

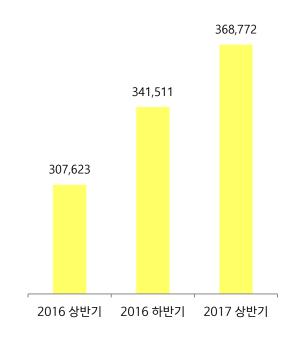
목차

- 1. 프로젝트 개요
- 2. 프로젝트 과정
- 3. 프로젝트 내용
- 4. 결론 및 기대 효과
- 5. 한계점 및 향후 계획

1) 개발 배경: 국내 온라인 쇼핑몰 시장 상황

전체 시장 규모

온라인 쇼핑 매출액 현황(억원)



2017년 상반기 약 36조 8,700억원, 지속적인 성장세

온라인 쇼핑 시장 주요 이슈



네이버 쇼핑의 성장 본격 시장 진입 3년 만에 3위



인공지능 Al기술 활용한 시스템 구축

1) 개발 배경: 스마트스토어 현 상황분석

네이버 쇼핑 플랫폼
"샵N" –"스토어팜" – "스마트스토어"



무료 온라인 창업 쇼핑몰 지원 플랫폼, 입점·판매수수료를 없애 비용 절감에 민감한 중소상공인 공략 현재 10만명의 판매자가 스마트스토어를 통해 온라인 사업을 새롭게 시작, 성장 이중 1만명 이상이 연매출 1억원을 넘어섰다. 올 3월 기준 중소상공인들의 온라인 매출 중 90%가량은 네이버의 자체 인터넷 쇼핑몰인 스마트스토어(스토어팜)를 거친 것으로 나타났다.

2) 프로젝트의 주제 및 기획의도

"네이버 스마트스토어 구매평 분석을 통한 마케팅 인사이트 도출" - 크롤링/데이터 시각화를 바탕으로

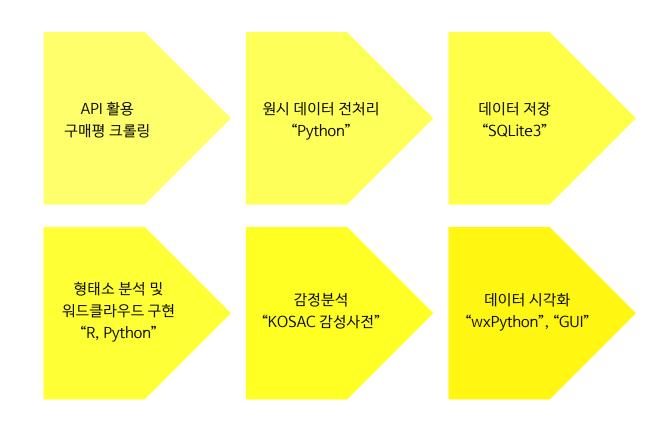


소비자 정보의 양과 질 측면에서 중요한 "구매평" 웹크롤링, 데이터 분석 및 시각화 등을 적용 중소상공인들의 "구매자 needs" 파악에 도움

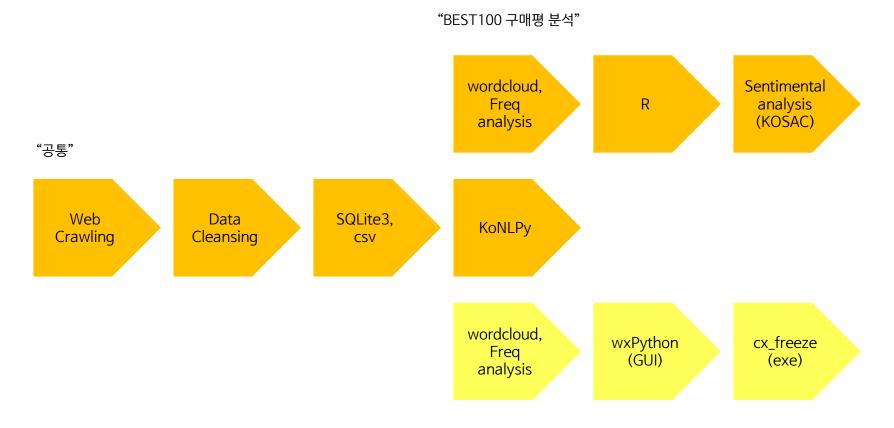
"카테고리 별 BEST100 구매평 분석", "제품 구매평 분석기 App 개발"을 목표로 프로젝트 수행

