Data
Analysis

Bank DATA with Naver Blog

Text Big Data Analysis Project

20182838 경영정보학부 김상준
20220335 행정학과 조은나라

Data
Analysis

CONTENTS

01

데이터 소개

- Keywords
- Data Source

02

데이터 수집

- API를 활용하여 은행 이름을 검색하고 블로그 본문 수집
- 데이터 수집 함수 설정: fetch_blog_posts
- csv 형태의 원본 데이터 수집 결과

03

데이터 전처리

- 데이터 전처리 대상 파악
- 데이터 전처리 후 데이터의 수 파악
- 2차 데이터 전처리: remove_emoji
- 2차 데이터 전처리: preprocess_korean

04

데이터 모델링

- WordCloud 시각화 & Insight
- LDA 시각화 & Insight

Chapter
01

Keywords

중앙은행

- 한국은행

시중은행

- 국민은행
- 우리은행
- SC제일은행
- 한국씨티은행
- 신한은행

인터넷전문은행

- 케이뱅크
- 카카오뱅크
- 토스뱅크

국책은행

- 한국산업은행
- 중소기업은행
- 한국수출입은행

특수은행

- 수협은행
- 농협은행

지방은행

- 대구은행
- 부산은행
- 경남은행

저축은행

- IBK저축은행
- 웰컴저축은행

Data Source

NAVER API를 활용하여 Blog 본문 수집

Chapter
01

기대되는 은행별 특성

시중은행

개인과 기업에게 예금, 대출, 외환 등 다양한 금융 서비스를 제공, 전국적인 네트워크와 광범위한 지점

중앙은행

국가의 통화 정책을 관리하고 금융 시스템의 안정성을 유지하는 역할

인터넷전문은행

온라인을 통해 모든 금융 서비스를 제공, 기술로 고객의 편의성을 극대화

국책은행

국책은행은 정부의 경제정책을 지원하고, 기업과 국가 프로젝트에 자금을 공급

특수은행

정부산업을 지원하기 위해 설립된 은행으로, 해당 분야의 종사자들에게 맞춤형 금융 서비스를 제공

지방은행

특정 지역을 중심으로 금융 서비스를 제공하며, 지역 경제 활성화와 지역 주민의 금융 편의성을 높이는 데 중점

상호저축은행

지역 사회의 중소기업과 개인을 대상으로 예금 및 대출 서비스를 제공하며, 상호금융의 형태로 운영

Chapter
02

데이터 수집

API를 활용하여 은행 이름을 검색하고 블로그 본문 수집

- API 아이디, 비밀번호, 수집할 곳 설정
- 20개의 키워드 설정, 수집 시간을 고려해 2번에 나눠 수집
- 한번에 가져오는 결과는 30개, 수집 시 브라우저가 뜨지 않도록 설정

```
1
2  # 네이버 API 설정
3  [redacted]
4  [redacted]
5  base_url = 'https://openapi.naver.com/v1/search/blog.json'
6
7  # 2번에 나눠서 수집
8   keywords = ['국민은행', '우리은행', 'SC제일은행', '하나은행', '신한은행','한국은행',
9              '케이뱅크','토스뱅크','농협은행','수협은행','대구은행','부산은행','경남은행']
10 # keywords = ['한국산업은행', '중소기업은행', '한국수출입은행','한국씨티은행','카카오뱅크' ]
11 n_display = 30  # 한 번에 가져오는 결과 수를 30개로 설정
12 sort = 'sim'
13
14 # 드라이버 설정
15 options = Options()
16 options.add_argument('--headless')  # 헤드리스 모드 설정, 창 안뜨어나게 설정
```

Chapter
02

데이터 수집

데이터 수집 함수 설정: fetch_blog_posts

- Edge 드라이버를 사용하여 진행
- 최대 999개의 결과를 가져오고, 1000을 넘어가면 중지하도록 설정
- urllib를 통해 API 정보를 기입
- 마찬가지로 URL에 대한 요청을 전송하고, 전송을 확인하는 과정 수행

```python
def fetch_blog_posts(keyword):
    # Edge 웹드라이버를 사용하여 크롤링을 진행
    driver = webdriver.Edge(options=options)
    visited_links = set()  # 방문한 링크를 추적하기 위한 집합
    all_results = []  # 모든 결과를 저장할 리스트

    start = 1  # 시작 페이지

    # 최대 999개의 결과를 수집할 때까지 반복
    while len(all_results) < 999:
        if start > 1000:  # 1000 페이지를 초과하면 중지
            break

        query = keyword  # 검색어 설정
        encQuery = urllib.parse.quote(query)  # 검색어 URL 인코딩
        url = f'{base_url}?query={encQuery}&display={n_display}&start={start}&sort={sort}'  # 검색 URL 생성

        my_request = urllib.request.Request(url)
        my_request.add_header("X-Naver-Client-Id", client_id)  # 네이버 API 클라이언트 ID 추가
        my_request.add_header("X-Naver-Client-Secret", client_secret)  # 네이버 API 클라이언트 시크릿 추가

        try:
            response = urllib.request.urlopen(my_request)  # 요청 전송
            rescode = response.getcode()  # 응답 코드 확인
            if rescode == 200:
                response_body = response.read()  # 응답 본문 읽기
            else:
                print(f"Error Code: {rescode}")  # 오류 코드 출력
                break

            search_results = response_body.decode('utf-8')  # 응답 본문 디코딩
            search_results = json.loads(search_results)  # JSON 파싱
```

Chapter
02

데이터 수집

데이터 수집 함수 설정: fetch_blog_posts

- link를 추출하고 처리하는 과정을 수행
- 조건문을 통해 link를 확인하고, 드라이버로 링크 열기
- 원활한 수집을 위해 driver 로딩 설정(iframe)
- 블로그 제목(title)과 본문(body) 위치 탐색
- 동적 로딩 문제가 있는 본문에 대해선 "No contents" 설정
- link, title, body, keyword를 포스트 정보로 저장
- 결과를 리스트에 저장하고, link 추출에 오류 발생 시 예외처리 설정

```python
        for item in search_results['items']:
            link = html.unescape(item['link']).replace('\\', '')  # 링크 추출 및 처리

            if 'blog.naver.com' in link and link not in visited_links:
                driver.get(link)  # 블로그 링크 열기
                try:
                    # iframe 로딩을 대기 후 전환
                    iframe = WebDriverWait(driver, 20).until(
                        EC.presence_of_element_located((By.ID, 'mainFrame'))
                    )
                    driver.switch_to.frame(iframe)

                    # 제목 위치 탐색
                    title_elem = driver.find_elements(By.CLASS_NAME, 'pcol1')
                    title = title_elem[0].text if title_elem else 'No title'

                    # 본문 위치 탐색 추가
                    body = 'No contents'
                    body_elem = driver.find_elements(By.CLASS_NAME, 'se-main-container')
                    if body_elem:
                        body = body_elem[0].text.replace('\n', '')
                    else:
                        body_elem = driver.find_elements(By.CLASS_NAME, 'post-view')
                        if body_elem:
                            body = body_elem[0].text.replace('\n', '')

                    # 포스트 정보 저장
                    post_info = {
                        'link': link,
                        'title': title,
                        'contents': body,
                        'keyword': keyword
                    }

                    all_results.append(post_info)  # 결과 리스트에 추가
                    visited_links.add(link)  # 방문한 링크에 추가
                    driver.switch_to.default_content()

                except Exception as e:
                    print(f"Error processing {link}: {e}")  # 예외 발생 시 출력
                    continue
```

Chapter
02

데이터 수집

데이터 수집 함수 설정: fetch_blog_posts

- 요청 간격을 위한 시간을 설정하고, 각종 오류에 대해 예외 처리 설정
- 만약 결과가 리스트에 저장되었다면, 데이터프레임을 생성
- 생성된 데이터 프레임은 다시 csv 형태로 저장
- ThredPool을 사용하여 키워드를 동시에 병렬로 데이터 수집하도록 설정
- 전체 소요된 시간 측정

