

[과제 1] 한글 문장의 유사도 계산 (1)

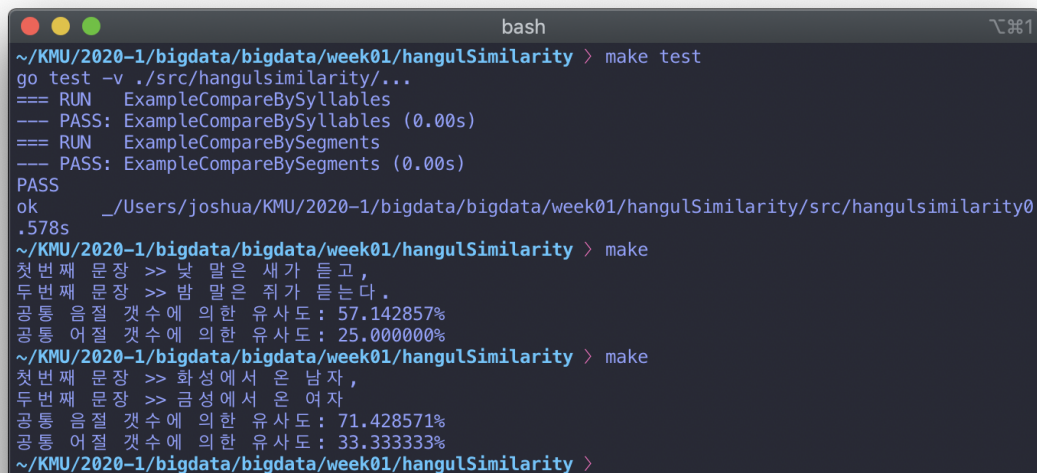
1. 한글 문장의 유사도 계산 (1)

- 입력: 한글 문장 2개 (매우 유사하거나 약간 유사한 문장)
- 출력: 유사도 (%)
- 방법: 공통 음절 개수, 공통 어절 개수 등에 의한 유사도 계산

<참고 1> 위 과제 수행에 사용하는 언어는 C/C++, 자바, 파이썬 등 각자 사용하기 편한 언어를 사용하면 됩니다.

<참고 2> 과제제출 내용 : 소스코드, 보고서 (PDF 파일: 구현 방법 및 실행하면 스샷 등의 설명 포함) → zip 파일 1개로 업로드

실행 결과



```
bash
~/KMU/2020-1/bigdata/bigdata/week01/hangulSimilarity > make test
go test -v ./src/hangulSimilarity/...
=== RUN   ExampleCompareBySyllables
--- PASS: ExampleCompareBySyllables (0.00s)
=== RUN   ExampleCompareBySegments
--- PASS: ExampleCompareBySegments (0.00s)
PASS
ok      _/Users/joshua/KMU/2020-1/bigdata/bigdata/week01/hangulSimilarity/src/hangulSimilarity0
.578s
~/KMU/2020-1/bigdata/bigdata/week01/hangulSimilarity > make
첫 번째 문장 >> 낮 말은 새가 듣고,
두 번째 문장 >> 밤 말은 쥐가 듣는다.
공통 음절 갯수에 의한 유사도 : 57.142857%
공통 어절 갯수에 의한 유사도 : 25.000000%
~/KMU/2020-1/bigdata/bigdata/week01/hangulSimilarity > make
첫 번째 문장 >> 화성에서 온 남자,
두 번째 문장 >> 금성에서 온 여자
공통 음절 갯수에 의한 유사도 : 71.428571%
공통 어절 갯수에 의한 유사도 : 33.333333%
~/KMU/2020-1/bigdata/bigdata/week01/hangulSimilarity >
```

Installation

```
go get github.com/joshua-dev/bigdata/week01/hangulSimilarity/src/hangulsimilarity
```

Run

```
make
```

Test

```
make test
```

- 제출하는 모든 과제 및 퀴즈의 소스 코드는 아래 깃허브 경로에 있습니다.

```
https://github.com/joshua-dev/bigdata
```

구현 방법

- 공통 음절 갯수에 의한 유사도 측정

다음과 같은 순서로 두 한글 문장의 유사도를 공통 음절 갯수를 기준으로 측정했다.

- 1. 문장 하나를 받아 문장 부호를 제거하고 각 음절이 얼마나 등장하는지 센다.
- 2. 두 문장을 받아 1의 함수로 음절을 세고 음절 수가 짧은 문장을 기준으로 공통 음절의 갯수를 센다.
- 3. 공통 음절의 갯수를 짧은 문장의 음절 수로 나누고 100을 곱하여 반환한다. (% 단위이므로)

- [1]을 구현한 함수 countBySyllables

```
// countBySyllables returns a map containing syllable counts of a given sentence and number of syllables.  
func countBySyllables(sentence string) (map[string]int, int)
```

이 때, 문장 부호를 제거하기 위해 정규식과 regexp 패키지를 이용하여 문장 부호를 제거하는 cleanse 함수를 만들었다.

```
// cleanse returns a string with punctuation removed.  
func cleanse(s string) string
```

- [2], [3]을 구현한 함수 CompareBySyllables

```
// CompareBySyllables returns similarity of given two strings  
// based on the common syllables.  
func CompareBySyllables(first, second string) float64
```

- 공통 어절 갯수에 의한 유사도 측정

다음과 같은 순서로 두 한글 문장의 유사도를 공통 어절 갯수를 기준으로 측정했다.

- 1. 문장 하나를 받아 문장 부호를 제거하고 공백 단위로 parsing하여 각 어절이 얼마나 등장하는지 센다.
- 2. 두 문장을 받아 1의 함수로 어절을 세고 어절 수가 짧은 문장을 기준으로 공통 어절의 갯수를 센다.
- 3. 공통 어절의 갯수를 짧은 문장의 어절 수로 나누고 100을 곱하여 반환한다. (% 단위이므로)

- [1]을 구현한 함수 countBySegments

```
// countBySegments returns a map containing segment counts of a given sentence and number of segments.  
func countBySegments(sentence string) (map[string]int, int)
```

공통 음절 갯수로 측정할 때와 같이 여기에서도 문장 부호를 제거하기 위해 cleanse 함수로 문장 부호를 제거한다.

- [2], [3]을 구현한 함수 CompareBySegments

```
// CompareBySegments returns similarity of given two strings  
// based on the common segments.  
func CompareBySegments(first, second string) float64
```