data, treating each input gene expression profile independently in their inference models. For cancer patients' transcriptomic data, pairwise comparisons of gene expression profiles between matched cancerous (CA) and non-cancerous (NC) samples of the same patient should circumvent the interferences from genetic and physiological variations, eliminating the prediction of false TF-target gene interactions caused by the variations.

Here, we developed TFmeta [91], a machine learning method for the reverse engineering of TF-metabolic enzyme interactions that pinpoint TFs governing cancer metabolic reprogramming. TFmeta integrates transcriptomic data and TF binding profiles inferred from genome-wide ChIP-seq experiments, to learn non-linear interactions between TFs and their targets. Using a gold standard data set, namely DREAM5 network inference challenge [85] data, we demonstrate that TFmeta outperformed the winner of the challenge

in reconstructing TF-target gene interactions. Taking 150 RNA-seq samples from 75 paired CA and NC human lung tissues and TF binding profiles as input, TFmeta predicted a set of key TFs that may control the transcription rate of metabolic enzymes in the central metabolic pathway glycolysis, which may cause the observed metabolic reprogramming in glycolysis pathway in non-small-cell lung cancer patients [91].

3.2 TFmeta method

3.2.1 RNA-seq analysis

We sequenced 150 RNA-seq samples from 75 paired CA and NC human lung tissues under IRB approval from the University of Kentucky. All patient information was de-identified and adhered to HIPPA guidelines. 100 bp paired-end reads were generated by Illumina HiSeq 2000 sequencer. RNA-seq reads were mapped to the human reference genome *GRCh38*, and gene expression values (TPM, transcripts per million) were estimated using RSEM package [92]. Gene expression profiles generated from RSEM were normalized and comparable between samples. Pairwise gene expression comparisons of CA and NA samples from the same patient were conducted through measuring the log2 ratios of gene expression values between CA and matched NC samples. Based on the log2 ratios, we maintained a master table for showing the regulation status of each gene in each individual patient. The regulation status of each gene was represented by a categorical variable that can take on one of the three possible values: upregulated, downregulated, and no change. Genes with the |log2| ratio greater than 0.8 were categorized as upregulated (or downregulated) genes, and the rest were genes with no expression change. The size of the

master table was 19,814 (number of genes) by 75 (number of patients). Using the gene expression log2 ratio of paired CA and NC tissue samples from the same patient should reduce the effects of individuality and the impact of tissue-specific genes and consequently, increase the accuracy of predicting clinical outcomes [93].

We then collected the detailed information of the major metabolic pathways in human, including glycolysis, the Krebs cycle, purine metabolism, and others from KEGG (Kyoto Encyclopedia of Genes and Genomes) pathway database [94]. The regulation status of metabolic enzymes involved in each metabolic pathways was extracted from the master regulation status table. According to the one-tailed one-proportion z-test (with a hypothesized proportion of 0.6667), we considered metabolic enzymes with consistent expression change (upregulated or downregulated) among at least 57 patients out of the 75 patients as altered metabolic enzymes (p-value for 57 patients: $p=0.0433<0.05$).

3.2.2 Transcription factor binding profiling

We integrated TF binding profiles which were inferred from genome-wide ChIP-seq experiments in four public databases, including ChEA [95], ENCODE [96], JASPAR [97], TRANSFAC [98]. We eventually accumulated 2,286,192 TF DNA binding activities, involving 493 TFs and 23,644 target genes. The minimum, median and maximum number of TFs binding to a target gene is 1, 104 and 279, and the minimum, median and maximum number of target genes for one TF is 4, 1853 and 21545, respectively. The total number of metabolic enzymes involved in the major metabolic pathways is 366. For each altered metabolic enzyme, we curated a list of TFs which bind to the transcription start site of that enzyme according to the TF DNA binding activities.

3.2.3 TF-metabolic enzyme interaction inference

Problem Definition We approached the problem of uncovering TFs that govern cancer metabolic reprogramming by measuring the interactions between the altered metabolic enzymes and TFs binding to the transcription start sites of them. Through RNA-seq analysis, we identified M altered metabolic enzymes with consistent expression change between CA and matched NC samples. We divided the problem of inferring TF-metabolic enzyme interactions involving M enzymes into M sub-problems. Each of these sub-problems uncovered the TFs regulating one of the enzymes. We generated M sub-tables from the master regulation status table, each of which contained the regulation status of one enzyme and TFs which bind to the transcription start site of that enzyme according to the TF DNA binding activities. In the sub-table, for enzyme m with T_m TFs binding to its transcription start site, every patient's regulation status profile can be expressed as (x_n^m, y_n^m), where $n \in \{1, \dots, N\}$ is the index of each patient out of N patients, and x_n^m is a tensor of T_m TF regulation status, and y_n^m is the regulation status of enzyme m.

Interaction Inference as a Feature Selection Problem TFs and their target genes are known to interact in a dynamic and nonlinear manner [99]. We hypothesize that the regulation status of the enzyme m is a function f_m of the regulation status of the T_m TFs, and the function f_m only employs the regulation status of the TFs that are direct regulators of the enzyme m. Identifying those TFs whose regulation status is predictive of the regulation status of the enzyme m can be considered as a feature selection problem, which is to rank the input features in the function f_m based on their relevance for predicting the output in machine learning terminology. Considering a large amount of TFs as input

features relative to a small set of learning patient regulation status profiles and the nonlinear relationship between input TFs and the output enzyme, we proposed to use gradient boosted trees [100, 101] to find the function f_m and rank the input TFs by their relevance. Gradient tree boosting is a scalable and highly effective machine learning algorithm, which works well in reliably extracting relevant features and identifying non-linear feature interactions.

Gradient Boosted Tree-based Model For each sub-problem, we fitted a multi-class classification model (f_m) to predict the regulation status (upregulated, downregulated, no change) of the enzyme m based on the combined regulation status of the T_m TFs. Gradient boosted trees were employed to find the function f_m which minimizes the multi-class classification error rate which is calculated as the number of wrong predictions divided by the number of all predictions. To achieve this goal, classification and regression tree (also known as CART) recursively partitions the N patients into smaller disjoint sets based on the input regulation status of TFs, aiming at minimizing the number of wrong predictions of the output enzyme regulation status in the resulting subsets. Classification and regression tree uses the tree structure to represent the recursive partition, and each of the leaves in the tree represents a cell of partition. The basic idea of tree boosting is to build additive models through classification and regression trees. Let $b_{m,k}(\mathrm{x}_n^m)$ be a classification and regression tree in m$^{\text{th}}$ sub-problem, which works as the base learner. In tree boosting, we built a model that is the sum of base learners as:

$$f_m(\mathrm{x}_n^m) = \sum_{k=1}^{K} b_{m,k}(\mathrm{x}_n^m),$$

where $k \in \{1, \dots, K\}$ is the index of each base learner out of K base learners. The target

additive model was built in a forward stagewise fashion. Namely, it started with the simple function $f_{m,0}(\mathrm{x}_n^m) = 0$, then iteratively adds base learners to minimize the multi-class classification error rate of $f_{m,k-1}(\mathrm{x}_n^m) + b_{m,k}(\mathrm{x}_n^m)$. Gradient Boosting attempts to solve this minimization problem numerically via steepest descent. By iteratively shifting the focus towards problematic observations that were difficult to predict, the performance of the classification and regression tree is very much boosted.

Feature Importance Measure: TF Ranking A benefit of using CART-based methods is that after the trees are constructed, it is relatively straightforward to retrieve estimates of feature importance that allow ranking the input features according to their relevance for predicting the output. The importance is calculated for a single classification and regression tree by the amount that each attribute split point reduces the Gini impurity, weighted by the number of observations the node is responsible for. The feature importance scores are then averaged across all the classification and regression trees within the model. In this application, every CART-based sub-model solving one sub-problem yields a separate ranking of TFs as potential regulators of a target enzyme m along with importance scores I_{m,t_m} for $t_m \in \{1, \dots, T_m\}$.

TF-metabolic enzyme Map Our primary goal is ultimately using this approach to find a relatively small number of robust target TFs based on multiple lines of evidence. We considered a variety of strategies to select an appropriate threshold on the TF ranking in each sub-model. For instance, we could apply an independent threshold for each sub-model, or we could use a uniform threshold across all sub-models. We found that optimal performance was obtained when we applied an overall threshold on the combined TF ranking. To combine the separate rankings of TFs in sub-models, we performed the

Wilcoxon signed-rank test on every pair of TFs to compare their ranks, which tested whether the ranks of one TF from all sub-models were significantly higher (or lower) than those of the other TF. Based on the test decisions of comparing all pairs of TFs, the orders of TFs were eventually determined to generate the combined ranking. Through evaluating the number of output TFs and their biological significance, we considered TFs in the top 5% of the combined ranking as robust targets. The interactions between the predicted TFs and their target enzymes were then displayed in a TF-metabolic enzyme map. The overall workflow of TF-metabolic enzyme interaction inference is shown in Figure 3.1.

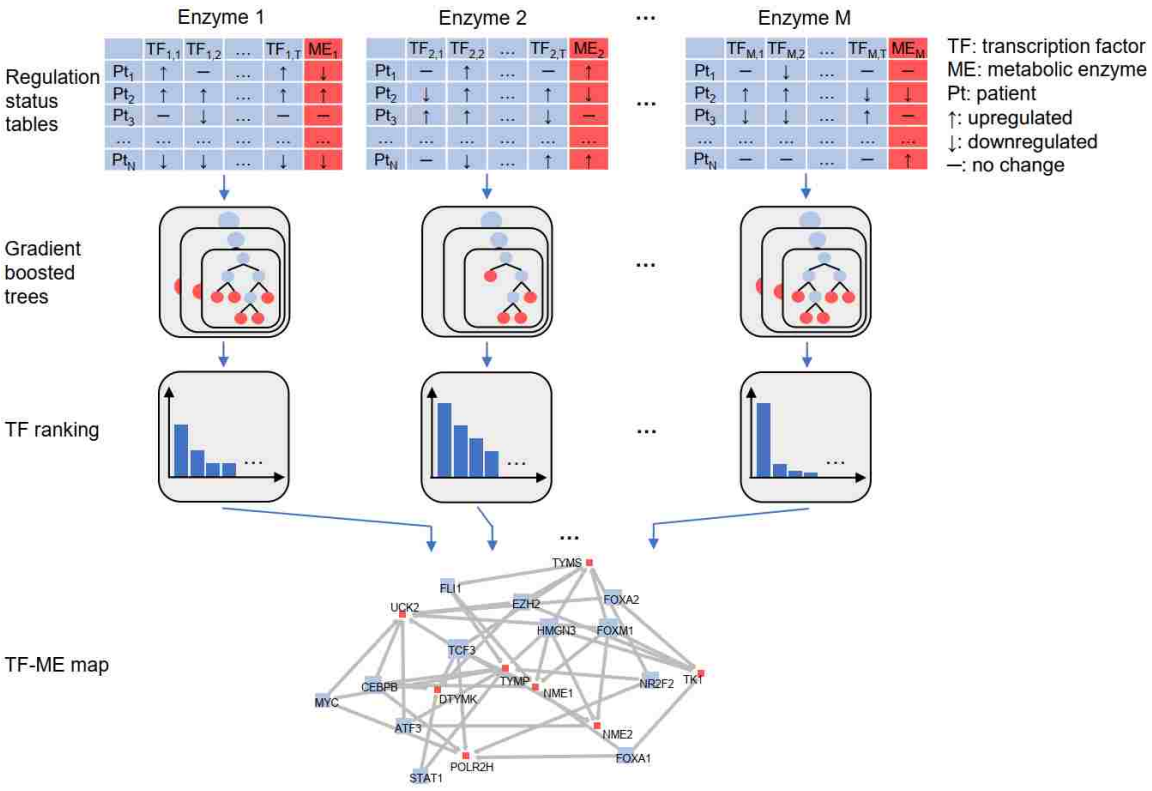

Figure 3.1 Overview of TF-metabolic enzyme interaction inference workflow. We divided the problem of inferring TF-metabolic enzyme interactions involving *M* enzymes into *M* sub-problems. In each sub-problem, taking the regulation status table of one enzyme and TFs binding to its transcription start site as input, we utilized gradient boosted trees to identify those TFs whose regulation status is predictive of the regulation status of the enzyme. This learning process was repeated on all the *M* enzymes. The predicted interactions between TFs and enzymes were then displayed in the TF-metabolic enzyme map as output.

