Histopathology Image analysis and NLP for Digital Pathology

by

Aishwarya Krishna Allada

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2021

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Chapter 6 is based on the following paper:

- Aishwarya Krishna Allada and Yuanxin Wang and Veni Jindaland Babaie Morteza and Hamid Reza Tizhoosh and Mark Crowley. “Analysis of Language Embeddings for Classification of Unstructured Pathology Reports”. (Accepted to 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) 2021)

I have contributed to implementation, experimentation, and preparation of the manuscript of the above mentioned paper

Abstract

Information technologies based on ML with quantitative imaging and texts are playing an essential role particularly in general medicine and oncology. DL in particular has demonstrated significant breakthroughs in Computer Vision and NLP which could enhance disease detection and the establishment of efficient treatments. Furthermore, considering the large number of people with cancer and the substantial volume of data generated during cancer treatment, there is a significant interest in the use of AI to improve oncologic care.

In digital pathology, high-resolution microscope images of tissue samples are stored along with written medical reports in databases which are used by pathologists. The diagnosis is made through tissue analysis of the biopsy sample and is written as a brief unstructured report which is stored as free text in Electronic Medical Record (EMR) systems. For the transition towards digitization of medical records to achieve its maximum benefits, these reports must be accessible and usable by medical practitioners to easily understand them and to help them precisely identify the disease.

Concerning the histopathology images, which is the basis of diagnosis and study of diseases of the tissues, image analysis helps us identify the disease's location and allows us to classify the type of cancer. Recently, due to the abundant accumulation of WSIs, there has been an increased demand for effective and efficient gigapixel image analysis, such as computer-aided diagnosis using DL techniques. Also, due to high diversity of shapes and structures in WSIs, it is not possible to use conventional DL techniques for classification. Though computer-aided diagnosis using DL has good prediction accuracy, in the medical domain, there is a need to explain the prediction of the model to have a better understanding beyond standard quantitative performance evaluation.

This thesis presents three different findings. Firstly, I provide a comparative analysis of various transformer models such as BioBERT, Clinical BioBERT, BioMed-RoBERTa and TF-IDF and our results demonstrates the effectiveness of various word embedding techniques for pathology reports in the classification task. Secondly, with the help of slide labels of WSIs, I classify them to their disease types, with an architecture having attention mechanism and instance-level clustering. Finally, I introduced a method to fuse the features of the pathology reports and the features of their respective images. I investigated the effect of combination of the features in the classification of both histopathology images and their respective reports simultaneously. This proved to be better than the individual classification tasks achieving an accuracy of 95.73%.

Acknowledgements

Writing this thesis has been fascinating and extremely rewarding.

I would like to sincerely thank my supervisor, Dr. Mark Crowley, who made this thesis possible by providing his constant support, encouragement, patience, valuable research direction and insight throughout my degree. I am glad to be part of his lab, UWECEML, where so many researchers are free to explore creative ideas.

I would also like to thank Prof. Hamid Tizhoosh and Dr. Morteza Babaie for their constant support and guidance for my experiments.

I am also thankful to Prof. Hamid Tizhoosh and Prof. Olga Vechtomova for taking the time to provide an in-depth review of the thesis and provide very helpful feedback. Your efforts and support are deeply appreciated.

Last but not least, thanks to my parents, brother and grandparents. The support they showed me during my research and thesis is the reason I could achieve this milestone.

Dedication

This thesis is dedicated to my parents, Srinivasa Rao Allada and Madhavi Allada, for raising me to value education and their unconditional love, no matter the circumstances. To my brother, Rahul Kishan Allada for his infinite support and motivation and my grandparents - Adatrow Gurunath and Sridevi for their care, affection and encouragement.

To my companion, Chanakya Jetti, who gave me the warmth of a family, away from home and for being my constant support system. You are one of the best things that have happened in my life.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AI Artificial Intelligence iv, 2, 4

ANN Artificial Neural Network 8, 11

ANNs Artificial Neural Networks 6, 8, 17

BERT Bidirectional Encoder Representations from Transformers 23, 29

BioBERT Bidirectional Encoder Representations from Transformers for Biomedical Text Mining 28–30, 33, 36

CNN Convolution Neural Network 9, 17, 21, 40, 49

CNNs Convolution Neural Networks 6, 8, 14, 17, 20

CV Computer Vision 4, 6, 11, 17, 21

DL Deep Learning iv, 2, 4, 6, 8, 20, 21, 37

DNN Deep Neural Network xii, 3, 5, 14, 15, 20, 32, 33, 36, 44, 45, 47, 50

GDC Genomic Data Commons 24, 25

GRU Gated Recurrent Unit 12, 16

IDF Inverse Document Frequency 30

LSTM Long Short Term Memory 6, 12, 16

ML Machine Learning iv, 2, 6–8, 18, 20

NCI National Cancer Institute 24

NHGRI National Human Genome Research Institute 24

NLP Natural language Processing iv, 3, 6, 11, 16–18, 20, 28, 29, 33

OCR Optical Character Recognition 26

PLMs pretrained language models 28

RNN Recurrent Neural network 9, 11, 12

RNNs Recurrent Neural Network 8, 11, 12, 16

SPR Surgical Pathology Reports 23

SSL self-supervised learning 28

TCGA The Cancer Genome Atlas 24

TF–IDF Term Frequency – Inverse Document Frequency 30, 31, 33, 35, 36, 50

TF-IDF Term Frequency-Inverse Document Frequency iv, 3, 28

WSI Whole Slide Image 2, 4, 21, 37–41, 44, 45

WSIs Whole Slide Images iv, 2–4, 21, 37–40, 42, 44

Chapter 1

Introduction

One of the leading causes of death globally is cancer, a group of diseases characterized by abnormal cell growth and the ability to invade or spread to other parts of the body. Both researchers and doctors are facing the challenges of fighting cancer [67]. In the United States, approximately 1.8 million new cancer cases were diagnosed in 2020, with 606,520 cancer deaths. Due to the increase in the mortality rate due to cancer, a rapid advancement concerning cancer research is being developed [53]. Early identification of cancer is essential to save the lives of many people. Generally, visual examination and manual techniques are used for cancer diagnosis, where manual interpretation of medical slides is time-consuming and prone to errors. In order to bypass the above cases, in the early 1980s [38], computer-aided diagnosis (CAD) systems were introduced to assist doctors in improving the efficiency of medical image interpretation.

Histopathology is the analysis and interpretation of size, shape, and patterns present in the cells and tissues of a patient's clinical records and other factors to study disease manifestation. "Histopathology" evolved from combining two of the major branches of science, namely, "histology" and "pathology." While histology involves studying microscopic structures of tissues, pathology involves the disease diagnosis with the help of microscopic examinations of surgically removed specimens. Throughout the healthcare delivery system, histopathology is a vital discipline exclusively studied and practiced by pathologists. The primary duty is to conduct a microscopic analysis of glass slides containing tissue specimens to render pathology reports. The pathology reports created by the pathologist are used for various reasons such as development of a diagnostic plan, screening for diseases, monitoring the disease severity, etc. Understanding and explaining the images of tissues and cells at high resolutions is the core of histopathology.

Digital pathology is a digital version of traditional microscopy that is used to examine glass pathology slides. Though Light Microscopy(LM) is considered as the standard reference for diagnosis in pathology, recent developments in ML and AI have drastically improved diagnosing of disease in digital pathology. Thus, in recent times, there has been an increase in the adoption of digital pathology, with pathology laboratories all over the world gradually replacing their microscopes with digital scanners and computers.

DL is a type of AI and ML, that imitates the working of human brain in data processing and creates patterns for decision making. There has been a significant increase of computational power and recent advancements in the technology of AI, especially DL, which is being used in multiple areas of health care like medical image diagnosis, digital pathology, prediction of hospital admission, drug design, classification of cancer and stromal cells, doctor assistance, etc. [141]. With the advancing technology, the pathology community started digitizing glass slides into gigapixel images through virtual microscopy or Whole Slide Imaging (WSI). The gigapixel images generated via Whole Slide Imaging are the digital slides, and with a large number of increase in digital slides, resulting in a massive creation of databases with WSIs. DL methods, along with their outstanding pattern-recognition capabilities when applied to these digital databases of WSIs, have significantly increased the value of digital pathology. In addition, computer-aided operations like segmentation of tissues and cell nuclei and identifying the cancer type and the severity have become possible after the glass pathology slides have been digitized.

Apart from the digital databases consisting of a massive number of WSIs related to unique cases, each of the WSIs is also accompanied by their respective diagnostic reports, which are written by the pathologists, after the tissue analysis of the biopsy sample. With the increase in the applications of Digital Pathology for research and teleconsultation, there is a growing increase in the amount of histopathology data. In this context, this thesis presents research of applying computer vision, Natural Language Processing and ML algorithms for the quantitative image and text-based analysis of histopathology images and reports.

1.1 Classification of Pathology Reports

In almost all cases, the diagnosis of a whole slide image WSI is made through tissue analysis which is described in a detailed format in a pathology report. Pathology reports hold important details of the WSI analysis, which is stored in most electronic medical record systems as unstructured free data. A part of clinical research focuses is on the manual extraction of data from pathology reports, which is very expensive and time-consuming.

The data helps identify the case, select the treatment and its plan, risk stratification, etc. The electronic medical record systems hold the records of essential details of the patient's health and pathologist's elucidation of the findings from the WSIs. It also has the information that helps physicians understand the details of WSIs and records that information for future clinical and research use.

For almost 50 years, researchers have worked to develop Natural Language Processing NLP algorithms to extract details from pathology reports. However, only a limited number of categorical data elements are typically extracted, and model outputs often lack reliable uncertainty estimates, limiting the clinical applicability of these systems and only 10% of which have been reported to be in real-world use [105]. Also, many attempts are being made to store the pathology report information in a specific template, but most of the information is recorded in free text. This text is a massive hurdle for immediately extracting and using the details and information a report holds by clinicians, researchers, and healthcare information systems. This ambiguity is due to the complexity of natural language, the complexity of pathology images description explained by a variety of sources [54].

Due to the above reasons, the pathology reports are not feasible in using automated methods to suggest proper treatment or promote clinical research. On top of that, a massive volume of pathology reports are produced every year. For instance, an average-sized laboratory produces more than 50,000 reports annually. Most of these reports have no direct connection to the tissue samples. Also, each patient's report is a customized document with high vocabulary discrepancies, such as misspelled words and lack of punctuation. It is common to find clinical diagnoses intermixed with nuanced explanations and multiple terminologies used to mark the same malignancy and data about various carcinomas in a single report [46]. Cancer registries are facing a considerable challenge in the manual analysis of the enormous quantity of pathology reports, with the rise in the number of patients with cancer and the improvement in treatment complexity [46] [134]. Moreover, the process of figuring out the diseases from a pathology report is challenging, time-consuming and requires extensive training when done manually [46].

The first part of this thesis demonstrates how to extract meaningful numerical embeddings from written pathology reports to help classify various types of cancer. One of our research's primary focuses is to evaluate and compare the effectiveness of existing machine learning methods for the automatic classification of a given pathology report to its respective disease type. We demonstrate that contextualized word embeddings combined with TF-IDF feature vectors, when given as inputs to a DNN, can be an effective method for classification, achieving 93.77% accuracy in our study. This study proves the competence of TF-IDF features and its effectiveness in recognizing essential keywords in a pathology report. Additionally, our experiments with digital pathology reports will allow researchers to develop a versatile way of extracting essential details from free-text pathology reports, which could benefit various medical diagnostic tasks.

1.2 Classification of Histopathology Images

Histopathology image analysis is used to help pathologists identify tumors and their subtypes and eases pathologist's workload. The most recent studies in digital pathology have found that supervised AI algorithms for classification are compelling and vital use of these digitized images is used to create diagnostic algorithms or applications which can augment the diagnostic workflow. Histopathology WSIs contains a massive number of pixels, where these pixels can now become part of a DL algorithm to look for shapes, features or patterns utilizing image analysis, DL and AI tools [61]. DL is at the forefront of CV, showcasing significant improvements over previous methodologies on visual understanding [37] of a pathology image. The emergence of DL is becoming very beneficial, especially in digital pathology, where it is practically not very easy to detect, segment and identify a given WSI having thousands of pixels to a specific disease type.

Recently, the Food and Drug Administration has approved usage of WSIs for primary diagnosis [108]. On the other hand, though DL is widely used for computer-aided diagnosis, there is always a trade-off between the accuracy of the model and its interpretability. In addition, General Data Protection Regulation (GDPR) has claimed the "right to explanation" for any decision made by artificially intelligent algorithmic systems. Moreover, in the medical domain, the model's prediction is given more importance than other fields because any mishap can become fatal for the patient.

The second part of this thesis demonstrates the result of the experiment performed for the classification of WSIs into their respective cancer types. The experiment is carried out in a weakly supervised setting with an attention scoring mechanism and instance-level clustering, yielding an accuracy of 90.12%.

1.3 Combined Classification of Histopathology Images and Pathology Reports

Both images and texts are essential for human intelligence to understand the real world. Many types of researches [43, 48, 83] were performed to bridge these two modalities. In order to investigate the effectiveness of combining image and language to predict and gain

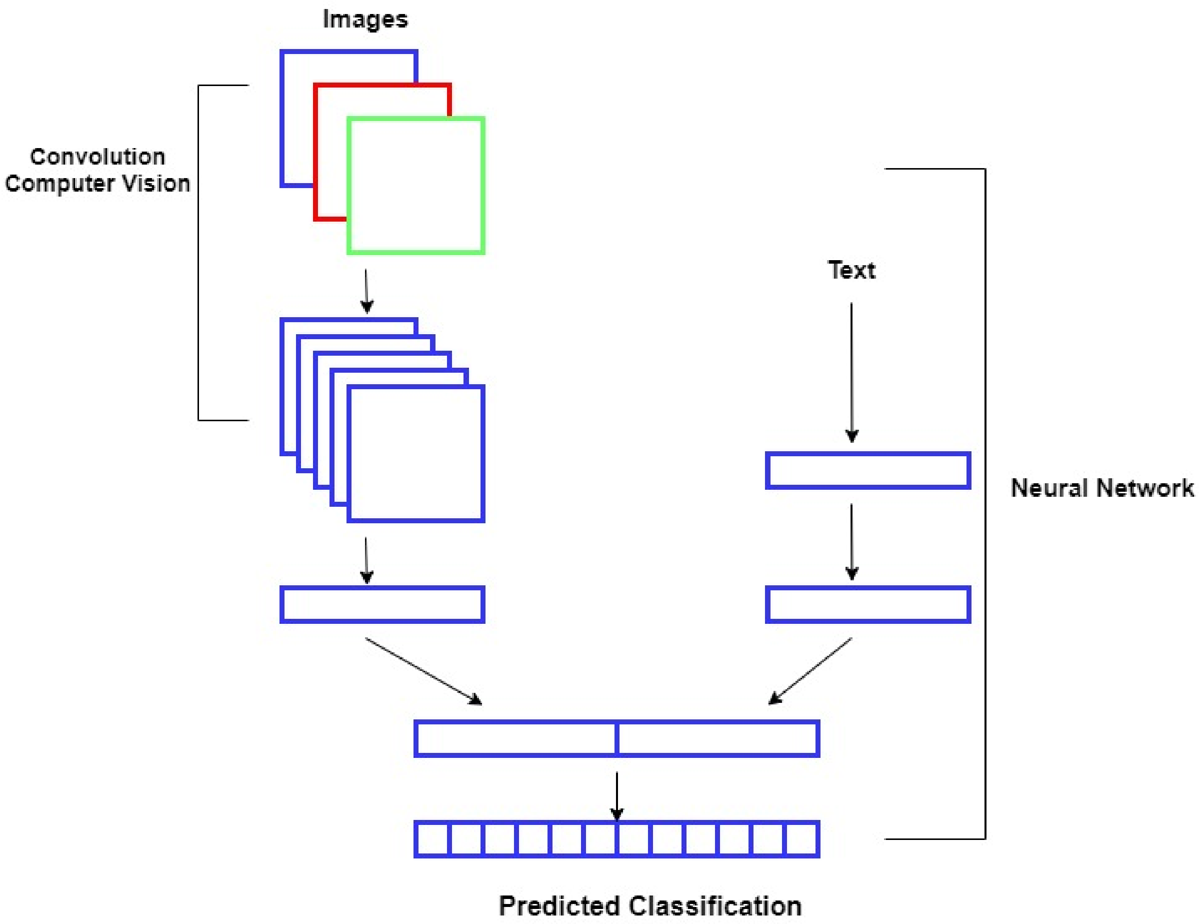


Figure 1.1: Combined Image + Text Model

insight, we proposed a method to classify the histopathology images by using associated metadata, i.e., their associated reports as shown in Figure 1.1. Intending to classify both histopathology reports and images together as they contain complementary information, the proposed method combines the features of the report's texts and images from the experiments in section 1.1 and section 1.2. These concatenated images and text features are used to classify images and reports using a DNN classifier. The experiment proved that the combination of features gave comparatively good accuracy of 95.7% compared to the individual experiments on histopathology reports and images, respectively.