2. 프로젝트 과정

1) 개발 과정 요약

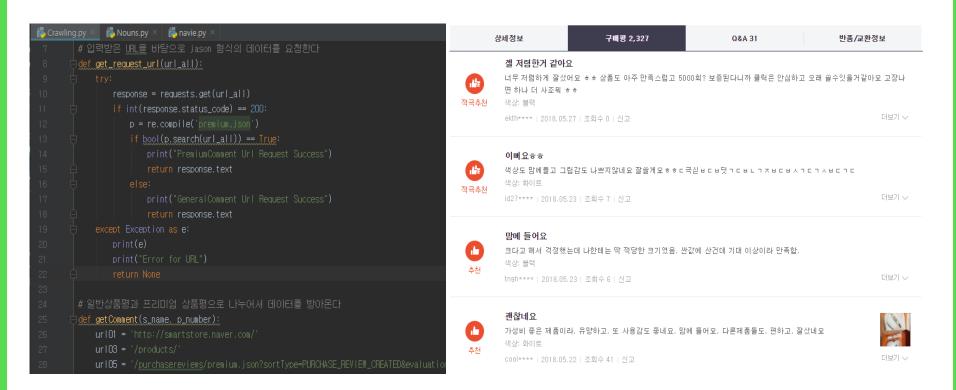


프로젝트 세부절차



"제품 구매평 분석기 App 개발"

1) 데이터 수집:크롤링



네이버 스토어팜 페이지에서 상품 구매자가 남긴 "구매평(프리미엄/일반) 크롤링"

1) 데이터 수집:크롤링



천","gradelcon":"icon_premium_verygood","contentViewUrl":"http://order.pay.naver.com/purchasePeview/412505838/premiumPe iew/view/shopn","contentType":"TEXT","best":false,"contents":null,"contentsSummary":"디렉터파이님의 착한 토너에서 보고

샀어요 ㅎㅎ 생각보다 굉장히 가볍고 순한 느낌의 토너에요

만족도

"gradeText":"추천"

댓글 내용

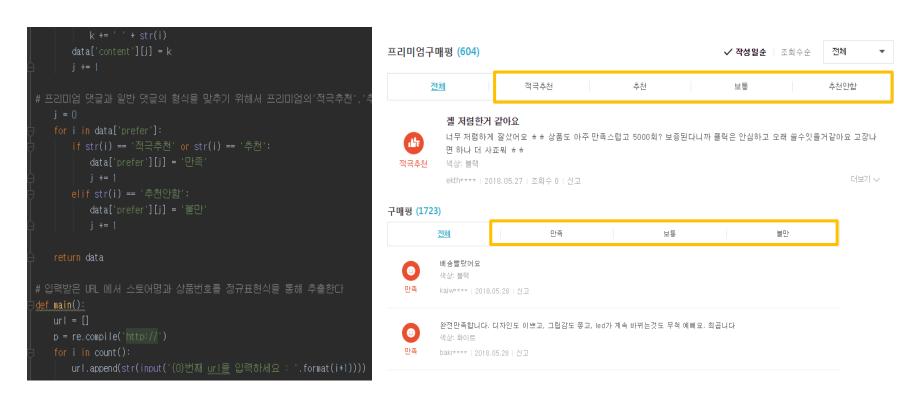
"contentsSummary": 무난하게 좋은 토너예요. 스킨팩하기도 좋고 향도 좋아요. 가격도 비싸지 않아서 팍팍 쓰고 있어여

날짜

"createdDate": 2018-05-04 14:15:47"

"Json 형식"으로 데이터를 요청하여 원하는 정보를 수집

2) 데이터 정제



만족, 보통, 불만의 "3점 척도로 통일" 시키기 위해 리코딩

2) 데이터 정제

```
data = pd.DataFrame(result, columns=('type', 'prefer', 'content', 'date'))

j = 0
p = re.compile('\w+')

for item in data['content']:
    b = item.replace('\n', '')
    c = p.findall(b)
    k = ''
    for i in c:
        k += ' ' + str(i)
    data['content'][j] = k
    j += 1
```

-> R에서 특수문자 중 일부가 데이터프레임을 망가뜨려 형태소 분석에 문제가 발생 : 낮은 분석 정확도

"정규표현식으로 특수문자 제거"

