Structural Design of Multimodal Medical Encoder for Physician's Diagnostic Support

A Dissertation
Presented to the Graduate School of Informatics
Kyoto University
in Candidacy for the Degree of
Doctor of Philosophy

by
Ryo Otsuki
Kyoto University, Japan

Supervisor: Prof. Tomohiro Kuroda

February, 2022

Abstract

Structural Design of Multimodal Medical Encoder for Physician's Diagnostic Support

Ryo Otsuki
Kyoto University, Japan
2022

Physicians diagnose diseases by comprehensively interpreting multimedia inspection data such as microbiological and other physical and chemical sample test data, clinical imaging data, and text data, often contained in hospital information systems. Physicians require deep clinical knowledge to diagnose their patients' abnormal state by interpreting such multiple information collectively in a short period of time. Clinical decision support systems (CDSS) could support physicians and reduce their burden borne in diagnosis by providing information to facilitate diagnosis and recommendations to help avoid or prevent oversight. In recent years, with the development of deep learning technology, many studies have been conducted in the medical field aiming to substitute automated deep learning models for diagnosis and identification of lesion sites. However, few studies have investigated the use of multimedia clinical data by CDSS. Moreover, deep learning models developed in foregoing studies force physicians to preprocess multimedia clinical data to make them legible for deep learning models, and consequently, to have sufficient knowledge on deep learning technologies. To promote the development of CDSS, it is essential to design and develop methodologies of deep learning models that can handle multimedia medical data. In addition, a system that can generate a model automatically without specialist knowledge by the user would be very useful.

As a definition in this paper, the proposed multimodal medical encoder (MME) is a deep learning model that uses the same data as would a physician to make a diagnosis and predict prognosis. In this study, a methodology to design the MME is proposed. Additionally, an automatic encoder generation system is also proposed.

The proposed design method could be roughly divided into a data input part, a preprocessing and feature extraction part, a feature integration part, and an output part. The preprocessing and feature extraction parts are designed to perform data preprocessing based on a unified standard for each data. For image data, we proposed to use a vision transformer (ViT) that can handle image data regardless of the size and shape of an input image. For electronic medical records (EMR), we designed an preprocessing layer consisting of a batch normalization (BN) layer and a fully connected (FC) layer that can perform

linear transformation while maintaining the data relationships for each batch. Features of multimedia medical data are integrated without bias by an integration layer that adjusts the number of dimensions of features obtained from image data and EMR data into the same number and integrates them. Finally, the output part interprets the integrated clinical features, such as diagnostic or prognostic results. In addition, based on the proposed design method, this study developed an automation design system of an appropriate deep learning model out of given the combination of multimedia clinical input and objective.

Verification of the proposed design was divided into three parts: preprocessing and feature extraction, feature integration, and automatic generation system. In the verification process we predicted postoperative visual acuity of age-related macular degeneration (AMD) as a case-study. EMR and fundus, and optical coherence tomography (OCT) images were used for verification. The dataset for verification included 315 data samples from patients who were diagnosed with wet AMD at Kyoto University Hospital macular clinic and completed a regimen of intravitreal injection of aflibercept for one year. In verifying the preprocessing and feature extraction part, the prediction errors of the proposed model with preprocessing and integration layers and the baseline model with manual preprocessing were compared. As a result, the prediction error of the proposed model was 0.054 and the prediction error of the baseline model was about 0.052. The result showed the prediction error of the proposed model and the baseline model were similar, which revealed that the proposed design method generated preprocessor which has equivalent performance as preprocessing performed by specialists' hands. In verifying the feature integration part, three models were prepared to verify the effectiveness of the proposed design: a model that uses only EMR as input, a model that uses only image data as input, and a model that uses both as inputs. In the experiment, this study compared the prediction errors of postoperative visual acuity. As a result, the prediction error was 0.081 for the model that inputs only the image data, 0.052 for the model that inputs only the EMR, and 0.047 for the model that uses both as inputs. This result suggests that more accurate support of physicians' diagnosis and prognosis can be achieved by integrating and interpreting multimedia medical data.

In verifying the automatic generation system this study confirmed the system could generate an appropriate model according to the given input data and purpose. In verifying the system, we applied data used to diagnose age related macular degeneration. As a result, a structure with BN layer and FC layer for array type numerical data, a design with BN layer and FC layer for dictionary type numerical data, and a structure with ViT for image data were generated. In addition, when the purpose is diagnosis, the softmax layer is generated as an output layer, and when the purpose is prognosis prediction, the

dense layer is generated as an output layer. The result showed that a model to which the design prepared for each data was applied was generated, and the model had an output layer according to the selected purpose.

From these results, it was shown that the proposed design of this study can handle data for diagnosing AMD with a unified standard, and can integrate medical data in clinical practice without bias.

It is expected that the results of this research will enable users to create prototypes of diagnostic support systems using deep learning models and analyze diseases using data of their choice. This result will lead to shorter turnaround time in development of CDSS and the improvement of medical safety.

Contents

1 Introduction **1**

2 MME Design Strategy **7**

2.1 CDSS development process 7

2.2 Physician's Diagnosis 8

2.2.1 Medical Tests and Data 8

2.2.2 Physician's Diagnosis Process 10

2.2.3 Diagnosis by Ophthalmologists 16

2.3 Examination of MME Design 17

2.3.1 Factors Required for MME 17

2.3.2 Conceptual Design 19

2.4 Study Aim 20

3 Embedding Preprocessing into MME **23**

3.1 Interpretation of Medical Data Used for Diagnosis 23

3.1.1 How to handle medical data of physicians 23

3.1.2 Embedding Physician's Data Interpretation 24

3.1.3 Conceptual Design for Auto Preprocessing 24

3.2 Proposed Design 25

3.2.1 Bias Applied by Manual Preprocessing 25

3.2.2 Preprocessing Layer for EMR 26

3.2.3 Preprocessing Layers for Image 27

3.2.4 Combination of Layers 31

3.3 Evaluation 31

3.3.1 AMD 31

3.3.2 Evaluation Task 33

3.3.3 Data for Evaluation 35

3.3.4 Experimental Conditions 36

3.3.5 Results 38
3.3.6 Discussion 40
3.4 Summary 41

4 Integrating Multimodal Medical Features 43
4.1 Interpretation of Medical Data by Physicians 43
4.2 Conceptual Design for Embedding Diagnosis Process 44
4.3 Proposed Design 45
4.3.1 Bias on Extracted Features 45
4.3.2 Feature Integration Layers for EMR 45
4.3.3 Feature Integration Layers for Image 46
4.3.4 Combination of Layers 48
4.4 Evaluation 50
4.4.1 Evaluation Task 50
4.4.2 Data for Evaluation 53
4.4.3 Results 54
4.4.4 Discussion 54
4.5 Summary 55

5 Auto Generation of Multimodal Medical Encoder 57
5.1 Input of Multimodal Medical Encoder 58
5.1.1 Input type 58
5.1.2 Layer Component Assignment corresponding to Input Types 58
5.2 Conceptual Design for Auto Generation System 59
5.2.1 Conceptual Flow of Auto Generation 61
5.2.2 Interpretation of input data 62
5.2.3 Interpretation with the model purpose 63
5.3 Auto Generation of Multimodal Medical Encoder 64
5.3.1 Implementation of Multi-Layer Perceptron 65
5.3.2 Implementation of Vision Transformer 66
5.3.3 Implementation of EMREncoder 69
5.4 Evaluation 71
5.4.1 Evaluation task 71
5.4.2 Data for evaluation 71
5.4.3 Results 71
5.4.4 Discussion 75
5.5 Summary 75

6 Discussion **77**

6.1 Use of the Proposed Design Method for Eye Diseases 77

6.2 Effect of Proposal Design Method on Disease Diagnosis 78

6.3 Effectiveness of Proposed Design Method on CDSS Development 81

6.4 Effectiveness of Deep Learning in the Medical Field 82

6.5 Limitations 83

7 Conclusion **85**

7.1 Contributions 87

List of Figures

1.1 History of deep learning model and application in the medical field 5

2.1 Research overview 8
2.2 Flow chart for headache diagnosis 13
2.3 Flow chart for pulmonary thromolism diagnosis 14
2.4 Flow chart for stroke diagnosis 15
2.5 Diagnosis process of eye disease 18
2.6 Design requirements 20
2.7 Proposal overview 21

3.1 Overview of preprocessing 26
3.2 Overview of Vision Transformer 29
3.3 Flow of extracting patches 30
3.4 Summary of AMD 32
3.5 Overview of evaluation model about preprocessing design 33
3.6 Overview of hyper parameter tuning models 34
3.7 Sample images used in this study 35
3.8 Results of K-means clustering 39
3.9 Results of hierarchical clustering 39

4.1 Overview of integrated model 46
4.2 Preprocessing of patient profile data 47
4.3 Flow differences between one-hot encoding and embedding 48
4.4 Flow differences between CNN and DenseNet 49
4.5 Overview of Concatenate layer 50
4.6 Overview of evaluation model about integration design 52
4.7 Sample of input image and preprocessing 53

5.1 Overview of auto generation system 60

5.2 Graph Model generated by Vision Transformer Algorithm 68
5.3 Graph Model generated by EMR Encoder Algorithm 70
5.4 Overview of generated models 74

6.1 Categorization of data for AMD diagnosis 79
6.2 Categorization of data in medical field 80

Acknowledgments

Acknowledgement Ryo Otsuki

Kyoto, February 9, 2022

First of all, I would like to express my deepest gratitude to Prof. Tomohiro Kuroda of the Division of Medical Informatics, for accepting to be my supervisor. He always giving excellent advice and offering his expertise and guidance with utmost kindness.

With equal emotion and sincerity, I also thank my advisor, Prof. Osamu Sugiyama, who not only offered his knowledge and experience as a teacher, but also gave me his support and encouragement, walking side by side with me in this challenging path.

I must also send a warm thank you to Prof. Masahiro Miyake, Yuki Mori and Yuta Nakanishi who collaborated with our team from the very beginning of this research, and made all the experiments and evaluations of this work possible. They advised not only the datasets needed to proceed with the research, but also the important findings from the doctor's point of view.

For selflessly donating his time and efforts to advise for my research, and for the feedback he provided, I am grateful to Prof. Takayuki Kanda. His comments and suggestions were essential to make this thesis as best as it could be.

I extend my appreciation to Mr. Luciano Santos. He taught me how to proceed with my research and how to think about it. In addition, he has greatly cooperated with my English work.

I also send a thank you to Kazumasa Kishimoto and Kana Ichikawa, who were a member of the same research group. They consulted me when I was stuck in research.

Equally, I thank to all other staffs of Medical Informatics Lab, including Profs. Masayuki Nambu, Hiroshi Tamura, Genta Kato, Masahiro Yakami, Hiroshi Sasaki, Goshiro Yamamoto, Shusuke Hiragi, Hiroki Ueshima who contributed to the success of this research by offering valuable comments and making me take so many different perspectives when reflecting upon my work, further deepening the discussions I conducted and making them much more fruitful.

A very special thank you is necessary to the hard working Medinfo secretaries, Yoko Hara, Yuko Furusawa, and Kaori Shiomi, without whom absolutely no one in this lab, be them students or professors, would be able to accomplish anything, me specially. I deeply thank them all for always helping us keep track of so many documents and deadlines, and scolding us accordingly whenever necessary.

Last but not least, I would like to thank my family, who always offered me their love and support. To my mother and my pet dog Roy, I will always owe my gratitude and love.

Chapter 1

Introduction

In clinical settings, physicians perform various tests and interviews regarding a patient's symptoms to diagnose the disease, determine the treatment policy and evaluate prognosis [46,64]. Various data are used to make a diagnosis, such as the patient's profile of conditions including blood/biochemical test sample results, CT (computed tomography) and MRI (magnetic resonance imaging) images. Over 100 items may be included in a blood test [1]. Physicians determine a patient's abnormal state by comparing these with other profile data including age and gender. Furthermore, physicians may make a comprehensive diagnosis by comparing biochemical and blood test results with medical imaging data.

The diagnostic process requires physicians to have clinical knowledge of diseases. It is necessary to understand disease symptoms and which tests are required to identify them. If appropriate samples are not obtained and tested, it is difficult to determine if an abnormal state of a patient is caused by a particular disease. There may also be heavy burdens for patients in the process, such as undergoing tests, providing samples, and incurring expensive medical costs. Since physicians have often been trained in specialized fields, such as cardiac surgery, oncology, and a multitude of others, diagnosing illnesses outside of their specialized fields takes them longer with the added burden of consulting diagnostic manuals.

Various CDSS (clinical decision support systems) have been developed and used in recent years. Until now, CDSS has been a system that searches for evidence of potential duplicate treatments by reading patient prescription information, and presents the physician with warnings about contraindications regarding harmful concomitant use of medicines. CDSS is intended to support physicians by enhancing clinical decisions with targeted clinical knowledge, patient information, and other clinical information [61]. A traditional CDSS is an application designed to directly aid clinical decision-making, in which recommendations are presented for physicians to consider when determining diagnosis and treatment. CDSS is primarily utilized at the point-of-care by physicians to augment their knowledge with information and to provide suggestions. With recent improvements in deep learning technologies, the core engines of CDSS include classifiers. Classifiers are programs that access a body of information to return a structured explanation or classification of the patient's illness. These are trained with a large number of clinical datasets. A CDSS receives a set of clinical data about a patient, its classifier uses this data in a process as described above and the CDSS outputs structured information and treatment suggestions. In this study, we propose methodologies to design CDSS core engines that emphasize the preprocessing stage of clinical input data. In precedent studies described in the latter paragraphs, internal processing in the engine was often a black box that did not reveal how the CDSS core engine interpreted the clinical data or how they affected output.

As for existing CDSS, they help prevent physicians from overlooking contraindications and prescribing erroneous treatments by reading patients' pharmacy prescription history and order information and detecting duplicate and/or inappropriate treatments. Raschke et al. developed a system to detect potential ADE (adverse drug events) and alert doctors [47]. Their system compares the patient's pharmacy orders, drug allergies, sample test results, and other information with the doctor's prescription. ADE is detected by preset conditions in the system, such as inappropriate administration of imipenem to patients with renal failure. Alerts include details of possible ADE and how to avoid them. Similar alert systems have been proposed for patients' drug allergies and prescription drugs [35].

Existing CDSS for diagnosis support [3] can be categorized into those that do and those that do not utilize deep learning methods. Towards making a diagnosis, physicians use multimedia medical data in various formats, such as letter-based text, numerical data, and images. They interpret abnormal values in multimedia inspection data and comprehensively diagnose associated disorders. To make an automatic and accurate diagnosis in accordance with this same process, a CDSS should receive multimedia clinical data and combine the data features.

Mcheick et al. proposed a model for diagnosing and predicting COPD (chronic obstructive pulmonary disease) exacerbation using the Bayesian network [37]. Their model inputs sensor data obtained from a context-aware application of the Bayesian network. It is combined with updated daily patient data to predict any deterioration in the patient's medical condition. The model is claimed to have provided opportunities for early medical intervention, reduced hospitalization costs, and alleviated overcrowding in emergency rooms, hospitals, etc. Esteban et al. developed a tool to identify predictors of 30 days to 60 days short-term mortality in emergency patients with COPD, using a decision tree aka CART (classification and regression tree) [12]. Five elements of baseline dyspnea are used to make predictions by CART: cardiac disease, presence of paradoxical breathing or use of accessory inspiratory muscles, age, and Glasgow Coma Scale score. Validation results in patients showed their model achieved 0.835 AUC(area under the curve) (95% CI(confidence interval): 0.783, 0.888) predictive reliability for mortality within 30 days and 0.794 AUC (95% CI: 0.723, 0.865) for mortality within 60 days. They were able to identify some easy-to-determine variables that allow clinicians to classify COPD patients by short-term mortality risk. They concluded the tool provides useful data for establishing appropriate clinical care.

As a significant improvement in deep learning technologies, many studies applied deep learning methods to the medical field for image classification and speech recognition. Figure 1.1 shows the history of deep learning models and studies in the medical field. First, perceptron was proposed in 1958 as a one-layer model. That was the predecessor of neural networks. Despite attempts around the 1960s, multilayer perceptron structures could not be trained efficiently and interest in perceptron temporarily waned. Then in 1985, Rumelhart et al. proposed a gradient descent method and an error backpropagation method to perform efficient training with multilayer perceptron [53]. With those methods, interest in multilayer perceptron was rekindled. Consequently, perceptron studies were pursued in the medical field [6, 69]. Since the latter half of the 2000s, study of neural networks accelerated due to improvements in computer performance called “deep” learning. In the field of image recognition, Krizhevsky et al. proposed a model that remarkably surpassed the accuracy of other algorithms in a 2012 image recognition competition [28]. A derivative image recognition system termed R-CNN (region based-convolutional neural network), which is a model of partial object recognition, was also proposed [20]. With R-CNN, the system could locate and identify an object in an image. R-CNN led to more accurate models such as Faster R-CNN [50], and YOLO (you only look once) [49]. Improvement by these image recognition models dramatically contributed to studies in the medical field. For example, a model for finding a cancer tumor and predicting the degree of infiltration from endoscopic images was proposed [41]. In addition, with 3D-CNN, systems could detect lesions such as ground glass shadows related to COVID-19 from lung CT images [62].

In the field of natural language processing, RNN (recurrent neural network) was adopted to handle time-series data [44], leading to LSTM (long short-term memory) that takes longer-term memory in sequence data [24]. Many efficient models were developed and released for different purposes based on those fundamental structures. The encoder-decoder model was often used in sequence-to-sequence transformation, such as translation [10]. Then, an attention model was proposed that overcame the slow calculation speed of RNN-based models [63]. The attention model was proposed with the notion that all input elements were mapped to elements on the output side. For example, Guo et al. proposed a model to diagnose diseases from contents described in EMR (electronic medical records) by performing language processing using LSTM [23]. In addition, Baumel et al. proposed an attention model that assigned disease labels based on (international classification of diseases) ICD10 from text in the patient's EHR [2].

A deep learning model could classify diseases that were relatively easy to diagnose with over 90% accuracy. Some studies achieved the same level of accuracy as physician made diagnoses. However, many classifiers could only process single inspection data, such as interviews, examinations, or images, and were unable to handle data of multiple types. Few studies investigated multimedia clinical data as input. Further, among conventional classifiers, even though the same clinical data was input, preprocessing criteria differed by classifier design. The structure of the above mentioned deep neural network model may also be changed. In a classifier that handles character / numerical data, the judgment of whether to separate character data into words or characters, the standard of numerical value normalization, and the standard of categorization often differ depending on the classifier design. For example, regarding image classifiers, image shaping and cropping processes differed by classifier design which hinders training. For more efficient training of classifiers, it is essential to develop a methodology to integrate the imbalanced handling of clinical data towards a more unified method for the preprocessing stage as well as for information integration. Furthermore, it is difficult for physicians themselves to develop a classifier because it requires technical knowledge of deep learning. Even if a physician employed a development vendor, it may take years before an effective system could be developed and used. It demands much time to make prototypes and evaluate a system through back-and-forth consultation between physicians and engineers. In order to solve this problem, a support system that automatically generates a DNN model according to the data is required. As an existing research, Google has developed a system that automatically creates a model and trains a given image data on its own cloud service [5]. However, there are few precedent studies in the medical field focusing on cost reduction of MME (Multimodal Medical Encoders) or on physician's modeling and training them.

All of handling a multitude of data for diagnosis, processing the same data for use by different classifiers, and associated high costs of time and finance, make developing a CDSS a challenge in clinical practice.

Classifiers designed in previous studies used different input data and had different purposes according to the study. However, they share the same design to the extent that they extract features from input data, integrate the features, and connect them to an output. The present study applied a deep learning model, which included a CDSS classifier,

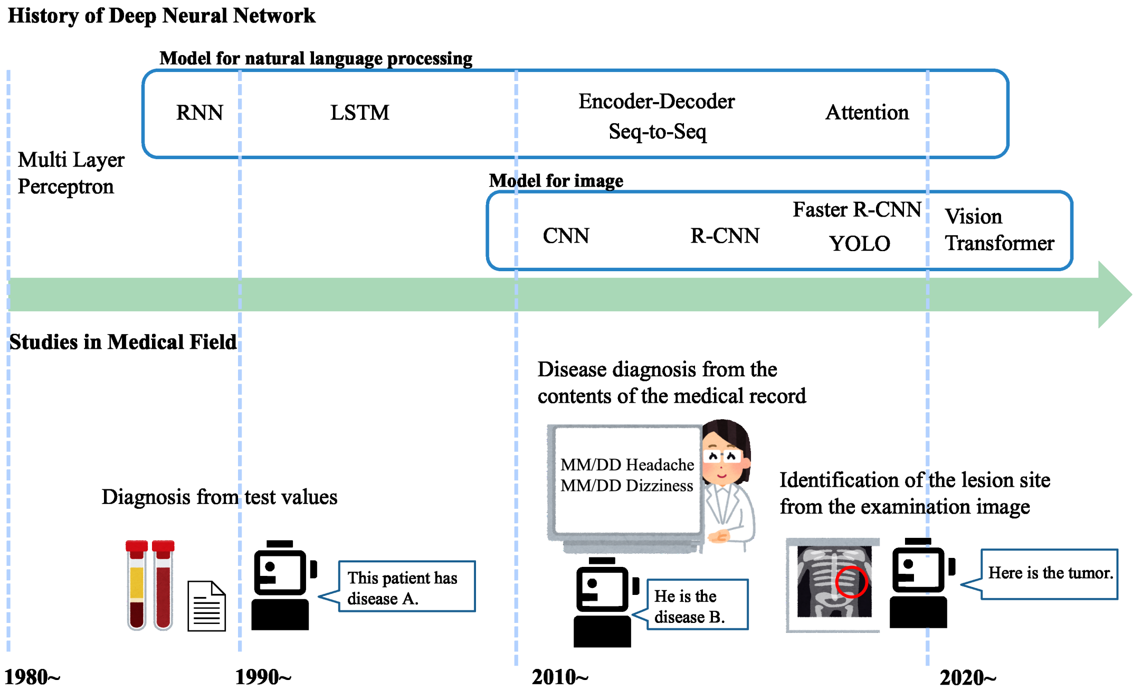

History of Deep Neural Network
Model for natural language processing
RNN
LSTM
Encoder-Decoder
Seq-to-Seq
Attention
Multi Layer
Perceptron
Model for image
CNN
R-CNN
Faster R-CNN
YOLO
Vision
Transformer
Studies in Medical Field
Disease diagnosis from the
contents of the medical record
MM/DD Headache
MM/DD Dizziness
Identification of the lesion site
from the examination image
Diagnosis from test values
This patient has disease A.
He is the disease B.
Here is the tumor.
1980~
1990~
2010~
2020~

Fig. 1.1: History of deep learning model and application in the medical field

to develop an MME.

This study proposes an MME design method. It was applied to receive input of interview and sample test data towards clinical practice diagnosis, to interpret each piece of data according to a unified standard, and to output diagnostic and prognosis prediction results. With our proposed design methods, a system for automatic generation of MME was proposed, which is novel from the viewpoint of supporting physician's studies. The system aims to enable physicians to develop a prototype CDSS without requiring specialized knowledge of preprocessing or neural network model design. All data handled by an actual physician can be interpreted by the MME, the standard for the interpretation is unified, and the output can be selected according to the purpose of either diagnosis or prognosis.

The following structure is required to create the MME: 1. Multimedia medical input, 2. Feature extraction of multimedia medical data, 3. Integration of extracted features, and 4. Output suited to the user's purpose. In this study, we propose the design method of a model that receives multimedia medical data, extracts features from each piece of data, integrates the features, and returns an output including either a diagnosis or a prognosis as determined by the user. It also aims to make redundant the need for manual adjustment to the data and learning process. This prevents the same data being manually processed differently by each model designer, and having different methods for integrating features. Consequently, with our proposed design method, the preprocessing is automatically configured according to input data type. For physicians and engineers who develop CDSS and disease analysis tools themselves, it takes a lot of time to study deep learning methods from scratch. To support their prototyping, this study also proposes an automatic generation system for MME according to purpose.

The proposed design method has a mechanism to receive multimedia medical data to handle all MME data for diagnostics. The mechanism for handling multimedia medical data was created with the following concept. In the data input and feature extraction part, a structure is included to receive input for each type of multimedia data, preprocessing is performed, and features are extracted for each data. It also incorporates a mechanism that automates preprocessing, such as data normalization and clipping. In the feature quantity integration layer for multimedia medical data, the feature quantity extracted for each data is designed to be integrated into the same dimension. The integration mechanism simulates making an integrated diagnosis of each item of test data. By matching the number of dimensions of the feature quantity, the output result of the subsequent stage is prevented from being affected by the difference in the number of dimensions of the input data. The data output layer returns a categorical output if the input purpose is diagnosis or returns a numerical value if the input purpose is prognosis.

The structure of the rest of this paper is as follows: Chapter 2 describes the overall concept of the proposals in this study. We detailed the diagnostic process of physicians and why we aim to generate MME automatically. Chapter 3 describes a design that automates the preprocessing of multimedia medical data with a deep learning model. Chapter 4 proposes a design that equally learns features extracted from multimedia medical data. Chapter 5 describes the automation of developing an MME without specialized clinical knowledge based on the proposed design policies. We discuss the contribution of our proposed methods in Chapter 6. Then finally, Chapter 7 summarizes and concludes this study and our contributions.

Chapter 2

MME Design Strategy

Chapter 2 first describes the development process of a CDSS that assists physicians in diagnosing. Second, it explains the usual diagnostic process of physicians in clinical practice which is supported by CDSS. Third, the diagnosis process in an ophthalmology department is outlined, which is the focus of this study. Fourth, this chapter extracts the processes for general diagnosis after summarizing each diagnosis process. The system requirements for designing the MME are extracted from the generalized process. This study designed the conceptual structure of the MME based on the generalized requirements. Finally, this chapter summarizes the aim of this study and discusses contributions to medical society.

2.1 CDSS development process

CDSS, which supports physicians' diagnoses, extracts features from medical data that is input to the system, and compares the features with characteristic abnormal values associated with diseases to provide diagnoses. CDSS includes a classifier that interprets feature extraction and determines which features to compare. In this study, the classifier is defined as MME. This section describes the MME and CDSS development process.

Figure 2.1 shows the CDSS development process to assist physicians' diagnoses. The development process includes consideration and recognition of significant data, data preparation, model creation, and subsequent creation of the CDSS and UI (user interface). First, in the process of understanding medical data, the physician sets the task of creating a CDSS for a particular disease. After that, the physician organizes the data used for diagnosing the disease and considers which data to enter in the CDSS. Next, in the process of preparing the data, the physician and the system engineer collect the necessary data, label it with a diagnostic label, and preprocess the data. In many cases, the labeling is determined by the physician from observation of the data. On the other hand, preprocessing may be per-

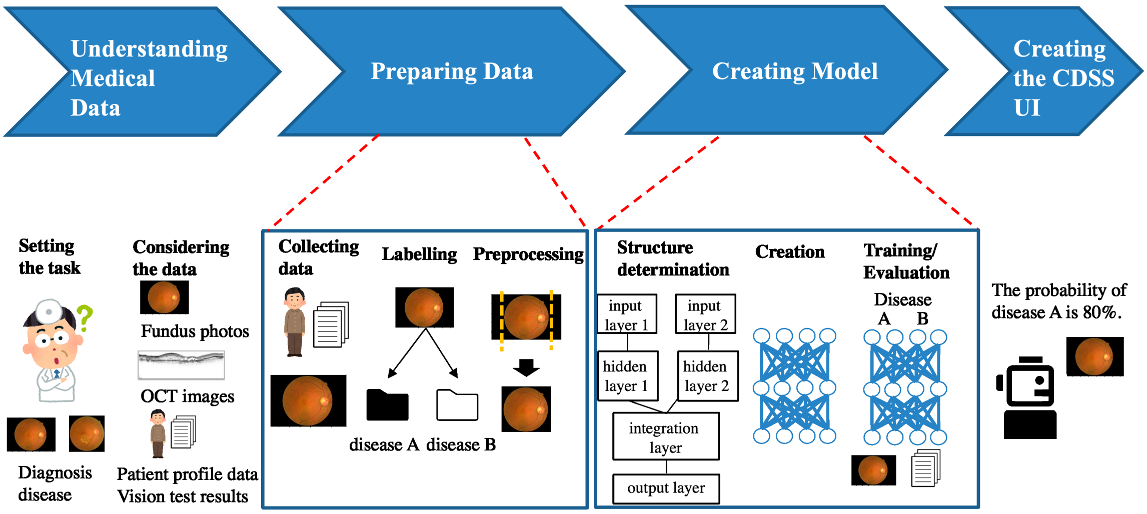


Fig. 2.1: Research overview

formed either manually by the physician or by a systems engineer who has been instructed by the physician on what to look for and how to prepare it. Next, in the model creation process, the systems engineer considers the structure appropriate for the model according to the preprocessed data, makes a prototype, evaluates it, and discusses the evaluation result with the physician. After the classifier is complete, the systems engineer decides the input/output UI and includes it as an application in the process of building the entire CDSS.

The following three points, which have been recognized as issues of concern for a CDSS in previous studies, are included in the preprocessing and model creation process: 1. An inability to handle all the data used for diagnosis, 2. Different processing of the same data for each classifier, and 3. The high costs required to produce a CDSS. This study proposes an appropriate design method for handling multimedia medical data in the preprocessing and model creation process, and proposes an automatic model generation system. With these proposals, we aim to enable development of a prototype CDSS by a user without any special knowledge of deep learning methods.

