

2017학년도 창의융합 과제연구 보고서

퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

2학년 14반 양현서
2학년 12반 이현빈
2학년 12반 이승현

Contents

1	요약서	1
1.1	탐구(연구) 동기	1
1.2	탐구(연구) 내용	1
2	서론	2
2.1	탐구의 목적 및 동기	2
2.2	이론적 배경	2
3	본론	3
3.1	visual basic언어의 역할과 원리	3
3.2	학습데이터	4
3.3	어플리케이션 제작	4
3.4	c++언어의 역할과 구동의 흐름	5
3.5	Trainer 흐름	5
3.6	안드로이드 애플리케이션 실행흐름	5
3.7	전체적인 구동의 흐름	5
3.8	실제 구동 사진 - Medinology Andriod 버전	10
3.9	실제 구동 사진 - Medinology PC 버전	10
4	결론	10
5	참고문헌	11

1 요약서

총품부문 정보

작품명 퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

1.1 탐구(연구) 동기

인공지능이 각광 받고 있는 이 시대에 인공지능을 공부하다 우리가 공부한 것들을 바탕으로 누군가에게 도움이 되는 무언가를 만들 수 있지 않을까 하는 고민에서 이 연구를 시작하게 되었다. 늦은 밤 혹은 약국이 없는 외진 지역에서 어떤 약을 구매해야 할지 몰라 약을 구매하지 못하거나 잘못된 정보로 약을 구매하는 것을 보았다. 잘못된 약 복용으로 부작용이 생기거나 약을 구매하지 못해서 생기는 사고를 방지하기 위해 시스템을 개발하고 싶다는 생각을 하게 되었다. 중상별로 약을 추천해줄 수 있는 방법을 고민하다가 퍼셉트론을 이용한 인공신경망을 떠올리게 되었다.

1.2 탐구(연구) 내용

선행 연구 고찰 및 탐구의 독창성

dbpia, krpia, 구글 검색등을 통해 약처방 자동화 시스템과 관련된 여러 선행여구를 찾으려 했으나 찾을 수 없었다. 비슷한 것이라고 해봐야 처방전을 넣으면 약을 주거나 약을 잘라서 봉투에 담아주는 기계 정도였다. 그래서 우리의 탐구가 더 독창적이고 필요한 것이라 생각된다.

2017 창의융합과제연구
Kyunggi High School
퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

탐구 절차 및 방법

퍼셉트론 인공신경망을 제작하는 데에 있어 원래는 퍼셉트론이라는 단위 클래스를 기반으로 전체 신경망을 구성하려 하였으나 참고 문헌을 보고 나서 이 방법이 너무 비효율적임을 깨닫고 행렬을 이용한 방법으로 방향을 바꾸었다. 기본적인 베이스 프로그램은 Visual Basic을 이용해 코딩하였고, 사용자가 데이터를 입력하면 입력된 데이터를 C++로 코딩된 퍼셉트론 클래스로 전달 후 퍼셉트론 클래스에서 출력된 데이터를 변환해 사용자에게 총 3순위의 질병 및 확률, 처방될 약과 주의할 점을 나타내는 순서로 시스템을 구성하였다.

작품의 주요 내용

우리 탐구의 가장 핵심내용은 C++을 이용한 퍼셉트론이라고 할 수 있다. `medinology_cpp.exe` 즉 인공신경망은 다음과 같은 부분으로 나누어진다.

- VB에서 전달된 환자의 건강 정보 자료를 파싱하는 부분
- weight.txt 파일의 내용을 신경망의 가중치, 편향에 로드하는 부분
- 신경망을 생성하여 초기화하고 Predict함수를 호출하여 결과를 계산하는 부분
- 계산 결과를 분석하여 질병 명을 확률 순서대로 정렬하여 파일에 출력하는 부분

탐구(연구) 결과

실제로 앱을 완성하고 구동시켜본 결과 증상에 맞는 질병과 약이 나오지만 확률이 낮은 부분을 확인 할 수 있었다. 이는 짧은 기간 수작업으로 데이터를 생성한 만큼 절대적인 데이터량의 부족으로 인한 것으로 생각된다.

2 서론

2.1 탐구의 목적 및 동기

인공지능이 각광 받고 있는 이 시대에 인공지능을 공부하다 우리가 공부한 것들을 바탕으로 누군가에게 도움이 되는 무언가를 만들 수 있지 않을까 하는 고민에서 이 연구를 시작하게 되었다. 늦은 밤 혹은 약국이 없는 외진 지역에서 약을 구매해야하는데 어떤 약을 구매해야 할지 몰라 약을 구매하지 못하거나 잘못된 정보로 약을 구매하는 것을 보았다. 잘못된 약 복용으로 부작용이 생기거나 약을 구매하지 못해서 생기는 사고를 방지하기 위해 시스템을 개발하고 싶다는 생각을 하게 되었다.