csv 형태의 원본 데이터 수집 결과

	link	title	contents	keyword
2	https://blog.naver.com/ha	저축은행주택담보대출	저축은행주택담보대출금리 비교 (신한은행, 우리은행, 국민은	국민은행
3	https://blog.naver.com/ho	국민은행주택담보대출	국민은행주택담보대출금리 신한은행, 우리은행, 하나은행, 농	국민은행
4	https://blog.naver.com/ok	KB 국민은행 신용카드	KB 국민은행 신용카드 추천 TOP 3 발급 혜택 정리안녕하세요	국민은행
5	https://blog.naver.com/wl	KB국민은행 청년 주택	청년층 사이에서 주택 청약에 대한관심도가 높아지고 있습니	국민은행
6	https://blog.naver.com/a4	KB국민은행 주택담보대	KB국민은행 주택담보대출금리 고정 변동 기간 궁금한 조건금	국민은행
7	https://blog.naver.com/ja	kb국민은행kb청년도약	사회에 진입하는 청년들에게 조금이나마 보탬이 되도록 정부	국민은행
8	https://blog.naver.com/hs	우리은행 국민은행 농	우리은행 국민은행 농협 주택담보대출 금리 한도 비교!계속되	국민은행
9	https://blog.naver.com/to	kb 부동산시세 보는법	안녕하세요.요새 아파트 매매 거래량과 매매가가 꿈틀거리고	국민은행
10	https://blog.naver.com/gr	국민은행 오피스텔 담!	안녕하세요 센터장입니다.오피스텔 주담대를 쉽게 알아보는	국민은행

```python
            start += n_display  # 다음 페이지로 이동
            time.sleep(3)  # 요청 간격을 두기 위해 잠시 대기

        except urllib.error.HTTPError as e:   # HTTP 오류 처리
            print(f"HTTP Error during fetching blog posts for {keyword}: {e.code} - {e.reason}")
            break
        except urllib.error.URLError as e:  # URL 오류 처리
            print(f"URL Error during fetching blog posts for {keyword}: {e.reason}")
            break
        except Exception as e:  # 기타 예외 처리
            print(f"Unexpected Error during fetching blog posts for {keyword}: {e}")
            break

    driver.quit()  # 드라이버 종료

    if all_results:  # 결과가 있을 경우
        df = pd.DataFrame(all_results)  # 데이터프레임 생성
        csv_file_path = f'{keyword}.csv'  # 파일 경로 설정
        df.to_csv(csv_file_path, index=False)  # CSV 파일로 저장
        print(f"Data for {keyword} saved to {csv_file_path}")  # 저장 완료 메시지 출력
    else:
        print(f"No data collected for {keyword}")  # 결과가 없을 경우 메시지 출력

# 전체 실행 시간 측정 시작
start_time_total = time.time()

with ThreadPoolExecutor() as executor:
    executor.map(fetch_blog_posts, keywords)  # 스레드 풀을 이용해 키워드 별로 함수 실행

# 전체 실행 시간 측정 끝
end_time_total = time.time()
elapsed_time_total = end_time_total - start_time_total
print(f"Total elapsed time: {elapsed_time_total:.2f} seconds")  # 전체 소요 시간 출력
```

Chapter
03

데이터 전처리

데이터 전처리 대상 파악

- csv 형태의 데이터에 대해 3가지 encoding 방법 시도
- 데이터프레임 형태로 다시 변환
- 중복되는 데이터 계산
- "No contents" 의 수 계산
- 본문 내용이 완전히 비어져 있는 경우 계산
- print를 통해 데이터의 현황 파악

```python
keywords = ['국민은행', '우리은행', 'SC제일은행', '하나은행', '신한은행', '한국은행', '한국씨티은행',
            '케이뱅크', '카카오뱅크', '토스뱅크',
            '한국산업은행', '중소기업은행', '한국수출입은행', '수협은행', '농협은행',
            '대구은행', '부산은행', '경남은행','IBK저축은행','웰컴저축은행']
print('키워드의 수: ', len(keywords))
print('-' * 40)

# 키워드별 데이터프레임 저장 딕셔너리
dataframes = {}

for keyword in keywords:
    file_path = f'{keyword}.csv'  # 파일 이름을 키워드에 맞게 설정
    for encoding in ['utf-8', 'euc-kr', 'cp949']:  # 다양한 인코딩 시도
        try:
            df = pd.read_csv(file_path, encoding=encoding)  # CSV 파일 로드

            # 중복된 행 세기
            dup_count = df['contents'].duplicated().sum()

            # 'No contents'의 수 세기
            no_count = df['contents'].str.contains('No contents', na=False).sum()

            # 본문 내용이 완전히 비어져 있는 경우 세기
            na_count = df['contents'].isna().sum()

            print(f'키워드: {keyword}')
            print('중복 수:', dup_count)
            print('No contents의 수:', no_count)
            print('본문 내용이 비어 있는 행의 수:', na_count)
            print('-' * 40)
```

Chapter
03

데이터 전처리

데이터 전처리 후 데이터의 수 파악

- 중복 데이터 제거
- 본문이 "No contents" 라면 제거
- 공백 데이터 제거
- 전처리된 데이터를 다시 csv 형태로 저장
- 모든 데이터 프레임의 데이터 총 개수 계산 → 17330개

```python
            # 중복 제거
            df = df.drop_duplicates(subset=['contents'])

            # 'No contents' 또는 공백 또는 NaN이 있는 행 제거
            df = df[~df['contents'].apply(lambda x: pd.isna(x) or (isinstance(x, str) and
            (x.strip() == '' or x.strip().lower() == 'no contents')))]

            # 정제된 데이터프레임 저장
            dataframes[keyword] = df

            # 전처리된 데이터프레임을 CSV 파일로 저장
            df.to_csv(f'preprocessed_{keyword}.csv', index=False)
            break  # 성공적으로 로드되면 다음 키워드로 이동
        except UnicodeDecodeError:
            continue  # 인코딩 오류 발생 시 다른 인코딩으로 시도

# 모든 데이터프레임의 데이터 총 개수 계산
total_rows = sum(len(df) for df in dataframes.values())
print(f'중복 및 불필요한 데이터 제거된 20개 데이터프레임의 데이터 총 개수: {total_rows}')
```

```
----------------------------------------
키워드: IBK저축은행
중복 수: 39
No contents의 수: 40
본문 내용이 비어 있는 행의 수: 0
----------------------------------------
키워드: 웰컴저축은행
중복 수: 23
No contents의 수: 24
본문 내용이 비어 있는 행의 수: 0
----------------------------------------
중복 및 불필요한 데이터 제거된 20개 데이터프레임의 데이터 총 개수: 17330
```

Chapter
03

데이터 전처리

2차 데이터 전처리: remove_emoji

- kiwi를 활용하여 2차 전처리 함수 정의
- 품사가 명사인 경우만 허용
- 본문에 포함되어 있는 이모티콘을 제거하는 함수 설정

```python
# 전처리 함수 정의
kiwi = Kiwi(typos='basic')
stopwords = Stopwords()

allowed_tags = {'NNG', 'NNP'}

def remove_emoji(text):
    emoji_pattern = re.compile("["
                            u"\U0001F600-\U0001F64F"  # emoticons
                            u"\U0001F300-\U0001F5FF"  # symbols & pictographs
                            u"\U0001F680-\U0001F6FF"  # transport & map symbols
                            u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
                            "]+", flags=re.UNICODE)
    return emoji_pattern.sub(r'', text)
```

Chapter
03

데이터 전처리

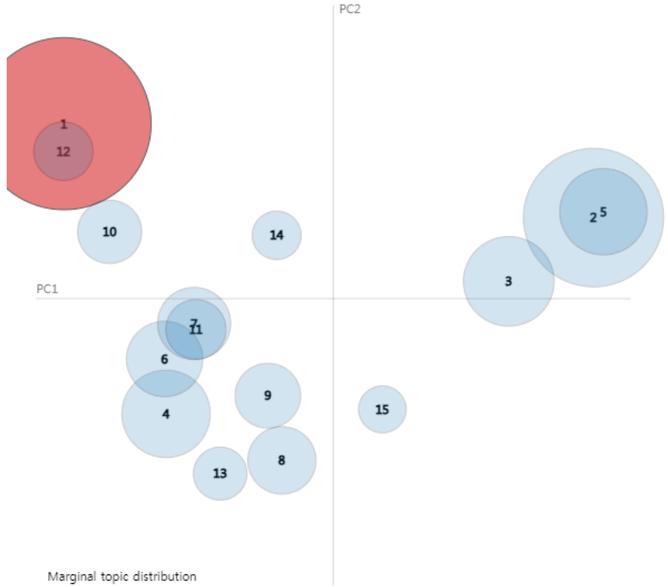
2차 데이터 전처리: preprocess_korean

- remove_emoji를 수행하여 이모티콘 제거
- 줄바꿈 문자 제거
- kiwi를 통해 띄어쓰기 교정
- 위 작업이 진행된 text에 대해 문장 토큰화 수행
- 토큰화된 문장에 대해, 다시 개별 문장에 형태소 분석 수행
- 형태소 분석 시 품사가 명사인지 판별
- 형태소와 표제어 길이가 두글자 이상인지 판별
- 형태소 분석 결과를 join하고 특수문자를 제거한 후 리스트에 결과 담기
- 문장을 다시 join하여 결과 담기

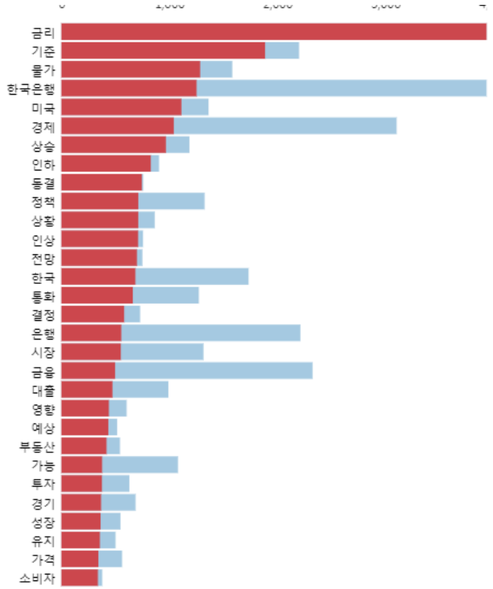
```python
def preprocess_korean(text, analyzer=kiwi):
    my_text = copy.copy(text)
    my_text = remove_emoji(my_text)  # (1) 이모지 제거
    my_text = my_text.replace('\n', ' ')  # (2) 줄바꿈 문자 제거
    my_text = analyzer.space(my_text)  # (3) 띄어쓰기 교정
    sents = analyzer.split_into_sents(my_text)  # (4) 문장 토큰화
    p = re.compile('[^ㄱ-ㅎㅏ-ㅣ가-힣 ]')
    all_result = []
    for sent in sents:
        token_result = analyzer.tokenize(sent.text, stopwords=stopwords)  # (5) 형태소 분석
        result = []
        for token in token_result:
            word = token.form  # 형태소
            lemma = token.lemma  # 표제어 추출
            tag = token.tag  # 품사 태그
            if tag in allowed_tags and lemma and len(lemma) > 1:
                result.append(lemma)
            elif tag in allowed_tags and word and len(word) > 1:
                result.append(word)
        token_result = ' '.join(result)  # (6) 형태소 분석한 결과를 다시 join
        token_result = p.sub(' ', token_result)  # (7) 특수 문자 제거 (=한글을 제외한 문자 제거)
        all_result.append(token_result)

    all_result = ' '.join(all_result)  # (8) 모든 문장을 하나의 string으로 join

    return all_result
```