2.2 Physician's Diagnosis

2.2.1 Medical Tests and Data

Various inspections are performed at medical institutions. Blood and biochemical tests, also called sample tests, measure the number of enzymes and other components of interest in samples of blood and urine. In addition, there are physiological examinations, such as abdominal ultrasonography, and electrocardiogram. Finally, imaging examinations refer to, among others, X-ray fluoroscopy, CT, MRI, and endoscopy.

Blood test results are related to organ conditions. Physicians can diagnose which organ is likely to have an abnormal state based on the sample item value. For example, multiple myeloma or dehydration (abnormalities in the liver) are suspected when the total amount of protein exceeds the standard range of 6.5 to 7.9 g / dL. Undernutrition, liver dysfunction, and nephrotic syndrome are suspected when the total protein value is below the lower bound of the standard range. The concentration of albumin in the blood is also related to nephrotic syndrome. Aspartate aminotransferase and alanine aminotransferase may be used to diagnose fatty liver, hepatitis, and liver cancer. In addition, there are tests for antibodies to check for infection by hepatitis B and C virus, etc.

Abdominal ultrasonography is an imaging test based on ultrasonic waves reflected back from internal organs. The physician has the patient lie supine for abdominal ultrasonography tests. The abdomen is coated with a jelly-like substance then the physician uses a particular tool for the test. Abdominal ultrasonography is used mainly to check the liver, gallbladder, bile duct, pancreas, kidney, spleen, and abdominal aorta. With abdominal ultrasonography, the physician can detect gallbladder polyps in which a mass of cholesterol raises the mucous membrane of the gallbladder. In addition, detection of organ deformations such as gallbladder stones is also possible. Gallbladder stones contains hard black components made of cholesterol and bilirubin calcium.

In an X-ray examination X-ray energy is irradiated through the human body. The radiogram is an image that shows the location and extent of X-ray absorption. Body parts through which X-rays can easily pass are shown in black, and those that are difficult for X-rays to pass through, such as bones, are shown in white. The patient may have an intravenous injection of a contrast medium in advance when organs such as the esophagus, stomach, and intestines are targeted. X-ray examinations can detect shadows in the images. From the shading, the physician can detect diseases such as pneumonia and pneumothorax, cancers of various organs, and bone abnormalities, such as fractures. However, X-ray examinations can take images of the body from only one direction at a time. That often means it is impossible to easily make a detailed diagnosis, as it is with CT and MRI examinations.

A CT examination is an examination in which X-rays are emitted around the body to make an image of its cross-section. The thickness of the cross-section is determined according to the part to be imaged and the purpose of the examination. A contrast medium may be injected intravenously for examination, as with X-ray examinations. A CT examination can reveal the position and size of tumors that have developed in deep parts of an organ, as well as the spread of cancer (metastasis), and the relationship with blood vessels and nerves in more detail than can X-ray examinations. In recent years, chest X-ray images and chest CT images have been used to diagnose and follow-up cases of COVID-19.

In addition to examination data, a basic patient profile is also available. The basic patient profile includes age, gender, and preoperative condition. Physicians collect basic information by interviewing the patient. In these interviews, physicians mainly inquire about the patient's background, the cause of the disease, and the current condition. The patient background includes age and gender, smoking history, and alcohol drinking habit. In the case of traumas, the cause of the disease is correlated with factors regarding the patient's background, such as time when the patient fell or when the accident occurred. Cancer is markedly related to the patient's family history and lifestyle. The patient's current state is assessed in the interview by verifying if there are any subjective symptoms, if there is pain in the affected body part, and how much it hurts. Physicians listen to this information while talking to the patient. Finally, physicians describe exactly the pain expressions that are specific to each patient.

In this way, at medical institutions, multimedia medical data such as text, numerical values, and images are acquired by examinations and interviews. Physicians use these multimedia data as a collection of items of evidence to formulate a comprehensive judgment.

2.2.2 Physician's Diagnosis Process

As examples of the physician's process of making a diagnosis, diagnoses in emergencies of suspected heart and brain diseases based on common symptoms are summarized. These two types of disease are those most commonly found in patients each year [39]. In association with these examples, is a summary of the process of making an emergency diagnosis when acute pulmonary thromboembolism or stroke are suspected [13, 54].

First, the process for emergency diagnosis of circulatory shock is outlined. Shock is a common symptom of heart disease. It is a condition wherein insufficient oxygen is supplied to meet the needs of tissues and cells due to blood circulation failure. In the classic shock classification strategy, the concept of four features defined by Weil et al. in 1971 is still widely used [65]. First, symptoms/signs of shock are considered, including alterations in consciousness which include agitation, restlessness, and decreased responsiveness. In addition, the respiratory rate generally increases, although it depends on the presence or absence of pneumonia and chronic obstructive pulmonary disease. The heart rate usually increases, developing into tachycardia. However, this may differ depending on age, normal heart rate, and time from onset. Blood pressure often drops, and significant hypoxemia, which is reduced oxygen level in the blood, occurs. The state of the jugular vein varies depending on the type of shock. It flattens in hypovolemia shock and dilates with increased right atrial pressure in cardiogenic shock and obstructive shock. The appearance of the skin becomes pale when the vascular resistance increases. The skin temperature becomes normal or warms up when the vascular resistance decreases. Reticular cyanosis, which is reddish-purplish web-like ring patterns, may be exhibited depending on the progress of the shock. Finally, the patient's urine output decreases. Physicians use arterial blood gas analysis, 12-lead ECG (electrocardiogram), and CT examination to discriminate different types of shock. The severity of the shock and treatment responsiveness can be evaluated from the results of lactate level measurement in arterial blood gas analysis. The results of the 12-lead ECG can be used as a basis for determining whether or not the shock is due to heart disease. The CT test results can be used for definitive diagnosis of active bleeding in the body, of lesions of sepsis, of pulmonary thrombosis/deep vein thrombosis, and of aortic lesions. Physicians identify cause of a shock by jointly evaluating the results of the tests above together with the patient's medical history, vital signs, ultrasound findings, and data from other tests.

Next, the process for emergency diagnosis of headaches is summarized. Headache is a common symptom of brain diseases and is divided into primary, secondary, and other headaches [14]. Migraine headaches, tension headaches, and cluster headaches are the main cause of primary headaches. Secondary headaches, in contrast, are caused by urgent and dangerous diseases such as subarachnoid hemorrhage and meningitis. It is essential not to overlook secondary headaches as an emergency diagnostic process. The process for diagnosing headache is shown in Figure 2.2. First, physicians check the onset pattern, exacerbation/remission factor, pain location/presence/absence/associated symptoms, pain intensity/nature, duration, family history, and patient medical history. Interviews are instrumental in determining whether headache treatment is urgent. From these interviews, for example, a sudden headache at a particular moment may be suspected to be a subarachnoid hemorrhage or cerebral artery dissection. If the pain is in the posterior cervical/occipital region, vertebral basilar artery dissection or subarachnoid hemorrhage is suspected. For the first physical examination, physicians check whether vital signs are unstable and evaluate the level of consciousness, eye-opening, spontaneous language, ability to answer questions, and limb movements. Physicians also check for excessive increase in blood pressure, for decrease in blood pressure, for laterality of blood pressure, and for circulatory insufficiency. Then, they will check for anemia, nail bed cyanosis, nutritional status, and trauma. Final checks include distension, induration, thickening, and tenderness of arteries and of superficial blood vessel. Neurological findings can identify headaches associated with intracranial organic lesions. Physicians confirm the presence or absence of focal neurological signs. Intracranial hypertension is strongly suspected if the headache is accompanied by vomiting, elevated blood pressure, bradycardia, and bilateral abduction nerve palsy. If meningeal irritation is accompanied, meningitis and subarachnoid hemorrhage are suspected. A general blood test and inflammatory markers to check for complications of infection were used to classify headaches. Erythrocyte sedimentation rate and CRP (C-reactive protein) increase in temporal arteritis may be observed in a general blood test. Cerebral venous sinus thrombosis is associated with elevated FDP (fibrin degradation products) and D-dimer levels. In arterial blood gas analysis, PaO_2, $PaCO_2$, pH, and bicarbonate ion concentrations are measured. These items are used in diagnosis of headaches due to hypercapnia. Imaging tests such as simple head CT, head MRI / MRA (magnetic resonance angiography), and CT angiography can detect abnormalities in the brain. The physicians check examination images for evidence of bleeding inside the brain, lesions such as in the cortical white matter boundaries, and obscure lenticular nuclei. A cerebrospinal fluid test is performed if necessary to confirm meningitis, encephalitis, or subarachnoid hemorrhage. Meningitis is diagnosed by an increase in the number of cerebrospinal fluid cells. Subarachnoid hemorrhage is diagnosed by the appearance of cerebrospinal fluid showing bloodiness and xanthochromia. Rapid diagnosis of cerebrospinal fluid using latex agglutination reaction is also useful for diagnosis.

Next, the diagnostic process for urgent acute pulmonary thromboembolism and suspected stroke is summarized.

Pulmonary thromboembolism is a disease wherein a thrombus formed in a vein, right atrium, or right ventricle is released and occludes the pulmonary artery acutely. In more than 80% of cases, the source of the embolus is a vein in the lower limbs or pelvis [42]. The mortality rate in the acute phase is higher than that in myocardial infarction, and it is 30% to 50% in cases with unstable hemodynamics due to shock. Among affected patients in cardiac arrest, the mortality rate within 1 h is high. A diagnostic flowchart for acute pulmonary thromboembolism is shown in Figure 2.3. First, typical symptoms are dyspnea, chest pain, fever, fainting wheezing, cold sweat, bloody sputum, and palpitation. In particular, sudden symptoms of dyspnea and chest pain are observed in many patients. Severe cases present with shock. Physical findings include tachypnea, tachycardia, jugular vein distension, and cold limbs. In severe cases, decreased blood pressure and Homans sign [32] is associated with deep vein thrombosis, which causes pulmonary thromboembolism. In addition, auscultation may show an increase in the sound of heart beats, which are associated with pulmonary hypertension. The third and fourth heart beats also increase in volume due to right heart failure. A 12-lead ECG test reveals tachycardia, clockwise

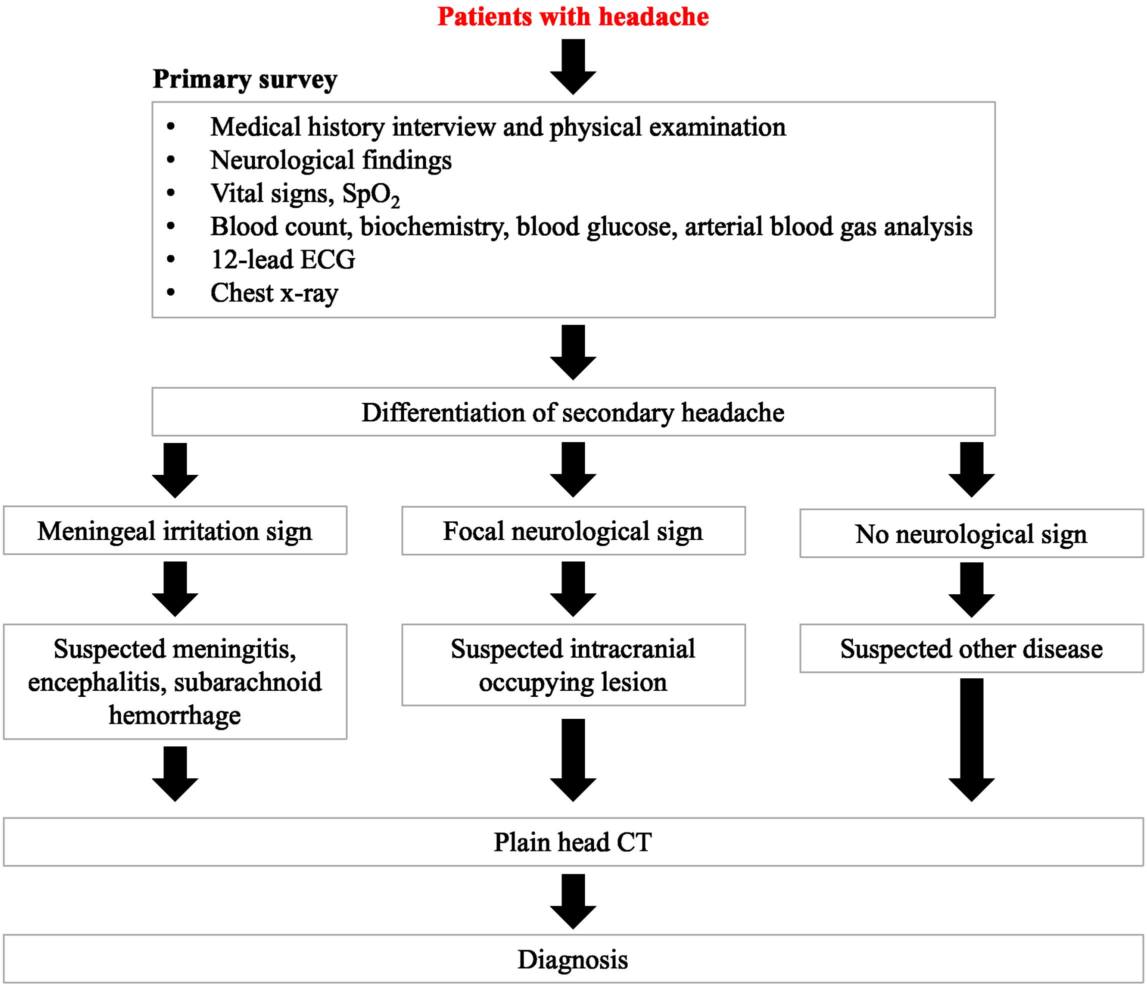

Patients with headache
Primary survey
Medical history interview and physical examination
Neurological findings
Vital signs, SpO2
Blood count, biochemistry, blood glucose, arterial blood gas analysis
12-lead ECG
Chest x-ray
Differentiation of secondary headache
Meningeal irritation sign
Focal neurological sign
No neurological sign
Suspected meningitis, encephalitis, subarachnoid hemorrhage
Suspected intracranial occupying lesion
Suspected other disease
Plain head CT
Diagnosis

Fig. 2.2: Flow chart for headache diagnosis

rotation, and S_1,Q_3,T_3. Blood tests show elevated D-dimer levels. Arterial blood gas analysis shows hypocapnia and hypoxemia. Chest X-ray examination shows an increase in the cardiothoracic ratio, pulmonary artery dilation at the hilar region, and a partial decrease in pulmonary vascular shadow. Transthoracic echocardiography reveals enlargement of the right ventricle, hypokinesis of the free wall of the right ventricle, flattening of the interventricular septum, and exclusion of the left ventricle. The accuracy of diagnosis with contrast-enhanced CT is high, making it particularly useful for diagnosing embolism in the central part of the pulmonary artery. A shadow defect in the image is found on the pulmonary artery in many cases. The thrombus image, which is the source of embolism, is often found in the pelvis or veins of the lower limbs.

In medical treatment for stroke, it is necessary to transport the patient to a medical institution immediately after the onset and take appropriate initial measures. The

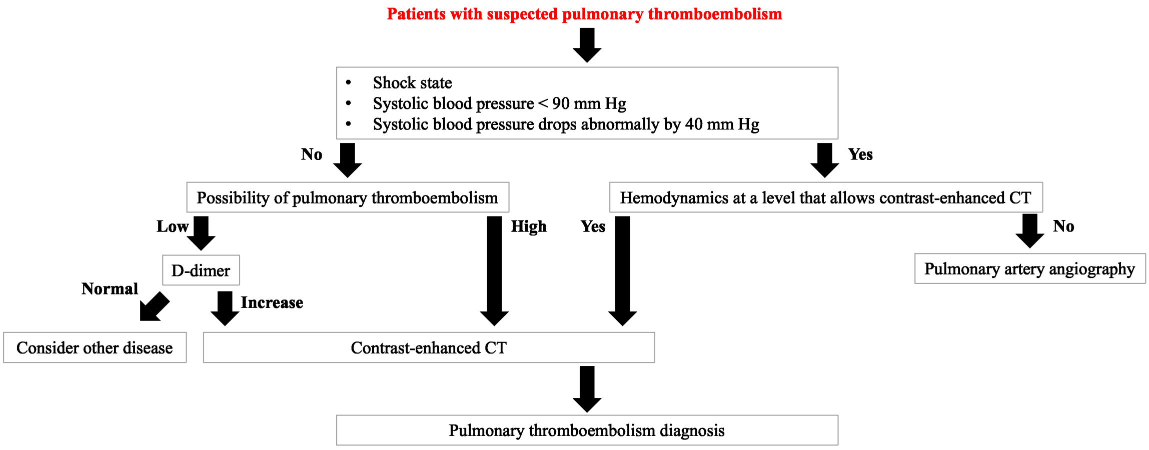

Patients with suspected pulmonary thromboembolism
Shock state
Systolic blood pressure < 90 mm Hg
Systolic blood pressure drops abnormally by 40 mm Hg
No
Yes
Possibility of pulmonary thromboembolism
Hemodynamics at a level that allows contrast-enhanced CT
Low
High
Yes
No
D-dimer
Pulmonary artery angiography
Normal
Increase
Consider other disease
Contrast-enhanced CT
Pulmonary thromboembolism diagnosis

Fig. 2.3: Flow chart for pulmonary thromolism diagnosis

neurological prognosis may be improved if intravenous alteplase therapy or endovascular treatment for hyperacute cerebral infarction is applied immediately after the onset. The stroke diagnosis flowchart is shown in Figure 2.4. First, consciousness disorder, hemiplegia, aphasia, and gait disorder suddenly develop in stroke. General physical signs include elevated blood pressure, bradycardia, carotid artery murmur, heart murmur, arrhythmia. Neurological signs include motor aphasia, sensory aphasia, eye position abnormalities, facial paralysis, cerebellar ataxia of the extremities, and signs of cerebral herniation. Items to be noted in blood tests are blood count, biochemistry (blood glucose, HbAlc, TC, TG, amylase, ammonia, electrolytes), and coagulation ability (PT-INR, APTT, FDP, D-dimer). In addition, 12-lead ECG is used to evaluate arrhythmias such as atrial fibrillation, old myocardial infarction, and cardiomyopathy. Aortic dissection is suspected if there is a mediastinal shadow enlargement on the chest X-ray image. X-ray examination is also used to evaluate pulmonary congestion and aspiration pneumonia. A simple CT scan of the head reveals findings immediately after the onset of a cerebral hemorrhage. In cerebral infarction, early ischemic change may be apparent, including disappearance of the lenticular nucleus structure, disappearance of the insular cortex, and blurring of the epithelial border. The sulcus disappears and gradually shows as a low absorption region after 6 h. DWI (diffusion weighted imaging)-MRI may reveal damage in the head from the early stage of onset. Since lesions are visible after 4 to 5 h in a FLAIR (fluid-attenuated inversion recovery) image, it may be possible to estimate the time of onset by differences between DWI and FLAIR images. In addition to these tests, there is an evaluation standard called Alberta Stroke Program Early CT Score [45]. This evaluation standard divides the middle cerebral artery region into ten regions and evaluates the range of early ischemic change by the origin method. If physicians strongly suspect subarachnoid hemorrhage, they perform a cerebrospinal fluid test because it may be difficult to diagnose after the subarachnoid phase, or there might be slight bleeding, when analyzing a simple CT scan of the head.

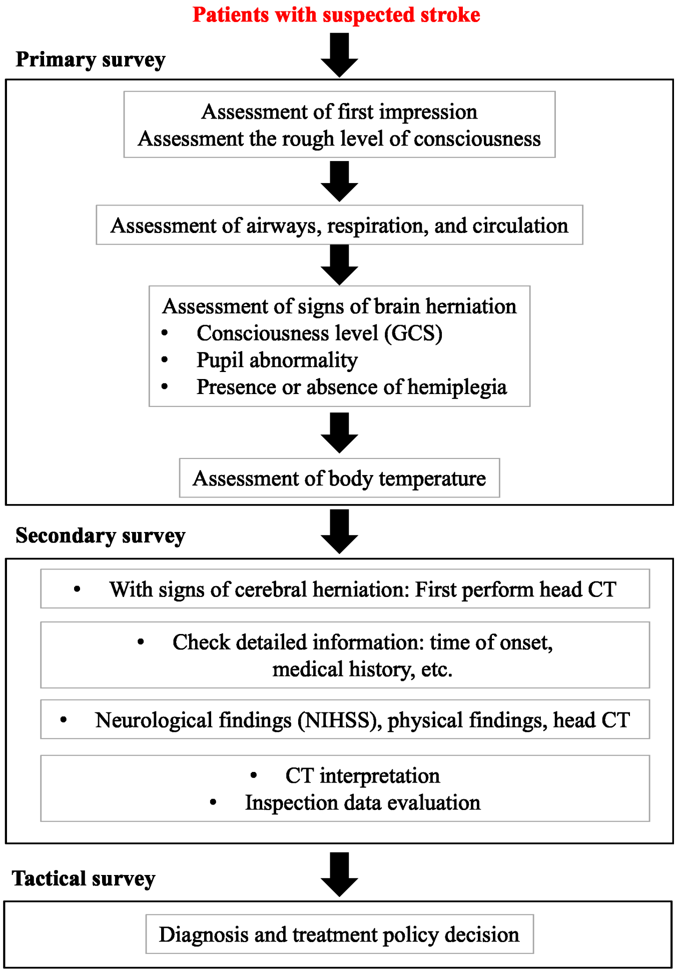

Patients with suspected stroke
Primary survey
Assessment of first impression
Assessment the rough level of consciousness
Assessment of airways, respiration, and circulation
Assessment of signs of brain herniation
• Consciousness level (GCS)
• Pupil abnormality
• Presence or absence of hemiplegia
Assessment of body temperature
Secondary survey
• With signs of cerebral herniation: First perform head CT
• Check detailed information: time of onset, medical history, etc.
• Neurological findings (NIHSS), physical findings, head CT
• CT interpretation
• Inspection data evaluation
Tactical survey
Diagnosis and treatment policy decision

Fig. 2.4: Flow chart for stroke diagnosis

2.2.3 Diagnosis by Ophthalmologists

This research focused on eye diseases as a study case. Many eye diseases lead to blindness if left untreated, but they show few initial subjective symptoms [7]. Improving the quality of each diagnosis with CDSS reduces the possibility of overlooking eye diseases. Figure 2.5 shows the process to diagnose eye diseases. When diagnosing eye diseases, the physician performs a visual field test at the time of the interview, a low vision test, a fundus test, a tonometry test, an OCT (optical coherence tomography) test, and FA/IA (fluorescein angiography/indocyanine green angiography) test. A visual field test is one that measures how wide an area the patient can see with each eye when they focus on a central point, namely a measure of the extent of their central and peripheral vision. Without special equipment, the physician may sit in front of the patient who must look directly ahead while the physician holds up one finger and slowly moves it in the horizontal direction across the patient's visual field. The patient indicates to the physician when the finger is visible and when it is not. The visual field is measured by the physician repeating this several times including for each eye while the non-tested eye is kept closed. In the VA (visual acuity) test, with each eye separately the patient reads a series of rows of Roman alphabet letters on a chart which progressively decrease in size with each row from top to bottom. The physician records the size of the letters (which row) that the patient can read. This is called a Snellen chart. A similar chart called the Landolt C, also known as Landolt ring, is used in Japan. Instead of Roman alphabet letters it contains rows of rings which each have a small segment missing which may be at the top, bottom, right, or left, in the ring. Instead of saying aloud the names of letters in rows that progressively decrease in size, patients are tasked to indicate the site of the missing segment by saying up, down, left, or right. Another chart called the E chart is used for people unable to read Roman alphabet letters, this is used in the same way as the Landolt C but instead of reporting the orientation of the missing segment, patients report the orientation of the limbs of the letter E. In a fundus examination, physicians use an ophthalmoscope to observe the interior of the eye (usually dilated). They check the cornea, lens, vitreous, blood vessels, and retina. In recent years, the fundus camera that can obtain FP (fundus photography) of the fundus has been developed making fundus examination easier to perform. The intraocular pressure test called tonometry measures the pressure in the eye. This test is mainly used to detect glaucoma and confirm the therapeutic effect of glaucoma treatments. The physician first anesthetizes the patient's eye, then, brings the tonometer into contact with the cornea to measure the intraocular pressure. In an OCT test, the physician obtains cross-section images of the retina using light waves. OCT images can reveal the degree of any swelling and range and depth of bleeding. OCT images are used to detect AMD (age-related macular degeneration), retinal vein occlusion, and diabetic retinopathy. The FA/IA test is performed for the purpose of taking clear images of the capillaries of the retina. By injecting a fluorescent contrast agent into the vein of the patient, it is possible to obtain sharp images of the internal eye with a fundus camera. FA tests are used to detect reticulochoroidal diseases such as diabetic retinopathy, AMD, central serous chorioretinal disease, and retinal vein occlusion. On the other hand, the IA test can observe subretinal choroidal blood vessels, which are difficult to detect by the FA test. This test is particularly useful for diagnosing AMD and polypoid choroidal angiopathy.

The process of eye disease diagnosis, just as for other diseases, involves interviews and tests. The collected data has the same format as that in diagnosing other types of diseases, including text recorded in interviews, numerical data of VA tests, and image data of eye inspections.

Based on the analysis of different diagnostic processes described above, the diagnosis flow is broken down into the following four components: 1. Patient interview, 2. Patient examination, 3. Test result interpretation, and 4. Comprehensive diagnosis. First, physicians interview patients face-to-face and record details of their basic profile. In the patient's interview, the physician aims to identify the cause of the disease and understand the current situation. Next, physicians will perform tests in the patient to check for any disease suspected as a result of the interview. The physicians determine which items require analysis and by what method. For example, they may select items to be imaged by the appropriate image inspection method then perform the test. Abnormal findings in the test results are then interpreted. Here, first, physicians observe data features such as abnormal values and images in each inspection item result. Finally, the data obtained is comprehensively interpreted, and a final diagnosis is made.

2.3 Examination of MME Design

2.3.1 Factors Required for MME

On analysis of physicians' diagnoses procedures in both emergency and ophthalmology departments, we defined that diagnosis consists of the following four processes: 1. Patient interview, 2. Patient examination, 3. Test result interpretation, and 4. Comprehensive diagnosis. In 1, the patient's interview, the physician determines the patient's background profile and current disease state face-to-face. Through the interview, they acquire some insight into the patient's disease. The interview process requires ***1a***. Asking the patient appropriate questions, ***1b***. Interpreting the patient's answers, and ***1c***. Predicting the

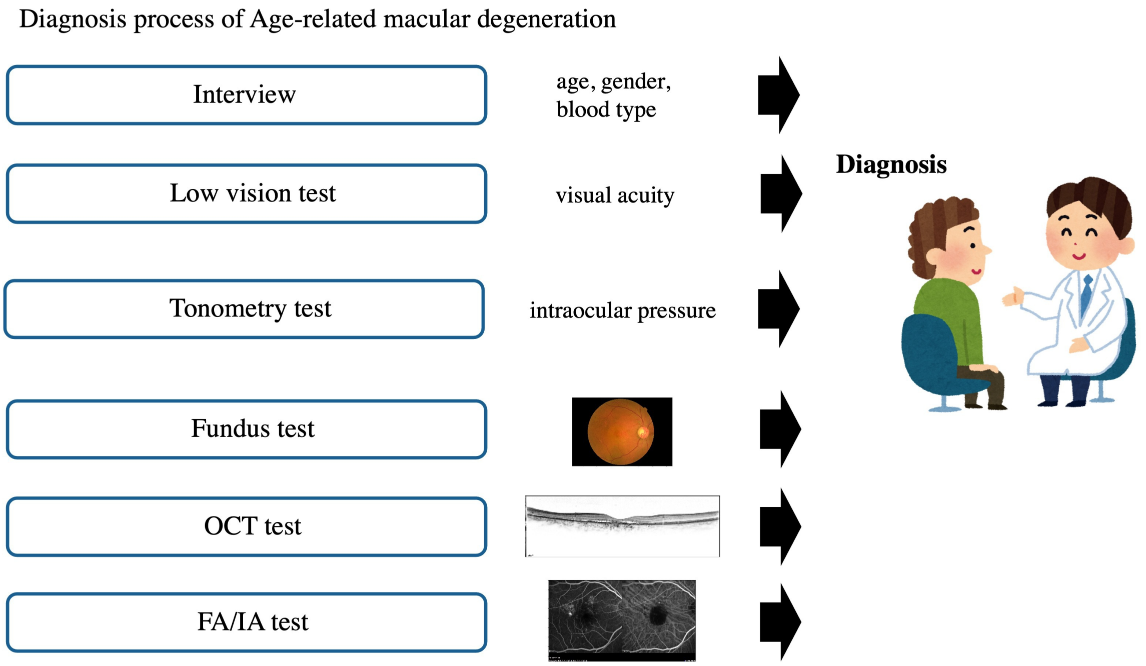


Fig. 2.5: Diagnosis process of eye disease

patient's morbidity. In 2, examination of the patient, the physician selects appropriate examination methods, carries them out, and obtains the results. Therefore, this process of inspection requires two elements: ***2a.*** Choosing the appropriate inspection parameters, and ***2b.*** Performing the inspection. In 3, interpretation of test results, the physician acquires the results and notes data regarding abnormal values or findings in each item. Accordingly, the interpretation process also requires two elements: ***3a.*** Acquisition of test results, and ***3b.*** Detection of abnormal values and abnormal sites. In 4, comprehensive diagnosis, the physician compares with normal the abnormal values and abnormal sites obtained identified in the test results, integrates the information, interprets it together with the patient's basic information, and finally diagnoses one or more diseases. Therefore, the diagnostic procedure requires three processes: ***4a.*** Information integration, ***4b.*** Interpretation of integrated data, and ***4c.*** Presentation of diagnostic results.

