Automated Diagnosis of Acute Appendicitis Based on Clinical Notes

Steven Kester Yuwono

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2018

Supervisor:
Professor Ng Hwee Tou

Examiners:
Professor Lee Wee Sun
Professor Ng See Kiong

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Steven Kester Yuwono

23 January 2018

Acknowledgments

I would like to express my gratitude to my supervisor, Provost's Chair Professor Ng Hwee Tou for his guidance and support. His sharp analytical skills and detail-oriented nature have made me a better researcher.

I would also like to thank Dr. Ngiam Kee Yuan, a consultant and surgeon in the Department of Thyroid and Endocrine Surgery, National University Hospital. Without him, this thesis would not have been possible. He has kindly given us access to hospital data and spent his invaluable time to provide medical insights to tackle problems from the correct angle.

I would like to thank my lab-mates in the NUS Natural Language Processing Group: Benjamin Yap, Christian Hadiwinoto, Kaveh Taghipour, Nhu Thao Nguyen, Shamil Chollampatt, Souvik Kundu, and Tapas Nayak; for many meaningful discussions and making the lab a conducive place to do research.

Contents

Summary

Medical diagnosis is a very important task which requires high accuracy and efficiency, especially for patients admitted to the accident and emergency (A&E) department. These patients have a wide range of medical conditions. However, it is highly improbable for a medical doctor to gain expertise in all medical fields. Therefore, it is extremely challenging for the attending doctors to perform quick and accurate diagnosis in order to prevent further complications.

Most of the relevant and useful information (e.g., signs and symptoms) are in the form of free text notes entered by medical doctors. The text itself does not consist of well-formed and well-structured sentences, but rather medical abbreviations and terms written in point forms, and frequent misspelling (due to the time constraint imposed on the doctors).

The main objective of this thesis is to develop a system to aid in the diagnosis of appendicitis during A&E admissions by giving diagnosis recommendations to doctors. Given an emergency department (ED) note, the system is expected to compute the probability of appendicitis based on the medical conditions of a patient, thereby helping to improve ED doctors' accuracy and reduce the number of misdiagnoses in A&E visits.

We have developed a novel neural network architecture named convolutional residual recurrent neural network (CR2). Our neural network model is able to learn from free texts and additional real-valued features without any feature engineering. The performance of our neural network model is promising and close to the performance of ED doctors. We have incorporated an attention mechanism in our neural network model to gain insights into how the model assigns importance to words and phrases. Visualization shows that the model is able to meaningfully learn important features, signs, and symptoms of patients from unstructured free-text ED notes, which helps medical doctors to make better diagnosis.

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Recent advances in natural language processing (NLP), machine learning, and neural networks have resulted in many successful studies in solving various problems in multiple domains, such as speech recognition (Graves and Jaitly, 2014), caption generation (Vinyals et al., 2015), machine translation (Bahdanau, Cho, and Bengio, 2015), automated essay scoring (Taghipour and Ng, 2016), and image classification (Simonyan and Zisserman, 2015; Cao et al., 2015). Despite the recent successes of machine learning and big data analytics, the use of such systems is still very limited in the medical and healthcare domain. This gap is attributed to two main factors: (1) privacy issues and (2) resistance by medical doctors.

In Singapore, personal data privacy is protected by Personal Data Protection Act (PDPA). Regulations on protecting medical records are stricter, where only authorized personnel are allowed to view or edit the records. Therefore, using medical records for research purposes requires patients' personal health information (PHI) to be de-identified. Some examples of PHI categories are patient's name, identification number, contact number, and date. In the United States of Amer-

ica, medical records are safeguarded by HIPAA (Health Insurance Portability and Accountability Act), where the rules and regulations of removing PHI are much stricter, including 17 different PHI categories (Uzuner, Luo, and Szolovits, 2007). Emergency department notes constitute an essential source of information to facilitate medical research. However, anonymizing free-text notes has proven to be challenging. We will provide more details in Section 2.1.

To address the second issue mentioned above, we propose to develop an automated diagnosis system which will work as an advisor to complement medical doctors instead of working autonomously to replace doctors. Our proposed system will aid doctors in diagnosing a patient by giving diagnosis recommendations **without** requiring additional work or actions. The inputs to the automated diagnosis system are the emergency department (ED) notes (free texts) and lab results obtained directly from the hospital computer system. Given an ED note and optional lab results, the system recommends a list of the most probable diagnoses to ED doctors. As such, ED doctors still make the final diagnosis and decision. Moreover, the system is also required to provide visualizations or reasoning steps on why it suggests certain diagnoses. Interpretability is crucial in medical domain, since doctors need to understand the reasoning behind the diagnosis suggestions and then to re-evaluate their decisions.

Creating a machine learning system which learns from free texts is very challenging, especially when the texts contain sentence fragments, bullet points, misspellings, and frequent use of medical abbreviations. As such, our ED notes pose additional challenges to subsequent processing by downstream natural language processing modules like part-of-speech tagging, coreference resolution, etc. Existing NLP techniques or toolkits will not work on our ED notes out-of-the-box without any modifications or re-training. In addition, we might need to incorporate information from other structured data (e.g., lab results, age, and gender) which

are useful features in diagnosing a patient. However, the main and most important source of information is the ED notes in the form of free text. We will describe our proposed approach in great detail in Chapter 3.

Our goal is to develop an automated diagnosis system which is able to learn from past ED notes without any feature engineering and diagnose appendicitis with accuracy competitive to ED doctors. In future, we plan to conduct a clinical trial to validate the model where there are two A&E settings, one with our system, and another one without our system. Our goal is to improve the diagnosis accuracy of doctors in the A&E department who use our system as compared to those who do not use our system, thus benefiting more patients. The work in this thesis is novel in that our automated diagnosis system mainly learns from ED notes (free texts) hence requiring no additional work by medical personnel to use the system.

1.2 Acute Appendicitis

In this thesis, we focus on diagnosing *acute appendicitis*. There are existing scoring schemes to aid doctors in diagnosing appendicitis. The most well-known scoring scheme currently used by clinicians is Alvarado Score (Alvarado, 1986). It is also known as the MANTRELS score, a mnemonic to remember the features **M**igration of pain to the right lower quadrant, **A**norexia, **N**ausea or vomiting, **T**enderness in the right lower quadrant, **R**ebound pain, **E**levated temperature (fever), **L**eukocytosis, and **S**hift to the left (Neutrophils). The score for each feature will be added together and a higher cumulative score indicates that a patient is more likely to have appendicitis.

(McKay and Shepherd, 2007) demonstrated that the Alvarado score performed very well to rule out appendicitis but it performed only fairly well for ruling in appendicitis. However, the Alvarado score is not useful for ruling in or out appendicitis with intermediate scores. Therefore, it was recommended in the liter-

ature for a patient to undergo a computed tomography (CT) scan when doctors are unsure, in particular in the middle range of the Alvarado score (4 to 6 out of 10). A CT scan was found to be 98% accurate in diagnosing acute appendicitis (Rao et al., 1998). However, CT scans are harmful to our body with a radiation factor (CT abdomen) of 400 times[1] of a regular chest X-ray. We chose to focus on diagnosing *acute appendicitis* in this thesis because there will be high clinical impact if our system is successful. Our system is expected to help reduce cost by minimizing the number of patients requiring CT scans.

Although previous studies on appendicitis have been published in the research literature, (Petroianu, 2012) reported that appendicitis is the most common abdominal emergency but the diagnosis of appendicitis remains a medical challenge. Therefore, this thesis aims to address the current gap in the literature by developing an automated system to diagnose acute appendicitis with potential for practical use in hospitals.

[1] https://www.fda.gov/radiation-emittingproducts/radiationemittingproductsandprocedures/medicalimaging/medicalx-rays/ucm115329.htm

Chapter 2

Related Work

2.1 Anonymization

Clinical discharge summaries and ED notes are essential sources of information to facilitate medical research. However, they contain patients' PHI which, if disclosed, would compromise patients' privacy. Various techniques have been applied to create de-identification systems and they have performed well (Uzuner, Luo, and Szolovits, 2007). These de-identification systems utilize either machine learning approaches such as support vector machines (Uzuner et al., 2008), conditional random fields (Wellner et al., 2007), and decision trees (Szarvas, Farkas, and Busa-Fekete, 2007), or rule-based approaches with pattern matching (Douglass et al., 2004).

We have successfully developed a simple rule-based anonymization algorithm with regular expression pattern matching that runs efficiently and achieves very high recall (Yuwono, Ng, and Ngiam, 2016).

2.2 Automated Diagnosis

Over the last few decades, research has been conducted to develop automated systems to improve patient care. The work ranges from a generic inpatient reminder system, to a specific disease diagnosis system. An example of an inpatient reminder system is Antibiotic Assistant (Evans et al., 1998), which links to a database of patient records, allowing effective retrieval of patient information to assist doctors in the use of anti-infective agents.