3.2.4 Implementation

TFmeta was implemented using scikit-learn library (version 0.19.1) [102] and XGBoost library (version 0.7) [101] in Python (version 2.7.13) as task parallelized program. TFmeta [91] is freely available for academic use and can be accessible at https://github.com/zhangyimc/TFmeta.

3.3 Experimental results

3.3.1 Benchmarking TFmeta with DREAM5 Network Inference Challenge data sets

We utilized the data sets in Dialogue on Reverse Engineering Assessment and Methods (DREAM) 5 network inference challenge [85]. The DREAM project is a framework to enable an assessment of computational methods through standardized performance metrics and common benchmarks. DREAM5 challenge performed a comprehensive blind assessment of 35 TF-target gene interaction inference methods on *Escherichia coli*, *Staphylococcus aureus*, *Saccharomyces cerevisiae* and *in silico* microarray data. Table 3.1 summarizes the number of TFs, the number of genes, and the number of microarray chips for each network. DREAM5 challenge organizer claimed that *Staphylococcus aureus* data was not used for the final evaluation for the lack of a sufficiently large set of experimentally validated interactions. Each microarray data set is represented as a $m * n$ gene expression matrix, where m is the total number of genes including both TFs and target genes, and n is the total number of microarray measurements. Based on descriptions provided by participants, DREAM5 challenge classified the 35 competing methods into six distinct categories: regression, mutual information, correlation, Bayesian networks, meta (methods that combine several different approaches) and others (methods that do not belong to any of the previous categories).

Table 3.1 Summary of DREAM5 Challenge Data Sets

Network	Number of TFs	Number of genes	Number of microarray chips
In silico	195	1643	805
S.aureus	99	2810	160
E. coli	334	4511	805
S. cerevisiae	333	5950	536

TFmeta was trained and tested on the same benchmark data sets used by the 35 competing methods. Since the input data is numerical, the gene expression values generated from microarray chips, the functionality of classification and regression trees (CART) in TFmeta was shifted from classification to regression. In DREAM5 challenge, standardized performance metrics were provided to evaluate the performance of different methods. An overall score was used to summarize the performance across the three networks, which is a comprehensive assessment on both the area under the precision-recall (AUPR) and receiver operating characteristic (AUROC) curves. We applied the same metrics used by the 35 competing methods to TFmeta. Figure 3.2 (a) shows the overall scores for TFmeta and the 35 competing methods. The winner of DREAM5 challenge, GENIE3 [88], achieved an overall score of 40.279. The overall score of TFmeta is 69.031, which outperforms the winner of DREAM5 challenge.

Transcription-factor perturbation experiments can be applied to validate the biological significance of the TFs predicted by computational methods. However, the usage of transcription-factor perturbation experiments is limited by their high cost and strong dependence on cellular type and context. Though TF-target gene interaction inference methods reconstruct gene regulatory networks with a large set of regulatory interactions, the number of TFs chosen for further experimental validation is always limited, and it is

highly likely that only the top predicted interactions will be selected for further validation. We then evaluated the accuracy of the top interactions predicted by GENIE3 and TFmeta. As shown in Figure 3.2 (b), TFmeta consistently achieved a higher accuracy than GENIE3 for the top predictions on *in silico* data set, indicating that the most significant interactions predicted by TFmeta are more likely to be true interactions than those by GENIE3. We further compared TFmeta with GENIE3 in terms of computational efficiency. Figure 3.2 (c) illustrates the total CPU running time of GENIE3 and TFmeta for reconstructing the testing gene regulatory networks. It took GENIE3 761.58 hours to finish the entire reconstruction job, but only 6.03 hours for TFmeta. TFmeta is orders of magnitude faster than GENIE3.

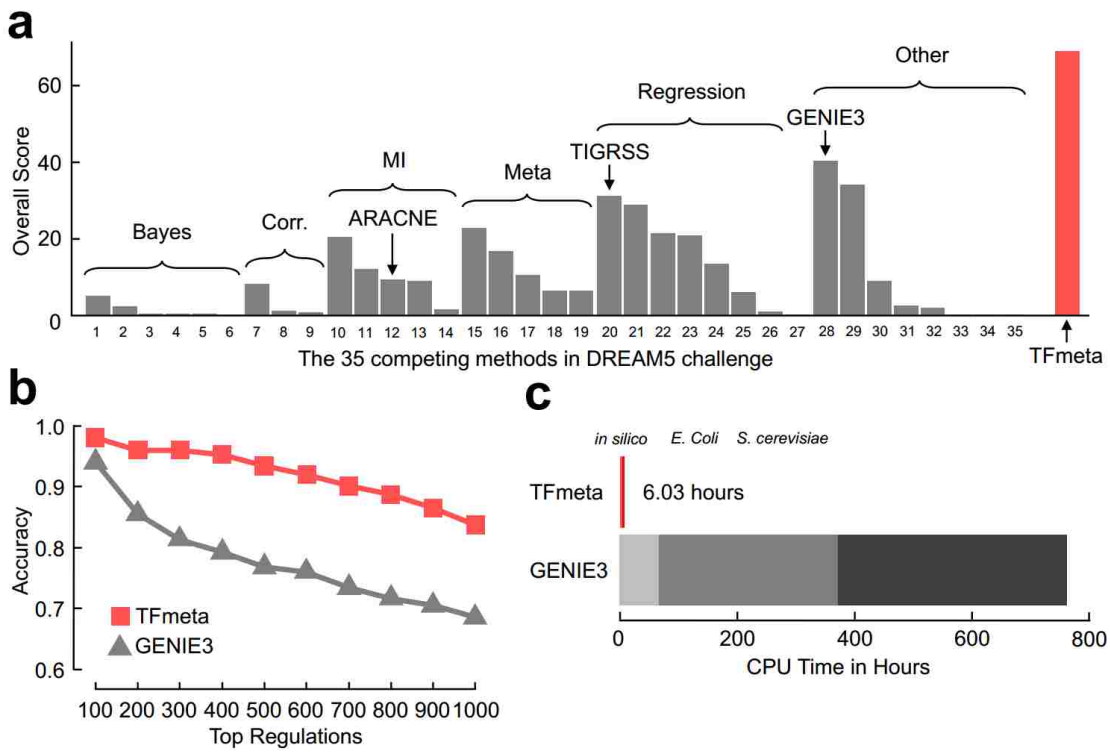

Figure 3.2 Performance evaluation of DREAM5 challenge data sets. (a) demonstrates the overall scores for TFmeta and the 35 competing methods. The winner of DREAM5 challenge, GENIE3, achieved an overall score of 40.279. The overall score of TFmeta is 69.031. (b) illustrates the accuracy of the top interactions predicted by GENIE3 and TFmeta. TFmeta consistently achieved a higher accuracy than GENIE3. (c) shows the total CPU running time of GENIE3 and TFmeta on the testing datasets. TFmeta is orders of magnitude faster than GENIE3.

3.3.2 Prediction of TFs governing the dysregulation of glycolysis in NSCLC patients

All parts of the body require energy to maintain non-equilibrium cellular states and perform work, and this energy is derived from consumption and oxidation of external nutrients. Typically, all food is broken down into smaller parts and coupled to the production of the main energy intermediate, ATP. ATP provides a uniformly usable store of biochemical energy that can be used to drive endergonic cellular reactions. The process of the breakdown of glucose, termed glycolysis, occurs in the cytoplasm of mammalian cells [103]. Since the early twentieth century, abnormalities of glycolysis in cancer cells

have been observed [104]. Marked progress has been made in understanding the molecular mechanisms leading to constitutive upregulation of glycolysis in tumor cells. Many glycolytic enzymes are often overexpressed in cancer cells. For example, phosphofructokinase-1 (PFK1) has been identified to be upregulated in types of breast cancer [105]. Another well-known classic glycolytic enzyme, glyceraldehyde-3-phosphate dehydrogenase (GAPDH) is also implicated in cancer. Overexpression of GAPDH is considered an important feature of numerous types of cancer [103]. GAPDH has been proposed as a promising target for the treatment of carcinomas [106]. Both MYC and HIF1a are known to upregulate expression of most of the glycolytic enzymes in cancers [107]. These results indicate that uncovering TFs that govern the abnormal expression patterns of these glycolysis and/or glycolytic enzymes in tumor cells may underlie the abnormalities of glycolysis, which could be highly effective for the treatment of different types of cancer.

We acquired 150 RNA-seq samples from 75 paired CA and NC human lung tissues. Through pairwise gene expression comparisons of CA and NA samples from the same patient, we identified 14 altered glycolytic enzymes with consistent expression changes. ENO1, ENO2, GAPDH, GPI, LDHA, PFKP, PKM, and TPI1 were upregulated, whereas ACSS2, ADH1B, ALDH2, ALDH3B1, FBP1, and HK3 were downregulated. Using a network editor, Omix [108], we visualized the patient-specific regulation status of part of glycolytic enzymes in four selected patients (UK022, UK059, UK084, and UK085) in the context of glycolysis pathway extracted from KEGG [94] pathway database, as shown in Figure 3.3. Each pie chart in Figure 3.3 depicts the regulation status of one enzyme in one patient. The pie chart with a larger slice of red (white) indicates the upregulation

(downregulation) of the enzyme. Though they are all non-small-cell lung cancer patients, individual differences in the regulation status of some enzymes can be observed. In the meanwhile, some well-known glycolytic enzyme, like PFKP, GAPDH, and PKM, are consistently upregulated in the four patients. This pairwise comparison analysis and patient-specific visualization eliminated the interferences from genetic and physiological variations, and also characterized the difference and consistency in regulation status among the patients.

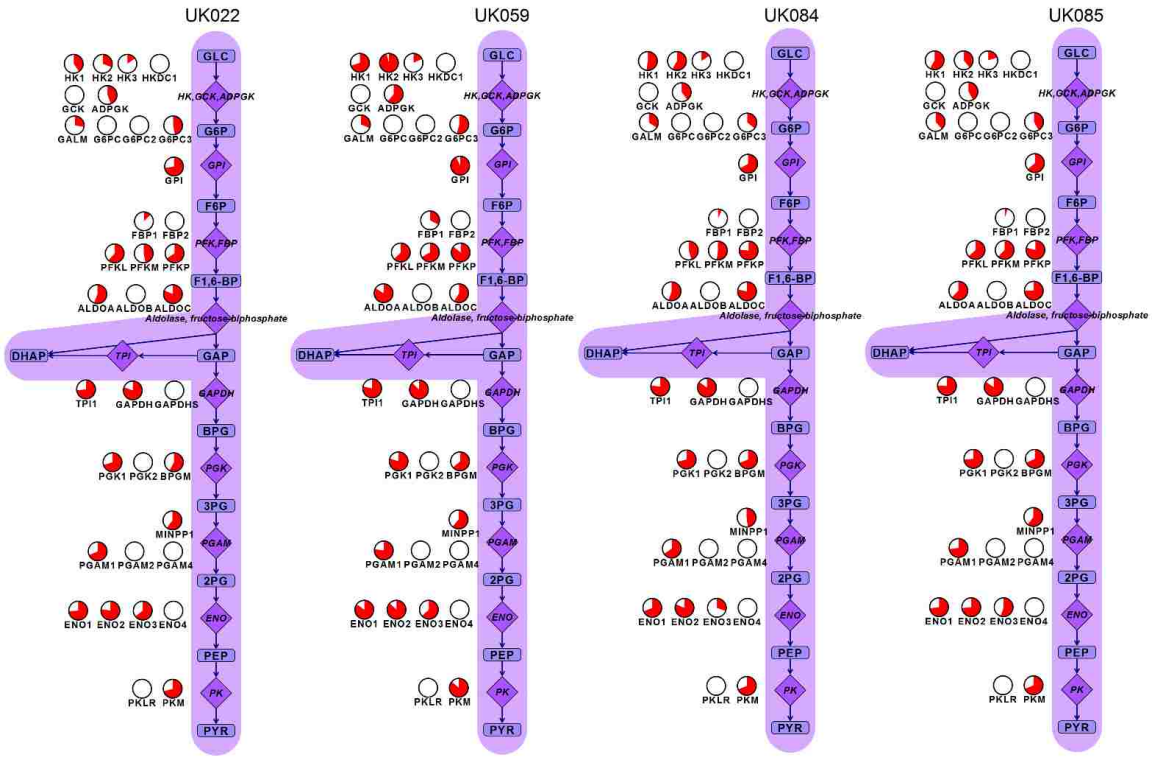

Figure 3.3 Visualization of the regulation status of part of glycolytic enzymes in the context of glycolysis pathway. We randomly selected four patients: UK022, UK059, UK084 and UK085 (from left to right). Each pie chart in the figure illustrates the regulation status of one enzyme in one patient. The pie chart with a larger slice of red (white) indicates the upregulation (downregulation) of the enzyme. Individual differences in the regulation status of some enzymes can be observed among the four patients. Meanwhile, some well-known glycolytic enzymes, like PFKP, GAPDH, and PKM, are consistently upregulated in the four patients. In total, twelve (three) out of thirty-five enzymes shown in the figure are consistently upregulated (downregulated) in the four patients. Glycolytic enzymes are more likely to be overexpressed in cancer cells.