For an MME to support the entire diagnostic process of physicians, all of the above steps need to be replaced by systems. Mechanical automation has become ubiquitous regarding collecting data such as interviews and sample tests, as illustrated by systems such as the Ubi AI interviews [1] and Hitachi's automated test systems [2]. In this research, it is assumed

[1] Ubie, Inc `https://ubie.life`

[2] Hitachi, Ltd. `https://www.hitachi-hightech.com/jp/science/products/medical-systems/clinical-analyzers/laboratory-test/`

that interview data and test results can be acquired automatically by using such available products. The present approach focuses on receiving the subsequent data, interpreting it, returning the diagnostic result, and simulating with a system.

Based on the above, the following five elements are necessary to support diagnosis by MME in this study: ***3a***. Acquisition of test results, ***3b***. Detection of abnormal values and abnormal sites, ***4a***. Information integration, ***4b***. Interpretation of integrated data, and ***4c***. Presentation of diagnostic results.

2.3.2 Conceptual Design

The elements necessary to support the diagnosis include ***3a***. Acquisition of test results, ***3b***. Detection of abnormal values and abnormal sites, ***4a***. Information integration, ***4b***. Interpretation of integrated data, and ***4c***. Presentation of diagnostic results. By proposing a design that can deal with each element, this research aims to simulate the physician's diagnostic process by using a system.

Figure 2.6 shows the unified design requirements. The necessary design is roughly divided into a data input part, an abnormal value/part interpreting part, a data integration/interpretation data part, and a data output part.

First, the system needs a data input component to acquire inspection results. The input data is expected to include numerical data obtained from sample examinations and electrocardiograms, motion image data obtained from ultrasonic examinations and endoscopy, static image data obtained from X-rays, CT, and MRI examinations, and text data obtained from records of interviews. In addition, there is a possibility that multiple tests will be performed for diagnosis. Also, physicians always interview the patient. It is necessary to receive these pieces of data in various combinations. Next, to find abnormal values and abnormal parts, the system needs a function to interpret the input data and capture the features. A mechanism for individually extracting features from each data type is required to extract features from multimedia medical data. After that, the system needs a function to combine the extracted features. Features obtained from multimedia medical data do not always have the same format. The system needs to convert features into a standardized format, such as numerical information. Then, a mechanism for integrating features taking into consideration differences in the sizes and numbers of dimensions is required. Additionally, to interpret the integrated data, the system needs a feature extraction function that can be used for diagnosis based on the integrated features. Finally, the system needs a function to output the diagnostic results. The system should extract the features for deriving the diagnosis result from the integrated features. This is done by comparing the values of the extracted features against diagnostic criteria of diseases to determine which disease

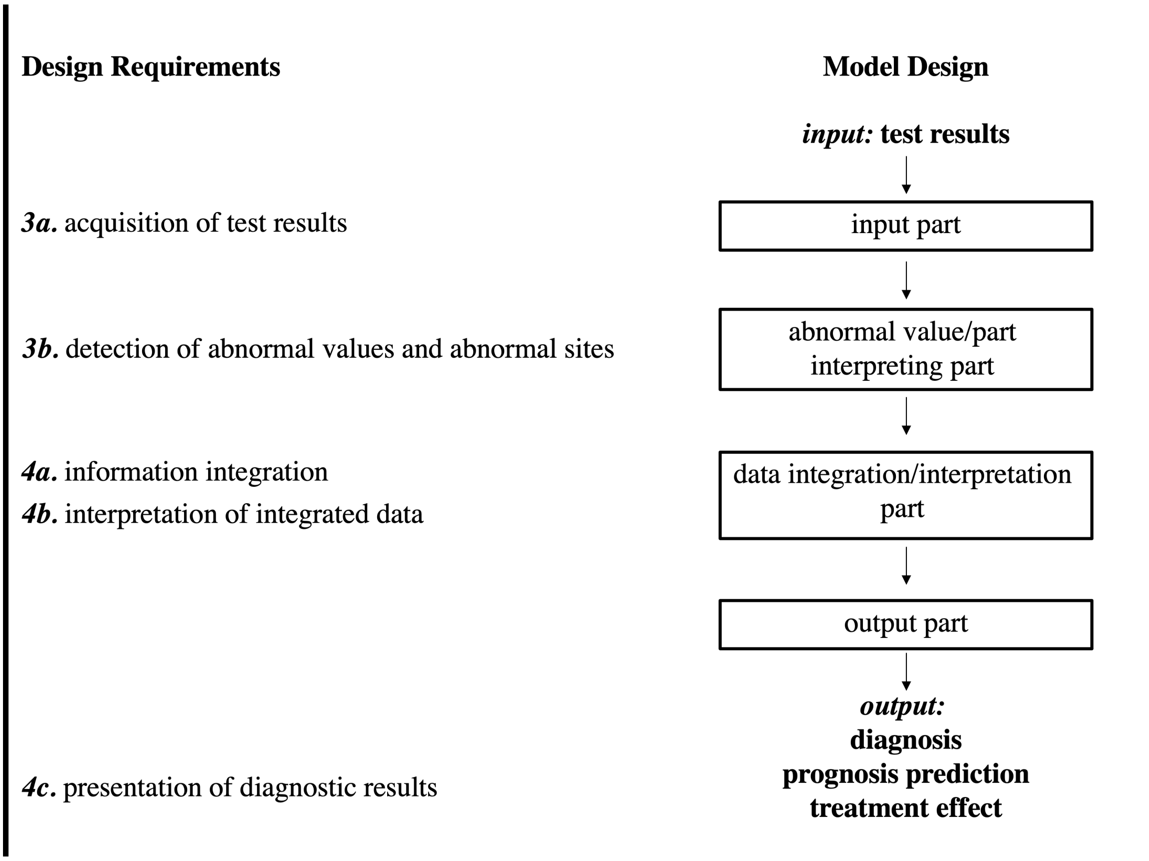

Design Requirements
Model Design
input: test results
3a. acquisition of test results
input part
3b. detection of abnormal values and abnormal sites
abnormal value/part interpreting part
4a. information integration
4b. interpretation of integrated data
data integration/interpretation part
output part
output:
diagnosis
prognosis prediction
treatment effect
4c. presentation of diagnostic results

Fig. 2.6: Design requirements

is more probable. The output function must present an output that is easy for physicians to quickly understand and handle, such as the probabilities of particular diseases in the diagnosis.

To support the physicians' diagnosis, as described above, the system must be designed to receive multimedia medical data as input data, perform feature extraction and feature combination, and output the diagnosis result in an easy-to-understand manner.

2.4 Study Aim

This research proposes a design method to support the physicians' comprehensive diagnosis process. The proposed design receives multimedia medical data and outputs results as diagnosis and analysis support. Sections 2.1 and 2.2 have identified the elements necessary to simulate the actual diagnostic process. The conceptual design for simulating the diagnosis process with the system was also described.

The proposed design method uses a deep learning method to model the MME. Three

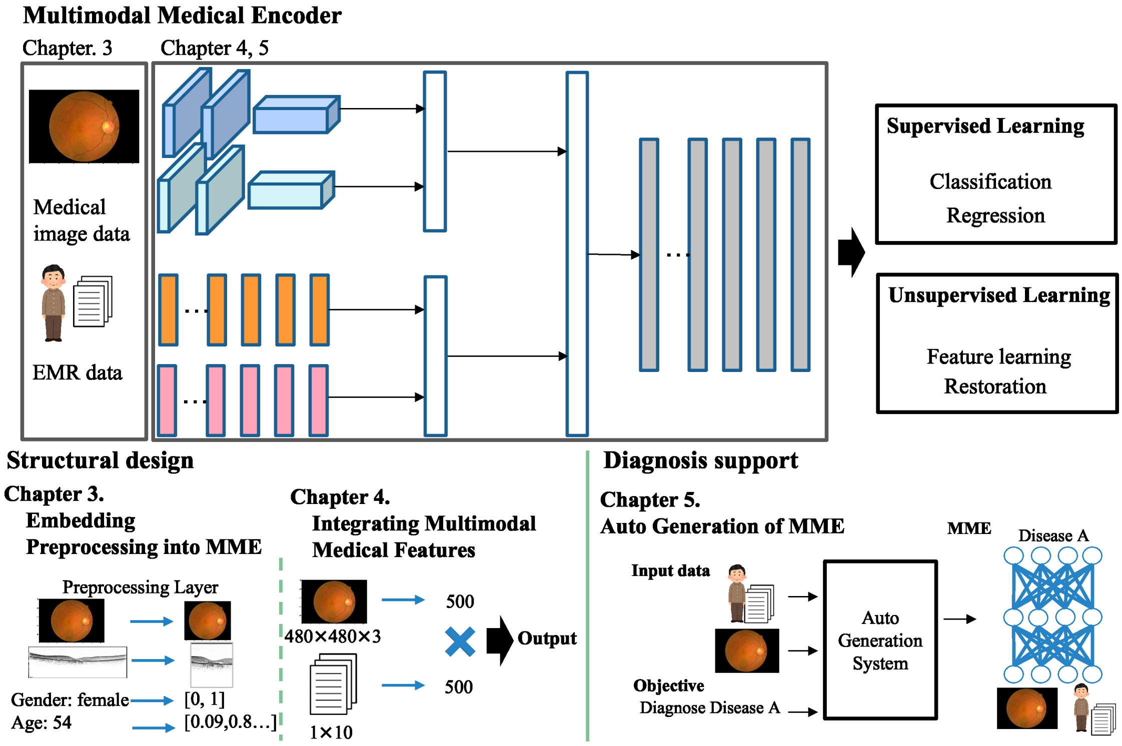


Fig. 2.7: Proposal overview

reasons to use deep learning methods include: several models for various data types have already been proposed; they enable integration of extracted features; and it is possible to support both diagnoses and prognosis predictions by changing the output layer.

Figure 2.7 gives an overview of the proposal. The deep neural network model generated by our proposed design method has three primary components which are: feature extraction layers corresponding with multimodal medical data inputs, integration layers of the extracted features which link to the output according to purpose, such as diagnostic results and prognosis prediction. Additionally, we propose an automatic generation system, which develops the deep neural network model based on its inputs and its output purpose determined by the user. Those components and auto-generation systems are described in the following sections.

The proposed design method eliminates the manual preprocessing process for multimedia medical data. Consequently, the method is expected to provide a more efficient model structure, and to be optimized for extracting better features of multimedia medical data. This work also proposes an automated method to design deep learning models. The method employs the efficient structures of neural networks to extract features from multimedia medical data as well as to integrate those features. Physicians who want a CDSS will be able to prototype a CDSS simply by collecting data using the proposal system. It can contribute to clinical practice as a diagnostic support tool that can be easily created and examined for CDSS.

Chapter 3

Embedding Preprocessing into MME

This chapter proposes a design method that handles multimedia medical data used for diagnosis in an MME. In clinical practice, numerical and image test data and text data of interview results are used to make a diagnosis. Because this study focuses on eye diseases, it is assumed that the proposed design will receive numerical and image data useful for diagnosing eye diseases. Until now, when inputting numerical values and images into a deep learning model, manual preprocessing was performed. However, designers could not perform this preprocessing without specialized knowledge, especially regarding medical data. In addition, general patient profile data such as gender and age data were individually preprocessed in formats that may differ among designers. The present study proposes a design for learning by embedding preprocessing inside a deep learning model. A case study of the predicted prognosis of AMD is used for verification in this study.

3.1 Interpretation of Medical Data Used for Diagnosis

3.1.1 How to handle medical data of physicians

For diagnosis, physicians use acquired image data and EMR that include basic patient information and examination data.

First, physicians predict the likelihood of a particular illness based on information in a patient's EMR. The basic patient profile in EMR includes patient age, gender, nationality, and medical history. Therefore, physicians first consider symptoms that are more likely to occur in that demographic, and the possibility of recurrence of any history. Next, The physician collects data from tests in the patient to use in detailed disease identification.

Most tests performed include items that can identify an illness suspected by the physician. The physician checks the result of each test for abnormal findings. From items with abnormal values, it is possible to predict which organs are involved. In addition, it may be possible to identify abnormal organs and predict the type of abnormality by observing evidence of any lesions in an image of the actual organ.

3.1.2 Embedding Physician's Data Interpretation

To include multimedia medical data and embed the process of interpreting each outlier in a deep learning model, a process to receive items of multimedia medical data as inputs and extract features of interest in each item is required.

The conceptual design of this study for auto preprocessing simulates the physicians manual diagnostic process of AMD as a case study. To diagnose AMD, ophthalmologists usually conduct multiple tests. First, they may measure VA by using the Landolt ring or other VA chart. Second, they may perform a dilated fundus examination using ophthalmoscopy or fundoscopy to document FP (fundus photography). images. Third, they might screen for exudative macular change and choroidal neovascularization by using OCT. Finally, to confirm the status of any lesion, they would look for presence of abnormal leakage from the choroidal neovascularization by imaging after intravenous injection of a fluorescent contrast agent. Ophthalmologists then comprehensively assess the results of all of the tests to decide the diagnosis and progression of AMD [4,67]. However, it is still difficult to accurately predict the extent of VA recovery after treatment.

This Chapter describes the design and integration of layers for preprocessing both clinical image and EMR data into multimedia deep neural network models. The preprocessing of clinical images is performed by a ViT (vision transformer) layer that selectively evaluates ROI (regions of interest) in the images. In contrast, the preprocessing of EMR and numerical data is performed by full-connection layers before and after normalization.

3.1.3 Conceptual Design for Auto Preprocessing

Physicians use images obtained by FP and OCT as clinical image data, and EMR data including gender, age, medical history, and affected side as well as pretreatment decimal VA to diagnose AMD. To handle multimedia medical data as would a physician, the model needs layers to receive data of different media type such as clinical image, text and numerical EMR data, and layers to preprocess and extract features from each of these types of data set.

Since input data differ markedly in size and range, these factors were accommodated by designing a proposed model that can receive input of imaging and EMR data, preprocess and extract features from imaging data, preprocess and extract features from EMR data, combine the extracted features, and output the predicted VA as a numerical value.

The proposed model does not require preprocessing of input data to predict posttreatment VA with greater accuracy than does a manual preprocessing model. Thus, this model eliminates the need to consider and apply manual preprocessing preventing any loss of information in input data during preprocessing.

3.2 Proposed Design

3.2.1 Bias Applied by Manual Preprocessing

Deep learning models have been proposed that learn features regardless of the preprocessing content of the input data. For example, use of a CNN (convolutional neural network) [30] led to a proposed model that utilizes MRI to predict the probability of patients with mild cognitive impairment developing Alzheimer's disease [34]. The CNN used in that model could extract both local and overall features in images by changing the filter size of the convolution layer and inserting a pooling layer. Therefore, the features could be learned using only the critical part of the input image. In partial image recognition models that apply CNN, such as Faster R-CNN [50], classification may be possible after cropping redundant sections of an image, thus reducing the area to be searched to identify lesions [15, 55]. In another study, based on information about insulin administration and other factors, a sequence-to-sequence model was able to predict rapid increase or decrease in of blood glucose concentration [4]. The sequence-to-sequence model learns using data obtained during patient examination, and primary data as time-series interrelated data [60]. In these models, however, the input data was preprocessed manually, and it is unclear whether the results of preprocessing were optimum for model training.

Many deep learning models that receive medical data as input require preprocessing of the input data. Images are frequently preprocessed by resizing and cropping. Resizing refers to their transformation to predefined uniform vertical and horizontal sizes to accommodate CNN requirements. Cropping refers to cutting out parts of an image to isolate and evaluate the lesional area. Cropping is performed to reduce noise in the input data and help the model focus on a target. Standard preprocessing of EMR includes normalization and categorization. Data are normalized to suit a specified range of input data. In contrast, categorization assists the model to determine relationships between single data points and the entirety of the data, as well as the order of the data, for example, by converting into a vector with clusters every 10-years.

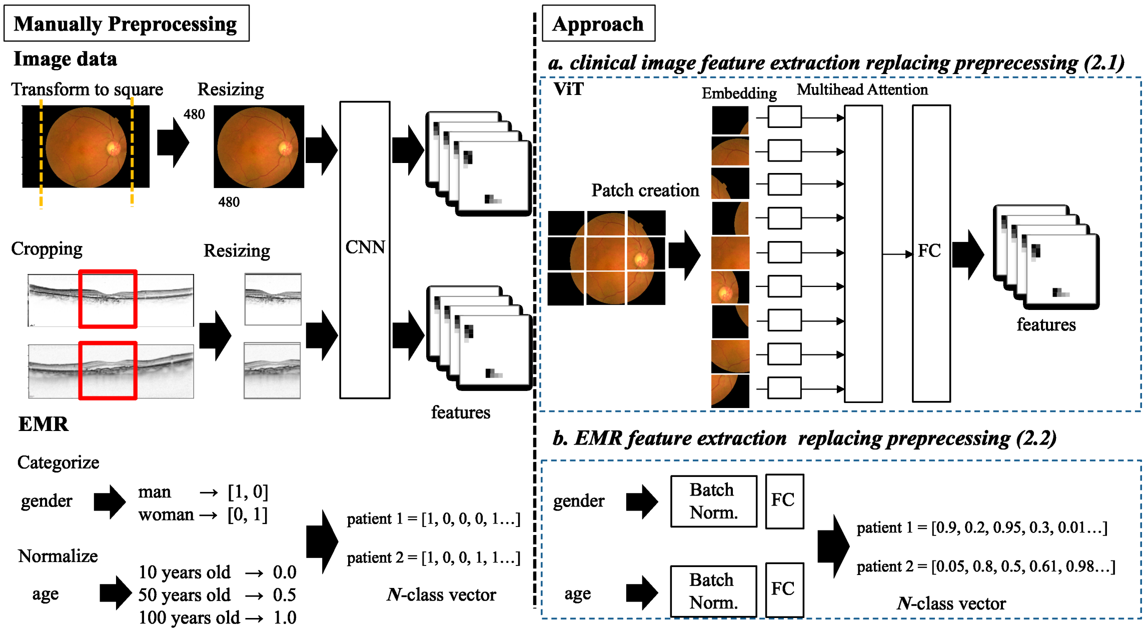


Fig. 3.1: Overview of preprocessing

These preprocesses are usually performed manually, and significantly impact the accuracy of a model. Rather than manually preprocessing multiple medical modalities, we proposed inclusion of preprocessing integration layers in the neural network. Figure 3.1 illustrates the processes of manual preprocessing and how our proposed integration layers replace them. It includes an attention mechanism which allows the system to process specific features in complex input one-at-a-time to categorize the whole dataset. Clinical images were split into same size patches, and features were extracted using the attention mechanism and FC (fully connected) layers. BN (batch normalization) and FC layers were used to preprocess EMR and extract features. By training the preprocessing engine, it is assumed that input data can be transformed to meet the requirements of the task required by the user.

3.2.2 Preprocessing Layer for EMR

The scalar and discrete values in a patient's EMR are essentially multi-dimensional. Manual preprocessing of EMR involves sorting into categorical multi-dimensional vectors based on clinical knowledge, such as whether interpretations of blood test data differ between young and elderly patients.

Figure 3.1b illustrates how the proposed integration layers extract EMR features. Scalar values, such as gender and age, are input to the layers, which output the feature vectors of an EMR.

EMR preprocessing and feature extraction was performed using a BN layer and an FC layer. The BN layer interprets the data distribution in the batch and transforms the data so that the mean output is close to 0 and its standard deviation is relative to 1. The BN layer has been used to prevent gradient disappearance and gradient explosion during model training. After normalization, the FC layer extracts and linearly transforms the feature of interest from the EMR. Using BN and FC layers results in a structure for preprocessing normalization and feature extraction.

3.2.3 Preprocessing Layers for Image

Figure 3.1a illustrates the role of preprocessing and shows the method by which the proposed integration layers extract features from clinical images. Clinical images are preprocessed by the ViT, which is a deep learning model for image recognition [11]. The ViT image recognition and classification model employs transformer like architecture rather than a CNN [63]. CNN have several advantages, including avoiding the need for hand-designed visual features by learning to perform tasks directly from data. However, CNN architecture is designed for a certain image size.

ViT transformer architecture was originally designed for text-based tasks. Transformer translation models use only attention without CNN or RNN and have an encoder-decoder structure. The encoder side maps an input sequence of words to a sequence of continuous representations. The decoder then generates an output sequence of symbols one element at a time from the sequence of continuous representations. The transformer uses self-attention on both the encoder and decoder sides. Self-attention maps a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. Self-attention expresses the input word group with three vectors of the query (Q), key (K), and value (V), and Q of the input word and each word. Then, $Q \cdot K$, the inner product of Q of the input word and K of each word, is calculated and treated as the degree of relevance. By applying the softmax layer to this degree of relevance, translated words are generated and sequentially output by the decoder.

Figure 3.2 shows the flow of processing by ViT. In ViT, an input image is split into patches. Figure 3.3 shows the flow of extracting patches. The size of patches is set as a hyperparameter. Image patches are treated the same way as tokens (words) in an natural language processing application. The image patches are flattened and input to the encoder side of the transformer. At the time of input, position embeddings are added to the patches to retain positional information. This process is similar to the sequence of word embeddings used when applying transformers to text, with the ViT directly predicting class labels for the image. These features enhance independence from image size, as well as increase the ability of the multihead attention mechanism to learn which image parts to focus on. Multihead attention, which is used to extract the features of each patch of ViT, is a mechanism that can be trained by combining the features of each patch. In typical attention, the features extracted from each patch are treated individually. Therefore, the model often responded to only one ROI. However, multihead attention allows the model to attend to information from different positions and different patches jointly.

Adoption of ViT may make possible the development of clinical image feature extraction layers as well as replacing some steps in image preprocessing, such as resizing and cropping.

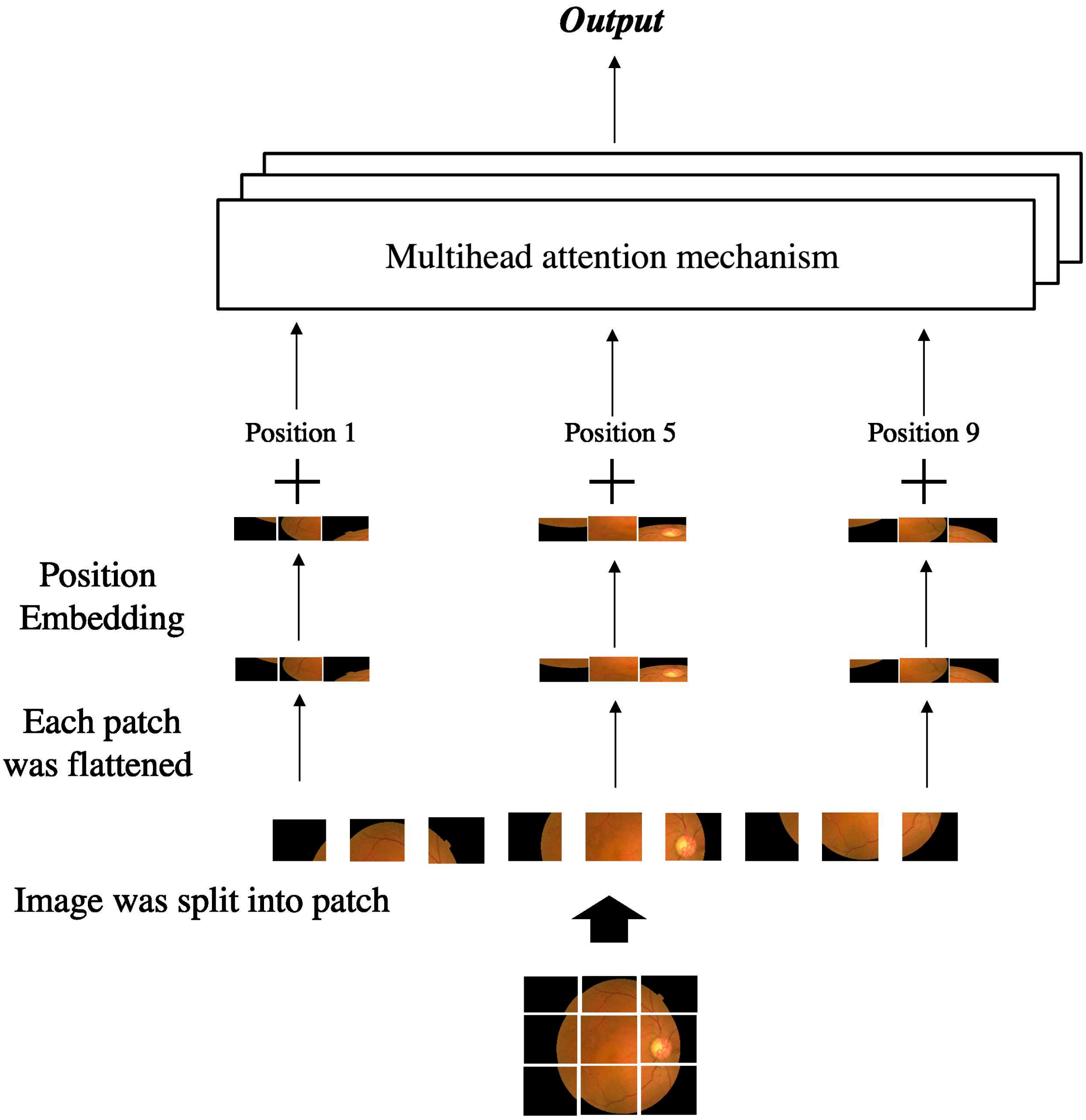


Fig. 3.2: Overview of Vision Transformer

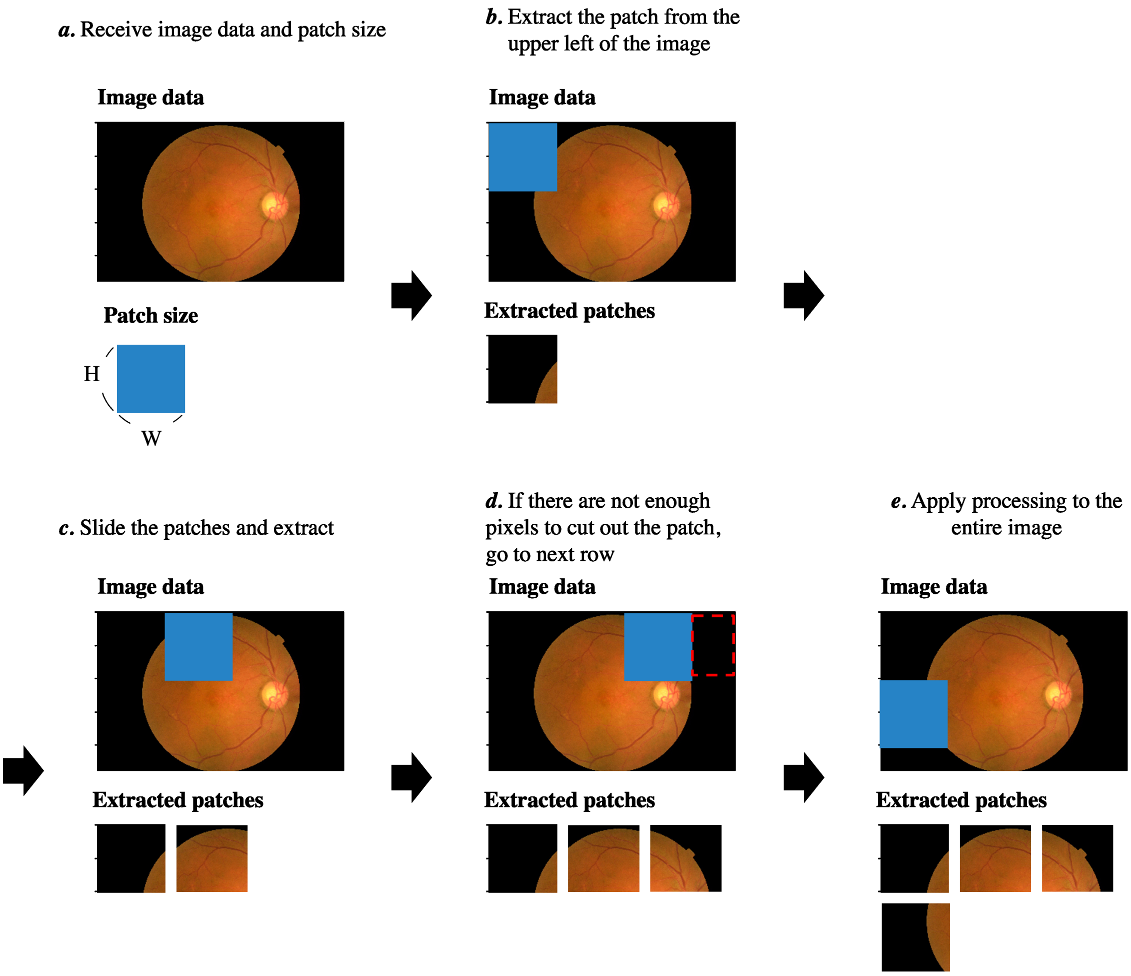

a. Receive image data and patch size
Image data
Patch size
H
W
b. Extract the patch from the upper left of the image
Image data
Extracted patches
c. Slide the patches and extract
Image data
Extracted patches
d. If there are not enough pixels to cut out the patch, go to next row
Image data
Extracted patches
e. Apply processing to the entire image
Image data
Extracted patches

Fig. 3.3: Flow of extracting patches

3.2.4 Combination of Layers

For classification or regression tasks, extraction of features from clinical images and EMR are concatenated and passed into the FC layers.