In this thesis, we focus on a specific clinical decision support system, namely a medical diagnosis decision support system. Studies on such diagnosis decision support systems have been carried out since 1970s. One example is the automated medical diagnosis system to diagnose heart diseases (Stensmo and Sejnowski, 1996). A more recent example is automatic glaucoma diagnosis through medical imaging informatics (Liu et al., 2013). The system is able to automatically diagnose a patient with glaucoma using the patient's data, medical retinal image, and genome. Another recent study is the classification of skin cancer with deep neural networks (Esteva et al., 2017). They claim that their deep neural network model (without any feature engineering) is able to classify skin lesions from clinical skin images better than the average dermatologist. A recent study by (Prakash et al., 2017) has proposed a novel neural network approach to perform clinical diagnostic inferencing using free-text discharge summaries. This work is similar to ours but not directly comparable. Moreover, diagnosis classification using discharge summaries might not be useful in real life, because a discharge summary is produced when a patient has fully recovered and has been discharged from the hospital, requiring minimal or no further action.

Structured data contain information such as laboratory test results, demographics, prescribed medication, medical images, etc. On the other hand, narrative clinical texts contain a significant amount of information which cannot be recorded

in structured fields. Examples of such clinical texts include emergency admission notes, radiology reports, and discharge summaries. The aforementioned notes describe the signs and symptoms, physical findings, and other conditions about patients. However, developing a system to retrieve relevant information from clinical texts has proven to be challenging, due to the fragmented nature of texts with high occurrences of misspelling, medical abbreviations, and symbols.

The idea of automated diagnosis is not entirely new and there were previous attempts to tackle this task. However, past research has not been widely accepted and used in clinical settings. One reason is that existing systems rely on only structured data or drop-down boxes, which require medical doctors to spend a significant amount of time to choose the correct descriptions, signs, and symptoms of a patient. Furthermore, most medical diagnosis systems reported to date rely mostly on structured data rather than narrative texts. In this thesis, we focus on a system that utilizes and learns mainly from free-text notes instead of structured data.

2.3 NLP in the Medical Domain

Some studies aim to solve this problem and use adapted NLP techniques to overcome the special characteristics of clinical texts. Such examples include acronym expansion in clinical texts (Joshi et al., 2006), part-of-speech tagging for biomedical texts (Smith et al., 2004), named entity recognition in biomedicine (Ananiadou, Friedman, and Tsujii, 2004), and semantic lexicon for medical language processing (Johnson, 1999).

There are a number of studies which explored the use of various NLP techniques to extract useful information from clinical texts. (Demner-Fushman, Chapman, and McDonald, 2009) reported that NLP techniques are imperative in clinical decision support systems. They illustrated various clinical decision support systems

and demonstrated the use of NLP techniques to extract information from clinical texts. (Roberts et al., 2015) have also developed a system using statistical (machine learning) approach to extract information in clinical texts to identify the risk factors of heart disease. (Deleger et al., 2013) use the pediatric appendicitis score as a basis to develop an automated appendicitis risk stratification system. In our work, we have to combine NLP and machine learning techniques to create a system that is able to learn from free texts and other structured fields to diagnose a patient.

NLP research in medical domain has been more prevalent recently due to the series of challenges organized by i2b2 (Informatics for Integrating Biology and the Bedside). i2b2 is an NIH-funded (National Institutes of Health) national center for biomedical computing (NCBC) based at Partners HealthCare System in Boston, Massachusetts, USA. These challenges encourage researchers to solve existing medical problems using NLP. All of the datasets of the i2b2 challenges are available[2] to the public after registering and signing a set of agreements. Up to today, i2b2 has organized the following challenges: (1) de-identification challenge, (2) smoking challenge, (3) obesity challenge, (4) medication challenge, (5) relations challenge, (6) coreference challenge, (7) temporal relations challenge, and (8) heart disease risk factors challenge. Various methods were presented to tackle the aforementioned challenges: support vector machines (SVM), logistic regression, conditional random fields (CRF), rule-based systems, regular expression pattern matching, and combinations of many machine learning algorithms and classifiers. We will explain each challenge in more details in the remainder of this section.

The **first** challenge is a de-identification challenge (Uzuner, Luo, and Szolovits, 2007; Stubbs and Uzuner, 2015; Stubbs, Kotfila, and Uzuner, 2015), which is equivalent to the anonymization task as described in Section 2.1. The main objective of this challenge is to remove personal health information (PHI) which can lead to

[2] `https://www.i2b2.org/NLP/DataSets/`

the identification of an individual. The de-identification challenge was organized two times: in 2006 and 2014. The **second** challenge is smoking challenge (Uzuner et al., 2008), where participating systems are required to classify a patient into one of the five categories: past smoker, current smoker, either current or past smoker, non-smoker, or unknown. The systems need to classify the patient based on his/her discharge summary narrative text.

The **third** challenge is obesity challenge (Uzuner, 2009), which requires systems to classify a discharge summary into one of the following classes: obesity present, absent, questionable, or unmentioned. There are two sub-categories of this challenge: to classify based on textual evidence, and to classify by intuition. The intuitive systems are much more complex due to the need to guess and infer from many other indirect features. The **fourth** challenge is medication challenge (Uzuner et al., 2010; Uzuner, Solti, and Cadag, 2010), where participating systems are supposed to detect all types of medications, their dosages, modes (routes) of administration, frequencies, durations, and reasons for administration in discharge summaries. This task is similar to named entity recognition (NER) task, and it is more challenging than the previous tasks due to the fragmented nature of the texts.

The **fifth** challenge is relations challenge (Uzuner et al., 2011). The challenge consists of three parts: extraction of medical concepts from discharge summaries; assignment of assertion types for medical problem concepts; and relation classification, assigning relation types that hold between medical problems, tests, and treatments. This challenge is tough because the system is expected to perform three processing steps as mentioned above. The **sixth** challenge is coreference challenge (Uzuner et al., 2012). As the name suggests, the task is to perform noun phrase coreference resolution in medical records. This task is identical to the standard NLP coreference resolution task other than the dataset being used, which are discharge summaries.

The **seventh** challenge is temporal relations challenge (Sun, Rumshisky, and Uzuner, 2013a; Sun, Rumshisky, and Uzuner, 2013b). The task targets clinically significant events (e.g., clinical concepts, tests, treatments, etc), temporal expressions (e.g., dates, movements within the hospital, etc), and temporal relations between clinical events and temporal expressions. The systems are required to detect the first two entities (clinical events and temporal expressions), and then produce a link/relation to connect the two. The **eighth** and last challenge is a heart disease risk factor challenge (Stubbs and Uzuner, 2015; Stubbs et al., 2015). This task required systems to detect medical risk factors related to coronary artery disease (CAD) in the narrative medical records of diabetic patients. The risk factors include hypertension, hyperlipidemia, obesity, smoking status, family history, and diabetes. This task is similar to NER, where the named entities are the aforementioned risk factors. The final goal is to perform analysis on the detected risk factors and their correlation with heart diseases.

2.4 Neural Networks

The recent improved performance of neural network models on various tasks and domains, ranging from image classification, speech recognition, to machine translation, has resulted in a resurgence of neural networks. This is attributed mainly to the availability of huge datasets for training (e.g., ImageNet and Wikipedia) and rapid advances in hardware (in particular, graphics processing units (GPU)).

Convolutional neural networks (CNN): CNNs (LeCun et al., 1989) have been very successful in the task of image classification, resulting in numerous network architectures such as AlexNet (Krizhevsky, Sutskever, and Hinton, 2012), ZF-Net (Zeiler and Fergus, 2014), VGG-Net (Simonyan and Zisserman, 2015), and GoogLeNet (Szegedy et al., 2015). Neural network architectures for image classification mainly consist of convolution layers, rectified linear units (ReLU) as activation

units, and max pooling. The aforementioned studies proposed different ways of constructing a convolution neural network, ranging from a shallower network, to a very deep convolutional neural network because of the increase in performance as the network grows deeper.

However, there was a limit to the depth of the network. In other words, adding more layers did not always improve the performance of the model. This led to the development of a simple but powerful CNN, residual network (ResNet) (He et al., 2016). In ResNet, there is a direct connection connecting two consecutive layers by performing a sum operation (i.e., the input vector of the current layer is added to the output vector of the current layer or equivalently the input vector of the next layer). Not only is CNN superior in image classification, it also achieves state-of-the-art performance on object detection. R-CNN (Girshick et al., 2014) was proposed to tackle object detection and has proven to be very successful. The authors developed two more iterations and improvements to R-CNN, namely Fast R-CNN (Girshick, 2015) and Faster R-CNN (Ren et al., 2015).

In recent years, there were attempts to trick image classification neural network models, by applying tiny distortions to an original image, such that the distorted image looks the same to humans, but the neural network model will misclassify the distorted image. This motivated the development of a new neural network architecture, namely generative adversarial networks (GAN) (Goodfellow et al., 2014). GAN works by producing two models, a generative and a discriminative model. The generative model aims to create distortions to the image to trick the classifier, and the discriminative model aims to determine whether the image is natural or it has been artificially distorted.