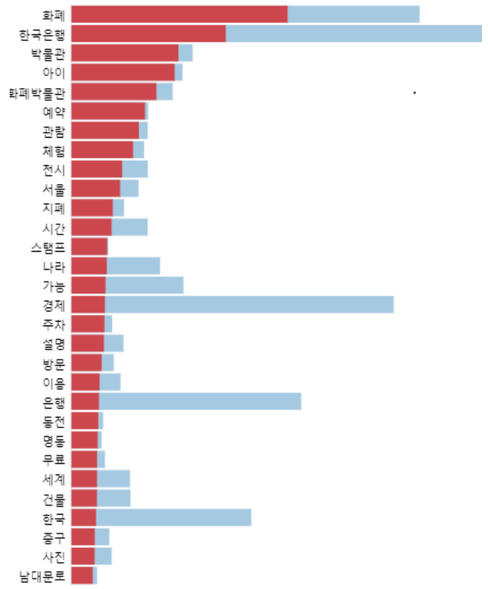
한국은행 - 중앙은행

Word Cloud



LDA - Topic 1





Topic 2

Word Cloud

- 한국은행의 주요 정책 수단 중 하나며, 인플레이션 등 이슈 때문에 기준 금리와 침체를 극복하기 위한 **금리 인하**가 크게 노출
- 한국 은행에서 운영하는 박물관에서 화폐의 역사와 문화를 알리고 전시 프로그램를 운영하기에 **화폐 박물관**이 크게 노출
- 한국 은행의 핵심 역할 중 하나로, 통화 공급과 금리 조절를 하는 **통화 정책**이 크게 노출

⇒Conclusion: 높은 성능

LDA

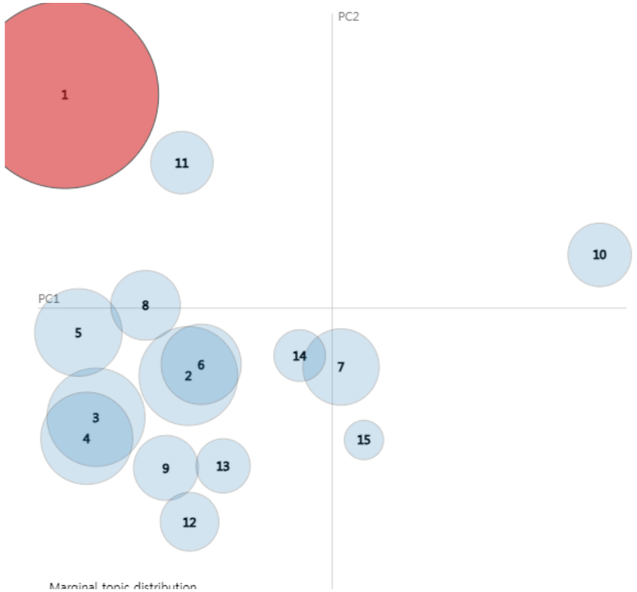
- Topic 1의 경우 **금리, 미국, 물가, 상승 등** 단어로 보아 금리와 물가 상승은 통화 정책의 핵심 요소이면서, 미국의 경제 상황과 정책이 한국 경제에 미치는 영향이 있음을 시사
- Topic 2의 경우 **박물관, 아이, 화폐, 관람 등** 단어로 보아 화폐 박물관이 교육과 문화적 역할을 수행하고 있음을 시사

중소기업은행 - 국책은행

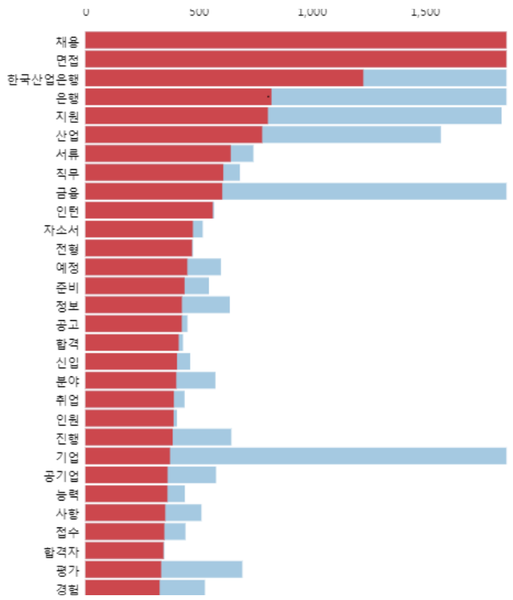
아파트
경매
말소
기준
임의
경우
기업
중소기업은행
기업은행
중소기업은행
벤처
기업
해당
사건
경매
공장
용지
이상
필요
관련
중소기업은행
근저당 설정
경매 신청
물건
생각
내용
금융 기관
입찰 보증금
사건
번호
가능
공공기관
지원
아파트 단지
정보
산업 단지 소유 이전
포함
상품
운영
상황
소재
다양
경매 진행
경매 사건
경매 자금
기업 은행
중소기업
공장 용지
근린 생활
생활 시설
블루 랜드
랜드 베스트
사용 승인
청구 금액
거래
정도
평균 연봉
생각
이자
사건 법원
경매 물건
중소기업
공장 용지
경매 종료
기업 감정
기업 은행
경매 사건
경매 자금
경매 진행
경매 감정
경매 입찰
경매 기업
경매 기관
경매 기간
경매 기준
경매 등기
경매 번호
경매 공공
경매 공장
경매 가능
경매 사건
경매 상황
경매 소재
경매 해당
경매 이상
경매 필요
경매 관련
경매 신청
경매 물건
경매 생활
경매 생각
경매 내용
경매 금융
경매 입찰
경매 보증금
경매 말소
경매 기준
경매 임의
경매 경우
경매 기업
경매 벤처
경매 사건
경매 다양
경매 종료
경매 제공
경매 지원
경매 아파트
경매 단지
경매 산업
경매 소유
경매 이전
경매 포함
경매 상품
경매 운영

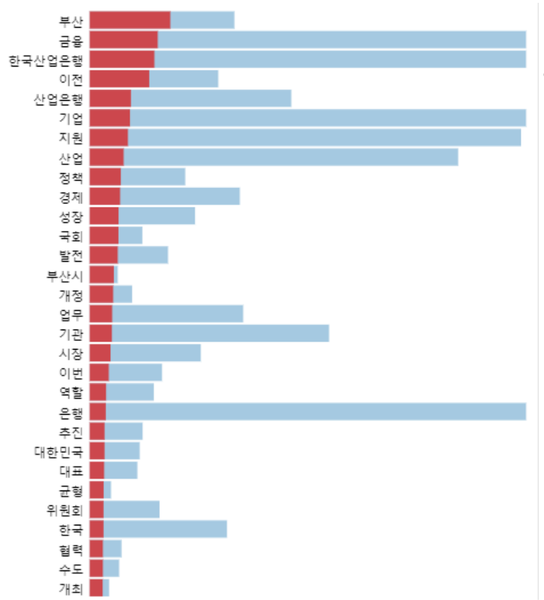
한국산업은행 - 국책은행

Word Cloud



LDA - Topic 1





Topic 2

Word Cloud

- 한국산업은행의 핵심 임무는 기업에 대한 금융 지원으로, 중소기업 및 벤처 기업을 육성하기 때문에 **기업**과 **지원**이 크게 노출
- 하지만 은행 이름 등 결과가 크게 노출됨 ⇒**Conclusion:** 낮은 성능

