2015 국어 정보 처리 시스템 경진 대회

발표 자료집

- 일자 : 2015년 10월 16일(금)
- 장소 : 전주대학교 스타센터 다목적홀
- 주관 : 한국정보과학회 언어공학연구회
- 주최 : 국립국어원

국립국어원

대 회 일 정

■ 등 록

13:00 ~ 13:30 등록 및 방명록 작성, 명찰 및 발표 자료집 배부

■ 개 회 식 사 회: 김학수 교수

13:30 ~ 13:40 개회식 개회사: 김재훈 교수

환영사: 김선철 언어정보과장 (국립국어원)

■ 발표 1부 (지정 분야: 국립국어원 질의-응답 시스템) 좌 장: 최정도 학예연구사 (국립국어원)

13:40 ~ 14:00 **서강 알짱 cQA**

권순재, 김주애, 신해빈, 안웅찬, 정유진, 서정연 (서강대)

14:00 ~ 14:20 **DNN**

최경호, 황현선, 오준호, 김건영, 이창기 (강원대)

14:20 ~ 14:40 **국어정보 키워드 추출방법을 이용한 질의응답 시스템**

전석종, 이수인, 이현아 (금오공대)

■ 시 연

14:40 ~ 15:10 본선 7팀 시스템 시연

■ 발표 2부 (일반 분야) 좌 장: 김유섭 교수

15:10 ~ 15:30 **누르미 – 터치 동작 기반 키보드**

박형순, 김민호, 박소영, 김도경, 김두환, 최윤승, 강승식 (국민대)

15:30 ~ 15:50 **한국어 문장 분할 및 구 묶음 추출 도구(KoSeCT)**

남상하, 원유성, 우종성, 함영균, 최기선 (KAIST)

15:50 ~ 16:10 **ESPRESSO TOOL**

박태호, 차정원 (창원대)

16:10 ~ 16:30 **Beoltong**

김중한, 차정원 (창원대)

■ 초청 강연

16:30 ~ 17:30 시간-공간 표준과 한국어 적용 워크샵 사 회: 이상곤 교수

김보겸/이재성(충북대), 유현조(서울대), 정영섭/최호진(KAIST)

17:30 ~ 18:10 훈민정음 창제 원리의 과학성 사 회: 이상곤 교수

변정용(동국대)

■ 시상식 및 폐회식 사 회: 김학수 교수

18:10 ~ 18:30 시상식 및 폐회식 시 상: 김선철 과장

심사평: 강현규 교수

목 차

서강 알짱 cQA

권순재, 김주애, 신해빈, 안웅찬, 정유진
서강대학교 컴퓨터공학과

차 례

제 1 장 소프트웨어 소개

1.1 소프트웨어 명칭

서강 알짬 cQA
: 알짬이란 여럿 가운데에 가장 중요한 내용이라는 뜻의 순 우리말로, 5,336개의 QA 쌍 중에서 가장 중요한 내용을 알려주겠다는 의미를 담고 있다.

1.2 소프트웨어 사용 환경

- Microsoft Windows 7 64bit 이상
- 실시간 가용 RAM 8GB 이상 (heap space 1GB 이상)
- JAVA 언어
- Eclipse jre 8, jdk 1.8 이상
- 인코딩 UTF-8

1.3 소프트웨어 특징

'서강 알짬 cQA'는 서강대학교 학부생과 석사생이 개발한 cQA System이다. cQA 시스템은 Community based Question/Answer System의 준말로, 사용자의 질문에 대하여 그것과 일치하거나 유사한 질문이 기존 커뮤니티에 존재한다면, 그 질문에 대한 답을 정답으로 제시한다. 그러므로 사용자는 이 시스템으로부터 받은 기존 질의문답을 통해 자신이 궁금했던 점을 해결할 수 있다. 본 소프트웨어는 국립국어원에 등재된 5,336개의 Q/A쌍을 바탕으로 사용자 질의에 대해 5,336개 중, 답 후보 Q/A쌍 10개를 출력한다.

일반적으로 질의가 비슷하면 비슷한 내용을 내포할 것이므로, 10개의 정답 후보 Q/A쌍과 사용자 질의의 유사도를 구하게 될 것이다. 하지만 cQA 시스템의 특성상 단순 유사도 계산만으로는 사용자가 원하는 답변의 Q/A쌍을 올바르게 제시하지 못 할 수 있다. 커뮤니티 내에서 사람들은 같은 내용을 다양한 형태로 질문하기도 하며, 실질적 질문과 상관관계가 낮은 내용도 다소 포함하여 질문하기 때문이다. 그래서 이러한 점을 해결하기 위해 본 시스템에서는 다음 두 가지 기법을 이용하여 답의 정확도를 향상시켰다.

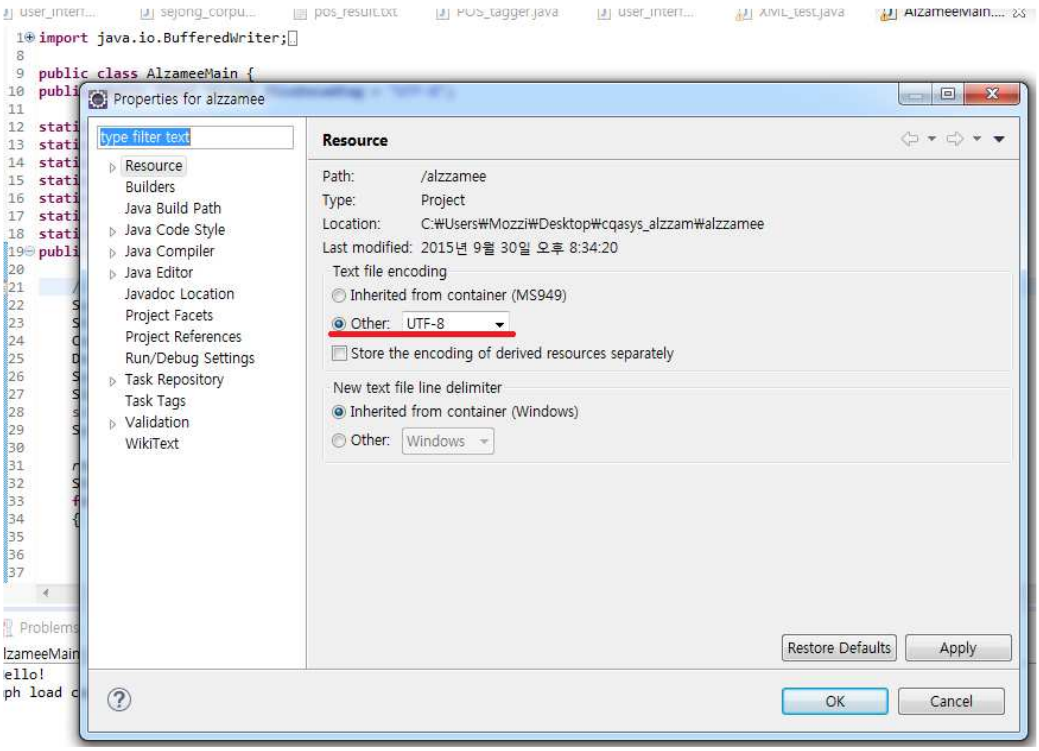
첫 번째 기법은 질의에서 실질적인 질문 내용이 되는 중요한 키워드를 뽑는 작업이다. 본 시스템에서 이 중요 키워드를 Focus라 부르도록 한다. 이는 Conditional Random Field 기법으로 5,336개의 Q/A쌍에서 중요 키워드 chunk를 학습 시킨 후, 사용자 질의를 학습 모델에 넣어서 Focus Chunk를 얻는 방법으로 구현되었다. 두 번째 기법은 질문에 카테고리를 할당하는 것이다. 카테고리를 할당함으로써 질의와 5,336개의 Q/A 쌍에서 단순 단어만 보았을 때 유사하지 않더라도 같은 카테고리를 유사한 질문으로 인식할 수 있게 한다. 이 기법은 Support Vector Machine으로 5,336개의 Q/A 쌍에 카테고리를 태깅한 데이터로 지도학습을 하여 구현하였다.

본 시스템은 위 두 가지 질의 분석 방법을 통해 질의 분석의 질을 향상시켰다. 특히 Machine Leraning(SVM, CRF)기법을 사용하여 시스템의 높은 확장성을 보장하였고, Rule based 학습을 최대한 배제함으로써 시스템의 견고성을 최대화 하였다.

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

1. cqasys_alzzam.zip 파일 압축 해제
2. alzzamee 프로젝트 폴더를 Import
3. 프로젝트 인코딩을 UTF-8로 반드시 변경



2.2 소프트웨어 파일 구조

2.2.1 주요 파일 설명

① 실행 파일

cqasys_alzzam/alzzamee/run.bat

② 데이터 파일

AllDocuments/ --- 5,336개의 QA 쌍들의 xml파일을 Question과 Answer로 나누어 txt 파일로 저장해 놓은 것

crf/ --- 5,336개의 파일들에서 Focus를 미리 추출해 놓은 것들의 set

crf/newModel --- 새 질의가 들어왔을 때, Focus를 추출하기 위한 CRF 모델 파일

data/ --- 형태소 분석기(KACREIL), CRF 라이브러리(mallet) 등 외부 jar 파일

Summary/ --- 5,336개의 데이터의 핵심 내용이 되는 summary 파일

svm/data --- libsvm을 사용하기 위한 data

svm/model --- 질의 분류를 위해 학습시켜 놓은 모델

svm/result --- 테스트 할 질의를 7개의 각 카테고리에 포함되는지에 대한 결과를 작성한 파일

svm/test --- predicate 할 질의의 Feature Map 파일

svm/word_chi --- 카이제곱으로 구한 상위 100개의 자질

③ 사전 파일

없음

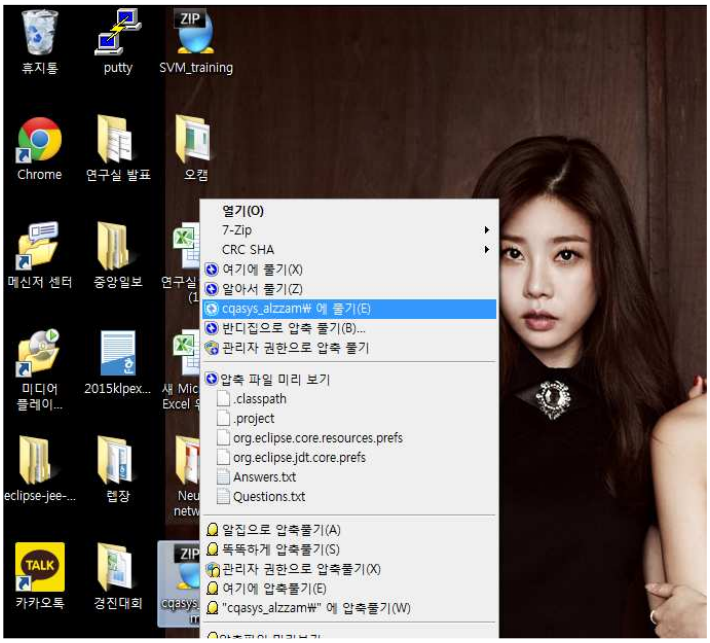
2.2.2 전체 구조

alzzam/

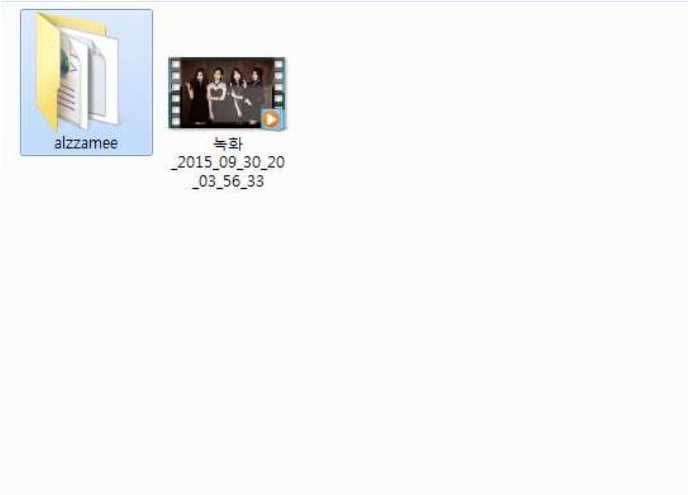
- **src/** .....소스파일폴더
  - **KACRTEIL-KMA.jar**..형태소분석기 라이브러리
  - **mallet-deps.jar**.....CRF Chunking 라이브러리
  - **mallet.jar**.....CRF Chunking 라이브러리(light ver.)
  - **libsvm.jar**.....,,,,,.SVM 라이브러리
- **AllDocuments/**....5,336개의 Q/A 쌍 정보
- **crf/**.....Focus 추출 데이터 및 추출 모델
- **data/**.....외부 jar 파일 및 데이터
- **summary/**.....5,336 Q/A쌍의 summary 데이터가 담긴 폴더
- **svm/**.....SVM 사용시 필요한 데이터 파일들