Recurrent neural networks (RNN): RNNs (Elman, 1990) have proven successful in sequence prediction tasks, or tasks where the input is in the form of a time series with variable length. There are three well-known instantiations of

RNNs: basic recurrent neural networks (Elman, 1990), gated recurrent neural networks (GRU) (Cho et al., 2014), and long short-term memory networks (LSTM) (Hochreiter and Schmidhuber, 1997). RNNs are suitable for NLP tasks where the input is a sequence of words (with variable length). An RNN is able to read the words sequentially from left to right, remembering only the words that are important. To train a neural network with a sequence of input words, each word needs to be represented by a word vector or word embedding. Continuous word representations where each word is represented by a word vector have proven useful for many NLP tasks such as part-of-speech (POS) tagging, word sense disambiguation, dependency parsing, name entity recognition (NER), and chunking (Turian, Ratinov, and Bengio, 2010; Collobert et al., 2011; Bansal, Gimpel, and Livescu, 2014; Taghipour and Ng, 2015). The word embeddings are usually added as additional features to the current machine learning model to improve its performance. RNNs have also proven successful in various NLP applications, including sentiment classification (Tang, Qin, and Liu, 2015), machine translation (Bahdanau, Cho, and Bengio, 2015), automated essay scoring (Taghipour and Ng, 2016; Alikaniotis, Yannakoudakis, and Rei, 2016), question answering (Kumar et al., 2016), speech recognition (Graves and Jaitly, 2014), and caption generation (Vinyals et al., 2015).

Although the main component of the aforementioned network architectures consists of mainly RNN, there are many modifications and improvements to the network to achieve better accuracy and interpretability (to visualize the the input and output of the network). Examples of these improved architectures include CNN with RNN for automated essay scoring (Taghipour and Ng, 2016), CNN with RNN for adverse drug reaction classification (Huynh et al., 2016), CNN with bi-directional RNN for generating image descriptions (Karpathy and Fei-Fei, 2015), dynamic memory networks (DNM) for question answering (Kumar et al., 2016), and condensed memory networks for clinical diagnostic inferencing (Prakash et al.,

2017).

Inspired by the recent success of attention mechanism (Bahdanau, Cho, and Bengio, 2015; Hermann et al., 2015; Rush, Chopra, and Weston, 2015), we have adopted attention mechanism in our pooling layer. Other than performance improvement, attention mechanism allows us to gain insights into how the neural network assigns weights to the input, in our case, the words in the ED notes. It is very important to be able to visualize the model and show clinicians the important words and the justifications for the neural network decisions.

We have explored various neural network architectures and combinations of CNN, RNN, residual network, and attention. CNN was explored because of its ability to capture n-gram *local context* where n is the window size of the convolution layer. RNN was explored because of its ability to learn a very rich representation of a sequence of words. We will describe our proposed architecture (CR2) in detail in Section 3.5.

Chapter 3

Automated Diagnosis

3.1 Dataset

3.1.1 Data Extraction

The corpus of hospital ED notes used in this project is obtained from the National University Hospital (NUH), spanning a period of ten years. The database containing the data is not indexed, where each patient visit is stored in a long text blob in XML format. Other than the free text notes, ED notes also contain some demographic information of patients, such as their date of birth, address, and gender. Lab results of a hospital visit can also be obtained from another database (laboratory results database). The full details of data extraction and processing are described in (Yuwono, 2016).

We have addressed the privacy issues of using medical records for research. As mentioned in Section 2.1, we have developed a simple and efficient algorithm to anonymize personal health information (PHI) in the free-text notes (Yuwono, Ng, and Ngiam, 2016). Therefore, we can use the medical records (ED notes in particular) for research purposes. The main challenge of this thesis is to create a model which learns from free texts consisting of sentence fragments, bullet points,

medical abbreviations, and misspellings. A sample anonymized ED note snippet from our hospital is shown in Table 3.1.

33/Chinese/M PMHX: - anemia - previously on iron supplement - nil OGD done Currently c/o: epigastric pain 1500H nil nasuea / vomiting nil fever noted nil dysuria / hematuria no changes in bowel movement no LOW/LOA no chest pain or SOB O/E on admission: Pt alert, attentive CVS: PR 78/min, Bp 120/70 S1S2 no murmurs, TWC 14 UC10 - nad soft abdo, normoactivew BS. direct and rebound tenderness RIF. nil guarding. nil rebound Imperssion: Acute appendicitis Pt was sent for op

Table 3.1: A sample anonymized ED note snippet from our hospital

3.1.2 Data Processing

As mentioned in Chapter 1, the true diagnosis of each patient visit to the emergency department is stored in the discharge summary (DS). However, there is no unique identifier linking an ED note to its associated discharge summary. As such, we created some heuristics to match an ED note and its associated discharge summary mainly based on a patient's identification number, admission date, and discharge date (i.e., if the same person is discharged from the emergency department and admitted to the hospital ward within 24 hours, they are considered a match).

Diagnosis description in the ED notes and discharge summaries is in the form of free text. Hence we need to perform mapping from free-text diagnosis description to one of positive, negative, or maybe class. We discussed with a medical doctor to create a comprehensive set of regular expression patterns to perform the aforementioned mapping. After running the regular expression mapping, the final list is manually checked by the medical doctor before it is used in the dataset in our experiments. After processing the data, we have about 180,000 ED notes and DS pairs. Each ED note contains 445 words on average. The statistics of the ED notes are shown in Table 3.2.

Total	181,271 ED notes
Average number of words per ED note	445 words
Average ED note size	~ 2.2 KB
Total ED notes size	~ 400 MB

Table 3.2: Overview of dataset

The dataset setup and preparation for the experiments are explained in detail in Chapter 4.

3.2 Task Description

We formulate the appendicitis diagnosis task as a binary classification problem. Given a free-text ED note, and optional real-valued features (from the structured fields), the model is required to classify the instance as positive appendicitis (represented with a `1`), or negative appendicitis (represented by a `0`). This is accomplished by producing a probability score, and comparing the score against a threshold, such that the class is positive if the probability score exceeds the threshold, and negative otherwise.

3.3 Evaluation Metric

The standard evaluation metrics of binary classification are recall, precision, specificity, F_1-score, and $F_{0.5}$-score as shown in Equation 3.1.

$$\begin{aligned} recall &= \frac{TP}{TP+FN} \\ precision &= \frac{TP}{TP+FP} \\ specificity &= \frac{TN}{TN+FP} \\ F_1 &= 2 \times \frac{precision \times recall}{precision + recall} \\ F_{0.5} &= (1+0.5^2) \times \frac{precision \times recall}{(0.5^2 \times precision) + recall} \end{aligned} \tag{3.1}$$

TP, FP, FN, and TN denote true positive, false positive, false negative, and true negative respectively. The positive class refers to class `1` (appendicitis), while the negative class refers to class `0` (not appendicitis). As clinicians favour precision and specificity over recall, we have adopted $F_{0.5}$-score as our main evaluation metric. We aim to have FP as low as possible to prevent patients from being operated on when they do not have appendicitis. Clinicians view FN as more tolerable (as compared to FP), because doctors are still required to investigate the condition of

patients not diagnosed as appendicitis until they recover.

To measure the success of automated diagnosis of appendicitis, we can compare the $F_{0.5}$-score of our system with that of ED doctors. If our system is able to outperform ED doctors, our system will have better predictive ability as compared to medical doctors in the emergency department.

3.4 Baseline Approaches

This section describes two baseline approaches reported in (Yuwono, 2016). To prevent confusion, we have omitted the results in (Yuwono, 2016) due to the difference in the dataset and experimental setup used. We have configured and re-run our models described in this section with our latest dataset. The latest experimental setup and results are described in Chapter 4.

3.4.1 Alvarado Score: A Rule-based Approach

As mentioned in Section 1.2, there is an existing well-known scoring system to diagnose *acute appendicitis*, namely the Alvarado score (Alvarado, 1986). It is also known as MANTRELS, which is a mnemonic to remember the score factors (signs, symptoms, and lab readings). The details of each factor with its assigned score are shown in Table 3.3. The score for each factor present in a patient is added to obtain the final score. A higher score indicates a higher probability of a patient having appendicitis. All of the factors can be found in the free-text ED notes except the last two, which are lab readings. The challenges of this approach include detection of negation. For example, "no fever" means a patient has no fever. Simply searching for the word "fever" would yield the wrong result, treating it as presence of fever. To solve this problem, we have adopted Negex algorithm (Chapman et al., 2001), a simple regular expression rule-based algorithm which has been modified to suit

our needs.

Symptoms	**Point**
Migration of pain to the right lower quadrant	1
Anorexia (loss of appetite)	1
Nausea or vomiting	1
Tenderness in the right lower quadrant	2
Rebound or guarding	1
Elevation of temperature (fever)	1
High Lab reading of **L**eukocytosis (white blood cell count)	2
High Lab reading of **S**hift to the left (Neutrophils)	1

Table 3.3: Alvarado features

Feature extraction: To be able to implement Alvarado scoring, we first have to extract the features from the ED notes. We have tried two methods to extract the features from the free-text ED notes: (1) conditional random fields (CRF) and (2) regular expression patterns. A subset of the ED notes was randomly chosen and the Alvarado features in these ED notes were hand-annotated by a medical doctor. Then we compare the performance of the two methods. Our results showed that regular expression patterns outperform CRF in detecting Alvarado features.

Rule-based scoring: We implemented Alvarado scoring to classify *acute appendicitis*. An Alvarado score ranges from 0 to 10. Scores strictly greater than the chosen threshold will be classified as positive, and negative otherwise. A threshold with the best $F_{0.5}$-score on the validation set is chosen as the final threshold.