증상별로 약을 추천해줄 수 있는 방법을 고민하다가 퍼셉트론을 이용한 인공신경망을 떠올리게 되었다. 퍼셉트론 인공신경망은 학습을 통해 판단이 가능하므로 여러 임상사례들을 학습시켜 약을 처방해줄 수 있을 것이라 판단하였다. 휴리스틱(사람스러운) 기능을 가지고 있는 약 처방 프로그램을 개발하여 어떤 약을 선택해야 할지 모르겠는 상황 또는 병원을 좋아하지 않으시는 어르신들이 병원을 가도록 경고를 해주거나 하는 인공지능 시스템을 개발하여 사람들을 편리하게 하고 건강을 유지하는데 도움을 줌을 목적으로 한다.

2.2 이론적 배경

퍼셉트론 인공신경망

퍼셉트론이란 입력단자가 여러 개이고 출력 단자가 한 개인 객체이다. 퍼셉트론은 각 입력마다 가중치가 있어서 여러 개의 입력 값을 주면 각 단자의 입력값과 가중치를 곱해 모두 더한 값을 구해 가중 합을 정하고 고정된 임계치보다 가중 합이 작으면 0, 그렇지 않으면 1과 같은 방식으로 출력값을 내보낸다. 이러한 퍼셉트론은 단순한 판정만이 가능할 뿐인데 다수의 퍼셉트론을 여러 계층으로 배열하여 인공신경망을 만들 수 있다. 이렇게 만든 인공신경망을 학습시키기 위해서는 먼저 학습을 위한 입력값들 즉 학습데이터를 만들어야 한다. 학습 단계에서는 학습데이터와 정답값을 함께 인공 신경망에 제공하고 출력값을 구한 후 정답에 해당하는 값과 차이가 줄어들도록 가중치를 갱신한다. 어떤 학습데이터가 주어지면 이 때의 출력값을 구하고 학습데이터와 함께 제공된 정답에 해당하는 값에서 출력값을 뺀 값 즉 오차 값을 구한다. 이 오차 값을 일부가 출력단자에서 입력 단자로 되돌아가면서 각 계층의 퍼셉트론 별로 출력 신호를 만드는데 관여한 모든 가중치들에 더해지는 방식으로 가중치들이 갱신된다. 이러한 과정을 다양한 학습 데이터에 대하여 반복하면 출력값들이 각각의 정답값에 수렴하게 되어 판정 성능이 좋아지게 되고 본연구에서는 이러한 퍼셉트론을 여러 계층으로 배열한 인공신경망을 제작하고 각 증상, 병원체별 학습데이터를 제작, 학습을 시켜 각 상황에 알맞은 약을 처방할 수 있는 시스템을 만든다.

딥러닝

컴퓨터가 여러 데이터를 이용해 마치 사람처럼 스스로 학습할 수 있게 하기 위해 인공 신경망(ANN: artificial neural network)을 기반으로 구축한 한 기계 학습 기술이다. 딥 러닝은 인간의 두뇌가 수많은 데이터 속에서 패턴을 발견한 뒤 사물을 구분하는 정보처리 방식을 모방해 컴퓨터가 사물을 분별하도록 기계를 학습시킨다. 딥 러닝 기술을 적용하면 사람이 모든 판단 기준을 정해주지 않아도 컴퓨터가 스스로 인지·추론·판단할 수 있게 된다. 음성·이미지 인식과 사진 분석 등에 광범위하게 활용된다. 본 연구에서는 퍼셉트론 인공신경망에 여러 가지 병에 관한 학습데이터를 제공해 딥러닝을 하고 약을 처방한다.

시그모이드 함수

시그모이드 함수(Sigmoid function)는 S와 같은 형태(Sigmoid curve)를 가진 함수를 말한다. 주로 학습 곡선등을 나타내는 함수로, 0에 가까운 작은 값에서 일정한 유한값에 접근하는 함수이다. 1과 0사이를 부드럽게 이어준다. 특징은 x가 어떤값이어도 바로 1혹은 0으로 값을 얻어낼수 있다는 것이다. 주로 어떤 현상을 단순화하여 1과 0으로 놓고, 그 사이값으로 확률을 추론하는데에 많이 사용된다. 함수식은 다음과같다.