For every altered glycolytic enzyme, we curated a list of TFs which bind to the transcription start site of that enzyme according. to the TF DNA binding activities inferred from ChIP-seq experiments. The average number of TFs selected for one enzyme is 134. We then fitted a gradient boosted tree-based classification model to predict the regulation status of each altered glycolytic enzyme based on the combined regulation status of the selected TFs. The optimal model configuration was achieved by extensive hyperparameter

search over various learning rate (0.001, 0.01, 0.1, and 1), maximum tree depth (1, 3, and 5), and number of rounds for boosting (100, 200, 300, and 400). To evaluate the performance of models with different parameter settings, 10-fold cross-validation was used. Table 3.2 summarizes the average prediction accuracy of models varying parameter settings upon the 14 altered glycolytic enzymes. Based on this results, we used 0.01 as learning rate, 3 as maximum tree depth, and 300 as number of rounds for boosting in our model to save the computing time without loss of classification accuracy.

Table 3.2 Performance evaluation of models with different parameter settings

Learning rate	Maximum tree depth	Number of rounds for boosting	Accuracy
0.001	3	300	0.696
0.01			0.723
0.1			0.661
1			0.634
0.01	1	300	0.714
	3		0.723
	5		0.723
0.01	3	100	0.679
		200	0.714
		300	0.723
		400	0.705

The application of TFmeta allows us to narrow down to a list of key TFs as modulating the dysregulated expression of those altered glycolytic enzymes. Figure 3.4 shows the TF-metabolic enzyme map predicted by TFmeta. In the map, the 14 altered glycolytic enzymes (red squares) and 19 predicted TFs (blue squares) are nodes, and an edge from one TF to one enzyme demonstrates that TF is predicted to regulate that enzyme, and all the edges are directed. Some predicted TFs and their interactions with glycolytic

enzymes in the map have already been supported by literature evidence. For example, transcription factor E2-alpha (TCF3) was identified as novel putative TF in lung cancer [109]. ETS Proto-Oncogene 1 (ETS1) was reported as a key TF involved in the metabolism of cancer cells, and ETS1 is particularly important in the metabolic shift towards glycolysis and anabolic means of energy production [110]. Enhancer of zeste homolog 2 (EZH2) promotes tumorigenesis and malignant progression in part by activating glycolysis. The mRNA expression of key enzymes involved in glycolysis in xenograft tumors was significantly increased in tumors derived from cells overexpressing EZH2, which suggests EZH2 overexpression leads to increases in glycolysis *in vivo* [111]. Forkhead box transcription factor-2 (FOXA2) was implicated as a suppressor of lung cancer, playing an important role in lipid and glucose metabolism in lung development using $Foxa2^{+/-}$ mice model [112]. Another well-known TF, MYC is a critical growth regulatory gene that is commonly overexpressed in a wide range of cancers. Overexpression of MYC leads to the upregulation of many glycolytic enzymes [113]. Zinc finger and BTB domain-containing protein 7A (ZBTB7A) acts as a tumor suppressor through the transcriptional repression of glycolysis, which directly binds to the promoter and represses the transcription of critical glycolytic enzymes, including GLUT3, PFKP, and PKM [114]. Krüppel-like factor 4 (KLF4) represses the transcription of the glycolytic enzyme LDHA in pancreatic cancer [115]. We propose that these TFs should be prioritized for follow-up experiments, both to validate predicted target metabolic enzymes and to evaluate specific biological functions for each TF. We further visualized the regulation status of 8 well-known classic glycolytic enzymes and 2 predicted TFs, KLF4 (known as a tumor suppressor gene in lung cancer) and EZH2 (known as an oncogene), in the 75 patients, as shown in Figure 3.5. The

regulation status of EZH2 and the 8 enzymes are positively correlated, on the contrary, KLF4 is negatively correlated with the 8 enzymes in terms of regulation status.

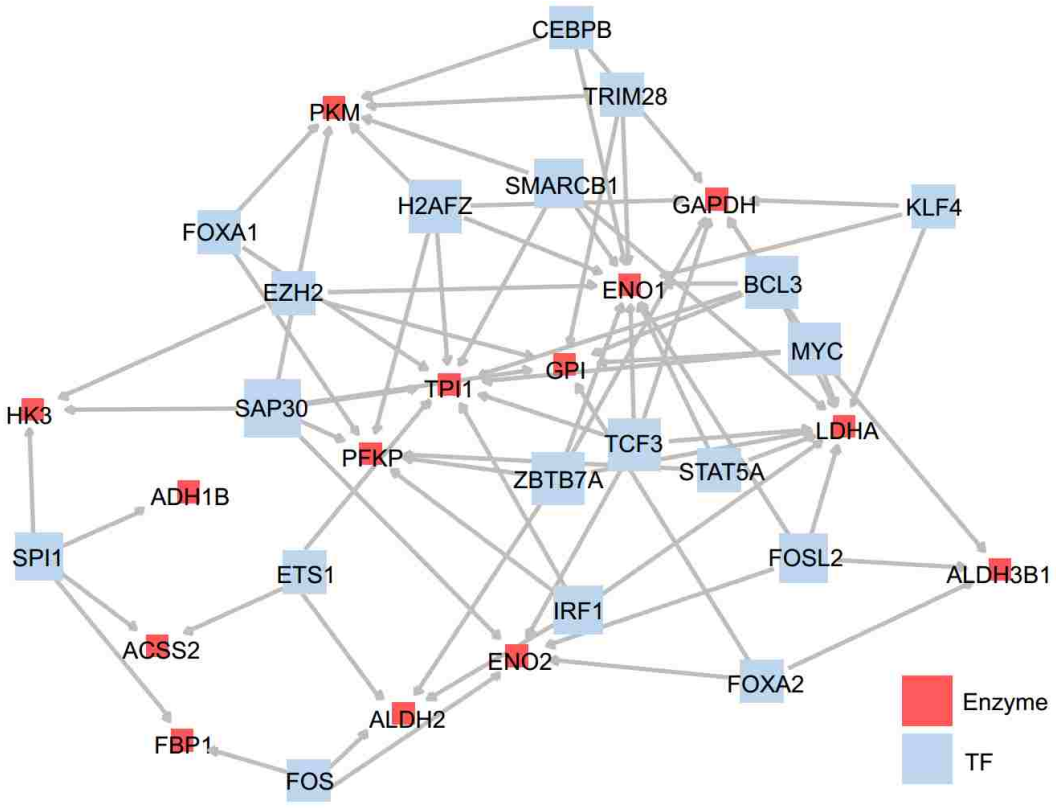

Figure 3.4 Visualization of the TF-metabolic enzyme map predicted by TFmeta. In the map, the 14 altered glycolytic enzymes (red squares) and 19 predicted TFs (blue squares) are nodes, and an edge from one TF to one enzyme demonstrates that TF is predicted to regulate that enzyme, and all the edges are directed.

Thus, in this pilot study, we demonstrated the feasibility of using TFmeta for uncovering TFs that govern glycolytic reprogramming in non-small-cell lung cancer patients. This approach should be equally powerful for deciphering other metabolic reprogramming in cancer cells, thereby enabling more comprehensive characterization of cancer metabolism.

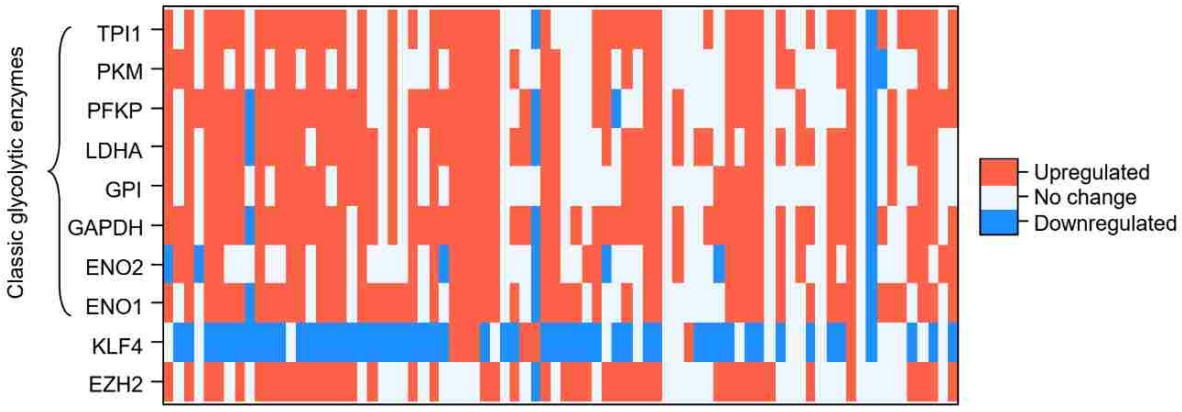

Figure 3.5 Heatmap of the regulation status of 8 well-known classic glycolytic enzymes and 2 predicted TFs, KLF4 and EZH2, in the 75 patients. The regulation status of EZH2 and the 8 enzymes are positively correlated, on the contrary, KLF4 is negatively correlated with the 8 enzymes in terms of regulation status. EZH2 is known as an oncogene, and KLF4 is a tumor suppressor gene in lung cancer.

3.3.3 Prediction of TFs governing other major metabolic pathways in NSCLC patients

We further applied TFmeta to infer TFs that govern other major metabolic pathways in non-small-cell lung cancer patients. The Krebs cycle is a central metabolic hub that integrates carbohydrate, lipid, and amino acid metabolism. The pentose phosphate pathway (PPP) is an alternative route for glycolysis, yielding ribose 5-phosphate for nucleotide biosynthesis and NADPH for fatty acid biosynthesis and decomposition of peroxides [116]. Purine metabolism maintains cellular pools of adenylate and guanylate via synthesis and degradation of purine nucleotides. The top TFs predicted for each metabolic pathway are shown as follows:

a) The Krebs cycle: ZBTB7A, MYC, SMARCB1, TAL1, TCF7L2;

b) The pentose phosphate pathway: FOXA2, MYC, EGR1, TCF3, ZEB1;

c) Purine metabolism: MYC, H2AFZ, EZH2, NFIC, ETS1, TCF3, BHLHE40,

CEBPB, STAT1, MAFK.

3.4 Summary

Metabolic reprogramming of cancer cells is recognized as one of the hallmarks of cancer. Tumors remarkably elevate the expression of the majority of metabolic enzymes, which play active roles in promoting cancer survival, metastasis, and invasion. One of the most common trends in anti-cancer metabolism therapies is to inhibit metabolic enzymes that are exclusively or mostly expressed or used in tumor cells. This therapeutic strategy would effectively eliminate tumors while minimizing damage to normal cells [117]. Thus, targeting TFs that control the transcription rate of those metabolic enzymes could be highly effective for novel cancer therapy.

In this work, we develop TFmeta, a machine learning approach to uncover TFs governing cancer metabolic reprogramming and reconstruct their interactions with metabolic enzymes. We demonstrated that TFmeta achieved state-of-the-art performance in recovering TF-target gene interactions on public benchmark data sets. We applied our model to non-small-cell lung cancer patients' data sets to predict TFs modulating the dysregulation of glycolysis in lung cancer, leveraging the pairing information of the samples and TF DNA binding activities that conventional approaches discard. Eventually, we predicted a list of key TFs that may motivate the upregulation of glycolysis observed in tumor cells, some of which have been supported by literature evidence, and some of which were predicted as novel putative TFs in lung cancer. Our model can also be easily deployed to uncover TFs governing other metabolic pathways, in addition to glycolysis.