3.4.2 Maximum Entropy Modeling

A model which requires manually defined classification rules is not scalable or generalizable to other diseases. Therefore we explored using a machine learning model that is able to learn from texts. We used a maximum entropy (maxent) classifier with bag-of-words features. Maximum entropy modeling has been shown to perform well on text classification tasks (Nigam, Lafferty, and McCallum, 1999). The advantage of using a maximum entropy classifier with bag-of-words features is that the weight of each word can be obtained and hence we are able to determine which words are important in diagnosing *appendicitis*. Each ED note is first tokenized, and negation is detected through the Negex algorithm (Chapman et al., 2001). We obtain the bag of words of each ED note, concatenate them with the lab results and other structured fields, and then use them as features for the maxent classifier.

3.5 Our Proposed Neural Network Architecture

We have created a novel neural network architecture named convolutional residual recurrent neural network (CR2). Our architecture is illustrated in Figure 3.1.

Lookup Table Layer: The first layer of our neural network projects each word into a d_{LT} dimensional space. Given a sequence of words $\mathbf{W}$ represented by their *one-hot* representations $(\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M)$, the output of the lookup table (LT) layer is calculated by Equation 3.2.

$$\begin{aligned} LT(\mathbf{W}) &= (\mathbf{E}\mathbf{w}_1, \mathbf{E}\mathbf{w}_2, ..., \mathbf{E}\mathbf{w}_M) \\ &= (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M) \end{aligned} \tag{3.2}$$

where $\mathbf{E}$ is the word embedding matrix which is learnt during training and M is the number of words in an ED note.

Convolution Layer: After the dense representation of the input sequence is computed from the lookup table layer, it is fed as the input to a convolution

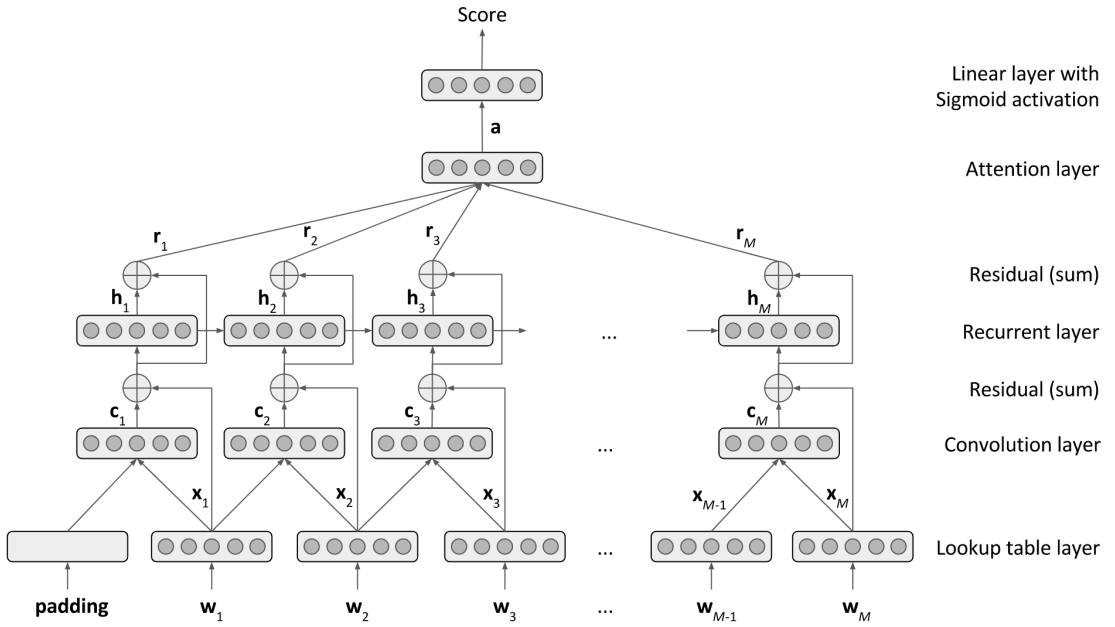

Figure 3.1: Our neural network architecture (CR2)

layer to extract *local features*. Given a window of word representations of length l, (i.e., $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_l$), they are first concatenated to form vector $\bar{\mathbf{x}}$, and then an output vector $\mathbf{C}$ of length d_c is computed as shown in Equation 3.3.

$$\mathbf{C} = \mathbf{W}_v \bar{\mathbf{x}} + \mathbf{b} \tag{3.3}$$

$\mathbf{W}_v$ and $\mathbf{b}$ are the trainable weight and bias parameters respectively, and they are shared across all windows in a sequence.

Residual Layer: We perform the sum operation on the sequence of word embeddings ($\mathbf{X} = \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M$) and the output of the convolutional layer ($\mathbf{C} = \mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_M$) as shown in Equation 3.4.

$$\text{Sum}(\mathbf{X}, \mathbf{C}) = \mathbf{X} + \mathbf{C} \tag{3.4}$$

To be able to perform the sum operation as shown above, the dimension of the word embedding (d_{LT}) and the dimension of the output vectors of the convolution layer (or the number of filters) (d_c) have to be equal.

Recurrent Layer: After combining *local features* extracted by the convolution layer with the original dense word representations, the resulting vectors are fed as an input to a recurrent layer. The recurrent layer processes the input to generate a representation of a given ED note. There are three well-known RNN instantiations: basic recurrent neural networks (Elman, 1990), gated recurrent units (GRU) (Cho et al., 2014), and long short-term memory networks (LSTM) (Hochreiter and Schmidhuber, 1997). Based on our experimental results, LSTM outperforms the other two instantiations of RNN and hence we only describe LSTM as our RNN unit.

LSTM is able to learn to preserve or forget information. To control the flow of information, LSTM uses three gates to forget or pass the information to the next time step. The formal definition of LSTM is described in Equation 3.5.

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_i\mathbf{x}_t + \mathbf{U}_i\mathbf{h}_{t-1} + \mathbf{b}_i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_f\mathbf{x}_t + \mathbf{U}_f\mathbf{h}_{t-1} + \mathbf{b}_f) \\
\tilde{\mathbf{c}}_t &= \tanh(\mathbf{W}_c\mathbf{x}_t + \mathbf{U}_c\mathbf{h}_{t-1} + \mathbf{b}_c) \\
\mathbf{c}_t &= \mathbf{i}_t \circ \tilde{\mathbf{c}}_t + \mathbf{f}_t \circ \mathbf{c}_{t-1} \\
\mathbf{o}_t &= \sigma(\mathbf{W}_o\mathbf{x}_t + \mathbf{U}_o\mathbf{h}_{t-1} + \mathbf{b}_o) \\
\mathbf{h}_t &= \mathbf{o}_t \circ \tanh(\mathbf{c}_t)
\end{aligned} \tag{3.5}$$

$\mathbf{x}_t$ is the input vector at time t, which is the vector representation of the t^{th} word in an ED note. LSTM produces one vector $\mathbf{h}_t$ at each time step t ($\mathbf{h}_0$ is the zero vector). $\mathbf{W}_i, \mathbf{W}_f, \mathbf{W}_c, \mathbf{W}_o, \mathbf{U}_i, \mathbf{U}_f, \mathbf{U}_c, \mathbf{U}_o$ are weight matrices and $\mathbf{b}_i, \mathbf{b}_f, \mathbf{b}_c, \mathbf{b}_o$ are the bias vectors. The circle symbol $\circ$ denotes element-wise multiplication and σ denotes the sigmoid function. The output of the recurrent layer is $\mathbf{H} = (\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_M)$. Following (Taghipour and Ng, 2016), we use every output of the intermediate states of the RNN and perform summing (residual) and then pooling in the next layer to have a better representation of the entire ED note.

Residual Layer: We perform the sum operation on the sequence of the output vectors from the recurrent layer ($\mathbf{H} = \mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_M$) and the output vectors of the previous residual layer ($\text{Sum}(\mathbf{X}, \mathbf{C})$) as shown in Equation 3.6.

$$\mathbf{R} = \text{Sum}(\mathbf{H}, \mathbf{X} + \mathbf{C}) = \mathbf{H} + \mathbf{X} + \mathbf{C} \tag{3.6}$$

To be able to perform the sum operation as shown above, the dimensions of the word embedding vectors (d_{LT}), output vectors of the convolution layer (d_c), and output vectors of the hidden RNN layer (d_r) have to be equal.

Attention layer: Visualizing the learned model is of high importance in the medical domain. By using an attention mechanism, we can show the degree of importance of words and phrases. Attention mechanism has been successful in many recent studies (Bahdanau, Cho, and Bengio, 2015; Hermann et al., 2015; Rush, Chopra, and Weston, 2015). The outputs of the previous residual layer $\mathbf{R} = (\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_M)$ are used as inputs of the attention layer. In other words, this layer receives M vectors of size d_r, where d_r is the output dimension of the recurrent layer. $\mathbf{R}$ is a rich representation of each word in the ED note using a combination of word embedding, CNN output, and RNN output. Each of the vectors $\mathbf{r}_t$ is multiplied by a learnable real-valued weight s'_t between 0 and 1 before adding the elements of all M vectors into a single vector $\mathbf{a}$ as a form of *weighted average*. The functions of the attention layer are defined in Equation 3.7.

$$\begin{aligned} s_t &= \mathbf{v} \cdot \tanh(\mathbf{W}_r \mathbf{r}_t) \\ s'_t &= \text{softmax}(\mathbf{s})_t \\ \mathbf{a} &= \sum_{t=1}^{M} s'_t \mathbf{r}_t \end{aligned} \tag{3.7}$$

$\mathbf{W}_r$ is a trainable matrix of size $d_r \times d_r$ and $\mathbf{v}$ is a trainable vector of size d_r. To learn more complex functions, $\mathbf{W}_r$ is introduced to increase the number of parameters and tanh is introduced to add non-linearity. $\mathbf{W}_r$ and $\mathbf{v}$ are *shared* across all time

steps t. To make sure that the weights for all time steps sum to 1, softmax function is performed on all the weights $\mathbf{s} = (s_1, s_2, ..., s_M)$. The attention mechanism is illustrated in Figure 3.2.

