

자소서 적합성 검출

AI야, 합격 시켜조

목차

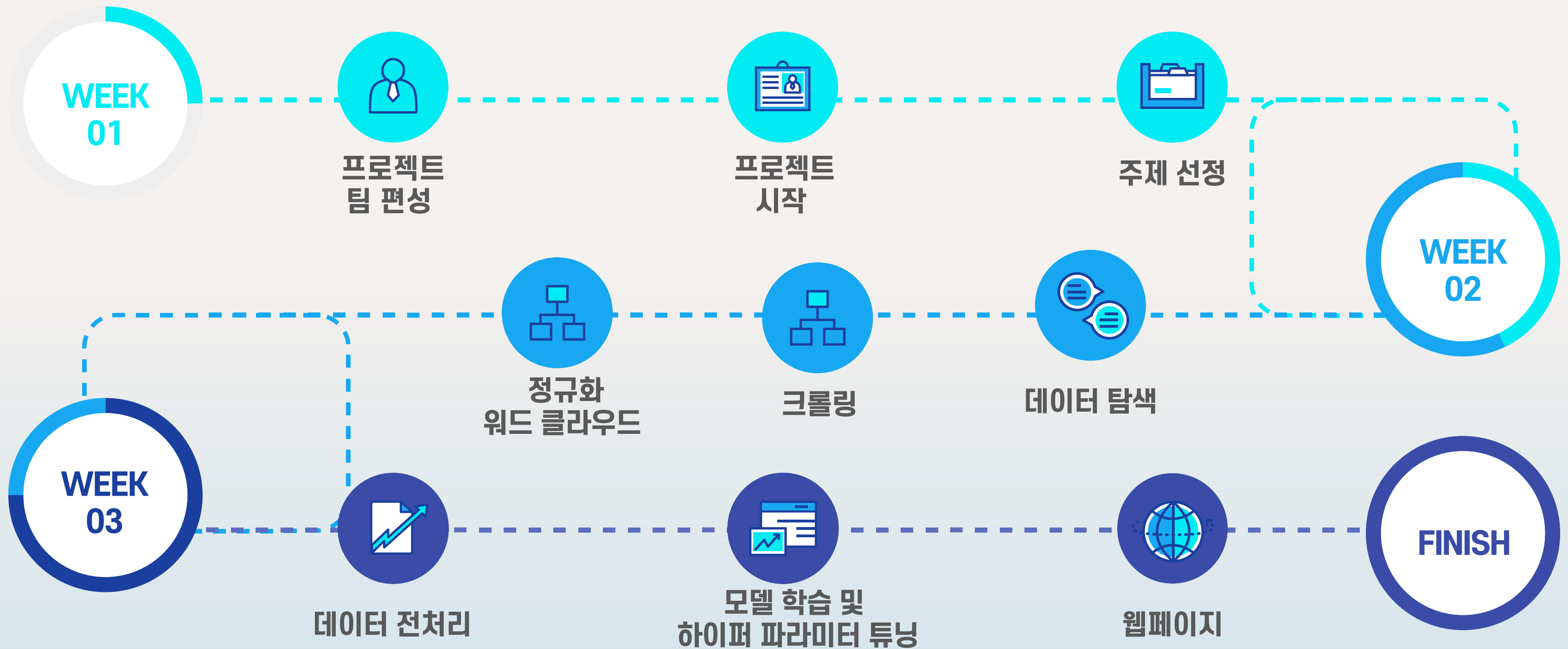
1. 조원소개
2. 프로젝트 일정
3. 주제선정 배경
4. 개발 공정
5. 웹 페이지

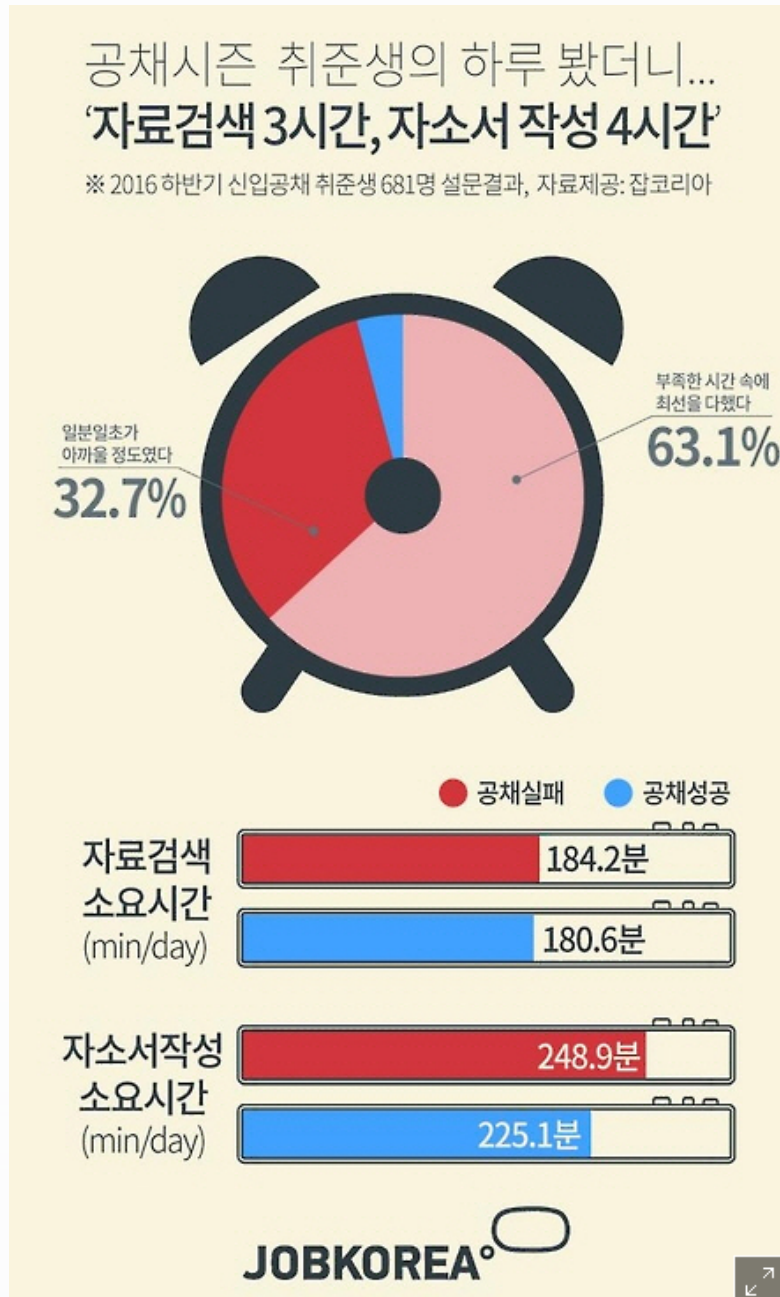
조원 소개

조장 : 서유진

조원 : 김한솔, 이혁빈, 차다훈

TIMELINE TITLE





공채시즌 취준생의 하루

취준생 681명을 설문한 결과 취업 준비생들은 하루 평균 182.8분(3시간)을 자료검색에 쓰고, 242.4분(4시간)을 자기소개서 작성에 소비했다고 밝혔다.