Chapter 2

Background

This chapter explains some of the components and building blocks used in our experiments. Firstly, various classes of ML models are outlined, after which we introduce Deep Learning DL and ANNs. We also introduce CNNs, class of ANNs which are generally used in CV and NLP literature and also LSTM networks, which are generally used in NLP literature. Also, theory regarding regularization, sequence-to-sequence models, attention mechanisms are introduced as we used them in our image classification model. Also, we brief about embeddings, which is an important concept in NLP.

2.1 Machine Learning

Machine Learning ML enables computers to learn from data, without being explicitly programmed, where the data is used to train the system to perform any specific task. In ML, model recognizes the patterns and intricacies within the data with the help of some form of mathematical optimization and statistical methods. The trained model can further automate tasks or guide decision-making based on data and the mathematical model.

ML is being used predominantly in our everyday lives and is developing at a fast pace. For example, all the email services use ML to filter out spam emails, while online shopping provides us with recommendations that use ML. In addition, several kinds of research are being performed daily with new algorithms and methodologies developed and applied in many areas ranging from the medical field to climate change. These advancements in intelligent systems are beneficial as it makes all the application's process more accessible and at the same time with less human intervention.

Generally speaking, there are two categories of ML methodologies, which are namely supervised learning and unsupervised learning, where the primary categorization is based on the type of learning process carried. The subsection below elaborates on each of the categories with examples.

2.1.1 Supervised Learning

Supervised learning algorithms are trained with data points containing the features (inputs) and their respective labels (outputs). This algorithm usually modifies the model parameters so that the desired output for a given input is obtained. In this algorithm, we have output labels that correspond to each input used to train the model.

Once the model is trained, it is then provided with new unseen data points as inputs, where the model will predict the target based on what it has previously learned. A few of the most popular supervised learning algorithms which are commonly used in classification and regression studies are linear and logistic regression [29], Support Vector Machine [14], Naive Bayes Classifier [98], Gradient Boosting [15], [44], classification trees, and random forest [16], [112].

To better understand the supervised learning method, let us consider an example with the image classification task. Initially, we feed the ML model with images of Golden Retriever, huskies, poodles, etc. and label them as dogs. Similarly, we can provide images of Persian, Maine, British shorthair, etc., all labeled as cats. Given the input images and the respective labels, the model tries to learn from the image pixel the characteristics that differentiate dogs from cats. Once after training, the model is then used to classify new images.

2.1.2 Unsupervised Learning

Unsupervised learning algorithms are fed with training data points containing the features (inputs) and do not require pre-existing output/labels. These algorithms can identify hidden patterns based on the distribution of the input data. This learning method can be used for detecting anomalies, wherein some parts of the data may not fit well with the rest of the data. One of the most common unsupervised learning methods is Clustering (e.g., hierarchical clustering [122], [34], K-means [89], [96]) and few other popular approaches are Latent Dirichlet Allocation (LDA)[12], principal component analysis (PCA) [45] and word2vec [102].

For example, suppose we train a model with unlabelled documents related to various topics like sports, politics, movies, etc. In that case, a model will automatically cluster similar documents that belong to similar topics, with the information provided in the documents, such as word usage and writing style.

2.2 Deep Learning

ANNs are an important class of machine learning models, used for both supervised and unsupervised tasks, where biological neural networks inspire the structure and their functioning. The brain consists of billions of interconnected neurons, which ANNs try to mimic. ANNs have multiple layers with simple processing units known as nodes where each of them are connected by edges with weights [51].

Recently, there has been an increasing interest in neural network architectures consisting of many layers. With the availability of a large volume of data and powerful hardware for computation, such model architectures were able to outperform humans in several cognitive tasks [119][104]. This led to the creation of a sub-field of ML known as DL [81].

The most basic version of an ANN model is a feed-forward neural network, where there exist other architectures such as RNNs, CNNs, etc., which are explained in detail in the sections below.

2.2.1 Introduction to Neural Networks

In order to understand the computational model of artificial neural networks, one needs to begin from its building block, known as the perceptron [117]. Inspired from the brain's neurons, a perceptron is a simple computational model that takes in one or more inputs and provides a single value as output. This is illustrated in Figure2.1. Based on the pre-defined threshold and its output, the perceptron acts as a binary classifier, i.e., if the output value is greater than the threshold, the input is assigned to class 1, else it is assigned to class 0.

Let the inputs to the perceptron model be *x1, x2, x3* and the series of model weights corresponding to each input variable be *w1, w2, w3*. This simple model consists of two operations:

- First, each input with its weights are multiplied, which is followed by summation, to which we also add the bias term *'b'*, so that the model has flexibility for location shift.

- Next, a class label (0 or 1) is assigned based on a binary activation function.

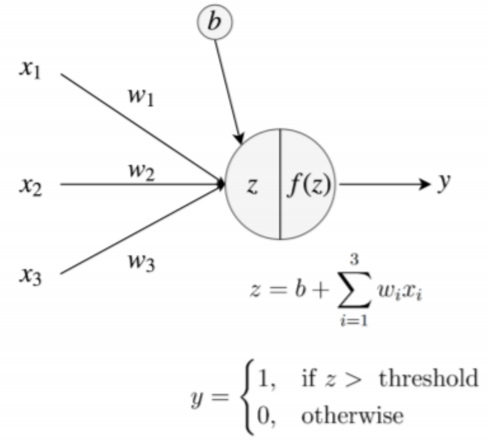

b
x_1
w_1
x_2
w_2
x_3
w_3
z
f(z)
y
$z = b + \sum_{i=1}^{3} w_i x_i$
$y = \begin{cases} 1, & \text{if } z > \text{ threshold} \\ 0, & \text{otherwise} \end{cases}$

Figure 2.1: Perceptron Model

The predicted value of output y corresponding to the given set of inputs, and in order for the predicted output to be close to the desired output (ground truth), we would need to make adjustments to the weights w1, w2, w3 and bias term b.

However, with the recent advancements, modern neural networks do not use the simple perceptron anymore. Instead, they consist of computational units known as neurons (or nodes), which replace the simple binary activation function with non-linear functions, combining multiple layers of neurons to form a more robust model known as the feed-forward neural network is possible. Each neuron is connected to every other neuron in the previous and subsequent layers. However, there are no connections between neurons within the same layer.

As illustrated in Figure 2.2, there can be multiple inputs and multiple outputs where they are connected via several hidden layers. The input at each neuron gets transformed by weighted summation followed by non-linear activation. The computation starts from the input layer until the output layer and is known as forward propagation. Feed-forward neural networks are often used for transforming the dimension of inputs and outputs of different complex models like CNN and RNN. They can learn non-linear representations of the data and have been successfully applied to many classification and regression tasks.

The non-linear activation functions can be different in each layer of the network. However, while all the hidden layers have a similar activation function, the final layer can have

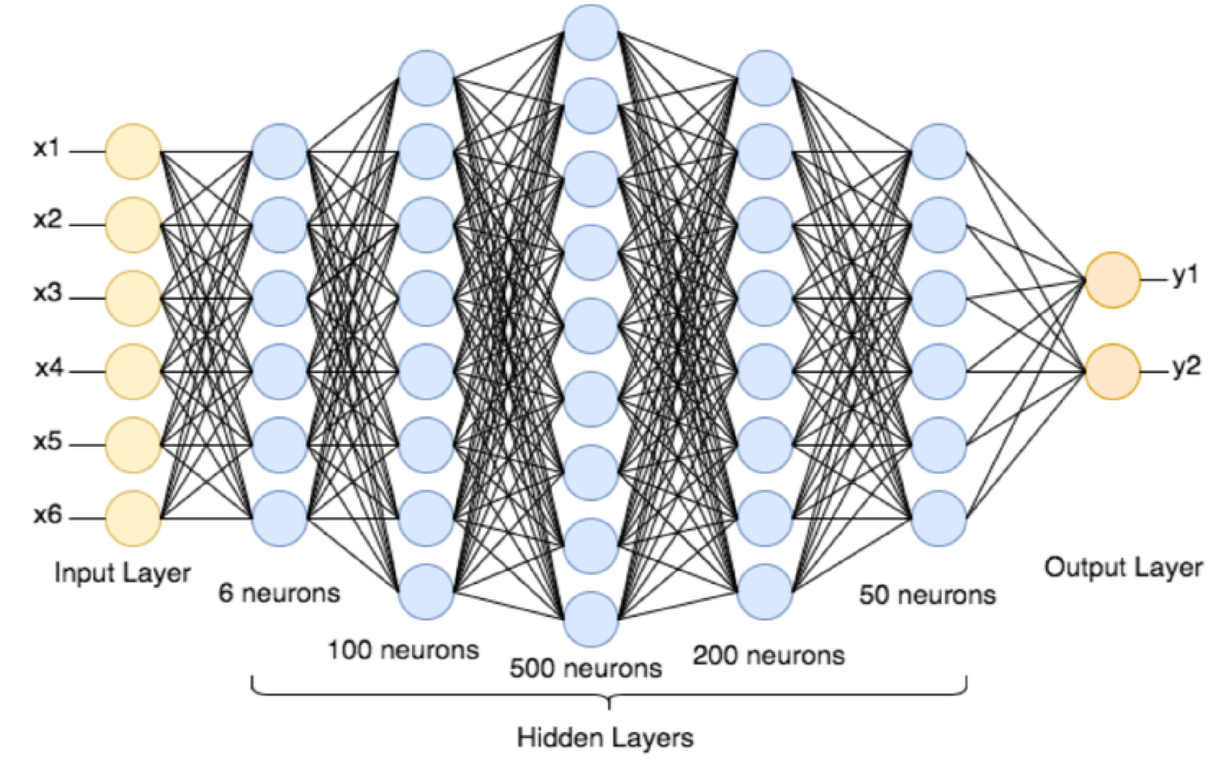


Figure 2.2: Feed Forward Neural Model [7]

a different one. Thus, there are many choices of activation functions, and some of them are listed below:

- **ReLU:** ReLU is known as rectified linear unit, is a popular activation function, which is known for its simple computations. A ReLU function simply returns the input if the input is bigger than one, and zero otherwise:

$$f(x) = \begin{cases} x & \text{if } x{>}0 \\ ax & \text{otherwise} \end{cases} \tag{2.1}$$

  The derivatives of ReLU activation function are computed with significantly less computational power, where it is 1 for values above 0 and 0 otherwise.

- **Sigmoid:** A sigmoid activation function has an S-shaped curve, and it clips the value of input between 0 and 1.

$$f(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1}$$

- **Softmax:** Softmax activation function is mainly used when the output of a neural network is a vector, and the goal is to pick the most probable component. This function amplifies the maximum value in a vector or an array and lessens or dampens the rest of the components' values.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_i}} \text{for } i = i, ...K \text{and } z = (z_1, ...z_k) \in R^K$$

- **Tanh:** Tanh activation function is also known as hyperbolic tangent Activation Function, which is similar to sigmoid activation function, but ranging from -1 to +1.

$$\tanh x = \frac{\sinh x}{\cosh x} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

With no structured methodology to adjust the model weights, other than by trial and error, there has been significant research [135, 118], which contributed to the development of the method known as backpropagation of errors, which made it possible to estimate the weights in an ANN model. Backpropagation computes the gradient of the network to individual connection weights, given a loss function, such as the mean squared error (MSE) loss.

2.2.2 Recurrent Neural Networks

Recurrent Neural Networks (RNN) is a type of neural network model which can handle variable-length data. Each RNN unit has a self-loop and an internal state which helps process sequential data like audio waveform, stock value, text and video. RNNs help process sequential data because they can utilize each data point as well as the relation of that data point with the preceding data points, which results in generating a more comprehensive and proper textual representation. As a result of their usefulness, RNNs are used in NLP and CV for various tasks [139].

Figure 2.3 shows an RNN network, where x_i denotes the input to an RNN which is usually the feature vector and h_i denotes the hidden state carried forward and outputted

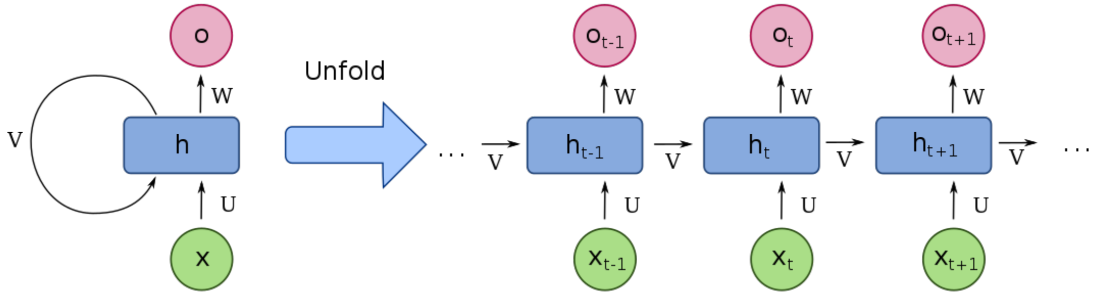


Figure 2.3: An un-rolled Recurrent Neural Network [1]

at each time-step by the RNN. This figure denotes the simplest form of RNN. In practice we often use a LSTM [56] or GRU [36] network. Both these networks are complex and perform much better at language tasks. They consist of internal states and gates combined with non-linearities to allow better learning of complex functions and mappings.

2.2.3 Long Short-Term Memory

As described in the previous section, RNNs learn representations of the data over temporal sequences, like text or audio features. However, in practice, RNNs do not seem to assign priorities to which of the past data they choose to ascribe higher importance, which is detrimental to their usage in tasks that require the processing of long sequences of data like in text, audio or video processing. This lack of ability to learn long-term dependencies in the input features is shown in previous work by [8]. LSTM seeking to address this issue by explicitly modeling how much to retain and forget at each time-step during the RNNs training procedure.

The structure of an LSTM is depicted in Figure 2.4. An LSTM unit is comprised of three distinct gating layers internally - namely, the forget gate, input gate and output gate - that determine its outputs: the new cell state and hidden state. The three gates are enclosed within a "cell" that remembers relevant information over different time intervals and regulates the flow of information into and out of the cell. The gating mechanism helps the LSTMs to capture distant temporal dependencies.