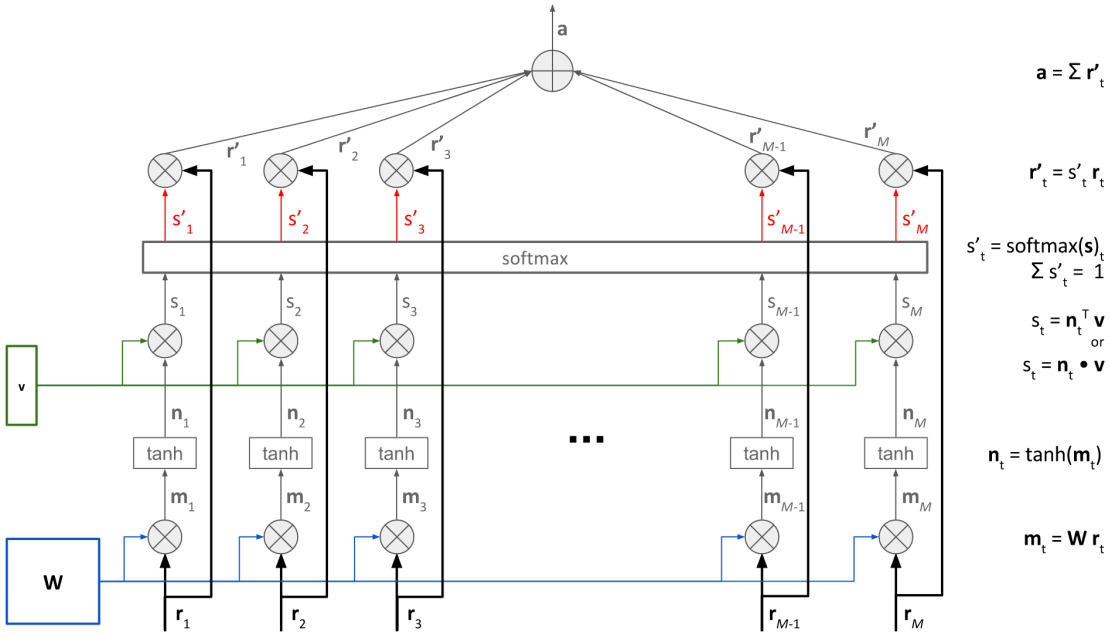

Figure 3.2: The attention layer components

The attention layer is able to learn to assign varying weights to different time steps t depending on the input $\mathbf{h}_t$. The main advantage of having an attention layer is that we can retrieve the weight s'_t (highlighted in red in Figure 3.2) for each time step, and hence we are able to visualize and measure the importance of each word in the ED note.

Linear Layer with Sigmoid Activation: If there are no additional real-valued features, the input of this layer is the vector $\mathbf{a}$. Otherwise, it will be $[\mathbf{a}, \mathbf{l}]$, the concatenation of $\mathbf{a}$ and $\mathbf{l}$, where $\mathbf{l}$ contains the additional real-valued features which will be described in the next subsection. The linear layer maps the input vector into a single scalar value. This mapping is a simple linear transformation, therefore the computed scalar value is unbounded. Since we are expected to predict

either class `0` or `1`, we will use a sigmoid function to ensure the scalar value is in the range of $(0, 1)$. The mapping of the linear layer after applying the sigmoid function is shown in Equation 3.8.

$$s(\mathbf{x}) = \sigma(\mathbf{w} \cdot \mathbf{x} + b) \tag{3.8}$$

where $\mathbf{x}$ is the input vector $\mathbf{a}$ or $[\mathbf{a}, \mathbf{l}]$, $\mathbf{w}$ is the weight vector, and b is the bias value.

3.5.1 Additional Real-valued Features

Before using additional real-valued features such as lab results in the neural network, the values need to be normalized. We have adopted `normal sigmoid` to normalize the real-valued features which is shown in Equation 3.9. $\bar{x}$ and σ represent the mean and standard deviation for a particular feature (e.g., white blood cell count).

$$\begin{aligned} \text{normal}(x) &= \frac{(x - \bar{x})}{\sigma} \\ \text{normal\_sigmoid}(x) &= \frac{1}{1 + e^{-\text{normal}(x)}} \end{aligned} \tag{3.9}$$

There are also entries where ED notes are not accompanied by any lab results. To deal with missing values, we calculate the mean ($\bar{x}$) of all existing entries in the training set of that particular feature (e.g., white blood cell count) and then use the average value to fill in the gap.

In order to include the L real-valued normalized features $\mathbf{l} = (l_1, l_2, ..., l_L)$ in the model, we concatenate L real numbers (after normalizing them) to the output of the attention layer, before going into the next layer. The input of the final layer will be $[\mathbf{a}, \mathbf{l}]$, a vector of size $d_r + L$. Figure 3.3 illustrates the process above.

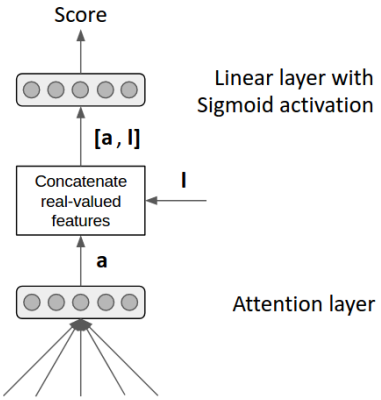

Figure 3.3: Concatenation of real-valued features before the final layer

3.5.2 Training

We use the RMSProp optimization algorithm (Dauphin, de Vries, and Bengio, 2015) to minimize a loss function over the training data. Given N training ED notes and their corresponding true class s_i^* (either `0` or `1`), the model computes the predicted score s_i in the range of $(0, 1)$ for all training ED notes and then update the network weights such that the loss function is minimized. The loss function we have adopted in our system is the binary cross-entropy loss function as shown in Equation 3.10.

$$L(\mathbf{s}, \mathbf{s}^*) = -\sum_{i=1}^{N} s_i^* \log(s_i) + (1 - s_i^*) \log(1 - s_i) \tag{3.10}$$

In our data set, the distribution of the classes is highly imbalanced. The proportion of ED notes in class `0` can be as high as 98.4%, with the remaining 1.6% ED notes in class `1`. To tackle this problem, we have adopted a *weighted* binary cross-entropy loss function shown in Equation 3.11.

$$L(\mathbf{s}, \mathbf{s}^*) = -\sum_{i=1}^{N} c_1 s_i^* \log(s_i) + c_0 (1 - s_i^*) \log(1 - s_i) \tag{3.11}$$

c_0 and c_1 are the weights assigned to the two classes `0` and `1` respectively. In order to balance the two classes, each class is weighted *inversely proportional* to the class

frequencies in the training data to allocate more weights on the less frequent class. Equation 3.12 describes how to obtain the class weights c_0 and c_1.

$$\begin{aligned} c_0 &= N/2f_0 \\ c_1 &= N/2f_1 \end{aligned} \tag{3.12}$$

where N is the number of ED notes in the training set, f_0 (f_1) is the number of ED notes in class `0` (`1`). This approach is similar to the technique used by (Chollampatt, Taghipour, and Ng, 2016) for rescaling.

To prevent overfitting, we have adopted dropout regularization (Srivastava et al., 2014). We also clip the gradient if gradient norm is larger than a certain threshold. We do not utilize early stopping methods in our experiments. We train the neural network for a specified number of epochs and evaluate the model on a validation set in every epoch. The epoch with the highest $F_{0.5}$-score on the validation set is then selected as the final model.

3.5.3 Threshold Adjustment

The output or score of the neural network is a real value between 0 and 1. However, we need to transform the score to either `1` (positive) or `0` (negative) to solve our binary classification problem. Therefore, there is a need to set a threshold as the decision boundary. The default threshold used to split the two classes is 0.5. For example, if the prediction score is greater than 0.5, then the predicted class is positive (appendicitis); otherwise negative (not appendicitis).

The aforementioned threshold can be used to tune the model to have lower FP but higher FN, and vice versa. In this thesis, we would like to achieve the lowest possible FP, trading for a higher FN. To achieve this, we use the validation set to search for a threshold with the best $F_{0.5}$-score. First, we use the model in the current epoch to predict the score of each instance in the validation set. Second, we sort the validation instances in ascending order of the predicted scores. Third, we

perform a linear search to find the cut-off threshold to achieve the best $F_{0.5}$-score on the validation set. This is repeated in every epoch, resulting in a unique threshold for each epoch. The epoch with the best $F_{0.5}$-score (using its own unique threshold) on the validation set is used as the final model to evaluate the test set using the same threshold which was used in the validation set.