3) 데이터 저장



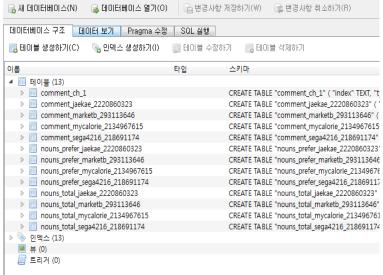
데이터 프레임 CSV로 데이터 저장 (타입, 만족도, 내용, 날짜)



명사 추출 및 감정 분석 백데이터

3) 데이터 저장





특정 스토어팜의 제품 댓글 내용
"전체 명사 및 해당 명사의 빈도 수"
"만족도별 명사 및 해당 명사의 빈도 수"



Python 내장 DBMS, SQLite3 활용 각각의 Dataframe을 table로 저장

4) 명사 추출 : KoNLPy 패키지 Twitter

```
beauty_nouns.csv - 메모장
nlp = Twitter()
                                                                 편집(E) 서식(O) 보기(V) 도움말(H)
                                                          # 만족,보통,불만 카테고리로 나누어서 문장의 명사 빈도수를 추출한다
a = ['만족', '보통', '불만']
for i in a:
   _data = data[data['prefer'] == str(i)]
      except Exception as e:
         print(e)
      a.append(k[0])
      b.append(k[1])
```

명사 빈도수 분석 후, 가장 많이 나온 단어부터 정렬한 데이터프레임 생성하여 csv로 저장

5) R을 활용한 데이터 전처리 & 시각화

#csv파일 → 데이터프레임 저장:

baby_nouns (- read.csv("baby_nouns.csv", header = T, stringsAsFactors = F, fileEncoding = "UTF-8")

##만족도별 데이터셋 분류:

baby_good (- baby_nouns[baby_nouns\$만족도 == "만족",]

###단어컬럼에서 두 글자 이상인 행만 저장 :

baby_nouns_two <- filter(baby_nouns, nchar(단어) >= 2)

####공통빈출단어 배제하기위한 변수 리코딩:

baby_good (- baby_good[baby_good\$단어 != "배송" & baby_good\$단어 != "너무" & baby_good\$단어 != "가격" & baby_good\$단어 != "구매" & baby_good\$단어 != "진짜" & baby_good\$단어 != "아직" & baby_good\$단어 != "생각" & baby_good\$단어 != "제품" & baby_good\$단어 != "상품" & baby_good\$단어 != "아주" & baby_good\$단어 != "완전",]

5) R을 활용한 데이터 전처리 & 시각화



"빈도분석 막대그래프(ggplot2)"

top30 <- comment_nouns_two %〉% arrange (desc(빈도수합계)) %〉% # 빈도수 상위30개 내림차순 정렬

order (- arrange(top30, 빈도수합계)\$단어 # 빈도수 상위30개 저장

ggplot(data=top30, aes(x=단어, y=빈도수합계, fill=단어)) + ylim(0,50000) + geom_col() + coord_flip() + scale_x_discrete(limit = order) + geom_text(aes(label=빈도수합계), hjust=-0.3)



"워드클라우드(wordcloud)"

```
pal (- brewer.pal(9,"Set1")
# 만족:Set1, 보통:Blues, 불만:OrRd # 색깔변수 지정
set.seed(1234) # 난수고정

wordcloud(words = baby_good$단어,
    freq = baby_good$빈도수,
    max.word = 100,
    random.order = F,
    random.color = T,
    rot.per = .1,
    scale = c(5, 0.5),
    colors = pal,
    )
```

6) 감정분석 - KOSAC 사전을 이용한 학습 데이터 생성

Home Corpus Lexicon Publications	Korean Sentiment Lexicon Release Page The sentiment lexicon can be used solely for academic and research purposes. o Agreement Form Download Send us the agreement to kosac.snu@gmail.com
	Sentiment Lexicon Version 1.0 o README o Annotation Scheme (PACLIC26) o Download