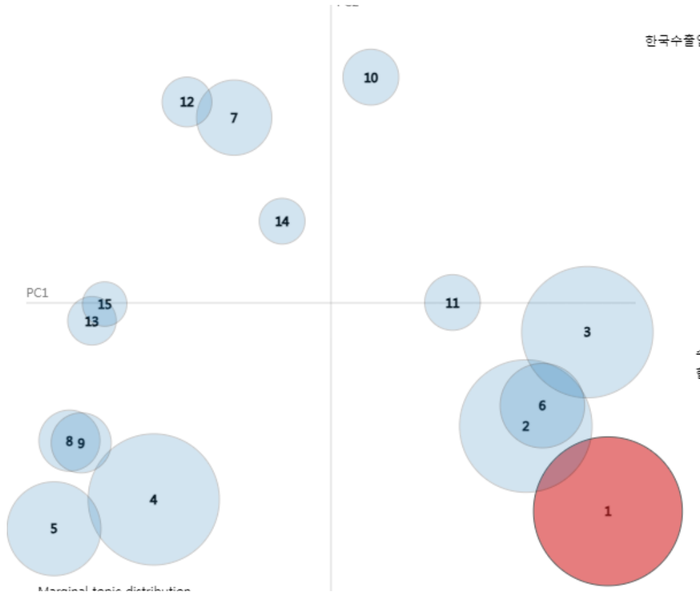
⇒**Solution:** LDA로 세부 결과 살펴보기

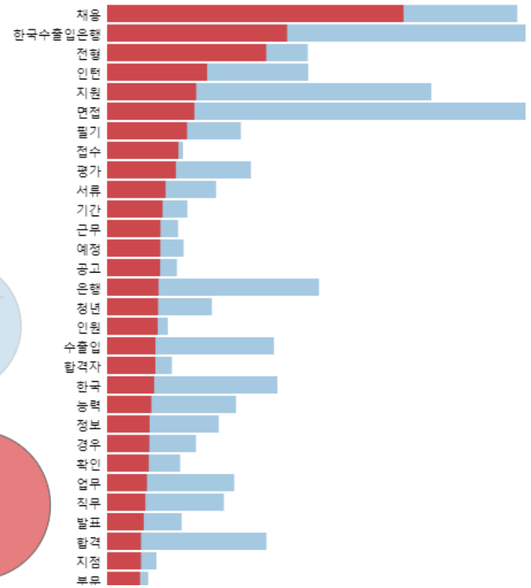
LDA

- Topic 1의 경우 **채용, 면접, 지원, 서류 등** 단어로 보아 지원자들의 공기업 정기적 인재 채용 관심도를 나타냄
- Topic 2의 경우 **금융, 투자, 기업, 산업 등**의 단어로 보아 다양한 금융 상품과 서비스를 통해 기업에게 자금을 지원함을 드러냄

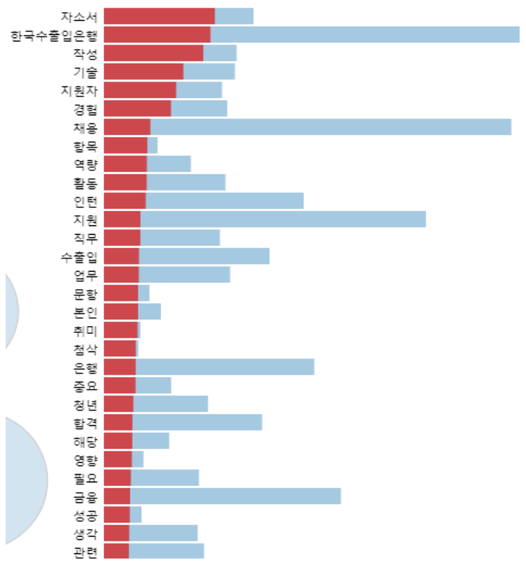
한국수출입은행 - 국책은행

Word Cloud





LDA - Topic 1



Topic 2

Word Cloud

- 공기업 채용과 관련하여 인재발굴을 위한 프로그램, 채용 과정에서의 주요 절차의 중요성 때문에 **청년인턴** 및 **필기 전형**이 크게 노출됨
- 은행으로서의 역할을 잘드러낸 것 같지는 않음 ⇒Conclusion: 낮은 성능

⇒Solution: LDA로 세부 결과 살펴보기

LDA

- Topic 1의 경우 **기업, 지원, 금융, 수출** 등 단어로 보아 국내 기업이 수출입 거래를 원활히 진행할 수 있게 금융 지원을 제공함을 나타냄
- Topic 2의 경우 **면접, 준비, 질문, 학원** 등 단어로 보아 공기업 채용 과정에서 지원자들의 관심도를 드러냄

대구은행 - 지방은행

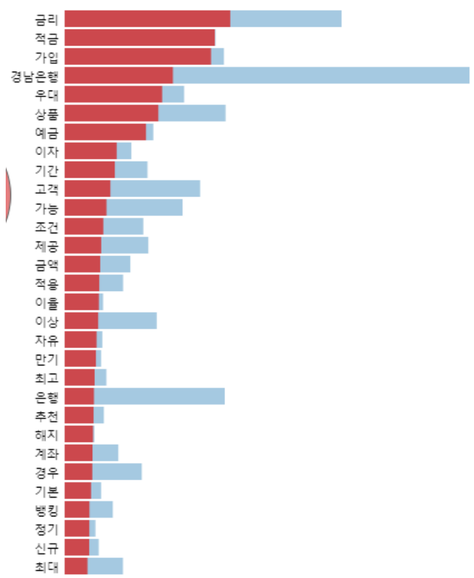
가능
다양
시중
은행
대구
은행
전환
대구
은행
진행
대구은행 파크
금융
방문
정도
시간
이용
필요
이번
지방 은행
대구은행 본점
대구은행 시중
사진 동영상
참고
마지막
천구
추가
신청
자리
오늘
혜택 대구은행
해당
아이
생각
아직
고객
선택
수성구 달구벌대로
대구광역시 수성구
대학생 홍보
비상금 대출
방법
관련
예정
기업
공간
마음
이름
전환 인가
최대
지금
이자
이상
인터넷 뱅킹
확인
주문
조회 서비스
메뉴
특별
처음
사용
상황
준비
배너 배너
우대 금리
기준
홍보 대사
활동
소개
다음
고기
조건
금융 그룹
계시물 대구은행
대구은행 소식
이벤트 사진
추천
관리
분위기
은행 시중
고객 센터
동영상 보기
은행 본점
할인
소식 이벤트
내용
대구 경북
설명
대구 수성구
안내
비교
대구은행 공식
대구은행 뱅크
대구은행 공식
시작
기대
오픈
이벤트
참여
일반
담보 대출
최근
느낌
포스팅
예약
가격
가게
카페
기본
이유
도움
최고
이상
인터넷 뱅킹
대구광역시 북구
대구광역시 수성구
지원
요즘
포함
경기 예금
경기장
금융 위원회
신용 대출
은행 시중
직원
선택
행사
기존
분위기
동영상 보기
네이버 블로그
대구은행 뱅크
대구은행 소식
대구은행 시중
대구은행 본점
대구은행 시중
대구은행 공식
대구은행 소식
대구은행 뱅크
대구은행 공식
대구은행 시중
대구은행 본점
대구은행 공식
대구은행 시중
대구은행 본점
대구은행 공식
대구은행 시중
대구은행 본점

부산은행 - 지방은행

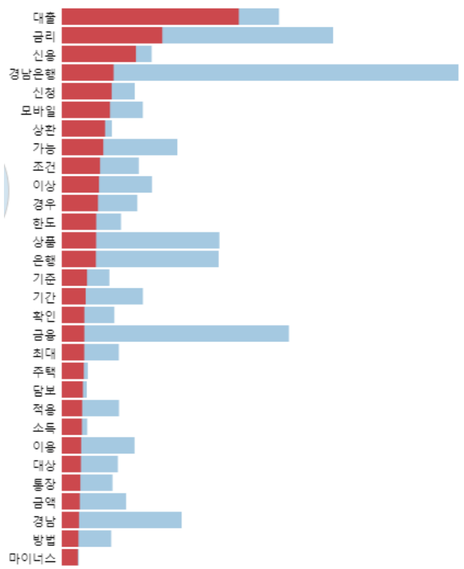
생각
가능
공유
금리
이용
적금 가입
이번
할인
서비스
혜택
해당
이벤트
예정
사용
이자
선택
추천
필요
진행
제공
금액
정기
적용
기준
부산은행
가입
기간
시간
안내
방문
최대
조건
상품
우대
이율
기본 금리
경우
정도
확인
부산
카드 발급
동백 패스
대출
카드
렉스 카드
다음
모바일
뱅킹
지원
주택
여행
고객
소개
오늘
금융
정보

경남은행 - 지방은행

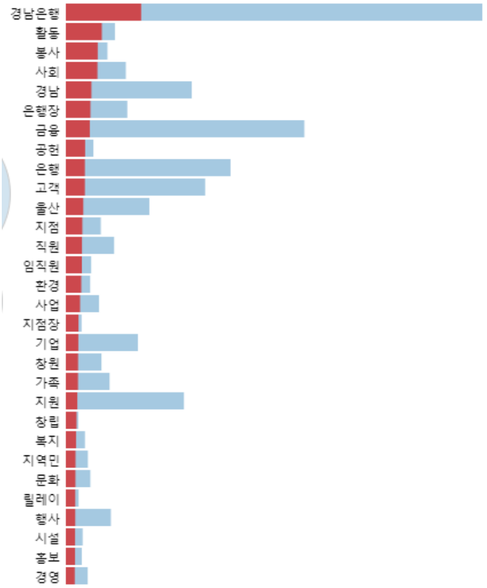
Word Cloud



LDA - Topic 1



Topic 2



Topic 3

Word Cloud

- 여러 은행에서 기본적으로 제공하는 금융 서비스에서 강조되고 있는 **우대 금리, 신용 대출**이 크게 나타남
- 최근 경남은행은 모바일 뱅킹 앱을 리뉴얼하여 고객 편의성 등을 향상시켰기에, **경남은행 모바일**이 크게 나타남
- 지방은행만의 특성을 나타낼 수 있는 요소가 보이지 않음 =>Conclusion: 낮은 성능