As can be seen in Figure 2.4, the cell state C_t passes through the LSTM mostly unperturbed, which is intended to address the problem of vanishing gradients. The forget gate

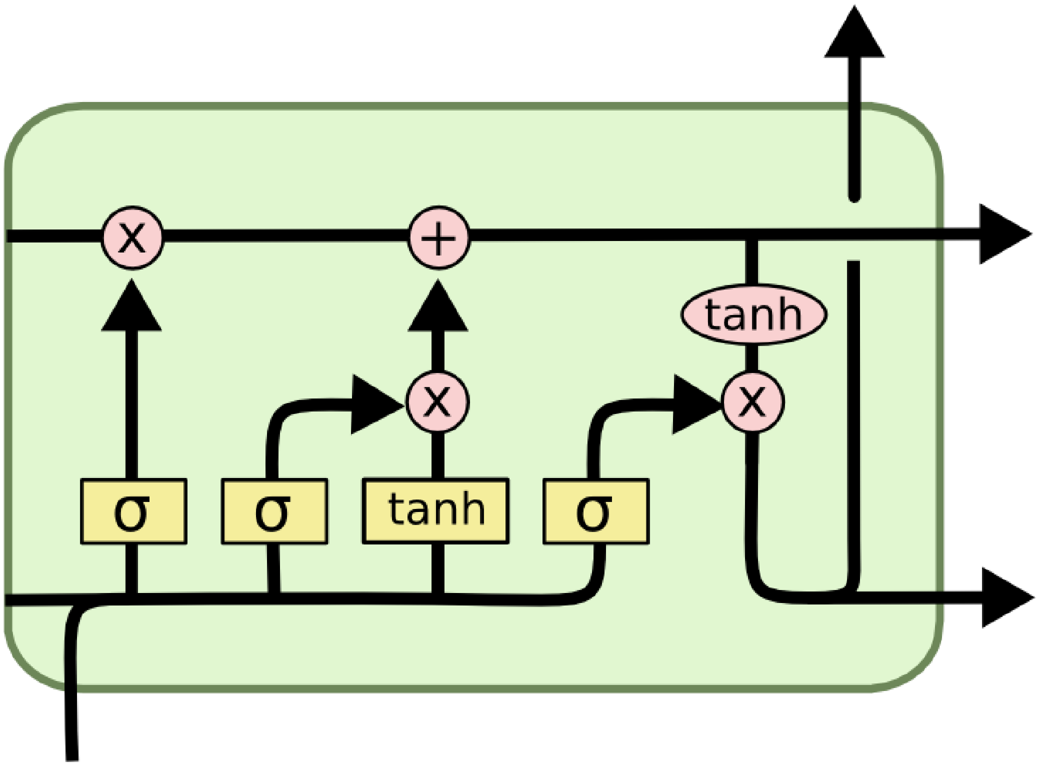


Figure 2.4: Long Short-Term Memory Unit [106]

uses a sigmoid activation to squash the output values of the previous time step between 0 and 1, which can be understood as being semantically equivalent to deciding how much of the previous cell state to forget.

$$f_{\text{t}} = \sigma(W_{\text{f}} \cdot [h_{\text{t-1}}, x_{\text{t}}] + b_{\text{f}})$$

The input gate also uses a sigmoid activation on the previous hidden state, which is multiplied by the candidate cell state $\hat{\text{C}}$. The value thus obtained is then multiplied by the forget-gated previous cell state to form the next cell state.

$$i_{\text{t}} = \sigma(W_{\text{i}} \cdot [h_{\text{t-1}}, x_{\text{t}}] + b_{\text{c}})$$

$$\hat{C}_{\text{t}} = \tanh(W_{\text{c}} \cdot [h_{\text{t-1}}, x_{\text{t}}] + b_{\text{c}})$$

$$C_{\text{t}} = f_{\text{t}} \cdot C_{\text{t-1}} + i_{\text{t}} \cdot \hat{C}_{\text{t}}$$

Now that we have obtained the cell state, we propagate that through to the next time step. To obtain the hidden state, we use an output gate, again with a sigmoid activation, to gate the cell-state and create the hidden state for the next time step.

$$o_{\text{t}} = \sigma(W_{\text{o}} \cdot [h_{\text{t-1}}, x_{\text{t}}] + b_{\text{o}})$$
$$o_{\text{t}} = \tanh(C_{\text{t}})$$

LSTMs outperform simple recurrent neural networks in learning long-term dependencies, making them suitable for various natural language processing tasks. LSTMs have also been combined with other types of neural networks, such as CNNs, to improve automatic image captioning [90].

2.3 Regularization

Regularization refers the technique used in machine learning models to improve the model's generalization capabilities. A few of the techniques are early stopping, dropout, parameter norm penalties, etc.

2.3.1 Dropout

One of the most common problems faced by DNN architectures is overfitting, which refers to the problem of a model learning functions that represent the training set very well but fail to generalize to the validation/test set. Dropout is used as a regularization technique to help mitigate the problem of overfitting [127].

Using dropout, neurons and their connections are randomly dropped during training, where each neuron can be dropped with a fixed probability that is a tunable hyperparameter. Using dropout in DNN helps the model prevent overfitting by limiting the development

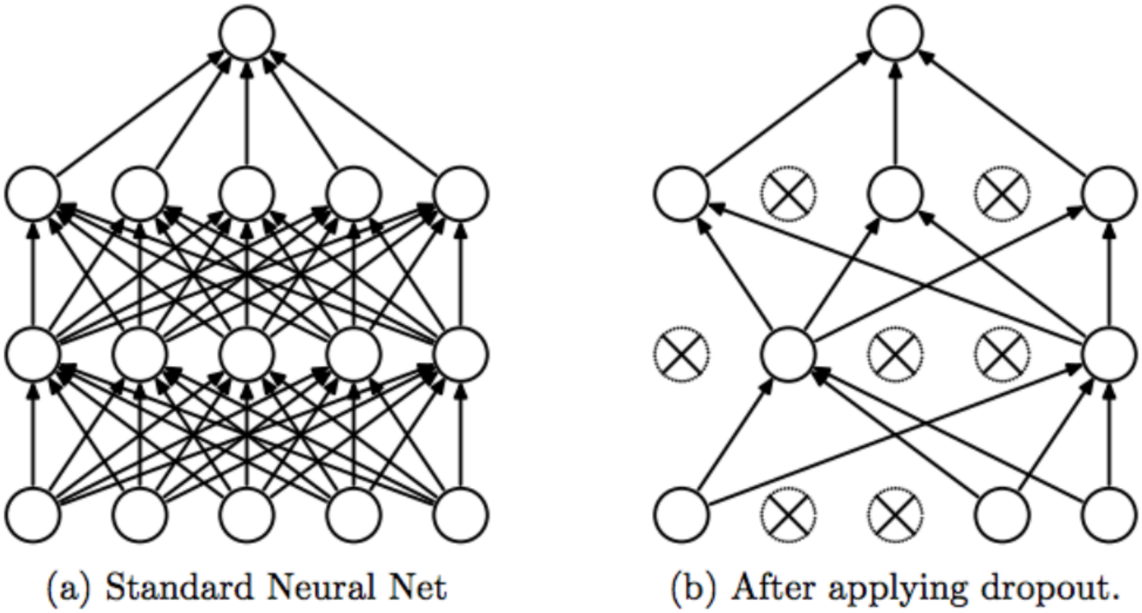


Figure 2.5: Illustration of Dropout [127]

of excessive codependency between units during training. We should remember that during the testing/inference step, no neurons are to be dropped. Figure 2.5 illustrates the idea behind dropout.

2.3.2 Batch Normalization

The batch normalization [64] technique is used to improve the training speed and stability of DNN, which works by reducing internal covariate shift. Internal covariate shift is defined as the change in the distribution of the network unit outputs due to the change in the network parameter values, and the reduction is obtained through batch normalization transformation of a layer's input defined as:

$$y = \frac{x - E[x]}{\sqrt{Var[x]}} \cdot \gamma + \beta$$

where x is the layer inputs and γ, β are learnable parameter.

2.4 Sequence to Sequence Models

Sequence-to-sequence (Seq2Seq) [128] models are mostly made up of an encoder and a decoder, which is predominantly used in Natural Language Processing (NLP) and is mainly used to convert a random input sequence to an output sequence. The sequence can range from whole documents to individual words. Both the encoder and the decoder are usually implemented with RNNs (usually an LSTM or GRU), where the encoder learns an intermediate representation of the input sequence, which the decoder will use to generate the desired output sequence. An overview of a seq2seq network used for translation is shown in Figure 2.6.

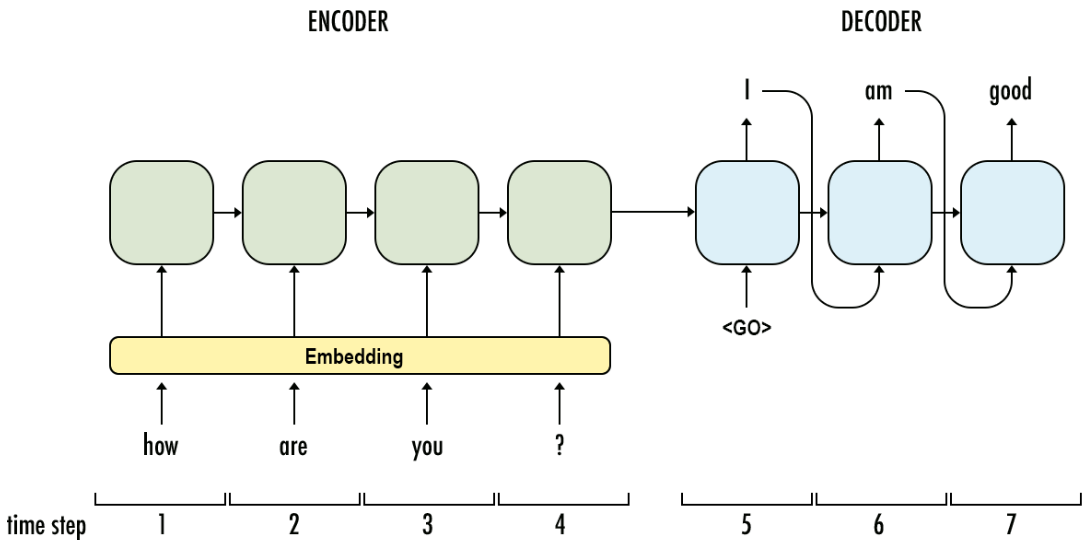


Figure 2.6: Sequence-to-sequence Model [19]

2.5 Attention Mechanism

Attention is a technique that imitates cognitive attention. Likewise, attention mechanism is used in neural networks to help the training process by indicating to the model what part of the inputs or features are to be focused upon. Here we will discuss the attention mechanism used in Natural Language Processing and Computer Vision.

In NLP, two popular attention mechanisms used in Seq2Seq models are Luong Attention [94] and Bahdanau Attention [6], which were introduced for machine translation, which outperformed vanilla Seq2Seq models. This attention is obtained by aligning tokens on the target side to the tokens on the source side. The attention mentioned in the above mechanisms used in Seq2Seq models differ in how the context vector is computed where Luong Attention takes a multiplicative form. In contrast, the Bahdanau Attention takes an additive form.

On the other hand, the attention mechanism was found to be helpful in CV especially in image captioning [137] and object detection [13]. Using the whole image in any task, for instance, in a classification task, will lead to significant noise, thus lowering detection accuracy. Also, in medical images, where the medical images are made up of thousands of pixels, identifying the lesion area, which may be small, will be difficult, given the entire image [49]. Recursive hard attention[49] is used by cropping out the discriminative parts of the image and classifying both the global image as well as the cropped portion together. While attention reweighs certain network features, soft attention allows these weights to be continuous while hard attention requires them to be binary.

2.6 Convolution Neural Network

CNNs are a class of ANNs most popularly used for visual analysis, and in simple terms, CNNs are regularized multilayer perceptron (MLP). CNNs use the convolution operation in at least one of their layers instead of general matrix multiplication. While multilayer perceptron (MLP) is prone to over-fitting due to their fully connected nature, CNNs use small and simple patterns to learn more extensive and more complex patterns in data, resulting in less connections and lower complexity.

An input layer, output layer and hidden layer are the three types of layers in CNNs. The hidden layers consist of a series of convolution operations, which convolve with an operation such as multiplication or dot product. These layers are commonly followed by additional convolutional layers, such as pooling and fully connected layers. The pooling layers reduce data dimension by reducing outputs from a group of data points to a single value. Different pooling operations result in different types of features. Global pooling [23], max-pooling [24], and average pooling [103] are the most common pooling operation employed. Figure 2.7 shows an overview of a simple CNN. Even though CNNs was primarily introduced in the computer vision community, they have been extensively explored for many NLP tasks [26].

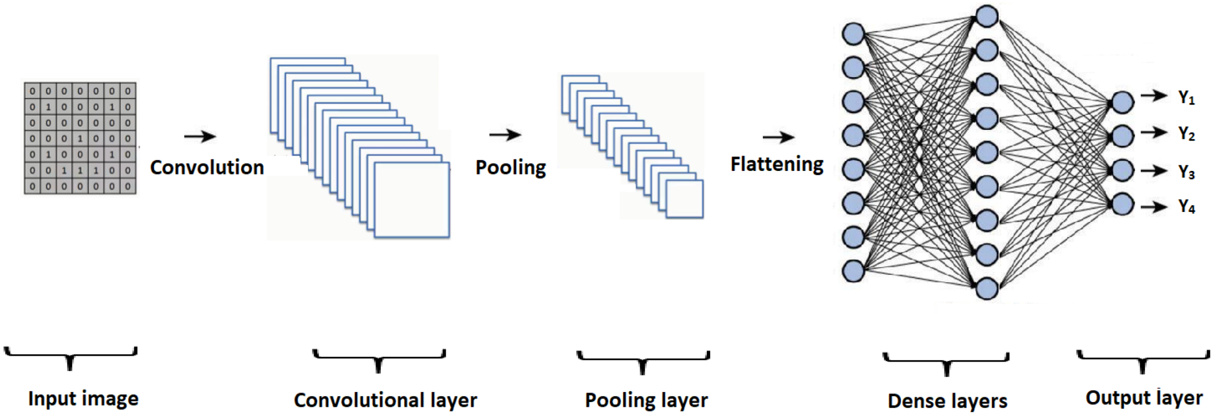


Figure 2.7: Convolutional Neural Network Architecture [97]

2.7 Embeddings

In ML, embeddings mean the mapping of discrete variables to a vector in continuous space. The vectors partially represent the semantics of the raw input data, which otherwise may not be easily comprehensible to a neural network. For example, we cannot directly train a neural network for image classification with the image pixels without transforming it into embedding space. We discuss embeddings for inputs from words and images in the following subsections.

2.7.1 Word Embeddings

In NLP, word embeddings are the vector representation of all the words in the vocabulary. Techniques used to learn word embeddings are unsupervised, though supervised and semi-supervised techniques have also been proposed. The embeddings are learned so that words appearing in similar contexts are close to each other. For example, words like "grapes" and "banana" will appear close to each other as they belong to the family of fruits. Figure 2.8 shows word embedding space. Various techniques like Word2vec [101], Glove [109], ELMo [110], FastText [70], Skip-thoughts [76], Quick-thoughts [91], InferNet [27], and Google's universal sentence encoder [18].

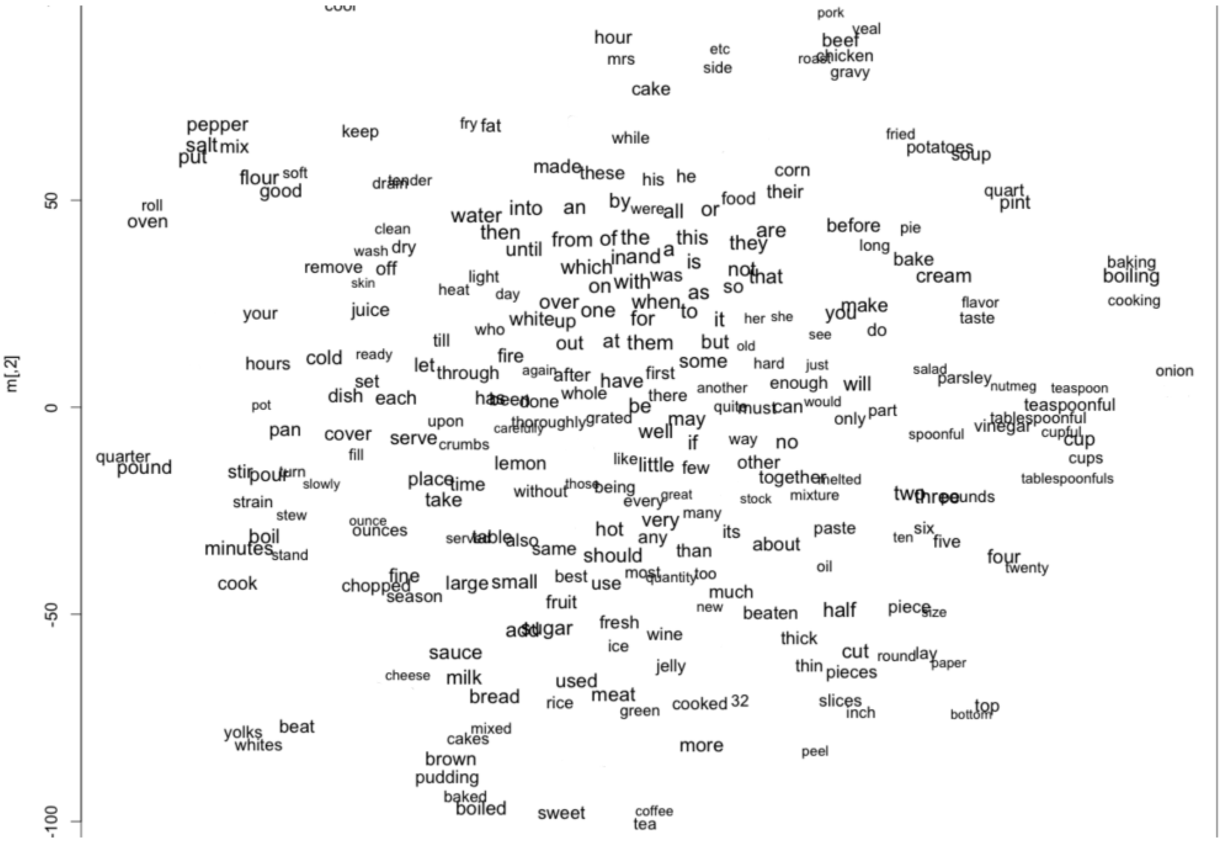


Figure 2.8: Word Embeddings [86]

2.7.2 Visual Embeddings

The main goal for pre-trained embeddings for visual modality is to learn a meaningful representation of images, though the methods used to learn those embeddings are not completely supervised. Pre-trained visual embeddings are nothing but the feature vectors extracted from the hidden layers of a neural network trained on any task. For example, a few popular choices of obtaining pre-trained image embeddings are Inception [130], VGG [123], SqueezeNet [57], and DeepLoc [79], are all trained on an image classification task.