자기소개서를 작성하는데 시간을 들인 사람 중 합격자(하루 평균 225.1분)보다 불합격자(248.9분)가 더 많았다

자소서 쓰면 안되는 단어

매우,엄청 -> 자극하, 대단히
대개 -> 거의, 대부분
그치만,하지만 -> 그러나,그럼에도
그러니까 -> 따라서
~라고 생각한다 -> ~로 생각된다. ~일 것이다
~일지도 모른다 -> ~될 가능성이 있다
어떻게해도 -> 어떠한 수단을 사용해도
~재미있다 -> ~는 중요하다
~가 잔뜩있다. -> ~가 다수 존재한다
~하는 편이 좋을 것 같다 -> ~해야한다.
~는 싫다 -> ~는 적절하지 않다
합격시켜주세요. -> 기여할 수 있다고 생각된다
~라는 건 사기다 -> ~가 진실이라는 증거는 없다
~는 읽고 싶지 않다 -> ~를 정당하게 평가하는 것은 어렵다
~는 읽지 않았다 -> ~의 평가는 정해지지 않았다
이제 끝났다 -> 의논이 고착상태이므로 원점으로 돌아가 보기로 한다.
하기 싫어졌다 -> 이만 펜을 내려놓기로 한다.
~라 느낀다 -> ~라 추측된다
~하는 편이 좋을 것 같다 -> ~해야한다 ~할 필요가 있다
~는 없었다 -> ~라는 사실은 알려져 있지 않다.
~라는 건 틀렸다. -> ~라는 주장은 잘못이다.
이런얘기는 들은 적이 있다. -> 이러한 사례를 들 수 있다.
다들 ~라고 한다 -> 일반적으로 ~라고 한다.
다들 ~라고 생각한다 -> ~라고 널리 알려져있다.
교과서에 ~라고 적혀있다 -> ~라는 것이 통설이다.
연구했습니다만~는 모르겠네요. -> ~는 앞으로의 과제로 삼고싶다.
[실정]으로



레포트컴퍼니

자기소개서 컨설팅 비용안내

구분	글자수	서비스비용	소요시간	추가비용
자기소개서	500자 기준	40,000원	1~3일	빠른 서비스(1일 이내) + 10,000
1분 자기소개서	500자 기준			
경력기술서	500자 기준			

구분	글자수	서비스비용	소요시간	추가비용
대학·대학원 편입 자소서	500자 기준	50,000원	1~3일	빠른 서비스(1일 이내) + 10,000
대학·대학원 학업계획서	500자 기준			

성남시 구직자, AI 자소서 '애스크토비' 쓴다



2024 성남시 채용박람회 방문객들이 그레이비랩의 인공지능(AI) 자소서 서비스 '애스크토비(ASKTobi)'를 체험하고 있다. /사진=그레이비랩

성남시 일자리지원센터에서 인공지능(AI) 자소서 서비스 '애스크토비(이하 ASKTobi)'로 기업 맞춤형 이력서를 작성할 수 있게 됐다.

이번 협력으로 성남시 일자리 지원센터는 ASKTobi를 활용해 구직자에게 보다 전문적인 자소서 작성 컨설팅을 제공하게 된다.

ASKTobi는 AI 기술을 통해 구직자가 기업에 본인을 어필할 수 있는 자기소개서 스토리 초안을 제공하는 커리어 관리 서비스다.

사용자는 본인의 이력 정보를 입력한 뒤 원하는 포지션을 검색하면 실시간 채용 공고를 확인할 수 있다. 자소서 질문 9가지가 제공돼 질문별 초안을 생성할 수 있다.

특히 대기업 공채의 경우 기업별 자소서 질문이 연동돼 자소서 준비에 직접적인 도움을 받을 수 있으며, 영문 기능을 제공해 해외 취업 준비에도 활용 가능하다

주제 선정 배경

ASKTobi

정보 입력

완료

포지션 검색

이력서 / 자소서 결과확인

대한민국

서울

직무별

회사명

개발

머신러닝 엔지니어

Q

✓

딥브레인에이아이

[400억↑투자] 딥러닝 연구원 / 엔지니어 (전문연구요원 가능)

IT, 콘텐츠 / 서울

퇴사율5%이하 51~300명 설립4~9년 스타트업 커피 간식 의료비 자기계발 산업기능요원 전문연구요원

✓

액션파워

머신러닝 엔지니어 / AI Researcher

IT, 콘텐츠 / 서울

스타트업 음료 커피 간식 안마의자 건강검진 자기계발 산업기능요원 전문연구요원

✓

마키나락스

시니어 머신러닝 엔지니어

IT, 콘텐츠 / 서울

연봉상위6~10% 인원급성장 퇴사율5%이하 51~300명 설립4~9년 스타트업 음료 간식 전문연구요원 인공지능 재택근무 IT, 콘텐츠

대분류

개발

디자인

경영·비즈니스

영업

물류·무역

고객서비스·리테일

마케팅·광고

엔지니어링·설계

미디어

게임 제작

HR

금융

건설·시설

교육

법률·법집행기관

의료·제약·바이오

제조·생산

식·음료

공공·복지

선택한 포지션

파이온코퍼레이션 - 브이켓 딥러닝 엔지니어

자기소개서 질문

성장과정

생성

생성을 누르면 [정보입력] 수정은 불가능합니다.

성장과정

"문제 해결을 향한 첫걸음: 디지털 세계에서의 깨달음"

제 성장 과정은 주로적으로 문제를 해결하고자 하는 자세와 깊은 관심에서부터 크게 영향을 받았습니다. 이 중 가장 저에게 변화를 가져다 준 사건은 대학교 시절 처음으로 컴퓨터 프로그래밍을 접하면서부터입니다. 해당 경험은 제가 정보의 바다에서 해답을 찾아내고, 복잡한 문제를 단순화하는 과정을 즐기게 만들었습니다. 이러한 경험은 제가 파이온코퍼레이션의 브이켓 딥러닝 엔지니어로서 필요한 문제 해결 능력과 창의적인 아이디어를 제시할 수 있는 기반을 마련해 주었습니다.

대학 시절의 경험은 단순히 코딩 기술을 향상시키는 것을 넘어, 제가 딥러닝, 특히 컴퓨터 비전 분야에 깊은 흥미를 갖게 만들었습니다. 이는 박사 과정 중에 집중적으로 연구했으며, 해당 연구는 Kaggle 대회에서 우수한 성과를 거두어 파이온코퍼레이션에서 요구하는 문제 해결 능력과 유사한 경험을 제공하였습니다. 또한, 다양한 프로젝트를 진행하며 Python을 포함한 프로그래밍 언어에 대한 중급 이상의 활용 능력을 갖추었으며, 자체적으로 개선점을 제안하고 해결책을 찾는 데 있어서 자율적이고 창의적인 접근 방식을 개발했습니다.