Based on our results, we found the majority of metabolic enzymes have interactions

with more than one TFs. TFs are known to have to work together to achieve needed specificity in both DNA binding and effector function [118]. In our current model, the analysis of TF-TF relationships is generally lacking. TFmeta could extend its functionality to evaluate the associations of TFs in future.

Due to the high variability in tumor morphology and the low signal-to-noise ratio inherent to mammography, manual classification of mammogram and tomosynthesis yields a significant number of patients being called back, and subsequent large number of biopsies performed to reduce the risk of missing cancer. The convolutional neural network (CNN) is a popular deep learning construct used in image classification. This technique has achieved significant advancements in large-set image-classification challenges in recent years. In this study, we had obtained over 3000 high-quality original mammograms and tomosynthesis with approval from an institutional review board at the University of Kentucky. Different classifiers based on CNNs were built to classify both the 2D mammograms and 3D tomosynthesis, and each classifier was evaluated based on its performance relative to truth values generated by histology results from biopsy and two-year negative mammogram follow-up confirmed by expert radiologists. Our results showed that CNN model we had built and optimized via data augmentation and transfer learning have a great potential for automatic breast cancer detection using mammograms and tomosynthesis.

4.1 Introduction

Breast cancer is the most common cancer in women. Approximately 40,000 breast cancer patients die each year in the U.S [119]. Early detection of cancer significantly reduces the death rate [120]. To find breast cancer in early stages, before patients exhibit symptoms, women are recommended to undergo a screening test, commonly a mammogram. Mammography entails exposing a patient's breasts to low levels of X-ray

radiation. Breast cancer are identifiable from mammograms thanks to the different X-ray absorption rates of normal and abnormal tissues. Tumors can appear as masses, distortions or micro-calcifications on mammograms [121]. In patient with dense breast tissue, the tumor mass may overlap with the dense tissue, creating masking effect and making mammography less sensitive. Breast tomosynthesis is a newly emerging breast imaging technique first approved by the FDA in 2011. It takes multiple X-ray images at different angles; the images are then reconstructed to yield a video from which a radiologist can identify abnormalities. Compared to traditional mammograms, tomosynthesis provides more accurate results because tumors can be more easily distinguished from dense tissues using images taken from different angles [122]. Normally, mammograms and tomosynthesis were acquired in two standard orientations: Craniocaudal (CC) and Medial-lateral-oblique (MLO) views during screening. Figure 4.1 shows an example of the CC and MLO views of mammogram of two breasts, and Figure 4.2 shows an example of the multiple slices of the right CC view of tomosynthesis from the same patient.

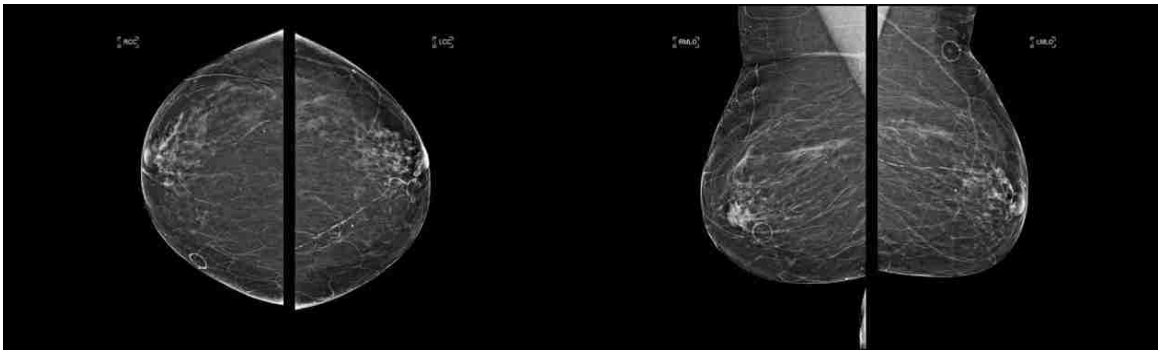

Figure 4.1 Illustration of 2D mammogram (from left to right): right CC view, left CC view, right MLO, left MLO view.

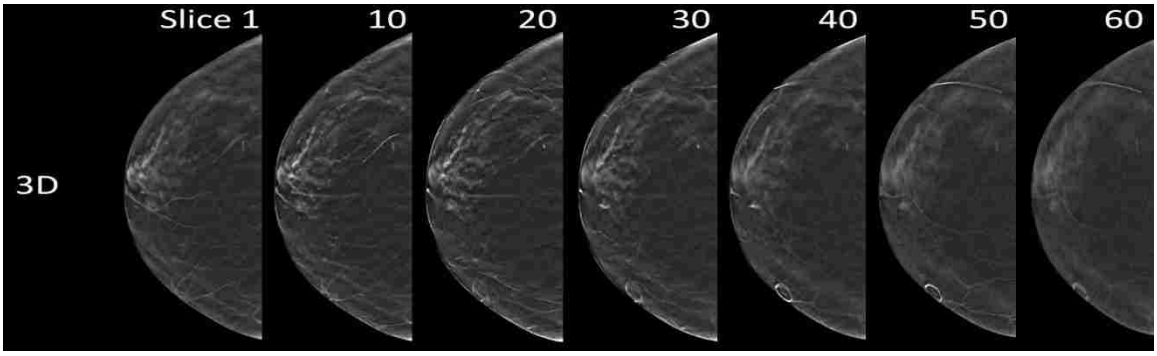

Figure 4.2 Illustration of 3D tomosynthesis: multiple slices of right CC view.

Screening mammography is the only imaging modality that has been proven to reduce breast cancer mortality [123]. However, mammography is also associated with high recall rates and high false-positive results [124]. With current practice, approximately 10% of all women screened for breast cancer are called back for additional work-ups, but only 0.5% are diagnosed with breast cancer (that is, 5 cancer detected out of 1,000 women screened, or 5 out of the 100 women called back). The use of the new technology, tomosynthesis in conjunction with mammography, was showed to improve the accuracy of cancer detection [125]. However, manual classification by radiologists still incurs a high recall rate and requires years of experience on the part of the radiologist. This high recall rate results in an abundance of additional diagnostic tests, including biopsy, and thus contributes to increased health-care costs as well as unnecessary emotional turmoil for the patients themselves [121, 126, 127].

Deep learning with convolutional neural networks has emerged as one of the most powerful machine-learning tools in image classification, surpassing the accuracy of almost all other traditional classification methods and even human ability [128]. The convolutional process can simplify an image containing millions of pixels to a set of small feature maps,

thereby reducing the dimension of input data while retaining the most important differential features. The application of CNNs to classify mammograms is not entirely new. However, most of the work focus on the classification of small patches, referred to as region of interest (ROI) [129]. An ROI is the region that is likely to contain a tumor. This is typically carved out of the whole images based on either clinical information or automatic segmentation. Daniel Lévy et al. used deep CNNs on small patches of mammograms, achieving a maximum accuracy of 93% [130]. Neeraj Dhungel et al. built a deep learning based method that automatically segments the area of lesions and then classifies the mammogram. Their best results were 0.74 for whole image, 0.8 for whole image plus automatically detected small lession patches, and 0.91 for whole image plus manually segmented small patches in terms of auROC [131]. In general, the classification of mammograms using small abnormality patches affords reasonable performance but requires very extensive pre-processing work.

An effective classification model for whole mammograms would offer multiple benefits, including (a) saving the work of annotating partial mammograms and its associated segmentation errors, (b) optimizing the use of contextual information surrounding tumors, (c) closely representing the real-world clinical practice, and (d) reducing the patient call-back rate, and thus the number of unnecessary tests conducted, without harming sensitivity. However, classification with whole images is much more challenging than with small patches due to the increased size and feature space. The best performance reported on whole mammography classification with CNN is 60.90% in terms of accuracy by Henry Zhou et al [132].

In this work, we developed and evaluated a number of CNN models for whole-mammography image classification [133]. We also present the first breast cancer classification model using 3D tomosynthesis data, a relatively new technology that is only available to 20% of major hospitals in the US. All images were collected at the Department of Radiology, University of Kentucky with an institutional review board approval (IRB17-0011-P3K). Techniques including data augmentation [134] and transfer learning [135] are combined with CNN models to optimize the performance of the classifiers.

4.2 Architecture overview

Our approach employs deep convolutional neural networks to classify whole-mammography images from both the 2D mammograms and 3D tomosynthesis data. The pipeline consists of three stages: data augmentation, transfer learning and CNNs. Ten different models in total were developed, optimized and compared through cross validation [136].

4.2.1 Data augmentation

Generally, deep neural networks require training on a large number of training samples to perform well. However, biomedical datasets like ours contain a relatively small number of samples due to limited patient volume. Data augmentation is a method for increasing the size of input data by generating new data from the original input data. Many strategies exist for image data augmentation [5, 128]. This study employed a combination of reflection and rotation. For the 2D mammograms, each original image was flipped horizontally. The original and reflected images were then rotated by each of 90, 180, and

270 degrees. Each original image was thus augmented to eight images. For each tomosynthesis sample, the 3D image sequences as a whole were either horizontally flipped or not flipped, and then randomly rotated 0, 90, 180 or 270 degrees. Such data augmentation generates relevant training samples because tumors may present themselves in various orientations.

The data augmentation can be performed either before the training or during training. Frontloading the augmentation process reduces the running time of the tests but requires 8 times more disk space to store all images. While this is applicable for 2D images, for the 3D tomosynthesis data, data augmentation was performed during the training phase to minimize storage usage.

4.2.2 Transfer learning

Transfer learning is the re-use of information obtained during the training phase of a previous project. In the field of image classification, the CNNs [128] trained in the course of successful projects are sometimes published for use by other researchers. Two popular transfer-learning methods involve (a) fine-tuning the parameters in certain layers of the trained CNN, or (b) using the trained CNN to calculate the feature maps of new types of data.

Mammography data is different from natural image data due to its limited color distribution and structures. However, it can still leverage the basic image features in terms of edges and shapes that can be soundly detected by well-trained CNN models. This study utilizes AlexNet [128], trained with ImageNet [137]. Considering the fact that mammograms differ dramatically from the images in the ImageNet dataset, the trained

AlexNet was used only to obtain the feature maps. Each image in the augmented dataset was resized to 832*832, which resolution was chosen with the goal of retaining tumor pixel information. The ImageNet trained AlexNet was deployed to generate the feature maps for the resized images. AlexNet output the feature maps with the shape of 25*25*256. The feature maps were then used in the training of the following shallow CNNs.

4.2.3 CNN architectures

We have built different architectures of convolutional neural networks to classify the 2D mammograms and 3D tomosynthesis images. A general shallow CNN architecture is shown in Figure 4.3. Each convolution layer (Conv layer) includes convolution, batch normalization [138], leaky ReLU [139] and max pooling [128]. All CNNs used Max pooling with stride 2. The optimizer used is the Adam optimizer [65]. L2 regularization was introduced in the loss function to prevent overfitting [66]. Dropout was also included to improve the model performance [67]. We also adopted two top-performing CNN architectures, AlexNet [128] and ResNet50 [140], to classify the 2D whole mammograms. Additionally, we have built several models incorporating transfer learning with feature maps learned from AlexNet. Detailed mathematical description of each step is omitted in this paper as they are well established deep learning techniques.

Figure 4.3 Sample convolutional neural network architecture used in this study. Conv layer denotes the convolution, batch normalization, leaky ReLU and max pooling process. Conv layers are followed by fully connected layers (Fully conn) and output layer.

The complete list of architectures is provided in Table 4.1. During the training phase, learning rate, dropout and L2 regularization beta were tuned with the range of 0.0001 to 0.1, 0.25 to 1 and 0.00001 to 0.1 respectively. The learning rate decay rate of Adam optimizer was set to 0.985 based on the preliminary results. The batch size was set by two rules: (1) power of two and (2) largest data size can fit the 8 GB memory.