Learning a joint representation of visual and other modalities like speech and word embeddings is still an active field of research.

Chapter 3

Related Work

In recent years, there has been ongoing research in applying ML methods to medical images and texts. Medical images pose unique challenges due to their significant variations, rich structures, and large dimensionality. On the other hand, electronic health records are challenging as they are highly unstructured, with many medical terminologies present in the text. As a result, researchers have begun to look into various image and text analysis techniques, and their applications in digital pathology [50].

The first section in this chapter briefs on a few of the past works done on the medical images. In contrast, the second section discusses the previous NLP techniques used in medical text data.

3.1 Medical Image Analysis

Recently, DL based approaches were performing better than conventional machine learning methods in image analysis tasks, automating end-to-end processing [21, 58, 60]. The main goal of histopathology is to differentiate between normal tissue, non-malignant and malignant lesions and to perform a prognostic evaluation [40]. Considering medical imaging, CNNs have been successful in diabetic retinopathy screening [113], bone disease prediction [132] and age assessment [59], and other problems [21]. DNN models achieved massive success in achieving state-of-the-art resulting in histopathology image analysis from disease grading, cancer classification to outcome prediction [126, 10, 87].

Lately, self-supervised and semi-supervised approaches are becoming increasingly popular to reduce the annotation burden by leveraging the readily available unlabeled data that can be trained with limited supervision. Promising results for the above methods in CV is seen in [69, 80, 124] and medical image analysis tasks [20, 131, 84].

Initially, most of the DL training on WSIs did not use the whole image as input; instead, they divided the WSIs into multiple patches and used those patches for training the model [66, 39, 17, 4]. Since the patch size is tiny compared to the complete WSI, there are large amounts of background regions that are useless for diagnosis. Hence it is essential to remove the background regions. However, this problem is often treated as a trivial part of the research, mostly solved via threshold-based methods. In [68] a threshold was used on the pixel intensity values to detect the tissue. As another example, [78] removed blank regions by setting a threshold on saturation and intensity of pixels. Other works used homogeneity criteria to only select patches containing a considerable part of the tissue [41]. Apart from threshold-based methods, [115] discusses different U-Net, a light weighted CNN architectures to segment only the tissue region from the background.

Concerning the pre-processing in medical images, one of the main steps followed in WSIs is color normalization of stained tissue samples [39]. Despite the standardized staining protocols, versions in the staining outcomes are prevalent due to variations within the staining parameters, e.g., Antigen awareness and incubation time and temperature, exceptional situations among slide scanners, etc. [140]. Such color versions can adversely affect the performance and accuracy of the classifier. Therefore, in these papers [65, 74, 85, 95] stain normalization techniques have been proposed for better performance of the model. One of the most frequent ways to remove the stains from the patches is to transform the color appearance of a source image to the target image [133] and [114] presents a method to transform the color appearance of a source image to the target image.

An efficient classification model was built [77] where the input size equals the patch size and inception-v3 network is used as the classifier and is trained them from scratch with an input size of 512 x 512 to detect and discriminate epithelial tumors in WSIs of stomach and colon and achieved an accuracy of 95.6%. On the other hand, [5] have used a slightly different CNN model, which was architecturally and conceptually inspired by AlexNet and VGG16 on a new dataset named “Kimia Path24” and achieved an accuracy of 44.80%. However, [75] has used the same “Kimia Path24” dataset and has compared the performance of DenseNet-161 and ResNet-50 pre-trained CNN models. The results proved that ResNet-50 obtained a validation accuracy of 97.77% which was better than the 95.79% achieved by DenseNet-161.

Once after classifying the WSIs, it is essential to explain the model’s prediction to justify their reliability, and more importance is given to the model’s interpretability when it comes to the medical domain. The paper [93] defines a suitable measure of feature importance in the SHAP (SHapley Additive exPlanations) framework. LIME [116] explains the choice by highlighting or segmenting the critical region in the particular patch on which the classifier made the decision, thus helping the practitioners to validate the model. In the paper, [138] heatmap is drawn based on confidence scores of each patch, based on which the interpretability of the model is explained.

In this thesis, I present an approach for histopathology microscopy image analysis for cancer type classification. Our approach is by using a weakly supervised method using instance-level clustering and attention mechanism.

3.2 NLP on Medical Textual Data

In the field of biomedical research, information extraction using NLP spans from rule-based systems [73] down to domain-specific systems using feature-based classification [134], and to the recent deep networks for end-to-end feature extraction and classification [46].

Several studies performed for information extraction techniques from pathology reports belonging to various types of tumor related to lungs, breast, colorectal, prostate, where [125] mentioned the overview of various tools developed. [25] performed an experiment to automatically extract diseases from pathology reports belonging to colon cancer. Rule-based systems in combination with machine learning methods were used to extract nine different classes from the reports.

While [107] extracted 28 different concepts from pathology reports for primary cutaneous melanoma (skin cancer), [99] classified colorectal cancer according to the TNM (Tumor, Node and Metastases) scale using Naïve-Bayes and Support Vector Machines. [100] applied rule-based methods to pathology reports for lung cancer, where the results obtained from the experiments gave F1-Score from 0.7 to 0.9.

[31] on the other hand, used rules to extract concepts in 5,826 breast cancer and 2,838 prostate cancer pathology reports. The experiment extracted around 80 fields and obtained 90-95 percent accuracy, and domain experts evaluated.

Two studies have applied rule-based methods to Norwegian pathology reports on the reports belonging to a language other than English. [32] extracted values for nine concepts from 25 pathology reports describing prostate biopsies, where they obtained F-scores ranging from 0.24 to 0.94. On the other hand, [134] extracted values for ten concepts related to breast cancer with an F-score ranging from 0.67 to 1.0 using 40 reports. Both studies were done on small data sets, but the results show that rule-based methods are promising for information extraction from pathology reports written in Norwegian.

In case of classification tasks or retrieving specific features from reports, successful studies in NLP for understanding pathology reports have been reported [63]. For example, The Cancer Text Information Extraction System(caTIES) is a framework developed in a caBIG project [30] that focuses on the extraction of crucial details from SPR to achieve high precision and recall. On the other hand, a system named Open Registry [28] was able to filter out the pathology reports having cancer specified in them, based on the disease codes.

In 2010, the Automated Retrieval Console(ARC) [33] was introduced, where machine learning models are used to predict the degree of association of a given radiology or pathology report to cancer. The performance of this approach varied from an F-measure of 0.75 for lung cancer to 0.94 for colon cancer. However, this approach utilized domain-specific rules, which may be disadvantageous when working with a wide variety of pathology reports. Other works have performed a classification of the pathology reports by extracting the TF-IDF features [71]. The extracted features were given as input to XGBoost, SVM and Logistic Regression, where improved ensemble results were obtained with the XGBoost classifier.

In addition, many algorithms that convert words to fixed-dimensional vectors that can be used to preserve syntactic and semantic relationships in a text corpus were introduced. These include word2vec [102], and GloVe [109] which use co-occurrences of words in the text and produce dense vectors such that words appearing in similar contexts have similar word embeddings. Significant improvements that could be achieved in the NLP field came when unsupervised model architectures were proposed to represent words as a fixed dimensional dense vector [102], [101]. The architectures were Continuous Bag of Words(CBOW) that predicts the target word given the context and the Skip-gram model, which predicts the context based on the target word. Another significant advance was BERT [35], which improves fine-tuning-based approaches by taking into account both left and the right context, unlike the traditional algorithms, which used the left-to-right direction only.

Chapter 4

Dataset

This chapter briefs the dataset used for our experiments and the tool used to extract the histopathology image and textual data.

4.1 Dataset from The Cancer Genome Atlas (TCGA)

TCGA is a cancer genomics program, having around 20,000 primary cancer and matched normal samples over 33 cancer types. TCGA is a collaboration between the NCI and NHGRI, which aims to generate comprehensive, multi-dimensional maps of the fundamental genomic changes in major types and subtypes of cancer. TCGA has analyzed matched tumor and normal tissues from 11,000 patients, allowing for the comprehensive characterization of 33 cancer types and sub-types, including ten rare cancers and the TCGA data, are hosted at the NCI GDC [121]. In addition, TCGA has the tissue slides and the report for different types of cancer for each organ and the tool we used to extract the data is described in the section below.

4.2 GDC Data Transfer Tool for Querying Dataset

The GDC Data Model is the primary method of organizing all data within the GDC, which can be thought of as a Directed Acyclic Graph composed of interconnected entities. An entity in the GDC is a unique component of the GDC Data Model where each entity has associated some attributes that can be used to describe it. GDC Data Portal allows the user to filter available data files according to different criteria. The user can download files individually or add them to the cart while the search is underway. The GDC Data Transfer Tool is a standalone client application that runs on the user's machine, and it helps download large volumes of data [121]. Due to the high volume of images to be downloaded, GDC Data Transfer Tool was used, and a manifest file consisting of all the files to be downloaded was provided. A sample pathology image from the dataset can be seen in Figure 4.1.

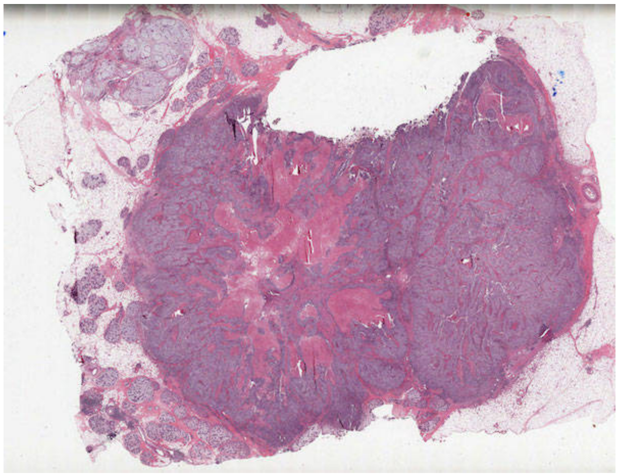


Figure 4.1: Sample Whole Slide Image from the Histopathology Image Dataset

Among the records, we picked a total of 1949 data points belonging to four primary sites: kidney, lungs, thymus, and testis with 937, 749, 124 and 139 records. Further, seven different diseases types belonging to the above organs are chosen which are namely, “Kidney Renal Papillary Cell Carcinoma”, “Kidney Renal Clear Cell Carcinoma”, “Lung Adenocarcinoma”, “Lung Squamous Cell Carcinoma”, “Testicular Germ Cell Tumors”, “Kidney Chromophobe” and “Thymoma”. Amongst the medical reports, the main criteria for choosing the reports are the quality of the PDF reports. Therefore, only those reports that were quite readable were considered, and the rest were discarded. The number of samples in each class used for our experiments based on primary site and disease type are as shown in the Tables 4.1 and 4.2.

Primary Site	**Number of samples**
kidney	937
lungs	749
Testis	139
Thymus	124

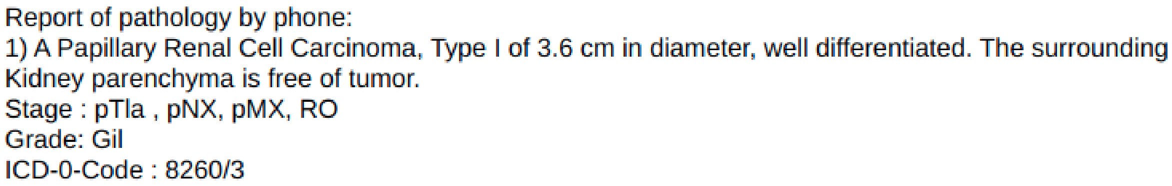
Table 4.1: Total Number of Records based on Primary Site

Disease Type	**Number of samples**
Kidney Renal Clear Cell Carcinoma	537
Lung Adenocarcinoma	372
Lung Squamous Cell Carcinoma	356
Kidney Renal Papillary Cell Carcinoma	283
Testicular Germ Cell Tumors	139
Thymoma	124
Kidney Chromophobe	110

Table 4.2: Total Number of Records based on Disease Type

4.3 Further Extraction of Pathology Reports

Once after selecting 1,949 reports, the next step was to extract the text from those reports. As it is not practical to type the relevant and valid information from those reports manually, which will be extensively time-consuming, OCR was implemented. This software converted all the PDF reports to text files, which we used for the experiments. Once after converting the image data of the reports to text files, manual inspection of spelling errors and grammar was inspected on those files. Also, irrelevant characters produced as an artifact by the OCR system were removed. These cleaned pathology reports are further used for experiments, and an example of a pathology report from the dataset is as shown in Figure 4.2.



Report of pathology by phone:
1) A Papillary Renal Cell Carcinoma, Type I of 3.6 cm in diameter, well differentiated. The surrounding Kidney parenchyma is free of tumor.
Stage : pTla , pNX, pMX, RO
Grade: Gil
ICD-0-Code : 8260/3

Figure 4.2: Sample Pathology Report from the Histopathology Reports Dataset

Chapter 5

Transformer Models used for Pathology Reports Classification

Transformer-based PLMs is made of the combination of transformers with transfer learning, and SSL has been booming in the field of NLP from the past few years. With a great success of PLMs in the general domain, there was an urge felt in the medical and biomedical communities to develop various PLMs in these domains. PLMs can learn language representations that are useful across tasks, and we can directly use the pre-trained model without training the models from scratch. These PLMs are trained over a huge corpus of unlabelled textual data using a self-supervised learning mechanism, learning between supervised and unsupervised learning. In self-supervised learning, as the labels are not created manually, they are generated automatically based on the relationship between different sections of the input data. Once after the training of the transformer-based PLMs over huge corpora of texts of a specific domain, the models can be further used for multiple downstream tasks just by fine-tuning by adding layers based on the task [35, 72].

Recently, there has been an increasing demand for text mining in the clinical domain, with a considerable increase in the number of biomedical documents produced with essential details and information in them. This lead to the generation of lots several PLMs each consisting of millions or even billions of parameters. In this section, we describe several contextualized word embedding models trained on biomedical and clinical corpora like BioBERT, Clinical BioBERT and BioMed-RoBERTa along with TF-IDF and their techniques to convert text into vectors.

5.1 BioBERT

BERT was a state-of-the-art model in NLP, which was proposed by Google researchers, which takes into consideration of the bi-directional context of the text in all the layers of its architecture [35]. These representations are useful for sequential data that depends heavily on the context in a text. The introduction of transfer learning in this field aids in carrying encoded information over to strengthen an individual's smaller tasks across domains. In transfer learning, we refer to this step as "fine-tuning," and it means that the pre-trained model is now being fine-tuned for the specific task at hand. BERT is originally an English-language model which was pre-trained using 2.5 billion words from Wikipedia and 0.8 billion words from BooksCorpus corpora. The architecture of BERT is as shown in the figure 5.1.

BioBERT[82] is an application of the BERT-based model [35], which is popularly used in the biomedical field. This model is obtained upon pre-training the BERT-base model on the biomedical corpus. We have used BioBERT-v1.1 for our experiments, which was obtained by pre-training BioBERT on PubMed Abstracts with 4.5 billion words and PMC Full-text articles having 13.5 billion words for 1M steps. The vocabulary size of the model is 28,996, each having 768 features. The output text from the data pre-processing step is tokenized using the BioBERT tokenizer. This tokenizer uses WordPiece Tokenization [136] to mitigate out of the vocabulary (OOV) problem that breaks down a word into multiple sub-words belonging to the BERT vocabulary. For example, the WordPiece tokenization of the word "penicillin", which will not be present in the vocabulary directly, is split into the sub-words, "pen", "##i" and "##cillin", which are available in the BERT vocabulary. The tokenized words are then fed to the classifier model for classification. Figure 5.1 depicts the overview of pre-training and fine-tuning of BioBERT.