2.3 소프트웨어 실행 방법

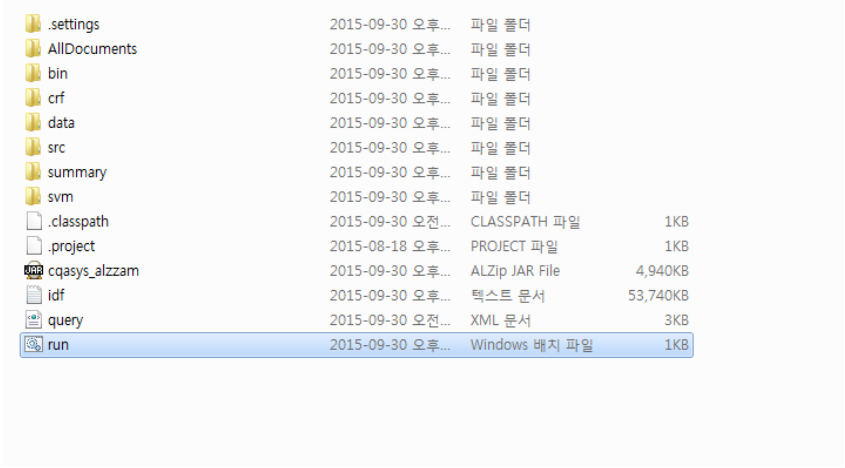
1. cqasys_alzzam.zip 파일의 압축을 해제한다.



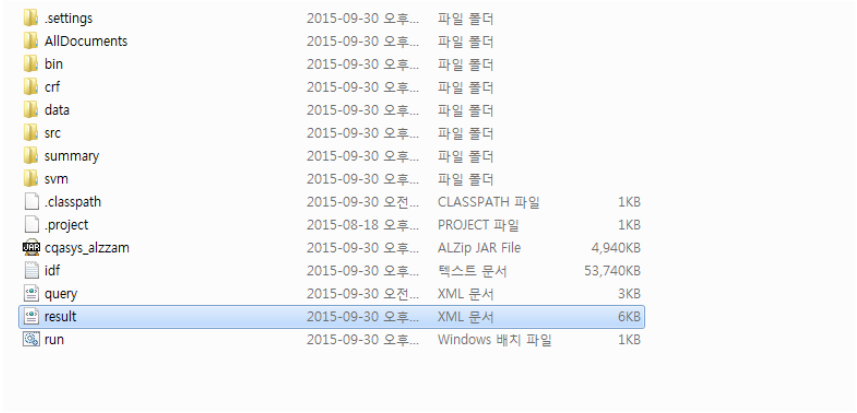
2. cqasys_alzzam/alzzamee 폴더에 들어간다.

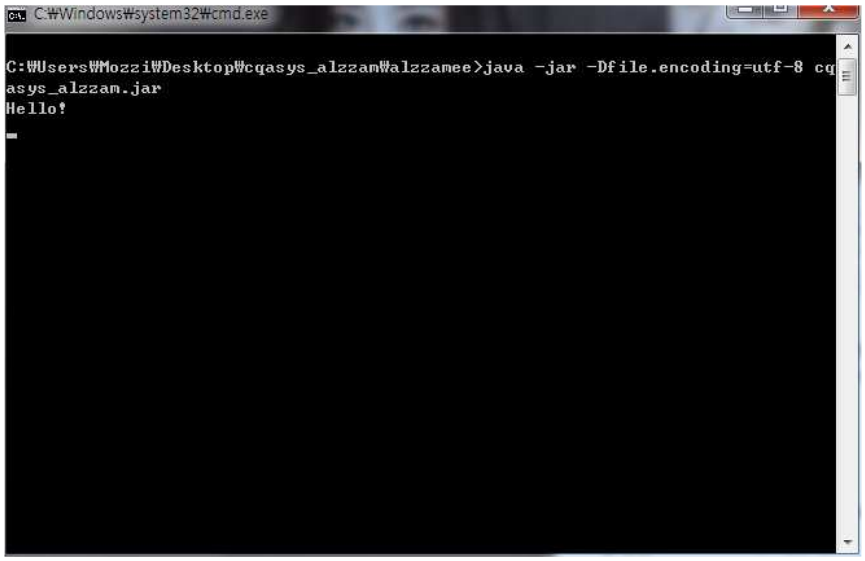


3. run.bat 파일을 실행한다.



4. result.xml 파일이 생성되면 이를 확인한다.





제 3 장 소프트웨어 기능

3.1 프로그램 기능

본 시스템은 사용자 질의를 분석한 내용을 바탕으로 총 9개의 유사도 알고리즘을 통해 국립국어원에 등재된 5,336개의 Q/A쌍 중에서 정답 후보 Q/A쌍 상위 10개를 출력한다.

질의 분석부분은 다음과 같이 두 부분으로 나뉜다.

1. Focus 추출

Focus란 문장에서 따옴표나 특수기호로 구분되어야 하는 사용용례를 의미한다. 이를 추출하기 위해 5,336개의 Q/A 쌍에서 Focus를 태깅한 후 Conditional Random Filed기법을 사용하여 학습 모델을 생성한다. 새로운 질의를 학습된 모델에 적용하여 Focus를 얻는다. 이 Focus들은 추후에 랭킹 모델에서 유사도를 측정하는 데에 자질로 사용된다.

2. 카테고리 분류

국립국어원에 등재된 5,336개의 Q/A쌍의 질의 유형을 총 8개의 카테고리로 분류하였다. 8개의 카테고리는 다음과 같다.

카테고리 번호	카테고리 이름
0	카테고리 없음
1	띄어쓰기
2	발음
3	대용어/순화어
4	줄임말
5	호칭어/지칭어
6	기호
7	외국어/외래어

[표 1] 분류 카테고리 이름

위 8개의 카테고리로 태깅 된 데이터를 Support Vector Machine 기법으로 학습시켰다. 새로운 질의가 들어오면 이 학습된 모델을 통해 카테고리를 부여받는다. 5,336개의 Q/A 쌍 중 질의와 같은 카테고리가 있다면, 해당 질문은 질의와 비슷하다고 할 수 있으므로 추후에 유사도 가중치를 증가 시킨다.

위의 질의 분석에서 분석된 결과를 이용하여 랭킹 모델에서 유사도 알고리즘을 통해 5,336개의 Q/A쌍 중에 정답 후보인 상위 10개의 Q/A쌍을 얻는다. 본 시스템에 사용된 유사도 알고리즘은 총 9개 이며, 이는 실험적으로 정해졌다. 이 9개의 유사도 알고리즘에서 나온 유사도 수치들로 Weighted Sum을 구하여 Weighted Sum수치가 높은 순서대로 상위 10개의 쌍을 결정한다. 이렇게 다양한 알고리즘을 쓰는 이유는,

안정화 효과와 상호 보완 효과 때문이다. 어떤 답이 되는 Q/A쌍이 유사도 계산방법으로 나온 수치들이 전부 상위의 수치가 아니어도, Weighted Sum을 통해 최종적으로 랭킹에서 상위가 될 수 있다는 것이 안정화 효과이다. 또, 특정 유사도 계산 방법에서는 굉장히 낮은 수치를 갖더라도, 다른 유사도 계산 방법에서 매우 높은 수치를 갖는다면 그것이 최종적으로 가장 상위에 랭크 될 수 있다는 것이 상호 보완 효과이다. 이렇게 최종 상위 10개의 답을 구하게 되는 9개의 유사도 알고리즘은 아래의 표에 작성한 내용으로 구현하였다.

	유사도 계산 방법	자질 형태	5,336개의 train 데이터 내 유사도 계산 대상
1	BM25	형태소 분석 Unigram	Answer
2	BM25	음절 Trigram	Answer
3	BM25	음절 Trigram	Question
4	Cosine Similarity	음절 Trigram	Question
5	Cosine Similarity	Focus 음절 Unigram	Question+Answer
6	Cosine Similarity	음절 Bigram	Summary
7	Cosine Similarity	Focus 음절 Bigram	Summary
8	Word Matching	Focus Word	Question+Answer
9	Word Matching	Focus Word	Summary

[표 2] 사용된 유사도 계산 알고리즘

본 시스템은 위의 랭킹 모델을 통해 사용자 질의에 대한 정답 후보 Q/A쌍 상위 10개를 출력한다.

3.2 프로그램 기능 제약

- 데이터 및 내부 코드에 한글 및 여러 나라의 언어가 들어있으므로, UTF-8 환경에서 사용해야 정상적인 동작을 보장함
- 프로그램의 실행을 위해서는 JDK 1.8.0_22 이상이 설치되어 있어야 한다.
- heap space를 최소 1GB 이상 설정하여야 한다.
- 입출력 양식

① 입력 파일 형식

```
<query><qnum>1</qnum>
<text>
10%감소했다를 읽을 때, 퍼센트라고 읽든 프로라고 읽든 상관없나요?
</text>
</query>
<query><qnum>2</qnum>
<text>
아니요를 아뇨로 줄여서 사용해도 괜찮은가요?
</text>
</query>
...
```

[그림 1] 입력 파일 query.xml

본 시스템의 입력 파일 형식은 [그림 1]와 같다. <qnum>에 해당하는 내용이 query의 번호가 되며, 총 20개의 query가 들어온다. <text>내의 내용이 하나의 사용자 질의query를 의미한다.

②출력 파일 형식

```
<query><qnum>1</qnum>
<rank>
5126    477.35672798561
5175    353.25825420771207
5125    291.1276230860242
11      191.70259807924745
2288    187.7015752816263
5124    178.33183938235558
2762    169.74815794433067
14      161.74102110482607
4304    160.21415419361344
1564    155.80620414723583
</rank>
</query>
<query><qnum>2</qnum>
<rank>
3330    331.1610479769186
3325    328.63892913265875
3328    323.1652893956374
3331    314.5947628529111
3329    276.8471988036495
3327    266.07495096550576
3332    264.7833594886736
973     253.5646114989745
3322    235.6110954986118
3334    211.39262305565364
</rank>
</query>
…
```

[그림 2] 출력 파일 result.xml의 형식

주어진 입력 query에 대해서, [그림 2]와 같은 출력 양식을 가진다. 입력 파일에서의 하나의 query는 출력파일에서도 하나의 <query>~</query>에 대응이 된다. 주어진 n번째의 질문 query에 대하여 가장 유사하다고 판단되는 10개의 Q/A결과가 <rank>~ </rank>사이에 작성된다. 첫 번째 열은 상위 10개의 index 번호를 의미하며, 두 번째 열은 각 index 번호의 유사도 score를 의미한다. 그리고 이 10개의 score가 내림차순으로 정렬된다. 즉, 상위에 위치할수록 더 유사한 질문이라고 할 수 있다.