서울대학교 컴퓨터 언어학과에서 만든 "KOSAC 사전"을 이용하여 긍정(POS), 부정(NEG), 중립(NEUT)의 단어를 학습 데이터로 생성

8622	시대/NNG;정신/NNG	NEG	1
8623	시대/NNG;정신/NNG;에/JKB	NEG	1
8624	시도/NNG	POS	1
8625	시도/NNG;하/XSV	POS	1
8626	시련/NNG	NEG	1
8627	시멘트/NNG	NEG	1
8628	시멘트/NNG;길/NNG	NEG	1
8629	시멘트/NNG;덩어리/NNG	NEG	1
8630	시사/NNG	NEUT	1
8631	시사/NNG;하/XSV	NEUT	1
8632	시선/NNG	POS	1
8633	시선/NNG;이/JKS	POS	1
8634	시선/NNG;이/JKS;두드러지/	POS	1
8635	시설/NNG	POS	1
8636	시설/NNG;을/JKO	POS	1
8637	시설/NNG;을/JKO;갖추/VV	POS	1
8638	시약/NNG	NEG	1
8639	시약/NNG;을/JKO	NEG	1
8640	시인/NNG	NEUT	1
8641	시인/NNG;의/JKG	NEUT	1

6) 감정분석 - NaiveBayseClassifier 를 이용한 데이터 학습 및 분석

```
🦰 Nouns.py 🗡 🛮 🖰 navie.py 🗡
|def word_feats(words):
positive_features = [(word_feats(pos), 'pos') for pos in positive_vocab]
train_set = negative_features + positive_features + neutral_features
classifier = NaiveBayesClassifier.train(train_set)
# 예측하기 위해 데이터셋 준비(_data.csv 앞에 글자만 작성. beauty를 예시로 써둠)
```

KOSAC의 긍정, 부정, 중립 단어 SET를 "NaiveBayseClassifier를 통해 학습"시킨 후 네이버 댓글을 분석

감정분석 - KOSAC 사전 수정



「**부정확**한 분석 결과 `

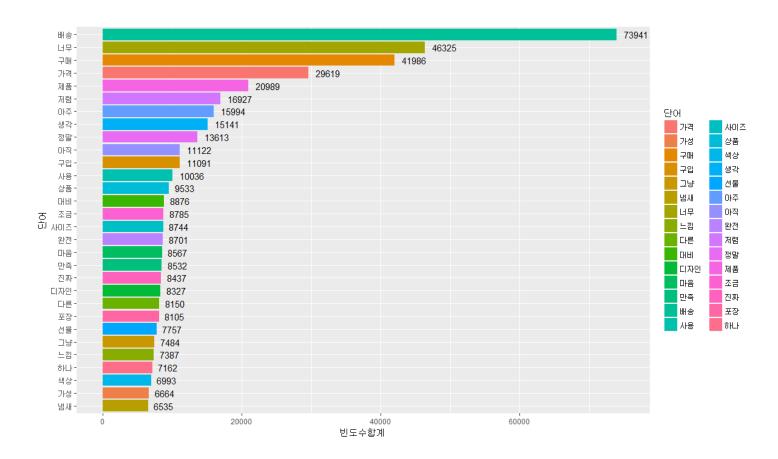
: '잘받았습니다', '잘받았어요', '배송빨라요' 등을 부정적인 단어로 인식 KOSAC 사전에 '잘', '잘받다' '배송빨라' 등을 긍정의 단어로 추가 하여 "**개학습**"