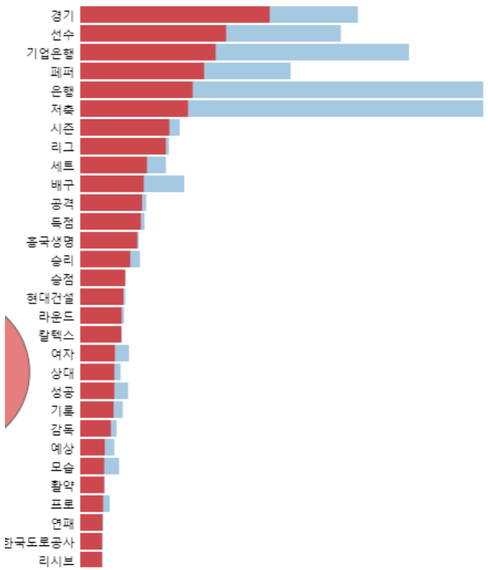
⇒Solution: LDA로 세부 결과 살펴보기

LDA

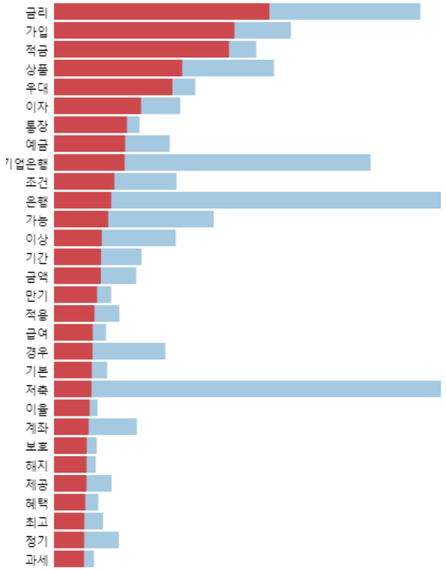
- Topic 1과 2의 경우 **금리, 가입, 적금, 대출, 신용, 모바일** 등 단어로 다른 은행처럼 금융 상품을 통해 고객 유치를 강화와 디지털 편의성을 재공함을 드러냄
- Topic 3의 경우 **울산, 지원, 금융** 등 단어로 보아 2024년 울산 특화 브랜드 '울산 경남은행' 출시로 지역 상생 이슈를 드러냄

IBK저축은행 - 저축은행

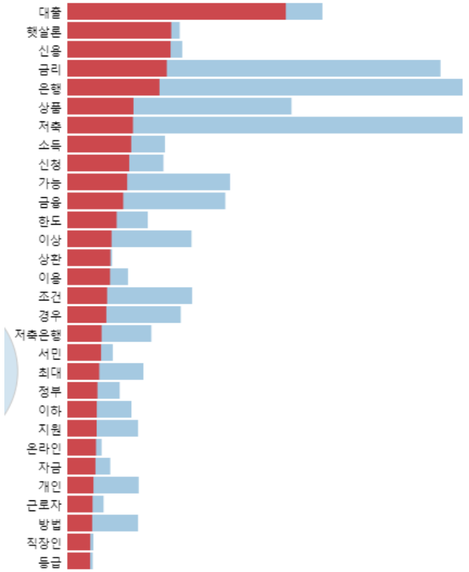
Word Cloud



LDA - Topic 1



Topic 2



Topic 3

Word Cloud

- 저축은행이 지원하는 여자배구 선수단 관련하여 부진에서 막 탈출한 기업은행, 최하위 순위 페퍼저축은행의 계속되는 부진이 주요 이슈로 나타나 선수, **페퍼저축**이 크게 나타남
- 저축은행으로서 다양한 저축 상품과 특히 청년층 대상 금융 지원을 많이 하기 때문에 **우리 금리, 종합저축, 청년 도약**이 크게 나타남

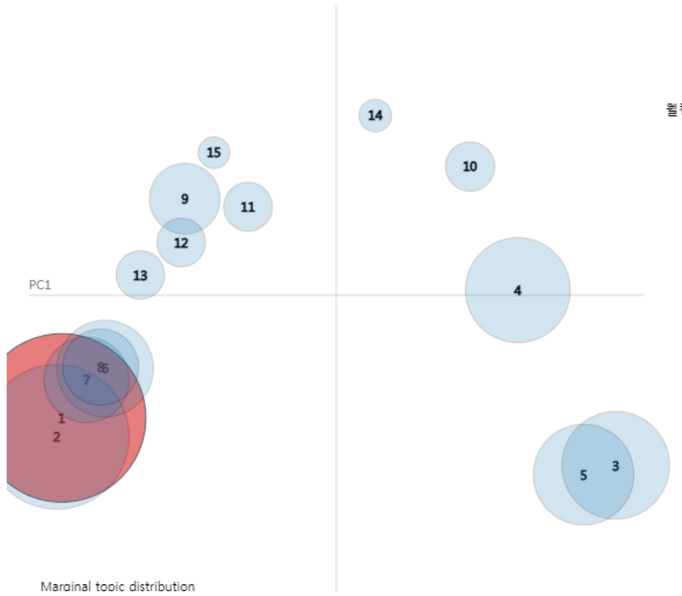
=>Conclusion: 높은 성능

LDA

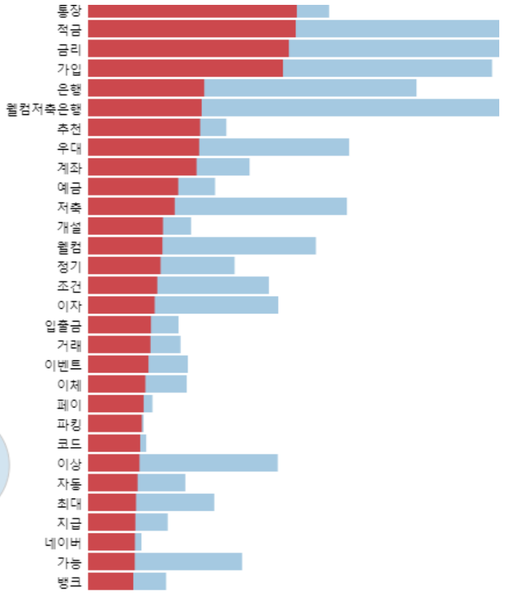
- Topic 1의 경우 **금리, 적금, 가입, 상품** 등 단어로 보아 다른 은행과 마찬가지로 기본적인 일반은행업무를 제공함을 드러냄
- Topic 2,3의 경우 **경기, 리그, 선수, 페퍼, 세트** 등 단어로 보아 저축은행이 지원하는 여자배구팀의 주요 이슈를 나타냄

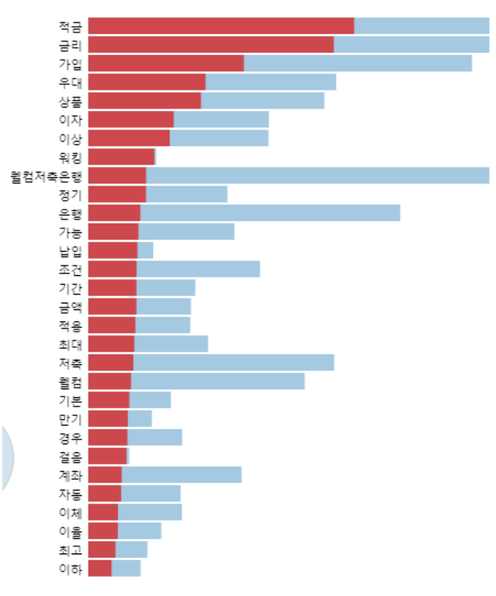
웰컴저축은행 - 저축은행

Word Cloud



LDA - Topic 1





Topic 2

Word Cloud

- 저축은행으로서 적금과 같은 다양한 저축 상품과 금융 지원을 많이하기 때문에 **우대 금리, 적금 가입, 정기 적금**이 크게 나타남
- 웰컴 저축은행은 프로 당구 협회(PBA)를 후원하고, 당구 관련 이벤트를 주최하고, 프로 당구팀을 보유하여 **팀 리그, 포스트 시즌**이 크게 나타남

=>Conclusion: 높은 성능

LDA

- Topic 1의 경우 **세트, 경기, 라운드, 팀리그** 등으로 보아 대회 개최와 같은 스포츠 마케팅을 통해 브랜드 홍보 역시 구사함을 드러냄
- Topic 2의 경우 **적금, 금리, 상품, 가입** 등으로 보아 다른 은행과 마찬가지로 기본적인 일반은행업무를 제공함을 드러냄

Word Cloud

- 시중은행

- 6개의 시중은행 전반적으로 **가능, 경우, 이용, 진행 등**이 크게 노출됨.
- 특정 상황이나 조건에 따라 달라지는 은행의 금융 상품 및 서비스의 특징을 드러냄.
- 은행의 상품 및 서비스를 실질적으로 이용할 때의 절차적인 측면도 드러냄.
  ex. '대출 **가능** 여부', '정기 적금을 해지하는 **경우**', '**이용**할 수 있는 혜택', '대출 심사 **진행**'

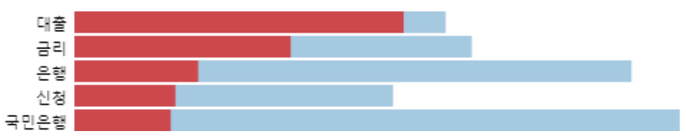
- 은행별 글씨 크기(노출 빈도)는 다르지만 전반적으로 시중은행의 주요 상품 및 서비스인 **우대 금리**와 **청년 도약**이 나타남.