Table 4.1 Detailed architectures of tested models for 2D mammogram and 3D tomosynthesis classification

Architecture	*Transfer Learning*	*Input Shape*	*conv1*	*conv2*	*conv3*	*fc1*	*fc2*	*Output*
2D-A1	No	224*224*3	6@5*5	16@3*3	--	1024	1024	2
2D-A2	No	224*224*3	16@3*3	32@3*3	64@3*3	1024	1024	2
AlexNet	No	224*224*3	--	--	--	--	--	2
ResNet50	No	224*224*3	--	--	--	--	--	2
2D-T1-Alex	Yes	25*25*256	256@1*1	--	--	1024	--	2
2D-T2-Alex	Yes	25*25*256	256@1*1	--	--	1024	1024	2
2D-T3-Alex	Yes	25*25*256	256@1*1	--	--	512	512	2
3D-A1	No	128*128*16*3	16@3*3*3	32@3*3*3	64@3*3*3	1024	1024	2
3D-T1-Alex	Yes	25*25*16*256	32@3*3*3	--	--	256	256	2
3D-T2-Alex	Yes	25*25*16*256	256@1*1*1	--	--	256	256	2

Imbalanced data represent a common problem in machine-learning projects [141]. If imbalances in the training data are not considered, the resulting model generally performs well on the larger class but poorly on the smaller class. The target dataset for this

study was classically imbalanced, with roughly 90% of samples representing negative diagnoses. To reduce the imbalance effect, the mini-batches [138] selected during the training phase were restricted to be balanced. During each training epoch, the training data were randomly split into m folds:

$$m = \frac{N_{pos}}{n/2}$$

where N_{pos} denotes the number of positive samples (smaller class) in the training set, and n is the batch size. In each iteration, all positive samples (n/2 samples) and n/2 randomly selected negative samples of 1-fold training data were fed to train the CNN.

For the data input, 2D mammograms and their feature maps were read as three dimensional tensors with shape defined as length*width*channels. 3D tomosynthesis data and their feature maps were read as four dimensional tensors with shape defined as length*width*depth*channels. Here, depth denotes the number of frames of 3D tomosynthesis data, which may vary across tomosynthesis samples. To obtain a fixed input shape, an equal number of frames were selected for each sample. Selected frames were start from frame 0 and with equal interval in one tomosynthesis sample. In this study, the frame number was set to 16 to fit the hardware limitation.

4.2.4 Implementation and performance evaluation

The convolutional neural networks were implemented using TensorFlow [70]. All the tests were performed on a machine with two groups of four Nvidia GTX 1080 GPUs, each with 8 GB memory.

To evaluate the performance of each prediction model, cross validation was used. The dataset was randomly partitioned into training and testing datasets. The training set

was used to train the model; the results of predictions made on the testing set were used to evaluate the performance of the model. The training-testing ratio used in all validation tests was 4:1.

Receiver operating characteristic curve (ROC) [142] is plotted as the true-positive rate versus the false-positive rate at various thresholds. The area under the ROC curve, auROC is used to measure the performance of a binary classifier. Tradeoffs can be made based on ROC curves to select the most appropriate classification model. When testing the prediction models in this study, probability of all test samples in each class was calculated. Using each value in the probability set as the threshold, we can derive true-positive rates (TPRs) and false-positive rates (FPRs). These TPR-FPR data were then used to plot the ROC curve and calculate the auROC.

4.3 Experimental results

4.3.1 Data description

High-quality mammogram data from the University of Kentucky Medical Center were obtained with institutional review board approval (IRB 17-0011-P3K). All mammography images were assessed by experienced radiologists. The dataset includes 3,018 negative and 272 positive mammogram images. Each of the positive image contains at least one biopsy-proven malignant tumor. The negative images do not contain malignant tumors confirmed with at least 2- year negative mammogram follow-up assessed by radiologists, but may have benign masses approved by biopsy or established more than 2 - year imaging stability. All exams in the dataset were taken in either CC or MLO view or

both. Negative images originated from 793 patients, most of which had 4 images taken: namely, CC and MLO views for both left and right breasts. Positive samples originated from 125 patients. Most positive patients have two images collected: CC and MLO views of the breast site with tumor. For each exam, both 2D mammogram and 3D tomosynthesis results were obtained. The 2D mammograms were provided in 12-bit DICOM format at 3328*4096 resolution. The 3D tomosynthesis images were provided in 8-bit AVI format with a resolution of 768*1024. Table 4.2 summarizes the dataset used in this study. All data were de-identified to protect the patients' privacy. In order to save storage space and reduce the time of file I/O, the pixel array for each 2D mammogram DICOM file was saved as a 16-bit JPEG image. For each 3D tomosynthesis AVI file, all frames were processed to a set of 8-bit JPEG images for the same purpose. The total number of frames for each 3D tomosynthesis exam varies from 21 to 120.

Table 4.2 2D mammogram and 3D tomosynthesis data used in this study

View	***Negatives***	***Positives***
RCC	758	77
RMLO	759	73
LCC	751	64
LMLO	750	58
Total	**3018**	**272**

4.3.2 Effect of data augmentation

Data augmentation increases the size of the training dataset 8-fold. It significantly improves the performance of almost every CNN architecture tested by roughly 0.1 auROC units. Figure 4.4 (A) and (B) depicts the training loss status of architecture 2D-T2 without and with data augmentation and Figure 4.4 (C) shows the associated ROC curves. The

auROC of the test with data augmentation is 0.73 comparing to 0.62 for the test without data augmentation. The training loss converged more smoothly with data augmentation than without. For this reason, all subsequent tests utilized the data augmentation strategy.

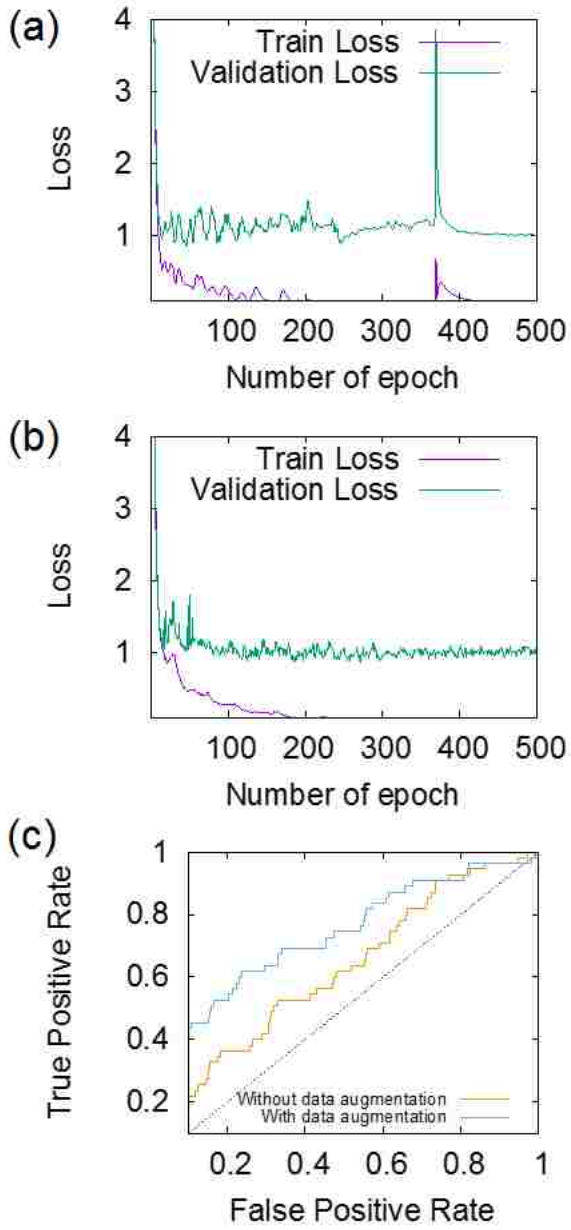

Figure 4.4 Loss converge status of tests using data without (a) and with augmentation (b) and ROC curves of them (c).

4.3.3 2D mammogram classification

We evaluated all CNN architectures on 2D mammography images listed in Table 4.1. The loss converge status during the training phase of all those architectures were shown in Figure 4.5. The optimized parameter combination and results of the best shallow-CNN model, the best classic-CNN model, and the best transfer-learning model for 2D mammograms are summarized in Table 4.3.

While classic-CNN models such as AlexNet do generate competitive results, the best architecture seems to be the one leveraging transfer-learning where feature maps derived from ImageNet-trained AlexNet are used for training. For example, 2D-T2-Alex delivers the best auROC approaching 0.73. The result suggests that utilizing the pre-trained model can be more sensitive in detecting key elements such as edges and shapes within a mammogram image as well. However, due to inherent difference between mammogram image and natural images, further training with these features using even a shallow CNN still delivers better classification accuracy than using AlexNet alone.

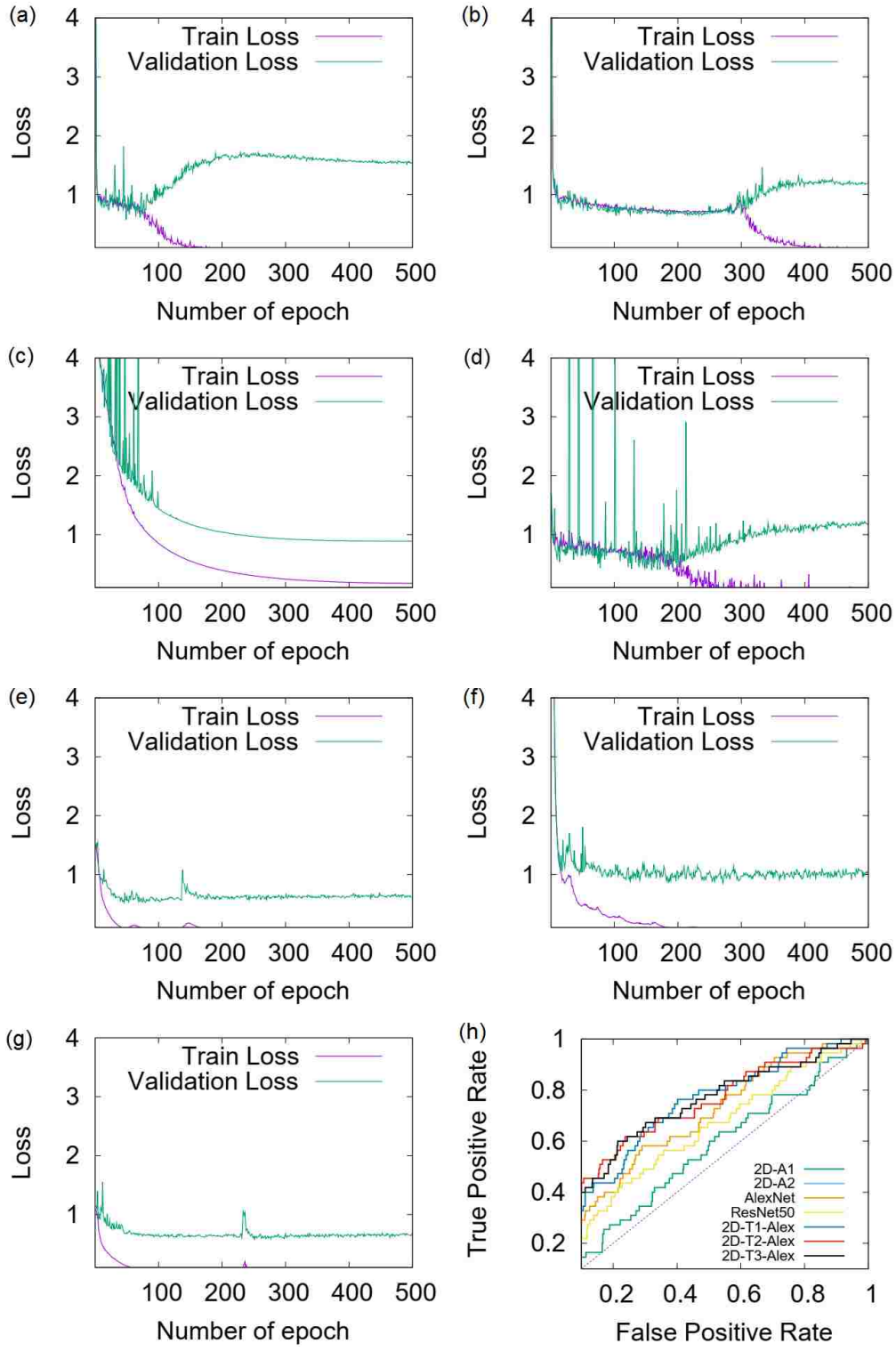

Figure 4.5 Loss converge status of 2D mammogram classification models: (a) 2D-A1, (b) 2D-A2, (c) AlexNet, (d) ResNet50, (e) 2D-T1-Alex, (f) 2D-T2-Alex, (g) 2D-T3-Alex. (h) illustrates the ROC curves of different models.