The data handled in this design are images and patient profile data. The size of the input data is $994 \times 1500 \times 3$ pixels for FP and $496 \times 1532 \times 3$ pixels for OCT-h (OCT horizonal cross section image) and OCT-v (OCT vertical cross section image), respectively. The patient profile data, gender, age, eye side, preoperative decimal VA, and preoperative logMAR (logarithm of the minimal angle resolution)are divided into one-dimensional arrays and input individually. The image features extracted by ViT are combined into 3072 dimensions, and the patient profile data is combined after expansion to 500 dimensions. Features extracted from different media may have large differences depending on the number of dimensions of the input data. Bias caused by differences in the numbers of dimensions can be a reason the training of a model does not proceed well. The dimension size of concatenated feature vectors for each medical modality should be identical to avoid biases. To match the dimensions of the vector, the features extracted from images are reduced in dimension in the FC layers.

Once image features and features extracted from the patient profile have the same number of dimensions, they are combined using the concatenate layer. The combined features are connected to the dense layer to output postoperative VA through FC layers that interpret integrated features.

3.3 Evaluation

3.3.1 AMD

This section focuses on AMD and uses postoperative VA prediction as a case study.

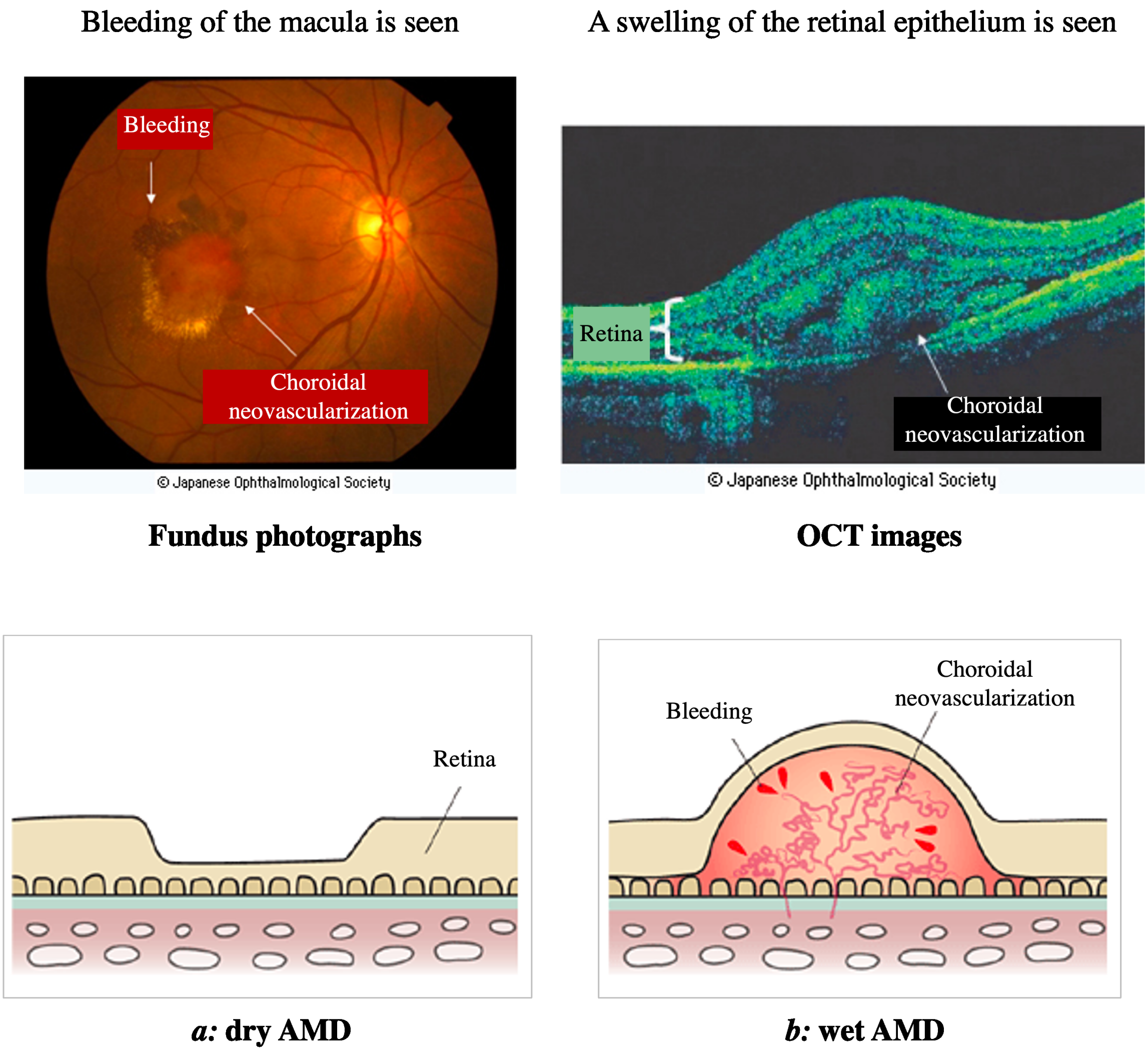
Figure 3.4 shows a summary of AMD [1]. AMD is a major cause of visual impairment in the elderly. AMD ranges from blurred central vision to complete blindness. There are two types of AMD, dry and wet [2]. In dry AMD (Figure 3.4a), RPE (retinal pigment epithelium) cells undergo gradual atrophy, leading to damaged retina and irreversible blindness. In wet AMD (Figure 3.4b), choroidal neovascularization develops beneath the RPE or between the retina and RPE, leading to damaged retina and loss of vision [33].

Photodynamic therapy and laser photocoagulation are used to treat wet AMD [3]. In

[1] Japanese Ophthalmological Society `http://www.nichigan.or.jp/public/disease/momaku_karei.jsp`

[2] EyeLife Age-related Macular Degeneration (Japanese) `https://www.eyelifemegane.jp/v2/sick_macular_degeneration.php`

[3] Japan Opthalmologists Association `https://www.gankaikai.or.jp/health/51/index.html`


Bleeding of the macula is seen
A swelling of the retinal epithelium is seen
Bleeding
Choroidal neovascularization
© Japanese Ophthalmological Society
Retina
Choroidal neovascularization
© Japanese Ophthalmological Society
Fundus photographs
OCT images
Retina
Bleeding
Choroidal neovascularization

a: **dry AMD**

b: **wet AMD**

Fig. 3.4: Summary of AMD

the former, weak laser energy is applied to the lesion after intravenous injection of a photoactive drug such as verteporfin. After the initial treatment, further treatment is continued while assessing the condition every 3 months. In the latter, a laser is used to burn off choroidal neovascularization not in the fovea. Part of the retina at the laser-irradiated area is damaged and disappears. However, since it prevents new blood vessels from reaching the fovea centralis, it prevents VA deterioration.

In recent days, intravitreal injection of antivascular endothelial growth factor agents, such as aflibercept, is commonly used. However, since not all patients respond to treatment, predicting posttreatment VA is difficult [48]. If it were predicted with high accuracy using a deep learning model, the burden on physicians and patients would be greatly reduced.

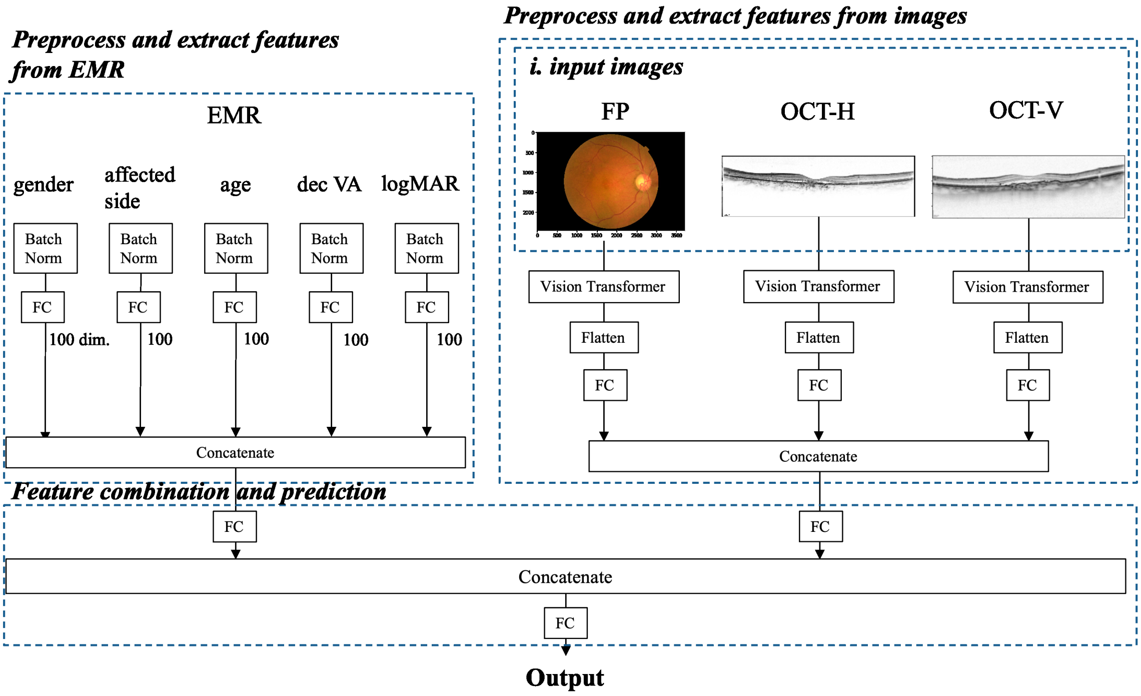


Fig. 3.5: Overview of evaluation model about preprocessing design

3.3.2 Evaluation Task

Experiments were performed to assess the effectiveness of the preprocessing-integration layers. One included the two tasks of evaluating the accuracy of the proposed model and visualizing the results following conversion of the data. Experiments were performed under four conditions, with the differences between predicted and actual outcomes compared by determining the MSE (mean square error).

Performance of the proposed integration layers was assessed using a VA prediction task. Specifically, a deep neural network model was developed to predict posttreatment VA in patients with AMD based on medical imaging and patient EMR data. Figure 3.5 outlines the proposed model. This model has three layers: 1. A layer that preprocesses and extracts features from imaging data, 2. A layer that preprocesses and extracts features from EMR data, and 3. A layer that combines the features of the first two layers and predicts VA.

In addition, hyperparameter tuning was performed on the preprocessing layer for EMR in the model shown in Figure 3.5. Figure 3.6 shows an image of preprocessing for EMR used for hyperparameter tuning. The target hyperparameters include: 1. BN layer or layer normalization layer, 2. With/without softmax layer, 3. With/without attention layer, 4. Change the number of fully connected layers, and 5. Change the loss function during training.

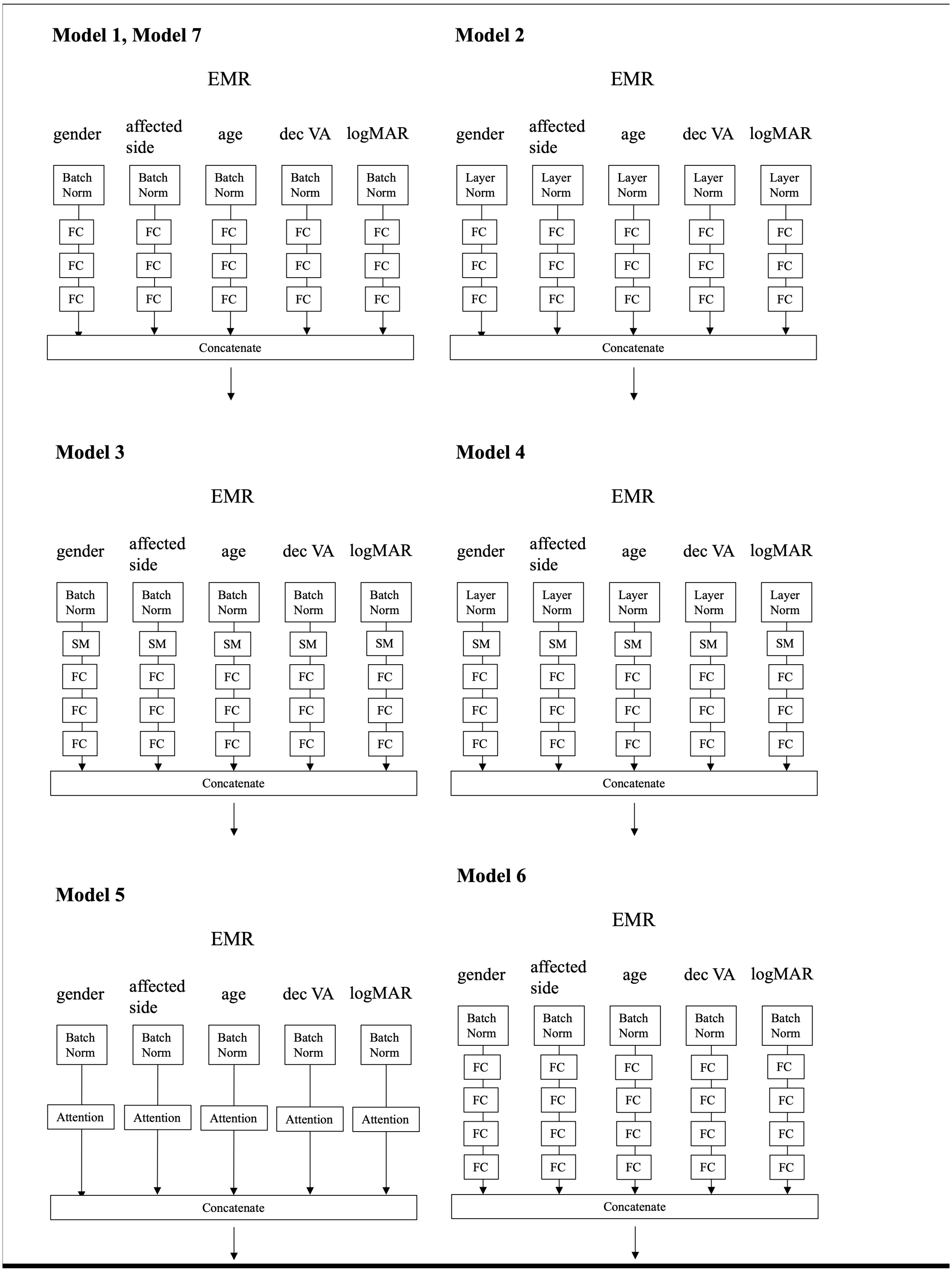


Fig. 3.6: Overview of hyper parameter tuning models

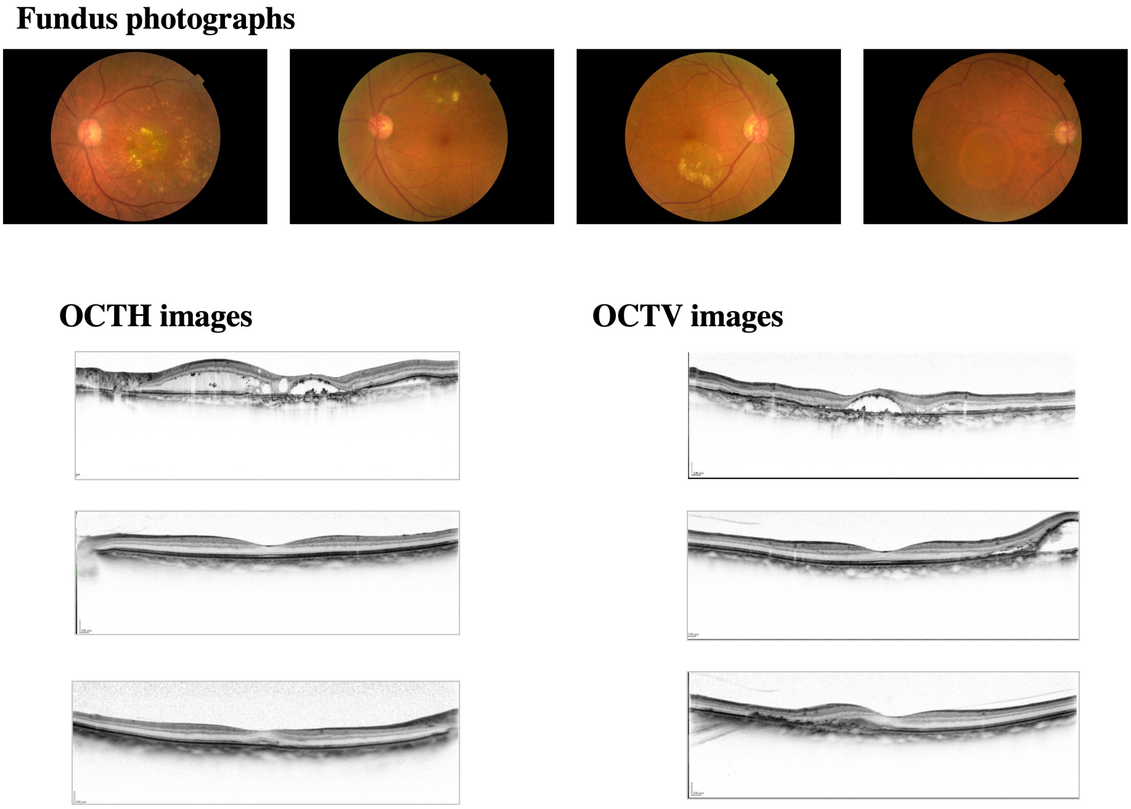


Fig. 3.7: Sample images used in this study

3.3.3 Data for Evaluation

Figure 3.7 shows examples of input imaging data, including FP and OCT images. FP is 1500 × 994 pixels in size and has three RGB channels. OCT images are of horizontal and vertical cross-sections, called OCT-h and OCT-v, respectively. Both are 1532 × 496 pixels in size and have three RGB channels. EMR includes gender, age, affected side (right or left), and pretreatment decimal VA [29]. To analyze VA as a continuous variable, decimal VA was converted to logMAR [17]. Both pretreatment and posttreatment decimal VA were converted to logMAR.

This study was approved by the Ethics Committee of Kyoto University Graduate School and Faculty of Medicine (R2366) and adhered to The Ethical Guidelines for Medical and Health Research Involving Human Subjects in Japan as well as the tenets of the Declaration of Helsinki. Informed consent was obtained from each participant using an opt-out method permitted by the Ethics Committee.

Data were obtained from 315 patients who visited the macular clinic at Kyoto University Hospital, were diagnosed with wet AMD, and completed a fixed regimen of IVA (intravitreal aflibercept) injection for one year. They were separated into a training set of 252 patients and a validation set of 63 patients for five-fold cross-validation.

Manual preprocessing of image data included resizing and cropping, whereas manual preprocessing of EMR data included normalization and categorization. FP images were resized to 480×480 pixels and split into three RGB channels. The central square regions of OCT-h and OCT-v were cropped and resized to 480×480 pixels. Figure 3.5e shows input EMR data. Gender and affected side were categorized (0 or 1). Patient age was subjected to min-max normalization, as shown in Eq. 3.3.3. then the normalized age was multiplied by ten to convert to it into integers. Basically, min-max normalization is a feature scaling method to normalize values enabling scores from different sources to be compared. In other words, it standardizes the distribution as does the Z-score. Min-max normalization linearly transforms X to $Y = \frac{x-min}{max-min}$ so that the full range of x values may be mapped to the range 0 to 1, or another standardized range as desired. Floating logMAR was converted to an integer by multiplying decimal VA by ten and adding the minimum logMAR then multiplying by ten according to the formula shown below.

$$convertedage = \frac{age - age_{min}}{age_{max} - age_{min}}$$

3.3.4 Experimental Conditions

The predictive accuracy of the manually designed network was compared with that of the network with the proposed preprocessing integration layers. Table 4.1 describes the structures of these models. The model incorporating preprocessing was found to preprocess the data and extract features by the ViT and BN layers, as described in Figure 3.5. In the manually preprocessed model, image features were extracted by a model called VGG16 [57], which is generally used for image classification, and features were extracted from EMR by an FC layer. At this time, for "Proposed," the input image was not trimmed, and the one-dimensional EMR data was divided into each item. While, for "Baseline," input images and five-dimensional EMR were preprocessed as described in Section 3.2. Because the model incorporating pretreatment was expected to perform pretreatment more suitable for prediction, we hypothesized that MSE would be lower in a model that included preprocessing than in a model that included manual preprocessing.

To determine whether the preprocessing layer was more effective with imaging or EMR data, its accuracy was compared using model b in Table 3.1. When the preprocessing layer was applied only to the image, ViT was applied to the image, and the FC layer was applied to EMR. Models to evaluate the preprocessing design were applied to EMR data which was not divided into items but input in a five dimensional state. When the preprocessing layer was applied only to EMR, VGG16 was applied to the image, and the BN layer was

Table 3.1: Models for evaluate preprocessing design

⟍	Model Name	Layer for Images	Layer for EMR
a	Baseline	VGG16	FC
	Proposed	ViT	BN and FC
b	Baseline	VGG16	FC
	Only Image Preprocessing	ViT	FC
	Only EMR Preprocessing	VGG16	BN and FC

Table 3.2: Models for hyper parameter tuning

Model Name	Layer for Preprocessing	Layer for Feature Extraction	number of FC	loss function
Model 1	BN	FC	3	MSE
Model 2	LN	FC	3	MSE
Model 3	BN + SM	FC	3	MSE
Model 4	LN + SM	FC	3	MSE
Model 5	BN	Attention	3	MSE
Model 6	BN	FC	4	MSE
Model 7	BN	FC	3	MAE

applied to EMR. Thus, EMR, which could only be converted at regular intervals by manual operations, would likely form a cluster more suitable for the prediction task. Therefore, we hypothesized that MSE would be higher when the preprocessing layer was applied only to the image than when it was applied only to EMR.

Table 3.2 shows a list of models and settings used for hyperparameter tuning, and Figure 3.6 gives an overview of the models. Model 1 has the same design as the proposed model. Models 1 and 2 have different normalization layers: the BN layer and the layer normalization layer. Models 1 and 3 and models 2 and 4 each have the same layer for normalization, except that they include a softmax layer to explicitly categorize data. Models 1 and 5 have the same layer for normalization, but differ in that the attention layer is used as the mechanism for extracting data features after that. Models 1 and model 6 have different numbers of fully-connected layers following the BN layer. Models 1 and 8 have the same layer design, but different loss functions during training.

Finally, we visualized how EMR was transformed by the preprocessing layer and compared its output with diagnosis by an ophthalmologist. In this experiment, after learning the proposed model, validation data were input, and the features were output before integrating the features in the EMR preprocessing layer. PCA (principal component analysis) was applied to this feature, with the results presented as a K-means clustering image and a hierarchical clustering image. The cluster was divided into two parts, with thresholds for age, pretreatment decimal VA, and pretreatment logMAR. We hypothesized that these

Table 3.3: MSE of evaluation models

╲	Model Name	MSE for Traning Data	MSE for Validation Data
a	Baseline	0.049	0.052
	Proposed	0.039	0.054
b	Baseline	0.049	0.052
	Only Image Preprocessing	0.050	0.118
	Only EMR Preprocessing	0.068	0.061

Table 3.4: MSE for hyper parameter tuning model

Model Name	MSE for Training Data	MSE for Validation Data
Model 1	0.039	0.054
Model 2	0.173	0.095
Model 3	0.047	0.058
Model 4	0.154	0.095
Model 5	0.046	0.063
Model 6	0.055	0.057
Model 7	0.082	0.056

thresholds would be consistent with those determined by an ophthalmologist.

3.3.5 Results

Table 3.3. shows the MSE between predicted and actual logMAR. The effect of the preprocessing layer was determined by comparison of MSE among the baseline and proposed models. The MSE of the proposed model with the preprocessing layer differed little from that of the baseline model. Comparison of MSE among only image and only EMR preprocessing models revealed the MSE was larger when the preprocessing layer was applied to only image data.

Table 3.4 shows prediction errors for each model used in hyperparameter tuning. First, among models 1 and 2, which have different pretreatment layers, model 1 using the BN layer for pretreatment had a smaller prediction error. Next, comparisons among models 1 and 3, and models 2 and 4, which have the same layer for normalization but with or without the softmax layer, showed when the BN layer was used, the prediction error of model 1 without the softmax layer became smaller. However, when the layer normalization layer was used, the prediction error was smaller when the softmax layer was used. Next, comparison of models 1 and 5, which have different feature extraction layers, showed the prediction error was smaller for model 1 whose feature extraction layer is the FC layer. And comparison of models 1 and 6 with different numbers of FC layers, showed model 1 with a smaller number of FC layers had a smaller prediction error. Finally, of models 1

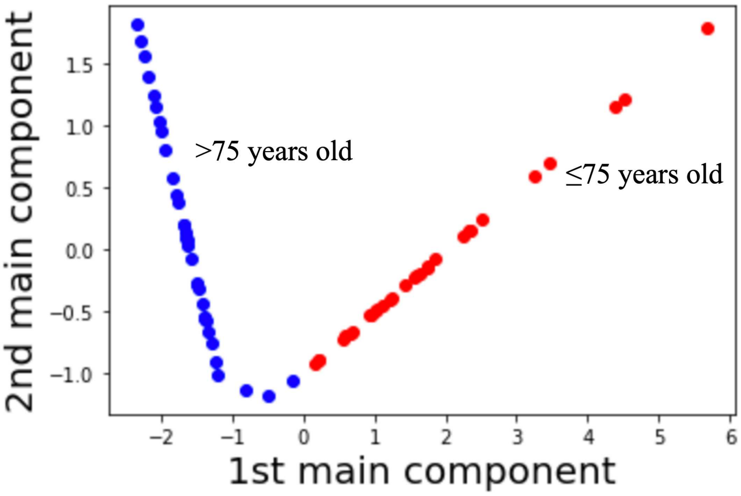


Fig. 3.8: Results of K-means clustering

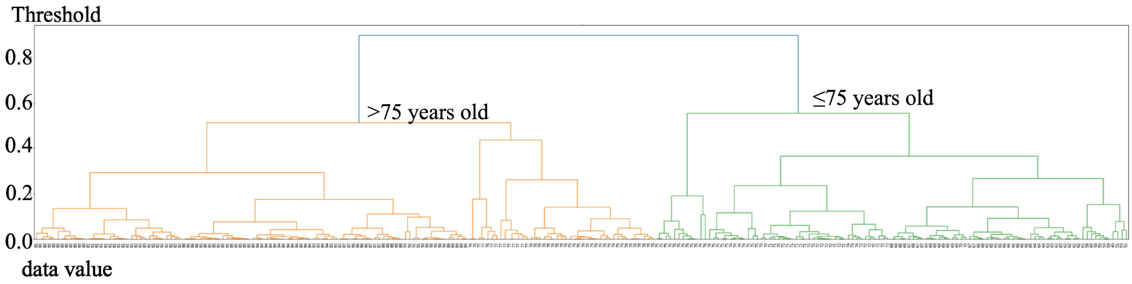


Fig. 3.9: Results of hierarchical clustering

and 7 that showed different error functions during training, the prediction error of model 1 was smaller.

Figure 3.8 shows a K-means clustering image of the pretreatment patient age output from the trained preprocessing layer. The output was made two-dimensional by PCA. The number of clusters visually judged as most appropriate was selected from the setting values of two to six. The cluster threshold was 75 years of patient age.

Figure 3.9 shows a hierarchical clustering image of the pretreatment patient age output from the preprocessing layer, which was made two-dimensional by PCA. When hierarchical clustering [36] was performed after PCA, two clusters appeared, as indicated by the color of the dots in the scatterplot. One contained data from pretreatment patients aged < 75 years and the other ≥ 75 years. This age cutoff was essential for predicting pretreatment logMAR.

3.3.6 Discussion

ViT preprocessed images and EMR data in this study were preprocessed by the BN and FC layers, with the MSE between predicted and actual results of 0.054. Since a logMAR difference of < 0.20 indicates effective treatment [27,52], the model had sufficient prediction accuracy for practical use.

Moreover, MSE at baseline was 0.051 (< 0.20), with the proposed integration layer demonstrating almost equivalent performance with baseline. Thus, the proposed layer can preprocess data to predict postoperative VA as effectively as manual preprocessing.

A comparison of each integration layer showed that preprocessing of EMR alone had a smaller MSE than preprocessing of images alone. This finding indicated that optimizing EMR data was more important than optimizing image data in predicting VA. Moreover, this finding is consistent with results showing that determination of preoperative VA by an ophthalmologist had the most significant effect on prediction accuracy.

Although the validation MSEs of only image and only EMR preprocessing were higher than that of baseline, the total validation MSEs of baseline and our proposed method were almost the same. This finding indicates the EMR and image features extracted by our proposed method had a coordinated impact compared with the baseline method.

The main advantage of integrating preprocessing operations into a deep learning model is a lower burden for optimization. Manual preprocessing of image resizing and cropping demands both medical knowledge of lesions and manual handling of images. ViT with its multiheaded attention mechanism, can automatically detect targets in clinical images, resulting in a lower load for designing the model. In addition, the proposed layer will be accepted regardless of image size or shape. Because a complete image can be input with ViT, an advantage may be lack of potential information loss, which may occur in cropped images by manual preprocessing.

If we consider the result of hyperparameter tuning, the prediction error was smaller when using the BN layer rather than the layer normalization layer. The former layer calculates the mean/variance for each mini-batch, whereas the latter calculates the mean/variance for each input sample. These properties indicate model 1 was able to train the relationships between each sample (patient data) by the BN layer. Next, the softmax layer is converted into a ratio so that the total output value of the layers is 1. In this chapter, the softmax layer was used to convert the data into a categorical vector. Still, it is considered that the softmax layer did not have a positive effect. When the attention layer was used for feature extraction, the prediction error was larger than that with the FC layer. The attention layer is often used for processing time-series data, and its effect has been proven. Since the EMR used as input is not time-series data, it is probable that the attention layer had no effect. Next, the model's prediction error in which the number of FC layers was increased by one was slightly larger than that before being increased. Deeper models were expected to be able to extract detailed EMR features, but they were not significantly effective. Finally, prediction error became large when the error function during training was changed from MSE to mean absolute error. When the model was evaluated using MSE, it was revealed to be effective to use MSE for the error function and proceed with training so that MSE becomes smaller.