파이온코퍼레이션의 자율적이고 창의적인 조직 문화는 제가 키워온 역량과 잘 부합하며, 제가 가진 문제를 인식하고 해결하는 능력은 브이켓 서비스의 개선과 발전에 기여할 것입니다. 저는 파이온코퍼레이션에서 제 기술적 능력과 창의적인 사고를 바탕으로 VCAT 서비스가 글로벌 B2B SaaS 분야에서 더욱 성장할 수 있도록 기여하고자 합니다.

취업실전전략

합격자소서

인적성·면접후기
인성역량검사
공기업 모의고사
직무인터뷰

역량테스트(MICT)

취업 TOOL

글자수 세기
어학점수 변환
사진크기조정
자소서 자동완성
자소서·면접 사전
연봉계산기
퇴직금계산기
실업급여계산기
학점계산기
온라인 도장

문서서식 자료

합격자소서 ⓘ

지원분야	근무형태	학력	30대 그룹별	기업형태	
기획·전략 법무·사무·총무 인사·HR 회계·세무 마케팅·광고·MD 개발·디자인	←대분류를 선택하세요.	<input type="checkbox"/> 신입 <input type="checkbox"/> 인턴	<input type="checkbox"/> 고등학교 졸업 <input type="checkbox"/> 대학 졸업(2,3년) <input type="checkbox"/> 대학교 졸업(4년) <input type="checkbox"/> 석/박사 졸업	<input type="checkbox"/> 교보생명보험그룹 <input type="checkbox"/> 농협그룹 <input type="checkbox"/> 대림그룹 <input type="checkbox"/> 대성그룹 <input type="checkbox"/> 롯데그룹 <input type="checkbox"/> 메리츠금융그룹	<input type="checkbox"/> 30대그룹 <input type="checkbox"/> 매출1000대기업 <input type="checkbox"/> 대기업 <input type="checkbox"/> 공공기관 공사협회 <input type="checkbox"/> 외국계 <input type="checkbox"/> 중소기업

선택한 검색조건
323건

개발·데이터-전체 × 초기화

선택 조건으로 검색

총 323건

✓ 관심기업 합격자소서 보기

✓ 전문가 분석 보기

평점높은순 ▼

기업명, 자소서 항목 검색 🔍

true friend

한국투자증권(주)

2020년 하반기 | 신입 | 웹개발자

자소서 작성

Q. 성장과정:가족사항, 학창시절, 교우관계, 생활습관, 자신에게 크게 영향을 미친 사건 등을 포함하여 ...

👁 150,434

A. 저희 가족은 회사원이신 아버지와 어머니 그리고 누나와 저 4인 가족입니다. 학창 시절 독서에 관심...

전문가 총평 ★★★★★ 5 | "본 자기소개서의 장점은 모든 질문에 대해 정확하게 답을 했다는 것입니다. 이 말은 회사와 의사소통이 잘 되는 지원자라는 말입니다. 자신..."

크롤링

합격 자소서 게시판 -> 지원분야 : 개발·데이터 -> 전문가 분석 보기 -> 평점 높은순 : 총 323건



```
url_list = ['https://www.jobkorea.co.kr/starter/PassAssay?FavorCo_Stat=0&Pass_An_Stat=1&OrderBy=2&EduType=0&WorkType=0&schPart=10031&isSaved=1&Page='
            + str(x) for x in range(1, 18)]

links = []
for url in url_list:
    driver.get(url)
    time.sleep(3)
    question_items = driver.find_elements(By.CSS_SELECTOR, 'div.item.question')
    for item in question_items:
        a_tag = item.find_element(By.TAG_NAME, 'a')
        href = a_tag.get_attribute('href')
        links.append(href)

driver.quit()
```

```
https://www.jobkorea.co.kr/starter/PassAssay/View/236140?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/236159?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/201630?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/200697?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/194528?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/193261?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/192318?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/192093?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/192016?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/191179?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/191434?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/191435?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
https://www.jobkorea.co.kr/starter/PassAssay/View/191439?Page=1&OrderBy=2&FavorCo_Stat=0&schPart=10031&Pass_An_Stat=1
```

```
def crawling_url(url):
    driver.get(url)
    response = driver.page_source
    one_page = bs4.BeautifulSoup(response, 'html.parser')

    # 불필요 요소 제거
    for p_element in one_page.find_all('p', {'class': 'txSpllChk'}):
        p_element.extract()

    for span_element in one_page.find_all('span', {'class': 'sup'}):
        span_element.extract()

    for a_element in one_page.find_all('a'):
        a_element.extract()

    # 공백 제거 및 텍스트만 추출
    kwd_list = []
    for text in one_page.find_all('div', {'class': 'tx'}):
        kwd_list.append(text.get_text(separator=' ', strip=True))

    return kwd_list
```

```
korean_essay_lists = []
for kwd_url in essay_urls:
    kwd_list = crawling_url(kwd_url)
    korean_essay_lists.append(kwd_list)
    time.sleep(5)
```

```
1 "[함께 하는 삶] 대학 시절, 선거 개표 아르바이트를 하면서 따뜻함을 실천한 적이 있습니다. 아르바이트 당시 개표
2 "1. 객체지향프로그래밍 A+,프로그래밍 수업을 들으면서 프로그램을 만들고 수정하는 과정이 어릴 때부터 좋아하던 수
3 "고등학교를 졸업하고 대입을 다시 준비하게 되었습니다. 학원을 통해 부족한 공부를 보충하고 싶었지만 무엇보다 부
4 "1) 00 모바일 앱 서비스 개발 - 기관명 : (주)00 - 기간 : 2015.01 ~ 2015.09 - 근무부서/직위 : 서비스 개발/개발
5 "저는 금융 it의 발전 가능성이 매우 크다고 생각합니다. 최근 국내의 금융산업 쪽에서 it 규제가 많이 완화되 다양한
6 제 삶에서 가장 큰 변화를 준 사람은 바로 아버지입니다. 아버지는 제가 초등학교 때부터 고등학교 때까지 대략 10년
7 "저를 돌아해보니 저는 항상 새로운 것을 배우는 걸 좋아했던 것 같습니다. 어렸을 때부터 운동을 좋아해서 다양한 운
8 ""슈어소프트테크""에 지원한 이유는 현재의 위치에 멈추지 않고 끊임없이 혁신을 준비하여 새로운 도약을 준비하는
9 "[위기를 기회로 만드는 것은 도전 정신이다.] 고등학교를 졸업하고 대학 입시를 다시 한 번 도전하였습니다. 학원을
10 "[선택과 집중] 저의 첫 번째 선택은 카페, 영화관, 호프집, 사무보조 등 다양한 아르바이트 경험 입니다. 학업과 경
11 "[열정, 호기심 그리고 성취감] 제가 S/W 개발자가 되려는 이유는 제 열정을 불타오르게 하기 때문입니다. 먼저 문제
12 저는 결과에 대해 책임을 지는 사람입니다. 저의 아버지는 제가 초등학교 때부터 고등학교 때까지 대략 10년 정도를
13 "매년 Las Vegas에서 개최하는 CES(세계가전박람회)를 Buyer자격으로 참관하여 최고수준의 기술 미리 체험할 수 있었
14 "타 금융투자 회사에서의 인턴활동은 저를 금융IT에 관심을 갖게 만들어 주었습니다. 금융에 대한 학습을 하며 IT인으로서
15 "인생은 혼자 살아가는 것이 아닙니다. 함께 사는 것입니다. 아무리 개인의 능력이 뛰어나도 결국은 서로 돕고 사는
16 "[ 유니폼] 단체 생활에서 소속감을 가지고 있는 유니폼은 저를 표현할 수 있는 단어라 생각합니다. 일례로 대학교 입
17 "[ERP의 시작과 경험] ** 소프트웨어 멤버십 소속으로 1년 가까운 시간을 지냈고 면접이 얼마 안 남은 상황이었습니
18 "저는 밝고 긍정적이신 부모님의 영향으로 어디서나 밝고 긍정적이며 다양성을 존중하는 열린 마음을 갖고 있습니다.
```