3 본론

3.1 visual basic언어의 역할과 원리

Visual Basic으로 코딩된 Medinology는 기본적인 베이스 프로그램으로 사용자가 질병과 약 처방을 결정 할 수 있는 정보들을 이 프로그램에 입력하면 입력된 데이터를 C++로 코딩된 퍼셉트론 클래스로 전달 후 퍼셉트론 클래스에서 출력된 데이터를 변환해 사용자에게 총 3순위의 질병 및 확률, 처방될 약과 주의할 점을 나타낸다. Medinology.exe를 실행하면 사용자가 성별, 나이, 임신 여부, 증상, 과거 병력, 현재 투약중인 약과 같은 정보를 입력할 수 있는 폼이 나타난다. 입력이 완료되면 System.IO.StreamWriter 클래스를 사용해 Input.WriteLine으로 사용자에게 입력받은 정보를 Input.txt에 저장한다. 이 후 Shell\(\Medinology.cpp.\exe)로 퍼셉트론 클래스인 Medinology.cpp.exe를 실행시켜 출력을 Output.txt로 받게 된다. 세 가지의 병 코드와 확률이 입력되어 있는 Output.txt를 System.IO.StreamReader 클래스를 사용해 Output.txt 내용을 읽어들인다. 읽어 들인 내용을 결과 폼에서 If문을 사용하여 변환된 질병 및 확률과 그에 따라 처방될 약이 출력되고 만약 임산부, 어린이, 어르신에게 위험한 약이라면 의사 및 약사에게 상담 받아야 한다는 권고 또한 출력된다. 실제로는 매우 방대한 양이지만 가장 핵심부분의 일부 소스코드는 다음과 같다.

```
1 //Form_Input.에서vb Input.를txt 쓰는과정
2 Dim Input As New System.IO.StreamWriter("Input.txt", False)
3 Input.WriteLine(Gender)
4 Input.WriteLine(Age)
5 Input.WriteLine(Preg)
6 Input.WriteLine(Symp01 & Symp02 & Symp03 & Symp04 & Symp05 & Symp06 & Symp07 &
    Symp08 & Symp09 & Symp10 & Symp11 & Symp12 &
    Symp13 & Symp14 & Symp15 & Symp16 & Symp17 & Symp18 & Symp19 & 중
    략
8
9     Symp49 & Symp50 & Symp51)
10 Input.WriteLine(His01 & His02 & His03 & His04 & His05 & His06 & His07 & His08
    & His09 & His10 & His11)
11 Input.WriteLine(Medi01 & Medi02 & Medi03 & Medi04 & Medi05 & Medi06 & Medi07 &
    Medi08 & Medi09 & Medi10 & Medi11)
12 Input.Close()
13
14 Me.Hide()
15 Form_Process.Show()
16
17 //Form_Process.에서vb Medinology_cpp.를exe 실행하는과정
18 Private Sub Form_Process_Load(sender As Object, e As EventArgs) Handles MyBase.Load
19     Shell("Medinology_cpp.exe")
20 End Sub
```

2017 창의융합과제연구
Kyunggi High School
퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

```
21 //Form_Output.에서vb Output.를txt 읽는과정
22 Dim Output As New System.IO.StreamReader("Output.txt")
23 Dim Per1 As Integer = Output.ReadLine()
24 Dim Di1 As Integer = Output.ReadLine()
25 Dim Per2 As Integer = Output.ReadLine()
26 Dim Di2 As Integer = Output.ReadLine()
27 Dim Per3 As Integer = Output.ReadLine()
28 Dim Di3 As Integer = Output.ReadLine()
29
30
31 //Form_Output.에서vb 순위1 질병과처방약과주의해야할점들을출력하는과정
32 If Di1 = 0 And Preg = "F" And Age >= 13 Then
33     Label_1_Di.Text = "비브리오" 불니피쿠스감염증 "
34     Label_1_Medi.Text = "크라포란주", 독시사이클린"
35     Label_1_Warn1.Text = "병원" 처방이필요합니다 ."
36 ElseIf Di1 = 0 And Age < 13 Then
37     Label_1_Di.Text = "비브리오" 불니피쿠스감염증 "
38     Label_1_Medi.Text = "크라포란주", 독시사이클린"
39     Label_1_Warn1.Text = "크라포란주" = 병원처방이필요합니다 ."
40     Label_1_Warn2.Text = "독시사이클린" = 병원처방이필요합니다 . 어린이에게위험하므로의사및약사
        와상담이필요합니다 ."
41 ElseIf Di1 = 0 And Preg = "T" Then
42     Label_1_Di.Text = "비브리오" 불니피쿠스감염증 "
43     Label_1_Medi.Text = "크라포란주", 독시사이클린"
44     Label_1_Warn1.Text = "병원" 처방이필요합니다 . 임산부에게위험하므로의사및약사와상담이필요합니
        다 ."
45 ElseIf Di1 = 1 And Preg = "F" And Age >= 13 Then
46     Label_1_Di.Text = "렙토스피라병"
47     Label_1_Medi.Text = "독시사이클린", 앰씰린캡슐"
48     Label_1_Warn1.Text = "병원" 처방이필요합니다 ."
49 ElseIf Di1 = 1 And Age < 13 Then
50     Label_1_Di.Text = "렙토스피라병"
51     Label_1_Medi.Text = "독시사이클린", 앰씰린캡슐"
52     Label_1_Warn1.Text = "독시사이클린" = 병원처방이필요합니다 . 어린이에게위험하므로의사및약사
        와상담이필요합니다 ."
53     Label_1_Warn2.Text = "앰씰린캡슐" = 병원처방이필요합니다 .
```

