2021년 2학기

**프로그래밍과 문제해결**

**Assignment #3**

담당교수: 윤은영

학번: 20210370

학과: 무은재학부

이름 권민경

POVID ID: mkkwon

명예서약(Honor Code)

“나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.”

**String functions와 Naïve Bayes Classifier를 활용한 스팸문자 분류**

1. **문제의 개요**

* Problem 1: string.h 안에 들어있는 함수를 string.h없이 직접 구현한다.
* Problem 2: Spam 과 Ham 메일을 구분하는 프로그램을 구현한다. Txt 파일을 읽어와 단어를 저장하고 처리하여 각각 Spam 문장에서 단어가 나오는 횟수, Ham 문장에서 단어가 나오는 횟수를 다른 txt에 저장한다.
* 저장 후 Bay’s theorem 을 이용하여 어떤 문장이 Spam인지 Ham 인지 테스트 한다.

Problem 2 구조

입력부: 처음 메뉴 화면에서 어떤 기능을 수행할지 입력 받는다. 또한 train 과정에 있어 train하고 싶은 파일을 입력 받는다.

처리부: 입력 받은 파일에서 단어를 저장하고 ham에서 등장한 단어와 spam에서 등장한 단어를 구분 짓는다. 이때 hfreq과 sfreq를 만들어 각각의 문장에서의 단어의 등장 횟수를 계산한다. 또한 test 에서 Bayes’ theorem으로 숫자들을 계산한다.

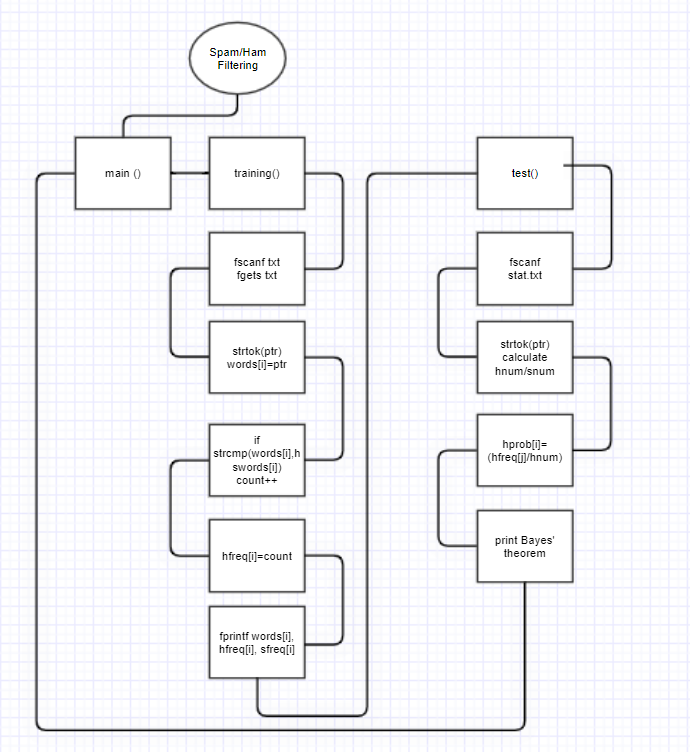
출력부: train에서 단어의 등장 횟수를 출력화면에 프린트해야 하며, 마찬가지로 stat.txt화면에 결과를 출력한다. test에서는 Bayes’ theorem을 통해 문장이 spam인지 ham인지 판단하는 과정과 결과를 출력한다.

1. **알고리즘**

본 프로그램 작성을 위해 Pseudo코드 형태로 나타내면 다음과 같다.

|  |
| --- |
| Pseudo-algorithm for spam/ham classification |
| //변수는 선언된 것으로 가정   1. Include essential headers <stdio.h>, <stlib.h>, <windows.h> and <string.h> 2. Print menu screen 3. For train, scan one sentence from txt, put each word in the array 4. Check if the word belongs to Ham or Spam sentence 5. Print the number (i.e. our 2,0) 6. For test, scan one sentence from txt, put each word in the array 7. Input for the sentence, cutting it down into words 8. Calculate frequencies and print whether the sentence is spam or ham. |

위의 의사 알고리즘을 flow chart를 통해 표현하면 아래와 같다.



1. **프로그램 구조 및 설명**
2. 시작화면 실행

**Train**

1. fscanf를 통해 명령한 파일을 입력 받음.

Train. txt파일을 연다. 파일의 원소를 fgets를 통해 한줄씩 파일을 읽어온다. 읽어온 후 strtok를 사용해 문장을 자르고 자른 문장을 words[i]에 넣는다. 이때 단어의 개수는 malloc을 사용해 처리하고 5개가 넘어갈경우 realloc을 사용한다. 단어를 저장하는 배열도 마찬가지로 malloc을 사용한다.