Optimization is enhanced by integrating preprocessing for EMR data. For example, manual optimization of age requires a decision to dichotomize age or categorize it into 10-year intervals. Optimal conversion requires the determination of a critical age threshold for diagnosis.

The proposed layer dichotomized patients by age into < 75 and ≥ 75 years, resulting in a predictive accuracy greater than by manual conversion of age with min-max normalization. The advantage of the proposed layer for EMR is that appropriate preprocessing can be performed without special knowledge of the disease.

3.4 Summary

This chapter proposed the integration of layers of preprocessing data in both clinical images and EMR by deep neural network models. In the proposed model, ViT and BN layers were utilized for preprocessing and feature extraction. To verify the effectiveness of the preprocessing layer, the ability of the model to predict VA was determined by comparing the MSE between actual and predicted logMAR in the model with and without the preprocessing layer. The MSE was 0.054 with and 0.051 without the preprocessing layer. The experimental results revealed that the regression model with the proposed preprocessing layers achieved an accuracy sufficiently close to that by manual preprocessing.

The proposed preprocessing integration layers bring several advantages. By learning preprocessing, input data could be converted into more efficient feature vectors for the prediction/regression task and output accuracy could be improved. In addition, there was no need to manually preprocess input data, reducing the time and effort required to create training data. Because the proposed layer can handle input data regardless of shape, it can likely be applied to various clinical decision support systems with multiple modalities, such as clinical image and EMR data.

These contributions have an important impact on the preprocessing work of the CDSS development process mentioned in Section 2.1. The proposed design method can automate preprocessing of multimedia medical data. Thus, enabling physicians and system engineers to reduce time and cost burdens of preprocessing.

Chapter 4

Integrating Multimodal Medical Features

Chapter 4 proposes a design that integrates features obtained from each item of input multimedia medical data. Multimedia medical data have different numbers of dimensions by data type. In addition, the feature size extracted from each item differs regarding number of input dimensions. Until now, among existing systems, there was an element of bias in the way features were combined. For example, if the numbers of dimensions of features differs significantly among features, only the feature with the larger number of dimensions is trained. Accordingly, prediction accuracy may be reduced. In this chapter, the numbers of dimensions of extracted features are adjusted to eliminate this initial bias applied during feature integration. A fully-connected layer is used to match the numbers of dimensions. The concatenate layer integrates features that have the same numbers of dimensions. To evaluate the proposed design, AMD prognosis prediction is used as a case study.

4.1 Interpretation of Medical Data by Physicians

As mentioned in Chapter 3, physicians receive and interpret multimedia medical data and make disease predictions based on each item. The physician understands the characteristics of abnormal values in each data type and predicts which disease is most likely for each abnormal value. Based on these results, physicians integrate and interpret the disease predictions individually obtained from each item and incorporate them to determine a final comprehensive diagnosis.

By integrated interpretation, when a patient with a headache and light-headedness is diagnosed, Meniere's disease may be possible if they are age 30s or 40s year old. However, if they are elderly, the possibility of cerebral infarction is higher. In addition, by hearing a patient's complaint that their abdomen hurts and seeing an examination image together, it is possible to obtain more detailed information such as whether and to what degree the stomach or intestines are involved.

To diagnose AMD, ophthalmologists usually conduct multiple tests. First, they may measure VA using a standardized tool such as Landolt C. Second, they may perform a dilated fundus examination using ophthalmoscopy, often documented in FP. Third, they may screen for exudative macular change and choroidal neovascularization using OCT images. Finally, to confirm lesion activity, they may evaluate the presence of abnormal leakage from the choroidal neovascularization after injecting a contrast agent containing a fluorescent dye into a vein in the arm. Ophthalmologists then comprehensively assess the results of all the tests to decide the diagnosis and progression of AMD [4, 67]. However, it remains difficult to accurately predict the extent of VA recovery after treatment.

4.2 Conceptual Design for Embedding Diagnosis Process

We propose a VA prediction model that utilizes both medical images and patient profile data to support prognosis prediction. The proposed model receives multimedia data as input and outputs predicted posttreatment VA. Multimedia inputs include medical images and profile data of the patient to predict posttreatment decimal VA. Figure 4.1i shows examples of input image data. Fundus photographs are resized into 480 × 480 pixels. The 480 × 480 pixel central square region of fundus OCT-h and OCT-v images isolated and obtained by cropping (Figure 4.1i, red square). The patient profile data includes gender, age, affected side (right or left), and pretreatment decimal VA [29]. Figure 4.2 shows how to preprocess patient profile data. Gender and affected side are categorized (0 or 1), and age is normalized with min-max normalization as shown in Eq. 3.3.3, then normalized age is multiplied by ten to convert the value to an integer in a process of standardization. We convert pretreatment decimal VA to logMAR [17] with Eq. 4.2.1 for analysis as a continuous variable, then we convert the floating value of logMAR to its integer by multiply decimal VA by ten and adding the minimum value of logMAR then multiplying again by ten. Finally, we concatenate all converted vectors as single patient profile data. We convert posttreatment decimal VA to logMAR in the same way.

$$logMAR = \log(\frac{1}{DecimalVA}) \tag{4.2.1}$$

The model includes three layers that perform different functions to handle multimedia input. Patient profile feature extraction layers extract features of gender, age, affected side, pretreatment decimal VA, and pretreatment logMAR, which represents the initial processes of a medical interview and general vision test (Fig. 4.1a). Image feature extraction layers extract features from FP and OCT images, which represents the process of fundus examination (Figure 4.1b). Feature combination layers concatenate features from the image feature and patient profile extraction layers, which represents the process of assessing the test results and making diagnostic and treatment decisions (Figure 4.1c).

4.3 Proposed Design

4.3.1 Bias on Extracted Features

Previous studies have reported deep neural network models that use images such as FP and OCT obtained from patients to diagnose ocular diseases [8, 21, 31]. Rohm et al. utilized machine learning algorithms including Random Forest and Lasso regression to predict posttreatment VA from clinical data, including examination results and treatment history of AMD patients in EMR [51]. Shun et al. used a convolutional neural network to predict posttreatment VA from pretreatment OCT images in patients with branch retinal vein occlusion [56]. However, few studies have reported deep learning algorithms to predict posttreatment VA utilizing pretreatment images and non-image clinical data.

This section proposes a model to predict posttreatment VA, which uses both pretreatment image data and non-image data of AMD patients. We use FP and OCT images as image data and use patient profile data, including gender, age, affected side, and pretreatment decimal VA as non-image data. Multimedia data may include multiple features that differ widely depending on data type. If a prediction model is created without considering such differences, its accuracy may decrease. To accommodate different types of input data, the proposed model was designed to receive input that includes a variable number of dimensions, extract features from image data, extract features from patient profile data, equalize the number of dimensions between input types, combine extracted features, and output predicted VA as a numerical value.

4.3.2 Feature Integration Layers for EMR

Figure 4.1a depicts how features are extracted from patient profile data. The process involves an embedding layer and FC layers. Figure 4.3 shows the flow differences between one-hot encoding and embedding. An embedding layer is used for vectorizing patient profile

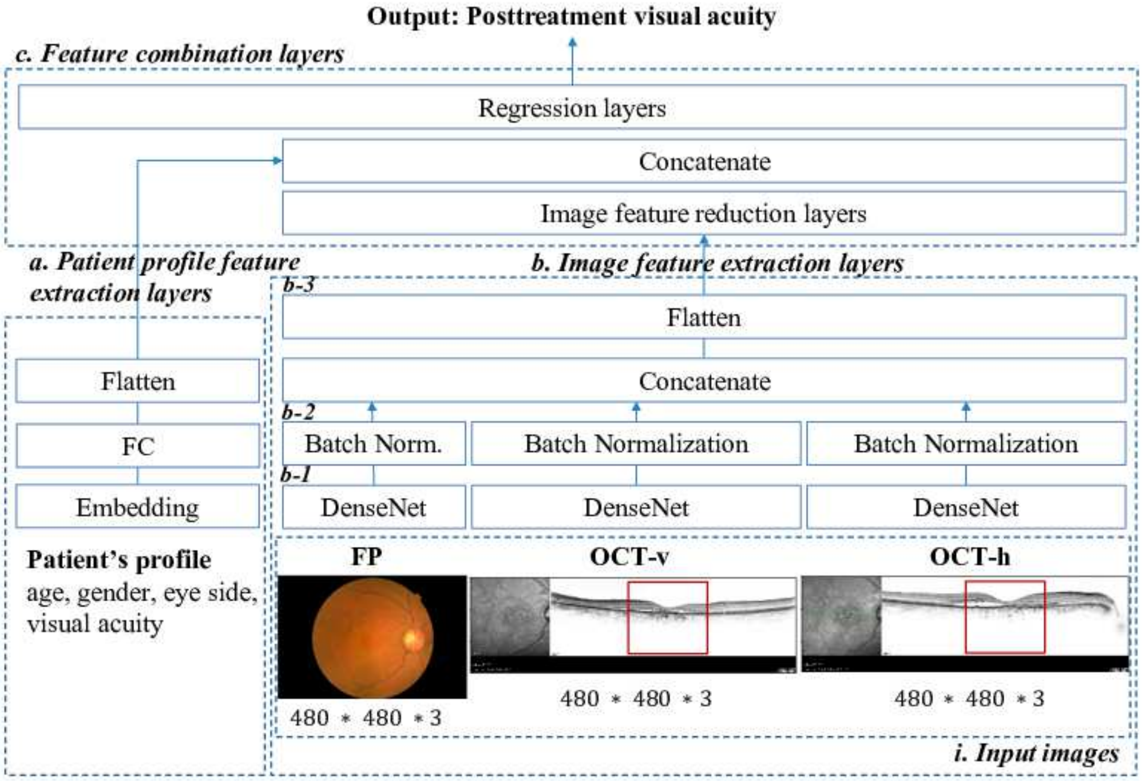


Fig. 4.1: Overview of integrated model

data. That is, one item of data is assumed to have latent variables for a given number of classes and is mapped to a multidimensional vector space. At this stage, since data with similar meanings are brought close to each other in vector space, data continuity is improved compared with the case of using one-hot encoding [22]. The embedding layer converts patient profile data into particular dimensional features to facilitate subsequent concatenation with image data. The embedding layer turns positive integers (indexes) into dense fixed-size vectors. This study utilizes this layer to convert patient profile data into 20 and 100 dense vectors. Those feature vectors are converted into 100 or 500 vectors by the latter FC layer. After conversion, FC layers are applied to each data set of each patient to extract the features. Then extracted features are flattened into a one-dimensional vector.

4.3.3 Feature Integration Layers for Image

FP, OCT-h, and OCT-v are input into each extraction layer and converted into image features as shown in Figure 4.1b.

To extract image features, DenseNet is used, which is a deep learning model commonly used to classify and identify images [25]. Figure 4.4 shows the differences between CNN

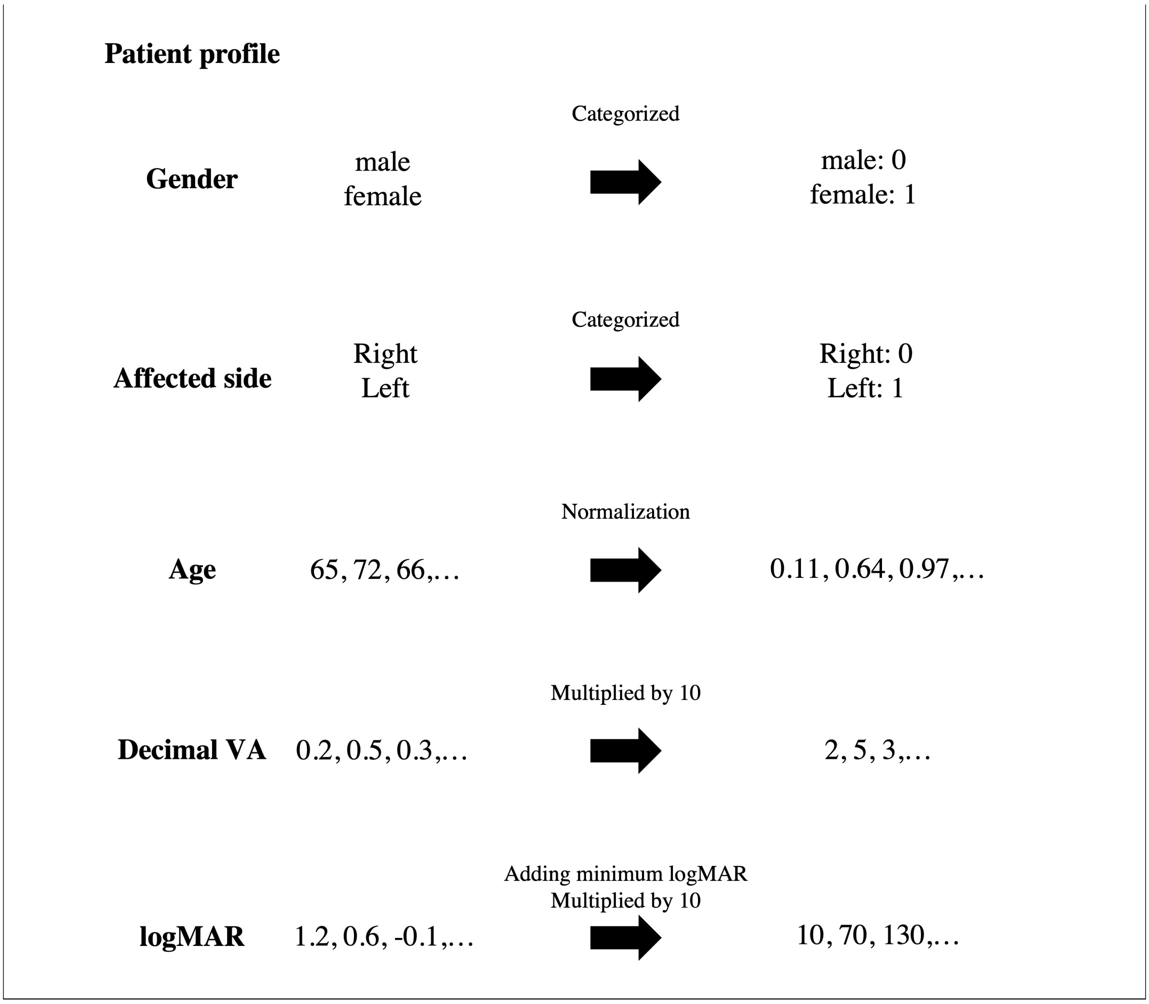
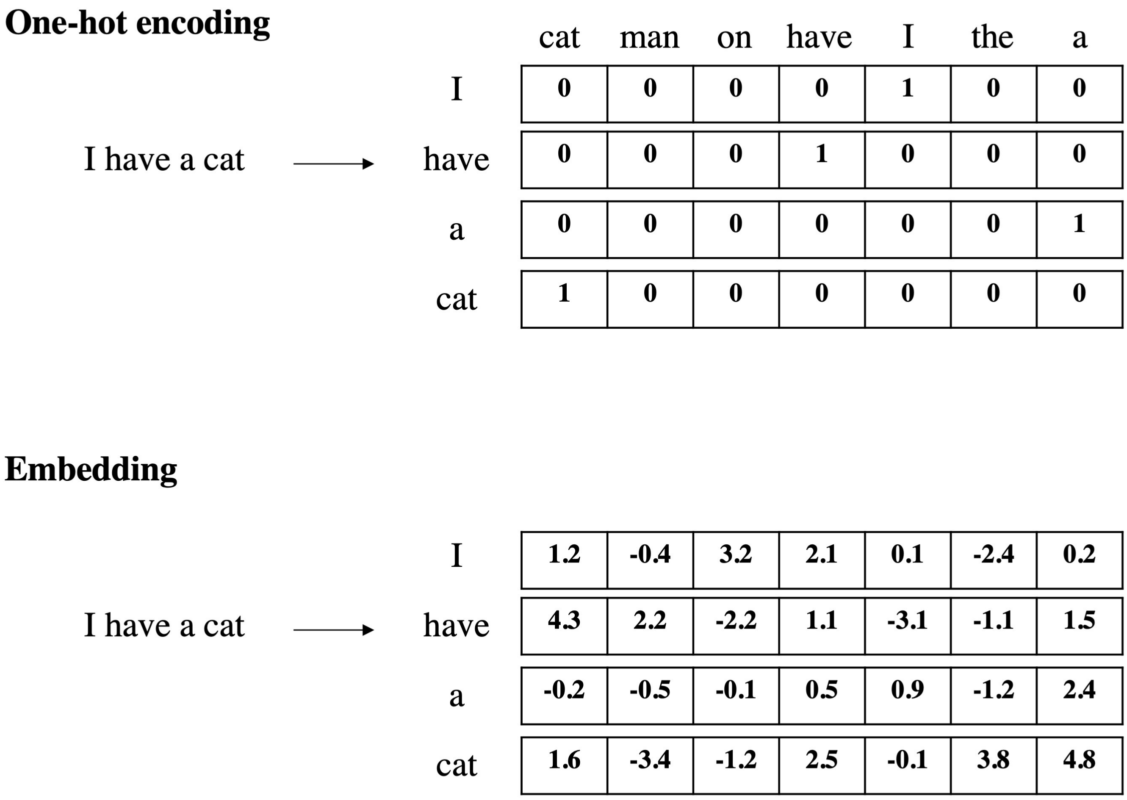


Fig. 4.2: Preprocessing of patient profile data

and DenseNet. DenseNet includes a layered structure called a dense block inside which all subsequent layers are connected directly from any layer. Consequently, the layer receives the feature maps of all preceding layers. This structure is different from the general convolutional process. The output of the dense block cannot connect to subsequent layers when the size of feature maps changes. However, downsampling layers that change the size of feature maps is an essential part of convolutional networks. To solve this, DenseNet prepares transition layers comprising a 1×1 convolutional layer followed by a 2×2 average pooling layer. DenseNet reduces gradient loss and efficiently transfers image features with the dense block.

In the proposed model, each image type is input to a different DenseNet (Figure 4.1b-1). The output layer of each DenseNet is deleted, resulting in the BN layer [26] (Figure 4.1b-2). We connect the DenseNet, which has $7 \times 7 \times 1024$ dimensions, to the BN layer,



	cat	man	on	have	I	the	a
I	0	0	0	0	1	0	0
have	0	0	0	1	0	0	0
a	0	0	0	0	0	0	1
cat	1	0	0	0	0	0	0

I	1.2	-0.4	3.2	2.1	0.1	-2.4	0.2
have	4.3	2.2	-2.2	1.1	-3.1	-1.1	1.5
a	-0.2	-0.5	-0.1	0.5	0.9	-1.2	2.4
cat	1.6	-3.4	-1.2	2.5	-0.1	3.8	4.8

Fig. 4.3: Flow differences between one-hot encoding and embedding

then integrate three outputs from FP, OCT-h, and OCT-v with the concatenation layer. The extracted features are flattened into one dimensional vector (Figure 4.1b-3).

4.3.4 Combination of Layers

The data handled in the proposed design are images and patient profile data. The original size of image data per patient is 480 × 480 × 3 pixels, and the features extracted by DenseNet are 7 × 7 × 1024 dimensions. Since this feature is added by three types of images and converted to one dimension for joining, the number of dimensions is the result of multiplying each. On the other hand, patient profile data has an original size of 1 × 5 dimensions, and is expanded to 100 or 500 dimensions by the embedding and FC layers. Each extracted feature may differ significantly in the number of dimensions. If the number of dimensions of the obtained features is significantly different, only the feature with the larger number of dimensions will be trained. Accordingly, prediction accuracy may be reduced.

To adjust for the dimensions of features, the model equalizes their dimensionality in two steps. First, the output from image feature extraction layers is reduced so that its

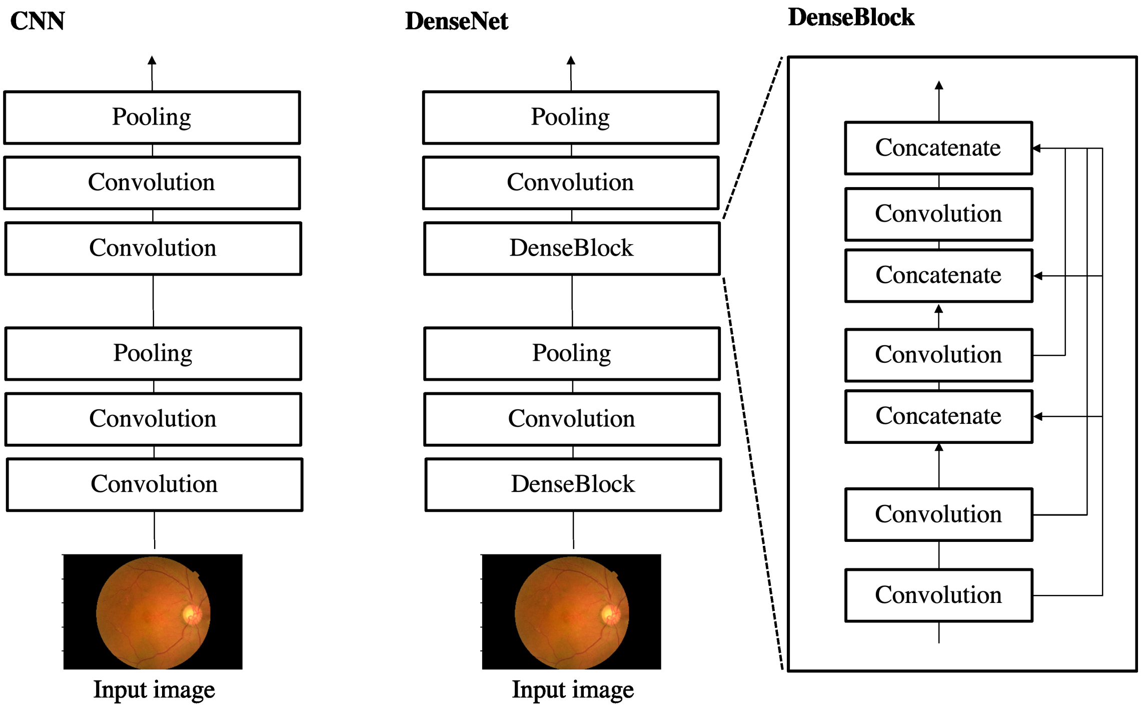


Fig. 4.4: Flow differences between CNN and DenseNet

dimensions match those of output from patient profile feature extraction layers. Next, the model concatenates the image and patient profile features together. Several FC layers are used to reduce the output from the image feature extraction layer. Gradually reduce the number of dimensions to keep the main image features. To reduce the output features to 500 dimensions, FC layers with dimensions of 1000, 500, 500 were used. To reduce the output features to 100 dimensions, FC layers with dimensions of 1000, 500, 100, 100 were used. Image features and patient profiles are concatenated by arranging one-dimensional feature arrays side by side. Concatenate layer was used for concatenation. Figure 4.5 shows an image of data concatenation in the concatenate layer, which concatenates data in a specified dimensional direction. For example, two-dimensional data such as an image can be concatenated in either the vertical or horizontal direction (Figure 4.5b). In the proposed design, since the features are one-dimensional, they are concatenated side by side, as shown in Figure 4.5a.

Finally, the concatenated result is connected to the output layer. Since the purpose of the proposed design is to predict postoperative VA numerically, the task to be solved is a regression task. Accordingly, the dense layer is utilized.

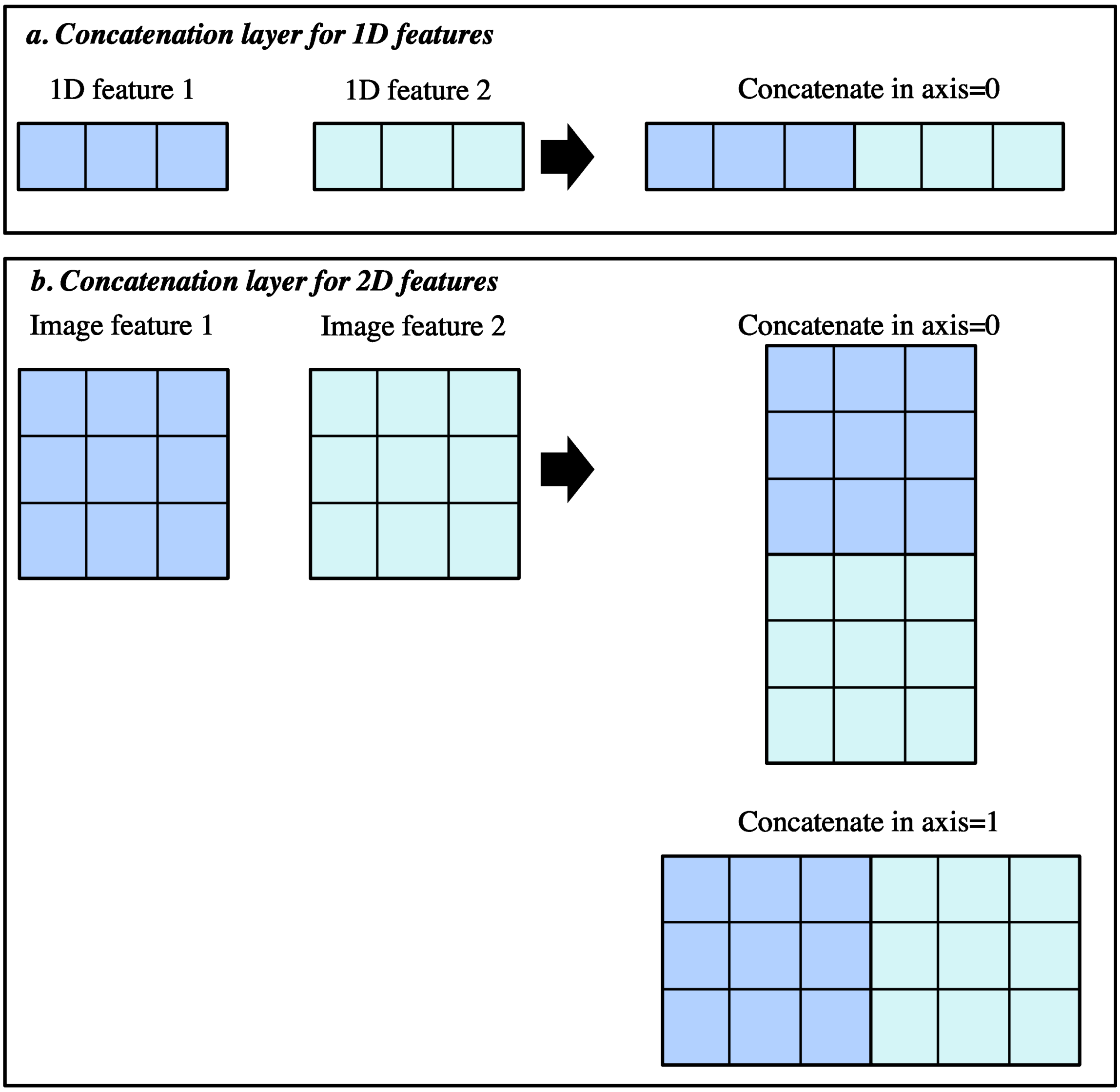


Fig. 4.5: Overview of Concatenate layer

4.4 Evaluation

4.4.1 Evaluation Task

We describe experiments to test the effectiveness of our proposed method. We prepared five experimental conditions and compared differences in MSE between predicted and outcome data.

Table 4.1 shows the five test model conditions. We prepared three models to compare the output accuracy among different input types, including single media and multimedia inputs.

Model OI (only image), which receives only image input data, and model OP (only

Table 4.1: Models for evaluate integration design

Model Name	Input Data	Embedding Class	Number of Dimensions
OI	image		
OP100	profile	20	
OP500	profile	100	
All-in100	both	20	100:100
All-in500	both	100	500:500

patient), which receives only patient profile input data, had FC layers after the component described in Section. 4.3.2 and Section. 4.3.3 (Figure 4.6; OI, OP100 and OP500). Therefore, OI received FP, OCT-h, and OCT-v, and output predicted logMAR, while OP received patient profile data and output predicted logMAR. Model All-in, which receives both image data and patient profile data, had layers 4.3.2 – 4.3.4. Thus, All-in received FP, OCT-h, OCT-v, and patient profile data, and output predicted logMAR. With these models, it was possible to compare accuracy under three conditions, namely when only image data are input, when only patient profile data are input, and when both types of data are input.

Two conditions were also prepared in OP and All-in. The two conditions differed in the number of embedding classes and the number of dimensions when combining features. One condition had 20 embedding classes and 100 dimensions when combining (Figure 4.6; OP100 and All-in100), and the other had 100 classes and 500 dimensions when combining (Figure 4.6; OP500 and All-in500). These models allowed us to compare the accuracy of predictions due to differences in the number of dimensions combined and the number of dimensional extensions of patient profile data.