제 4 장 기타

20개 query에 대한 결과는 아래와 같다.

10%감소했다를 읽을 때, 퍼센트라고 읽든 프로라고 읽든 상관없나요?										
5126	5175	5125	11	2288	5124	2762	14	4304	1564	
아니요를 아뇨로 줄여서 사용해도 괜찮은가요?										
3330	**3325**	3328	3331	3329	3327	3332	973	3322	3334	
김치는 gimchi라고 적는 것이 일반적인가요?										
728	**726**	3169	3849	1052	1564	237	3183	2659	1561	
해고당하다에서 당하다의 띄어쓰기가 궁금합니다.										
1001	**2025**	**1196**	4246	5358	1197	3162	1043	1701	2073	
장사가 안된다할 때 안된다의 띄어쓰기는?										
3422	3424	3421	3396	3420	4893	2208	1501	1701	3387	
및이라는 단어의 쓰임에 대해 알려주세요										
3467	**2130**	2690	1524	479	1060	1240	4061	1242	4182	2131,2132
로마자 표기법에서 고유 명사는 무조건 대문자로 시작하는지 아니면 소문자로 적는 경우도 있나요?										
407	1583	406	773	4867	1586	410	727	190	301	
조사 에와 의의 쓰임에 대해서 알려주세요.										
4594	4120	**4587**	3705	4586	4595	481	4593	2117	3701	4588
양복 한 벌이라고 말할 때 한 벌의 띄어쓰기에 대해 궁금합니다.										
3911	5284	5286	5285	4806	2426	5289	5288	5291	1112	
마침표의 의미와 종류는 무엇인가요?										
3891	1668	1667	1666	1670	1669	5029	1665	4702	4388	
선생님은 붙여서 사용하는데 누구누구님은 어떻게 띄어 쓰나요?										
4564	2861	1180	1085	554	2110	1464	3893	515	1545	
친구가 온 상태와 친구가 오는 상태에서 온과 오는의 차이가 무엇인가요?										
3885	3837	3451	175	3892	2030	1054	87	607	4452	
외래어와 외국어의 정확한 차이를 가르쳐 주세요.										
3950	3952	3949	**71**	3881	2311	4977	**1570**	3275	3951	
따옴표의 종류가 두가지가 있는데 각각의 쓰임이 궁금합니다.										
1477	**5029**	1478	4250	4388	1476	1475	2941	162	4387	
몇 일과 며칠의 차이에 대해 자세히 알려주세요.										
1883	1882	**1910**	**1884**	**1911**	1915	**1885**	1886	5142	363	
입니다는 어떻게 발음되는지 알려주세요.										
4304	**4305**	3467	5385	3117	4394	3365	1848	5342	3935	
라면은 외래어인가요?										
836	2668	4366	1570	4977	3881	3321	**3952**	1569	5180	
그것은을 준 대로 적으면 어떻게 되나요?										
4688	582	3572	3571	4162	3573	1747	5003	3565	626	
너도 할 수 있다 혹은 너도 할수있다 중 어떤 것이 알맞나요?										
5333	5334	4318	4689	4317	2347	5332	4323	5337	2885	4127
접미사 -지와 -치의 쓰임에 대해 자세히 설명해주세요										
4743	4520	4523	4524	4526	4521	4728	1230	4532	4525	2838,5051,2802

각 query의 아래에 적혀있는 10개의 숫자가 본 시스템에서 결과로 나온 상위 10개의 Q/A 쌍 번호를 의미한다. 가장 오른쪽의 열에 적힌 숫자는 답으로 나와야 하나, 본 시스템에서 찾지 못한 답을 의미한다. 최종 MAP 계산 결과는 0.7559761904761906 이다.

DNN

by legacy

최경호, 황현선, 오준호, 김건영
강원대학교 컴퓨터과학과

차 례

제 1 장 소프트웨어 소개

1.1 소프트웨어 명칭

DNN
: DNN을 응용하여 BM25와 결합하는게 주된 아이디어이므로 명칭을 DNN으로 하였다.

1.2 소프트웨어 사용 환경

- OS: Window 7/8/10 64bit
- Ram: 8Gb
- 사용 언어: Python2.7, C++
- 실행 환경: command line에서 cqasys_DNN.bat 실행
- 인코딩: UTF-8

1.3 소프트웨어 특징

본 시스템 에서는 cQA 검색시스템을 정보검색 방법으로 해석하여, 형태소 단위 BM-25와 bi-gram 단위 BM-25, 그리고 "Deep learning for answer sentence selection." 에서 제안한 딥 러닝, 세 방법으로 도출된 결과 스코어 병합하여 최종 Score를 구하여, 결과를 도출하는 시스템을 제안한다.

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

- cqasys_DNN.zip 압축파일을 해제한다.

2.2 소프트웨어 파일 구조

2.2.1 주요 파일 설명

① 실행 파일

cqasys_DNN/src/main.py (python main.py로 실행)

② 데이터 파일

cqasys_DNN/data ------ 프로그램이 필요한 QA쌍
cqasys_DNN/src/bin ---- POS_tagger
cqasys_DNN/src/data --- DNN
cqasys_DNN/src/model -- DNN모델과 Word Embedding
cqasys_DNN/src/rsc ---- POS_tagger 리소스

③ 사전 파일

cqasys_DNN/src/model -- Word Embedding
cqasys_DNN/src/rsc ---- pos_tagger가 쓰는 사전

2.2.2 전체 구조

```
cqasys_DNN
|   LICENSE
|   query.xml
|   query4pos.txt
|   README.md
|   result.xml
|
├──data ..... BM25에 쓰이는 QA쌍이 bigram과 pos_tag처리되어 들어있다.
|       corpus.txt
|       corpus_bi.txt
|       corpus_cnvrtd_ansi.txt
|       corpus_tagged.txt
|       corpus_tagged_n-2 (2).txt
|       corpus_tagged_n.txt
|       count.py
|       kor_tagger_x64.old.exe
```

```
|       mk_cqa_pos_data.py
|       nonparsed_corpus.txt
|       utf2ansi.py
|
└──src ..... 프로그램 소스들
    |   bm25.py
    |   bm25.pyc
    |   cqa_dnn.py
    |   cqa_dnn.pyc
    |   invdx.py
    |   invdx.pyc
    |   main.py #실행파일
    |   parse.py
    |   parse.pyc
    |   pos_tagger.py
    |   pos_tagger.pyc
    |   query.py
    |   query.pyc
    |   rank.py
    |   rank.pyc
    |   readme.txt
    |   __init__.py
    |
    ├──bin ..... POS_tagger.exe
    |       kor_tagger_x64.exe
    |       kowiki-20100403-abstract.xml
    |       kowiki_abstract.txt
    |       README.txt
    |       sejong.parse.pos.txt
    |       sejong.parse.raw.txt
    |
    ├──data ..... DNN에 쓰이는 QA쌍
    |       a_data_con.txt
    |       a_list.txt
    |       cqa10_a_data_con.txt
    |       cqa10_q_data_con.txt
    |       cqa11_a_data_con.txt
    |       cqa11_q_data_con.txt
    |       cqa7_new_a_data_con.txt
    |       cqa7_new_q_data_con.txt
    |       cqa8_new3_a_data_con.txt
```

```
|       cqa8_new3_q_data_con.txt
|       q_data_con.txt
|       q_list.txt
|
├──model ..... DNN에 쓰이는 워드임베딩과 모델 정보
|       cqa10_temp_model_65_12.5_42.8.txt
|       cqa10_temp_we_65_12.5_42.8.txt
|       cqa11_temp_model_66_12.5_43.4.txt
|       cqa11_temp_we_66_12.5_43.4.txt
|       cqa5_temp_model_46_16.0.txt
|       cqa5_temp_we_46_16.0.txt
|       cqa6_temp_model_28_51.4.txt
|       cqa6_temp_we_28_51.4.txt
|       cqa7_temp_model_45_13.6_41.6.txt
|       cqa7_temp_we_45_13.6_41.6.txt
|       cqa8_temp_model_28_11.3_41.6.txt
|       cqa8_temp_we_28_11.3_41.6.txt
|       cqa9_temp_model_73_17.5_48.5.txt
|       cqa9_temp_we_73_17.5_48.5.txt
|
└──rsc ..... POS_tagger가 쓰는 리소스들
        bigram.korean.txt
        cluster.kor_phrase.txt
        cluster.srl.txt
        cluster.srl2.txt
        cluster_dic.txt
        cluster_dic.txt.old
        dic.txt
        kor_ner_model.bin
        kor_ner_model.bin.plo.old
        kor_tagger_model.bin
        morph_rule.txt
        morph_rule_lp.txt
        ne_dic.txt
        ne_dic4pos.enc.txt
        ne_dic4pos.txt
        srl_arg_model.bin
        srl_pred_model.bin
        tparser_model.bin
        V_freq.txt
        V_surface_freq.txt
```

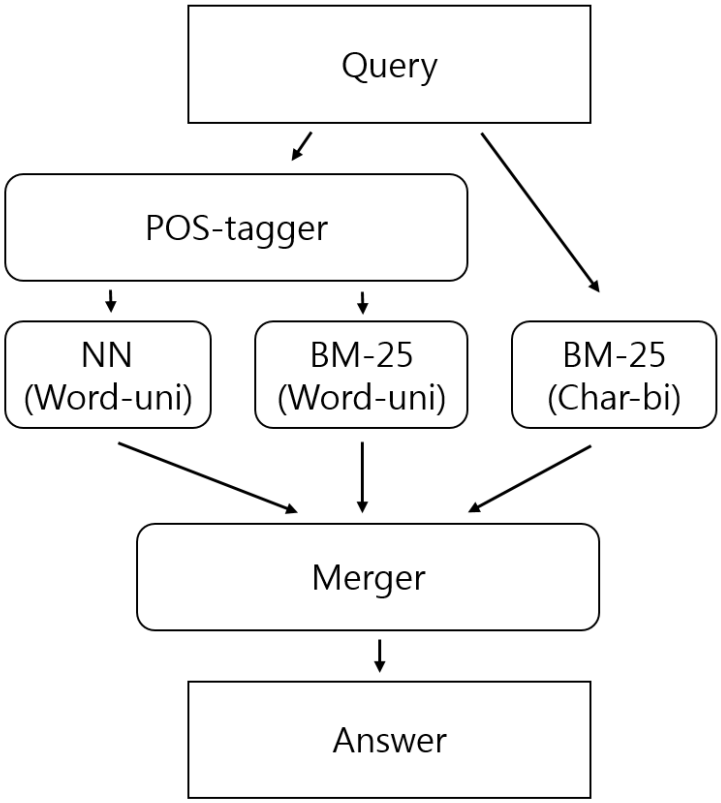
2.3 소프트웨어 실행 방법

1. cqasys_DNN 폴더에 query.xml을 복사하여 넣는다.
2. command line 에서 cqasys_DNN 폴더에 있는 cqasys_DNN.bat를 실행한다.
3. intel i7 4770 환경에서 약 1분 지연 후 cqasys_DNN 폴더에 결과 파일인 result.xml 가 생성된다..

제 3 장 소프트웨어 기능

3.1 프로그램 기능

본 시스템은 쿼리를 받아 형태소 단위로 연산한 BM25와 음절단위 Bigram을 연산한 BM25, 그리고 딥러닝을 사용한 모듈의 스코어 값을 Merger에서 통합하여 결과를 구한다.



3.1.1 BM25

사용한 BM25 수식은 다음과 같다.