Table 4.3 Validation results and optimized parameter combination of 2D mammogram classification models

Architecture	*Batch size*	*Learning rate*	*Dropout*	*L2 regularization beta*	*auROC*
2D-A1	128	0.1	0.5	0.001	0.5488
2D-A2	128	0.1	1	0.001	0.6295
AlexNet	128	0.0001	0.25	0.001	0.6749
ResNet50	32	0.01	0.5	0.001	0.6239
2D-T1-Alex	256	0.0001	0.5	0.0001	0.7234
2D-T2-Alex	256	0.001	0.5	0.0001	0.7274
2D-T3-Alex	256	0.001	0.5	0.01	0.7237

4.3.4 3D tomosynthesis classification

We also evaluated three architectures listed in Table 4.1 designed for 3D tomosynthesis images. Cross validation was used to test one model, 3D-A1, on 3D tomosynthesis data, and two models on 3D tomosynthesis feature maps derived from transfer learning. The loss converge status during the training phase of all 3D classification architectures were shown in Figure 4.6. The optimized parameters and auROCs for the three models are shown in Table 4.4. Based on the tests, 3D-T2-Alex exhibited the best performance on 3D tomosynthesis feature maps; similar to 2D images, transfer learning using ImageNet-trained AlexNet was able to improve the performance of 3D tomosynthesis classification models.

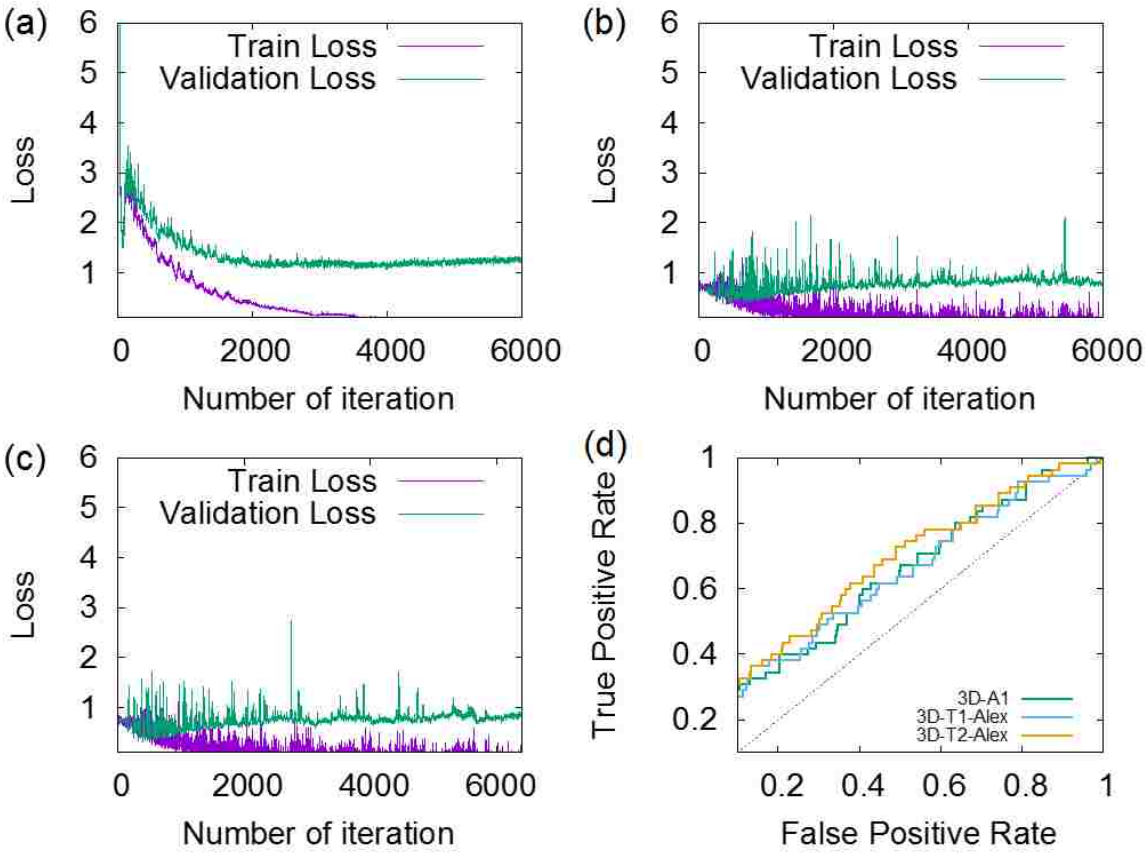

Figure 4.6 Loss converge status of 3D tomosynthesis classification models: (a) 3D-A1, (b) 3D-T1-Alex, (c) 3D-T2-Alex. (d) illustrates the ROC curves of different models.

Table 4.4 Validation results and optimized parameter combination of 3D tomosynthesis classification models

Architecture	***Batch size***	***Learning rate***	***Dropout***	***L2 regularization beta***	***auROC***
3D-A1	128	0.01	0.5	0.001	0.6312
3D-T1-Alex	16	0.01	0.5	0.0001	0.6116
3D-T2-Alex	16	0.0001	0.5	0.0001	0.6632

4.3.5 Comparison of classification results of 2D mammogram and 3D tomosynthesis

Our current results suggest that the 2D mammogram classification model performs slightly better than the 3D tomosynthesis classification model. However, radiologists generally achieve better classification accuracy on 3D tomosynthesis data. One possible

explanation for this phenomenon is that this study used only a subset of the 3D tomosynthesis frames due to memory limitations and the consistent shape requirement of the input. If the discarded frames contained information for diagnosing cancer that the selected frames lacked, then the frame sampling may have contributed to significant information loss. Another possible reason is that the 2D mammograms have better resolution than the 3D tomosynthesis data used in this study, such that the 2D mammograms may benefit from a higher signal-to-noise ratio [143].

4.4 Summary

This chapter reports our work on developing and optimizing machine learning models for whole image classification of both 2D and 3D mammograms. We evaluated 10 different CNN architectures and conclude that combining both data augmentation and transfer learning methods with a CNN is the most effective in improving classification performance.

We report the first work that study both 2D and 3D mammography images for breast cancer classification. Our current work sheds light on how each type of dataset performs when trained independently. But in practice, 2D and 3D images are complementary to each other, where 2D offers high resolution while 3D offers multiple views. One of our future work is to develop an assembled classifier that integrates the 2D and 3D data to achieve optimal performance.

3D tomosynthesis has proven to be much more powerful in manual detecting of tumors in clinical practice than conventional 2D imaging. However, 3D data is much more challenging to deal with, as it often corresponds to a much bigger feature space, requiring

a larger training dataset to obtain better performance and requiring more memory space for training. We believe there is still a great opportunity to improve the performance of 3D image classification model. We are currently collecting more images while simultaneously obtaining more precise annotation of each slice of 3D tomosynthesis data. Typically, only a few frames in 3D images of a positive exam do contain the tumor. Using negative frames within a positive exam may mislead the training of the model. In the meantime, we are also investigating alternative strategies, such as RNN model, that can leverage the sequence information among slices to perform classification.

CHAPTER 5. CONCLUSION

In the era of big data, transformation of biomedical big data into valuable biological insights has become one of the most important challenges in bioinformatics. Large quantities of biomedical data, including DNA/RNA sequencing data, and biomedical imaging data, have been generated. Modern machine learning techniques, such as deep learning, have been widely used in extracting meaningful patterns from big data sets. This dissertation presents three novel machine learning applications in different but closely related bioinformatics domains, two of them focus on next-generation sequencing data analysis, and the other one is designed for biomedical imaging data analysis.

Alternative splicing is a regulated process that enables the production of multiple mRNA transcripts from a single multi-exon gene. The availability of large-scale RNA-seq datasets has made it possible to predict splice junctions, as well as splice sites through spliced alignment to the reference genome. This greatly enhances the capability to decipher gene structures and explore the diversity of splicing variants. However, existing ab initio aligners are vulnerable to false positive spliced alignments as a result of sequence errors and random sequence matches. These spurious alignments can lead to a significant set of false positive splice junction predictions, confusing downstream analyses of splice variant detection and abundance estimation. In chapter 2, we illustrated that splice junction sequence characteristics can be ascertained from experimental data with deep learning techniques. We employed deep convolutional neural networks for a novel splice junction classification tool named DeepSplice. It performs better than the currently available splice site prediction tools. We found that there is valuable information to be gained from splice

junction sequences that conventional tools discard. The meaningful representations learning from input sequences improve accuracy.

Metabolic reprogramming is a hallmark of cancer. In cancer cells, transcription factors govern metabolic reprogramming through abnormally increasing or decreasing the transcription rate of metabolic enzymes, which provides cancer cells growth advantages and concurrently leads to the altered metabolic phenotypes observed in many cancers. Consequently, targeting transcription factors that govern metabolic reprogramming can be highly effective for novel cancer therapeutics. In chapter 3, we presented TFmeta, a machine learning approach to uncover transcription factors that govern reprogramming of cancer metabolism. Our approach achieves state-of-the-art performance in reconstructing interactions between transcription factors and their target genes on benchmark data sets. Leveraging TF binding profiles inferred from genome-wide ChIP-seq experiments and 150 RNA-seq samples from 75 paired cancerous and non-cancerous human lung tissues, our approach predicted 19 key TFs that may be the major regulators of the gene expression changes of metabolic enzymes of the central metabolic pathway glycolysis, which may underlie the dysregulation of glycolysis in non-small-cell lung cancer patients.

Mammography is the most popular technology used for breast cancer early detection. Manual classification of mammogram images is a difficult task because of the variability of tumors, which yields a noteworthy number of patients being called back to perform biopsies, ensuring no missing diagnosis. The convolutional neural network has succeeded in lots of image classification challenges recent years. In chapter 4, we designed an approach to perform 2D mammogram and 3D tomosynthesis classification based on convolutional neural networks. Our study demonstrated that CNN-based models with data

augmentation and transfer learning have good potential for automatic breast cancer detection based on the mammograms and tomosynthesis data.

All software packages of the models described in this dissertation are open-source, released, and freely available to the research community.