3.2 학습데이터

퍼셉트론 인공신경망을 학습시켜주기 위해서는 학습데이터가 필요하다. 학습데이터는 증상-질병, 질병-약, 약-특징의 3가지 행렬데이터를 엑셀을 통해 제작하였다. 1은 해당항목에 해당됨을 0은 해당되지않음을 의미한다. 학습데이터 중 증상-질병의 원핫 인코딩 데이터의 일부는 다음과 같다.

3.3 어플리케이션 제작

이 프로젝트는 편리성을 중점으로 두고 개발하는 것이므로 모바일 버전도 있으면 좋을것 같다고 판단했다. 그래서 제작해놓은 퍼셉트론 클래스로 앱을 제작하기로 결정하였다. 안드로이드 개발환경은 PC의 Android Studio와 모바일의 AIDE가 있다. 어플리케이션에 대해 간단히 설명하자면 1.첫 번째 화면에서 환자의 기초 정보를 수집한다. 2.두 번째 화면에서는 환자의 증상을 고르게 한다. 3.세 번째 화면에서는 환자의 질병명과 그 확률, 그리고 그에 맞는 약제들의 이름을 순서대로 출력하고, 노인같은 저시력자들을 위해 음성으로 결과를 읽어준다. 이때 면책 조항도 나오는데 이는 이 프로그램이 아직 초기 버전이므로 예측이 틀릴 수 있고, 이것이 중대한 결과를 초래할 수 있기 때문에 주의하고 병원에 가라는 내용이 담겨 있다. 기술을 상세히 설명해보겠다. 이 애플리케이션은 C++로 구현되는 PC버전을 포팅하는 것이므로 안드로이드 NDK와 JAVA의 JNI(Java Native Interface)를 활용하였다. 안드로이드의 프로그램 시작점은 MainActivity 클래스의 OnCreate()메소드이다. 이 메소드에서는 주로 앱의 화면과 데이터를 초기화한다 그리고 .res/ 디렉토리의 layout xml파일 리소스를 화면에 전개하는 일을 한다. 이것이 무슨 말이냐 하면, 안드로이드에서는 화면 모양을 정의하는 파일이 데이터로 따로 존재하는데 이 화면 모양을 레이아웃이라 하고 이것을 담은 파일 내용대로 안드로이드 운영체제에게 표시해 달라고 한다는 의미이다. 그런데 본 어플리케이션의

MainActivity에서는 레이아웃을 담은 파일조차 없다. 왜냐하면 초기 Medinology에서는 xml을 이용하여 화면 레이아웃을 정의하였다. 이후 화면에 뿐려진 View(버튼, 텍스트 등)들의 id를 얻어오고 그들에게 여러 가지 초기화를 실행했다. 하지만 실행시 매번 Oncreate에서 크래시가 발생하는 상황이 발생했다. 그래서 하나의 해결방안으로 동적으로 뷰를 생성하는 방식을 선택하게 되었다.

3.4 c++언어의 역할과 구동의 흐름

c++은 VB에서 input을 받아 Shell 함수로 medinology_cpp.exe 파일을 호출한다. /▷▷ VB에서 input을 받아 파싱하여 인공신경망 predict 함수의 입력값을 생성한다. main 함수에서 input.txt를 분리한 후 parse 함수, do 함수, save함수를 차례로 호출한다. parse함수는 main에서 잘라놓은 input.txt의 라인들을 파싱하여 증상, 나이, 성별 등의 데이터를 추출한다. do함수는 LoadWeights함수에서 퍼셉트론의 가중치를 담고 있는 weight.txt 파일을 로드하여 신경망의 가중치를 로드하고, predict함수로 질병을 예측한 후, 그것을 분석하여 확률과 병 인덱스를 구한다. save함수는 계산된 결과를 VB를 위한 형식에 맞게 output.txt에 출력한다.