- 하지만 한국씨티은행을 제외한 나머지 5개의 은행에서는 공통적인 단어들이 많고, 적금 및 예금 등 상품명이 비슷하기 때문에
은행별 특징을 나타내는 단어가 무엇인지 판별하기 어려움 ⇒Conclusion : 낮은 성능
⇒Solution: 5개 은행에 대해 LDA로 세부 결과 살펴보기

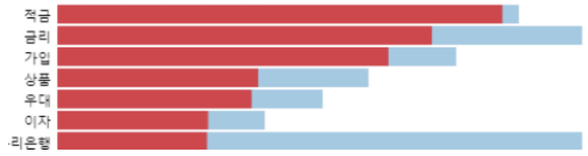
LDA

- 시중은행

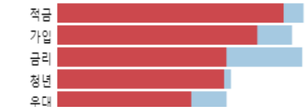
Topic 1



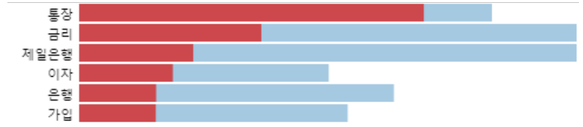
국민은행



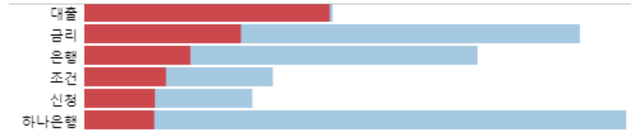
우리은행



신한은행

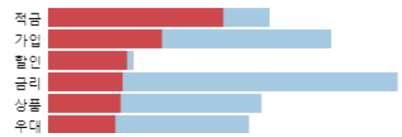


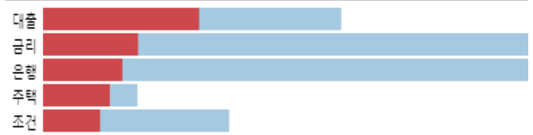
SC제일은행

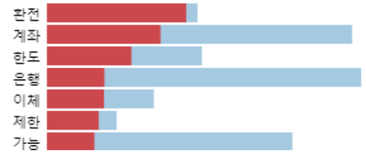


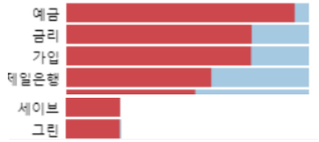
하나은행

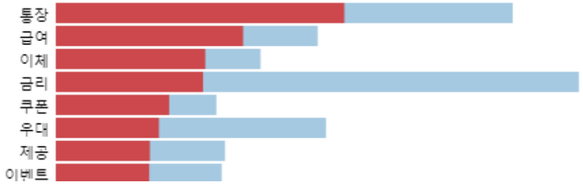
Topic 2











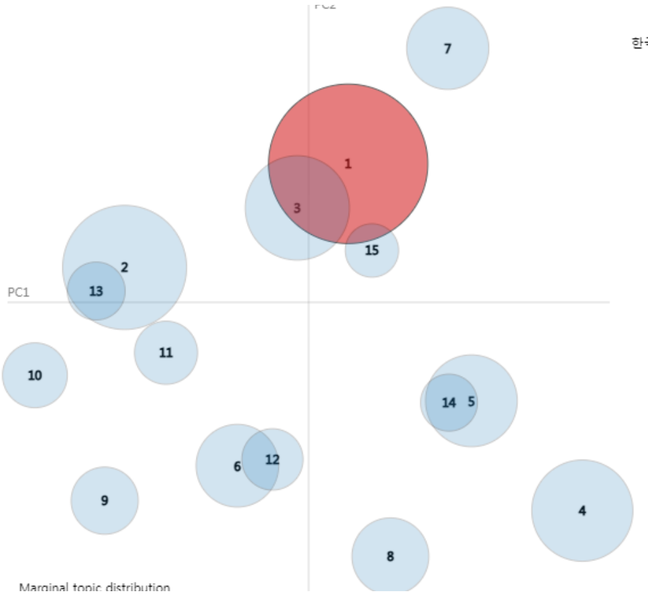
- LDA에서도 시중은행들의 토픽은 비슷하게 나타남
- 가장 비중이 높은 상위 3개의 토픽(Topic 1,2,3)은
  주로 '대출', '적금', '환전' 관련 토픽
- 다른 토픽들도 마찬가지로 주제가 비슷함
(대출, 적금, 환전, 계좌 및 카드, 이벤트, 인증서, 채용 관련 토픽)
- 토픽별 단어도 유사함

- 그래도 하나은행과 제일은행에서 은행별 특징이 드러남
- 하나은행의 Topic2의 경우 **급여, 이체**가 연관성이 높은 단어
→실제로 하나은행이 최근 매달 특별한 혜택을 받을 수 있는 급여이체 통장을 출시
→하나은행만의 특징이 잘 나타남

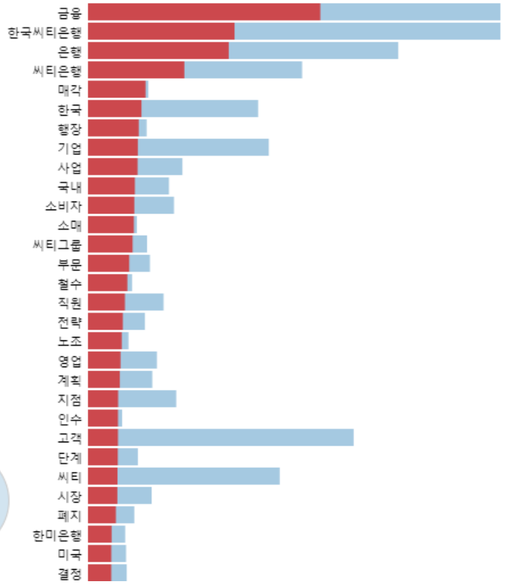
- 제일은행의 Topic2의 경우 토픽과 가장 연관성이 높은 단어는 **예금**이고
이 토픽에서 가장 중요도가 높은 단어는 **세이브**와 **그린**
→실제로 제일은행의 그린세이브 예금이 제일은행의 주요 상품이며,
다른 은행보다 금리가 높은 경쟁력 있는 상품
→제일은행만의 특징이 잘 나타남

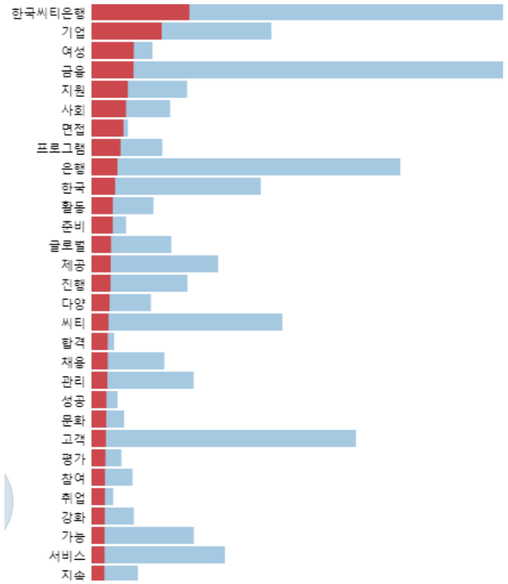
한국씨티은행 - 시중은행

Word Cloud



LDA - Topic 1





Topic 2

Word Cloud

- 은행의 금융 상품과 서비스의 특징을 나타내는 **가능, 경우, 이용, 진행**과 같은 단어들이 크게 노출됨
- 다른 시중은행에서 볼 수 없었던 **기업, 소비자 금융, 단계 폐지, 한국, 직원, 노조** 등의 단어들도 노출됨
- 소비자 금융의 분리 매각이 불발되면서 한국에서는 소매 금융을 단계적 폐지하고 기업 금융 부문만 투자를 지속하기로 하였고, 폐지에 따라 직원들의 퇴직 사항에 대해 노조와 협의를 벌인 이슈가 잘 반영됨.

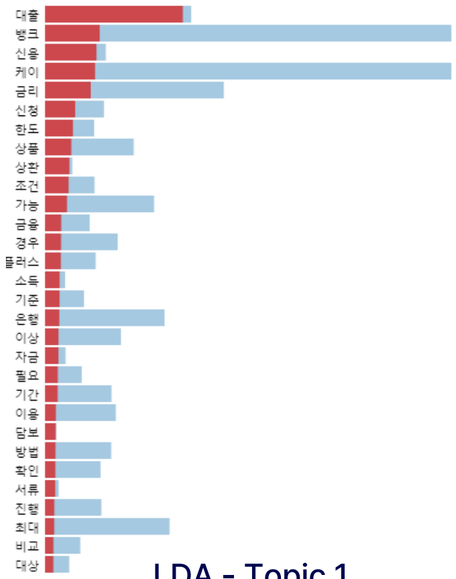
⇒Conclusion: 높은 성능

LDA

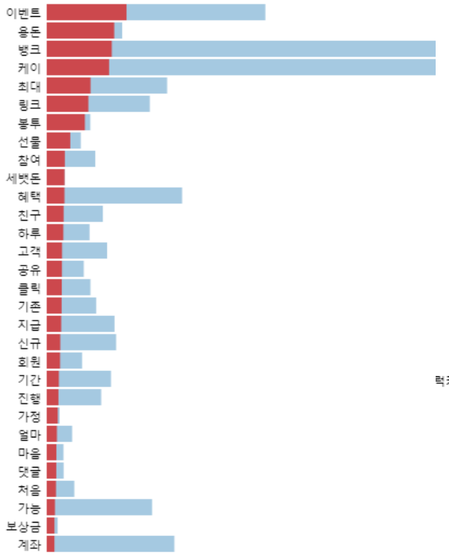
- Topic 1의 경우 **금융, 매각, 기업, 한국 등** 단어로 보아 한국에서 소매 금융 부분의 매각이 불발된 이슈를 드러냄
- Topic 2의 경우 **여성, 지원, 면접 등**의 단어로 보아 기업의 여성 인재 지원 및 채용 프로그램이 활발히 진행됨을 시사