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f(q_i,D) \cdot (k_1+1)}{f(q_i,D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})}$$

위 수식에서 D는 문서를 뜻하며 Q는 쿼리를 뜻한다. IDF는 inverse Document Frequency를 뜻하고, avgdl은 문서들의 평균 단어개수를 뜻하며, |D|는 해당문서의 단어 개수를 뜻한다. k_1과 b는 수작업으로 튜닝하며 최적의 값을 구하였다. 시스템에서 사용한 BM25 모듈들에서는 문서의 제목과 내용을 별도의 구분 없이 하나로 합쳐 문서로 사용하였다.

3.1.2 Word Embedding

Word Embedding이란 단어를 Language modeling과 자질학습을 통해 작은 차원의 실수 Vector 형태로 나타내는 방법을 뜻한다.

3.1.3 딥 러닝

“Deep learning for answer sentence selection”에서 제안한 모델을 한국어 및 한국어 문법에 대한 사용자 간 질의-응답 문서에 적합하게 변형하여 사용하였다. 해당 논문에서는 수식 (1)처럼 문장단위로 Embedding vector를 구했으나, 본 시스템은 수식 (2)와 같이 문서 단위로 Embedding vector를 구했다. 수식에서 s는 문장을 뜻하며 s_i는 문장 s에서의 i번째 단어의 Word embedding을 뜻한다. d는 문서를 뜻하며 d_i는 문서 d에서 i번째 Word embedding을 뜻한다.

$$s = \frac{1}{|s|} \sum_{i=1}^{|s|} s_i \quad (1)$$

$$d = \frac{1}{|d|} \sum_{i=1}^{|d|} d_i \quad (2)$$

사전 학습된 Word embedding과 수식 (2)를 이용하여 쿼리와 문서의 Embedding vector를 구한 후 수식 (3)을 통해 쿼리-문서 쌍의 스코어를 구한다.

$$S(Q,D) = \sigma(Q^T \cdot M \cdot D + b) \quad (3)$$

수식 (3)에서 D는 문서의 Embedding vector를 뜻하고 Q는 쿼리의 Embedding vector를 뜻한다. M은 학습을 통해 기록되는 Weight Matrix이고, b는 bios이며 σ는 sigmoid 함수이다. 본 시스템은 Neural Network Language Modelling을 사용하여 학습한 50차원의 Word embedding을 사용하였다.

3.1.4 Merger

Merger는 앞서 설명한 세 가지 방법으로 모든 쿼리-문서 쌍에 대한 스코어를 구하고, 그 값들을 통합하여 최종 스코어를 구한다. 그 방법은 아래 수식과 같다.

$$S_{merged}(Q,D) = a \times S_{bi}(Q,D) + b \times S_{pos}(Q,D) + c \times S_{dnn}(Q,D)$$

수식에서 $S_{bi}(Q,D)$는 음절단위 bigram을 이용한 BM25의 스코어 이고, $S_{pos}(Q,D)$는 형태소 단위 BM25의 스코어, $S_{dnn}(Q,D)$ 은 딥러닝을 사용한 모듈의 스코어를 뜻한다. a, b, c 는 수작업으로 조정된 각 스토어에대한 가중치로 각각 7, 3, 1 일 때 top10 정확도 80%로 최고성능을 보였다.

3.2 프로그램 기능 제약

- 사용 언어: Python2.7, C++
- 실행 환경: command line에서 cqasys_DNN.bat 실행
- 텍스트 인코딩: UTF-8

국어정보 키워드 추출방법을 이용한 질의응답 시스템

(Korean information extraction using the keyword to a question and answer system)

전석종, 이수인, 이현아
금오공과대학교 컴퓨터소프트웨어공학과
자연언어처리 연구실

Copyright© 2015

차 례

제 1 장 소프트웨어 소개

1.1 소프트웨어 명칭

국어정보 키워드 추출방법을 이용한 질의응답 시스템
: 국립국어원의 온라인가나다 서비스는 한국어 어문 규범, 어법, 표준국어대사전 내용 등에 대하여 문의하는 인터넷 서비스이다. 이 서비스는 2000년 8월 경 시작하여, 현재까지 약 14만 개의 한국어 관련 지식정보 데이터를 사용자에게 제공한다. 서비스는 사용자가 게시판에 질문을 올리면 전문성을 가진 관리자가 답변을 등록하는 방식으로 운영되고 있어 한국어에 대한 정확한 정보를 제공한다. 이와 같이 방대한 전문 데이터에 대하여 편리한 검색 시스템이 제공된다면, 사용자는 관리자의 답변 작성을 기다리지 않고 즉시 정보를 얻을 수 있고, 관리자는 유사한 질문들에 대해 동일한 답변을 반복적으로 작성하지 않게 되어 시스템의 효율성을 높일 수 있다.

1.2 소프트웨어 사용 환경

- Microsoft Windows 7 64bit 이상
- 실시간 가용 RAM 8GB 이상 (heap space 3GB 이상)
- java jdk 1.6 이상
- 인코딩 UTF-8

1.3 소프트웨어 특징

국어정보 키워드 추출방법을 이용하여 사용자의 질문과 유사도 계산을 하여 가장 유사하다고 판별되는 Q/A 쌍들을 추출하여 온라인 가나다의 시스템 효율성을 높인다.

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

1. cqasys_suseokTeam.zip 압축 해제
2. 압축해제된 파일 내의 QnAAnalazy 프로젝트를 이클립스로 import
3. 압출해제된 jar파일 디렉토리 내 jar 파일들을 이클립스로 import한 QnAAnalazy 프로젝트로 [project] - [propertise]-[java build path]-[libraries]의 [add External JARS]로 import

2.2 소프트웨어 파일 구조

2.2.1 주요 파일 설명

실행 및 주요 데이터에 대한 설명(간단히 어떤 파일들이 실행파일이고, 어떤 파일이 어떤 데이터를 저장하고 있는지 등을 설명(파일 또는 폴더단위))

① 실행 파일

QnAAnalazy 프로젝트 (jar 파일을 이용하여 실행)

② 데이터 파일

josaADF.dat	데이터 파일
josaATF.dat	데이터 파일
josaQDF.dat	데이터 파일
josaQTF.dat	데이터 파일
UgADF.dat	데이터 파일
UgATF.dat	데이터 파일
UgDF.dat	데이터 파일
UgTF.dat	데이터 파일
query.xml	질문 xml 파일

2.2.2 전체 구조

QnAAnalazy/

src/	소스파일폴더
josaADF.dat	데이터 파일
josaATF.dat	데이터 파일
josaQDF.dat	데이터 파일
josaQTF.dat	데이터 파일
UgADF.dat	데이터 파일
UgATF.dat	데이터 파일

UgDF.dat	데이터 파일
UgTF.dat	데이터 파일
query.xml	질문 xml 파일

2.3 소프트웨어 실행 방법

1. 이클립스로 프로젝트와 외부 jar파일들을 import 시킨 것을 확인

2. 이클립스 내의 run-configurations의 arguments 탭의 vm arguments의에 다음 명령을 시행
   -Xms2048m
   -Xmx4096m

3. run 수행

제 3 장 프로그램 기능 제약

1. 프로젝트 파일을 이클립스를 구동시켜 실행

2. 추가적인 외부 jar파일을 import를 시켜서 실행

3. heap space를 최소 2048M 최대 4096M로 지정

4. 지정 파일 형식인 query.xml을 토대로 하려하였으나 root element 가 없어 data를 root로 지정하여 파일을 구성

누르미 – 터치 동작 기반 키보드

by 새벽네시

박형순, 김민호, 박소영, 김도경, 김두환, 최윤승

국민대학교 컴퓨터공학과

Copyright© 2015

차 례

제 1 장 소프트웨어 소개

1.1 누르미 키보드는

현재 지구는 스마트 폰 없이 살기는 어려운 '스마트폰의 행성'이 되었습니다. 이와 관련하여 영국 주간지 이코노미스트는 "세상이 스마트폰 없이 살기 어려운 '포노 사피엔스(Phono Sapiens)' 시대로 변했다.'라고 표현했습니다. 실제로 스마트폰은 역사상 가장 빨리 팔린 기계에 속합니다. 또한 현재 전 세계 인구의 절반이 스마트폰을 가지고 있는 것으로 나타났습니다. 이와 같은 자료를 통해서 2020년에는 세계 인구의 89%가 스마트 기기를 소유할 것으로 측정되며 스마트폰이 현재보다 더 우리의 생활에 깊숙이 침투할 것으로 예상되고 있습니다.

세계보건기구(WHO)의 통계에 따르면 세계적으로 약 10억 명의 장애인이 존재합니다. 과거 장애인들은 비장애인들에 비해 정보에 접근하기가 어렵기 때문에 정보 격차가 심할 수밖에 없었습니다. 그러나 스마트 문명 시대의 도래를 통해 장애인과 비장애인의 구분이 사라질 수 있게 되어 정보 격차를 해소할 수 있게 되었습니다. 문명의 발달로 인해 장애인들 또한 자신이 지닌 장애 요소를 어느 정도 보완할 수 있게 된 것입니다. 다시 말해 장애인들도 스마트 기기를 활용하여 세상과 자유롭게 소통할 수 있으며 다양한 정보를 손쉽게 얻을 수 있게 되어 비장애인들과의 정보 격차를 줄일 수 있게 되었습니다.

하지만 대부분의 스마트 기기에서 비장애인들을 대상으로 한 사용 편리성이나 보안 기능을 중점으로 개발하고 있어 장애인이 이용하기에는 현실적으로 불가능하다고 할 수 있습니다. 특히 시각장애인들은 터치스크린 기반의 스마트 기기를 자유롭게 이용하기 어려우며, 이용할 수 있더라도 매우 제한된 범위 내에서 매일 고군분투하여 사용하고 있습니다. 스마트 기기가 앞서 말한 것과 반대로 정보 격차를 심화시키는 요소로 작용하고 있게 될 것입니다.

약 10억 명의 장애인들 중에서도 2억 8500만 명 정도는 시각장애인으로 다양한 장애 요소를 고려한다면 시각 장애가 꽤 큰 비율을 차지하고 있음을 알 수 있습니다. 시각장애인의 경우 위와 같은 문제를 해결하기 위해 많은 기업에서 '음성인식 및 명령 기능'을 이용하여 시각장애인용 스마트폰 및 애플리케이션을 출시했습니다. 그러나 실제 이러한 입력 방식은 시각장애인들이 공공장소에서 사용하기에는 주변에 방해가 될 수 있기 때문에 매우 부담스러운 기능입니다. 그렇다고 해서 값비싼 블루투스 점자 키보드나 제법 무게가 나가는 리더기를 번거롭게 매번 들고 다닐 수도 없는 상황이 시각장애인들의 현실입니다.

본 어플리케이션은 스마트 문명 시대의 도래로 인해 다양하고 넓은 사용자 층이 스마트 기기를 이용하게 되었지만 장애인과 같은 일부 특정 사용자들은 편리성이 많이 결여되어 스마트 기기를 사용하기 불편하다는 것을 인지하고 만든 어플리케이션입니다. 시각장애인이 스마트 기기를 사용하기 어려운 가장 근본적인 문제이자 위에서 언급한 문제들을 해결하고자 시각장애인을 위한 새로운 텍스트 입력 방식을 고안하였습니다. 기존에 존재하지 않는 한글 입력 방식을 적용한 새로운 한글 특화 멀티 터치 키보드를 고안 및 제공하는 것을 목표로 했습니다.

화면 터치 동작을 기반으로 글자를 입력한다는 특징에서 이 애플리케이션의 이름은

누르미가 되었습니다.