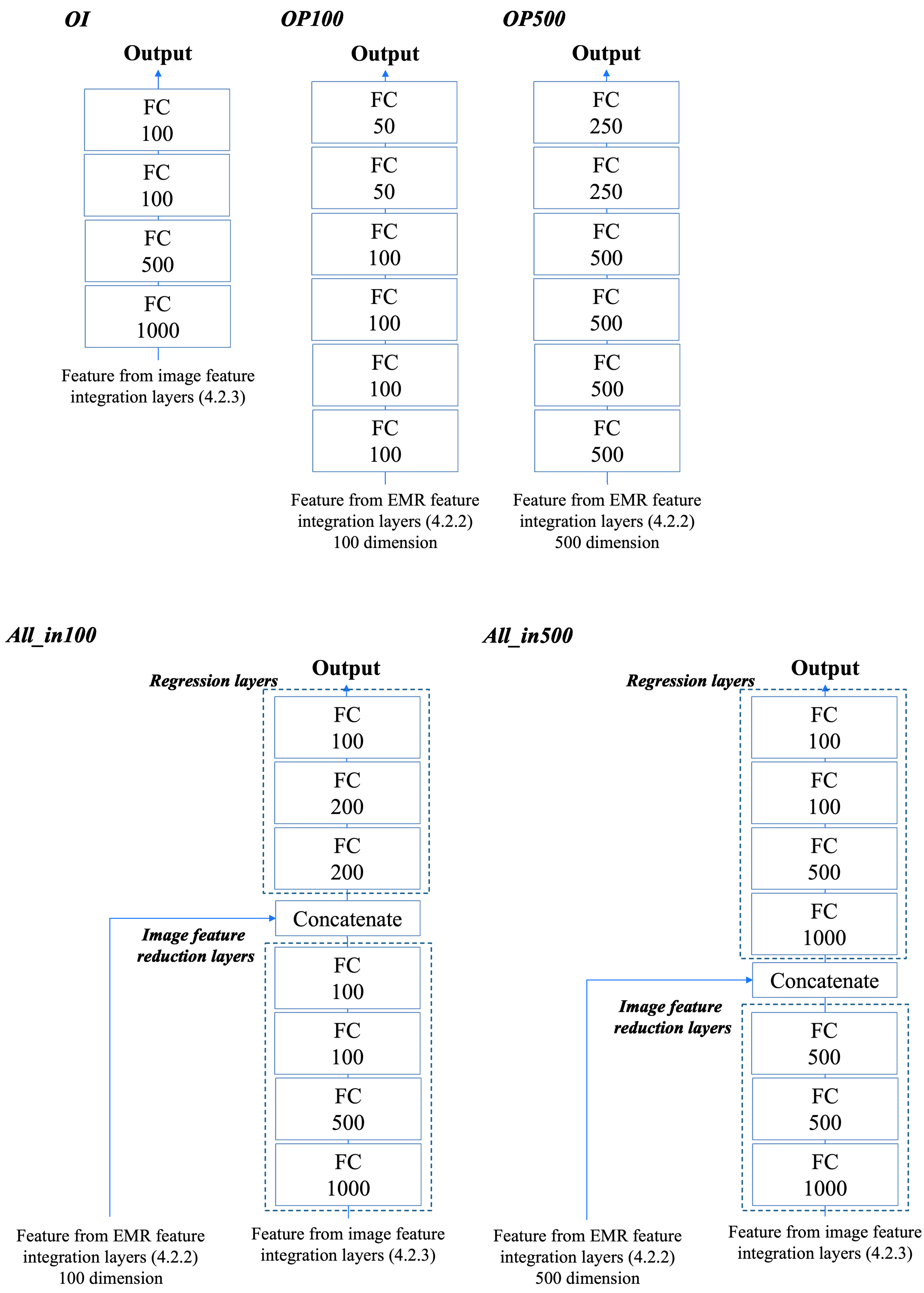

OI
Output
FC 100
FC 100
FC 500
FC 1000
Feature from image feature integration layers (4.2.3)
OP100
Output
FC 50
FC 50
FC 100
FC 100
FC 100
FC 100
Feature from EMR feature integration layers (4.2.2) 100 dimension
OP500
Output
FC 250
FC 250
FC 500
FC 500
FC 500
FC 500
Feature from EMR feature integration layers (4.2.2) 500 dimension
All_in100
Regression layers
Output
FC 100
FC 200
FC 200
Concatenate
Image feature reduction layers
FC 100
FC 100
FC 500
FC 1000
Feature from EMR feature integration layers (4.2.2) 100 dimension
Feature from image feature integration layers (4.2.3)
All_in500
Regression layers
Output
FC 100
FC 100
FC 500
FC 1000
Concatenate
Image feature reduction layers
FC 500
FC 500
FC 1000
Feature from EMR feature integration layers (4.2.2) 500 dimension
Feature from image feature integration layers (4.2.3)

Fig. 4.6: Overview of evaluation model about integration design

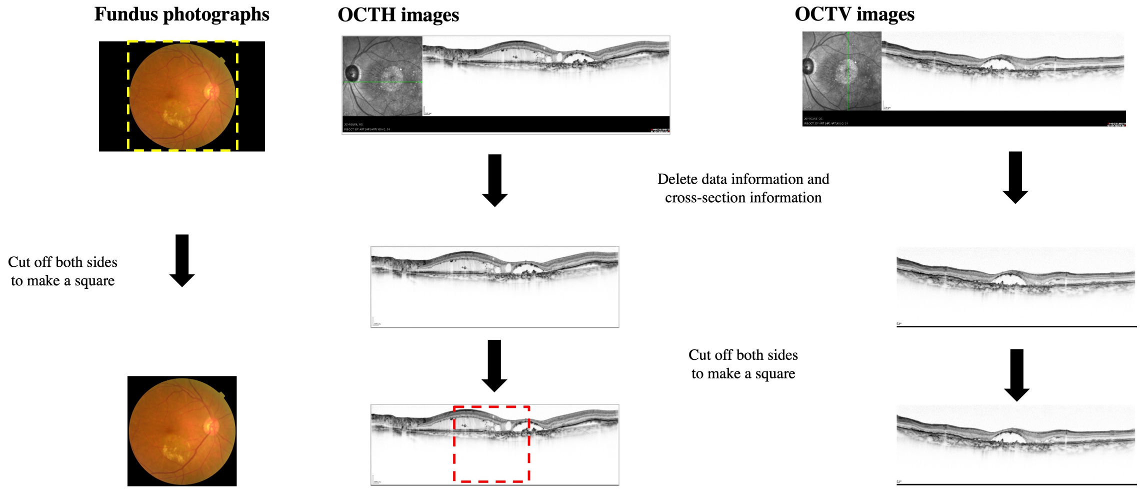

Fundus photographs
OCTH images
OCTV images
Delete data information and cross-section information
Cut off both sides to make a square
Cut off both sides to make a square

Fig. 4.7: Sample of input image and preprocessing

4.4.2 Data for Evaluation

Figure 4.7 shows examples of input imaging data, and how they were preprocessed, including FP, OCT-h images, and OCT-v images. FP are originally rectangular, but since DenseNet can only handle square images, the sides are cropped to render squares of size 480 × 480 pixels. Likewise, OCT images originally include a section which details the camera, date, and cross-section orientation-the latter can be seen as a green line in the examples in the figure. This section is cropped, then the OCT image is further cropped resulting in a square of size 480 × 480 pixels.

The patient profile data includes gender, age, affected side (right or left), pretreatment decimal VA [29], and logMAR. The patient profile data are converted according to the method described in Section. 4.2.

This study was approved by Ethics Committee of Kyoto University Graduate School and Faculty of Medicine (approval number R2366) and adhered to The Ethical Guidelines for Medical and Health Research Involving Human Subjects in Japan as well as the tenets of the Declaration of Helsinki. Informed consent was obtained using an opt-out method agreed by the mentioned Ethics Committee.

The dataset included 315 data samples from patients who were diagnosed with wet AMD at Kyoto University Hospital macular clinic and completed a regimen of intravitreal injection of aflibercept for one year. Cross-validation to avoid overfitting was performed since the sample number was relatively small. The 315 samples were separated into 295 for five-fold cross-validation, and 20 for testing. That is, five-fold cross-validation was performed on 295 samples.

Table 4.2: MSE of evaluation models

Model Name	MSE for Training data	MSE for Validation Data	MSE for Test Data
OI	0.043	0.074	0.081
OP100	0.060	0.079	0.058
OP500	0.048	0.098	0.052
All-in100	0.097	0.103	0.082
All-in500	0.020	0.063	0.047

4.4.3 Results

Table 4.2 shows the five-measurement averaged MSE between actual and predicted logMAR VA for each model. All-in500 yielded the best results among the models with the smallest MSE. For test data, both OP models and All-in500 yielded smaller MSE than did OI. Regarding the differences among embedding classes and the number of dimensions when combined, MSE of All-in500 was smaller than that of All-in100 for training, validation, and test data. MSE differed little between the two OP models within training, validation, and test data.

Regardless of number of dimensions when combined with embedding class, MSE decreased in the order of OI, OP, and All-in with the exception of All-in100, for test data. The All-in500 model resulted in a smaller MSE under all conditions, providing the best prediction accuracy for test data. OI was accurate for training data, but its prediction accuracy for test data was low, indicating low versatility. OP was accurate, but All-in500 was more accurate for training data and slightly more accurate for test data. MSE was well below 0.2 logMAR in both All-in100 and All-in500 for test data. According to previous studies [27, 52], if the MSE of logMAR exceeds 0.2, the clinical intervention can be regarded as significantly effective. Based on the results presented, our proposed model provides sufficient prediction accuracy.

4.4.4 Discussion

One advantage of a deep neural network is that it can efficiently integrate learning processes using multimedia feature extractors, weighted feature addition, and a regression converter. In comparison, a shallow model requires each function to be created separately. Moreover, manual intervention may be required in the latter [43]. Meanwhile, a disadvantage of the deep neural network is that we cannot explicitly determine the respective magnitudes of influence of the image and patient profile features or their extraction algorithms.

Since the relative impact of image and patient profile features is unknown, adopting our novel strategy of using non-image data in combination with image data takes advantage of both methods, suiting our purpose. Additionally, this approach allows us to extend the model to include additional patient profile features, increase the amount of information available to make predictions, and potentially improve its accuracy.

Our proposed model adopts a concatenation layer to equally utilize image and patient profile features. The image and patient profile features are extracted in equal dimensional orders. Especially for patient profile feature extraction, we adopt embedding layers that convert inputs to certain density vectors to extract scalar and categorical values in the patient profile data (see subsequent discussion for detail).

To validate the above concept, we set up two conditions for the concatenation layer in the experiment. One concatenated the features in 100:100 dimensions and the other in 500:500 dimensions. In experiments the 100:100 condition yielded a larger MSE for the test data than the 500:500 condition due to loss of image features with excessive dimensional reduction by image feature reduction layers. However, the MSE of 500:500 condition was smaller than that of other models for training, validation, and test data. Therefore, the experimental results indicate that this concatenation process utilizing multimedia data is useful to predict posttreatment VA.

Preprocessing the patient profile data is essential in our proposed model. As described above, we adopt an embedding layer to convert scalar and categorical values of patient profile data into dense vectors of certain sizes. This conversion contributes to the latter process in concatenating the features equally. Additionally, we adopt logMAR order to represent VA, which is intuitive for measuring differences in VA based on ophthalmologists' opinions. However, the embedding layer may not always convert data that are related to similar expressions [9]. Reliable conversion of the consultation data is essential to extract valid patient profile data features.

4.5 Summary

This chapter proposed a deep neural network model that can predict posttreatment VA given multimedia data of patients treated with anti-vascular endothelial growth factor drugs. The proposed model receives pretreatment FP, OCT-h, and OCT-v image data, and patient profile data as inputs and outputs predicted posttreatment VA. In the proposed model, features from each input mode are extracted separately and then combined. To ameliorate the effect of any difference in numbers of dimensions of input data, the model adjusts the numbers of dimensions and features accordingly.

To verify the effectiveness of multimedia input and the numbers of dimensions when combined, we compared the MSE of predicted VA using the data of OI, OP100, OP500, All-in100, and All-in500. We found that for test data, MSE was 0.081 for OI, 0.058 for OP100, 0.052 for OP500, 0.082 for All-in100, and 0.047 for All-in500. These experiments reveal that All-in100 and All-in500 give sufficient prediction accuracy because MSE is below 0.2 log MAR, a criterion for positive treatment outcome according to previous studies.

The main contributions of this proposed design include the following. It can predict posttreatment VA with high accuracy from multimedia input to support physicians and patients as a treatment guide. It showed a method of effective feature integration when working with images and patient profile data. Since the same image and patient profile data are often used in different medical scenarios, this multimedia method may be applied to different predictors and diagnostic models that use similar data.

These contributions have an important impact in the design stages of developing a CDSS development. The proposed design method was effective for feature integration of images and patient profile data. The design method avoids some of the features obtained from multimedia medical data from being eliminated by differences in the numbers of dimensions. We showed an effective design method for the issues to be considered in determining the structure of the classifier in the design stage of creating a model.

Chapter 5

Auto Generation of Multimodal Medical Encoder

Chapter 5 proposes basic algorithms for the automatic MME generation system based on proposed design methods for input and output data for diagnosis support in the medical field.

The previous chapters presented a deep learning model design method that focused on numerical and image data among multimedia medical data. The model was designed to extract features from multimodal medical data and predict prognosis. The verification results in Chapters 3 and 4 showed the optimum model design method for numerical data and image data. If a model could be automatically generated according to input data, physicians could develop a prototype of CDSS instantly without needing specialist deep learning knowledge.

It is necessary to discriminate input data and apply the design according to data type for automatic model generation. First, when the model receives input data, it determines whether it is a numerical value or an image based on its format and form of array. Next, the model sets the input layers to suit the data. Then the user selects the purpose as diagnosis or prognosis. Finally, the model forms the output layer according to the selected purpose. The proposed design also allows physicians to extract appropriate data features without requiring any specialized knowledge of the disease with which they are dealing. Users can also use a model that appropriately learns to integrate medical data.

This chapter focuses on numerical and image data frequently used to diagnose eye diseases. First, the pertinent medical field numerical and image data are summarized. Then, a deep learning model corresponding to each data type is described. Next, a system design that understands the input data and the output purpose is described. Next, diagnostic data for AMD is processed into the type of data required. The diagnostic data were described in Chapters 3 and 4. Then, a model is generated by the proposed system using the data. Next, the generated model is verified and discussed to be appropriately designed according to the input data. Finally, the automatically generated model is shown to be as accurate as the other models proposed in this paper.

5.1 Input of Multimodal Medical Encoder

5.1.1 Input type

Chapter 2 mentioned that the data handled by clinicians includes text data such as patient interview records, findings of imaging tests, and numerical data results of sample tests such as blood tests and biochemical tests. Data in clinical practice also includes images from X-ray fluoroscopy, and CT and MRI examinations, plus video data from abdominal ultrasonography and endoscopy. Physicians integrate these multimedia medical data and use them for diagnosis.

This study focused on numerical and image data often used to diagnose eye diseases. This chapter describes how to automatically generate models that can handle these multimedia medical data. It is necessary to prepare an input layer for each data and prepare a layer for preprocessing and feature extraction when handling multimedia medical data in a deep learning model.

In clinical practice, the primary purpose of physicians using multimedia medical data is to diagnose. However, it also includes prognosis, treatment decisions, and confirmation of treatment effects. The deep learning model also requires a design that returns appropriate output according to purpose by changing the structure of the output layer.

5.1.2 Layer Component Assignment corresponding to Input Types

Many basic models have already been proposed as deep learning models for handling numerical values and images in multimedia medical data. Chapter 3 mentioned that numerical data used to be manually min-max normalized and age data categorized into ten year vectors before data entry. However, it is possible to automatically convert it to a categorical vector as an internal process using BN as a layer of a deep learning model or by using softmax layer. The numerical data once converted is often feature-extracted in the fully connected layer. For image data, the image size and shape is transformed manually with unnecessary parts cropped by manual preprocessing. In addition, color tone, lightness, and darkness, were sometimes adjusted. Resizing, cropping, and processing the input image by dividing it into patches can be achieved using ViT as described in Chapter 3. It is also possible to perform processing such as automatically determine the color tone of an image and perform binarization by a program. The image identification model has been improved since the proposal of CNN for image identification. MobileNet and Inception v3 were proposed as deeper and more accurate models. In addition, DenseNet was proposed to minimize information loss during backpropagation. Many partial object recognition models have also been proposed to detect what object is in an image instead of identifying the image. Typical models include R-CNN, YOLO, and SSD (single shot multibox detector). Previous studies [41,62] used partial object recognition models to identify disease sites and tumors from medical images such as CT and MRI.

5.2 Conceptual Design for Auto Generation System

Figure 5.1 gives an overview of our proposed auto-generation system. To create a system that receives input data and information about the purpose in order to output a design of an optimal deep learning model requires mechanisms to interpret the input data, interpret the purpose, create an input layer, create the intermediate layer, and create an output layer according to the purpose.

The pattern of data to be input is defined as a mechanism for interpreting input data and creating an input layer corresponding to the input data. In addition, input layers are prepared corresponding to each in advance.

First, numerical test results such as those of test samples are passed as int (integer) type or float (floating point number) type data. Two patterns are assumed for the number of dimensions of the passed data. In one, only the inspection result is passed as a one-dimensional array. In the other, the inspection result as a string item is used as a key, and the result of inspection by int type or float type value is passed as a dictionary type. Since the range of data differs by inspection item, various numerical data may be passed.

Next, a three-dimensional int-type array is passed as image data. Image data have the content of vertical width, horizontal width, and three RGB channels. When the image data has three RGB channels, the numerical value for each pixel is between 0 and 255. By determining that the data content is between 0 and 255, the data are interpreted as images.

Users select an output format according to the purpose as a mechanism for interpreting the purpose and creating an output layer according to the purpose. The model prepares an appropriate output layer according to each option. The output objectives targeted in this study can assume multiple patterns such as diagnosis and prognosis. Still, in the case of a

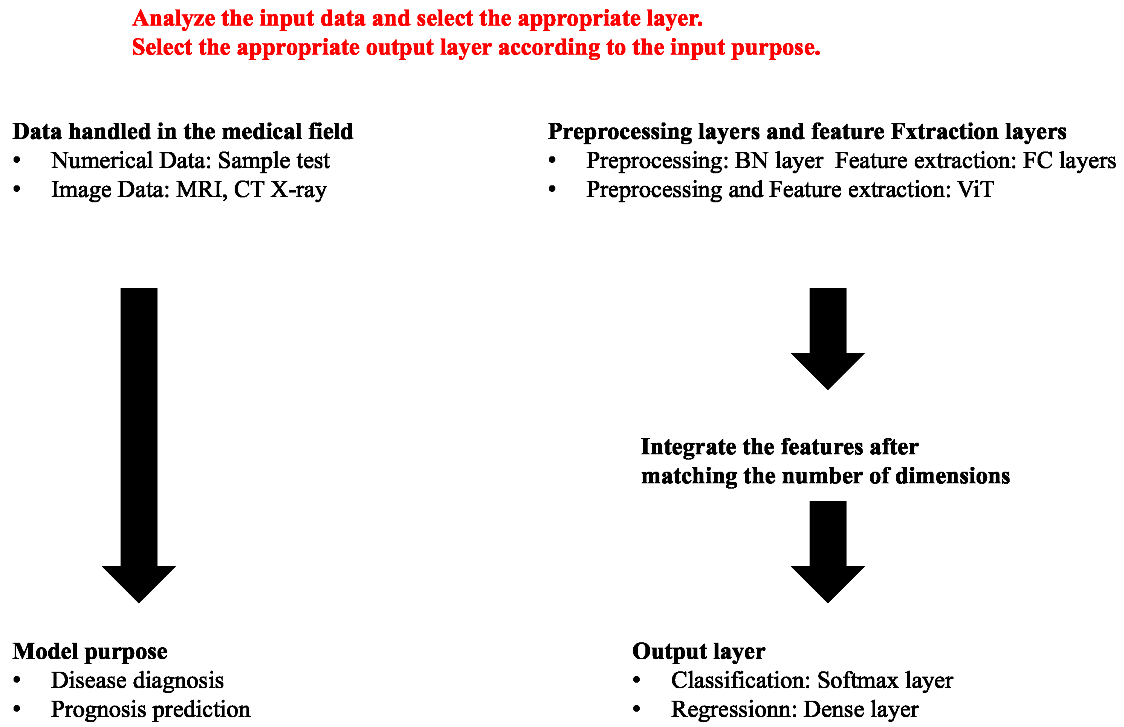

Analyze the input data and select the appropriate layer.
Select the appropriate output layer according to the input purpose.
Data handled in the medical field
• Numerical Data: Sample test
• Image Data: MRI, CT X-ray
Preprocessing layers and feature Fxtraction layers
• Preprocessing: BN layer Feature extraction: FC layers
• Preprocessing and Feature extraction: ViT
Integrate the features after matching the number of dimensions
Model purpose
• Disease diagnosis
• Prognosis prediction
Output layer
• Classification: Softmax layer
• Regressionn: Dense layer

Fig. 5.1: Overview of auto generation system

deep learning model, the output is either categorical or non-categorical. In other words, if it is a diagnosis for a specific disease, it will output a categorical vector showing a probability from 0 to 1 for that disease. When dealing with a regression problem that predicts a numerical value, such as postoperative VA prediction, that non-categorical numerical value is output. To make it easier for the user to make a judgment, users select their purpose from the option of diagnosis or prognosis. As an internal design, these options determine whether it is categorical or not. Two patterns of output layers are prepared according to the purpose input.

Creating the intermediate layer mechanism includes the model designs proposed in Chapters 3 and 4 which are prepared according to the input data. In other words, image data is processed by the ViT layer, and numerical data by the BN and FC layers. When combining the features obtained from each data, a mechanism is adopted in which the combination is performed after the number of dimensions of the features is adjusted to 1: 1.

5.2.1 Conceptual Flow of Auto Generation

This section shows the conceptual flow of the system that automatically generates the MME.

First, line 1 to line 2 in algorithm 1 indicate the user's data input. The user inputs multimedia medical data X to be handled by the system and the purpose Y of the model. The input data is assumed to be array type numerical data, dictionary type numerical data, and array type image data. The objective of output is diagnosis or prognosis.

Next, the system interprets X by Function 1. Function 1 processing content is shown in Section 5.2.2.

Then, as shown in line 4 to line 9 of algorithm 1, an input layer, a preprocessing layer, and a feature extraction layer according to data type are added in the model. In the case of numerical data, the pretreatment layer separates numerical data by type, and feature extraction is performed by the BN layer and fully connected layers proposed in this study. Next, in the case of image data, the input layer receives the image data as it is used which is then divided into patches and the features extracted by ViT.

Next, as shown in the line 15 of the algorithm 1, the system adds to the model layers that match the number of dimensions of the extracted features and a layer that integrates. First, the number of dimensions of the model is determined whether the input data is image data or numerical data. Next, the number of dimensions of the extracted features is increased in the case of image data, and it is decreased in the case of numerical data. Finally, increasing/decreasing the number of dimensions is repeated until the output of each model has the same number of dimensions. The feature integration layer integrates the outputs of each layer with the same number of dimensions side by side.

Next, the system interprets Y by the Function 2. The processing contents of Function 2 are shown in Section 5.2.3.

Then, as shown in line 17 to line 20 of algorithm 1, an output layer corresponding to the data type is added to the model. In the case of $y1$, the model prepares a softmax layer which calculates the probability of each diagnosis and outputs that with the highest probability. In the case of $y2$, the model prepares a one-dimensional dense layer which outputs a numerical value converted into a unit according to the purpose. Finally, the generated input layer, feature extraction layer, and output layer are combined to generate one MME.

Finally, the generated input layer, feature extraction layer, and output layer are combined to generate one multimodal medical encoder.

The rest of this section summarizes the functions that each interpret the input and the

Algorithm 1 Main process of auto generation system

```
Require: $X$, list of input data $Y$, Purpose
Ensure: $M$, Multimodal Medical Encoder
$X \leftarrow x1, x2, x3$, User inputs data
$Y \leftarrow y1$ or $y2$, User inputs purpose
$IL \leftarrow Fun1(X)$
for all $i$ in $IL$ do
    if $i$ is dict image or $i$ is array image then
        $M \leftarrow$ layer for image
    else
        $M \leftarrow$ layer for numerical data
while all $m$ in $M$ is same number of dimension do
    for all $m$ in $M$ do
        if $m$ is layer for image then
            $M \leftarrow$ layer to decrease no. of dimensions of output of $m$
        else
            $M \leftarrow$ layer to increase no. of dimension of output of $m$
$M \leftarrow$ layer to integrate all model in $M$
$OL \leftarrow Fun2(Y)$
if $OL$ is categorical then
    $M \leftarrow$ Softmax layer
else
    $M \leftarrow$ Dense layer
return $M$
```

output data.

5.2.2 Interpretation of input data

This section assumes that data [$x1$, $x2$, $x3$] is input, and the processing for each input data is explained using pseudo-code. For each input data item, $x1$ is array type numerical data, $x2$ is dictionary type numerical data, and $x3$ is image data.

First, as shown in line 1 of algorithm 2, each input data is separated. In other words, the obtained input data is separated in the form of [$x1$], [$x2$], [$x3$]. As a method of dividing, the outermost large array is assumed as one data block in order from the front, and each large array is interpreted as different data.

Next, the type of the separated data is interpreted as shown in line 2 to line 11 of algorithm 2. Each data type can be divided into patterns. If-then rules are used to determine which pattern each data applies to. There are two question patterns to be judged by the if-then rule which are, is the data type dictionary or array type? And does the data array have the same number of elements and is the data range 0-255? First,

Algorithm 2 Interpret input data

```
Require: list $X$, include multiple $x$ and data type is $x1$, $x2$ or $x3$
Ensure: list $IL$, list of input data types
for all $x$ in $X$ do
    if $x$ is dict then
        if $0 < x$.value$< 255$ then
            $IL \leftarrow$ dict image
        else
            $IL \leftarrow$ dict numerical
    else
        if $0 < x < 255$ then
            $IL \leftarrow$ array image
        else
            $IL \leftarrow$ array numerical
return $IL$
```

depending on whether the data is dictionary or array type, it is possible to classify it into two patterns. Next, suppose the range of the data is 0-255. In that case, whether the data is numerical data or image, data of the inspection results arranged in chronological order can be determined. This is because the range of image data does not differ between dictionary and array type. All patterns for which input can be expected can be determined by determining these in stages.

Finally, the types of input data are combined into one array and returned. The model produces input layers and feature extraction layers depending on the contents of this array.

5.2.3 Interpretation with the model purpose

This section describes the mechanism to interpret the selected output purpose and set the output layer using a concrete pseudo-code. The data given for the output purpose is [$y1$, $y2$]. It is assumed that $y1$ is the diagnosis result and $y2$ is the prognosis.

First, the output purpose is interpreted as shown in line 1 to line 4 of algorithm 3. The Y received by the model is either $y1$ as diagnosis or $y2$ as prognosis selected by the user. Next, the system determines whether the input Y is $y1$ or $y2$. In addition, $y1$ is set as categorical output and $y2$ is set as non-categorical output in the system in advance and the function outputs the determination result accordingly. The automatic generation system generates an output layer according to this return value.

Algorithm 3 Interpret purpose

```
Require: $Y$, data type of y is $y1$ or $y2$
Ensure: $OL$, whether Y is categorical or non-categorical
if $Y$ is $y1$ then
    $OL \leftarrow$ categorical
else
    $OL \leftarrow$ non-categorical
return $OL$
```

5.3 Auto Generation of Multimodal Medical Encoder

Algorithm 4 shows the sequence of MME automatic generation. Algorithm 4 outputs the *Tensor* object, which represents a partially defined computation that will eventually produce a value. The concept of *Tensor* object has been introduced into popular deep learning programming packages, such as tensorflow and pytorch. The program with the tensor object first builds a graph of *Tensor* objects, details how each tensor is computed based on the other available tensors, and then runs parts of this graph to achieve the desired results.

Algorithm 4 Auto generation of Multimodal Medical Encoder

```
Require: $X$, list of input data $Y$, model's objective
Ensure: $M$, a tensor containing computation of Multimodal Medical Encoder
$X \leftarrow \{x\}_i^N$ , User input the model's objective
$Y \leftarrow y$, User output the model's objective
$inputs \leftarrow Input(X)$
Declare $EARY$ as an array for each input layer
Declare $dim$ as a particular dimension size
for $i$ in $inputs$ do
    Declare $E$ as a single-modal encoder
    if $i$ is dict image or $i$ is array image then
        $E \leftarrow$ VisionTransformer($i$)
    else
        $E \leftarrow$ EMREncoder($i$)
    $EARY$.append($E$)
$M \leftarrow$ Concatenate($EARY$)
$M \leftarrow$ mlp($M$)
if is_categorical($y$) then
    $M \leftarrow$ Softmax($M$)
else
    $M \leftarrow$ Dense($M$)
return $M$
```

In algorithm 4, *Dense* implements the operation eq.(5.3.1),

$$\mathbf{y} = \eta(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{5.3.1}$$

where η is the element-wise activation function, $\mathbf{W}$ is a weights matrix created by the layer, $\mathbf{b}$ is a bias vector created by the layer. *Dense* layer is known as the full-connection layer in the neural network.

Concatenate is the layer that concatenates a list of inputs. It takes a list of a tensor as input, all of the same shape except for the concatenation axis, and returns a single tensor that is the concatenation of all inputs.

MLP is an implementation of Multi-layer perceptron, which is the piles of *Dense* and *Dropout* layers (See Section 5.3.1).

Finally, the *Softmax* layer is an implementation of the softmax activation function, also known as softargmax or normalized exponential function. It is a generalization of logistic function to multiple dimensions shown as eq. (5.3.2),

$$\sigma(\mathbf{z}_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e_j^z} \qquad for \quad i = 1, \ldots, K \quad and \quad z = (z_1, \ldots, z_K) \in \mathbb{R}^K \tag{5.3.2}$$

It applies the standard exponential function to each element z_i of the input vector $\mathbf{z}$ and normalizes these values by dividing by the sum of all these exponentials. This normalization ensures that the sum of the components of the output vector $\sigma(\mathbf{z})$ is 1, which is also used to determine whether or not $\mathbf{y}$ is a categorical vector.