6) 감정분석 - 최종 결과

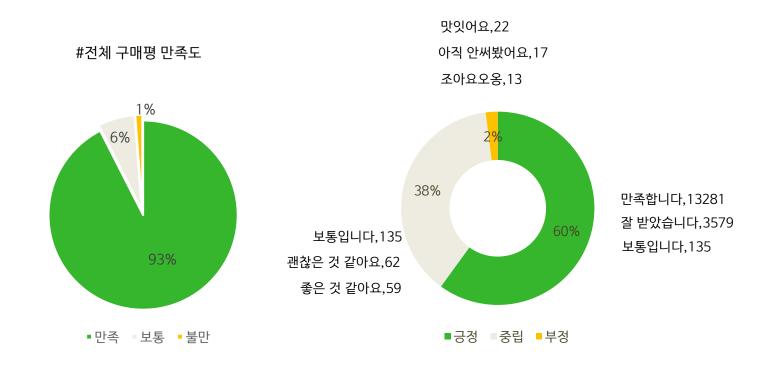
digital			fashion_a			fashion_c		
긍정	중립	부정	긍정	중립	부정	긍정	중립	부정
66%	33%	1%	60%	39%	1%	62%	36%	2%
만족합니다,1587	보통입니다,14	이쿠조쿠요,3	만족합니다,831	보통입니다,13	무난하네용,3	만족합니다,585	보통입니다,10	사진 그대로입니다,3
잘받았습니다,449	괜찮은거같아요,13	그립감 좋네요,3	좋아요,244	그냥저냥 괜찮아요,6	머 가격대로 입니다,3	좋아요,139	보통이예요,5	강추드립니다,3
감사합니다,427	좋은것같아요,7	아직안써봄,3	잘받았습니다,191	그냥그래여,6	사진과 똑같아요,3	잘받았습니다,118	입을만해요,5	팔왜케 짧아요,2
좋아요,378	괜찮은것 같아요,7	감사함니다,2	감사합니다,113	심플하고 예뻐요,6	별로입니다,2	잘받았어요,98	괜찮아용,4	속바지로 짱,2
좋습니다,256	가성비 갑,6	너무 약함,2	잘받았어요,113	다들 이쁘다고 그래요 :	고냥저냥 πㅋㅋㅋ,2	이뻐요,91	괜찮은거 같아요,4	그냥그냥그냥,2
잘 받았습니다,252	가성비갑입니다,6	잘 들립니다,2	예뻐요,107	심플하니 이뻐요,4	딱 그가격입니다,2	맘에들어요,83	그럭저럭 괜찮아요,3	생각보다 두꺼워요,2
아주좋아요,179	가성비 짱입니다,5	별로입니다,2	맘에들어요,93	괜찮은거 같아요,4	그냥저냥,2	예뻐요,56	재구매했어요,3	냉장고바지같아요,2
0 빠른배송 감사합니다,1	가성비갑이네요,4	굳입니다,2	이뻐요,82	좋아요 다만 사이즈 선	조아요오용,2	괜찮아요,47	믿고사는 마약잠옷,3	조아요오몽,2
잘받았어요,137	좋은것 같아요,4	구우우우욷,2	잘 받았습니다,70	매번 편의점에서 구매하	그냥그저그럼,2	감사합니다,46	그냥그래여,3	그냥그냥,2
빠른 배송 감사합니다,:	좋은 것 같아요,3	기대이상입니다,2	좋습니다,64	심플하니예뻐요,4	편할꺼 같아요,2	좋아요좋아요,44	그냥 괜찮아요,3	그냥 기본티,2
배송빨라요,120	괜찮은 것 같습니다,3	그립감 좋아요,2	배송빨라요,54	괜찮은거같아요,4	원터치라 착용하기 편하	배송빨라요,43	편하게 입을것 같아요,3	시원한 소재입니다,2
아주 좋아요,101	굿뜨,3	삼각대 개꿀띠,2	맘에 들어요,52	배송이 굉장히 빨라요	사진그대로입니다,1	잘 받았습니다,39	안녕 하세요 잘받앗습니	시원할꺼같아요,1
빠른배송감사합니다,94	와이파이 짱짱합니다,3	써보니까 좋네요,2	편하고 좋아요,49	쫀쫀하고 면이 톡톡해요	O 시 ㄷ ㅇ ㄷ 시 ㄷ 느 ㄷ 듣-	그냥그래요,38	11개월아기 딱맞을것같	사진과동일,1

사전 수정 후 분류 정확도가 매우 향상 되었으나 [**맞춤법이 틀린 경우**, **부정적인 단어로 인식**]

1) 네이버 스토어팜 BEST100 구매평 381,287건 (두자이상의 단어) - 빈도분석



1) 네이버 스토어팜 BEST100 구매평 381,287건 (두자이상의단어) - 빈도/감정분석



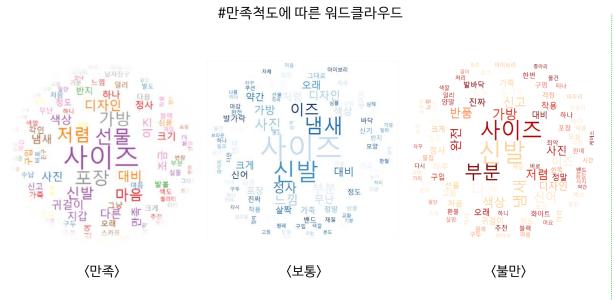
2) 패션의류: 빈도/감정분석(18,798건)



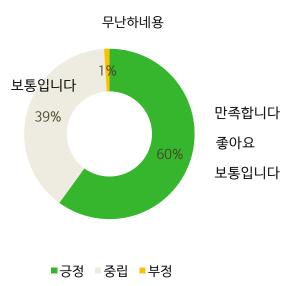
[인기상품은 "여름의상", 주요 요인은 "색상", "재질", "사이즈"]