Chapter 4

Experiments

4.1 Setup

Our network has several hyper-parameters which need to be set. We use the RMSProp optimizer with decay rate (ρ) of 0.9 and learning rate of 0.001. Mini-batch[3] size is 32 and the model is trained for 25 epochs. The vocabulary is created using all words in the training set. Out-of-vocabulary words are replaced by a special <unknown> token. Words that contain any digits are replaced by a special <num> token. The network is regularized by using dropout (Srivastava et al., 2014) with probability 0.5. During training, if the norm of the gradient exceeds 10, it will be clipped to a maximum value of 10. Word embedding dimension (d_{LT}), output dimension of the hidden layer for the RNN (d_r), and the number of filters for the CNN (d_c) are set to 300. The convolution window size (l) is set to 3. We initialize the lookup table layer with our custom pre-trained word embeddings, which were trained using our entire corpus of about 180,000 ED notes excluding the notes used as validation and test set. We use the word2vec skip-gram model (Mikolov et al.,

[3]To create mini-batches for training, all the ED notes in a mini-batch are padded using a dummy token to have the same length. To remove the effect of padding tokens during training, they are masked to prevent the network from miscalculating the gradients.

2013) to train our word embeddings. Although the lookup table layer is initialized with pre-trained word embeddings, the lookup table layer is trainable and not fixed. We utilized 4 additional features from the structured patient data, namely age, gender, and two lab test results (white blood cell count and neutrophils), and incorporate them into the network as described in Section 3.5.1.

4.2 Dataset

Diagnosis description in the ED notes and discharge summaries is in the form of free text. Hence we need to perform mapping from free-text diagnosis description to one of positive, negative, or maybe class. We discussed with a medical doctor to create a comprehensive set of regular expression patterns to perform the aforementioned mapping. After running the regular expression mapping, the final list is manually checked by the medical doctor before it is used in the dataset in our experiments.

Using some rules to match ED notes and discharge summaries, we have about 180,000 ED and DS pairs. The distribution of the ED notes in the entire corpus is shown in Table 4.1 (second and third column). The first class (symbol) listed in the first column is the class predicted by ED doctors in the ED notes, while the second class (symbol) listed in the first column is the true diagnosis class obtained from the discharge summaries. All of the ED notes used for training and validation in all our experiments have their impressions automatically removed unless stated otherwise. Instances with maybe diagnosis in either an ED note or a discharge summary are excluded in all of our experiments. The class distribution of the dataset after removing the maybe diagnosis is shown in the last two columns of Table 4.1.

Class	**# ED notes**	**%**	**# ED notes**	**%**
++ / TP	2,194	1.204 %	2,194	1.2 %
+− / FP	1,071	0.588 %	1,071	0.6%
−+ / FN	796	0.437 %	796	0.4 %
−− / TN	177,210	97.29 %	177,210	97.8 %
+?	22	0.012 %	-	-
−?	19	0.010 %	-	-
?+	325	0.178 %	-	-
?−	506	0.277 %	-	-
??	5	0.003 %	-	-
Total	182,148	100 %	181,271	100 %

Table 4.1: Class distribution of ED notes

Class	**Dataset 1**		**Dataset 2**		**Dataset 3**	
	# ED notes	**%**	**# ED notes**	**%**	**# ED notes**	**%**
++ / TP	216	1.2 %	734	36.7 %	734	36.7 %
+− / FP	104	0.6 %	6	0.3 %	36	1.8 %
−+ / FN	78	0.4 %	266	13.3 %	266	13.3 %
−− / TN	17,709	97.8 %	994	49.7 %	964	48.2 %
Total	18,107	100 %	2,000	100 %	2,000	100 %

Table 4.2: Class distribution of ED notes in test sets

Most of the ED notes contain ED doctors' impression (i.e., ED doctors' diagnosis description). The impressions in the free-text notes need to be removed to prevent the model from biasing towards assigning high weights to the ED doctors' initial diagnosis instead of learning from the signs and symptoms in the ED notes. Most of the ED doctors write their impressions in a certain format. We have developed a comprehensive set of regular expression patterns to automatically remove such impressions from the ED notes. To ensure that the test dataset does not have any trace of impressions which might benefit the model, we adopted manual removal by two independent medical doctors. Based on 750 ED notes with double annotations, their agreement is very high, with Kappa value of 0.813. The Kappa value is calculated with each line (terminated by a newline character) in an ED note as the matching unit. The annotators are required to remove an entire line as long as it contains any hint of impression.

4.2.1 Dataset 1: Natural Distribution (Original Dataset)

Using the corpus shown in Table 4.1, we randomly sample 10% for training, 10% for validation, and 10% for test. The number of ED notes is 18,111, 18,108, and 18,107 respectively following its natural class distribution (about 1.6% positive ED notes). To speed up training, we only use ED notes with 750 words or less in the training set, resulting in 16,854 instead of 18,111 ED notes for training. We do not pose any length limit for both the validation and test set. Impressions in the test set of dataset 1 are automatically removed due to the large size of the test set. The class distribution of the test set is shown in Table 4.2.

4.2.2 Dataset 2: Equal Class Distribution with Random Negative ED Notes

In our second dataset, we obtain a subset of the 181,271 ED notes (from Table 4.1) to create a dataset with 50% positive and 50% negative ED notes. There are 2,980, 1,000, and 2,000 ED notes for training, validation, and test respectively with equal distribution of positive and negative classes in each set. The negative ED notes consist of randomly sampled negative ED notes of all diagnosis classes that are not appendicitis. In dataset 2, the ED doctor's impressions in the test set have been *manually* removed by a medical doctor. The class distribution of the test set is shown in Table 4.2.

4.2.3 Dataset 3: Equal Class Distribution with Abdominal-related Negative ED Notes

Our third dataset is very similar to our second dataset (in Section 4.2.2) with the same class distribution. The only difference is that the negative ED notes in this dataset only consist of abdominal-related diagnosis instead of any random diagnosis that is not appendicitis. ED doctors' impressions in the test set have also been *manually* removed by a medical doctor. The number of ED notes in the training, validation, and test set are the same as those in dataset 2. The 1,000 positive ED notes in this test set are identical to the 1,000 positive ED notes in the test set in dataset 2. This was done to reduce manual work to remove ED doctors' impressions. Dataset 3 is more challenging than dataset 2 because the signs and symptoms of appendicitis are very similar to those of other abdominal conditions. The class distribution of the test set is shown in Table 4.2.

Set	Model	TP	FP	FN	TN	FP+FN	Rec	Prec	Spec	F1	F05	Acc
(1)	ED	216	104	78	17,709	182	0.735	0.675	0.994	0.704	0.686	0.990
(1)	Maxent	138	126	156	17,687	282	0.469	0.523	0.993	0.495	0.511	0.984
(1)	Alvarado	124	90	170	17,723	260	0.422	0.579	0.995	0.488	0.539	0.986
(1)	Best CR2	141	90	153	17,723	243	0.480	0.610	0.995	0.537	**0.579***	0.987
(1)	Avg CR2	154.8	109.2	139.2	17,703.8	248.3	0.527	0.588	0.994	0.553	0.573	0.986
		±16.9	±18.8	±16.9	±18.8	±8.3	±0.058	±0.021	±0.0011	±0.030	±0.016	±0.00046
(2)	ED	734	6	266	994	272	0.734	0.992	0.994	0.844	0.927	0.864
(2)	Maxent	952	62	48	938	110	0.952	0.939	0.938	0.945	0.941	0.945
(2)	Alvarado	617	12	383	988	395	0.617	0.981	0.988	0.758	0.877	0.803
(2)	Best CR2	912	27	88	973	115	0.912	0.971	0.973	0.941	**0.959***	0.943
(2)	Avg CR2	912.1	28.6	87.9	971.4	116.4	0.912	0.970	0.971	0.940	0.958	0.942
		±17.1	±6.1	±17.1	±6.1	±13.7	±0.017	±0.0058	±0.0061	±0.0076	±0.0037	±0.0069
(3)	ED	734	36	266	964	302	0.734	0.953	0.964	0.829	0.900	0.849
(3)	Maxent	880	125	120	875	245	0.880	0.876	0.875	0.878	0.876	0.878
(3)	Alvarado	617	72	383	928	455	0.617	0.896	0.928	0.731	0.821	0.773
(3)	Best CR2	831	79	169	921	248	0.831	0.913	0.921	0.870	**0.895***	0.876
(3)	Avg CR2	832.1	84.2	167.9	915.8	252.1	0.832	0.908	0.916	0.868	0.892	0.874
		±28.8	±12.2	±28.8	±12.2	±19.0	±0.029	±0.0096	±0.0122	±0.0125	±0.0045	±0.0095

Table 4.3: Summary of the best model against ED doctors and the baselines on three datasets. The baseline for the statistical significance tests is underlined and statistically significant improvements ($p < 0.05$) are marked with '*'

4.3 Results and Discussions

The experimental results of the best model (CR2) on the three datasets are summarized in Table 4.3.