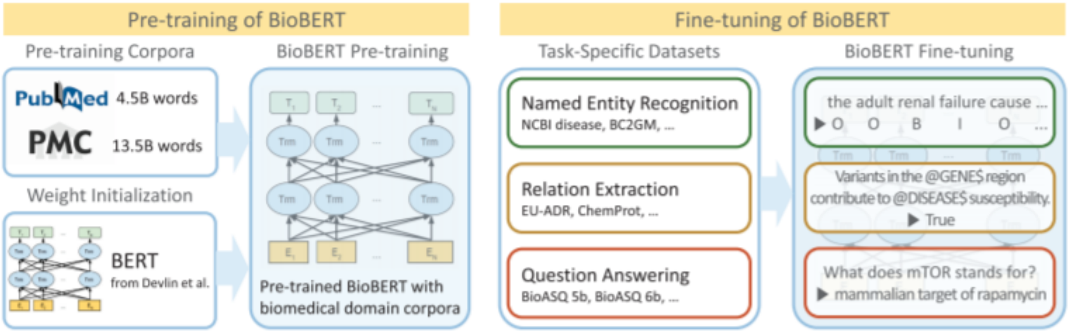


Figure 5.1: Overview of the pre-training and fine-tuning of BioBERT [82]

5.2 Clinical BioBERT

Clinical textual data such as physician notes have different linguistic features compared to non-biomedical or general texts. This difference encouraged the necessity for a specifically trained model for clinical domain texts, and thus Clinical BioBERT was introduced. Clinical BioBERT [2] is initialized from BioBERT model (BioBERT-Base vx1.0 + PubMed 200K + PMC 270K) and is trained on all the MIMIC-III notes (880M words), which is a database containing health reports of the ICU admitted patients at the Beth Israel Hospital in Boston, MA. This data is used to pre-train the model for 150k steps with batch-size set to 32. Like BioBERT, the vocabulary size of Clinical BioBERT is also 28,996 tokens, each having 768 features following WordPiece Tokenization. The processed data is tokenized using the *"Bioclinical_BERT"* tokenizer, which is then used as input data to the classifier model.

5.3 BioMed-RoBERTa

BioMed-RoBERTa [88] is a recent model initialized from RoBERTa-base, is pre-trained for 12.5K steps with a batch size of 2,048 using 2.68M scientific papers (7.55B tokens) from Semantic Scholar [52, 3]. The vocabulary size of this model is 50,265 tokens with 768 features, which is acquired using BPE (byte pair encoding[120]) word pieces with \u0120 as the special signaling character. The *"biomed_roberta_base"* tokenizer from HuggingFace is used for tokenization.

5.4 Term Frequency-Inverse Document Frequency

TF–IDF stands for Term Frequency-Inverse Document Frequency is a mixture of two separate terms, namely Term Frequency (TF) and IDF. This metric says how important a word is to a document in a document set. Term Frequency (TF) is defined as the frequency of a term in a document [39]. Term Frequency (TF) assumes equal weightage to all the words, including the words which are commonly occurring, while IDF gives more importance to words that are frequently found in a set of documents. By multiplying the number of times, a word appears in a document (Term Frequency (TF)) and the number of times a word occurs in several documents (IDF), the statistical measure is used to evaluate words based on their value among the rest of the terms. Therefore each sentence should have representation according to the meaning of each word in the sentence. The calculation of TF–IDF for a word is as given below:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}}$$

$$idf(word) = \log(\frac{N}{df_t})$$

With the above two equations the score for every word can be obtained, which is represented as:

$$word_{i,j} = tf_{i,j} \times \log(\frac{N}{df_t})$$

where the $tf_{m,n}$is the number of times a word i appear in document j, df_i is the number of documents with the word i, and the total number of documents is represented by N.

Chapter 6

Analysis of Language Embeddings for Classification of Unstructured Pathology Reports

In order to achieve the best classification for the free-text pathology reports, various transformer base models are analyzed and the best medical transformer model for our DNN architecture is chosen. This chapter explains the data preprocessing steps performed on the pathology report texts, the experimental setup, evaluation metrics, and results.

6.1 Data Preprocessing

The main challenge in classification with DNN using text data is transforming the data into a clean format, which can be converted into numerical vectors. Therefore, before initializing the data preprocessing step, all samples consisting of empty documents were removed. Further, all bullet numberings, stop-words, and special-, numeric-, or null-characters are removed. Occurrences of spatial dimensions of tumor or organ size were also standardized by converting $l \times b \times h$ cm into a single entity with no spaces (i.e., as **lxbxhcm**). The preprocessing steps are summarized as in the Figure 6.1.

After data preprocessing, to estimate the model's performance on unseen data, we have performed the k-fold cross-validation with $k = 5$.

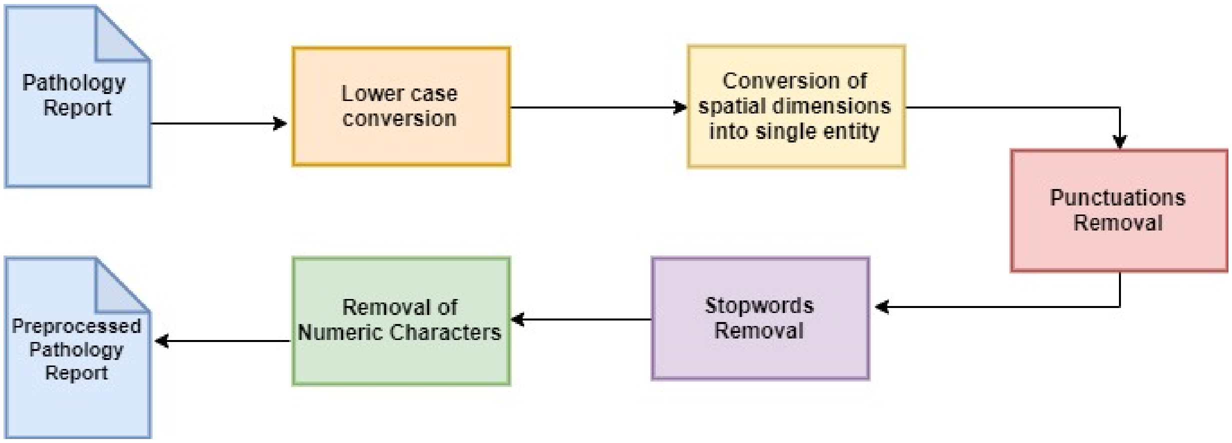

Pathology Report
Lower case conversion
Conversion of spatial dimensions into single entity
Punctuations Removal
Stopwords Removal
Removal of Numeric Characters
Preprocessed Pathology Report

Figure 6.1: Data Preprocessing steps

6.2 Experimental Setup

In this section, we will discuss the experimental setup of our analysis. Figure 6.2 depicts the DNN topology we have used. The main purpose of this network is to analyze the word embeddings as an initial investigation for NLP in Digital Pathology. The proposed network can be customized with one or two input layers based on the analysis. Firstly, the data is preprocessed and based on the maximum length of tokens amongst all the preprocessed reports; we chose 300 as the maximum length of each report. To understand the effectiveness transformer models, respective tokenizers of BioBERT, Clinical BioBERT and BioMed-RoBERTa are used to tokenize the data. In the case of TF–IDF, the preprocessed text is vectorized with maximum features of 300 and a minimum threshold value of 5. The tokenized data is then converted to an array of vectors which is given as the input to the DNN.

For the DNN having single input, the respective token embeddings from the pre-trained word embedding model or feature vectors from TF–IDF vectorization along with their labels are passed to the classifier for training. On the other hand, for the DNN having dual input, both the token embeddings from the language models and the feature vectors calculated using TF–IDF, along with their labels, are given as input to the model.

For analyzing the contextualized word embedding models, we have extracted the weights from the respective pre-trained model's word embedding layer. The weights are then converted into an embedding matrix, which is later initialized as weights to the embedding layer in our DNN classifier.

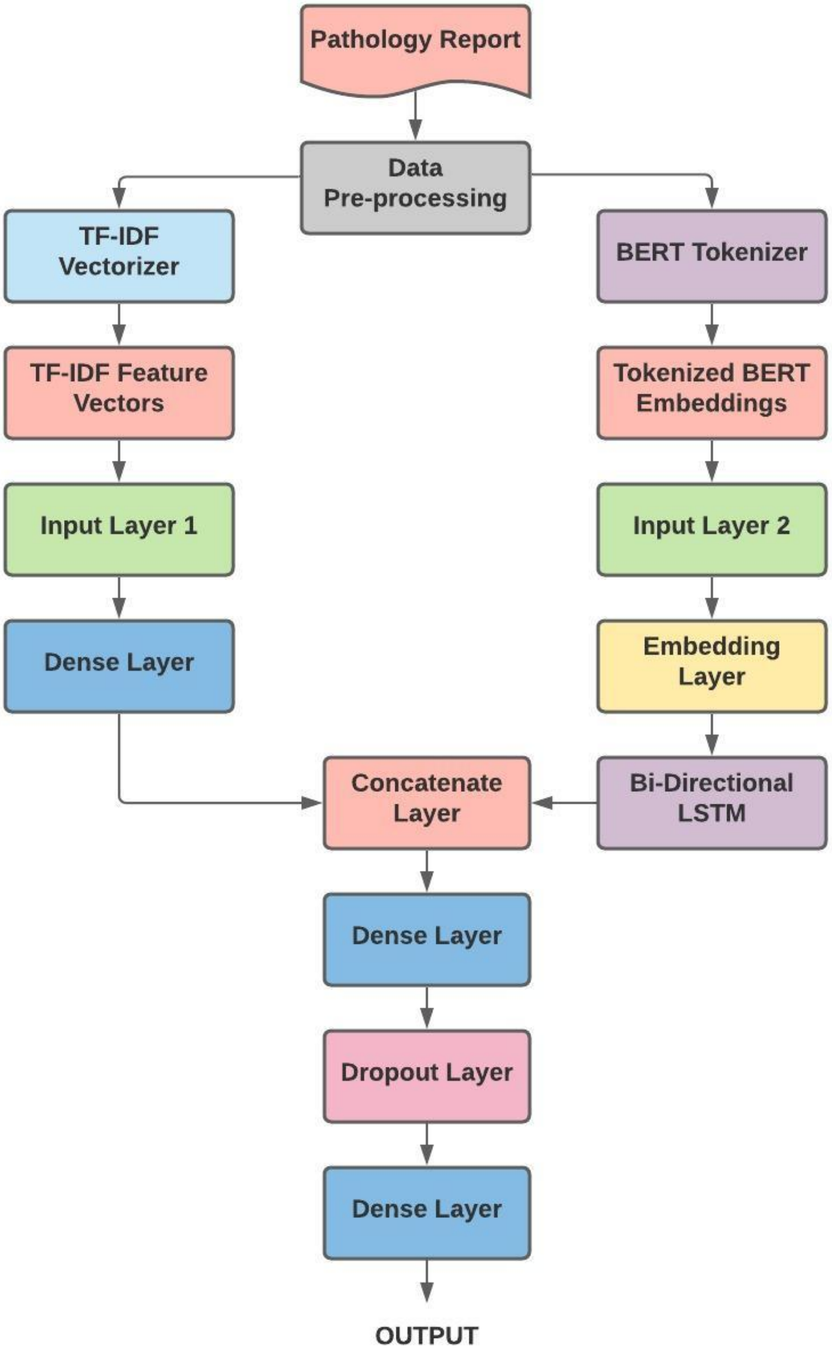


Figure 6.2: Deep Neural Network Topology

The word embeddings obtained from the pre-trained model tokenizer are passed through the Embedding Layer, followed by the Bidirectional LSTM layer. In parallel, the TF–IDF feature vectors are passed through the dense layer with "ReLU" activation function. The respective vectors from both the layers are concatenated and are then passed through the dense layer with "ReLU" activation function, followed by a dropout layer with the dropout rate of 0.3. The vectors are then finally sent to the dense output layer having the "softmax" activation function.

The model is trained using Adam optimizer, which uses the stochastic gradient descent method based on adaptive estimation of first-order and second-order moments. First, the optimizer is initialized with the default learning rate of 0.01. Regarding the loss function used for the DNN classifier, we used "categorical cross-entropy", a multi-class classification task. Then, the best vectorization method is decided based on the evaluation metrics such as precision, recall, F1-Score and classification accuracy, calculated as an average of all the five folds cross-validation. The results obtained are mentioned in Section 6.4.

6.3 Evaluation Metrics

To evaluate the performance of the classification model, precision, recall, F1-score and accuracy are used as evaluation metrics. Precision is defined as the percentage of pertinent texts to the number of matches of the human judgment outcomes. Recall is the ratio of the amount of relevant text and the text in the human judgemental system. F1-score, known as the harmonic mean of precision and recall, is one of the significant key evaluation metrics of the whole experiment. Finally, accuracy is defined as the ratio of correctly predicted observations to the total number of observations. The formulas for precision, recall, F1-score and accuracy are as follows-

$$Precision = \frac{\text{TP}}{\text{TP + FP}}$$

$$Accuracy = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$$

$$Recall = \frac{\text{TP}}{\text{TP+FN}}$$

$$F1score = \frac{\text{2*Precision*Recall}}{\text{Precision+Recall}} = \frac{\text{2*TP}}{\text{2*TP+FP+FN}}$$

Where TP is called true positive, TN is a true negative, FP is false positive and FN is a false negative.

Vectorization Method	**Precision**	**Recall**	**F1-Score**	**Accuracy**
BioBERT alone	0.76	0.78	0.76	79.76%
BioMed-RoBERTa alone	0.83	0.84	0.83	81.03%
Clinical BioBERT alone	0.80	0.81	0.81	81.29%
TF-IDF alone	0.81	0.78	0.79	88.90%
BioMed-RoBERTa plus TF-IDF	0.90	0.93	0.91	90.58%
BioBERT plus TF-IDF	0.88	0.92	0.90	91.65%
Clinical BioBERT plus TF-IDF	**0.91**	**0.92**	**0.91**	**93.77%**

Table 6.1: Evaluation of vectorization methods for the classification of disease types with a DNN classifier

6.4 Results

This section describes the quantitative results obtained by our experiments which show that our approach of combining contextualized word embeddings along with TF–IDF feature vectors provides best results than the model with a single input. The results of experiments using various vectorization techniques using DNN classifier for disease type classification on unseen data is shown in Table 6.1. Amongst the vectorization techniques used, *Clinical BioBERT embeddings in combination with TF-IDF feature vectors* yields the best accuracy of 93.77%, the precision of 0.91, F1-score of 0.91 and recall of 0.92. The next best were the combinations of *BioBert with TF-IDF* and *BioMed-RoBERTa with TF-IDF*. We believe this is because the Clinical BioBERT embeddings are generated from the language model that is initially pre-trained BioBERT which is trained on a medical corpus and then further trained on clinical texts with similar terminology, whose words are more likely to appear in pathology reports. Also, TF–IDF vectorization on these reports contributes to give important information about the word distributions in a pathology report. Thus the combination of them performed the best on our classification model. A tedious and error-prone task in this field is creating an embedding for cancer and tumor detection phrases from any given pathology report.

Retrieving the disease type is one of the most critical aspects of deciphering a pathology report, which will be very useful while combining content-based image retrieval with visual information. In addition, the accuracy obtained by these models supports the use of machine learning techniques to extract meaningful and relevant information from pathology reports.

Chapter 7

Histopathology Image Classification using Attention Mechanism and Instance Level Clustering

This chapter explains the idea behind the histopathology image classification architecture, preprocessing steps that include WSI segmentation, patching and feature extraction, experimental setup and training and finally, the results obtained.

7.1 Computational Hardware and Software

Due to the large memory requirements, the WSIs are stored in Compute Canada in the home space and project space file system. The initial segmentation and patching steps are performed using 2 x Intel E5-2683 v4 Broadwell @ 2.1GHz allocating three CPUs per task. For feature extraction from the patches obtained from the pre-trained neural network model of ResNet50, we used batch-wise parallelization using P100-PCIE GPUs to speed up the process. For processing the WSIs, openslide, OpenCV and pillow libraries are used, and for loading the data and training the DL models, PyTorch is used.

7.2 Histopathology Image Classification Architecture

The main idea behind the architecture is from [92] which built on the Multiple Instance Learning (MIL) framework and is a DL based weakly-supervised technique. The architecture considers each WSI as a bag that consists of hundreds of thousands of patches, known as instances. The model is built with two parts. First, the instance level clustering, which is used to refine the feature space and second, the attention mechanism, which calculates attention of all the patches, thus informing each patch importance in slide-level classification. Then, the slide-level representation is calculated as an average of all patches' attention scores. Then, each slide level representation is examined by the final classification layer. Finally, the probability score for each class is calculated using the "softmax" activation function, thus predicting the class of the whole slide image.