VisionTransformer is a function that implements ViT model which was proposed by Alexey Dosovitskiy et al. [11] for image classification. ViT applies transformer architecture with self-attention to sequences of image patches without using convolutional layers. A detailed-implementation of ViT is described in Section 5.3.2. *EMREncoder* is a function that implements the feature extraction operation from a single scalar value. An in-detailed implementation of *EMREncoder* is shown in Section 5.3.3.

5.3.1 Implementation of Multi-Layer Perceptron

Algorithm 5 shows implementation of a multi-layer perceptron.

mlp function in this system is implemented as a pile of *Dense* function, as shown in eq.(5.3.1), and Dropout function. The *Dropout* layer randomly sets the input units to zero with the frequency of *dropoutrate* at each step during training time. This operation helps

Algorithm 5 Implementation of Multi-Layer Perceptron

```
function MLP($x, outdim, nlayers$)
    Declare $x$, as a tensor variable of the neural network model.
    Declare $outdim$, as a output dimension size.
    Declare $nlayers$, as a number of layers.
    Declare $dropoutrate$, as a rate variable in dropout operation.
    for $n$ in $nlayers$ do
        $x \leftarrow$ Dense($outdim$)($x$)
        $x \leftarrow$ Dropout($dropoutrate$)($x$)
    return $x$
```

prevent overfitting. Inputs not set to zero are scaled up by $\frac{1}{1-dropoutrate}$ such that the sum is unchanged.

Function *mlp* is commonly utilized for each algorithm in $VisionTransformer$, $EMREncoder$, and our proposed auto-generation main procedure.

5.3.2 Implementation of Vision Transformer

Algorithm 6 shows implementation of ViT.

The $VisionTransformer$ is a pile of multiple transformer blocks, which use multihead attention as a self-attention mechanism applied to the patch sequence. The transformer blocks produce a $[batch\_size, num\_patches, projection\_dim]$ tensor, which is processed via a classifier head with $softmax$ function (eq.(5.3.2)) to produce the final class probabilities output.

Algorithm 6 Implementation of Vision Transformer

```
function VisionTransformer($x$)
    Declare $x$, as a tensor variable for the neural network model.
    Declare $patchSize$, as a patch size for dividing the images into certain patches.
    Declare $numPatches$, calculated by $(imageSize//patchSize)^2$.
    Declare $projectionDim$, as a dimension size for the projection.
    Declare $numHeads$, as a headNumber in MultiHeadAttention.
    Declare $transformerLayers$, as the number of repeatition of transformer layers.
    $patches \leftarrow$ Patches($patchSize$)($x$)
    $encodedPatches \leftarrow$ PatchEncoder($numPatches, projectionDim$)($patches$)
    for $i$ in $transfomerLayers$ do
        $x1 \leftarrow$ LayerNormalization($e^{-6}$)($encodedPatches$)
        $attentionOutput \leftarrow$ MultiHeadAttention($numHeads, projectionDim$)($x1, x1$)    ▷ self attention
        $x2 \leftarrow$ Add()([$attentionOutput, encodedPatches$])
        $x3 \leftarrow$ LayerNormalization($e^{-6}$)($x1$)
        $x3 \leftarrow$ mlp(x3)
        $encodedPatches \leftarrow$ Add()($x2, x3$)
    $x \leftarrow$ LayerNormalization($e^{-6}$)($encodedPatches$)
    $x \leftarrow$ Flatten()($x$)
    $x \leftarrow$ Dropout(0.5)($x$)
    $x \leftarrow$ mlp($x$)
    return $x$
```

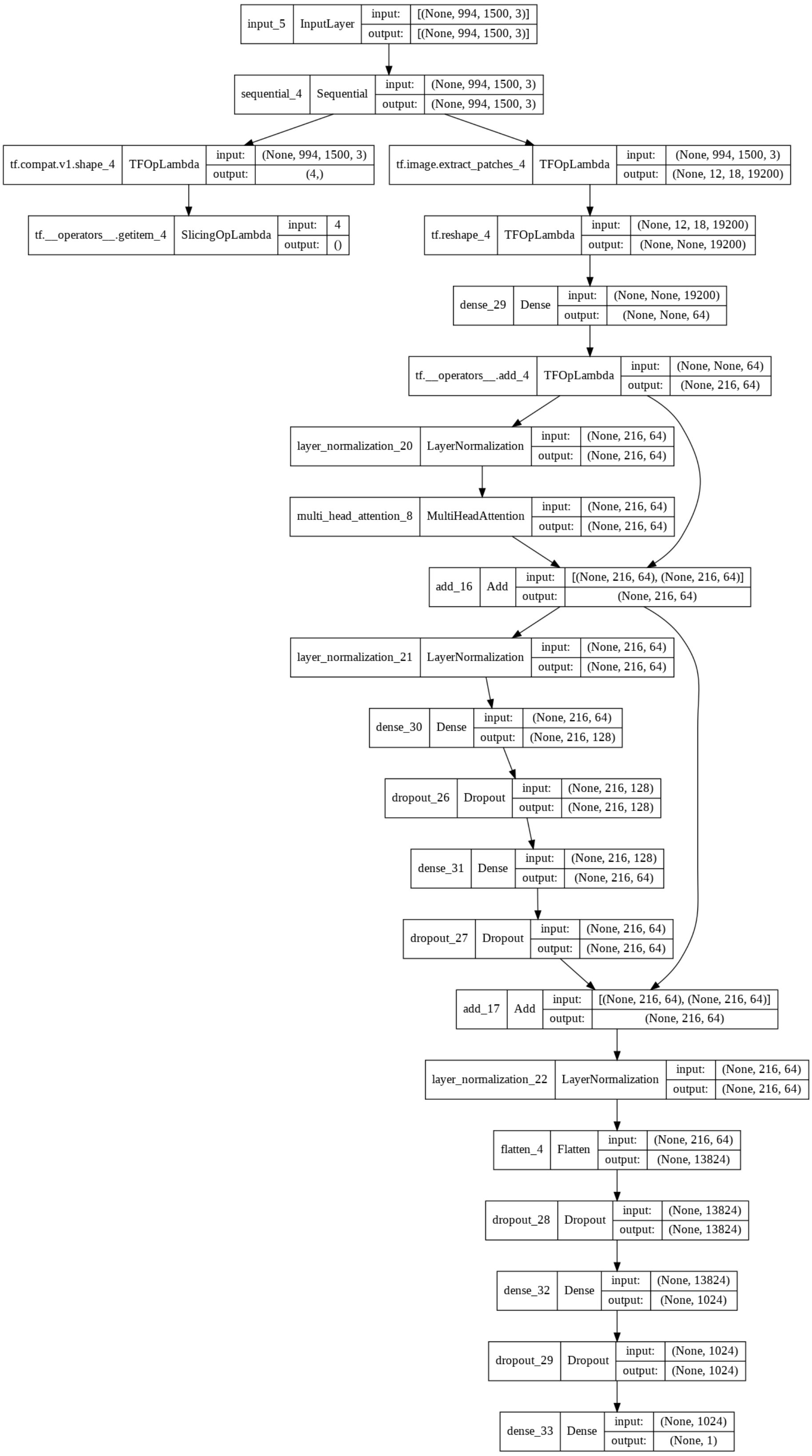

input_5 | InputLayer | input: [(None, 994, 1500, 3)] | output: [(None, 994, 1500, 3)]
sequential_4 | Sequential | input: (None, 994, 1500, 3) | output: (None, 994, 1500, 3)
tf.compat.v1.shape_4 | TFOpLambda | input: (None, 994, 1500, 3) | output: (4,)
tf.image.extract_patches_4 | TFOpLambda | input: (None, 994, 1500, 3) | output: (None, 12, 18, 19200)
tf.__operators__.getitem_4 | SlicingOpLambda | input: 4 | output: ()
tf.reshape_4 | TFOpLambda | input: (None, 12, 18, 19200) | output: (None, None, 19200)
dense_29 | Dense | input: (None, None, 19200) | output: (None, None, 64)
tf.__operators__.add_4 | TFOpLambda | input: (None, None, 64) | output: (None, 216, 64)
layer_normalization_20 | LayerNormalization | input: (None, 216, 64) | output: (None, 216, 64)
multi_head_attention_8 | MultiHeadAttention | input: (None, 216, 64) | output: (None, 216, 64)
add_16 | Add | input: [(None, 216, 64), (None, 216, 64)] | output: (None, 216, 64)
layer_normalization_21 | LayerNormalization | input: (None, 216, 64) | output: (None, 216, 64)
dense_30 | Dense | input: (None, 216, 64) | output: (None, 216, 128)
dropout_26 | Dropout | input: (None, 216, 128) | output: (None, 216, 128)
dense_31 | Dense | input: (None, 216, 128) | output: (None, 216, 64)
dropout_27 | Dropout | input: (None, 216, 64) | output: (None, 216, 64)
add_17 | Add | input: [(None, 216, 64), (None, 216, 64)] | output: (None, 216, 64)
layer_normalization_22 | LayerNormalization | input: (None, 216, 64) | output: (None, 216, 64)
flatten_4 | Flatten | input: (None, 216, 64) | output: (None, 13824)
dropout_28 | Dropout | input: (None, 13824) | output: (None, 13824)
dense_32 | Dense | input: (None, 13824) | output: (None, 1024)
dropout_29 | Dropout | input: (None, 1024) | output: (None, 1024)
dense_33 | Dense | input: (None, 1024) | output: (None, 1)

Fig. 5.2: Graph Model generated by Vision Transformer Algorithm

Patches function renders the images into patches of particular size with the given *patchSize* (See Section 3.3). *LayerNormalization* normalizes the activations of the previous layer for each given example in a batch independently, rather than across a batch like BN [26].

MultiHeadAttention is an implementation of multiheaded attention as described in [63], which is a module for self-attention mechanisms that run through an attention mechanism several times in parallel. The independent attention outputs are concatenated and linearly transformed into the given dimension. Definitions of *mlp* and *Dropout* functions are described in Section 5.3.1.

Figure 5.2 shows the graph structure generated by our proposed vision transformer generation algorithm, where both $transformerLayers$ in $VisionTransformer$ and *nlayers* in *mlp* function are set to be one for simplifying the structure. As shown in Fig. 5.2 The structure of ViT is not linear but is a complex graph model with several branches, which can be found in many high-performance models proposed in recent deep learning studies.

5.3.3 Implementation of EMREncoder

Algorithm 7 shows implementation of EMR encoder. It extracts features of EMR with the piles of *Dense* and *Dropout* layers, which have been defined.

Algorithm 7 Implementation of EMR Encoder

```
procedure EMRENCODER($x$)
    Declare $x$, as a tensor variable for the neural network model.
    $x \leftarrow$ BatchNormalization()($x$)
    $x \leftarrow$ mlp($x$)
    return $x$
```

$BatchNormalization$ is a function that applies a transformation that maintains the mean output close to zero and the output standard deviation close to one. The operation of $BatchNormalization$ with training is shown in eq.(5.3.3).

$$\mathbf{y} = \gamma \cdot \frac{batch - mean(batch)}{\sqrt{var(batch) + \epsilon}} + \beta \tag{5.3.3}$$

where, ϵ is a small constant, γ is a learned scaling factor (initialized as 1), and β is learned offset factor (initialized as 0).

Operation of $BatchNormalization$ without training is shown in eq.(5.3.4).

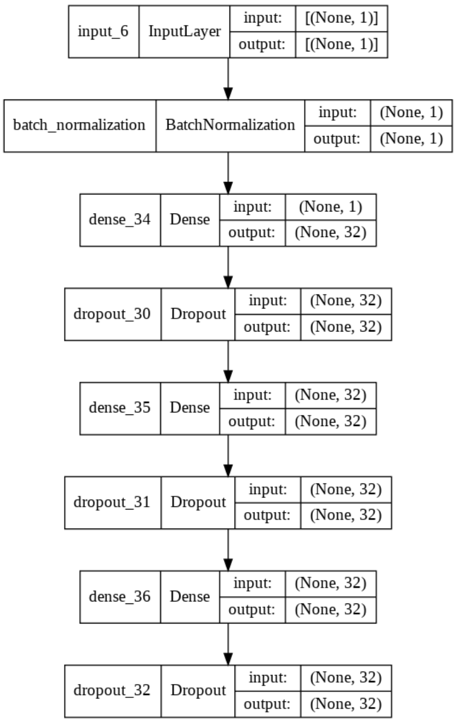


Fig. 5.3: Graph Model generated by EMR Encoder Algorithm

$$\begin{aligned}\mathbf{y} &= \gamma \cdot \frac{batch - moving\_mean}{\sqrt{moving\_var + \epsilon}} + \beta \qquad (5.3.4)\\ moving\_mean &= moving\_mean \cdot momentum + mean(batch) \cdot (1 - momentum)\\ moving\_var &= moving\_var \cdot momentum + var(batch) \cdot (1 - momentum)\end{aligned}$$

where, *moving_mean* and *moving_var* are non-trainable variables updated each time the layer is in training mode.

Figure 5.3 shows the graph structure generated by our proposed EMR encoder generation algorithm, where *nlayers* in *mlp* function is set at three to simplify the structure. As shown in Fig. 5.3, The structure of *EMREncoder* is linear compared to the graph structure of ViT. This is because these features are concatenated in the latter operations, and there is a list of features for each scalar of EMR.

5.4 Evaluation

5.4.1 Evaluation task

This section describes verification that the proposed automatic generation system works as expected. The accuracy of the generated model is also verified by comparing it with the proposed model in Chapters 3 and 4.

First, to verify whether the behavior is as expected, simulated case data is input to the system. When users select the output purpose, the system confirms the input data and model according to purpose are generated correctly. The automatically generated model is compared with the model using the design proposed in Chapter 3 in the task of postoperative VA prediction. By confirming similar accuracy, it is clarified that the automatically generated model meets the proposed design of this study.

Three patterns of input data were prepared including array type numerical data and image data, dictionary type numerical data and image data, array type numerical data, and dictionary type numerical data and image data. Each input pattern was of two patterns, one for diagnosis and the other for prognosis. That is, verification was performed in a total of six patterns. Three prognosis and three prediction models were used for accuracy verification.

5.4.2 Data for evaluation

AMD patient data used for verification in Chapters 3 and 4 was converted for input. Numerical data, decimal VA and logMAR used so far could be used as array type data. In addition, a dictionary type was prepared by adding “decimal VA” and “logMAR” and keys to the preoperative decimal VA and logMAR, respectively. Image data, vertical and horizontal cross-sectional FP and OCT image data used in verification so far could be used as is. With three channels of RGB, the FP are 994 pixels vertical and 1500 pixels horizontal, and the OCT images are 496 × 1532 pixels, respectively.

5.4.3 Results

Table 5.1 shows the combination of input data and generated model. Figure 5.4 also shows an overview of the generated models. In combination 1, the BN and FC layers are shown in Table 5.1: Models to evaluate integration design were selected for numerical data, and ViT was selected as pretreatment and feature extraction layers for image data. In combination 2, the models with the BN and FC layers were combined, and the ViT was selected. In combination 3, the model in which the BN and FC layers were selected for each

Table 5.1: Models for evaluate integration design

Model Name	Input Data	Purpose	Input Layers	Output Layer
Model 1	numerical array, image	Diagnosis	BN and FC, ViT	Softmax
Model 2	numerical array, image	Prediction	BN and FC, ViT	Dense
Model 3	numerical dict, image	Diagnosis	BN and FC, ViT	Softmax
Model 4	numerical dict, image	Prediction	BN and FC, ViT	Dense
Model 5	numerical array, numerical dict, image	Diagnosis	BN and FC * 2, ViT	Softmax
Model 6	numerical array, numerical dict, image	Prediction	BN and FC * 2, ViT	Dense

Table 5.2: MSE for auto generated models

Model Name	Input Data	MSE for train data	MSE for validation data
Chapter3 Model	numerical array, image	0.039	0.054
Model 2	numerical array, image	0.40	0.055
Model 4	numerical dict, image	0.060	0.057
Model 6	numerical array, numerical dict, image	0.042	0.055

numerical data, ViT was selected for image data. Output layer was selected corresponding to the purpose in each pattern. From the results, it can be said that the proposed system design could understand the input data correctly. In addition, the design was able to select a pretreatment layer and a feature extraction layer suitable for the input data.

Table 5.2 shows postoperative VA prediction errors of the model used for verification in Chapter 3 and that generated by the proposed system. Results with the smallest prediction error are underlined. The results showed that model 4 in which dictionary type numerical data and image data was input had the largest prediction error for both training and verification data. The MSE of model 4 for training data was 0.060 and that for validation data was 0.057. The prediction errors of the model in Chapter 3 and models 2 and 6 were about the same. The MSE of the model in Chapter 3 was 0.039 for training data and 0.054 for validation data, whereas that of model 2 was 0.040 for training data and 0.055

for validation data. The MSE of model 6 was 0.042 for training data and 0.055 for the validation data. The order of prediction error for training data was model 4 $>$ Chapter 3 model $\fallingdotseq$ model 2 $\fallingdotseq$ model 6. The order of prediction error for training data was the same.

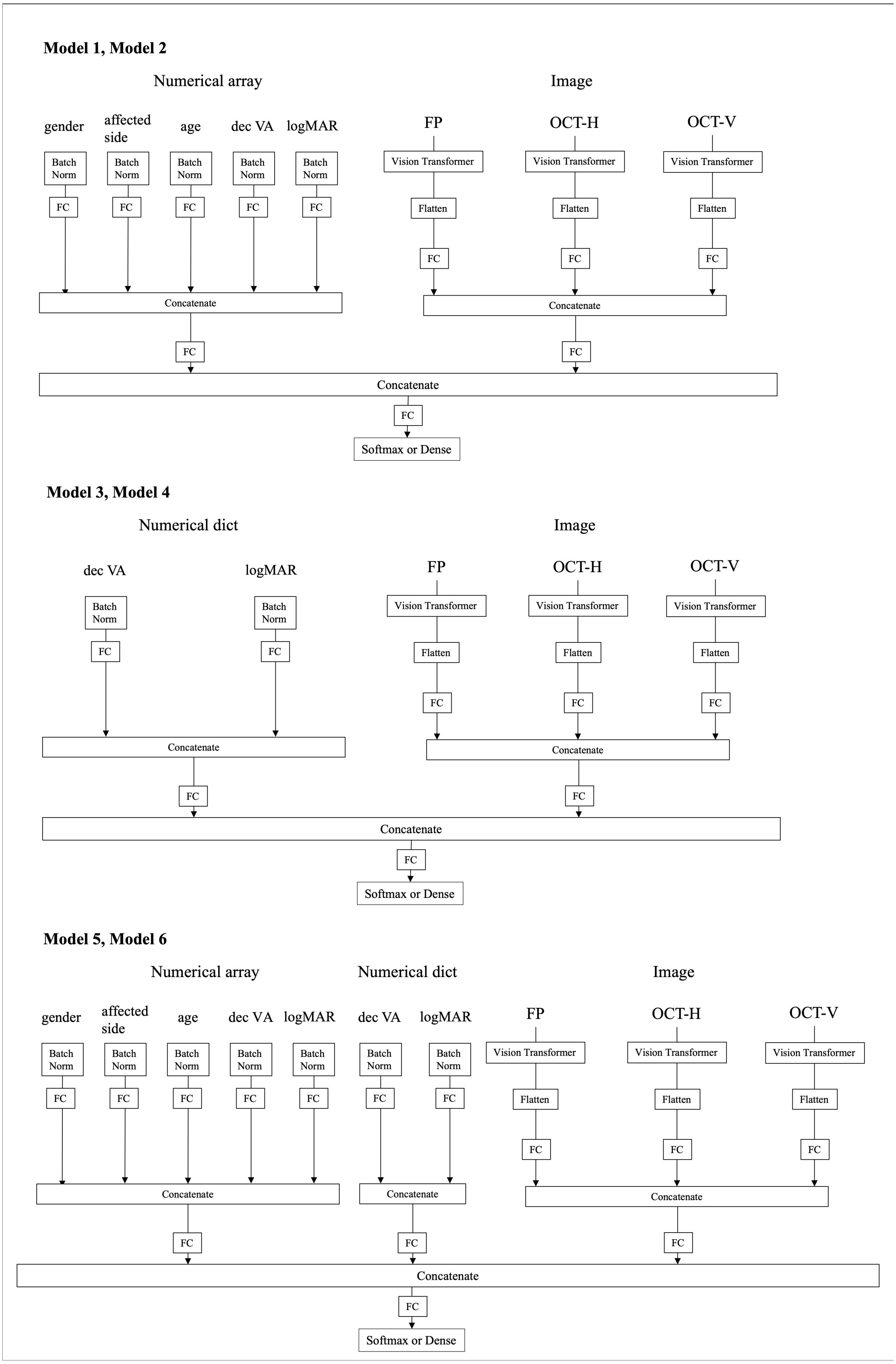


Fig. 5.4: Overview of generated models

5.4.4 Discussion

The first verification result shows that the proposed system design could generate a suitable model for each combination of input data. This indicates that the input data could be correctly patterned, and the proposed design could select the target. In this verification, the data for diagnosing eye diseases were modified to create pseudo data. However, even in the case of handling test data of other illnesses, sample test results are numerical data and image test results are image data. That is, the contents would not significantly differ. Recognition of the given input data can be altered by slightly changing the if-then rule of the proposed design. Due to this versatility, the system can be applied to diagnose other diseases.

Verification of the prediction accuracy of postoperative VA showed that the accuracy of the model generated by the proposed system was similar to that of the model proposed in this paper for both training data and verification data. This result revealed that the automatically generated model was trained according to the proposed design of this study. That is, the proposed system can automatically generate models that can be trained in input data preprocessing and feature extraction. It was shown that the proposed system could be used as an auxiliary tool for users without medical knowledge when developing a CDSS engine.

5.5 Summary

In actual clinical practice, numerical data, image data, text data, and video data are used. This chapter focused on numerical and image data used to diagnose eye diseases and proposed a system design that generates an appropriate deep learning model design. The proposed design showed the model objectives were selected for all data types that may be handled. Input data type was determined based on predefined if-then rules in the proposed design. After separating the input data, the appropriate pretreatment and feature extraction layers are selected for each input data. In addition, the output layer creates a softmax or Dense layer, depending on the user purpose of diagnosis or prognosis.

In verification of the proposed system design, it was confirmed that the intended model was generated by converting data for diagnosing eye diseases into data of array type numerical value, dictionary type numerical value, and image. In addition, accuracy in predicting postoperative VA by the generated model was verified. The results showed that the generated model that combines the designs set according to each data type. Verification showed that the proposed design could generate a model according to the input data and purpose to make a diagnosis and predict prognosis.

An automatic generation system of a deep learning model that can handle multimedia medical data has two advantages. The user simply inputs input data they want to use, selects the purpose, and generates the model. That is, no special knowledge is required to create the deep learning model. Using the proposed system, physicians can easily make prototypes of CDSS and possibly train them to diagnose efficiently. In addition, the model performs appropriate feature extraction and feature integration according to medical data.

These contributions have an important effect on model creation and evaluation in the CDSS development process. The proposed system allows physicians and systems engineers to prototype CDSS for a task they desire. Automatic generation of prototypes may reduce the duration of trial and error in modeling.

Chapter 6

Discussion

This study proposed a design method that performs appropriate pretreatment and feature extraction for automatic generation of an efficient MME for a CDSS. In this chapter, the proposed model in this study is considered from the viewpoint of its effectiveness in diagnosis/prognosis as a deep learning model, and the influence of the deep learning model on the medical field.

6.1 Use of the Proposed Design Method for Eye Diseases

This study aimed to design a deep learning model that can predict postoperative VA using the same input data and interpretation method for the diagnosis of AMD. The proposed design method can handle EMR and inspection image data used to diagnose AMD.

Chapter 3 proposed a design that eliminates the need for manual preprocessing of input data by automatically preprocessing according to specific criteria. The preprocessing mechanism is embedded within the deep learning model in the proposed design method. The model trains the preprocessing rules to perform preprocessing suited to the input data. Verification of the utility of the proposed design method was performed by assessing its ability to predict of posttreatment VA for AMD. The resultant prediction error when it performed preprocessing of input data was similar to that when manual preprocessing was performed, which implies the proposed design method was effective to automate preprocessing of required input data. Further, since the design of the model is fixed, the same processing style is applied for each type of data eliminating any potential designer bias.

Chapter 4 proposed a design method for feature integration that eliminates bias due to differences in the number of dimensions of the features extracted from the input data. The proposed design method combined features 1:1 by reducing or increasing the number of dimensions of the extracted features. Verification of the utility of the proposed design method was performed by assessing its ability to predict posttreatment VA for AMD. The resultant prediction error was smaller in the model that combined the number of dimensions of 1:1 by the than in the model that did not match the number of dimensions, which implies it is possible to connect to output without missing any feature by treating and interpreting multimedia medical data equally.

In Chapter 5, we designed a system that proposes an appropriate model design by giving input data and the purpose of the model. The proposed system automatically determines the input data and prepares an appropriate preprocessing and feature extraction layer. In addition, the proposed system prepares an appropriate output layer depending on whether the purpose is diagnosis or prognosis prediction. The data used to diagnose eye disease was entered and the design of the generated model was validated. Results of validation showed that a proper model design was proposed for input data. This result implies it has become possible to automate the generation of classifiers that require knowledge of deep learning.

The results described in each chapter suggested that the proposed design method was optimal for the task of postoperative VA in AMD. In addition, it has become possible to use the data that physicians and technicians want to use as a tool to assist diagnosis and analysis.

We focus on ophthalmic data treated in this study. EMR data is classified according to the criteria of statistical scale, and image data is classified according to the criteria of image type, with respect to the ophthalmic data treated in this study. Figure 6.1 shows the diagnostic process of AMD and data classification. Statistical scales include nominal, ordinal, interval, and proportional data [58]. The EMR data used in this study are classified into gender as nominal scale, decimal VA as interval scale, and age and logMAR as proportional scale. Types of images include digital images of body parts taken with a camera and reconstructed images in which other information is reconstructed as a two-dimensional image. FP and OCT image data used in this study are classified into digital and reconstructed images, respectively. The proposed design can be applied to medical data with the same schema. Section 6.2 also focuses on and discusses the schema of data handled by other clinical departments.

6.2 Effect of Proposal Design Method on Disease Diagnosis

This study focused on data used by physicians during diagnosis and aimed to replicate their manual processing of the same data by deep learning in the model. The data

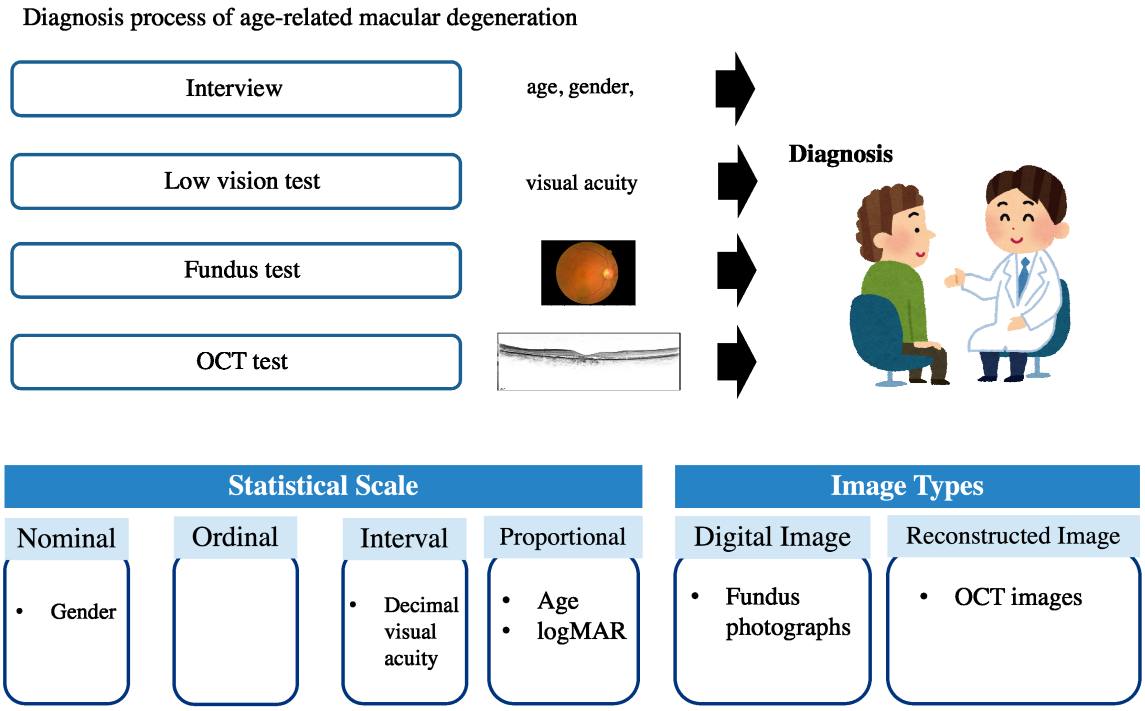
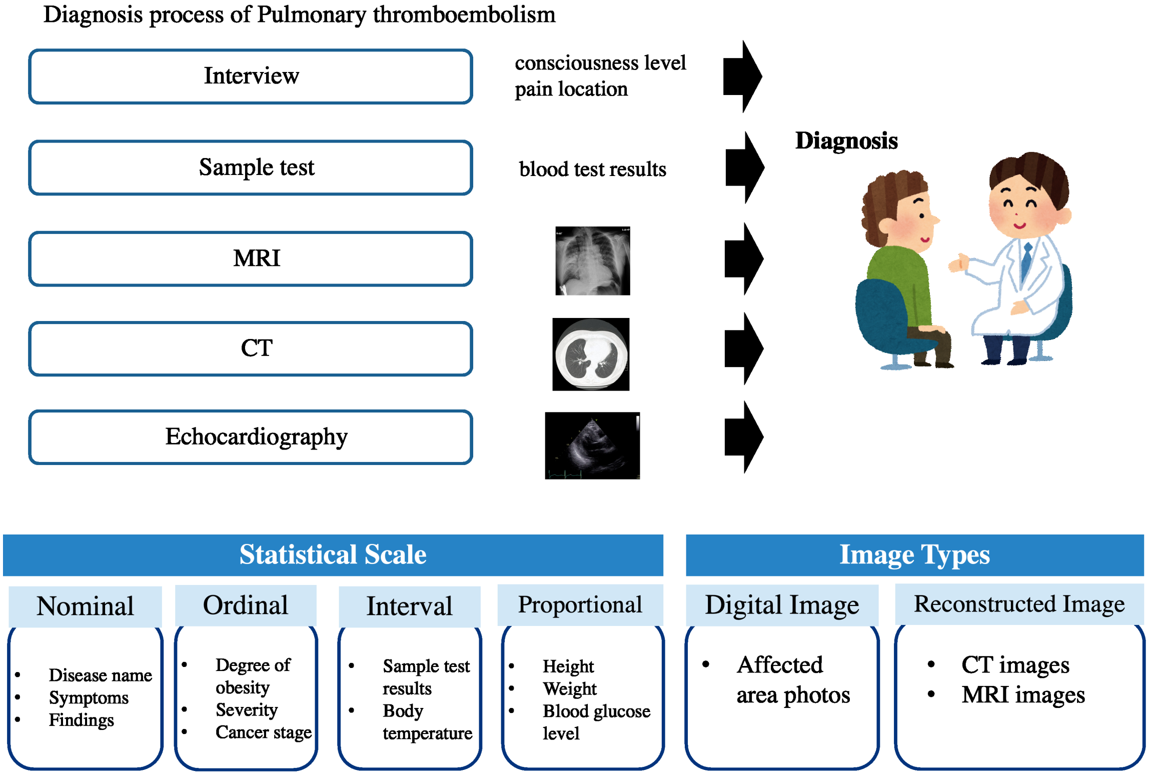


Fig. 6.1: Categorization of data for AMD diagnosis

to be handled included numerical data of sample test results, patient profile text data, and image data from imaging tests. In the proposed design method, preprocessing and feature extraction are performed for each input data, feature quantities are integrated, and diagnostic results and prognosis prediction results are output. The proposed design method was verified separately for the pretreatment/feature extraction part and the feature integration part. The pretreatment/feature extraction part achieved the same level of prediction accuracy as the existing model. The feature integration part achieved a better prediction error than the baseline model. The results implied that the design method could automate preprocessing of data as effectively as preprocessing performed with the special knowledge of the physician. In addition, it was shown that the features of each data could be integrated and judged and that learning was possible without manually giving the relationships among each data.