3.5 Trainer 흐름

trainer는 실제로 가중치를 이용해 퍼셉트론을 학습시키는 일을 한다, 이 과정을 흔히 딥러닝이라고 부른다. main함수에서는 데이터베이스를 읽는 LoadDataSet(), 실제 신경망을 학습시키는 Train(), 학습된 가중치를 파일에 기록하는 SaveWeights함수를 차례로 호출한다. LoadDataSet함수는 인터넷에서 csv 파일을 파싱하는 소스를 구해 왔다. 이를 이용하여 symdis.csv를 파싱하여 이 결과를 datasheet 2차원 배열에 저장한다. Train함수는 신경망의 그라디언트 구하는 함수를 호출하여 신경망의 인풋 값에 따른 기울기를 구하게 한 후 getXXX함수들로 그 값을 구해오고, 그 값을 이용하여 신경망의 가중치, 편향을 갱신한다. Train_sub함수가 실제 일을 하고 train함수는 이 과정을 n번 반복시킨다. Save함수는 Train함수에서 업데이트된 신경망의 가중치를 파일에 기록한다. 파일 형식은 행렬의 차원을 첫 줄에 적고 가중치를 개행으로 구별하여 출력하는 것이다.

3.6 안드로이드 애플리케이션 실행흐름

기본 구조는 java side에서는 사용자 입력과 입출력 처리, 그리고 결과 보여주기의 일을 한다. C++ side에서는 실제 예측과 결과 리턴을 한다.

3.7 전체적인 구동의 흐름

안드로이드 앱이 처음 실행되면 호출되는 메소드는 MainActivity class의 OnCreate라고 해도 무방하다. 이 메소드에서는 화면을 초기화하고 데이터베이스를 로드하는 일을 한다. 보통 안드로이드 앱들의 OnCreate에서는 XML 레이아웃 id를 사용하여 안드로이드 운영체제에게 화면을 구성하도록 지시하는데, medinology에서는 버그들로 인해 버그를 고치는 과정에서 동적으로 모든 뷰를 생성하는 방식으로 바꾸었다. 안드로이드의 프로그램 시작점은 MainActivity클래스의 OnCreate() 메소드이다. 이 메소드에서는 주로 앱의 화면과 데이터를 초기화한다 그리고 .res/ 디렉토리의 layout xml파일 리소스를 화면에 전개하는 일을 한다. 이것이 무슨 말이냐 하면, 안드로이드에서는 화면 모양을 정의하는 파일이 데이터로 따로 존재하는데 이 화면 모양을 레이아웃이라 하고 이것을 담은 파일 내용대로 안드로이드 운영체제에게 표시해 달라고 한다는 의미이다. 그런데 본 모바일 앱의 MainActivity에서는 레이아웃을 담은 파일조차 없다. 왜냐하면 초기 Medinology에서는 xml을 이용하여 화면 레이아웃을 정의하였었다. 이후 화면에 뿐려진 View(버튼, 텍스트 등)들의 id를 얻어오고 그들에게 여러 가지 초기화를 실행했다. 하지만 실행시 매번 OnCreate에서 크래시가 발생하는 상황이 발생했다. 그래서 하나의 해결방안으로 동적으로 뷰를 생성하는 방식을 선택하게 되었다. 소스 코드의 일부를 제시하였다. (MainActivity.java 94 101행) 지면상 앞으로 변수 선언하는 부분과 중요하지 않은 부분은 생략하기로 한다.

```
1 mainLayout.addView(firstFrame);
2 setContentView(mainLayout);
3 LoadSymptoms();
4 LoadDiseaseAndMedinames();
5 ReadDisease_Drug(diseaseNames.size(), mediNames.size());
6 ReadDrug_Detail(mediNames.size, 4);
```

2017 창의융합과제연구
Kyunggi High School
퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

LoadSymptoms 메소드에서는 증상명들을 로드하고 이에 따라 체크박스 증상선택 화면을 구성한다. 이 메소드의 일부이다. (MainActivity.java 151~190행 - 파일 파싱)

```
1  BufferedReader br = new BufferedReader(new
2      InputStreamReader(getResources().openRawResource(R.raw.symptoms), "euc-kr"));
3  while ((line = br.readLine()) != null)
4  {
5      buf = new ArrayList<String>();
6      if (i % 2 == 0)
7      {
8          kind = line;
9          kinds.add(kind);
10         if (i != 0)
11         {
12             symptomNames.add(new ArrayList<String>(buf));
13         }else{
14             StringTokenizer tokenizer=new StringTokenizer(line, " ");
15             while (tokenizer.hasMoreTokens())
16             {
17                 sym = tokenizer.nextToken(" ");
18                 buf.add(sym);
19             }
20         }
21         ++i;
22     }
23     if (buf != null)
24     {
25         symptomNames.add(new ArrayList<String>());
26     }
27 }
```