크롤링

데이터 번역 및 출력

```
for row in rdr:
    if len(row[0]) >= 3000:
        dot_index = row[0][3000:2500:-1].index('.')
        x1, x2 = row[0][:3000-dot_index+1], row[0][3000-dot_index+1:]
        input_text = x1 # 번역할 텍스트
        input_box = driver.find_element(By.CSS_SELECTOR, "textarea#txtSource")
        input_box.clear()
        input_box.send_keys(input_text)
        time.sleep(20)

        translate_button = driver.find_element(By.CSS_SELECTOR, "button#btnTranslate")
        translate_button.click()

        # 번역 결과 대기
        output_element = WebDriverWait(driver, 30).until(EC.presence_of_element_located((By.CSS_SELECTOR, "div#txtTarget")))

        # 번역 결과 추출
        output = output_element.text
        translated_x1 = output

        input_text = x2 # 번역할 텍스트
        input_box = driver.find_element(By.CSS_SELECTOR, "textarea#txtSource")
        input_box.clear()
        input_box.send_keys(input_text)
        time.sleep(20)

        translate_button = driver.find_element(By.CSS_SELECTOR, "button#btnTranslate")
        translate_button.click()

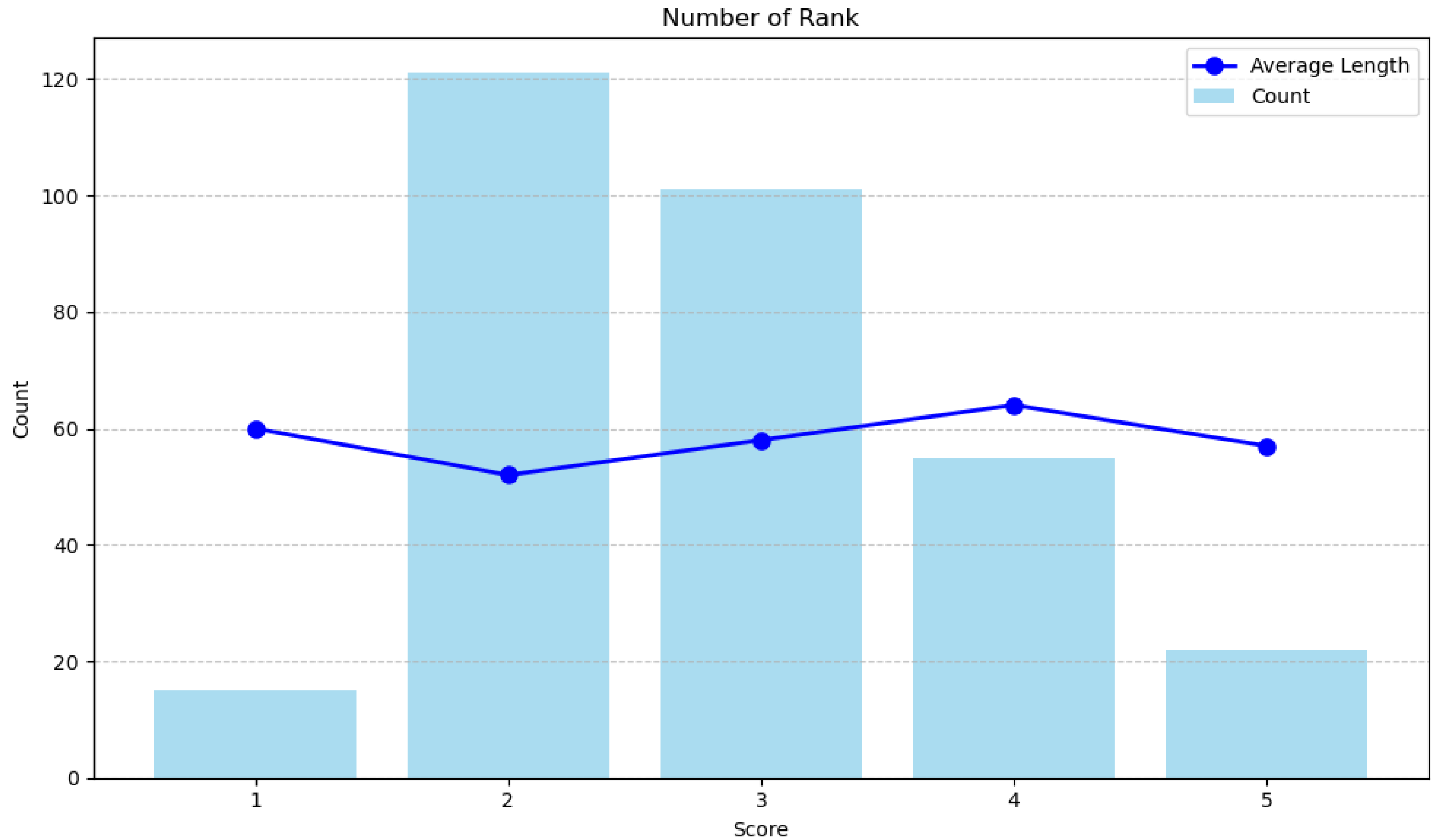
        # 번역 결과 대기
        output_element = WebDriverWait(driver, 30).until(EC.presence_of_element_located((By.CSS_SELECTOR, "div#txtTarget")))

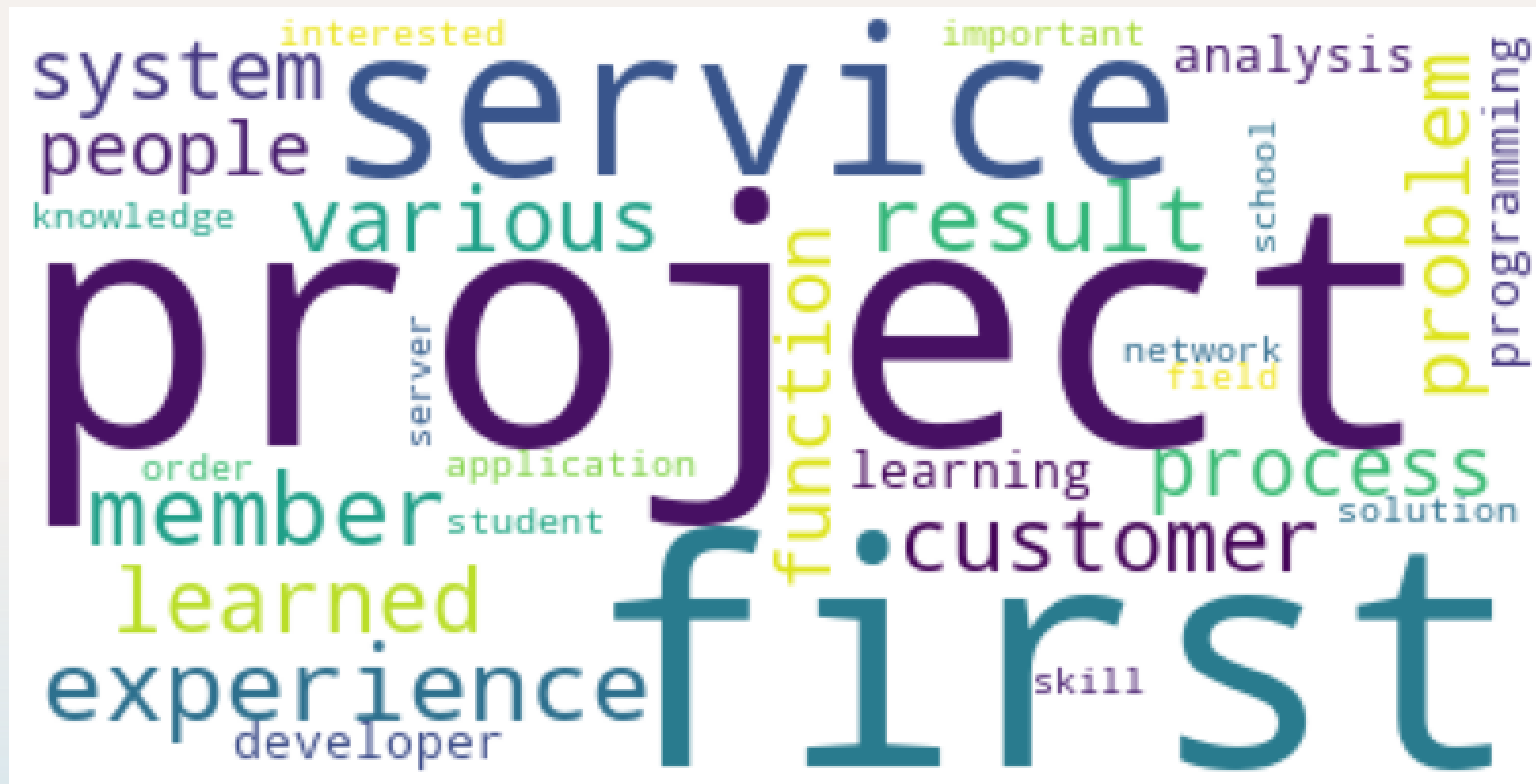
        # 번역 결과 추출
        output = output_element.text
        translated_x2 = output
        sentence = translated_x1 + translated_x2

        translated_list.append(sentence)

# 입력 상자 초기화
input_box.clear()
```