First, we focus on the data types handled by medical institutions. Medical data handled in clinical departments other than ophthalmology are classified according to the same criteria in Section 6.1. Figure 6.2 shows the pulmonary thromboembolism diagnostic process and data classification given as an example of the diagnostic process in other clinical departments. Nominal scales include data on disease names, medical conditions, and findings, and ordinal scales include data on disease severity and cancer stage. Interval scales



Statistical Scale				Image Types	
Nominal	Ordinal	Interval	Proportional	Digital Image	Reconstructed Image
• Disease name • Symptoms • Findings	• Degree of obesity • Severity • Cancer stage	• Sample test results • Body temperature	• Height • Weight • Blood glucose level	• Affected area photos	• CT images • MRI images

Fig. 6.2: Categorization of data in medical field

include sample test results and body temperature, and proportional scales include height, weight, and blood glucose levels. Digital images include photographs of the affected area. Reconstructed images include CT and MRI images. In this way, most data types used in medical practice can be classified by this standard. Even for data from different clinical departments, the proposed design method of this study is effective, and it is possible to generate MME with this automatic generation system if the data has the same schema.

Next, we focus on the model structure of deep learning. Many examples of ensemble learning models have been proposed in the medical field [40, 59]. Ensemble learning is a model that integrates output results obtained from a weak learner, which is a model that performs slightly better than random guessing, and outputs the final result. The weak learner handles each multimedia medical data separately, and outputs results. After that, the output results are summarized by a method such as majority voting and yielded as the final result.

In the actual medical field, there are radiologists and image interpreters for each type of examination image, and there are examination engineers who each specialize in genomic examinations in one diagnostic task. In many hospitals, physicians are finely specialized and subdivided according to specialty. Clinical departments are divided, and diagnosis and treatment are performed by each. On the other hand, in modern medical care in Japan, the number of medical institutions that have general medical care departments has increased. These are departments that understand human beings as a whole and provide multifaceted medical care without being limited to specific organs or diseases. In addition, the number of general medical care specialists is also increasing [38]. The general clinical department provides medical examinations and diagnoses to patients who present to the hospital with symptoms. In the case of emergency physicians, they also need to judge patients from various angles regardless of their specialty, and the number of emergency transports to hospitals is increasing year by year [18, 19]. Considering these social needs and trends, it can be said that there is a need for human resources that can derive a diagnosis after integrating and interpreting each data, rather than making individual judgments. The proposed design method of this study is a model that combines input data on a feature quantity basis. That is, the proposed design method drops all the data into one hyperspace, outputs diagnostic results, and predicts prognosis. In addition to being more accurate than the existing model, the system design that matches social trends is an advantage of the proposed design method in this research.

6.3 Effectiveness of Proposed Design Method on CDSS Development

This study focused on three issues in CDSS development including an inability to handle all the data used for diagnosis, different processing of the same data for each classifier, and high costs required to make a CDSS.

Embedding the pretreatment process proposed in this study into a deep learning model was as effective as manual pretreatment based on the knowledge of physicians. This result reveals that the time-consuming task of manual preprocessing can be replaced by the proposed design method, and the preprocessing can be performed according to a unified standard. In addition, it was shown that the proposed design method would allow systems engineers to perform appropriate pretreatment without medical knowledge or physician's instructions.

Next, the feature integration proposed in this study eliminates the binding bias from features with different dimensional numbers and integrates evenly. Multimedia medical data used for diagnosis in clinical settings may contain large differences in numbers and types of extracted features, and it is necessary to treat each item of data with equivalence for an unbiased diagnosis. The proposed design method allows for interpretation of features obtained from each data, similar to the best practice of a physician.

Finally, the automatic generation system proposed in this study generates MME simply by giving input data and purpose. The proposed system allows physicians to prototype CDSS for the task they seek. Physicians can proceed with prototype considers in advance, and the duration of trial and error in modeling is reduced.

6.4 Effectiveness of Deep Learning in the Medical Field

Deep learning models have made remarkable achievements in image recognition, object detection, and speech recognition, to mention a few. In the medical field, many studies using deep learning models have been conducted, such as detection of lesions in images, prediction of prognosis from test results, and diagnosis of specific diseases from multiple test images.

However, especially in the research of diagnostic imaging systems, the accuracy of lesion detection and diagnosis from medical images is lower than image recognition for discriminating general dogs and cats. This is because the difficulty level is different. In general image recognition and object detection, tasks such as detecting a person in an image of a person running on a hill and finding an airplane flying in a blue sky background are performed in which the features of the target object and characteristics of the background are profoundly different. In contrast, the positions of feature points such as color, shape, blood vessels, and bones, which are general feature quantities, are similar in medical images. The task is to find one subtly different lesion site from among them. Just as layperson without medical knowledge cannot identify lesions or abnormal sites in medical images, the task is complicated for deep learning models.

In the field of speech recognition, some researchers attempted to perform more accurate speech recognition by training not only speech information but also the image of the speaker's mouth [16]. Such a model facilitates judgment by using information supplementary to the original information. This concept describes a physician's diagnosis process. Physicians do not consider images, blood test results, or patient interviews in isolation.

This study adopted an approach that combines evidence of multimedia medical data for problems that are difficult to diagnose with a single medium. The proposed design method is consistent with the handling of multimedia medical data by physicians. As a result, the proposed design method can predict postoperative VA of AMD, which is difficult even for physicians, with a sufficient level of accuracy for clinical practice. This implies that information which would be considered redundant noise in sample test results and patient profiles were eliminated noise, making it easier for the model to extract appropriate features for tasks difficult to judge from image data alone. The training was also possible while having a large number of features. That is, a deep learning model that combined multiple modalities was effective, especially when considering the diagnostic process of physicians. Moreover, it will be a practical approach when providing diagnostic support using a deep learning model in the medical field.

6.5 Limitations

There are limitations to this research. All validations in this study were performed using multimedia medical data collected to diagnose AMD. The input patterns of numerical values and images required for the automatic generation system validation of the deep learning model were created in a pseudo manner by converting eye disease data. Still, they were not obtained from an existing EMR system. A possible problem with this limitation could be different data patterns entered in an actual medical setting. By passing data that is not expected as input data, there is a possibility that inappropriate input layers and feature extraction layers would be generated. There is a concern that model learning will not proceed well and that model generation will not be possible. The algorithm of the proposed system built an algorithm that refers to the value of the element in the input data and separates it. Still, it is assumed that it will not be effective if another data is added and the dimensions have changed.

In addition, this study excluded data with time-series information from multimodal medical data, since the focus of our study at this time was a first encounter between a physician making a diagnosis in a patient. Actually, physicians not only diagnose the patient's condition with a multimodal medical data in one visit, but with multiple visits to the hospital. Several deep learning models have been proposed for handling these time-series data [66, 68]. To embed those concepts into our MME design methods and its automatic generation is the next essential challenge for supporting the rapid prototyping of MME.

Chapter 7

Conclusion

The proposed system aims to enable physicians to develop CDSS prototypes without specialized knowledge of preprocessing and neural network model design. CDSS currently used in the medical field often warns of duplication of treatment from drug order information and detects contraindications for concomitant use of drugs. In recent years, research on CDSS that supports physicians' diagnosis using machine learning methods has increased. Thus, this research focused on CDSS that supports diagnosis.

For example, blood tests and biochemical tests in the medical field that measure the number of enzymes in blood and urine are collectively called sample tests. In addition, abdominal ultrasonography, physiological examinations, and imaging examinations are performed in addition to electrocardiograms, X-ray fluoroscopy, and CT examinations and image examinations include MRI examinations and endoscopy. The physician uses multimedia inspection data, including numerical and image data and palpation results, to comprehensively diagnose the patient's disorder.

This study named the deep learning model, which is a classifier included in CDSS, a multimodal medical encoder (MME). The design methodology to comprehensively develop the deep learning model with multimedia medical data input for diagnosing patient disorders was proposed. The proposed design method includes receiving multimedia medical data as input and extracting features from each data, combining the obtained features, and outputting a result according to the user's determined purpose, which may be either diagnosis or prognosis prediction. The proposed design method could be roughly divided into main parts including data input, preprocessing, feature extraction, feature integration, and output. Different input layers are prepared for each data type and operate separately in the data input part. The preprocessing and feature extraction parts are connected to layers that perform preprocessing of input data. The layers combine ViT for clinical images and fully connected layers for EMR. The proposed layers were prepared to concatenate multimodal clinical data features passed from the preprocessing and feature extraction parts in the feature integration part. Before the integration part, the layer to adjust the number of dimensions of the feature was prepared. Finally, the output part interprets the integrated clinical features, calculates the disease probability, and outputs the diagnostic or prognostic results. In addition, based on the proposed design method, this study developed an automation design system of an appropriate deep learning model regardless of the combination of multimedia clinical input and its output purpose.

Verification of the utility of the proposed design was performed for preprocessing and feature extraction, feature integration, and automatic generation. The dataset for verification included 315 data samples from patients who were diagnosed with wet AMD at Kyoto University Hospital macular clinic and completed a regimen of intravitreal injection of aflibercept for one year. In verifying the preprocessing and feature extraction part, predicting postoperative VA of AMD was performed. EMR, FP, and OCT images were used as input data. In experiments, prediction errors in the proposed model with preprocessing and integration layers were compared with those in the baseline model with manual preprocessing. The prediction error of the proposed model incorporating the proposed design was 0.054 and that of the baseline model was about 0.052 for verification data, which are similar revealing that the proposed design method can preprocess multimodal medical data automatically.

In verifying the feature integration part, prediction of postoperative VA of AMD was selected as a task. Three models were prepared to verify the effectiveness of the proposed design: a model that utilizes only image data as input, one that uses only EMR data, and a model that uses both. The prediction errors for the test data were 0.081, 0.052, and 0.047, respectively. This result suggests that more accurate support of physicians' diagnosis and prognosis can be achieved by integrating and interpreting multimedia medical data based on features.

In verifying the automatically generated system, this study confirmed whether the system could generate an appropriate model according to the given input data and purpose. In verifying the accuracy of the developed model, data used to diagnose AMD was converted into data expected to be received by the automatic generation system, and the model generated by actually inputting was confirmed. As a result, a structure with BN layer and an FC layer for array type numerical data, a design with BN layer and FC layer for dictionary type numerical data, and a structure with ViT for image data were generated. In addition, when the purpose is diagnosis, the softmax layer is generated as an output layer, and when the purpose is prognosis prediction, the Dense layer is generated as an output layer. The result showed that a model to which the design prepared for each data was applied was generated, and the model had an output layer according to selected purpose.

Verification results of the preprocessing and the feature extraction part showed the proposed method could generate a classification model that achieved performance similar to that of a physician manually preprocessing input data. The experimental results revealed that our proposed preprocessing and feature extraction design could handle multimedia medical data with a unified standard.

Verification results of the feature integration part revealed that combining multimedia medical data and our proposed integration layers yielded better performance than baseline results due to optimization with training.

Verification results of the automatic generation system suggested that an appropriate model could be automatically generated according to input data and purpose. Thus, it is expected that the automatically generated structure proposed in this study will enable users to create prototypes of diagnostic support systems using deep learning models and analyze diseases using data of their choice.

7.1 Contributions

Since physician diagnosis requires deep clinical knowledge in multiple clinical fields, it is a burden for physicians to diagnose patients in a short time period. It is essential to develop a better CDSS to support comprehensive diagnosis to reduce these burdens. For example, CDSS could inform physicians of possible diseases to prevent them from overlooking those diseases. To be able to handle data similar to the way a physician does brings several advantages. It can exhaustively search and refer to diagnosis histories in the hospital information system (HIS) . In addition, by comprehensively interpreting the features of the data as physicians do, CDSS could find an abnormal condition by transverse analysis. This proposed design method and automatic generation system could build an appropriate CDSS engine model using a combination of sample test data in a medical institution. This proposal has two significant contributions to developing CDSS in clinical settings; the automation of preprocessing of input data and automatic generation of a deep learning model.

As for preprocessing automation, from the viewpoint of physicians performing diagnoses, most do not have technical knowledge of deep learning. CDSS can inform physicians of system estimated diagnostic results by a deep learning model. In addition, even though a target disease may not be within the clinical field for which a physician was trained, the model can assist the physician since it can be trained with cross-medical-field datasets in HIS. In current practice for developing deep learning models, physicians play a major role in categorizing patient profile data and trimming and resizing clinical images. The proposed design method releases physicians from these burdens by automatically deciding appropriate preprocessing layers inside the model structure.

The automatic generation system contributes to shortening the development period of the entire CDSS. If the automatic generation system enables physicians to develop prototypes of CDSS without specialized knowledge of preprocessing and neural network model design, physicians will be able to prototype and consider models themselves. In the development process, physicians decide the target disease, collect data, preprocess it, and then ask the system vendor to develop a CDSS which could take years since they may have to make and test many prototypes in consultation between physicians and engineers. Long-term development is also susceptible to increased human costs and model modifications due to changes in diagnostic criteria. Physicians can prototype a model without special knowledge about deep learning by collecting data with the automatic generation system. The time required for creating the classifier can be reduced by examining the CDSS in advance with our proposed model auto-generation system.

Through those improvements of CDSS development, it is expected that patients receiving medical care will have more opportunities to receive safer medical care from a physician using CDSS. The reduction of the development cost of CDSS by the automatic generation system will lead to the development of CDSS corresponding to various diseases. Diagnosis by physicians and CDSS for many diseases will give patients more peace of mind.

Bibliography

[1] *Guidelines for clinical examination (reference range / clinical judgment value) (Japanese)*. Japanese Society of Laboratory Medicine, 2018.

[2] Emily Alsentzer and Anne Kim. Extractive summarization of ehr discharge notes. *arXiv preprint arXiv:1810.12085*, 2018.

[3] Gopi Battineni, Getu Gamo Sagaro, Nalini Chinatalapudi, and Francesco Amenta. Applications of machine learning predictive models in the chronic disease diagnosis. *Journal of personalized medicine*, 10(2):21, 2020.

[4] Ananth Reddy Bhimireddy, Priyanshu Sinha, Bolu Oluwalade, Judy Wawira Gichoya, and Saptarshi Purkayastha. Blood glucose level prediction as time-series modeling using sequence-to-sequence neural networks. In *KDH@ ECAI*, pages 125–130, 2020.

[5] Ekaba Bisong. Google automl: cloud vision. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*, pages 581–598. Springer, 2019.

[6] David G Bounds, Paul J Lloyd, Bruce G Mathew, and Gordon Waddell. A multilayer perceptron network for the diagnosis of low back pain. In *ICNN*, volume 2, pages S481–489, 1988.

[7] Rupert RA Bourne, Jost B Jonas, Seth R Flaxman, Jill Keeffe, Janet Leasher, Kovin Naidoo, Maurizio B Parodi, Konrad Pesudovs, Holly Price, Richard A White, et al. Prevalence and causes of vision loss in high-income countries and in eastern and central europe: 1990–2010. *British Journal of Ophthalmology*, 98(5):629–638, 2014.

[8] Philippe Burlina, David E Freund, Neil Joshi, Y Wolfson, and Neil M Bressler. Detection of age-related macular degeneration via deep learning. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 184–188. IEEE, 2016.

[9] Billy Chiu, Gamal Crichton, Anna Korhonen, and Sampo Pyysalo. How to train good word embeddings for biomedical nlp. In *Proceedings of the 15th workshop on biomedical natural language processing*, pages 166–174, 2016.

[10] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[12] Cristóbal Esteban, Inmaculada Arostegui, Susana Garcia-Gutierrez, Nerea Gonzalez, Iratxe Lafuente, Marisa Bare, Nerea Fernandez de Larrea, Francisco Rivas, and José M Quintana. A decision tree to assess short-term mortality after an emergency department visit for an exacerbation of copd: a cohort study. *Respiratory research*, 16(1):1–10, 2015.

[13] Masaru Suzuki et al. *Internal Medicine Emergency Medical Care Guidelines 2016 (Japanese)*. Japanese Society of Internal Medicine, 2016.

[14] Dominik A Ettlin. The international classification of headache disorders, (beta version). *Cephalalgia*, 33(9):629–808, 2013.

[15] R. Ezhilarasi and P. Varalakshmi. Tumor detection in the brain using faster r-cnn. *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on*, pages 388–392, 2018.

[16] Weijiang Feng, Naiyang Guan, Yuan Li, Xiang Zhang, and Zhigang Luo. Audio visual speech recognition with multimodal recurrent neural networks. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 681–688. IEEE, 2017.

[17] Frederick L Ferris III, Aaron Kassoff, George H Bresnick, and Ian Bailey. New visual acuity charts for clinical research. *American journal of ophthalmology*, 94(1):91–96, 1982.

[18] Fire, Ministry of Internal Affairs Disaster Management Agency, and Communications. Current status of emergency / rescue. 2020.

[19] Tsuguya Fukui. 1. general medical department (japanese). *Journal of the Japanese Society of Internal Medicine*, 91(11):3106–3110, 2002.

[20] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[21] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, 316(22):2402–2410, 2016.

[22] Cheng Guo and Felix Berkhahn. Entity embeddings of categorical variables. *arXiv preprint arXiv:1604.06737*, 2016.

[23] Donglin Guo, Min Li, Ying Yu, Yaohang Li, Guihua Duan, Fang-Xiang Wu, and Jianxin Wang. Disease inference with symptom extraction and bidirectional recurrent neural network. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 864–868. IEEE, 2018.

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

[25] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[26] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[27] Peter K Kaiser. Prospective evaluation of visual acuity assessment: a comparison of snellen versus etdrs charts in clinical practice (an aos thesis). *Transactions of the American Ophthalmological Society*, 107:311, 2009.

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[29] C Lange, N Feltgen, B Junker, K Schulze-Bonsel, and M Bach. Resolving the clinical acuity categories “hand motion” and “counting fingers” using the freiburg visual

acuity test (fract). *Graefe's Archive for Clinical and Experimental Ophthalmology*, 247(1):137–142, 2009.

[30] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[31] Cecilia S Lee, Doug M Baughman, and Aaron Y Lee. Deep learning is effective for classifying normal versus age-related macular degeneration oct images. *Ophthalmology Retina*, 1(4):322–327, 2017.

[32] M Levi, W Hart, and HR Büller. Physical examination–the significance of homan's sign. *Nederlands tijdschrift voor geneeskunde*, 143(37):1861–1863, 1999.

[33] Laurence S Lim, Paul Mitchell, Johanna M Seddon, Frank G Holz, and Tien Y Wong. Age-related macular degeneration. *The Lancet*, 379(9827):1728–1738, 2012.

[34] Weiming Lin, Tong Tong, Qinquan Gao, Di Guo, Xiaofeng Du, Y Yang, G Guo, M Xiao, M Du, and X Qu. The alzheimer's disease neuroimaging initiative. convolutional neural networks-based mri image analysis for the alzheimer's disease prediction from mild cognitive impairment. *Front Neurosci*, 12:777, 2018.

[35] Marta Luri, Leire Leache, Gabriel Gastaminza, Antonio Idoate, and Ana Ortega. A systematic review of drug allergy alert system. *International journal of medical informatics*, page 104673, 2021.

[36] 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.

[37] Hamid Mcheick, Lokman Saleh, Hicham Ajami, and Hafedh Mili. Context relevant prediction model for copd domain using bayesian belief network. *Sensors*, 17(7):1486, 2017.

[38] Ian R McWhinney. Being a general practitioner: what it means. *The European Journal of General Practice*, 6(4):135–139, 2000.

[39] Labour Ministry of Health and Welfare. Overview of 2020 vital statistics monthly report (japanese). *https://www.mhlw.go.jp/toukei/saikin/hw/jinkou/geppo/nengai20/dl/gaikyouR2.pdf*, 2020.

[40] Woo Kyung Moon, Yan-Wei Lee, Hao-Hsiang Ke, Su Hyun Lee, Chiun-Sheng Huang, and Ruey-Feng Chang. Computer-aided diagnosis of breast ultrasound images using ensemble learning from convolutional neural networks. *Computer methods and programs in biomedicine*, 190:105361, 2020.

[41] Kentaro Nakagawa, Ryu Ishihara, Kazuharu Aoyama, Masayasu Ohmori, Hiroko Nakahira, Noriko Matsuura, Satoki Shichijo, Tsutomu Nishida, Takuya Yamada, Shinjiro Yamaguchi, et al. Classification for invasion depth of esophageal squamous cell carcinoma using a deep neural network compared with experienced endoscopists. *Gastrointestinal endoscopy*, 90(3):407–414, 2019.

[42] Mashio Nakamura, Tetsuro Miyata, Yasushi Ozeki, Morimasa Takayama, Kimihiro Komori, Norikazu Yamada, Hideki Origasa, Hirono Satokawa, Hideaki Maeda, Nobuhiro Tanabe, et al. Current venous thromboembolism management and outcomes in japan. *Circulation Journal*, 78(3):708–717, 2014.

[43] Kitsuchart Pasupa and Wisuwat Sunhem. A comparison between shallow and deep architecture classifiers on small dataset. In *2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pages 1–6. IEEE, 2016.

[44] Barak A Pearlmutter. Learning state space trajectories in recurrent neural networks. *Neural Computation*, 1(2):263–269, 1989.

[45] JH Warwick Pexman, Philip A Barber, Michael D Hill, Robert J Sevick, Andrew M Demchuk, Mark E Hudon, William Y Hu, and Alastair M Buchan. Use of the alberta stroke program early ct score (aspects) for assessing ct scans in patients with acute stroke. *American Journal of Neuroradiology*, 22(8):1534–1542, 2001.

[46] Fulvio Pomero, Francesco Dentali, Valentina Borretta, Matteo Bonzini, Remo Melchio, James D Douketis, and Luigi Maria Fenoglio. Accuracy of emergency physician–performed ultrasonography in the diagnosis of deep-vein thrombosis. *Thrombosis and haemostasis*, 109(01):137–145, 2013.

[47] Robert A Raschke, Bea Gollihare, Thomas A Wunderlich, James R Guidry, Alan I Leibowitz, John C Peirce, Lee Lemelson, Mark A Heisler, and Cynthia Susong. A computer alert system to prevent injury from adverse drug events: development and evaluation in a community teaching hospital. *Jama*, 280(15):1317–1320, 1998.

[48] Mark Ratner. Next-generation amd drugs to wed blockbusters. *Nature biotechnology*, 32(8):701–703, 2014.

[49] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[50] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015.

[51] Markus Rohm, Volker Tresp, Michael Müller, Christoph Kern, Ilja Manakov, Maximilian Weiss, Dawn A Sim, Siegfried Priglinger, Pearse A Keane, and Karsten Kortuem. Predicting visual acuity by using machine learning in patients treated for neovascular age-related macular degeneration. *Ophthalmology*, 125(7):1028–1036, 2018.

[52] Daniel A Rosser, Simon N Cousens, Ian E Murdoch, Fred W Fitzke, and David AH Laidlaw. How sensitive to clinical change are etdrs logmar visual acuity measurements? *Investigative ophthalmology & visual science*, 44(8):3278–3281, 2003.

[53] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[54] Sou Sakamoto. *Emergency outpatient-under diagnosis now!- (Japanese).* Chugai-igakusya, 2015.

[55] Kabid Hassan Shibly, Samrat Kumar Dey, Md Tahzib-Ul Islam, and Md Mahbubur Rahman. Covid faster r–cnn: A novel framework to diagnose novel coronavirus disease (covid-19) in x-ray images. *Informatics in Medicine Unlocked*, 20:100405, 2020.

[56] Nakamura Shun, Ueno Shinji, Suzuki Yoshiro, Jill Keeffe, Janet Leasher, Kovin Naidoo, Maurizio B Parodi, Konrad Pesudovs, Holly Price, Richard A White, et al. Extrapolating visual acuity after treatment from pre-treatment optical coherence tomography images in brvo using cnn. *Medical Imaging and Information Sciences*, 36(3):136–140, 2019.

[57] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[58] Catriona M Steele and Karen Grace-Martin. Reflections on clinical and statistical use of the penetration-aspiration scale. *Dysphagia*, 32(5):601–616, 2017.

[59] Heung-Il Suk, Seong-Whan Lee, Dinggang Shen, Alzheimer's Disease Neuroimaging Initiative, et al. Deep ensemble learning of sparse regression models for brain disease diagnosis. *Medical image analysis*, 37:101–113, 2017.

[60] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[61] Reed T. Sutton, David Pincock, Daniel C. Baumgart, Daniel C. Sadowski, Richard N. Fedorak, and Karen I. Kroeker. An overview of clinical decision support systems: benefits, risks, and strategies for success. *npj Digital Medicine*, 3(1):17, 2020.

[62] Weijun Tan and Jingfeng Liu. A 3d cnn network with bert for automatic covid-19 diagnosis from ct-scan images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 439–445, 2021.

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

[64] David R Vinson. Epiploic appendagitis: a new diagnosis for the emergency physician. two case reports and a review. *The Journal of emergency medicine*, 17(5):827–832, 1999.

[65] Max Harry Weil and Herbert Shubin. Proposed reclassification of shock states with special reference to distributive defects. In *The fundamental mechanisms of shock*, pages 13–23. Springer, 1972.

[66] Kun Xia, Jianguang Huang, and Hanyu Wang. Lstm-cnn architecture for human activity recognition. *IEEE Access*, 8:56855–56866, 2020.

[67] Tao Xu, Han Zhang, Xiaolei Huang, Shaoting Zhang, and Dimitris N. Metaxas. Multimodal deep learning for cervical dysplasia diagnosis. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 115–123, 2016.

[68] Yiwen Xu, Ahmed Hosny, Roman Zeleznik, Chintan Parmar, Thibaud Coroller, Idalid Franco, Raymond H Mak, and Hugo JWL Aerts. Deep learning predicts lung cancer treatment response from serial medical imaging. *Clinical Cancer Research*, 25(11):3266–3275, 2019.

[69] Hongmei Yan, Yingtao Jiang, Jun Zheng, Chenglin Peng, and Qinghui Li. A multilayer perceptron-based medical decision support system for heart disease diagnosis. *Expert Systems with Applications*, 30(2):272–281, 2006.

List of Publications

Refereed Journals

R. Otsuki, O. Sugiyama, Y. Mori, M. Miyake, S. Hiragi, G. Yamamoto, L. Santos, Y. Nakanishi, Y. Hosoda, H. Tamura, S. Matsumoto, A, Tsujikawa, T. Kuroda, *Deep Learning Model to Predict Postoperative Visual Acuity from Preoperative Multimedia Ophthalmic Data*, Advanced Biomedical Engineering, Vol. 9, pp. 241-248, 2020

R. Otsuki, O. Sugiyama, Y. Mori, M. Miyake, S. Hiragi, G. Yamamoto, L. Santos, Y. Nakanishi, Y. Hosoda, H. Tamura, S. Matsumoto, A, Tsujikawa, T. Kuroda, *Integrating Preprocessing Operations into Deep Learning Model: Case Study of Posttreatment Visual Acuity Prediction*, Advanced Biomedical Engineering, Vol. 11, pp. 16-24, 2022