The Multiple Instance Learning (MIL) [62] algorithm was developed for binary-level classification, for example, positive and negative labels. This algorithm classifies a WSI to a positive label, even if one patch amongst the instances is classified as positive. Otherwise, if all the patches are classified into a negative label, the whole WSI is classified as negative. Despite having thousands of instances or patches, a standard MIL algorithm uses max pooling operation, thus using the gradient signal from the highest calculated patch to update the learning parameters of the model, which is a significant drawback. So, instead of max pooling, attention-based pooling is adopted in the model used. With the attention scores predicted by the network and the ground truth slide-level labels, pseudo labels are created to classify the patches into highly attended and weakly attended by the instance level classifier, thus helping in visualization.

7.3 Pre-processing of Whole Slide Images

Segmentation and Patching

Due to the large dimensions of WSIs, it is not possible to train the deep learning models directly. The experiments start by tissue segmentation of the WSIs into holes and tissue boundaries where each of the WSI is downsampled to a lower resolution. The downsampled images are then converted from RBG to HSV (Hue Saturation Value) scale. The edges are then smoothed out using median blurring. The binary mask for the tissue regions in the foreground is calculated, and morphological closing is done to fill up tiny holes and gaps in the image. Based on the threshold area, the contours on the detected objects identified in the foreground are filtered. Then images are stored for further downstream processing, keeping the segmentation mask on each of the WSI available for optional visual inspection. An example of image segmentation is shown in Figure 7.2. On the other hand, a text file is also generated, consisting of all the preprocessed files and corresponding edited files with the set of segmentation parameters used.

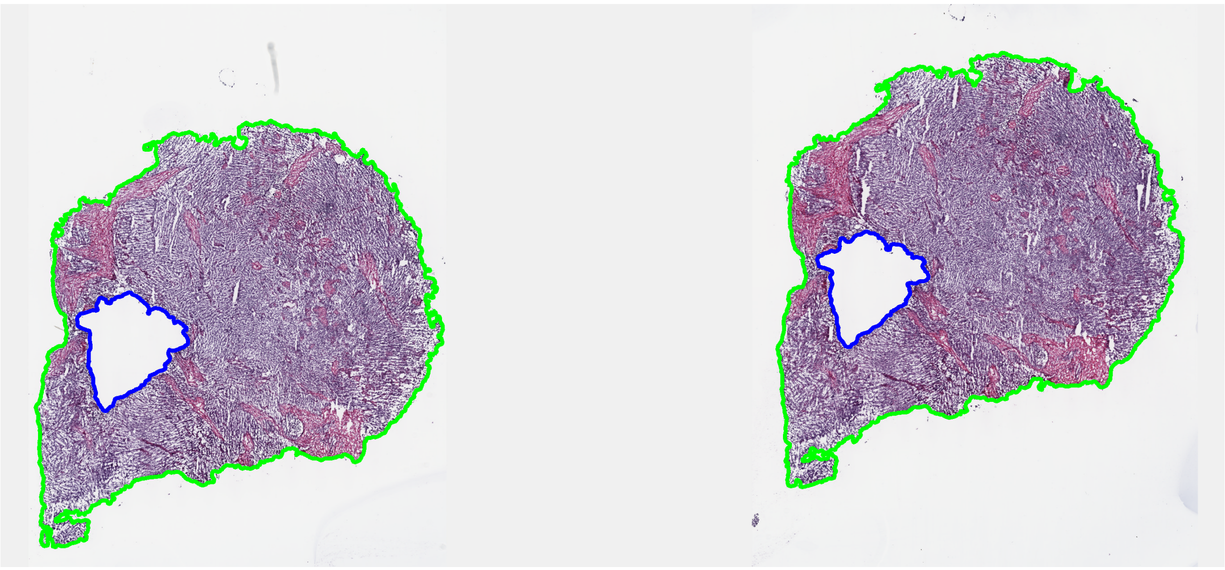


Figure 7.1: A Segmented Pathology Whole Slide Image

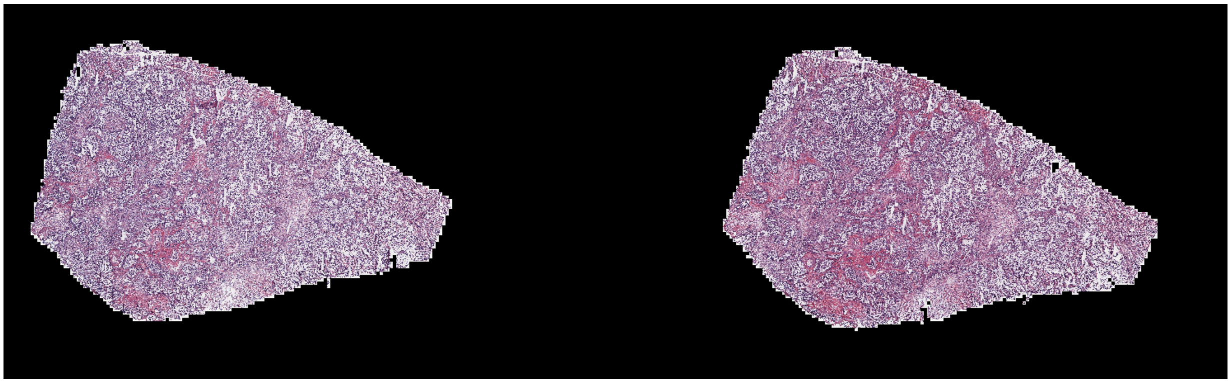


Figure 7.2: Downsampled Visualization of Pathology Whole Slide Image

Once the segmentation step is done, all the WSIs are patched with the dimensions of 256 × 256, and the patches are stored with their coordinates along with their metadata in hdf5 format. The number of patches extracted from each WSI is around hundreds of thousands as each of them is around 40x magnification.

7.4 Feature Extraction

Using a deep CNN, a low-dimensional feature representation of each patch is computed for each slide, which serves as input to the model. This is done to reduce the training time and reduce the computational cost. For feature extraction, we used a pre-trained CNN model, ResNet50, which is initially trained on the ImageNet dataset. Then, by using adaptive mean-spatial pooling after the 3rd residual block of the ResNet50 network, each patch with dimension 256×256 is converted into a feature vector of dimension 1024. Thus, by converting raw pixels to low-dimensional features, it is possible to fit all the patches belonging to a WSI into GPU memory at a time.

7.5 Experimental Setup and Training

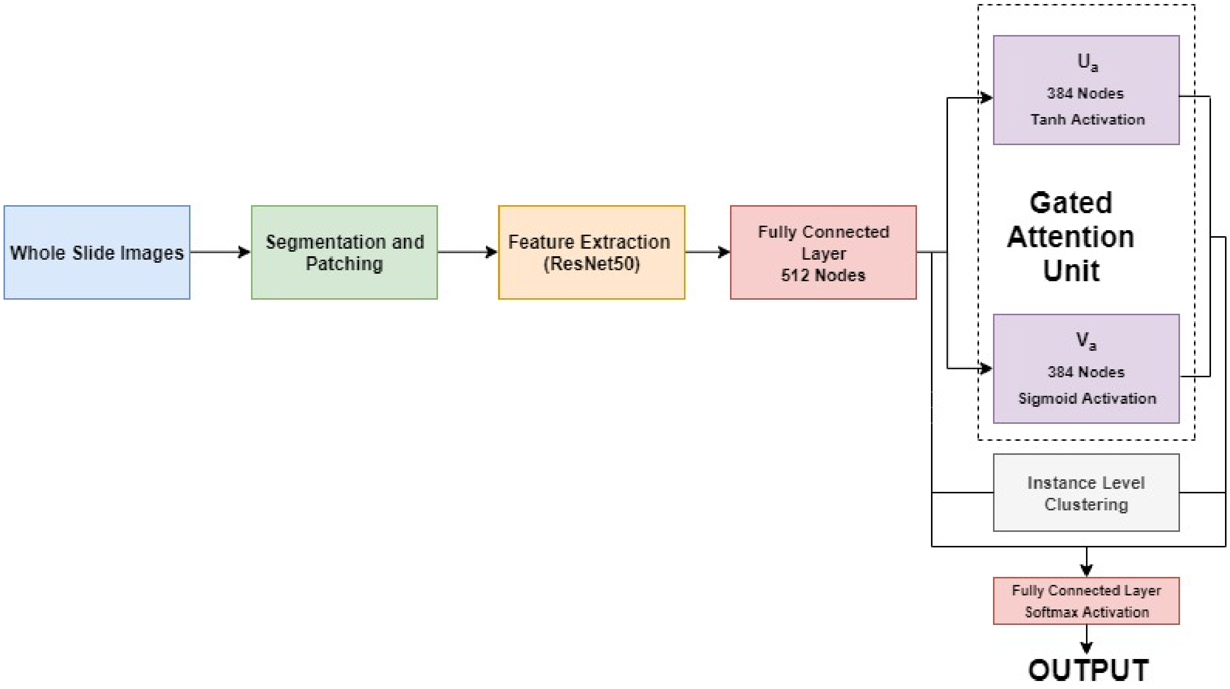

Whole Slide Images
Segmentation and Patching
Feature Extraction (ResNet50)
Fully Connected Layer 512 Nodes
U_a
384 Nodes
Tanh Activation
Gated Attention Unit
V_a
384 Nodes
Sigmoid Activation
Instance Level Clustering
Fully Connected Layer Softmax Activation
OUTPUT

Figure 7.3: Model Architecture with Gated Attention Unit and Instance Level Clustering

First, we segment and patch the gigapixel wide WSIs into tiny patches of dimension 256×256 from which the features are extracted from the patches using transfer learning technique. The features are extracted from the ResNet-50 model with an output feature vectors of dimension 1024, with the representation i_k, where k is the total number of patches. These features are then passed to a fully connected layer W_1 of dimension 512, j_k, where $(j_k = W_1 \times z_k^T)$. The 512-dimensional vector j_k is passed to the gated attention network with two layers U_a and V_a, also known as the attention backbone.The two layers are made up of "Tanh" and "sigmoid" activation function and 384 nodes each. The attention of all the patches belonging to WSI and the slide-level representation from the attention scores of the all the patches are calculated as shown in the equations 7.1:

$$a_k = \frac{exp[W_a(tanh(V_a j_k^T) \cdot (sigm(U_a j_k^T)))]}{\sum_{k=1}^{N} exp[W_a(tanh(V_a j_k^T) \cdot (sigm(U_a j_k^T)))]} \tag{7.1}$$

The slide-level representation $h_{slide}$is calculated by aggregrating the attention scores of all the patches as given in the equation 7.2:

$$h_{slide} = \sum_{k=1}^{N} a_k j_k \tag{7.2}$$

$$S_{slide} = W_c h_{slide}^T \tag{7.3}$$

The unnormalized slide-level score S_{slide}, is calculated through the classifier layer W_c, as shown in the equation 7.3. The probability is calculated for each class using the softmax function to predict the slide-to-slide level score.

The model also consists of an instance-level classifier after the 512-dimensional fully connected layer W_1. There are "m" classifiers depending on the total number of labels in our classification problem. In our case, there are seven binary instance classifiers. The weights belonging to each of the clustering network "m" is denoted as $W_{inst,m} \in R^{2\times512}$ and the score predicted for k^{ht} patch is given by $P_{m,k}$, where $P_{m,k} = W_{inst,m} j_k^T$. The attention values are used as the pseudo labels for each WSI for every training iteration to supervise the cluster. Not all the patches of a WSI are used for clustering, rather the patches are sorted in descending order, and the top-k and bottom-k patches are used for clustering for a given ground-truth label. While the top-k patches are set out to be positive evidence for the class, the bottom-k patches are denoted as negative evidence. In our experiment, we set the 'k' value to 15.

While the standard cross-entropy loss function is used for the classification part, smooth top1 SVM loss is used for the instance-level clustering task. The total loss of a given WSI is calculated as the sum of losses obtained from slide-level classification and instance-level clustering. The total loss is calculated and is backpropagated through the network, and the weights are updated during training.

The Adam optimizer with L2 decay of 1e-5 and learning rate of 2e-4 is used to train the model. In order to generalize the model's performance, ten-fold cross-validation is used, and the model is trained for 50-200 epochs based on the early stopping criteria. Finally, the best model with low validation loss is saved and is tested on the test set, and the results are discussed below, and the classification model is shown in Figure 7.3.

7.6 Results

The results obtained are promising as performance on the test set by the model produced an accuracy of 90.12%, with the precision of 0.89, recall of 0.90 and F1-score of 0.90. Also, the visualizations obtained on a few WSIs are shown in the figure 7.4. The attention heatmaps obtained for the WSIs are used to identify the tumor tissue as Regions of Interest. The attention scores are normalized between 0 and 1.0, where the value 0 represented negative evidence of the tumor while 1.0 represented strong evidence of the tumor patch. Using a diverging colormap, the normalized scores are converted into RGB colors in the slide, this enabling excellent visualization and interpretability. While high attention regions are displayed in red color in the attention map, the blue color regions are of low attention depicting negative evidence of tumor.

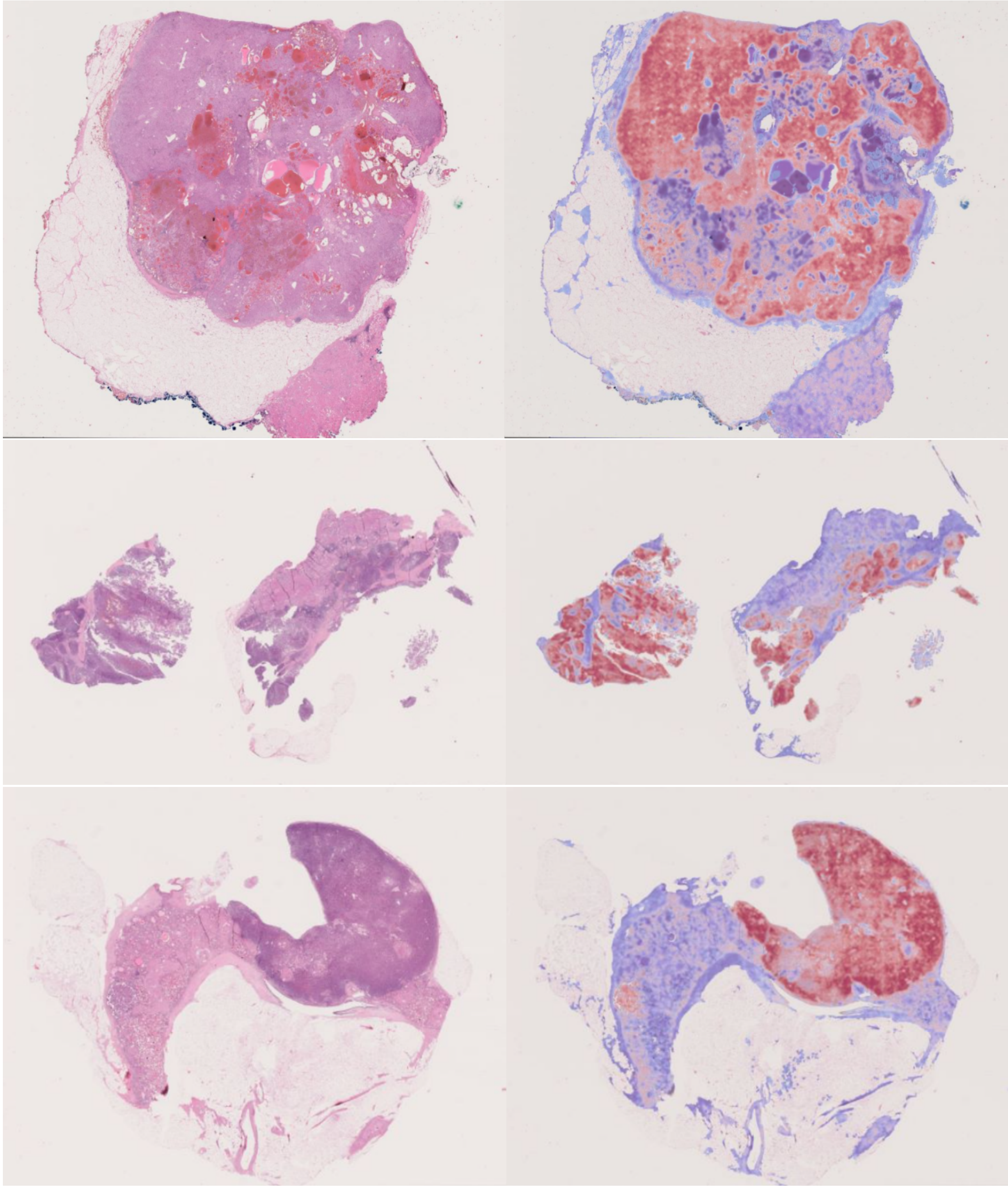


Figure 7.4: Attention Maps of Whole Slide Images

Chapter 8

Combined Classification of Histopathology Images and Reports

With the increase in the number of histopathology images and their respective reports these days, we are interested to know how pathology reports corresponding to a WSI aid in classification. The goal is to construct a single classification system for WSIs and pathology reports together, which uses the features of both reports as well as images. First, the weights are extracted from the second last layer of the DNN which has 512 nodes in the classification model for text as well as from the image classification model. Then the extracted features are used as the inputs to the model, which are concatenated and used for classifying the histopathology images and their respective reports to their disease type. As we can see, both the histopathology images and their reports are quite varied. Each of the images is gigapixel wide, and for the texts, each of the reports come in various form and structure with varying levels of details in it. The ultimate goal of the experiment is to identify a data point, which is an entire record of a patient at a point in time, to what disease it is. This chapter elaborates on how we extracted the image and text features and combined them for classification, experimental setup, and the results.