['My family consists of four members: my father, mother, older sister
ng books and continued to read books in a wide range of fields, especi
nd Western philosophy. As a result, I have acquired the ability to und
e also building up my knowledge of the humanities and trying to imitat
ding books and who have desirable moral senses and humility, and I stu
aving school after self-study at night, following the example of since
ized that by having a big dream, the criminal can be reborn as a great
have a lifelong dream to live a better society." "There is a case in w
ing data allocated during the internship program at the AI research in
building strategic competencies and overcoming them. Having the most i
ictive accuracy calculations by seven laggle competitions, but volunta
capabilities by overachieving work-level overachievement. As a result,
s.", The securities industry majored in business administration hoping
with data-based strategies and insights. As I decided on my career pat
latform that is integrated with customers\' daily lives based on data
g data analysis and AI implementation capabilities and SW app developm
the improvement of our Korean investment, Ministock and Bankies FinTec
We are applying to the Trading Development Department and the BIZ Devel





전체

project	1205
experience	925
first	846
system	809
members	717
various	624
information	599
learned	593
people	563
working	519
school	494
problem	473
customers	467
knowledge	464
service	455
result	448
process	443
skills	432
security	424
design	423

5점

project	82
people	57
experience	56
design	42
games	40
problem	39
first	39
members	38
service	37
system	35
various	35
information	32
knowledge	32
business	31
working	31
learned	31
school	29
skills	29
activities	28
users	27

1점

project	73
system	54
design	50
experience	47
computer	42
school	41
first	41
graduate	40
safety	40
software	40
communication	39
information	35
application	35
knowledge	35
korea	35
management	34
related	33
research	32
programming	31
college	30

제일 많이 사용되는 단어

project experience first

5점에만 있는 단어

```
{'activities',
'business',
'games',
'learned',
'members',
'people',
'problem',
'service',
'skills',
'users',
'various',
'working'}
```

1점에만 있는 단어

```
{'application',
'college',
'communication',
'computer',
'graduate',
'korea',
'management',
'programming',
'related',
'research',
'safety',
'software'}
```


Top 20 Bigrams



Top 20 Trigrams



Top 20 Bigrams:

(solve, problem)	101
(working, project)	92
(various, fields)	76
(talented, person)	73
(information, telecommunication)	72
(artificial, intelligence)	67
(sense, responsibility)	65
(industrial, revolution)	65
(management, system)	63
(graduate, school)	60
(information, security)	59
(communication, skills)	58
(problem, solving)	53
(elementary, school)	48
(inventory, management)	46
(information, protection)	46
(olive, networks)	43
(complete, project)	43
(middle, school)	41
(solve, problems)	41

Top 20 Trigrams:

(inventory, management, system)	37
(lotte, information, telecommunication)	33
(fourth, industrial, revolution)	29
(order, solve, problem)	21
(information, processing, technician)	18
(central, inventory, management)	18
(system, analysis, design)	17
(people, various, fields)	15
(problem, solving, ability)	13
(developer, training, course)	12
(problem, solving, skills)	12
(information, telecommunication, electronics)	11
(information, security, technician)	11
(korea, trade, information)	11
(experience, various, projects)	11
(unmanned, parking, system)	10
(trade, information, telecommunication)	10
(existing, shopping, malls)	10
(grade, elementary, school)	10
(safety, service, application)	10

Logistic Regressor

	precision	recall	f1-score	support
0	0.80	0.67	0.73	24
1	0.70	0.83	0.76	23
accuracy			0.74	47
macro avg	0.75	0.75	0.74	47
weighted avg	0.75	0.74	0.74	47

SVM

	precision	recall	f1-score	support
0	0.79	0.79	0.79	24
1	0.78	0.78	0.78	23
accuracy			0.79	47
macro avg	0.79	0.79	0.79	47
weighted avg	0.79	0.79	0.79	47

Random Forest

	precision	recall	f1-score	support
0	0.78	0.58	0.67	24
1	0.66	0.83	0.73	23
accuracy			0.70	47
macro avg	0.72	0.70	0.70	47
weighted avg	0.72	0.70	0.70	47