1. Hwords와 Swords저장

Ham 문장을 읽어와 마찬가지로 strtok로 문장을 자르고 hwords[i]에 넣는다. Spam 문장은 swords[i]에 넣는다. 저장후 words[i]와 hwords[i]를 비교하며 만약 hword와 words가 일치할경우 count의 숫자를 올려 단어 수를 센다. 센 단어수를 hfreq[i] 배열에 저장한다.

1. 학습한 결과 출력

우선 실행 화면에 출력해야 한다. Words[i], hfreq[i], sfreq[i] 이 순서대로 출력하고 txt 파일에 출력하기 위해 fprintf를 이용하여 출력한다.

**Test**

1. Stat.txt를 읽어오기

파일을 한 문장 씩 일어와 strtok를 사용해 문장을 자른다. 자른후 단어를 words[i]에 넣는다. 이때 마찬가지로 동적할당을 사용한다.

1. 문장 입력 받기

테스트 하고 싶은 문장을 입력 받는다. 이때도 마찬가지로 strtok를 통해 문장을 자르고 stwods[i]에 넣는다.

1. Bayes’ theorem 계산

첫번째 p(Ham)과 p(Spam)은 단순히 나눗셈으로 구상하면 된다. 이후 등장하는 p(wi|Ham)과 p(wi|Spam)은 겹치는 횟수/nham으로 계산한다. 다음 계산을 위해 이전 계산값은 hprob sprob에 저장한다.

1. **프로그램 실행방법 및 예제**
2. 시작화면

만약 train, test, exit이 아닌 파일을 입력한다면 Error: invalid input이라는 메시지가 출력되고 1초후 다시 메인 화면을 출력한다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

1. train함수 시작

train을 입력할 경우 filename 을 입력 받는다. 이때, 입력값과 일치하는 파일이 없으면 Error: File does not exist를 출력한다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

학습을 완료한 뒤 결과 창을 실행 화면에 출력한다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

실행화면에 출력한 결과를 stat.txt에도 출력한다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

1. test함수 시작

메시지를 입력 받는다. 이후 메시지로 계산한 결과를 출력한다.

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

1. train1.txt로 실행한 결과

텍스트, 전자기기이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명

**텍스트이(가) 표시된 사진

자동 생성된 설명**

1. **토론**

* 재귀함수의 사용이 메모리를 차지해 에러를 일으킬 수 있다는 것을 배웠기 때문에 코드를 짜면서 재귀함수를 사용하지 않았다.
* 처음에 free로 할당 해제를 해주는 부분에서 계속 에러가 났다. 처음에는 realloc 하는 타이밍이 맞지 않아서 생긴 에러인줄 알고 그것을 수정하려고 노력했다. 하지만 계속 에러가 발생해서 할당하는 부분을 수정하려고 노력했다. 그러던 중 데이터타입이 char인 부분에 size of int로 할당하고 있는 것을 발견했고, 즉시 이부분을 수정해서 free가 정상 작동될 수 있게 하였다.
* Bayes’ theorem으로 probability를 계산하는 것이 쉽지 않았다. Strtok를 실행한 다음 무조건 probability를 계산하려고 했는데 최종 단계에서 SPAM/HAM을 구별해내는것이 쉽지 않았다. 그래서 probability를 저장하는 배열을 따로 만들어서 이 문제를 해결하였다.

1. **결론**

* 이번 어싸인을 통해 string을 처리하는 방법을 완벽하게 이해했다. 또한, 동적할당을 하고 해제하는 부분의 순서를 이해했다. 이것이 이후 자료를 처리하는데 도움이 될 것이다.
* 문제 pdf에 하나의 문장 내에 단어가 여러 번 등장하더라도, 등장 여부만을 고려하여 한 번만 센다고 나와있는데, 처음에 그냥 모든 단어를 셈에 포함하여서 example에 나온 결과와 같지 않았다. 그래서 저장된 문장을 검사하는 함수를 따로 만들어서 이미 문장 내에 단어가 나왔다면 셈에 포함하지 않게 하였다.
* Strcmp/ strlen등 string.h 에 포함된 주요 함수의 기능을 이해했다.

1. **개선 방향**

* 처음 코드를 짰을 때 반복되는 구문이 매우 많았다. 반복을 줄이기 위해 코드 수정을 여러 번 해야했는데, 만약 다시 짜보라고 한다면 반복되는 부분을 함수로 만들고 싶다.
* 이중 포인터 변수 할당, 할당 해제가 있었는데, 처음에 많이 고민되었다. 때문에 수업 노트와 랩시간에 배운 내용을 복습했고, 이후 어떻게 처리하는지 알게 되어 더 고차원의 배열도 사용할 수 있을 것 같다.