200~253행 - 실제로 뷰 생성후 숨기기

```
1  for (int j=0;j < sz;++j)
2  {
3      String symnam=kinds.get(j);
4      TextView tv=new TextView(this);
5      tv.setText(symnam);
6      <그리드 레이아웃 파라미터 설정 등등 생략>
7      grid.addView(tv, par);
8      int ssz=symptomNames.get(j).size();
9      cboxes=new ArrayList<CheckBox>();
10     for (int k=0;k < ssz;++k)
11     {
12         cboxes.add(new CheckBox(this));
13         CheckBox cb=cboxes.get(k);
14         cb.setText(symptomNames.get(j).get(k));
15         cb.setChecked(false);
16         grid.addView(cb);
17         <중략>
18     }
19     button.setOnClickListener(this); //버튼을 클릭했을때 이 클래스의 핸들러 호출
20     nextButtons.add(button); //다음 화면 넘어가는 버튼 추가
21     grid.addView(button); //이 버튼을 그리드에 추가
22     scr.addView(grid); //스크롤뷰에 그리드 추가
23     frame.addView(scr); //프레임에 스크롤뷰 추가
24     frame.setVisibility(frame.GONE); //일단 보이지 않게 설정
25     mainLayout.addView(frame); //프레임을 메인화면에 등록
26     symptomTabFrames.add(frame); //프레임을 다중 프레임 관리 탭에 등록 (Unused)
```

2017 창의융합과제연구
Kyunggi High School
퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

320 330행 LoadWeights() 메소드 가중치 리소스를 C++에서 읽을 수 있도록 export한다.

```
1 inStream = getResources().openRawResource(R.raw.weights);
2     outStream = new FileOutputStream(file);
3     int length; //copy the file content in bytes
4     while ((length = inStream.read(buffer)) > 0)
5     {
6         outStream.write(buffer, 0, length);
7     }
8 <파일 핸들 닫기 및 로그 남기기 생략>
```

346 388행 Invoke 메소드 내부. C++부분을 호출한다.

```
1 //자료를 제공하여 Native로 보낸다.
2 byte [] syms= toPrimitives(symptombytes);
3 symptombytes.clear();
4 initData(male, preg, age, 50, syms, 31);
5 LoadWeights();
6 calcData();
7 <중략>
8 disid1 = getDisID();
9 <중략 - 질병명과 확률을 얻어온다.>
10 mediid1 = getDrugID(disid1);
11 <중략 - 마찬가지로 약 id를 구해온다.>
12 String medi1=mediIDsToString(mediid1);
13 <중략 - 마찬가지로 약 id에서 약 이름을 얻어온다.>
14 String DiseaseThree=diseaseNames.get(disid3);
15 Intent intent=new Intent(this, ShowActivity.class);
16 intent.putExtra("com.kyunggi.medinology.diseaseone.MESSAGE", new Integer(prob1).toString() +
→ "%확률로 " + DiseaseOne + "이며 치료제는 " + medi1 + "입니다.");
17 <중략 - 마찬가지로 문자열을 결합하여 인텐트에 담는다.>
18 intent.putExtra("com.kyunggi.medinology.diseaseone.DRUGS", mediid1);
19 <중략 - 마찬가지로 여러 기본 정보를 담는다.>
20 intent.putExtra("com.kyunggi.medinology.basic.preg",preg);
21 startActivityForResult(intent); //새로운 화면으로 넘어간다!
22 <예외 처리는 후략>
```

410 438행 질병에 따른 약 id들 얻어오기

```
1 private int[] getDrugID(int n)
2 {
3     int siz=disdrugTable.get(n).size(); // One hot encoded
4     ArrayList<Integer> resa=new ArrayList<Integer>();
5     for (int i=0;i < siz;++i)
6     {
7         if (disdrugTable.get(n).get(i) == 1)
8         {
9             resa.add(new Integer(i));
10        }
11    }
12 <resa Aligning 생략>
13     return res;
14 }
15 private String mediIDsToString(int[] ids)
16 {
17     int len=ids.length;
18     String ret=new String();
19     for (int i=0;i < len;++i)
```

2017 창의융합과제연구
Kyunggi High School
퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

```
20     {  
21         ret += mediNames.get(ids[i]) + " ";  
22     }  
23     return ret;  
24 }
```

515 590행 질병-약 테이블 로드 ; ReadDisease_Drug 메소드, ReadDrug_detail 메소드 생략; 592 621행 ;
임신여부 체크 모순방지처리 메소드 생략; 622 690행 증상 받기 버튼 눌렸을 때 처리

```
1 //버튼이 클릭되었을 때 호출된다.  
2 @Override  
3 public void onClick(View p)  
4 {  
5     <타입 캐스팅 및 iterator인자 초기화 생략>  
6     ScrollView sc= (ScrollView) symptomTabFrames.get(1).getChildAt(0);  
7     GridLayout gv=(GridLayout) sc.getChildAt(0);  
8     for (int j = 0; j < gv.getChildCount(); j++)  
9     {  
10        View v = gv.getChildAt(j);  
11        if (v instanceof CheckBox)  
12        {  
13            symptombytes.add(new Byte((byte)((CheckBox)v).isChecked() ?1: 0))); //어떤 체크박스가  
→ 체크되어있는가?  
14        }  
15    }  
16 }  
17 else  
18 {  
19     age = NP_Age.getValue();  
20     preg = CB_Preg.isChecked();  
21     male = RB_Male.isChecked();  
22     if(!male&&!RB_Female.isChecked())  
23     {  
24         Toast.makeText(this,"성별을 체크해 주세요.",3).show();  
25         return;  
26     }  
27     firstFrame.setVisibility(View.GONE);  
28 }  
29     symptomTabFrames.get(1).setVisibility(View.VISIBLE);  
30     if (i == sz - 1)  
31     {  
32         Invoke();  
33     }  
34     mainLayout.invalidate();  
35     <후략>
```