The first column shows the dataset used in the evaluation: (1) refers to dataset 1 (Section 4.2.1). (2) refers to dataset 2 (Section 4.2.2). (3) refers to dataset 3 (Section 4.2.3). We train the neural network model (end-to-end) on a single GPU (Nvidia TITAN X Pascal), and the training time is 3.2 hours for dataset 1, and 35 minutes for each of the datasets 2 and 3. After the model is trained, it is able to perform acute appendicitis diagnosis rapidly, at 400 ED notes per second. The *best* single CR2 model is chosen based on the highest $F_{0.5}$-score in the validation set over 50 runs with different seeds. The *average* score for the CR2 model in each column is calculated over 50 runs with different seeds. The $\pm$ sign represents the standard deviation over the 50 runs.

We have two baselines methods, namely a maxent (maximum entropy, also known as logistic regression) classifier and an Alvarado rule-based scoring system. This is inspired by prior work (Deleger et al., 2013) which performs appendicitis risk stratification using an Alvarado rule-based scoring system with features obtained from free text. The aforementioned two methods have been adopted in our previous work (Yuwono, 2016) and explained in Section 3.4. The maximum entropy classifier utilizes bag-of-words representation for an ED note, with lab results and structured fields (age, gender) as additional features. The Alvarado scoring scheme requires a threshold to classify each patient. Different threshold values (scores strictly greater than the threshold will be classified as positive, and negative otherwise) were explored and the threshold with the best $F_{0.5}$-score was chosen. The thresholds for Alvarado scoring in datasets 1, 2, and 3 are 6, 5, and 5 respectively.

Our neural network model (CR2) outperforms the two baselines in $F_{0.5}$-score on all three datasets. We also perform a statistical significance test ($p < 0.05$)

to determine whether the obtained improvement is statistically significant. We found that our neural network improvements against maxent on all datasets **are statistically significant**. This shows that our neural network model is superior to the maxent classifier and Alvarado scoring system.

Based on the first row in Table 4.3, we can see that ED doctors' performance is better compared to our model. This is mainly caused by class imbalance (1.6% positive and 98.4% negative). Learning and predicting on a dataset with extremely skewed class distribution is challenging. However, as we can see from the results, the performance of our best model is close to that of ED doctors, with 14 fewer FP instances and only 75 more FN instances out of 18,107 ED notes in the test set.

Based on the results of datasets 2 and 3, our model achieved lower FP+FN (in other words, higher accuracy) when compared to ED doctors. With equal distribution of positive and negative ED notes, our model performs better than ED doctors with much lower FN in exchange for slightly higher FP. Our model's $F_{0.5}$-score exceeds that of ED doctor on dataset 2 and is very close to that of ED doctor on dataset 3. Our model also consistently achieves better sensitivity (recall) than the ED doctor.

To visualize the model and gain insights into how the model assigns importance to words and phrases, we retrieve the weights of the attention layer. The weights can be used to show the degree of importance of words and phrases in an ED note. From our observation, the model is able to pick up meaningful signs and symptoms of appendicitis most of the time. Figure 4.1 shows the visualization of our model, with appendicitis features highlighted, such as RIF (right iliac fossa) pain, and tenderness with rebound. In Figure 4.1, darker shade of red color indicates a higher weight assigned to a word. These signs and symptoms have been validated and used in practice as features of Alvarado scoring scheme (Alvarado, 1986).

On the other hand, the model is also able to pick up the features of non-appendicitis. In Figure 4.2, the model is able to pick up diarrhea and a few other features suggesting non-appendicitis.

We will explore other neural network architectures and more (deeper) layers in the future. We will also design our experiments to be able to fully utilize the entire 180,000 ED notes to train and validate our model.

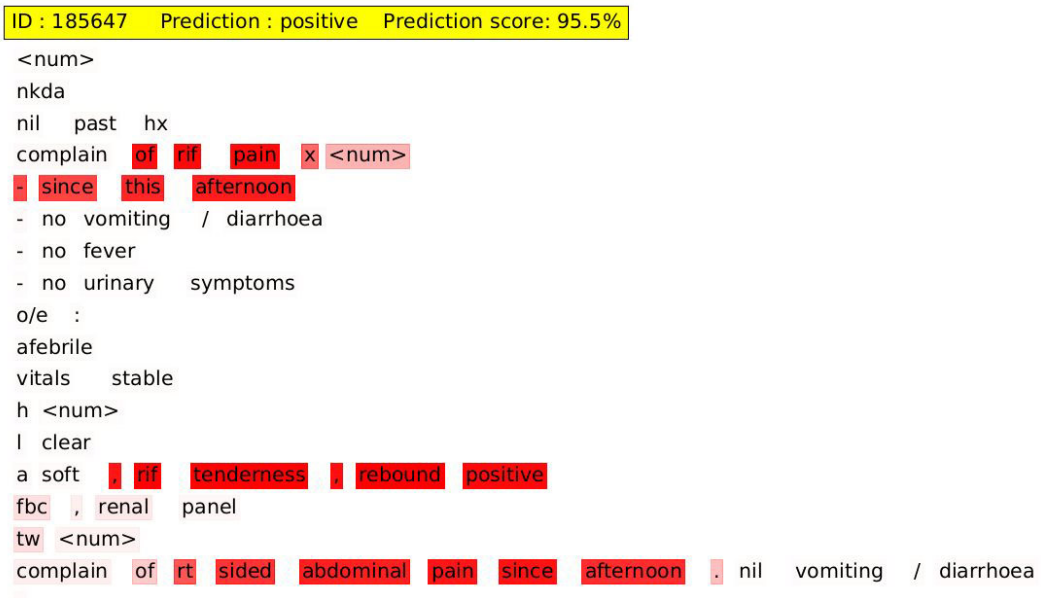

Figure 4.1: Visualization of how our model interprets a positive ED note

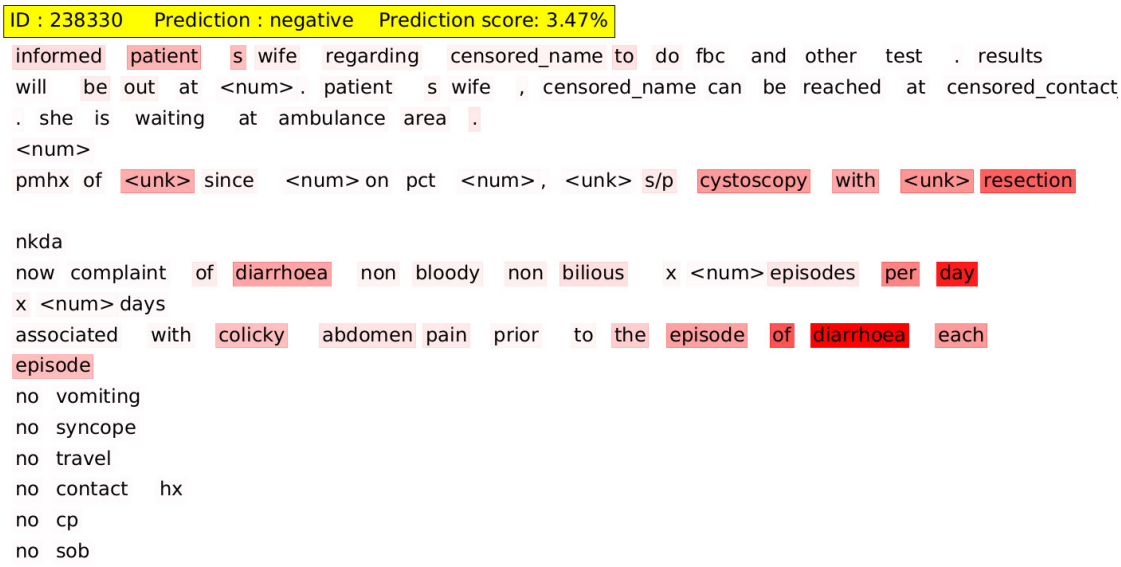

Figure 4.2: Visualization of how our model interprets a negative ED note

Chapter 5

Conclusion and Future Work

In this thesis, we have tackled the task of automated diagnosis using free-text ED notes. We present a neural network model which is able to learn from free texts and optional additional features without any feature engineering. We show that the performance of our novel neural network architecture is promising and close to the performance of ED doctors. Analysis of the visualization shows that the attention layer is able to meaningfully learn the importance of words and phrases in ED notes and to change its emphasis depending on the context of the words. This is helpful in highlighting certain key description (i.e., signs and symptoms) that might have been missed otherwise by medical doctors in a real-life setting.

Overall, our system is expected to increase the efficiency and accuracy (i.e., less misdiagnoses) of doctors, which will lead to a decrease in waiting time and healthcare cost. Furthermore, it can save human lives because appropriate actions can be performed more quickly and accurately. Lastly, there will be more efficient use of hospital resources (e.g., ward rooms, scan rooms, operating rooms, etc) if patients are directed to the correct departments and treatments from the beginning of their hospital visits. This will have a great impact especially in the context of Singapore, where healthcare resources are very scarce.

Next, we plan to incorporate more information from the structured fields (e.g., more lab tests results, scan/radiology images, etc), aiming to further improve the accuracy of our model. We would also like to generalize our model to be able to diagnose multiple abdominal-related diseases, beyond just appendicitis. We can also explore solving other medical and clinical problems such as predicting future diseases with respect to time (e.g., heart attacks or diabetes), based on the past medical history of a patient.