Light GBM

	precision	recall	f1-score	support
0	0.70	0.79	0.75	24
1	0.75	0.65	0.70	23
accuracy			0.72	47
macro avg	0.73	0.72	0.72	47
weighted avg	0.73	0.72	0.72	47

XG boost

	precision	recall	f1-score	support
0	0.73	0.67	0.70	24
1	0.68	0.74	0.71	23
accuracy			0.70	47
macro avg	0.70	0.70	0.70	47
weighted avg	0.70	0.70	0.70	47

```
file_path = r'.\crawling_data\preprocessed_essay.csv'
```

```
# 자소서 리스트화
documents = []
with open(file_path, 'r') as f:
    reader = csv.reader(f)
    for row in reader:
        documents.append(row[0])
```

```
# 사용하지 않은 2점짜리 자소서 리스트
documents = documents[-175:-77]
```

```
# 불용어 리스트 가져오기
stop_words = set(stopwords.words('english'))
```

```
def preprocess(text):
    tokens = word_tokenize(text)
    filtered_tokens = [word for word in tokens if word.isalpha() and word not in stop_words]
    return filtered_tokens
```

```
documents_corpus = []
for document in documents:
    documents_corpus.append(preprocess(document))
```

```
rf_vec = [model_rf.infer_vector(doc) for doc in documents_corpus]
svc_vec = [model_svc.infer_vector(doc) for doc in documents_corpus]
xgb_vec = [model_xgb.infer_vector(doc) for doc in documents_corpus]
lgbm_vec = [model_lgbm.infer_vector(doc) for doc in documents_corpus]
lr_vec = [model_lr.infer_vector(doc) for doc in documents_corpus]
```

```
print(clf_rf.predict(rf_vec).sum())
print(clf_svc.predict(svc_vec).sum())
print(clf_xgb.predict(xgb_vec).sum())
print(clf_lgbm.predict(lgbm_vec).sum())
print(clf_lr.predict(lr_vec).sum())
```

분류 모델

```
# Optuna 목적 함수
def objective(trial):
    # Doc2Vec 하이퍼파라미터 튜닝
    vector_size = trial.suggest_int('vector_size', 50, 300, step=10)
    window = trial.suggest_int('window', 3, 15)
    min_count = trial.suggest_int('min_count', 1, 20)
    epochs = trial.suggest_int('epochs', 20, 50, step=5)
    dm = trial.suggest_int('dm', 0, 1)

    # Doc2Vec 모델 학습
    model = Doc2Vec(my_tagged_corpus, vector_size=vector_size, window=window, min_count=min_count, epochs=epochs, dm=dm,
X_train_vec = [model.infer_vector(doc) for doc in X_train]
X_test_vec = [model.infer_vector(doc) for doc in X_test]

    # SVM 하이퍼파라미터 튜닝
    C = trial.suggest_float('C', 0.01, 100.0, log=True)
    kernel = trial.suggest_categorical('kernel', ['linear', 'rbf', 'poly'])
    degree = trial.suggest_int('degree', 2, 5) if kernel == 'poly' else 3
    gamma = trial.suggest_float('gamma', 1e-4, 1.0, log=True) if kernel in ['rbf', 'poly'] else 'scale'

    # SVM 모델 생성
    model = SVC(
        C=C,
        kernel=kernel,
        degree=degree,
        gamma=gamma,
        random_state=123
    )

    model.fit(X_train_vec, y_train)
    score = f1_score(y_test, model.predict(X_test_vec))

    return score

# Optuna 최적화 수행
study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=200, n_jobs=-1)
```

```
best_params = study.best_params
```

```
# Doc2Vec 모델 생성
```

```
model = Doc2Vec(my_tagged_corpus, vector_size=best_params['vector_size'],
                window=best_params['window'], min_count=best_params['min_count'],
                epochs=best_params['epochs'], dm=best_params['dm'], seed=123)
```

```
# SVM 분류 모델 생성
```

```
clf = SVC(C=best_params['C'], kernel=best_params['kernel'], gamma=best_params['gamma'], random_state=123)
```

```
# Doc2Vec 벡터 생성
```

```
X_train_vec = [model.infer_vector(doc) for doc in X_train]
X_test_vec = [model.infer_vector(doc) for doc in X_test]
```

```
# 분류 모델 학습 및 예측
```

```
clf.fit(X_train_vec, y_train)
y_pred = clf.predict(X_test_vec)
```

```
from sklearn.metrics import classification_report
```

```
report = classification_report(y_test, y_pred)
```

Best hyperparameters: {'vector_size': 240, 'window': 7, 'min_count': 5, 'epochs': 25, 'dm': 1, 'C': 77.87975872156572, 'kernel': 'rbf', 'gamma': 0.09979163094399905}

Logistic Regressor

	precision	recall	f1-score	support
0	0.85	0.71	0.77	24
1	0.74	0.87	0.80	23
accuracy			0.79	47
macro avg	0.80	0.79	0.79	47
weighted avg	0.80	0.79	0.79	47

SVM

	precision	recall	f1-score	support
0	0.78	0.88	0.82	24
1	0.85	0.74	0.79	23
accuracy			0.81	47
macro avg	0.81	0.81	0.81	47
weighted avg	0.81	0.81	0.81	47

Random Forest

	precision	recall	f1-score	support
0	0.69	0.75	0.72	24
1	0.71	0.65	0.68	23
accuracy			0.70	47
macro avg	0.70	0.70	0.70	47
weighted avg	0.70	0.70	0.70	47

Light GBM

	precision	recall	f1-score	support
0	0.73	0.67	0.70	24
1	0.68	0.74	0.71	23
accuracy			0.70	47
macro avg	0.70	0.70	0.70	47
weighted avg	0.70	0.70	0.70	47

XG boost

	precision	recall	f1-score	support
0	0.72	0.75	0.73	24
1	0.73	0.70	0.71	23
accuracy			0.72	47
macro avg	0.72	0.72	0.72	47
weighted avg	0.72	0.72	0.72	47

```
file_path = r'.\crawling_data\preprocessed_essay.csv'
```

```
# 자소서 리스트화
```

```
documents = []
```

```
with open(file_path, 'r') as f:
```

```
    reader = csv.reader(f)
```

```
    for row in reader:
```

```
        documents.append(row[0])
```

```
# 사용하지 않은 2점짜리 자소서 리스트
```

```
documents = documents[-175:-77]
```

```
# 불용어 리스트 가져오기
```

```
stop_words = set(stopwords.words('english'))
```

```
def preprocess(text):
```

```
    tokens = word_tokenize(text)
```

```
    filtered_tokens = [word for word in tokens if word.isalpha() and word not in stop_words]
```

```
    return filtered_tokens
```

```
documents_corpus = []
```

```
for document in documents:
```

```
    documents_corpus.append(preprocess(document))
```

```
rf_vec = [model_rf.infer_vector(doc) for doc in documents_corpus]
```

```
svc_vec = [model_svc.infer_vector(doc) for doc in documents_corpus]
```

```
xgb_vec = [model_xgb.infer_vector(doc) for doc in documents_corpus]
```

```
lgbm_vec = [model_lgbm.infer_vector(doc) for doc in documents_corpus]
```

```
lr_vec = [model_lr.infer_vector(doc) for doc in documents_corpus]
```

```
# 총 98개 자소서 예측
```

```
print(clf_rf.predict(rf_vec).sum())
```

```
print(clf_svc.predict(svc_vec).sum())
```

```
print(clf_xgb.predict(xgb_vec).sum())
```

```
print(clf_lgbm.predict(lgbm_vec).sum())
```

```
print(clf_lr.predict(lr_vec).sum())
```