3) 패션잡화: 빈도/감정분석(30,369건)



[인기상품은 "가방","신발","귀걸이" 등 선물, 주요 요인은 "사이즈","가격","냄새"]



4) 화장품 및 미용 : 빈도/감정분석(36,526건)



[인기상품은 "스킨케어", "메이크업 제품", 주요 요인은 "가격", "성능"]



5) 디지털 및 가전: 빈도/감정분석(44,854건)



[인기상품은 "공유기", "휴대폰 케이스", "스피커", 주요 요인은 "가격", "디자인", "성능"]



6) 가구 및 인테리어: 빈도/감정분석(57,304건)



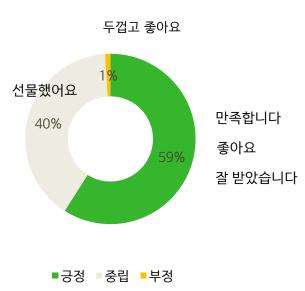
[인기상품은 "조립식 가구","의자", 주요 요인은 "색상","가격","냄새"]



7) 출산 및 육아: 빈도/감정분석(44,661건)



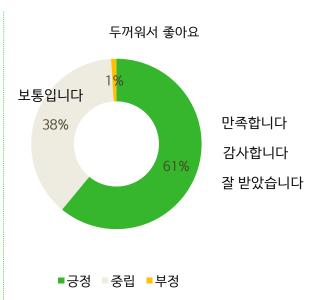
[인기상품은 "아기용 물티슈","이유식","유모차", 주요 요인은 "편리함","가격"]



8) 스포츠 및 레저 : 빈도/감정분석(33,442건)



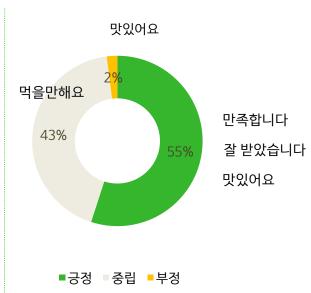
[인기상품은 "운동매트","밴드", 주요 요인은 "가격", "사이즈", "성능"]



9) 식품: 빈도/감정분석(53,012건)



[인기상품은 "선물용", "다이어트 식품", 주요 요인은 "가격", "포장", "식감"]

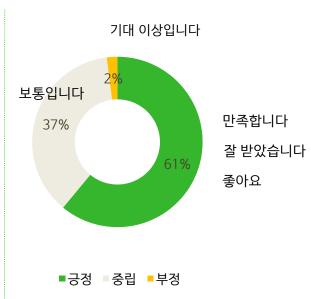


10) 생활용품: 빈도/감정분석(62,231건)

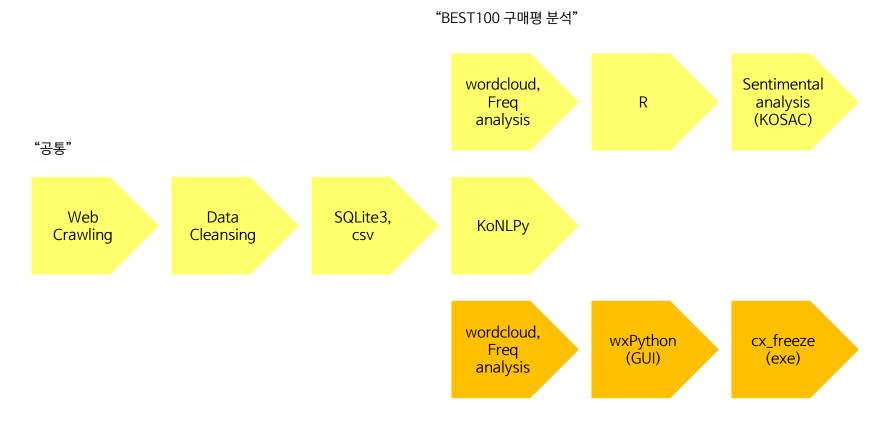
#만족척도에 따른 워드클라우드



[인기상품은 "충전기","애견용품", 주요 요인은 "가격", "디자인", "성능"]



프로젝트 세부절차



"제품 구매평 분석기 App 개발"

1) 제품 구매평 분석기 App 개발



"GUI기반의 프로그램 개발"