1.2 소프트웨어 사용 환경

- 지원 OS : 3.0 이상의 Android (Honeycomb, API 11)
- 사용 메모리 : 약 130MB
- 사용 언어 : 한국어
- 실행 환경 : 3.0 이상의 Android (Honeycomb, API 11)

1.3 소프트웨어 특징

1.3.1 특징과 장점

시각 장애인들은 스마트폰 화면에 있는 자판을 보는 것에 어려움이 있습니다. 이 소프트웨어는 기존의 키보드처럼 버튼이라는 시각적인 정보에 의존하지 않고도 글자를 입력할 수 있다는 것이 특징입니다. 화면에 다섯 손가락을 터치하여 기준 위치를 정하므로 스마트폰과 태블릿pc 어느 쪽에서도 사용자의 손 크기와 모양에 맞는 키보드가 활성화 됩니다, 또한, 버튼을 보고 누르는 방식이 아니라 사용자가 정한 기준점에서 스와이프나 탭 등의 터치 동작으로 글자를 입력하는 방식으로 시각이 제한된 상황에서도 입력이 가능합니다.

1.3.2 기대효과

◆ 시각 장애인들의 보다 편리한 스마트기기 사용

스마트 폰은 기본적으로 비장애인들을 위해 설계되고 개발되었기 때문에 장애인들이 사용하기에는 많은 어려움이 존재합니다. 특히 시각 장애인들처럼 시각이 제한되면 터치스크린 내부에서의 버튼 경계를 인식할 수가 없어 스마트 기기를 사용하는 데에 있어 큰 문제가 발생합니다. '누르미'는 시각을 배제한 상태에서도 빠르고 정확한 텍스트를 입력할 수 있는 방식으로 고안되었기 때문에 시각 장애인들이 스마트 폰을 보다 편리하게 사용할 수 있을 것입니다.

◆ 무겁고 비싼 시각 장애인용 키보드를 대체

시각 장애인용 키보드는 무겁고 가격이 비싸다는 단점이 있습니다. 하지만 업무를 처리해야 하는 시각 장애인들에게는 필수적으로 필요로 되는 하드웨어입니다. 이 하드웨어를 대체할 수 있도록 '누르미'가 개발되었습니다. '누르미' 키보드 애플리케이션을 통해 태블릿과 같은 스마트 기기를 이용하여 대체할 수 있습니다. 스마트 기기를 이미 가지고 있다면 별도의 비용이 들지 않고 무게도 가볍기 때문에 시각 장애인들이 스마트 기기를 통해 본 애플리케이션을 이용할 경우 이득을 줄 수 있다고 판단합니다.

◆ 더 이상의 버튼 위치 암기는 없다

시각 장애인들도 현재 스마트기기에서 쿼티, 나랏글, 천지인 등 일반인이 사용하는

키보드를 사용하고 있다고 합니다. 이를 사용하기 위해서 사용법을 교육받고 버튼의 위치를 암기하고 있습니다. 하지만 기기를 여럿 사용하거나 혹은 변경하게 될 경우 화면 크기가 달라져서 버튼의 위치를 다시 암기해야하는 상황이 발생합니다. 본 애플리케이션은 어떤 기기에서든 사용자의 손으로 기준점을 잡고 같은 동작으로 입력하기 때문에 그러한 걱정 없이 사용할 수 있습니다.

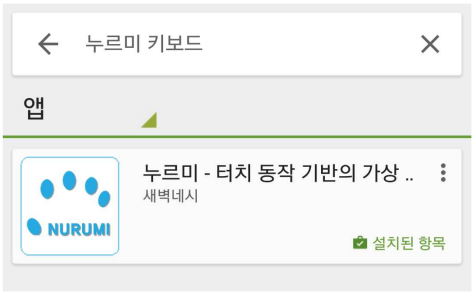
◆ 특수한 상황에서의 사용자

스마트 폰을 사용할 때에 시야가 제한되는 경우가 발생할 수 있습니다. 예를 들어 기자와 같이 키보드가 아닌 시야를 다른 곳에 두며 텍스트를 입력해야 하는 경우, 자동차 운전 중인 경우와 텍스트를 입력해야 하나 양 손으로 터치하기가 힘든 상황인 경우 등이 있습니다. 운전 중인 경우에는 키보드를 사용하기 위하여 시선을 돌리지 않아도 키보드를 사용할 수 있으므로 위험에 노출될 확률을 줄일 수 있습니다. 이러한 특성으로 단지 시각장애인 뿐 아니라 일반인도 활용할 수 있다고 판단합니다.

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

1. 안드로이드 기기의 Google Play를 실행합니다.
2. 누르미 키보드 검색



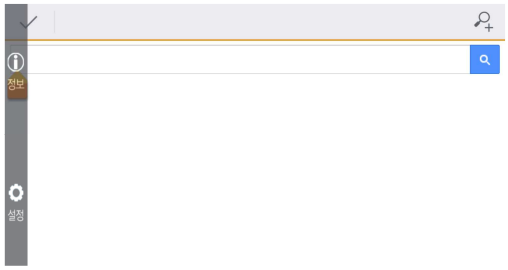
3. 애플리케이션을 설치합니다.

2.2 소프트웨어 파일 구조

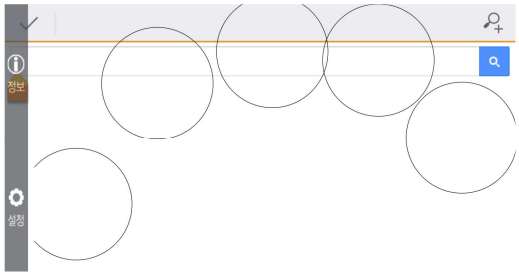
- NurumiKeyboard.apk 안드로이드 응용 프로그램 패키지 파일

2.3 소프트웨어 실행 방법

1. 안드로이드 기기의 '설정'을 실행합니다.
2. '언어 및 입력'을 들어갑니다.
3. 키보드 및 입력 방식을 누르미 키보드로 설정합니다.
4. 텍스트 박스에 커서를 두면 누르미 키보드가 활성화됩니다.



5. 키보드가 활성화 된 상태에서 화면에 다섯 손가락으로 터치를 하면 다섯 개의 입력 기준지점이 생깁니다.



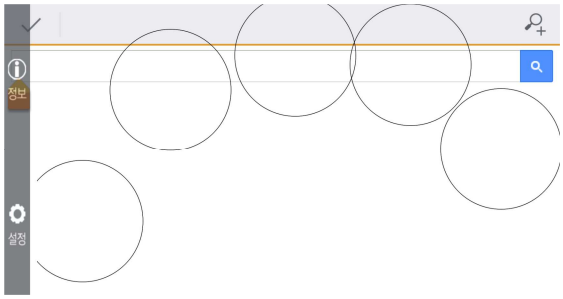
제 3 장 소프트웨어 기능

3.1 프로그램 기능

프로그램 기능은 크게 3가지로 구분할 수 있습니다.

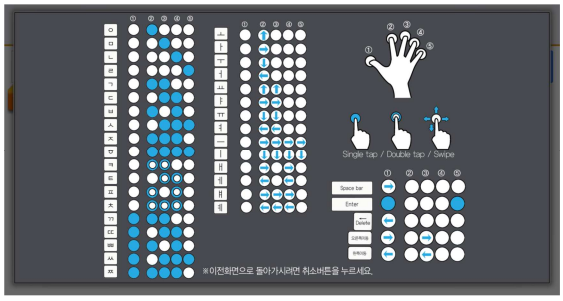
(입력 기능)

누르미는 터치 동작 기반 키보드로 기본적으로 입력 기능을 가지고 있습니다. 빈 화면에 다섯 손가락을 터치하면 다섯 개의 원이 생성 되며, 그 원을 기준으로 각각의 원을 탭하거나 스와이프 하는 형식으로 글자를 입력할 수 있습니다.

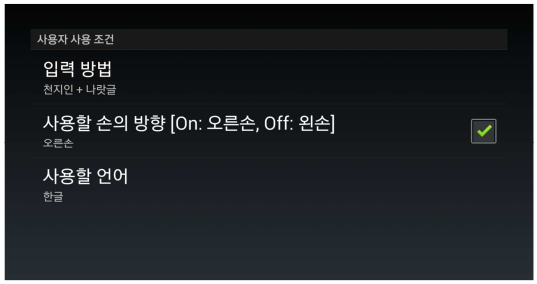


(정보 버튼 기능)

왼쪽의 회색 막대에서 정보를 누르시면 현재 어떠한 문자나 기능이 어떠한 동작에 매핑 되어있는지 확인 할 수 있습니다. 현 버전에서는 입력 방식과 사용할 손의 방향에 따라 나오는 정보 이미지가 나누어져 있습니다.



설정 버튼 기능) 왼쪽의 회색막대에서 설정을 누르시면 키보드 입력 설정에 관한 내용을 바꿀 수 있습니다. 또한 사용할 언어는 한글/영어/특수문자 세가지 이며, 설정 화면에서 뿐 아니라 다섯 손가락을 모두 탭 하는 동작으로도 전환이 가능합니다.



3.2 프로그램 기능 제약

해당 프로그램은 안드로이드를 기반으로 제작 했습니다. Apple 사의 IOS와 다르게 안드로이드는 기기의 제조사가 매우 다양합니다. 또한 각각의 제조사들은 최신 안드로이드 OS로 업데이트를 지원해주지 않는 경우가 대부분입니다. 따라서 사용되는 API의 범위가 오래된 것부터 최신 것까지 매우 넓습니다.

멀티 터치 모션 기반 한글 입력기를 개발하려면 모션에 적합한 한글 오토마타가 필요합니다. 기존에 제공되는 한글 오토마타를 이용하면 직관성 결여 및 타수의 증가와 같은 문제가 발생합니다. 새롭게 개발하는 한글 오토마타로 인해 본 애플리케이션을 사용하는 사용자들은 새로운 입력 체계를 배워야 한다는 문제가 발생합니다.

제 4 장 기타

◇ 차기 버전 업데이트 목록.

1. 소스 코드 개편 및 새로운 기능 추가

각 손가락의 값을 비트 연산으로 처리하여 데이터 베이스 안의 오토마타 정보와 연동하여 해당되는 자음이나 모음을 출력해주는 방식으로 수정하겠습니다. 비트 연산 코드로 변경 시 현재 프로그램 보다 입력 값 확인이 적어 성능도 향상 될 것이고 현재 소스코드로 되어 있는 오토마타도 데이터베이스로 변환 되므로 소스코드의 유지 보수 및 유저들이 직접 원하는 키보드 입력 형식으로 오토마타 제작도 가능 할 것입니다.

2. 새로운 기능에 대한 UI 제작과 UI 개편

입력 방식을 새롭게 작성 할 수 있도록 기능을 추가하고 그 기능에 씌워줄 UI를 제작할 예정입니다. 그에 따라서 보여줄 수 있는 UI를 개편할 예정입니다.

한국어 문장 분할 및 구 묶음 추출 도구

(KoSeCT: Korean Sentence Segmentation and Chunking Tool)

by Semantic Web Research Center

남상하, 원유성, 우종성, 함영균
카이스트 시맨틱 웹 첨단 연구 센터

Copyright© 2015

차 례

제 1 장 소프트웨어 소개

1.1 한국어 문장 분할 및 구 묶음 추출 도구 (KoSeCT)

최근 시맨틱 웹이 발달함에 따라 전 세계적으로 자연어 문장에서부터 유용한 정보 혹은 지식을 추출하기 위한 정보 추출(information extraction)에 관한 연구가 활발히 진행 중이다. 이때 보다 정확한 정보 추출을 위해서는 신뢰성 있는 자연 언어 처리 시스템들이 갖추어져 있어야 한다. 약 20년 전부터 자연 언어 처리 분야에서 다양한 자연 언어 처리 분석 도구들이 오픈 소스 라이브러리로 배포되어 왔다. 그러나 이들 대부분이 형태소 분석에 관한 도구들이고 구 묶음(chunking) 및 문장 분할(sentence segmentation)에 대해서는 신뢰성을 갖춘 공개된 도구가 많지 않다. 게다가 최근 공개된 국어 정보 처리 도구는 형태소 분석과 구문 분석에 중점을 두고 있어서 구 묶음 추출이나 문장 분할 같은 기능을 제공하고 있지 않고, 이로 인해 높은 한국어 정보 추출 결과를 기대하기 어렵다.