8.1 Feature Extraction from Images

Understanding the content of images is hard, especially medical-related and, more precisely, histopathology images. Unless with the help of a pathologist, we cannot easily understand the information a WSI holds. The images are segmented, patched and the patches are sent through a gated attention unit in the network. The weights after training the model are extracted from the second last layer of the DNN after the instance level clustering. The weights are saved in a pickle file which is used as image input to our experiment.

8.2 Feature Extraction from Texts

As with medical images, understanding medical reports is very challenging as they consist of medical terminology, each report is written differently and they are highly unstructured. In Chapter 6, we leveraged a pre-trained machine learning model using transfer learning, where one takes a well-trained model from one dataset or domain and applies it to a new one. Thus clinical BioBERT, which was pre-trained on clinical corpora, was used in our experiment as its relevant to use it in our reports, which are clinical related. The best model finalized was having the features from the pre-trained clinical BioBERT and TF-IDF features. We extracted the features of the trained reports from those reports from the layer before the final fully connected dense layer. The extracted weights are saved in a pickle file and are used as text input to our experiment.

8.3 Experimental Setup

This section explains the experimental setup used for our combined analysis. First, each of the WSI vector along with its report text vector is given as the input to the DNN model. These features are initially concatenated, and the combined feature vector is passed to the dense layer. Next, the features are sent to the batch normalization layer, which standardizes the inputs to a layer for each mini-batch in a DNN. This layer stabilizes the learning process and dramatically reduces the number of training epochs required to train deep networks. The normalized batch features are then sent to a series of dense layers, batch normalization layer and dropout layer, with the number of neurons in the dense layer being 256, 128, 64, 16; while that of the dropout values being 0.8, 0.6, 0.6, 0.4, respectively. Being a multi-class classification problem, with a total number of classes being 7, our neural network has the final layer as a dense layer with seven neurons with a “softmax” activation function. The overall architecture is shown in Figure 8.1.

The model is trained using Adam optimizer, which uses the stochastic gradient descent method based on adaptive estimation of first-order and second-order moments. The optimizer is initialized with the default learning rate of 0.01. Regarding the loss function used for the DNN classifier, we used “categorical cross-entropy”. Therefore, the combined

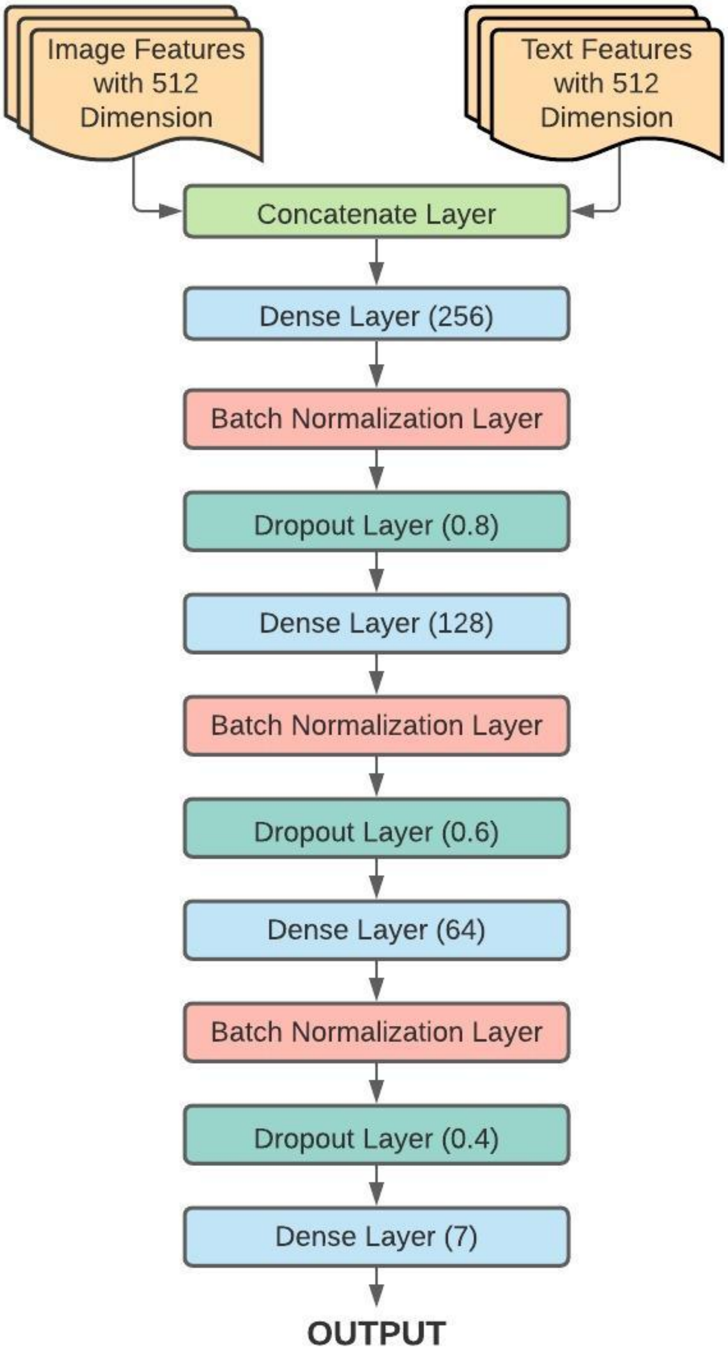


Figure 8.1: Architecture of the Combined Model

Table 8.1: Performance accuracy of Histopathology Reports only, Histopathology Images only and their combination

Histopathology Images Only	Histopathology Reports Only	Combination
90.12%	93.77%	95.7%

model is different from the model used to classify images and texts alone. The model is trained with 5-fold cross-validation, and its performance is evaluated based on the average of all the five folds. The section below elaborates the results obtained.

8.4 Results

This section describes the quantitative results obtained by our experiment, which combines the image features along with the text features for classification, where the features are obtained from a layer in the respective DNN models with 512 dimensions. In order to understand and justify the performance gains of the model, it makes sense to look at each component separately and compare them to the final model, as in Table 8.1.

Disease Type	Precision	Recall	F1-Score
Kidney Renal Papillary Cell Carcinoma	0.91	0.93	0.92
Kidney Renal Clear Cell Carcinoma	0.98	0.97	0.98
Lung Adenocarcinoma	0.97	0.93	0.95
Lung Squamous Cell Carcinoma	0.91	0.94	0.93
Testicular Germ Cell Tumors	0.96	1.00	0.98
Kidney Chromophobe	0.91	0.88	0.89
Thymoma	0.93	1.00	0.96

Table 8.2: Classification Report of the Image-Text Model

The combined model gave an excellent accuracy score of 95.7% and the classification report is as shown in Table 8.2 with a precision value of 0.96, recall value of 0.94 and F1-score of 0.96. The results are quite interesting where the medical text-only analysis is slightly better than the histopathology image-only analysis (while at the same time being a lot cheaper to compute). However, it is striking that the histopathology images alone can also deliver reasonable classifications. Furthermore, given that the two are competitive in performance, we expect that the combination of both the images and the text could lead to increased performance since they are picking up on different things. Indeed, when we look at the combined model performance, we see a significant boost in accuracy by 5%.

Chapter 9

Conclusions and Future Directions

9.1 Conclusions

In this paper, we presented three classification tasks and performed a comparative study. First, we performed histopathology image classification on the TCGA image data. Second, we experimented on the pathology reports classification, and finally, we created an architecture to combine both the images and their reports features to classify. The total number of classes available for the classification tasks was 7, representing disease types that belong to the organs kidneys, lung, thymus and testis.

Concerning the pathology images classification, we implemented an architecture that calculated the patches' attention scores and has binary instance-level clustering for classification. Firstly, the images which are gigapixel wide are segmented and patched. Features are then extracted from the patches from a pre-trained CNN, ResNet-50 model, which is pre-trained on the ImageNet dataset. Next, the features are extracted from the third residual block, thus obtaining a 1024 dimensional vector for all the patches. Feature extraction was performed to increase the training time and to decrease the computational cost of the model. Finally, the 1024 dimensional vector patches are passed through the fully connected layer with 512 hidden units sent to the gated attention unit, where the attention scores are calculated. The attention scores are used to perform instance-level clustering while sending them to the final output layer for prediction. The test accuracy obtained by the model is 90.12%.

Concerning the pathology reports classification, we experimented with unstructured free-text pathology reports with medical terminologies. In order to classify the reports into seven different tumor types, we examined the pathology reports. We reported several experiments by evaluating the word embeddings of the pre-trained models, the TF–IDF vectorization technique and also the combination of both. We found that the combination of TF–IDF with pre-trained model word embeddings consistently outperformed contextualized word embeddings and TF–IDF when performed individually. The best performance for pathology report classification was observed for the model that concatenated the embeddings from Clinical BioBERT with the TF–IDF vectors. This seems to form a reasonable baseline and provides valuable insights into the future of digital pathology report analysis. The text classification experiment reported accuracy of 93.77%.

Finally, to understand the performance of the classification model when fusing the image and text features were performed, which gave interesting results. The inputs given to the final combination classification model were the features extracted from the layer before the final output layer of both the image and text classification experiments. The features of the histopathology images and their corresponding pathology reports are concatenated and are passed through the DNN. This combined model gave a classification accuracy of 95.7%, thus elevating the importance of features.

9.2 Future Studies

The study of histopathology images and reports in machine learning is very vast. The proposed system can be adapted for diverse tasks associated with histopathology image-text-based classification relevant to clinical settings. Improved results can be obtained by changing the model design and its parameters. Future studies will investigate the combined model's performance on other datasets with more comprehensive cancer cases. Also, deep analysis of the words, embeddings and corresponding attention areas of the images can be performed.

Furthermore, Autoencoder can be implemented to reduce the histopathology image size by compressing without losing essential features as it can regenerate up to 90% of the original images. The study can also be extended to the automatic generation of reports given a histopathology image. Therefore with future perspectives and further development in this field, we can transform the personalized diagnosis into an improved diagnostic system, thus reducing the workload of pathologists.

References

[1] Deep learning: Recurrent neural networks. https://en.wikipedia.org/wiki/File:Recurrent_neural_network_unfold.svg. Accessed : 2021-07-09.

[2] Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*, 2019.

[3] Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew Peters, Joanna Power, Sam Skjonsberg, Lucy Lu Wang, Chris Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. Construction of the literature graph in semantic scholar. *arXiv preprint arXiv:1805.02262*, 2018.

[4] Guilherme Aresta, Teresa Araújo, Scotty Kwok, Sai Saketh Chennamsetty, Mohammed Safwan, Varghese Alex, Bahram Marami, Marcel Prastawa, Monica Chan, Michael Donovan, and et al. Bach: Grand challenge on breast cancer histology images. *Medical Image Analysis*, 56:122–139, Aug 2019. Elsevier BV.

[5] Morteza Babaie, Shivam Kalra, Aditya Sriram, Christopher Mitcheltree, Shujin Zhu, Amin Khatami, Shahryar Rahnamayan, and Hamid R. Tizhoosh. Classification and retrieval of digital pathology scans: A new dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 760–768, 2017.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2016.

[7] Meriem Bahi and Mohamed Batouche. Deep learning for ligand-based virtual screening in drug discovery. In *2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–5, 2018.

[8] Y. Bengio, P. Simard, and P. Frasconi. Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[9] Yoshua Bengio. Learning Deep Architectures for AI. *Found. Trends Mach. Learn.*, 2(1):1–127, January 2009.

[10] Kaustav Bera, Kurt Schalper, David Rimm, Vamsidhar Velcheti, and Anant Madabhushi. Artificial intelligence in digital pathology — new tools for diagnosis and precision oncology. *Nature Reviews Clinical Oncology*, 16, August 2019.

[11] E Biganzoli, P Boracchi, L Mariani, and E Marubini. Feed forward neural networks for the analysis of censored survival data: a partial logistic regression approach. *Statistics in medicine*, 17(10):1169—1186, May 1998.

[12] David Blei, Andrew Ng, and Michael Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, May 2003.

[13] Ali Borji, Dicky N. Sihite, and Laurent Itti. What/where to look next? modeling top-down visual attention in complex interactive environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(5):523–538, 2014.

[14] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 144–152, New York, NY, USA, 1992. Association for Computing Machinery.

[15] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth and Brooks, Monterey, CA, 1984.

[16] Leo Breiman. Arcing classifier (with discussion and a rejoinder by the author). *The Annals of Statistics*, 26(3):801 – 849, 1998.

[17] Péter Bándi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, Quanzheng Li, Farhad Ghazvinian Zanjani, Svitlana Zinger, Keisuke Fukuta, Daisuke Komura, Vlado Ovtcharov, Shenghua Cheng, Shaoqun

Zeng, Jeppe Thagaard, Anders B. Dahl, Huangjing Lin, Hao Chen, Ludwig Jacobsson, Martin Hedlund, Melih Çetin, Eren Halıcı, Hunter Jackson, Richard Chen, Fabian Both, Jörg Franke, Heidi Küsters-Vandevelde, Willem Vreuls, Peter Bult, Bram van Ginneken, Jeroen van der Laak, and Geert Litjens. From detection of individual metastases to classification of lymph node status at the patient level: The camelyon17 challenge. *IEEE Transactions on Medical Imaging*, 38(2):550–560, 2019.

[18] Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

[19] Manish Chablani. Sequence to sequence model: Introduction and concepts. https://towardsdatascience.com/sequence-to-sequence-model-introduction-and-concepts-44d9b41cd42d. Accessed : 2021-07-09.

[20] Liang Chen, Paul Bentley, Kensaku Mori, Kazunari Misawa, Michitaka Fujiwara, and Daniel Rueckert. Self-supervised learning for medical image analysis using image context restoration. *Medical Image Analysis*, 58:101539, 07 2019.

[21] Travers Ching, Daniel Himmelstein, Brett Beaulieu-Jones, Alexandr Kalinin, T. Do, Gregory Way, Enrico Ferrero, Paul Agapow, Michael Zietz, Michael Hoffman, Wei Xie, Gail Rosen, Benjamin Lengerich, Johnny Israeli, Jack Lanchantin, Stephen Woloszynek, Anne Carpenter, Avanti Shrikumar, Jinbo Xu, and Casey Greene. Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15:20170387, April 2018.

[22] François Chollet. Xception: Deep learning with depthwise separable convolutions. *arXiv preprint arXiv:1610.02357*, 2017.

[23] Dan C. Cireşan, Ueli Meier, Jonathan Masci, Luca M. Gambardella, and Jürgen Schmidhuber. Flexible, high performance convolutional neural networks for image classification. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, page 1237–1242. AAAI Press, 2011.

[24] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3642–3649, 2012.

[25] Anni Coden, Guergana Savova, Igor Sominsky, Michael Tanenblatt, James Masanz, Karin Schuler, James Cooper, Wei Guan, and Piet Groen. Automatically extracting cancer disease characteristics from pathology reports into a disease knowledge representation model. *Journal of biomedical informatics*, 42:937–49, 10 2009.

[26] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, November 2011.

[27] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680. Association for Computational Linguistics, 2017.

[28] Paolo Contiero, Andrea Tittarelli, Anna Maghini, Sabrina Fabiano, Emanuela Frassoldi, Enrica Costa, Daniela Gada, Tiziana Codazzi, Paolo Crosignani, Roberto Tessandori, and Giovanna Tagliabue. Comparison with manual registration reveals satisfactory completeness and efficiency of a computerized cancer registration system. *J. of Biomedical Informatics*, 41(1):24–32, February 2008.