2) 빈도 분석 및 워드클라우드

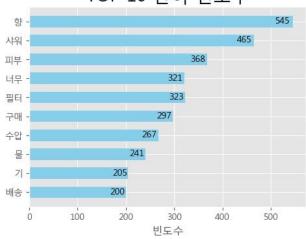
```
def createChart(df list, s name, p number):
       nouns = list(i['단어'].loc[0:9])
       nouns.reverse()
       freq = list(i['빈도수'].loc[0:9])
       freq.reverse()
       plt.figure(figsize=(5.5, 4))
       rects = plt.barh(np.arange(10), freq, align='center
       plt.yticks(np.arange(10), nouns)
       for k, rect in enumerate(rects):
           plt.text(0.99 * rect.get_width(), rect.get y(
                    str(freq[k]), ha='right', va='center
       plt.xlabel('빈도수')
       plt.title('TOP 10 단어 빈도수', size=20)
           fig = plt.gcf()
           fig.savefig('total_bar_{0}_{1}.jpg'.format(s_
           fig = plt.gcf()
           fig.savefig('good bar {0} {1}.jpg'.format(s na
           fig = plt.gcf()
           fig.savefig('moderate_bar_{0}_{1}.jpg'.format(
```

〈Wordcloud, PIL 패키지〉



〈Matplotlib.pyplot 패키지〉

TOP 10 단어 빈도수



3) Py파일 모듈화 및 연결

```
manyproduct = {'klairs':'119945641'}
for oneproduct in manyproduct :
    s_name = oneproduct
   p number = manyproduct[s name]
    url01 = 'http://smartstore.naver.com/'
   url03 = '/products/'
   url05 = '/purchasereviews/premium.json?sortType=PURCHASE RE
   url all = url01 + s name + url03 + p number + url05
    source_code = requests.get(url_all)
    plain text = source code.text
   mydatas = json.loads(plain text)
    list_comment = []
    for comment in mydatas['htReturnValue']['pagedResult']['con
       list comment.append(comment['contentsSummary'])
        list comment.append(comment['createdDate'])
    result = []
    _result = []
        result.append(item)
```



```
lef get request url(url all):
       response = requests.get(url all)
       if int(response.status code) == 200:
           if bool(p.search(url all)) == True:
               return response.text
               return response.text
       print(e)
lef getComment(s name, p number):
   url01 = 'http://smartstore.naver.com/'
   url03 = '/products/'
   url05 = '/purchasereviews/premium.json?sortType=PURCHAS
   url all1 = url01 + s name + url03 + p number + url05
   url01 = 'http://smartstore.naver.com/'
   url03 = '/products/'
   url05 = '/purchasereviews/general.json?satisfactionGrad
   url all2 = url01 + s name + url03 + p number + url05
   plain text1 = get request url(url all1)
```

4) 시각화(GUI)



mainGUI.py : 메인 화면(댓글)

```
mport wx
mport wx.xrc
mport sqlite3
import crawlingFinal
mport nounExtract
lass MyFrame1(wx.Frame):
      wx.Frame.__init__(self, parent, id=wx.ID_ANY, title='네이버 스토어팜 소비자 변
                        size=wx.Size(1300, 700), style=wx.DEFAULT_FRAME_STYLE | wx.
       self.SetSizeHintsSz(wx.DefaultSize, wx.DefaultSize)
      bSizer14 = wx.BoxSizer(wx.VERTICAL)
      self.m_panel4 = wx.Panel(self, wx.ID_ANY, wx.DefaultPosition, wx.DefaultSize
      bSizer15 = wx.BoxSizer(wx.HORIZONTAL)
      self.m_staticText10 = wx.StaticText(self.m_pane14, wx.ID_ANY, u"URL :", wx.De
      self.m_staticText10.Wrap(-1)
      bSizer15.Add(self.m_staticText10, 0, wx.ALL, 5)
      self.txtUrl = wx.TextCtrl(self.m_panel4, wx.ID_ANY, wx.EmptyString, wx.Defaul
      hSizer15 Add(self txtllrl 1 wx All 5)
```

chartGUI: 차트 화면(워드클라우드, 빈도 차트)

```
import wx
class MyDialog(wx.Dialog):
  __init__(self, parent, id, title, s_name, p_number);
       text.SetFont( wx.Font( 26, 72, 90, 90, False, "HV수평선M")))
       panel = wx.Panel(self, -1, (20, 150), (550, 400), style=wx.SUNKEN_BORDER)
       self.picture = wx.StaticBitmap(panel)
       panel.SetBackgroundColour(wx.WHITE)
       pane12 = wx.Pane1(self, -1, (600, 150), (550, 400), style=wx.SUNKEN_BORDER)
       self.picture2 = wx.StaticBitmap(panel2)
       pane12.SetBackgroundColour(wx.WHITE)
       self.images = ['image_data/total_wc_{0}_{1}.jpg'.format(s_name, p_number), 'image_data/good_wc
       wx.ComboBox(self, -1, pos=(20, 80), size=(100, -1), choices=authors, style=wx.CB_READONLY)
```