38

35

51

42

49

분류 모델

```
# Optuna 목적 함수
def objective(trial):
    # Doc2Vec 하이퍼파라미터 튜닝
    vector_size = trial.suggest_int('vector_size', 20, 60)
    window = trial.suggest_int('window', 2, 10)
    min_count = trial.suggest_int('min_count', 1, 10)
    epochs = trial.suggest_int('epochs', 10, 30, step=5)
    negative = trial.suggest_int('negative', 3, 15)
    dm = trial.suggest_categorical('dm', [0, 1])

    # Doc2Vec 모델 학습
    model = Doc2Vec(my_tagged_corpus, vector_size=vector_size, window=window, min_count=min_count, epochs=epochs, dm=dm, negative=negative)
    X_train_vec = [model.infer_vector(doc) for doc in X_train]
    X_test_vec = [model.infer_vector(doc) for doc in X_test]

    # SVM 하이퍼파라미터 튜닝
    C = trial.suggest_float('C', 0.01, 100.0, log=True)
    kernel = trial.suggest_categorical('kernel', ['linear', 'rbf', 'poly'])
    degree = trial.suggest_int('degree', 2, 5) if kernel == 'poly' else 3
    gamma = trial.suggest_float('gamma', 1e-4, 1.0, log=True) if kernel in ['rbf', 'poly'] else 'scale'

    # SVM 모델 생성
    model = SVC(
        C=C,
        kernel=kernel,
        degree=degree,
        gamma=gamma,
        random_state=123
    )

    model.fit(X_train_vec, y_train)
    score = f1_score(y_test, model.predict(X_test_vec))

    return score

# Optuna 최적화 수행
study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=200, n_jobs=1)

# 최적 하이퍼파라미터 출력
print('Best hyperparameters: ', study.best_params)
```

```
best_params = {'vector_size': 52, 'window': 4, 'min_count': 2, 'epochs': 25, 'negative': 5, 'dm': 1, 'C': 60.91278167538605, 'kernel': 'rbf', 'gamma': 0.06294729784153419}
```

```
# Doc2Vec 모델 생성
```

```
model_svc = Doc2Vec(my_tagged_corpus, vector_size=best_params['vector_size'],
                    window=best_params['window'], min_count=best_params['min_count'],
                    epochs=best_params['epochs'], dm=best_params['dm'],
                    negative=best_params['negative'], seed=123)
```

```
# SVM 분류 모델 생성
```

```
clf_svc = SVC(C=best_params['C'], kernel=best_params['kernel'], gamma=best_params['gamma'], random_state=123)#degree=best_params['degree']
```

```
# Doc2Vec 벡터 생성
```

```
X_train_vec = [model_svc.infer_vector(doc) for doc in X_train]
X_test_vec = [model_svc.infer_vector(doc) for doc in X_test]
```

```
# 분류 모델 학습 및 예측
```

```
clf_svc.fit(X_train_vec, y_train)
y_pred = clf_svc.predict(X_test_vec)
```

```
report = classification_report(y_test, y_pred)
print(report)
```

```
Best hyperparameters: {'vector_size': 52, 'window': 4, 'min_count': 2, 'epochs': 25, 'negative': 5, 'dm': 1, 'C': 60.91278167538605, 'kernel': 'rbf', 'gamma': 0.06294729784153419}
```

Random Forest

	precision	recall	f1-score	support
불합격	0.69	0.64	0.67	14
합격	0.72	0.76	0.74	17
accuracy			0.71	31
macro avg	0.71	0.70	0.70	31
weighted avg	0.71	0.71	0.71	31

	precision	recall	f1-score	support
0	0.53	0.57	0.55	14
1	0.62	0.59	0.61	17
accuracy			0.58	31
macro avg	0.58	0.58	0.58	31
weighted avg	0.58	0.58	0.58	31

	precision	recall	f1-score	support
0	0.67	0.57	0.62	14
1	0.68	0.76	0.72	17
accuracy			0.68	31
macro avg	0.68	0.67	0.67	31
weighted avg	0.68	0.68	0.67	31

SVM

	precision	recall	f1-score	support
불합격	0.62	0.57	0.59	14
합격	0.67	0.71	0.69	17
accuracy			0.65	31
macro avg	0.64	0.64	0.64	31
weighted avg	0.64	0.65	0.64	31

Voting

	precision	recall	f1-score	support
불합격	0.58	0.79	0.67	14
합격	0.75	0.53	0.62	17
accuracy			0.65	31
macro avg	0.66	0.66	0.64	31
weighted avg	0.67	0.65	0.64	31

Random Forest

	precision	recall	f1-score	support
불합격	0.50	0.60	0.55	15
합격	0.54	0.44	0.48	16
accuracy			0.52	31
macro avg	0.52	0.52	0.51	31
weighted avg	0.52	0.52	0.51	31

	precision	recall	f1-score	support
0	0.56	0.60	0.58	15
1	0.60	0.56	0.58	16
accuracy			0.58	31
macro avg	0.58	0.58	0.58	31
weighted avg	0.58	0.58	0.58	31

	precision	recall	f1-score	support
0	0.54	0.47	0.50	15
1	0.56	0.62	0.59	16
accuracy			0.55	31
macro avg	0.55	0.55	0.54	31
weighted avg	0.55	0.55	0.55	31

konlpy

SVM

	precision	recall	f1-score	support
불합격	0.40	0.40	0.40	15
합격	0.44	0.44	0.44	16
accuracy			0.42	31
macro avg	0.42	0.42	0.42	31
weighted avg	0.42	0.42	0.42	31

Voting

	precision	recall	f1-score	support
불합격	0.48	0.67	0.56	15
합격	0.50	0.31	0.38	16
accuracy			0.48	31
macro avg	0.49	0.49	0.47	31
weighted avg	0.49	0.48	0.47	31

SVM

```
# SVM Regressor  
0.11437047168352532  
0.9500224166962041
```

Linear Regression

```
# LinearRegression  
0.09882837533950806  
0.9583222552830236
```

Random Forest

```
# RandomForestRegressor  
0.19711461354956705  
0.9045541814034423
```