케이뱅크 - 인터넷전문은행

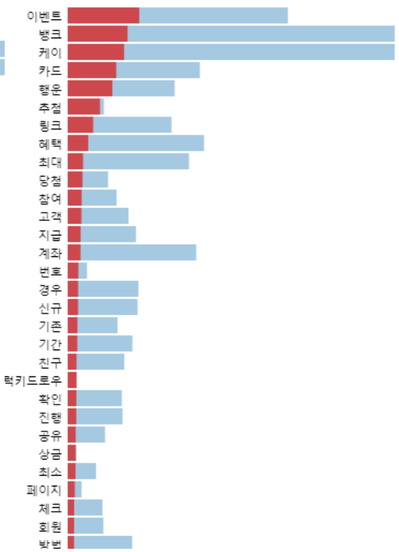
Word Cloud



LDA - Topic 1



Topic 4



Topic 5

Word Cloud

- 인터넷 뱅킹인 케이 뱅크가 제공하는 서비스인 **생활 통장, 용돈 봉투, 행운 카드** 등이 크게 노출
- 은행의 주요 기능인 계좌 개설, 신용 대출과 은행 서비스의 특징을 나타내는 **가능, 사용** 등이 잘 나타남

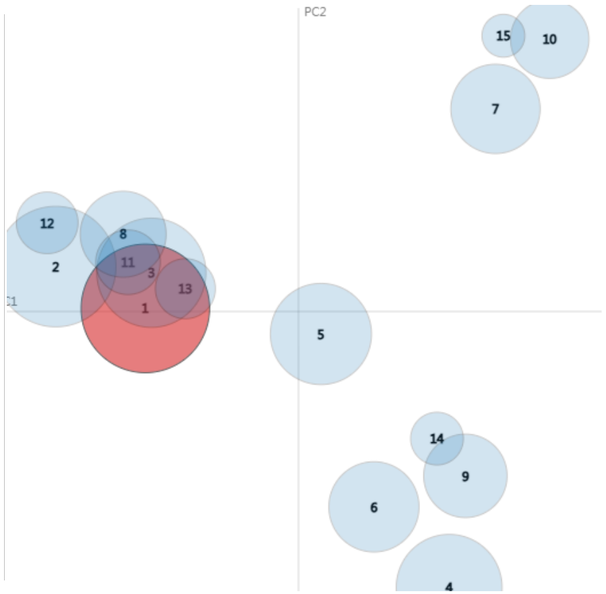
⇒Conclusion: 높은 성능

LDA

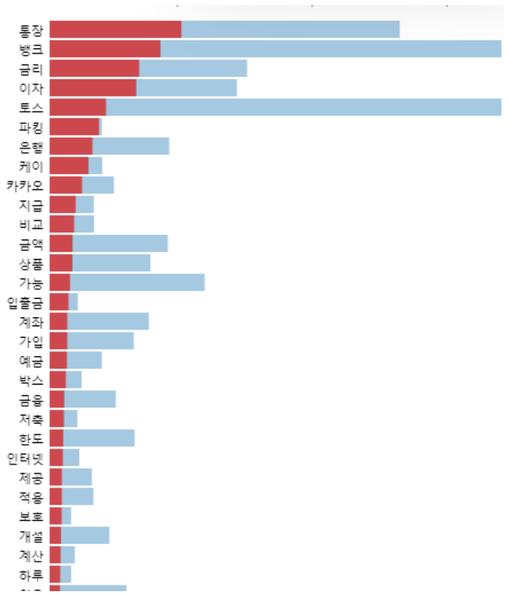
- Topic 1의 경우 **대출, 신용, 금리, 상품** 등 단어로 보아 은행의 핵심적인 기능과 서비스를 인터넷 뱅킹인 케이 뱅크에서도 잘 제공됨을 시사
- Topic 4와 5의 경우 **이벤트, 용돈, 행운, 링크** 등 단어로 보아 케이 뱅크만의 이벤트를 제공하고 있음을 시사

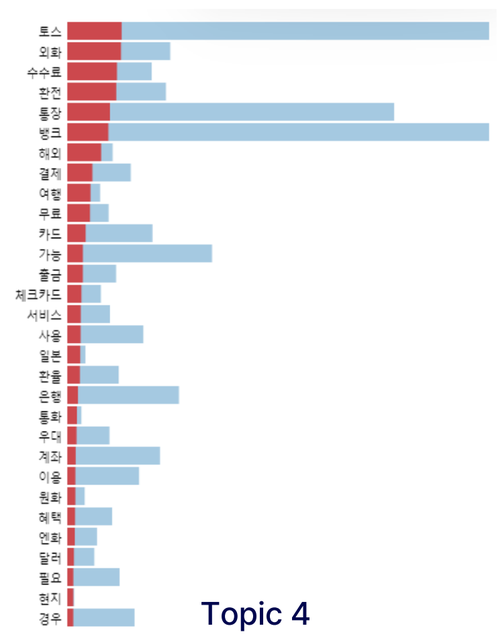
토스뱅크 - 인터넷전문은행

Word Cloud



LDA - Topic 1





Topic 4

Word Cloud

- 국내 금융사 최초로 환전 수수료가 무료인 외화 통장을 출시하였기에 **외화 통장, 뱅크 외화, 환전 수수료**가 크게 노출됨
- 은행에서 제공하는 서비스와 관련된 **파킹 통장, 이용, 경우, 상품** 등이 잘 나타남

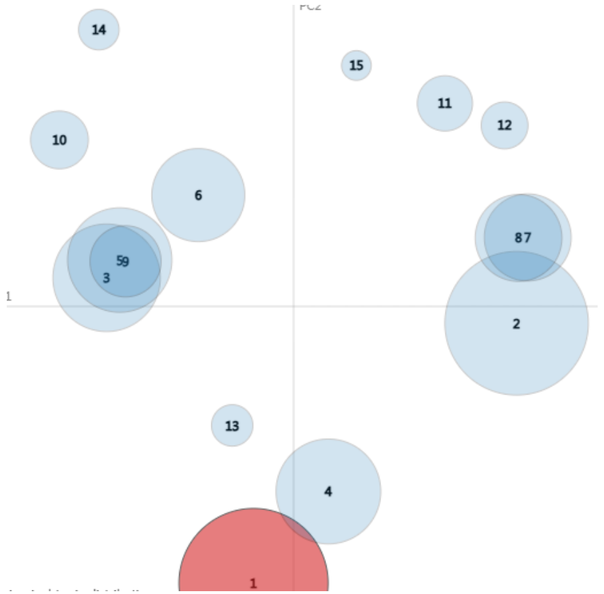
⇒Conclusion: 높은 성능

LDA

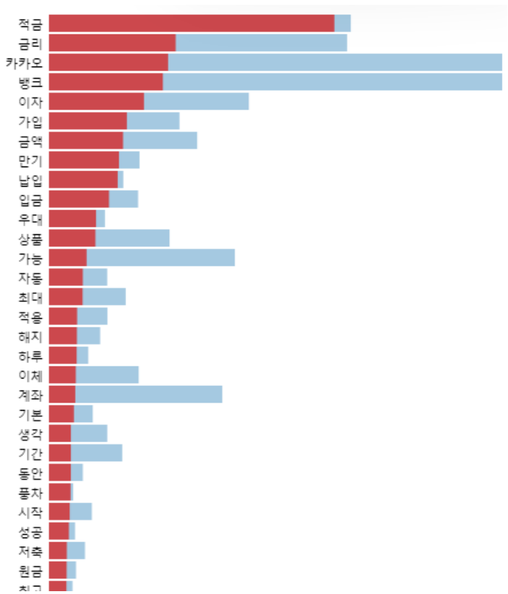
- Topic 1의 경우 **통장, 금리, 이자** 등의 단어로 보아 은행의 핵심적인 기능과 서비스를 인터넷 뱅킹인 토스 뱅크에서도 잘 제공됨을 시사
- Topic 4의 경우 **외화, 수수료, 환전, 해외** 등 단어로 보아 환전 수수료가 무료인 외화 통장 서비스가 토스 뱅크의 주력 서비스임을 시사

카카오뱅크 - 인터넷전문은행

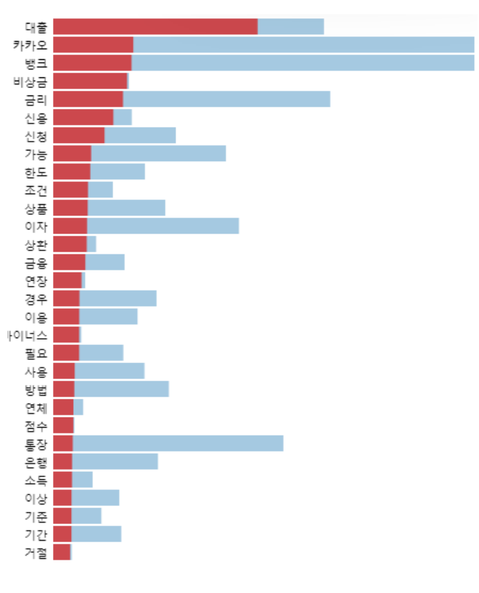
Word Cloud



LDA - Topic 1



Topic 2



Word Cloud

- 은행에서 제공하는 서비스와 관련된 **통장 사본, 경우, 이용, 상품** 등이 노출됨
- 하지만 인터넷뱅킹이나 카카오뱅크만의 특징을 나타내는 단어들이 잘 안 보임 =>Conclusion: 낮은 성능