BIBLIOGRAPHY

1. Mohri, M., A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. 2018: MIT press.
2. Wikipedia, C. *Machine learning*. 18 May 2019 22:31 UTC 21 May 2019 20:06 UTC]; Available from: https://en.wikipedia.org/w/index.php?title=Machine_learning&oldid=897720835.
3. Wikipedia, C. *Learning to rank*. 14 May 2019 17:50 UTC 21 May 2019 20:13 UTC]; Available from: https://en.wikipedia.org/w/index.php?title=Learning_to_rank&oldid=897087901.
4. Wikipedia, C. *Cluster analysis*. 20 April 2019 21:19 UTC 21 May 2019 20:12 UTC]; Available from: https://en.wikipedia.org/w/index.php?title=Cluster_analysis&oldid=893363946.
5. LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* Nature, 2015. **521**(7553): p. 436-444.
6. Bengio, Y., *Learning deep architectures for AI.* Foundations and trends® in Machine Learning, 2009. **2**(1): p. 1-127.
7. Grapov, D., et al., *Rise of deep learning for genomic, proteomic, and metabolomic data integration in precision medicine.* Omics: a journal of integrative biology, 2018. **22**(10): p. 630-636.
8. Alipanahi, B., et al., *Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning.* Nature biotechnology, 2015. **33**(8): p. 831.
9. Chen, Y., et al., *Gene expression inference with deep learning.* Bioinformatics, 2016. **32**(12): p. 1832-1839.
10. Arvaniti, E. and M. Claassen, *Sensitive detection of rare disease-associated cell subsets via representation learning.* Nature communications, 2017. **8**: p. 14825.
11. Ma, J., et al., *Using deep learning to model the hierarchical structure and function of a cell.* Nature methods, 2018. **15**(4): p. 290.
12. Altae-Tran, H., et al., *Low data drug discovery with one-shot learning.* ACS central science, 2017. **3**(4): p. 283-293.
13. Nadeem, M., A. Hussain, and A. Munir, *Fuzzy logic based computational model for speckle noise removal in ultrasound images.* Multimedia Tools and Applications, 2019: p. 1-18.
14. Esteva, A., et al., *Dermatologist-level classification of skin cancer with deep neural networks.* Nature, 2017. **542**(7639): p. 115.
15. Chlebus, G., et al., *Automatic liver tumor segmentation in CT with fully convolutional neural networks and object-based postprocessing.* Scientific reports, 2018. **8**(1): p. 15497.
16. Mardani, M., et al., *Deep Generative Adversarial Neural Networks for Compressive Sensing MRI.* IEEE transactions on medical imaging, 2019. **38**(1): p. 167-179.
17. Birkhead, G.S., M. Klompas, and N.R. Shah, *Uses of electronic health records for public health surveillance to advance public health.* Annual review of public health, 2015. **36**: p. 345-359.
18. Charles, D. and M. Gabriel, *Adoption of electronic health record systems among US non-federal acute care hospitals: 2008-2014.*

19. Miotto, R., et al., *Deep patient: an unsupervised representation to predict the future of patients from the electronic health records.* Scientific reports, 2016. **6**: p. 26094.
20. Rajkomar, A., et al., *Scalable and accurate deep learning with electronic health records.* NPJ Digital Medicine, 2018. **1**(1): p. 18.
21. Chen, J., et al., *A natural language processing system that links medical terms in electronic health record notes to lay definitions: system development using physician reviews.* Journal of medical Internet research, 2018. **20**(1): p. e26.
22. Zhang, Y., et al., *Discerning novel splice junctions derived from RNA-seq alignment: a deep learning approach.* BMC genomics, 2018. **19**(1): p. 971.
23. Wang, E.T., et al., *Alternative isoform regulation in human tissue transcriptomes.* Nature, 2008. **456**(7221): p. 470-476.
24. Roy, B., L. M Haupt, and L. R Griffiths, *Review: alternative splicing (AS) of genes as an approach for generating protein complexity.* Current genomics, 2013. **14**(3): p. 182-194.
25. Cloonan, N., et al., *Stem cell transcriptome profiling via massive-scale mRNA sequencing.* Nature methods, 2008. **5**(7): p. 613-619.
26. Marioni, J.C., et al., *RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays.* Genome research, 2008. **18**(9): p. 1509-1517.
27. Mortazavi, A., et al., *Mapping and quantifying mammalian transcriptomes by RNA-Seq.* Nature methods, 2008. **5**(7): p. 621-628.
28. Sultan, M., et al., *A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome.* Science, 2008. **321**(5891): p. 956-960.
29. Dobin, A., et al., *STAR: ultrafast universal RNA-seq aligner.* Bioinformatics, 2013. **29**(1): p. 15-21.
30. Kim, D., B. Langmead, and S.L. Salzberg, *HISAT: a fast spliced aligner with low memory requirements.* Nature methods, 2015. **12**(4): p. 357-360.
31. Kim, D., et al., *TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions.* Genome biology, 2013. **14**(4): p. R36.
32. Wang, K., et al., *MapSplice: accurate mapping of RNA-seq reads for splice junction discovery.* Nucleic acids research, 2010: p. gkq622.
33. Wu, T.D. and S. Nacu, *Fast and SNP-tolerant detection of complex variants and splicing in short reads.* Bioinformatics, 2010. **26**(7): p. 873-881.
34. Mann, D.L., et al., *Braunwald's heart disease: a textbook of cardiovascular medicine*. 2014: Elsevier Health Sciences.
35. Li, Y., et al., *TrueSight: a new algorithm for splice junction detection using RNA-seq.* Nucleic Acids Research, 2013. **41**(4): p. e51-e51.
36. Nellore, A., et al., *Human splicing diversity and the extent of unannotated splice junctions across human RNA-seq samples on the Sequence Read Archive.* Genome Biology, 2016. **17**(1): p. 266.
37. Nellore, A., et al., *Rail-RNA: Scalable analysis of RNA-seq splicing and coverage.* Bioinformatics, 2016: p. btw575.
38. Hu, Y., et al., *DiffSplice: the genome-wide detection of differential splicing events with RNA-seq.* Nucleic acids research, 2012. **41**(2): p. e39-e39.
39. Trapnell, C., et al., *Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks.* Nature protocols, 2012. **7**(3): p. 562-578.

40. Gatto, A., et al., *FineSplice, enhanced splice junction detection and quantification: a novel pipeline based on the assessment of diverse RNA-Seq alignment solutions.* Nucleic acids research, 2014. **42**(8): p. e71-e71.
41. Pickrell, J.K., et al., *Noisy splicing drives mRNA isoform diversity in human cells.* PLoS genetics, 2010. **6**(12): p. e1001236.
42. Stormo, G.D., et al., *Use of the 'Perceptron' algorithm to distinguish translational initiation sites in E. coli.* Nucleic acids research, 1982. **10**(9): p. 2997-3011.
43. Noordewier, M.O., G.G. Towell, and J.W. Shavlik, *Training knowledge-based neural networks to recognize genes in DNA sequences.* Advances in neural information processing systems, 1991. **3**: p. 530-536.
44. Brunak, S., J. Engelbrecht, and S. Knudsen, *Prediction of human mRNA donor and acceptor sites from the DNA sequence.* Journal of molecular biology, 1991. **220**(1): p. 49-65.
45. Degroeve, S., et al., *SpliceMachine: predicting splice sites from high-dimensional local context representations.* Bioinformatics, 2005. **21**(8): p. 1332-1338.
46. Huang, J., et al., *An approach of encoding for prediction of splice sites using SVM.* Biochimie, 2006. **88**(7): p. 923-929.
47. Sonnenburg, S., et al., *Accurate splice site prediction using support vector machines.* BMC bioinformatics, 2007. **8**(10): p. S7.
48. Reese, M.G., et al., *Improved splice site detection in Genie.* Journal of computational biology, 1997. **4**(3): p. 311-323.
49. Pertea, M., X. Lin, and S.L. Salzberg, *GeneSplicer: a new computational method for splice site prediction.* Nucleic acids research, 2001. **29**(5): p. 1185-1190.
50. Baten, A.K., et al., *Splice site identification using probabilistic parameters and SVM classification.* BMC bioinformatics, 2006. **7**(5): p. S15.
51. Lee, T. and S. Yoon. *Boosted Categorical Restricted Boltzmann Machine for Computational Prediction of Splice Junctions*. in *ICML*. 2015.
52. Chuang, J.S. and D. Roth, *Splice Site Prediction Using a Sparse Network of Winnows*. 2001, University of Illinois at Urbana-Champaign.
53. Zhang, M.Q., *Identification of protein coding regions in the human genome by quadratic discriminant analysis.* Proceedings of the National Academy of Sciences, 1997. **94**(2): p. 565-568.
54. Zhang, Y., et al., *Splice site prediction using support vector machines with a Bayes kernel.* Expert Systems with Applications, 2006. **30**(1): p. 73-81.
55. Wei, D., et al., *A novel splice site prediction method using support vector machine.* J Comput Inform Syst, 2013. **920**: p. 8053-60.
56. Yeo, G. and C.B. Burge, *Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals.* Journal of computational biology, 2004. **11**(2-3): p. 377-394.
57. Zhang, Q., et al., *Splice sites prediction of Human genome using length-variable Markov model and feature selection.* Expert Systems with Applications, 2010. **37**(4): p. 2771-2782.
58. Ghandi, M., et al., *Enhanced regulatory sequence prediction using gapped k-mer features.* PLoS computational biology, 2014. **10**(7): p. e1003711.
59. Harrow, J., et al., *GENCODE: the reference human genome annotation for The ENCODE Project.* Genome research, 2012. **22**(9): p. 1760-1774.

60. Li, J., et al., *High-accuracy splice site prediction based on sequence component and position features.* Genet Mol Res, 2012. **11**(3): p. 3432-3451.
61. Zhang, Y., et al. *DeepSplice: Deep classification of novel splice junctions revealed by RNA-seq*. in *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. 2016. IEEE.
62. Montavon, G., et al., *Explaining nonlinear classification decisions with deep taylor decomposition.* Pattern Recognition, 2017. **65**: p. 211-222.
63. Senapathy, P., M.B. Shapiro, and N.L. Harris, *[16] Splice junctions, branch point sites, and exons: Sequence statistics, identification, and applications to genome project.* Methods in enzymology, 1990. **183**: p. 252-278.
64. Rampone, S., *Recognition of splice junctions on DNA sequences by BRAIN learning algorithm.* Bioinformatics, 1998. **14**(8): p. 676-684.
65. Kingma, D. and J. Ba, *Adam: A method for stochastic optimization.* arXiv preprint arXiv:1412.6980, 2014.
66. Ng, A.Y. *Feature selection, L 1 vs. L 2 regularization, and rotational invariance*. in *Proceedings of the twenty-first international conference on Machine learning*. 2004. ACM.
67. Srivastava, N., et al., *Dropout: a simple way to prevent neural networks from overfitting.* Journal of machine learning research, 2014. **15**(1): p. 1929-1958.
68. Konečný, J., et al., *Mini-batch semi-stochastic gradient descent in the proximal setting.* IEEE Journal of Selected Topics in Signal Processing, 2016. **10**(2): p. 242-255.
69. Smith, T.F. and M.S. Waterman, *Identification of common molecular subsequences.* Journal of molecular biology, 1981. **147**(1): p. 195-197.
70. Abadi, M., et al., *Tensorflow: Large-scale machine learning on heterogeneous distributed systems.* arXiv preprint arXiv:1603.04467, 2016.
71. Pollastro, P. and S. Rampone, *HS3D: homosapiens splice site data set.* Nucleic Acids Research, 2003(Annual Database).
72. Sercu, T., et al. *Very deep multilingual convolutional neural networks for LVCSR*. in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. 2016. IEEE.
73. Hogg, R.V. and A.T. Craig, *Introduction to mathematical statistics.(5"" edition)*. 1995: Upper Saddle River, New Jersey: Prentice Hall.
74. Burset, M., I. Seledtsov, and V. Solovyev, *Analysis of canonical and non-canonical splice sites in mammalian genomes.* Nucleic acids research, 2000. **28**(21): p. 4364-4375.
75. Leung, M.K., et al., *Deep learning of the tissue-regulated splicing code.* Bioinformatics, 2014. **30**(12): p. i121-i129.
76. Xiong, H.Y., et al., *The human splicing code reveals new insights into the genetic determinants of disease.* Science, 2015. **347**(6218): p. 1254806.
77. Sibley, C.R., L. Blazquez, and J. Ule, *Lessons from non-canonical splicing*. Nature Reviews Genetics, 2016. **17**(7): p. 407-421.
78. Vander Heiden, M.G., L.C. Cantley, and C.B. Thompson, *Understanding the Warburg effect: the metabolic requirements of cell proliferation.* science, 2009. **324**(5930): p. 1029-1033.