[29] J.S. Cramer. The origins of logistic regression. *Tinbergen Institute, Tinbergen Institute Discussion Papers*, 01 2002.

[30] Rebecca Crowley, Melissa Castine, Kevin Mitchell, Girish Chavan, Tara McSherry, and Michael Feldman. caties: A grid based system for coding and retrieval of surgical pathology reports and tissue specimens in support of translational research. *Journal of the American Medical Informatics Association : JAMIA*, 17:253–64, 05 2010.

[31] Anne-Marie Currie, Travis Fricke, Agnes Gawne, Ric Johnston, John Liu, and Barbara Stein. *Automated Extraction of Free-Text from Pathology Reports*. AMIA, 2006.

[32] A. Dahl, A. Ozkan, and H. Dalianis. Pathology text mining - on norwegian prostate cancer reports. In *2016 IEEE 32nd International Conference on Data Engineering Workshops (ICDEW)*, pages 84–87, Los Alamitos, CA, USA, may 2016. IEEE Computer Society.

[33] Leonard D'Avolio, Thien Nguyen, Wildon Farwell, Yongming Chen, Felicia Fitzmeyer, Owen Harris, and Louis Fiore. Evaluation of a generalizable approach to clinical information retrieval using the automated retrieval console (arc). *Journal of the American Medical Informatics Association : JAMIA*, 17:375–82, July 2010.

[34] D. Defays. An efficient algorithm for a complete link method. *The Computer Journal*, 20(4):364–366, 01 1977.

[35] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2019.

[36] Rahul Dey and Fathi M. Salem. Gate-variants of gated recurrent unit (gru) neural networks. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1597–1600, 2017.

[37] Neofytos Dimitriou, Ognjen Arandjelović, and Peter D Caie. Deep learning for whole slide image analysis: An overview. *arXiv preprint arXiv:1910.11097*, 2019.

[38] K. Doi. Computer-aided diagnosis in medical imaging: Historical review, current status and future potential. *Computerized medical imaging and graphics : the official journal of the Computerized Medical Imaging Society*, 31 4-5:198–211, 2007.

[39] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, , and the CAMELYON16 Consortium. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *JAMA*, 318(22):2199–2210, 12 2017.

[40] C. Elston and I. Ellis. pathological prognostic factors in breast cancer. i. the value of histological grade in breast cancer: experience from a large study with long-term follow-up. *Histopathology*, 19, 1991.

[41] Hamed Erfankhah, Mehran Yazdi, Morteza Babaie, and Hamid R. Tizhoosh. Heterogeneity-aware local binary patterns for retrieval of histopathology images. *IEEE Access*, 7:18354–18367, 2019.

[42] Andre Esteva, Alexandre Robicquet, Bharath Ramsundar, Volodymyr Kuleshov, M. DePristo, K. Chou, Claire Cui, Greg Corrado, S. Thrun, and J. Dean. A guide to deep learning in healthcare. *Nature Medicine*, 25:24–29, 2019.

[43] Fartash Faghri, David J. Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2018.

[44] Jerome Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38:367–378, 02 2002.

[45] Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 1*, 2:559–572.

[46] Shang Gao, M. T. Young, John X. Qiu, Hong-Jun Yoon, J. B. Christian, P. Fearn, G. Tourassi, and Arvind Ramanthan. Hierarchical attention networks for information extraction from cancer pathology reports. *Journal of the American Medical Informatics Association : JAMIA*, 25:321 – 330, 2018.

[47] R. Grossman, Allison P. Heath, V. Ferretti, H. Varmus, D. Lowy, W. Kibbe, and L. Staudt. Toward a shared vision for cancer genomic data. *The New England journal of medicine*, 375 12:1109–12, 2016.

[48] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. *arXiv preprint arXiv:1904.00560*, 2019.

[49] Qingji Guan, Yaping Huang, Zhun Zhong, Zhedong Zheng, Liang Zheng, and Yi Yang. Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. 2018.

[50] Metin N. Gurcan, Laura E. Boucheron, Ali Can, Anant Madabhushi, Nasir M. Rajpoot, and Bulent Yener. Histopathological image analysis: A review. *IEEE Reviews in Biomedical Engineering*, 2:147–171, 2009.

[51] Kevin Gurney. *An Introduction to Neural Networks.* Taylor &amp; Francis, Inc., USA, 1997.

[52] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

[53] Douglas Hanahan and Robert A Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144(5):646—674, March 2011.

[54] S. Hassanpour and C. Langlotz. Information extraction from multi-institutional radiology reports. *Artificial intelligence in medicine*, 66:29–39, 2016.

[55] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[56] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997.

[57] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and less than 0.5mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[58] Vladimir Iglovikov, Sergey Mushinskiy, and Vladimir Osin. Satellite imagery feature detection using deep convolutional neural network: A kaggle competition. *arXiv preprint arXiv:1706.06169*, 2017.

[59] Vladimir Iglovikov, Alexander Rakhlin, Alexandr A. Kalinin, and Alexey Shvets. *Pediatric Bone Age Assessment Using Deep Convolutional Neural Networks*. Cold Spring Harbor Laboratory, 2018.

[60] Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiv preprint arXiv:1801.05746*, 2018.

[61] Osamu Iizuka, Fahdi Kanavati, Kei Kato, Michael Rambeau, Koji Arihiro, and Masayuki Tsuneki. Deep learning models for histopathological classification of gastric and colonic epithelial tumours. *Scientific Reports*, 10, 01 2020.

[62] Maximilian Ilse, Jakub M. Tomczak, and Max Welling. Attention-based deep multiple instance learning. *arXiv preprint arXiv:1802.04712*, 2018.

[63] Timothy D. Imler, J. Morea, C. Kahi, and T. Imperiale. Natural language processing accurately categorizes findings from colonoscopy and pathology reports. *Clinical gastroenterology and hepatology : the official clinical practice journal of the American Gastroenterological Association*, 11 6:689–94, 2013.

[64] S. Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, abs/1502.03167, 2015.

[65] A. Janowczyk, Ajay Basavanhally, and A. Madabhushi. Stain normalization using sparse autoencoders (stanosa): Application to digital pathology. *Computerized medical imaging and graphics : the official journal of the Computerized Medical Imaging Society*, 57:50–61, 2017.

[66] A. Janowczyk and A. Madabhushi. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *Journal of Pathology Informatics*, 7, 2016.

[67] Ahmedin Jemal, Rebecca Siegel, Elizabeth Ward, Yongping Hao, Jiaquan Xu, and Michael Thun. Cancer statistics, 2009. *CA: a cancer journal for clinicians*, 59:225–49, 07 2009.

[68] T. Jiang, N. Navab, J.P.W. Pluim, and M.A. Viergever, editors. *Medical image computing and computer-assisted intervention (MICCAI 2010) : 13th international conference, Beijing, China, September 20-24, 2010, proceedings, part I*. Lecture notes in computer science. Springer, Germany, 2010.

[69] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *arXiv preprint arXiv:1902.06162*, 2019.

[70] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

[71] Shivam Kalra, Larry Li, and Hamid R. Tizhoosh. Automatic classification of pathology reports using tf-idf features. *arXiv preprint arXiv:1903.07406*, 2019.

[72] Katikapalli Subramanyam Kalyan, Ajit Rajasekharan, and Sivanesan Sangeetha. Ammu – a survey of transformer-based biomedical pretrained language models. *arXiv preprint arXiv:2105.00827*, 2021.

[73] Ning Kang, Bharat Singh, Z. Afzal, E. V. Mulligen, and J. Kors. Using rule-based natural language processing to improve disease normalization in biomedical text. *Journal of the American Medical Informatics Association : JAMIA*, 20:876 – 881, 2013.

[74] Adnan Khan, Nasir Rajpoot, Darren Treanor, and Derek Magee. A non-linear mapping approach to stain normalisation in digital histopathology images using image-specific colour deconvolution. *IEEE Transactions on Biomedical Engineering*, 61, 06 2014.

[75] Brady Kieffer, Morteza Babaie, Shivam Kalra, and H. R. Tizhoosh. Convolutional neural networks for histopathology image classification: Training vs. using pre-trained networks. *arXiv preprint arXiv:1710.05726*, 2017.

[76] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Skip-thought vectors. *arXiv preprint arXiv:1506.06726*, 2015.

[77] Daisuke Komura and Shumpei Ishikawa. Machine learning methods for histopathological image analysis. *Computational and Structural Biotechnology Journal*, 16, 09 2017.

[78] Sonal Kothari, John Phan, and May Wang. Eliminating tissue-fold artifacts in histopathological whole-slide images for improved image-based prediction of cancer grade. *Journal of pathology informatics*, 4:22, 08 2013.

[79] Oren Z. Kraus, Ben T Grys, Jimmy Ba, Yolanda T. Chong, B. Frey, Charles Boone, and B. Andrews. Automated analysis of high-content microscopy data with deep learning. *Molecular Systems Biology*, 13, 2017.

[80] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2017.

[81] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[82] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, Sep 2019.

[83] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. *arXiv preprint arXiv:1803.08024*, 2018.

[84] Xiaomeng Li, Lequan Yu, Hao Chen, Chi-Wing Fu, Lei Xing, and Pheng-Ann Heng. Transformation consistent self-ensembling model for semi-supervised medical image segmentation. *arXiv preprint arXiv:1903.00348*, 2020.

[85] Xingyu Li and Konstantinos N. Plataniotis. A complete color normalization approach to histopathology images using color cues computed from saturation-weighted statistics. *IEEE Transactions on Biomedical Engineering*, 62(7):1862–1873, 2015.

[86] Sam Liebman. Mapping word embeddings with word2vec. https://towardsdatascience.com/mapping-word-embeddings-with-word2vec-99a799dc9695. Accessed : 2021-07-09.

[87] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, Dec 2017.

[88] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[89] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[90] K. Loganathan, R. Kumar, V. Nagaraj, and Tegil John. Cnn & lstm using python for automatic image captioning. *Materials Today: Proceedings*, 12 2020.

[91] Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. *arXiv preprint arXiv:1803.02893*, 2018.

[92] Ming Y. Lu, Drew F. K. Williamson, Tiffany Y. Chen, Richard J. Chen, Matteo Barbieri, and Faisal Mahmood. Data efficient and weakly supervised computational pathology on whole slide images. *arXiv preprint arXiv:2004.09666*, 2020.

[93] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*, 2017.

[94] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[95] Marc Macenko, Marc Niethammer, J. S. Marron, David Borland, John T. Woosley, Xiaojun Guan, Charles Schmitt, and Nancy E. Thomas. A method for normalizing histology slides for quantitative analysis. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1107–1110, 2009.

[96] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[97] Valeria Maeda-Gutiérrez, Carlos E. Galván-Tejada, Laura A. Zanella-Calzada, José M. Celaya-Padilla, Jorge I. Galván-Tejada, Hamurabi Gamboa-Rosales, Huizilopoztli Luna-García, Rafael Magallanes-Quintanar, Carlos A. Guerrero Méndez, and Carlos A. Olvera-Olvera. Comparison of convolutional neural network architectures for classification of tomato plant diseases. *Applied Sciences*, 10(4), 2020.

[98] M. Maron. Automatic indexing: An experimental inquiry. *J. ACM*, 8:404–417, 1961.

[99] David Martínez, G. Pitson, Andrew D. MacKinlay, and L. Cavedon. Cross-hospital portability of information extraction of cancer staging information. *Artificial intelligence in medicine*, 62 1:11–21, 2014.

[100] Iain Mccowan, Darren Moore, Anthony Nguyen, Rayleen Bowman, Belinda Clarke, Edwina Duhig, and Mary-Jane Fry. Collection of cancer stage data by classifying free-text medical reports. *Journal of the American Medical Informatics Association : JAMIA*, 14:736–45, 08 2007.

[101] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[102] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*, 2013.

[103] Sparsh Mittal. A survey of fpga-based accelerators for convolutional neural networks. *Neural Computing and Applications*, 32:1109–1139, 2018.

[104] Maryam Najafabadi, Flavio Villanustre, Taghi Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2, 12 2015.

[105] Anobel Odisho, Briton Park, Nicholas Altieri, John DeNero, Matthew Cooperberg, Peter Carroll, and Bin Yu. Natural language processing systems for pathology parsing in limited data environments with uncertainty estimation. *JAMIA Open*, 3:431–438, 10 2020.

[106] Christopher Olah. Understanding lstm networks. https://colah.github.io/posts/2015-08-Understanding-LSTMs/. Accessed : 2021-07-09.

[107] Ying Ou and Jon Patrick. Automatic population of structured reports from narrative pathology reports. In *Proceedings of the Seventh Australasian Workshop on Health Informatics and Knowledge Management - Volume 153*, HIKM '14, page 41–50, AUS, 2014. Australian Computer Society, Inc.

[108] Anil Parwani. Next generation diagnostic pathology: Use of digital pathology and artificial intelligence tools to augment a pathological diagnosis. *Diagnostic Pathology*, 14, 12 2019.

[109] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[110] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[111] Ryan Poplin, Avinash Varadarajan, Katy Blumer, Yun Liu, Michael McConnell, Greg Corrado, Lily Peng, and Dale Webster. Predicting cardiovascular risk factors from retinal fundus photographs using deep learning. *Nature Biomedical Engineering*, 2, 03 2018.

[112] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 2004.

[113] Alexander Rakhlin. Diabetic retinopathy detection through integration of deep learning classification framework. *bioRxiv*, 2018.

[114] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley. Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41, 2001.

[115] Abtin Riasatian, Maral Rasoolijaberi, Morteza Babaei, and H. R. Tizhoosh. A comparative study of u-net topologies for background removal in histopathology images. *arXiv preprint arXiv:2006.06531*, 2020.

[116] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. ”why should i trust you?”: Explaining the predictions of any classifier. *arXiv preprint arXiv:1602.04938*, 2016.

[117] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

[118] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. *Learning Representations by Back-Propagating Errors*, page 696–699. MIT Press, Cambridge, MA, USA, 1988.

[119] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, Jan 2015.

[120] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2016.

[121] Marzia Settino and Mario Cannataro. Survey of main tools for querying and analyzing tcga data. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1711–1718, 2018.

[122] R. Sibson. SLINK: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34, 01 1973.

[123] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2015.

[124] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

[125] I. Spasić, J. Livsey, J. Keane, and G. Nenadic. Text mining of cancer-related information: Review of current status and future directions. *International journal of medical informatics*, 83 9:605–23, 2014.

[126] Chetan Srinidhi, Ozan Ciga, and Anne Martel. Deep neural network models for computational histopathology: A survey, 12 2019.

[127] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 06 2014.

[128] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*, 2014.

[129] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014.

[130] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *arXiv preprint arXiv:1512.00567*, 2015.

[131] David Tellez, Geert Litjens, Jeroen van der Laak, and Francesco Ciompi. Neural image compression for gigapixel histopathology image analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(2):567–578, Feb 2021.

[132] Aleksei Tiulpin, Jérôme Thevenot, Esa Rahtu, Petri Lehenkari, and Simo Saarakkala. Automatic knee osteoarthritis diagnosis from plain radiographs: A deep learning-based approach. *Scientific Reports*, 8, 01 2018.

[133] Abhishek Vahadane, Tingying Peng, Shadi Albarqouni, Maximilian Baust, Katja Steiger, Anna Melissa Schlitter, Amit Sethi, Irene Esposito, and Nassir Navab. Structure-preserved color normalization for histological images. In *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, pages 1012–1015, 2015.

[134] Rebecka Weegar, Jan F Nygård, and Hercules Dalianis. Efficient encoding of pathology reports using natural language processing. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 778–783, Varna, Bulgaria, September 2017. INCOMA Ltd.

[135] Paul Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78:1550 – 1560, 11 1990.

[136] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

[137] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2016.

[138] Yan Xu, Zhipeng Jia, Liang-Bo Wang, Yuqing Ai, Fang Zhang, Maode Lai, and Eric Chang. Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC Bioinformatics*, 18, 05 2017.

[139] Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*, 2017.

[140] Chaoyi Zhang, Yang Song, Donghao Zhang, Sidong Liu, Mei Chen, and Weidong Cai. Whole slide image classification via iterative patch labelling. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 1408–1412, 2018.

[141] Wan Zhu, Longxiang Xie, Jianye Han, and Xiangqian Guo. The application of deep learning in cancer prognosis prediction. *Cancers*, 12:603, 03 2020.