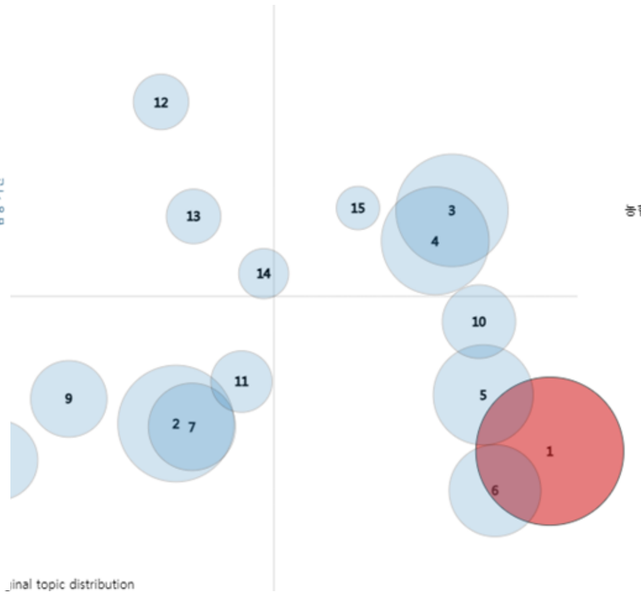
⇒Solution: LDA로 세부 결과 살펴보기

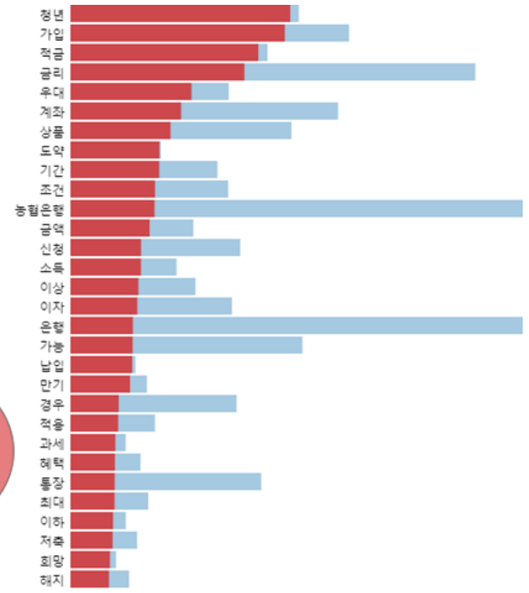
LDA

- Topic 1의 경우 **적금, 금리, 이자, 가입, 만기** 등 단어로 보아 카카오뱅크에서 적금 관련 서비스의 수요와 관심도가 높음을 시사
- Topic 2의 경우 **대출, 비상금, 금리, 신용** 등의 단어로 보아 비상금 대출 서비스의 수요와 관심도가 높음을 시사

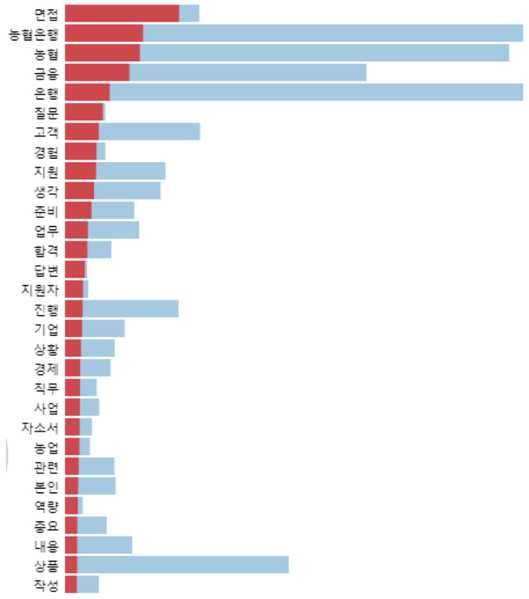
농협은행 - 특수 은행

Word Cloud





LDA - Topic 1



Topic 2

Word Cloud

- 시중은행의 워드클라우드 결과와 비슷하게 은행의 상품 및 서비스의 특징을 드러내는 **경우, 가능, 필요, 진행, 청년도약, 우대금리** 등이 크게 노출
- 하지만 농협은행만의 특징을 정의하긴 어려움 =>**Conclusion:** 낮은 성능

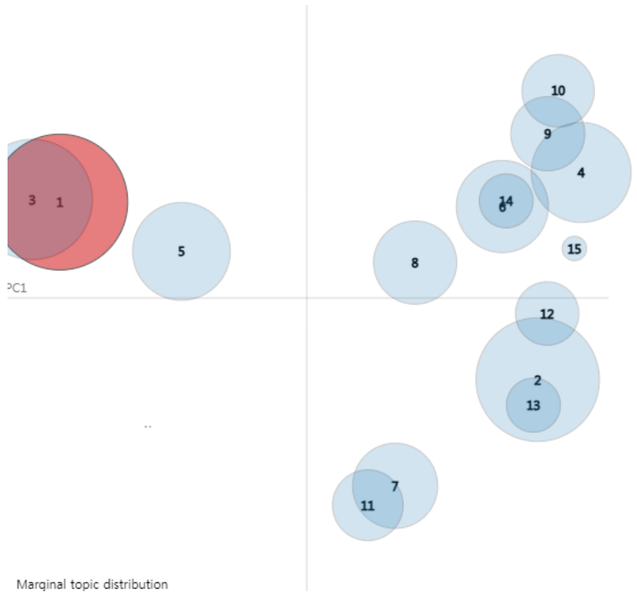
⇒**Solution:** LDA로 세부 결과 살펴보기

LDA

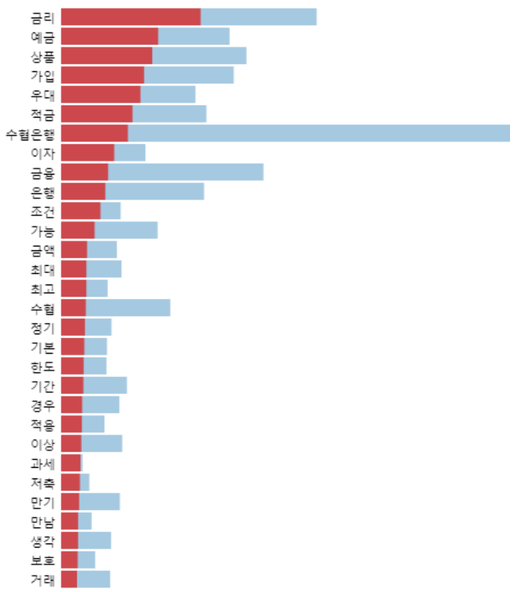
- Topic 1의 경우 **청년, 상품, 도약** 등 단어로 보아 농협은행의 청년도약계좌 상품의 수요와 관심도가 높음을 시사
- Topic 2의 경우 **면접, 질문, 고객, 경험, 생각** 등의 단어로 보아 농협은행의 채용에 대한 관심도가 높음을 시사

수협은행 - 특수 은행

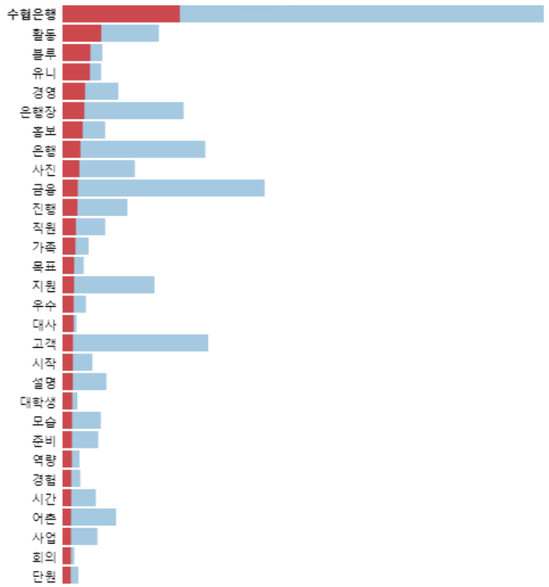
Word Cloud



LDA - Topic 1



Topic 2



Word Cloud

- 은행의 상품 및 서비스의 특징을 드러내는 **상품, 가능** 등이 나타남
- 수협의 대학생 서포터즈 활동인 유니 블루에 관한 홍보와 후기글이 많고, 강신숙 은행장이 1분기 높은 순익성장률을 달성한 이슈가 반영되어 **유니 블루, 은행장** 등이 크게 노출됨

⇒Conclusion: 높은 성능

LDA

- Topic 1의 경우 **금리, 예금, 상품** 등 단어로 보아 수협의 예금 상품이 수요가 많은 주력 상품임을 시사
- Topic 2의 경우 **활동, 블루, 유니, 경영, 은행장** 등의 단어로 보아 대학생 서포터즈 블루 유니와 은행장의 경영 관련 관심도가 높음을 시사