References

Alikaniotis, Dimitrios, Helen Yannakoudakis, and Marek Rei. 2016. Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Alvarado, Alfredo. 1986. A practical score for the early diagnosis of acute appendicitis. *Annals of Emergency Medicine*, 15(5):557 – 564.

Ananiadou, Sophia, Carol Friedman, and Junichi Tsujii. 2004. Introduction: Named entity recognition in biomedicine. *Journal of Biomedical Informatics*, 37(6):393–395.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*.

Bansal, Mohit, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*.

Cao, Chunshui, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, Deva Ramanan, and Thomas S. Huang. 2015. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision*.

Chapman, Wendy W, Will Bridewell, Paul Hanbury, Gregory F Cooper, and Bruce G Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, 34(5):301–310.

Cho, Kyunghyun, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase

representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Chollampatt, Shamil, Kaveh Taghipour, and Hwee Tou Ng. 2016. Neural network translation models for grammatical error correction. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*.

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Dauphin, Yann N., Harm de Vries, and Yoshua Bengio. 2015. Equilibrated adaptive learning rates for non-convex optimization. In *Advances in Neural Information Processing Systems 28*.

Deleger, Louise, Holly Brodzinski, Haijun Zhai, Qi Li, Todd Lingren, Eric S Kirkendall, Evaline Alessandrini, and Imre Solti. 2013. Developing and evaluating an automated appendicitis risk stratification algorithm for pediatric patients in the emergency department. *Journal of the American Medical Informatics Association*, 20(e2):e212–e220.

Demner-Fushman, Dina, Wendy W Chapman, and Clement J McDonald. 2009. What can natural language processing do for clinical decision support? *Journal of Biomedical Informatics*, 42(5):760–772.

Douglass, M, GD Clifford, A Reisner, GB Moody, and RG Mark. 2004. Computer-assisted de-identification of free text in the MIMIC II database. In *Computers in Cardiology 2004*.

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

Esteva, Andre, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter,

Helen M Blau, and Sebastian Thrun. 2017. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118.

Evans, R Scott, Stanley L Pestotnik, David C Classen, Terry P Clemmer, Lindell K Weaver, James F Orme Jr, James F Lloyd, and John P Burke. 1998. A computer-assisted management program for antibiotics and other antiinfective agents. *New England Journal of Medicine*, 338(4):232–238.

Girshick, Ross. 2015. Fast R-CNN. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*.

Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*.

Graves, Alex and Navdeep Jaitly. 2014. Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the 31st International Conference on Machine Learning*.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*.

Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28*.

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

Huynh, Trung, Yulan He, Alistair Willis, and Stefan Rueger. 2016. Adverse drug reaction classification with deep neural networks. In *Proceedings of the 26th International Conference on Computational Linguistics*.

Johnson, Stephen B. 1999. A semantic lexicon for medical language processing. *Journal of the American Medical Informatics Association*, 6(3):205–218.

Joshi, Mahesh, Serguei Pakhomov, Ted Pedersen, and Christopher G Chute. 2006. A comparative study of supervised learning as applied to acronym expansion in clinical reports. In *Proceedings of American Medical Informatics Association Annual Symposium*.

Karpathy, Andrej and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*.

Kumar, Ankit, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. 2016. Ask me anything: Dynamic memory networks for natural language processing. In *Proceedings of the 33rd International Conference on Machine Learning*.

LeCun, Yann, Bernhard Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne Hubbard, and Lawrence D. Jackel. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551.

Liu, Jiang, Zhuo Zhang, Damon Wing Kee Wong, Yanwu Xu, Fengshou Yin, Jun Cheng, Ngan Meng Tan, Chee Keong Kwoh, Dong Xu, Yih Chung Tham, et al. 2013. Automatic glaucoma diagnosis through medical imag-

ing informatics. *Journal of the American Medical Informatics Association*, 20(6):1021–1027.

McKay, Robert and Jessica Shepherd. 2007. The use of the clinical scoring system by Alvarado in the decision to perform computed tomography for acute appendicitis in the ED. *The American Journal of Emergency Medicine*, 25(5):489–493.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*.

Nigam, Kamal, John Lafferty, and Andrew McCallum. 1999. Using maximum entropy for text classification. In *Proceedings of IJCAI 1999 Workshop on Machine Learning for Information Filtering*.

Petroianu, Andy. 2012. Diagnosis of acute appendicitis. *International Journal of Surgery*, 10(3):115–119.

Prakash, Aaditya, Siyuan Zhao, Sadid A Hasan, Vivek V Datla, Kathy Lee, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2017. Condensed memory networks for clinical diagnostic inferencing. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*.

Rao, Patrick M, James T Rhea, Robert A Novelline, Amy A Mostafavi, and Charles J McCabe. 1998. Effect of computed tomography of the appendix on treatment of patients and use of hospital resources. *New England Journal of Medicine*, 338(3):141–146.

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28*.

Roberts, Kirk, Sonya E Shooshan, Laritza Rodriguez, Swapna Abhyankar, Halil Kilicoglu, and Dina Demner-Fushman. 2015. The role of fine-grained anno-

tations in supervised recognition of risk factors for heart disease from EHRs. *Journal of Biomedical Informatics*, 58:S111–S119.

Rush, Alexander M., Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Simonyan, Karen and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations*.

Smith, L, Thomas Rindflesch, W John Wilbur, et al. 2004. Medpost: A part-of-speech tagger for BioMedical text. *Bioinformatics*, 20(14):2320–2321.

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

Stensmo, Magnus and Terrence J Sejnowski. 1996. Automated medical diagnosis based on decision theory and learning from cases. In *Proceedings of the 1996 World Congress on Neural Networks*.

Stubbs, Amber, Christopher Kotfila, and Özlem Uzuner. 2015. Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task track 1. *Journal of Biomedical Informatics*, 58:S11–S19.

Stubbs, Amber, Christopher Kotfila, Hua Xu, and Özlem Uzuner. 2015. Identifying risk factors for heart disease over time: Overview of 2014 i2b2/UTHealth shared task track 2. *Journal of Biomedical Informatics*, 58:S67–S77.

Stubbs, Amber and Özlem Uzuner. 2015. Annotating risk factors for heart disease in clinical narratives for diabetic patients. *Journal of Biomedical Informatics*, 58:S78–S91.

Sun, Weiyi, Anna Rumshisky, and Ozlem Uzuner. 2013a. Annotating temporal information in clinical narratives. *Journal of Biomedical Informatics*, 46:S5–S12.

Sun, Weiyi, Anna Rumshisky, and Ozlem Uzuner. 2013b. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Szarvas, György, Richárd Farkas, and Róbert Busa-Fekete. 2007. State-of-the-art anonymization of medical records using an iterative machine learning framework. *Journal of the American Medical Informatics Association*, 14(5):574–580.

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*.

Taghipour, Kaveh and Hwee Tou Ng. 2015. Semi-supervised word sense disambiguation using word embeddings in general and specific domains. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics*.

Taghipour, Kaveh and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Tang, Duyu, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Turian, Joseph, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

Uzuner, Özlem. 2009. Recognizing obesity and comorbidities in sparse data. *Journal of the American Medical Informatics Association*, 16(4):561–570.

Uzuner, Özlem, Andreea Bodnari, Shuying Shen, Tyler Forbush, John Pestian, and Brett R South. 2012. Evaluating the state of the art in coreference resolution for electronic medical records. *Journal of the American Medical Informatics Association*, 19(5):786–791.

Uzuner, Özlem, Ira Goldstein, Yuan Luo, and Isaac Kohane. 2008. Identifying patient smoking status from medical discharge records. *Journal of the American Medical Informatics Association*, 15(1):14–24.

Uzuner, Özlem, Yuan Luo, and Peter Szolovits. 2007. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

Uzuner, Özlem, Tawanda C Sibanda, Yuan Luo, and Peter Szolovits. 2008. A de-identifier for medical discharge summaries. *Artificial Intelligence in Medicine*, 42(1):13–35.

Uzuner, Özlem, Imre Solti, and Eithon Cadag. 2010. Extracting medication information from clinical text. *Journal of the American Medical Informatics Association*, 17(5):514–518.

Uzuner, Özlem, Imre Solti, Fei Xia, and Eithon Cadag. 2010. Community annotation experiment for ground truth generation for the i2b2 medication challenge. *Journal of the American Medical Informatics Association*, 17(5):519–523.

Uzuner, Özlem, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Vinyals, Oriol, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show

and tell: A neural image caption generator. In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition*.

Wellner, Ben, Matt Huyck, Scott Mardis, John Aberdeen, Alex Morgan, Leonid Peshkin, Alex Yeh, Janet Hitzeman, and Lynette Hirschman. 2007. Rapidly retargetable approaches to de-identification in medical records. *Journal of the American Medical Informatics Association*, 14(5):564–573.

Yuwono, Steven Kester. 2016. Medical text mining. Bachelor of Computing final year project report, National University of Singapore.

Yuwono, Steven Kester, Hwee Tou Ng, and Kee Yuan Ngiam. 2016. Automated anonymization as spelling variant detection. In *Proceedings of the COLING 2016 Workshop on Clinical Natural Language Processing*.

Zeiler, Matthew D and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *Proceedings of 2014 European Conference on Computer Vision*.