- **분장 분할의 필요성**: 구문 분석 결과의 성능은 문장의 길이와 문장 복잡도와 관련이 많다. 따라서 만약 문장 분할이 선행된다면, 보다 신뢰성 있는 구문 분석 결과를 도출할 수 있다. 이러한 신뢰성 있는 구문 분석 결과는 구 묶음 추출 및 정보 추출에서 매우 유용하게 사용될 수 있다.

- **구 묶음 추출의 필요성**: 구 묶음 추출은 K-NN을 이용한 방법과 CRF(Conditional Random Field)를 이용한 방법 등이 있다. 그러나 이 방법들은 학습을 해야 하기 때문에 학습 데이터를 만들기 위한 노력이 별도로 필요할 뿐만 아니라 학습 데이터에 사용되는 예제(instance)를 잘 선정하는 것도 하나의 문제가 된다. 따라서 본 팀에서는 의존 구조 분석 결과를 활용한 구 묶음 추출 모듈을 개발하여 학습 데이터에 의존적이지 않도록 한다. 그에 따라 구 묶음 추출 결과가 정보 추출에서 매우 유용하게 사용될 수 있도록 한다.

1.2 소프트웨어 사용 환경

- 지원 OS: Java가 설치된 모든 OS
- 권장 메모리: 10MB 이하
- 사용 언어: JAVA
- 실행 환경: Eclipse 4.4.2, JDK 1.8
- 텍스트 인코딩: UTF-8

1.3 소프트웨어 특징

본 소프트웨어는 한국어 문장 분할 및 구 묶음 추출 기능을 가지고 있다. 먼저 문장 분할 모듈은 단일 문장을 입력으로 받아서 해당 문장의 성질을 분석하여 다양한 연결 어미를 기준으로 문장을 나누어 복수개의 문장을 출력하는 모듈이다. 예를 들어, '한 가지 이상의 일을 나열한 것' 이나 '서로 상반됨'을 기술한 어절의 길이가 다소 긴 문장들에 대해서 간소화(simplification) 과정을 수행하여, 이후의 정보 추출 과정이나 기타 국어 정보 처리 기능들의 성능을 높인다. 이때 문장이 분할된 경우에는 복수개의 문장에 적절한 주어를 설정하는 기능도 포함되어 있다.

두 번째로 구 묶음 추출 모듈은 단일 문장을 입력으로 받아서 해당 문장에서 복수개의 명사구 묶음과 동사구 묶음을 출력하는 모듈이다. 기본적으로 구문 분석 결과를 사용하여 별다른 기계학습 없이 구 묶음 추출을 수행하는데, 현재 공개된 언어 처리 도구의 구문 분석 결과는 최적의 구 묶음 추출을 수행하기에는 적합하지 않다. 따라서 형태서 분석 결과와 구문 분석 결과를 함께 활용하여 해당 모듈이 자동으로 구문 분석 결과를 재작성 한 후 본 팀에서 설계한 알고리즘을 바탕으로 구 묶음 추출을 수행한다.

본 소프트웨어는 순수 자바로 작성되었기 때문에 자바가 설치된 모든 디바이스에서 사용 가능하고, 다른 정보 추출 및 자연어 처리 모듈과의 연동을 위해 낮은 메모리 점유율 및 빠른 실행 속도를 고려하여 설계 및 구현하였다.

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

- 설치 불필요, JAR 파일 실행.
- 실행 방법은 2.3절에 기술

2.2 소프트웨어 파일 구조

2.2.1 주요 파일 설명

① 실행 파일

KoSeCT.java ------- 프로젝트 전체 모듈
Chunker.java ------- 구 묶음 추출 모듈
StmtSegmter.java --- 문장 분할 모듈

2.2.2 전체 구조

KoSeCT/
- src/ ..... 소스파일폴더
  - KoSeCT.java ..... 전체 모듈 실행 파일
  - data/ ..... 내부 데이터 타입
    - Chunk.java ..... Chunk 저장 타입
    - Sentence.java ..... Sentence 저장 타입
  - modules/ ..... 핵심 모듈
    - Chunker.java ..... 구 묶음 추출 모듈 실행 파일
    - StmtSegmter.java .... 문장 분할 모듈 실행 파일
  - submodules/ ..... 핵심 모듈을 돕기 위한 세부 모듈
    - CoreExtractor.java .. 나머지 모든 세부 모듈 실행 시 메모리 감소를 위한 RAW 데이터 축소 모듈 (구 묶음 추출을 위함)
    - DPWDChanger.java .. 구문 분석 및 형태소 분석 결과를 본 팀에서 정한 기준에 맞게 새로운 분석 결과로 변형하는 모듈 (구 묶음 추출을 위함)
  - tools/ ..... 전체 프로젝트를 위한 기타 도구들
    - Globals.java ..... 변수 저장을 위한 클래스
    - KoreanAnalyzer.java .. 한국어 언어분석 도구 서비스 연결 모듈
    - StringEdit.java ..... 스트링 편집기
- lib/ ..... 라이브러리폴더
  - *.jar ..... REST API 및 JSON API

2.3 소프트웨어 실행 방법

1. 문장 분할 모듈
   Window: java -jar gnrsys_kosect_1.x.jar modules/StmtSegmter.java
   Linux: java -cp gnrsys_kosect_1.x.jar modules.StmtSegmter

2. 구 묶음 추출 모듈
   Window: java -jar gnrsys_kosect_1.x.jar modules/Chunker.java
   Linux: java -cp gnrsys_kosect_1.x.jar modules.Chunker

3. KoSeCT
   WIndow: java -jar gnrsys_kosect_1.x.jar
   Linux: java -jar gnrsys_kosect_1.x.jar

※ 1.x는 1.8 또는 1.6

제 3 장 소프트웨어 기능

3.1 프로그램 기능

3.1.1 문장 분할 모듈

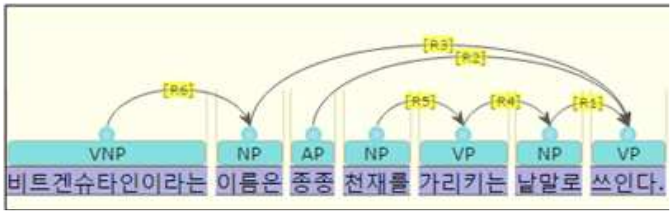
하나의 문장을 연결어미를 기준으로 복수개의 문장으로 나눌 수 있다면, 보다 정확한 정보 추출이 가능하다. 그러나 한국어에는 상당히 많은 연결어미가 존재하고 그 성질이 다양하기 때문에, 어떠한 기준으로 문장을 나눌 것인지 선정하는 것이 중요하다. 본 팀에서는 “한국어 교원을 위한 한국어의 이해, 충남대학교” 도서를 참고하여 문장 분할을 위한 다음과 같은 4가지 성질을 선정하였고, 이를 바탕으로 문장 분할을 수행한다.

성질	예시
한 가지 이상의 일을 나열하는 것	~고, ~(으)며, ...
한 가지 이상의 일이 동시에 일어남을 보이는 것	~(으)면서, ...
두 가지 일이 거의 동시에 잇달아 일어남을 보이는 것	~자
서로 상반됨을 보이는 것	~(으)나, ~지마는, ~라도 ...

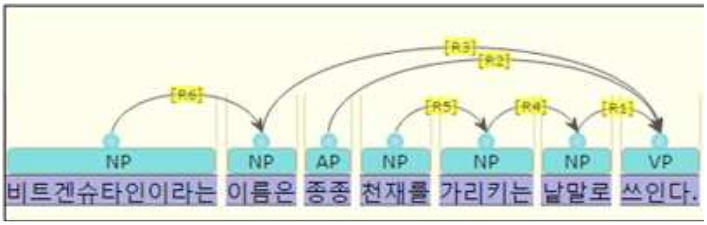
위 예시에서 보는 바와 같이, ‘~고’, ‘~자’ 등의 연결 어미들은 겉만 보았을 때 연결 어미로 오해할 수 있는 경우가 많다. 예를 들어, ‘차고’, ‘줄자’ 등의 단어들은 ‘~고’, ‘~자’의 형태를 보이고 있으나 모두 명사들이다. 따라서 본 모듈은 POS(part-of-speech) 태그 정보를 활용하여 형태소 분석 결과가 “연결어미” 즉 “EC”이면서 위 예시에 해당하는 단어들만 문장 분할 후보로 간주한다. 그 다음, 문장 분할 후보들 중 휴리스틱한 조건에 만족하는 후보들만 문장 분할 기준점으로 선정된다. 그 조건은 “문장 분할을 하였을 때 각 문장이 최소 3어절 이상으로 구성될 것”이다. 이런 조건을 둔 이유는 너무 짧은 단위로 문장이 분할 될 경우 오히려 정보 추출에 손실을 가져다 줄 수 있기 때문에 주어, 동사, 목적어의 최소 단위인 3어절로 제한하였다.

3.1.2 구 묶음 추출 모듈

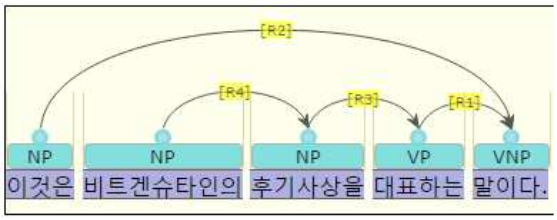
본 모듈은 한국어 언어 분석 결과 중 의존 구조 분석 결과를 이용하여 구 묶음 추출을 수행한다. 이때, 구 묶음을 추출하기 위해 의존 구조 분석 결과를 약간 변경할 필요가 있다. 예를 들어, “비트겐슈타인이라는 이름은 종종 천재를 가리키는 낱말로 쓰인다.” 라는 문장의 의존 구조 분석 결과를 textAE 도구를 사용하여 살펴보면 아래 그림과 같다.



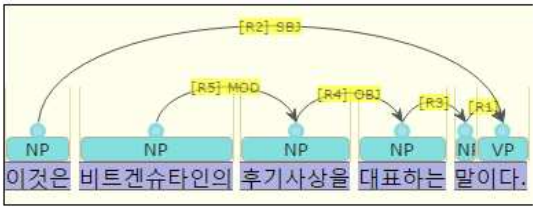
이때 “비트겐슈타인이라는” 어절은 동사와 명사의 결합으로 이루어진 VNP로 라벨이 되어 있고, “가리키는”이라는 어절은 명사를 수식하는 동사표현으로 라벨이 되어 있다. 먼저 “비트겐슈타인이라는” 어절은 VNP보다 “이름은”을 수식하는 NP로 보는 것이 구 묶음 추출에 있어서 보다 명확하고 간결하다. 그리고 “가리키는”이라는 어절은 “낱말”을 수식하는 어절로써 이 역시 NP로 보는 것이 보다 명확하다. 즉 아래와 같은 의존 구조 분석 결과로 수정해주는 작업을 먼저 수행한다.



이처럼 VNP와 VP들 중 NP로 바꾸어주어야 하는 어절의 조건은 다음과 같다. “*형태소 분석 결과, ETM을 포함하고 있는 VNP혹은 VP어절*”. 또 다른 예제로, “이것은 비트겐슈타인의 후기사상을 대표하는 말이다.”라는 문장을 보면 다음과 같다.



이 예시에서는 “대표하는” 이라는 어절, 그리고 “말이다.” 라는 어절에서 수정이 필요한 경우이다. “대표하는”은 위에서 설명한 조건에 해당하므로 NP로 수정하고, “말이다”라는 어절은 문장의 맨 마지막 부분에 위치한 명사와 동사의 결합 어절이다. 이러한 구조는 구 묶음 추출을 수행하기에는 적합하지 않기 때문에 아래와 같이 변경하는 작업을 선행한다.