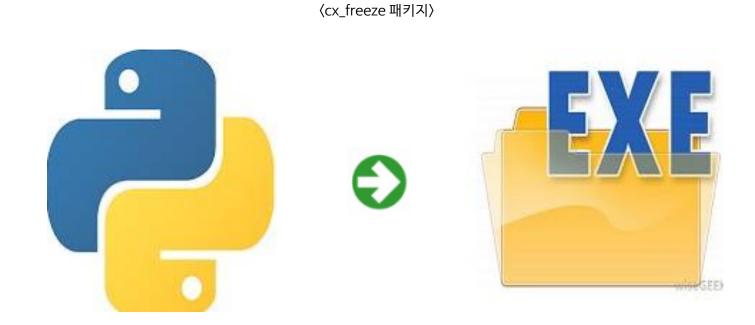
4) 시각화(GUI)







5) exe 실행파일 배포

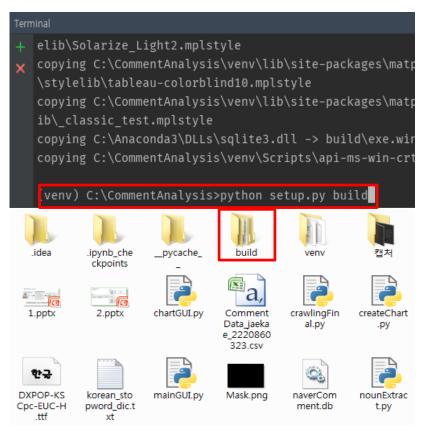


6) exe 실행파일 생성 및 확인

setup.py : 빌드에 관한 설정

```
from cx Freeze import setup, Executable
buildOptions = dict(packages = ["wordcloud", "idna", "re", "sys", "os", "jpype", "konlpy", "re", "sys", "os", "os
      base = None
              if sys.platform == "win32":
                                        base = "Win32GUI"
  exe = [Executable("mainGUI.py", base=base)]
    setup(
```

실행파일 빌드



4. 결론 및 기대효과

공통발생단어 배제 후, 카테고리별 or 만족도별

"차별화된 워드클라우드 결과"



"인기상품의 트렌드 확인목적 활용가능"

"소비자 반응분석 프로그램 개발"



"판매자가 관심 상품에 대해 간편한 분석가능"

[판매상품의 소비자반응 피드백 / 신상품 출시 전 소비자반응 사전탐색]

예를 들어, 가구/인테리어 스토어팜 판매자라면, 가구/인테리어 구매평 빈도분석을 통하여 소비자가 의자나 조립식가구에 대해 수요가 높으며 색상과 가격특성 외에도 냄새특성(가구냄새)이 만족도를 결정하는 주요요인임을 고려하여 판매전략이나 신제품 런칭계획을 수립하는데 참고할 수 있다.

5. 한계점 및 향후 계획

수치적인 분석결과 도출에 한계

"스토어팜 크롤링 수 제한" "네이버쇼핑 API권한 획득, 트래픽제한 회피" 트래픽과다 이유로 1회 URL접근 시, 크롤링 가능 구매평 1,000개 제한 데이터양(volume) 확보에 어려움 "수동적 웹 크롤링" "카테고리별 크롤링 자동화 구현" 일일이 URL을 입력하여 크롤링 데이터수집단계 작업시간 증가 "KOSAC 감성사전 부정어 처리" "만족도가 '불만'인 구매평 수집확대, KOSAC사전 및 분류 알고리즘 개선" 부정어의 구매평 데이터 부족 KOSAC사전 및 분류 알고리즘 한계 등 "판매량정보 등 활용정보 부족" "네이버쇼핑 API권한 개선요구 판매량과 클릭 수를 통한 인기순위만 제공 및 현업프로젝트 수행 시 판매량정보 요청"

Thank You