

로그인하세요.
sign in sign up

뉴스 피드

포럼

뉴스
자유게시판
질문과 답변
과거 게시판

위키

페이지 목록

온라인 저지

문제 풀기
랜덤 문제 고르기
최근 제출된 답안
사용자 랭킹
튜토리얼

캘린더

알고스팟 대화방

초대장 받기
이용 안내


검색하기

AOJ 문제 바로가기

다가오는 이벤트들

Hacker Cup 2018 Round 3
(8/19 02:00)

see all



문제 정보

문제 ID	시간 제한	메모리 제한	제출 횟수	정답 횟수 (비율)
OCR	60000ms	65536kb	1257	316 (25%)
출제자	출처	분류		
JongMan	알고리즘 문제 해결 전략	보기		

문제

“ 알림: 채점 서버 속도 문제로 시간 제한을 60초로 늘리고, 테스트 케이스 수를 20으로 줄입니다.

광학 문자 인식(Optical Character Recognition)은 사람이 쓰거나 기계로 인쇄한 글자를 스캔한 이미지를 다시 기계가 읽을 수 있는 문자로 변환하는 과정을 말합니다. OCR 알고리즘들은 대개 수많은 필기 샘플을 통계적으로 분석하고 패턴을 찾아내어 각 단어들을 인식하곤 합니다. 하지만 단순히 각 단어들을 개별적으로 인식하기 보다, 단어의 분포나 문법 등을 고려하면 더 나은 결과를 얻을 수 있는 경우가 많습니다. 이 문제에서는 과거 자료로부터 추출한 정보를 이용해 문자 인식의 정확도를 높여 봅시다.

과거에 인식했던 수많은 문장들을 분석해 원본 문장의 형태를 파악하려고 합니다. 이 작업을 위해 우선 과거 자료에 출현하는 모든 단어의 목록을 만든 뒤, 각 단어가 문장의 첫 단어로 사용된 비율을 계산했습니다. 그리고 각 단어 쌍에 대해, 한 단어가 다른 단어 다음에 출현할 확률을 계산했습니다. 이때 우리가 인식해야 할 원본 문장은 과거 자료와 똑같은 분포를 가진다고 가정합니다. 달리 말해 이 확률 테이블만 있으면 어떤 원본 문장이 출현할 확률을 정확히 계산할 수 있다고 가정한다는 얘깁니다.

우리의 문자 인식 알고리즘은 원문 그림을 여러 조각으로 쪼갠 후 각 조각을 비슷해 보이는 단어로 분류합니다. 이 분류하는 알고리즘을 분류기(classifier)라고 부릅니다. 이 분류기는 완벽하지 않기 때문에 특정 단어를 다른 단어로 잘 인식할 수도 있습니다. 예를 들어 boy라는 단어를 buy나 bay로 인식할 수 있다는 이야기입니다. 수많은 예제 입력에 대해 분류기를 시험하여, 각 단어가 적힌 조각을 분류기에 입력했을 때 어떤 출력을 얻을 수 있는지, 그리고 각각의 확률은 얼마였는지를 계산했습니다. 예를 들어 분류기에 실제 boy라고 씌어 있는 조각을 입력했을 때, 정확하게 boy로 인식할 확률은 0.7, bay일 확률은 0.25, buy일 확률은 0.04, bye일 확률은 0.01이었다는 식입니다.

이와 같은 정보들을 이용하면 좀더 나은 문자 인식을 할 수 있습니다. 각 조각을 앞에서 예로 든 분류기를 이용해 인식한 결과 "I am a bay."라는 문장을 결과로 얻었다고 합시다. 그런데 자료를 살펴보니 a 후에 bay가 올 확률은 얼마 없는 반면, a 후에 boy가 올 확률은 매우 컸다고 합시다. 우리의 분류기가 bay라고 인식한 조각이 사실은 boy일 확률이 0.25나 되기 때문에, 이 문장의 인식 결과를 "I am a boy."로 고치는 편이 더 올바른 분류일 것입니다.

어떤 문장을 단어별로 인식한 결과가 주어졌을 때, 원본일 조건부 확률이 가장 높은 문장을 찾아내는 프로그램을 작성하세요.

입력

입력은 분석이 끝난 과거 자료의 통계치와, 분류기가 인식한 문장으로 구성됩니다.

입력의 첫 줄에는 원문에 출현할 수 있는 단어의 수 m ($1 \leq m \leq 500$)과 처리해야 할 문장의 수 q ($1 \leq q \leq 20$)가 주어집니다.

두 번째 줄에는 원문에 출현할 수 있는 m 개의 단어가 공백으로 구분되어 주어집니다. 각 단어는 알파벳 대소문자로만 구성되어 있습니다. 모든 단어의 길이는 10 이하입니다.

세 번째 줄에는 각 단어가 문장의 처음에 출현할 확률 $B[i]$ 가 m 개의 실수로 주어집니다. $B[i]$ 는 i 번 단어가 첫 단어로 출현할 확률입니다. 모든 $B[i]$ 의 합은 1입니다.

그 후 m 줄에 $m \times m$ 크기의 실수 행렬 T 가 주어집니다. 이 행렬에서 i 행 j 열의 숫자 $T[i, j]$ 는 i 번 단어의 다음 단어가 j 번 단어일 확률을 나타냅니다. 각 행에 있는 확률의 합은 항상 1입니다.

그 후 m 줄에 $m \times m$ 크기의 실수 행렬 M 이 주어집니다. 이 행렬에서 i 행 j 열의 숫자 $M[i, j]$ 는 i 번 단어가 적힌 조각을 j 번 단어로 분류할 확률을 나타냅니다. 각 행에 있는 확률의 합은 항상 1입니다.

그 후 q 줄에 한 줄에 하나씩 분류기로 인식한 문장이 주어집니다. 각 줄의 처음에 단어의 수 n ($1 \leq n \leq 100$)이 주어지고, 그 후 n 개의 단어로 분류기의 인식 결과가 주어집니다. 모든 단어는 처음에 주어진 m 개의 단어 중 하나입니다.

입력의 크기가 크므로 빠른 입력 방식을 사용하기를 권장합니다.

출력

한 문장마다 한 줄에 주어진 인식 결과에 대해 조건부 출현 확률이 가장 높은 문장을 출력합니다. 주어지는 입력에서 가장 확률이 높은 문장이 여러 개인 경우 어느 것을 출력해도 좋습니다.

예제 입력

```
5 3
I am a boy buy
1.0 0.0 0.0 0.0 0.0
0.1 0.6 0.1 0.1 0.1
0.1 0.1 0.6 0.1 0.1
0.1 0.1 0.1 0.6 0.1
0.2 0.2 0.2 0.2 0.2
0.2 0.2 0.2 0.2 0.2
0.8 0.1 0.0 0.1 0.0
0.1 0.7 0.0 0.2 0.0
0.0 0.1 0.8 0.0 0.1
0.0 0.0 0.0 0.5 0.5
0.0 0.0 0.0 0.5 0.5
4 I am a buy
4 I I a boy
4 I am am boy
```

예제 출력

```
I am a boy
I am a boy
I am a boy
```

노트

6개의 댓글이 있습니다.