VNP에 대한 제 2 조건은 “*문장의 맨 마지막에 위치하였을 경우, NP와 VP로 각각 나누어서 의존 분석 결과를 재구성한다.*”이다. 이처럼 의존 구조 분석 결과를 수정하게 되면, 구 묶음 추출 기준을 단순화할 수 있어서 알고리즘이 간단해지고 보다 빠른 동작이 가능하다. 구 묶음 추출 기준은 다음과 같다.

- 동사구 묶음은 모든 VP들
- 명사구 묶음은 VP로 Head를 갖는 최장의 NP Sequence들
- 기타 AP, DP 등은 Head를 가지는 VP나 NP에 부착

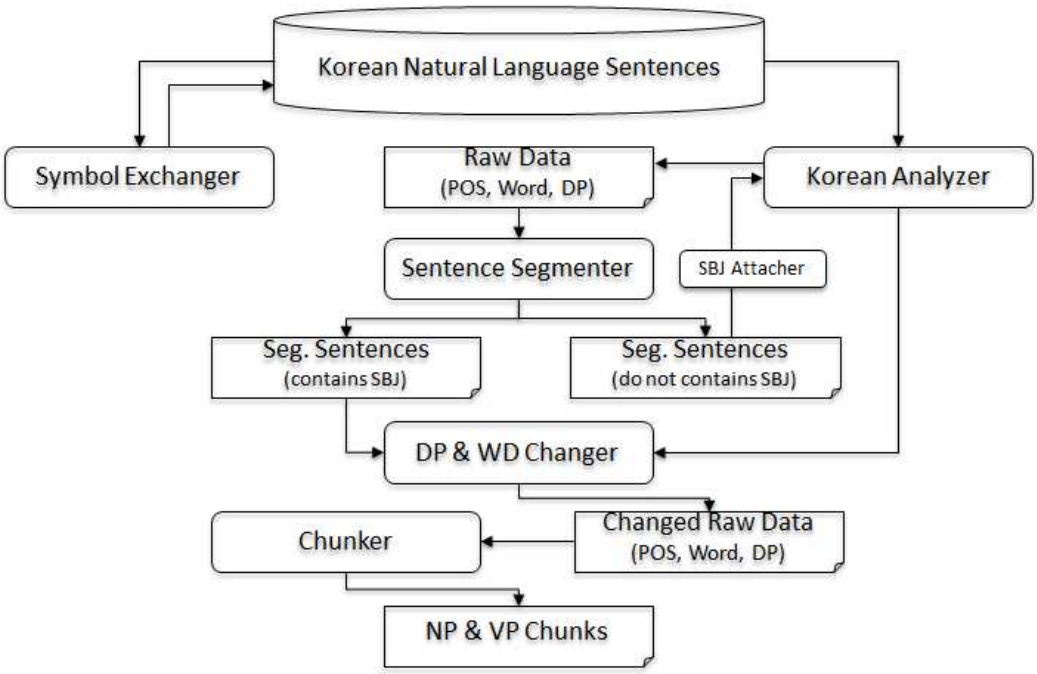
위 기준을 바탕으로 앞서 2가지 예시의 구 묶음 추출 결과를 살펴보면 아래와 같다.

```
==NPChunks==
이것 / 은 / 0 / NP_SBJ
비트겐슈타인의 후기사상을 대표하는 말 /  / 4 / NP

==VPChunks==
이 / 다. / 5 / [0, 4]
```

```
==NPChunks==
비트겐슈타인이라는 이름 / 은 / 1 / NP_SBJ
현재를 가리키는 낱말 / 로 / 5 / NP_AJT

==VPChunks==
종종 쓰이 / ㄴ다. / 6 / [1, 5]
```

3.1.3 KoSeCT

위 두 모듈을 모두 합친 본 프로그램의 메인 모듈은 아래 그림과 같은 흐름도로 구성된다.



먼저 한국어 자연어 문장에 대해서 “Symbol Exchanger” 모듈을 수행하여 여러 기호들을 하나의 통일된 기호(“”)로 수정한 후, 한국어 언어 분석기로 문장을 넘겨준다. 한국어 언어 분석기에서 나온 형태소 분석 결과와 의존 구조 분석 결과, 어절 정보들을 바탕으로 문장 분할 모듈이 수행되고, 문장 분할 결과는 주어를 가진 것과 가지지 않은 것 두 가지 데이터로 구분된다. 주어를 가지지 못한 분리된 문장들은 원 문장의

주어를 새롭게 부착하여 다시 한국어 언어 분석기로 넘겨준다. 예를 들어, “딤미는 사회 전반적으로 노예에 비해서는 지위상 월등히 우월하였으나, 무슬림보다는 낮은 수준의 권리를 행사하였다.”의 문장은 문장 분할 모듈에 의해 “우월하였으나” 어절 전과 후로 나뉘어진다. 이때 뒷 문장 즉, “무슬림보다는 낮은 수준의 권리를 행사하였다.”는 주어가 없으므로 원 문장의 주어인 “딤미는”을 부착해 “딤미는 무슬림보다는 낮은 수준의 권리를 행사하였다.”라는 문장으로 만들어준다. 그 다음으로는 구 묶음 추출 모듈이 수행되는데, 이때 의존 구조 분석 결과와 어절 정보를 수정하는 일은 “DP & WD Changer” 모듈이 담당하고, 이 결과를 바탕으로 “Chunker”가 명사구와 동사구 결과를 반환한다.

3.2 프로그램 기능 제약

- 입력 : (UTF-8 인코딩된) 1개의 한국어 문장
- 출력 : 분할된 여러 문장 및 해당 문장별 복수개의 명사구 및 동사구
- 기본 인코딩 : UTF-8

제 4 장 기타

1) 한국어 Wikipedia 알찬글 1000개의 문장을 대상으로 성능 분석 중
→ 추후에 논문으로 발표 예정.

2) 문장 나누기 모듈의 세분화 예정
→ 조건이나 가정을 보이는 경우, 이유나 원인을 보이는 경우 등 각기 다른 처리 방법

3) 데모 페이지: http://143.248.135.187:11112/

ESPRESSO TOOL

박태호
창원대학교 컴퓨터공학과

Copyright© 2015

차 례

제 1 장 소프트웨어 소개

1.1 Espresso Tool

Espresso Tool은, 본 연구실에서 개발한 한국어 형태소 분석 및 품사 부착기인 Espresso의 이름을 본 따서 사용하였다. Espresso는 모든 커피의 기본 재료가 되며, 형태소 품사가 다양한 자연어처리 시스템의 기본 정보로 사용되기 때문에 이와 같은 이름을 지었다. Espresso Tool은 형태소 분석 및 품사 부착기를 기본으로 하여 그 상위로 구문 분석기, 개채명 인식 및 분류기, 의미역 인식 및 분류기가 하나로 통합된 도구이다. Espresso Tool은 하나의 도구로 다양한 자연어처리 결과를 얻기 위해서 개발하였으며, 궁극적으로 의미 분석과 질의 응답 시스템의 기본 시스템으로 활용하는데 목적이 있다.

1.2 소프트웨어 사용 환경

- 운영체제 : CentOS x64
- 사용 언어 : C++
- 라이브러리 : CRF library (CRF++)[1)]

1.3 소프트웨어 특징

Espresso는 한 번의 명령으로 다양한 자연어처리를 할 수 있다는 장점을 가지고 있다. 현재는 형태소 분석 및 품사 부착, 구문 분석, 개체명 인식 및 분류, 의미역 인식 및 분류 총 4가지의 작업을 수행한다. Espresso는 각각의 독립된 모델이 하나의 자질 클래스를 공유하여 하위의 자연어처리 결과로부터 상위의 자연어처리를 수행할 수 있도록 개발되었다. 이는 4개의 독립된 모델이 각각의 작업을 수행하기 위해 필요한 사전 작업이나 결과 데이터의 관리에 드는 노력을 감소시킨다. 현재 최상위에 있는 모델은 의미역 인식 및 분류 모델로 나머지 3개의 모델의 결과 데이터를 모두 사용한다. 출력된 결과는 JSON 또는 CoNLL 형식으로 출력한다.

1) http://taku910.github.io/crfpp/

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

2.1.1 라이브러리 설치

Espresso Tool을 실행하기 위해서는 CRF++ 설치가 필요하다. CRF++은 "https://taku910.github.io/crfpp/"에서 받을 수 있다. CRF++을 설치하기 위해서는 gcc3.0 이상의 버전이 설치되어 있어야 한다.

```
$ ./configure
$ make
$ su
$ make install
```

2.1.2 컴파일

본 도구는 다음과 같은 방법으로 컴파일한다. 컴파일 위치는 도구가 설치된 최상위 폴더에서 진행한다.

```
$ make
```

2.2 소프트웨어 파일 구조

2.2.1 주요 파일 설명

① 실행 파일

Espresso --- 소프트웨어 실행 파일

② 사전 파일

dic/ ----- 형태소태그 부착을 위한 데이터파일

NeDic/ --- 개체명 사전 (개체명 사전, 개체명 정보, 일반명사 사전)

③ 모델 파일

model/ --- 개체명, 구문분석, SRL모델, 워드벡터

④ 라이브러리 파일

model/ --- 개체명, 구문분석, SRL모델, 워드벡터

⑤ 코드 및 헤더파일

src/ ----- 프로젝트 코드

include/ --- 헤더 파일

json/ --- JSON 라이브러리

2.2.2 전체 구조

[프로젝트폴더]

dic/ ..... 형태소태그 부착을 위한 데이터파일 폴더
lib/ ..... 형태소태거 라이브러리, CRF 라이브러리 폴더
model/ .... 개체명, 구문분석, SRL모델, 워드벡터 폴더
NeDic/ .... 개체명 사전 폴더
src/ ..... 프로젝트 코드 폴더
include/ ..... 헤더 파일 폴더
json/ ..... JSON 라이브러리 폴더
Makefile

2.3 소프트웨어 실행 방법

본 도구는 컴파일 후 생성된 "Espresso"파일로 실행한다. [입력파일]을 결정하는 -i 옵션과 [출력파일]을 결정하는 -o 옵션을 사용하여 다음과 같이 실행한다. 입력파일은 확장자까지 정확하게 적어준다. 실행 파일은 컴파일이 완료되면 도구가 설치된 최상위 폴더 위치에 생성된다.

```
$ ./Espresso -i [입력파일] -o [출력파일] -j

[예] $ ./Espresso -i input.txt -o json.txt -j
```

출력파일은 JSON형식의 결과 파일이 출력된다. [-j] 옵션 없이 실행시키면, CoNLL 2008[2]형식의 결과 파일이 출력된다. ([예] $ ./Espresso -i input.txt -o json.txt)

2) http://catalog.ldc.upenn.edu/LDC2006T03

제 3 장 소프트웨어 기능

3.1 프로그램 기능

Espresso Tool은 하나의 도구로 여러 가지 자연어처리 기능을 수행하는 도구로 현재 형태소 분석 및 품사 태깅, 구문 분석, 개체명 인식 및 분류, 의미역 인식 및 분류의 기능을 수행한다. 한 행이 하나의 문장인 말뭉치를 입력으로 넣고 실행하면 각각의 모델의 결과가 출력 옵션에 따라 JSON 또는 CoNLL 2008형식으로 출력된다. 출력된 문서는 입력된 문서의 문장을 분석하여 형태소 품사 태그, 구문 태그, 개체명 태그, 의미역 논항 태그가 레이블링 되어 있다.