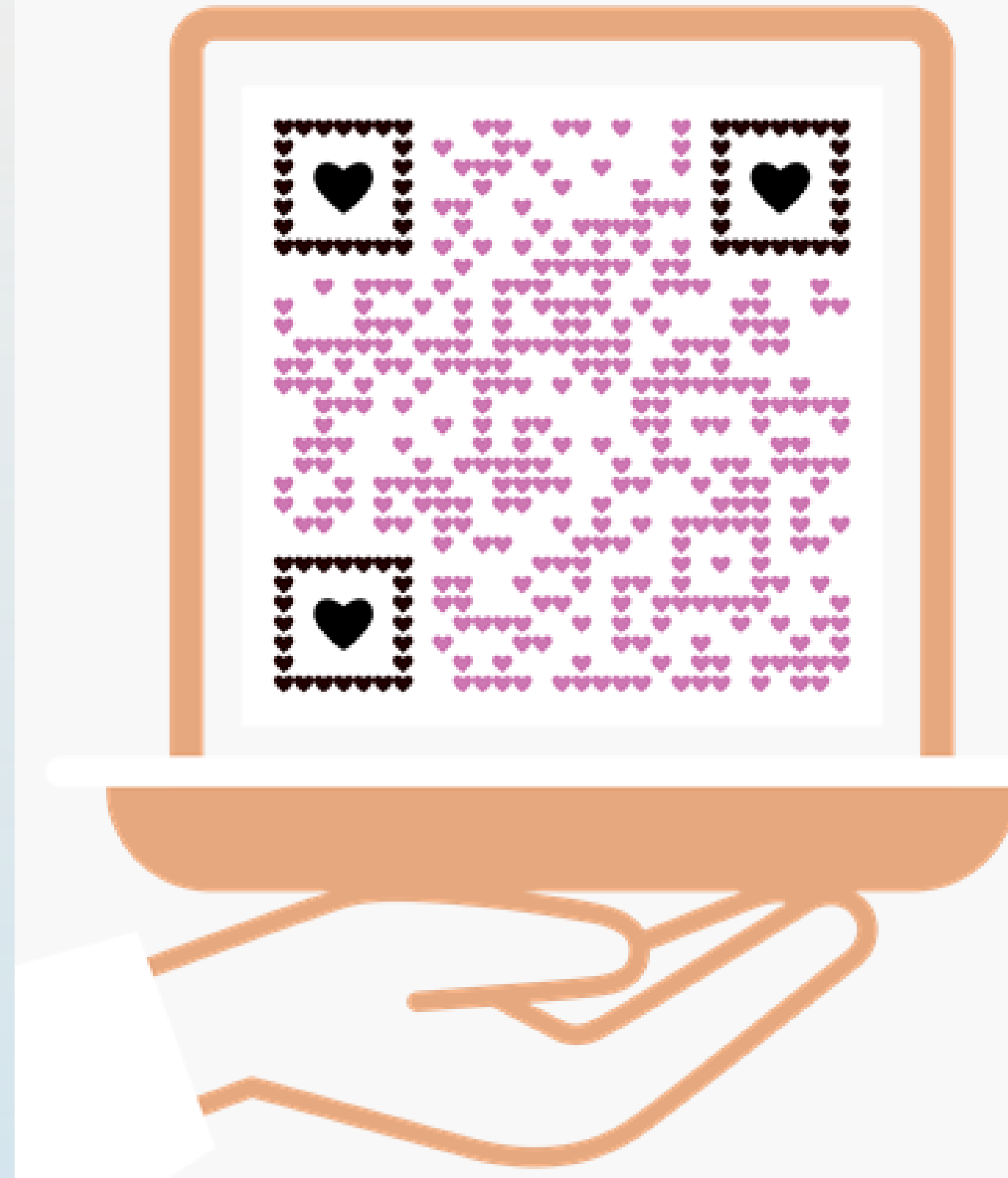
Light GBM

```
# LightGBM Regressor  
0.22561849344028584  
0.8883524313442634
```

XGBoost

```
# XGBoost Regressor  
0.0727090835571289  
0.9721109011825732
```

시야, 합격시켜조!





AI야 합격 시켜조

자기소개서 보조 프로그램으로
당신이 원하는 회사에 한걸음 더 나아가세요

시작하기

<https://essay-classifier.onrender.com>



자기소개서 적합 예측기

저희 가족은 회사원이신 아버지와 어머니 그리고 누나와 저 4인 가족입니다. 학창 시절 독서에 관심을 갖고 폭넓은 분야의 책을 지속적으로 읽었으며, 특히 위인전과 고전 소설, 동서양 철학 등 인문 고전을 탐독했습니다. 덕분에 다양한 인간상에 대한 이해와 공감 능력, 그리고 뛰어난 이성과 논리를 습득하는 한편, 인문학적 소양을 쌓으며 위인을 본받으려는 마음가짐을 다지게 되었습니다. 이런 특성으로 인해 독서를 좋아하며 바람직한 도덕 관념과 겸손함을 갖춘 친구들과 주로 교우관계를 맺었으며, 성실하고 청렴하신 부모님을 본받아 아침 일찍 등교하고 야간 자율 학습 후 하교하는 성실한 생활 습관을 바탕으로 공부에 임했습니다. 이지성 작가의 '꿈꾸는 다락방'을 읽으며 범인도 큰 꿈을 가짐으로써 위인으로 거듭나고 결국 그 꿈을 이룰 수 있다는 것을 깨달았고, 이것이 저에게 크게 영향을 미쳐 더 나은 사회를 위한 삶을 살고자 하는 평생의 꿈을 품게 되었습니다. "AI 연구 기관 OOOOO 인턴쉽 과정에서 할당 받은 데이터 분석 업무 달성에 대해 초기에 부족한 실력으로 인해 크게 좌절했으나 결국 전략적으로 역량을 쌓아 인턴 중 최고 성과를 달성하며 극복해낸 사례가 있습니다. 인턴 중 가장 부족한 데이터 분석 역량을 갖고 있던 저는 7개의 kaggle competition에서 일정한 예측 정확도 산출을 할당 받았으나, 자발적으로 더 큰 목표에 도전하여, 업무 단계별 초과달성을 통해 체계적으로 역량을 성장시키는 전략을 실천했습니다. 그 결과 최고 성과를 내어 인센티브와 정규직 제의를 받았습니다.",증권업은 데이터 기반의 전략과 통찰로 고객의 안정적인 경제적 삶을 보장해주기에 금융인이 되길 희망하며 경영학을 전공했습니다. IT직무로 진로를 결정하면서 증권업에서 데이터 기반의 투자 전략을 개발하는 데 관심이 생겼으며, 이를 바탕으로 증권사 개발 역량을 통해 당사의 한국투자과 미니스탁과 뱅키스에 지원했습니다,

예측 결과

적합

확인



자기소개서 적합 예측기

출생지 : 서울

나이 : 29세

장점: 없음

단점: 없음

예측 결과

미흡

확인

아쉬운 점

1. 데이터의 양이 다소 적은점
2. 자연어 처리를 한국어가 아닌 영어로 진행한점
3. EDA 과정에서 뚜렷한 인사이트를 찾지 못함
4. 서버 구동되는 시간이 오래 걸리는 점
5. 분류 모델이 아닌 회귀 모델을 웹사이트에 탑재하지 못한점

감사합니다.