다음은 사용자에게 결과를 출력하는 ShowActivity 소스 중 중요한 일부를 제시한 것이다.
ShowActivity 22 47행 TTS 처리 부분

```
1 public void onInit(int p1)  
2 {  
3     Locale loc = new Locale("korea");  
4     if (myTTS.isLanguageAvailable(loc) == TextToSpeech.LANG_AVAILABLE)  
5     {  
6         myTTS.setLanguage(loc);  
7     }  
8     else{  
9         myTTS.setLanguage(Locale.KOREA);  
10    }
```

2017 창의융합과제연구
Kyunggi High School
퍼셉트론 인공 신경망을 활용한 약 처방 자동화 시스템

```
11     myTTS.setPitch((float) 0.1);
12     myTTS.setSpeechRate(1); // 빠르기 설정 1이 보통
13     myTTS.speak("결과. 이용자님의 진단 질병과 해당 약은 다음과 같습니다.", 
14             → TextToSpeech.QUEUE_FLUSH, null); // tts 변환되어 나오는 음성
15     myTTS.speak(disease1, TextToSpeech.QUEUE_ADD, null); //QUEUE_FLUSH 다음에 나오는
16             → QUEUE_ADD
17 <후략>
```

93 131행 MakeComments() 메소드 약에 대한 주의 사항 만드는 부분

```
1 ret=MainActivity.mediNames.get(ids[i]);
2     inisize=ret.length();
3     ret+="은(는)";
4     ArrayList<Integer> arr=drugditTable.get(ids[i]);
5     if(arr.get(0)==1){
6         ret+=" 처방이 필요합니다.";
7     }
8     if((arr.get(1)==1)&&(preg)){
9         ret+=" 임산부에게 위험합니다.";
10    }
11 <후략>
```

49 86행 전달받은 정보를 화면에 출력하는 부분

```
1 public void onCreate(..)
2 {
3     ReadDisease_Drug(31,36);
4     setContentView(R.layout.showresult);
5     // Get the message from the intent
6     Intent intent = getIntent();
7     disease1 = intent.getStringExtra("com.kyunggi.medinology.diseaseone.MESSAGE");
8     <중략>
9     TextView disThreeTextView=(TextView)findViewById(R.id.diseasethreeTextView);
10    disThreeTextView.setText(disease3);
11    <중략>
12    int[] mediid3=intent.getIntArrayExtra("com.kyunggi.medinology.diseasethree.DRUGS");
13    age=intent.getIntExtra("com.kyunggi.medinology.basic.age",20);
14    <성별과 임신여부를 인텐트에서 추출하기 생략>
15    comment1=BuildComment(mediid1);
16    <약에 대한 주의사항 문자열 생성하기 중략>
17    TextView commenttv3=(TextView) findViewById(R.id.commentThree);
18    commenttv3.setText(comment3);
19    myTTS = new TextToSpeech(this, this); //곧 OnInit() 호출된다.
20 }
```

다음은 MainActivity의 native functions 를 구현한 것이다.

```
1 JNIEXPORT void JNICALL Java_com_kyunggi_medinology_MainActivity_initData(JNIEnv* env, jobject
2     → thiz,jboolean male, jboolean _preg,jint _age,jint _weight,jbyteArray _symptoms,jint
3     → _diseases)
4     {
5         gender=male==JNI_TRUE?MALE:FEMALE;//성별 저장
6         preg=_preg;//임신 여부 저장
7         age=_age;//나이 저장
8         diseasenum=_diseases;//행렬의 차원 테스트용
9         symptomlen=env->GetArrayLength(_symptoms); //JNI에서 배열 액세스
10        symptoms= new float[symptomlen];
11        jbyte *byte_buf = env->GetByteArrayElements(_symptoms, NULL);
```

```
10     for(int i=0;i<symptomlen;++i)
11     {
12         symptoms[i]=(float)byte_buf[i];
13     }
14     env->ReleaseByteArrayElements(_symptoms, byte_buf, 0);
15 }
16 JNIEXPORT void JNICALL Java <중략> _calcData(JNIEnv* env, jobject thiz)
17 {
18     MatrixXd x(1,symptomlen);
19     for(int i=0;i<symptomlen;++i)
20     {
21         x(0,i)=symptoms[i];
22     }
23     MatrixXd result=net.Predict(x);           //예측
24     int i,j;
25     prob1=result.maxCoeff(&i,&j)*100;    //확률이 가장 높은 것의 확률과 인덱스를 구한다.
26     result(i,j)=0;      //두 번째 결과를 얻기 위해 이것을 0으로 만들어
27     disease1=j;
28     prob2=result.maxCoeff(&i,&j)*100;    //또다시 확률과 인덱스를 구한다.
29     <후략>
```