79. Wise, D.R., et al., *Myc regulates a transcriptional program that stimulates mitochondrial glutaminolysis and leads to glutamine addiction.* Proceedings of the National Academy of Sciences, 2008. **105**(48): p. 18782-18787.
80. Gottlieb, E. and I.P. Tomlinson, *Mitochondrial tumour suppressors: a genetic and biochemical update.* Nature Reviews Cancer, 2005. **5**(11): p. 857.
81. Paccez, J.D. and L.F. Zerbini, *Oncogenic Transcription Factors: Target Genes.* eLS, 2007.
82. Libermann, T.A. and L.F. Zerbini, *Targeting transcription factors for cancer gene therapy.* Current gene therapy, 2006. **6**(1): p. 17-33.
83. Yuneva, M.O., et al., *The metabolic profile of tumors depends on both the responsible genetic lesion and tissue type.* Cell metabolism, 2012. **15**(2): p. 157-170.
84. Gao, P., et al., *c-Myc suppression of miR-23a/b enhances mitochondrial glutaminase expression and glutamine metabolism.* Nature, 2009. **458**(7239): p. 762.
85. Marbach, D., et al., *Wisdom of crowds for robust gene network inference.* Nature methods, 2012. **9**(8): p. 796-804.
86. Bonneau, R., et al., *The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo.* Genome biology, 2006. **7**(5): p. R36.
87. Faith, J.J., et al., *Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles.* PLoS biology, 2007. **5**(1): p. e8.
88. Irrthum, A., L. Wehenkel, and P. Geurts, *Inferring regulatory networks from expression data using tree-based methods.* PloS one, 2010. **5**(9): p. e12776.
89. Haynes, B.C., et al., *Mapping functional transcription factor networks from gene expression data.* Genome research, 2013. **23**(8): p. 1319-1328.
90. Reimand, J., et al., *Comprehensive reanalysis of transcription factor knockout expression data in Saccharomyces cerevisiae reveals many new targets.* Nucleic acids research, 2010. **38**(14): p. 4768-4777.
91. Zhang, Y., et al. *TFmeta: A Machine Learning Approach to Uncover Transcription Factors Governing Metabolic Reprogramming*. in *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. 2018. ACM.
92. Li, B. and C.N. Dewey, *RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome.* BMC bioinformatics, 2011. **12**(1): p. 323.
93. Ishibashi, Y., et al., *Profiling gene expression ratios of paired cancerous and normal tissue predicts relapse of esophageal squamous cell carcinoma.* Cancer research, 2003. **63**(16): p. 5159-5164.
94. Kanehisa, M. and S. Goto, *KEGG: kyoto encyclopedia of genes and genomes.* Nucleic acids research, 2000. **28**(1): p. 27-30.
95. Lachmann, A., et al., *ChEA: transcription factor regulation inferred from integrating genome-wide ChIP-X experiments.* Bioinformatics, 2010. **26**(19): p. 2438-2444.
96. Consortium, E.P., *The ENCODE (ENCyclopedia of DNA elements) project.* Science, 2004. **306**(5696): p. 636-640.

97. Mathelier, A., et al., *JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles.* Nucleic acids research, 2015. **44**(D1): p. D110-D115.
98. Matys, V., et al., *TRANSFAC® and its module TRANSCompel®: transcriptional gene regulation in eukaryotes.* Nucleic acids research, 2006. **34**(suppl_1): p. D108-D110.
99. Li, H. and M. Zhan, *Unraveling transcriptional regulatory programs by integrative analysis of microarray and transcription factor binding data.* Bioinformatics, 2008. **24**(17): p. 1874-1880.
100. Friedman, J.H., *Greedy function approximation: a gradient boosting machine.* Annals of statistics, 2001: p. 1189-1232.
101. Chen, T. and C. Guestrin. *Xgboost: A scalable tree boosting system*. in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016. ACM.
102. Pedregosa, F., et al., *Scikit-learn: Machine learning in Python.* Journal of Machine Learning Research, 2011. **12**(Oct): p. 2825-2830.
103. Sreedhar, A. and Y. Zhao, *Dysregulated metabolic enzymes and metabolic reprogramming in cancer cells.* Biomedical Reports, 2018. **8**(1): p. 3-10.
104. Warburg, O., *Versuche an überlebendem Carcinomgewebe (Methoden).* Biochem. Zeitschr., 1923. **142**: p. 317-333.
105. Zancan, P., et al., *Differential expression of phosphofructokinase-1 isoforms correlates with the glycolytic efficiency of breast cancer cells.* Molecular genetics and metabolism, 2010. **100**(4): p. 372-378.
106. Krasnov, G.S., et al., *Deregulation of glycolysis in cancer: glyceraldehyde-3-phosphate dehydrogenase as a therapeutic target.* Expert opinion on therapeutic targets, 2013. **17**(6): p. 681-693.
107. Dang, C.V., A. Le, and P. Gao, *MYC-Induced Cancer Cell Energy Metabolism and Therapeutic Opportunities.* Clinical Cancer Research, 2009. **15**(21): p. 6479-6483.
108. Droste, P., et al., *Visualizing multi-omics data in metabolic networks with the software Omix—a case study.* Biosystems, 2011. **105**(2): p. 154-161.
109. El-aarag, S.A., et al., *In silico identification of potential key regulatory factors in smoking-induced lung cancer.* BMC medical genomics, 2017. **10**(1): p. 40.
110. Verschoor, M.L., et al., *Ets-1 regulates energy metabolism in cancer cells.* PLoS one, 2010. **5**(10): p. e13565.
111. Pang, B., et al., *EZH2 promotes metabolic reprogramming in glioblastomas through epigenetic repression of EAF2-HIF1α signaling.* Oncotarget, 2016. **7**(29): p. 45134.
112. Tang, X. and F. Luo, *The forkhead box transcription factor-2 (Foxa2) and lung disease.* Inflammation and Cell Signaling, 2014. **1**(4).
113. Sabnis, H.S., R.R. Somasagara, and K.D. Bunting, *Targeting MYC dependence by metabolic inhibitors in cancer.* Genes, 2017. **8**(4): p. 114.
114. Liu, X.-S., et al., *ZBTB7A acts as a tumor suppressor through the transcriptional repression of glycolysis.* Genes & development, 2014. **28**(17): p. 1917-1928.
115. Shi, M., et al., *A novel KLF4/LDHA signaling pathway regulates aerobic glycolysis in and progression of pancreatic cancer.* Clinical Cancer Research, 2014. **20**(16): p. 4370-4380.

116. Patra, K.C. and N. Hay, *The pentose phosphate pathway and cancer.* Trends in biochemical sciences, 2014. **39**(8): p. 347-354.
117. Phan, L.M., S.-C.J. Yeung, and M.-H. Lee, *Cancer metabolic reprogramming: importance, main features, and potentials for precise targeted anti-cancer therapies.* Cancer biology & medicine, 2014. **11**(1): p. 1.
118. Lambert, S.A., et al., *The human transcription factors.* Cell, 2018. **172**(4): p. 650-665.
119. Siegel, R.L., K.D. Miller, and A. Jemal, *Cancer Statistics, 2017.* CA Cancer J Clin, 2017. **67**(1): p. 7-30.
120. Tabár, L., et al., *Swedish two-county trial: impact of mammographic screening on breast cancer mortality during 3 decades.* Radiology, 2011. **260**(3): p. 658-663.
121. Poorolajal, J., et al., *Breast cancer screening (BCS) chart: a basic and preliminary model for making screening mammography more productive and efficient.* J Public Health (Oxf), 2017: p. 1-8.
122. Kopans, D.B., *Digital breast tomosynthesis: a better mammogram.* Radiology, 2013. **267**(3): p. 968-9.
123. Hellquist, B.N., et al., *Effectiveness of population-based service screening with mammography for women ages 40 to 49 years.* Cancer, 2011. **117**(4): p. 714-722.
124. Nelson, H.D., et al., *Harms of breast cancer screening: systematic review to update the 2009 US preventive services task force recommendationharms of breast cancer screening.* Annals of internal medicine, 2016. **164**(4): p. 256-267.
125. Gennaro, G., et al., *Digital breast tomosynthesis versus digital mammography: a clinical performance study.* European radiology, 2010. **20**(7): p. 1545-1553.
126. Kim, S.Y., et al., *Breast Cancer Detected at Screening US: Survival Rates and Clinical-Pathologic and Imaging Factors Associated with Recurrence.* Radiology, 2017. **284**(2): p. 354-364.
127. Tosteson, A.N., et al., *Consequences of false-positive screening mammograms.* JAMA internal medicine, 2014. **174**(6): p. 954-961.
128. Krizhevsky, A., I. Sutskever, and G.E. Hinton. *Imagenet classification with deep convolutional neural networks*. in *Advances in neural information processing systems*. 2012.
129. Hua, K.-L., et al., *Computer-aided classification of lung nodules on computed tomography images via deep learning technique.* OncoTargets and therapy, 2015. **8**.
130. Lévy, D. and A. Jain, *Breast mass classification from mammograms using deep convolutional neural networks.* arXiv preprint arXiv:1612.00542, 2016.
131. Dhungel, N., G. Carneiro, and A.P. Bradley. *Fully automated classification of mammograms using deep residual neural networks*. in *Biomedical Imaging (ISBI 2017), 2017 IEEE 14th International Symposium on*. 2017. IEEE.
132. Zhou, H., Y. Zaninovich, and C. Gregory, *Mammogram Classification Using Convolutional Neural Networks.*
133. Zhang, X., et al., *Classification of whole mammogram and tomosynthesis images using deep convolutional neural networks.* IEEE transactions on nanobioscience, 2018. **17**(3): p. 237-242.

134. Tanner, M.A. and W.H. Wong, *The calculation of posterior distributions by data augmentation.* Journal of the American statistical Association, 1987. **82**(398): p. 528-540.
135. Pan, S.J. and Q. Yang, *A survey on transfer learning.* IEEE Transactions on knowledge and data engineering, 2010. **22**(10): p. 1345-1359.
136. Zhang, X., et al. *Whole mammogram image classification with convolutional neural networks*. in *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. 2017. IEEE.
137. Deng, J., et al. *Imagenet: A large-scale hierarchical image database*. in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009. IEEE.
138. Ioffe, S. and C. Szegedy. *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. in *International Conference on Machine Learning*. 2015.
139. Xu, B., et al., *Empirical evaluation of rectified activations in convolutional network.* arXiv preprint arXiv:1505.00853, 2015.
140. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
141. Chawla, N.V., N. Japkowicz, and A. Kotcz, *Special issue on learning from imbalanced data sets.* ACM Sigkdd Explorations Newsletter, 2004. **6**(1): p. 1-6.
142. Hanley, J.A. and B.J. McNeil, *The meaning and use of the area under a receiver operating characteristic (ROC) curve.* Radiology, 1982. **143**(1): p. 29-36.
143. Johnson, D.H., *Signal-to-noise ratio.* Scholarpedia, 2006. **1**(12): p. 2088.

VITA

Yi Zhang

- Education
  - ¤ Wuhan University of Science and Technology

    B.E. in Computer Science, 09/2010 – 06/2014

- Employment History
  - ¤ University of Kentucky

    Research Assistant, 08/2014 - Present
  - ¤ Genentech, Inc.

    Intern, 06/2018 – 09/2018

- Publications
  - ¤ Zhang, Yi, Xinan Liu, James MacLeod, and Jinze Liu. "Discerning novel splice junctions derived from RNA-seq alignment: a deep learning approach." *BMC genomics* 19, no. 1 (2018): 971.
  - ¤ Zhang, Yi, Xinan Liu, James N. MacLeod, and Jinze Liu. "DeepSplice: Deep classification of novel splice junctions revealed by RNA-seq." In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 330-333. IEEE, 2016.

- ¤ Zhang, Yi, Xiaofei Zhang, Andrew N. Lane, Teresa WM Fan, and Jinze Liu. "TFmeta: A Machine Learning Approach to Uncover Transcription Factors Governing Metabolic Reprogramming." In *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pp. 351-359. ACM, 2018.
- ¤ Zhang, Xiaofei, Yi Zhang, Erik Y. Han, Nathan Jacobs, Qiong Han, Xiaoqin Wang, and Jinze Liu. "Classification of whole mammogram and tomosynthesis images using deep convolutional neural networks." *IEEE transactions on nanobioscience* 17, no. 3 (2018): 237-242.
- ¤ Yu, Ye, Jinpeng Liu, Xinan Liu, Yi Zhang, Eamonn Magner, Erik Lehnert, Chen Qian, and Jinze Liu. "SeqOthello: querying RNA-seq experiments at scale." *Genome biology* 19, no. 1 (2018): 167.
- ¤ Zhang, Xiaofei, Yi Zhang, Erik Y. Han, Nathan Jacobs, Qiong Han, Xiaoqin Wang, and Jinze Liu. "Whole mammogram image classification with convolutional neural networks." In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 700-704. IEEE, 2017.

- Scientific Abstracts
  - ¤ Gene expression patterns that distinguish equine articular cartilage from other cartilaginous tissues

    Emma Adam, Yi Zhang, Xinan Liu, Stephen Coleman, Robert Flight, Joshua Lambert, Arnold Stromberg, Jinze Liu, James MacLeod

In *The American College of Veterinary Surgeons (ACVS) Surgery Summit*, 2017.

- Comparative chondrogenic potential of equine fetal progenitor cells and adult mesenchymal stem cells

  Emma Adam, James MacLeod, Yi Zhang, Xinan Liu, Jinze Liu

  In *International Plant & Animal Genome XXIV*, 2016.