3.2 프로그램 기능 제약

◇ 입력 문서

- 한 행이 하나의 문장인 형식의 말뭉치

제 4 장 기타

4.1 처리 속도 : 초당 80문장

4.2 메모리 사용 : 1,500MB

4.3 시스템 성능

- **형태소 분석 및 형태소 품사 태그 부착** : 96%
- **구문 분석** : 81%
- **개체명 인식 및 분류** : 89% (sport domain)
- **의미역 인식 및 분류** : 65%

Beoltong

김중한
창원대학교 컴퓨터공학과

Copyright© 2015

차 례

제 1 장 소프트웨어 소개

1.1 소프트웨어 명칭

Beoltong은 벌들이 모여사는 집을 의미한다. 그 속에서 사람들의 의견들이 모이고 요구사항에 맞는 정보로 구조화한다는 의미에서 사용되었다. 소셜 네트워크 서비스가 시대의 트렌드가 되면서 생산되는 수많은 데이터를 사용하여 보다 의미 있는 정보를 생산하는 것에 의의를 두고 있다.

1.2 소프트웨어 사용 환경

◇ **서버**

- OS : CentOS release 5.4 (Final)
- 사용언어 : Python, Java, PHP
- 실행 환경 : Python2.5, JDK 1.6, PHP 5. 2. 10 (cli)
- DB : phpMyAdmin - 2.11.11
- 외부 라이브러리 : jquery, xmlrpc

◇ **클라이언트**

- 웹 베이스 시스템으로 PC와 모바일 환경에서 모두 가능

1.3 소프트웨어 특징

첫째, 초기 문서 수집기(crawler)를 이용하여 트위터를 수집하고 사용자의 검색어를 사용하여 지속적인 추가 수집을 수행한다. 그리고 검색 키워드에 대해서 지속적으로 문장들을 수집하고 저장함으로써 부족한 데이터를 충족시키고, 시간의 변화에 따라 정서분류 결과를 확인 가능하다. 이는 트렌트 분석이나 아이템의 인식 변화 등에 활용할 수 있다.

둘째, 지속적으로 수집되는 데이터에 대해 색인 DB를 만들어서 검색 속도를 향상시킨다. 이는 초기 문서 수집기를 통해 수집된 데이터를 정서 분석 엔진을 사용하여 분석 결과 업데이트를 진행한다. 따라서 한번이라도 검색된 아이템의 데이터를 증가시켜 보다 효율적인 데이터 분석이 가능해지도록 한다. 또한 동일한 아이템 검색 시 이미 구축한 데이터를 바탕으로 빠른 속도로 정서 분석 결과를 확인할 수 있다.

셋째, Bar Graph, Line graph를 통해 검색하는 아이템에 대한 정서 분석 결과를 쉽게 확인할 수 있다. 그리고 키워드에 대해 누적 빈도, 일별, 월별 기간별로 긍정과 부정의 변화를 확인 가능하다.

◇ **활용 방안**

1) 기업

- 기업 이미지 및 선호도 분석

대중이 느끼는 기업의 이미지나 선호도 등을 분석할 수 있다.

- 자사 제품/서비스에 대한 시장의 반응이나 만족도 평가

자사의 제품이나 서비스에 대한 시장의 반응이나 만족도를 분석할 수 있다.

- 신제품 개발, 마케팅을 위한 시장 조사

경쟁 업체들의 제품에 대한 정서 분석을 통해 신제품을 개발하거나 차별화된 마케팅을 할 수 있는 정보를 분석해낼 수 있다.

2) 공공기관

- 공공서비스에 대한 만족도 평가

공공서비스에 대한 정서를 분석하여 사람들의 만족도를 평가하여, 부족한 부분을 개선하고, 좋은 점은 홍보에 활용할 수 있다.

- 시책에 대한 여론 분석

공공기관이 추진하고 있거나, 새롭게 추진하려고 하는 시책에 대한 사람들의 반응을 분석하여 시책을 보완/개선하는데 사용할 수 있다.

3) 공인

- 정치인/연예인 등에 대한 대중의 이미지 및 영향력 분석

소셜 네트워크 서비스의 데이터에 나타나 있는 대중에 비친 공인, 특히 정치인이나 연예인의 이미지나 영향력을 분석하는데 유용하게 활용할 수 있다.

4) 기타

- 악성 댓글 판별 서비스

댓글을 정서 분석하면 악성 여부를 판단할 수 있으며, 그 결과로부터 악성 댓글을 차단하거나 필터링하는 서비스에 응용할 수 있다.

- 지능화된 타겟팅 광고 서비스

문서를 정서 분석한 결과를 바탕으로, 긍정적인 내용의 문서에는 관련 광고를 추천해주고, 부정적인 내용의 문서에는 관련 광고는 노출되지 않도록 지능화된 타겟팅 광고 서비스를 제공할 수 있다.

제 2 장 소프트웨어 설치 및 실행

2.1 소프트웨어 설치 방법

서버 - 클라이언트 구조로 서버에 시스템 설치 후 웹페이지 접속이 가능하다.

2.2 소프트웨어 파일 구조

2.2.1 주요 파일 설명

① 실행 파일

BeoltongCrawler/run.sh --- 트위터 크롤러 실행 스크립트
BeoltongEngine/Makefile --- 벌통 엔진 컴파일 파일
BeoltongEngine_NewIndexer/Makefile --- 벌통 웹 기능 컴파일 파일

② 데이터 파일

ParserWithTagger/dic/LEXICON.morph2.dat --- 형태소 분석기 어휘 파일
ParserWithTagger/dic/LEXICON.morph2.idx --- 형태소 분석기 어휘 인덱스 파일
ParserWithTagger/dic/LEXICON.prob.mhdict --- 형태소 분석기 확률 파일
ParserWithTagger/dic/LEXICON.sym2.dat --- 형태소 분석기 기호 파일
ParserWithTagger/dic/LEXICON.sym2.idx --- 형태소 분석기 기호 인덱스 파일
ParserWithTagger/model/D.model --- Arc 분석기 학습 모델
ParserWithTagger/model/E.model --- 구문 태그 분석기 학습 모델

③ 사전 파일

BeoltongEngine/dic/noun.kor.list --- 긍정/부정 명사 단어 사전
BeoltongEngine/dic/negation.kor.list --- 부정 단어 사전
BeoltongEngine/dic/conj.kor.list --- 활용 사전
BeoltongEngine/dic/LEXICON.txt --- 어휘 사전
BeoltongEngine/dic/adj.kor.neg --- 부정 형용사 사전
BeoltongEngine/dic/adj.kor.pos --- 긍정 형용사 사전

2.2.2 전체 구조

Beoltong/
BeoltongCrawler/ ..... 크롤러 파일 폴더
TwitterCrawler_lib/ ..... 크롤러 라이브러리 파일 폴더
TwitterCrawler.jar ..... 크롤러 Jar 파일
run.sh ..... 크롤러 실행 스크립트

BeoltongEngine/ ..... 정서분석 파일 폴더
model/ ..... 모델 파일 폴더
dic/ ..... 사전 파일 폴더
BeoltongEngine_NewIndexer/ .. 웹 기능 파일 폴더
model/ ..... 모델 파일 폴더
dic/ ..... 사전 파일 폴더
ParserWithTagger/ ..... 형태소, 구문 분석 파일 폴더
KGuruTagger/ ..... 형태소 분석 파일 폴더
model/ ..... 모델 파일 폴더
dic/ ..... 사전 파일 폴더

2.3 소프트웨어 실행 방법

웹 서비스 URL : http://eos.nlp.wo.tc/

제 3 장 소프트웨어 기능

3.1 프로그램 기능

3.1.1 초기 문서 수집기 (Crawler)

초기 문서 수집기는 트위터(twitter)에서 키워드에 대한 글을 수집한다. 또한 수집되는 글에는 아이디, 작성자, 작성시간, 작성자의 위치를 포함하여 저장된다.

3.1.2 키워드 DB (Query DB)

사용자가 검색했던 키워드를 저장하는 DB이다. 이 기능은 사용자가 검색한 키워드에 대한 데이터를 지속적으로 수집하기 위해서 사용된다.

3.1.3 정서 분석 엔진 (Sentiment Engine)

키워드에 대해서 수집된 데이터를 정서 분석하여 긍정적인 문장과 부정적인 문장을 분류하는 동작을 수행한다. 가장 먼저 수집된 데이터를 형태소 분석과 구문 분석을 수행한다. 분석된 데이터의 정서 분석을 위해 사전 파일을 사용해 키워드에 대해 긍정적인 단어를 포함하고 있으면 긍정적인 문장, 부정적인 단어를 포함하고 있으면 부정적인 문장으로 판별하게 된다.

3.1.4 색인 DB (Indexing DB)

사용자가 더욱 빠른 정보 검색을 위해서 정서 분석 엔진을 통해 분석된 결과를 색인하여 저장하고 있다.

3.1.5 인터페이스

클라이언트 서비스 인터페이스로 웹을 기반으로 키워드 검색과 해당 키워드의 정서 분석 결과를 확인할 수 있다. 정서 분석 결과는 파란색은 긍정, 빨간색은 부정으로 표현된다.

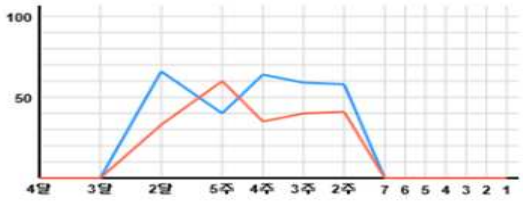
◇ 메인 화면

웹 서비스의 메인 화면으로 검색된 키워드에 대한 랭킹과 해당 키워드에 대한 정서 분석 결과를 보여준다. 키워드 랭킹은 사용자가 검색한 키워드에 대한 빈도로 결정된다.

◇ 정서 분석 결과

검색한 키워드에 대해서 수집된 데이터의 정서 분석 결과를 비율로 보여준다.

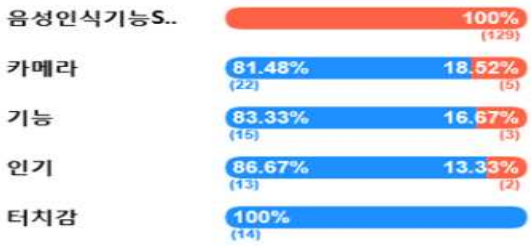
◇ 데이터 수집 그래프



검색하는 키워드에 대해서 수집된 데이터의 양을 그래프로 보여준다. 이 그래프로

해당 키워드에 대해서 수집되는 데이터의 양을 일 단위, 주 단위, 월 단위로 살펴볼 수 있다.

◇ **부주제에 대한 정서 분석 결과**



검색하는 키워드와 같이 등장하여 키워드와 관련된 단어을 포함하고 있는 데이터에 대한 정서 분석 결과를 보여준다.

◇ **검색 키워드에 대한 실제 문장**

검색한 키워드에 대해서 수집된 실제 문장을 보여준다.

3.2 프로그램 기능 제약

검색된 키워드를 중점으로 문서 수집기가 동작하게 됨으로써 수집된 키워드에 대해서만 데이터가 축적된다. 따라서 새로운 키워드에 대해서는 능동적인 동작을 수행하기 힘들다.

조직 위원장

김 학 수 강원대학교

조직 위원

고 영 중 동아대학교
김 유 섭 한림대학교
최 성 필 경기대학교
김 선 철 국립국어원
이 창 기 강원대학교
황 용 주 국립국어원

심사 위원장

강 현 규 건국대학교

심사 위원

김 경 선 다이퀘스트
김 학 수 강원대학교
이 창 기 강원대학교
김 선 철 국립국어원
이 도 길 고려대학교
최 정 도 국립국어원

2015 국어 정보 처리 시스템 경진 대회

- 국립국어원 국어생활 질의 응답 시스템 개발 및 적용 -

발 행 인 송 철 의
발 행 처 국립국어원
서울특별시 강서구 금낭화로 154
등록번호 국립국어원 2015-01-08
인 쇄 일 2015년 10월 8일
발 행 일 2015년 10월 16일
주 관 한국정보과학회 언어공학연구회
서울시 서초구 방배3동 984-1 머리재빌딩 401호
주 최 국립국어원