getProb(), initWeights(), finalizeNative() 함수는 간단하므로 지면상 생략합니다. 나머지 핵심 함수들은 PC 버전과 동일하다.

3.8 실제 구동 사진 - Medinology Andriod 버전

세 번째 화면에서 면책 조항도 나오는데 이는 이 프로그램이 아직 초기 버전이므로 예측이 틀릴 수 있고, 이것이 중대한 결과를 초래할 수 있기 때문에 주의해야 하고, 의사 및 약사와 상담이 필요함을 당부하고 있다.

3.9 실제 구동 사진 - Medinology PC 버전

4 결론

결론적으로, 정리해보자면 C++에서 코딩한 프로그램은 퍼셉트론 클래스를 담당하는 중추로. 크게 4 가지 역할이 있다. Visual Basic에서 전달된 환자의 건강 정보 자료를 파싱하는 역할, weight.txt 파일의 내용을 신경망의 가중치, 편향에 로드하는 역할, 신경망을 생성하여 초기화하고 Predict함수를 호출하여 결과를 계산하는 역할 그리고 계산 결과를 분석하여 질병명을 확률 순서대로 정렬하여 출력하는 일을 한다.

Visual Basic에서 코딩한 프로그램은 UI 프로그램으로, 정보들을 이 프로그램에 입력하면 입력된 데이터를 C++로 코딩된 퍼셉트론 클래스로 전달하고 출력데이터를 다시 전달해주는 역할을 한다. 그리고 JNI는 일련의 과정을 앱에서 진행할 수 있도록 포팅을 해준다. 이러한 일련의 과정을 거쳐 모바일 앱이 구동된다. 퍼셉트론 클래스도 성공적으로 빌드하였고 모바일 앱도 정상적으로 잘 구동되지만 아쉬운 점이 하나 있다.

약을 처방해주는 것은 환자의 안전과 건강이 직결된 문제인 만큼 굉장히 중요한 사안이다. 그만큼 의학과 약학에 관해서 전문적인 지식이 필요하다. 하지만 우리는 전문가가 아니고 학습데이터를 제작함에 있어서 전문 서적을 여러 권 찾아보는 등의 노력을 하였지만 의학적으로 완전하고 신뢰할 수 있는 정보라고 할 수 없다. 그리고 빅 데이터가 아닌 짧은 기간 동안 사람이 엑셀작업으로 직접 만든 데이터로 데이터의 양이 충분히 많지 않다. 따라서 모바일 앱에서 처방해주는 약과 병은 정확하지 않을 수 있다. 하지만 이러한 문제점들은 의학, 약학 관련 전문가들의 도움을 받아 데이터를 수집하기만 한다면 쉽게 해결 할 수 있다. 또한 기존 시장에 존재하지 않던 프로그램을 제작함으로써 다른 개발자들의 참여를 유도 할 수 있고 병원 등 의료시설에 접근하기 어려운 도시 노인이나 빈민, 촌락의 주민들이 기초적 의료 서비스를 받을 수 있도록 할 수 있다. 자신의 증상에 대한 병명을 알려주고 경고함으로써 병원에 가지 않으려는 사람을 설득할 수 있다. 그리고 사람의 실수를 막아주는 컴퓨터의 막강한 장점을 적극 활용하여 의료사고를 줄이는데 기여할 수 있다. 또한 간단한 질병이나 증상은 꼭 병원을 가지 않더라도 편리하게 약을 선택 할 수 있으며 만약 이 시스템을 활용하여 자판기와 같은 것을 만든다면 약국이 문을 닫았을 때 혹은 약국이 없는 지역에서 유용하고 편리하게 사용할 수 있다. 이렇듯 우리가 공부한 내용을 가지고 직접 사회에 도움이 될 만한 시스템을 만들어 냈다는 것에 가장 큰 의의가 있다.

5 참고문헌

- 알찬 진료의를 위한 약처방 가이드 개정2판 대한내과학회 편(한국의학원)
- 맙바닥부터 시작하는 딥러닝 사이토 고키 저 (한빛미디어)

List of Figures

List of Listings