Appendix

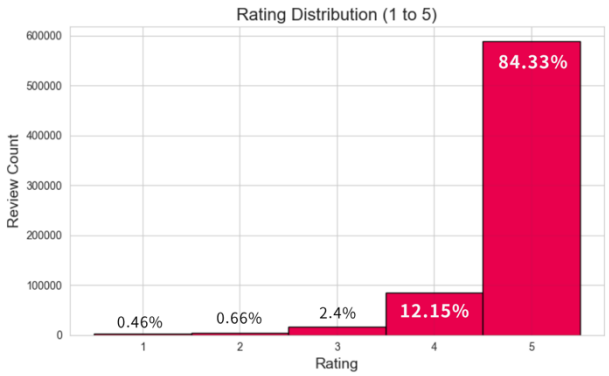
*프로젝트의 간단한 흐름 및 감성분석 모델 참고자료입니다.

<리뷰 AI score 도입을 통한 Airbnb 평점 체계 신뢰성 제고>

시켜줘! 명예호스트 팀 - (김유찬, 김훈래, 박원우, 안병민, 채주형)

[문제인식]

1. 에어비앤비 '별점'의 긍정편향 확인.

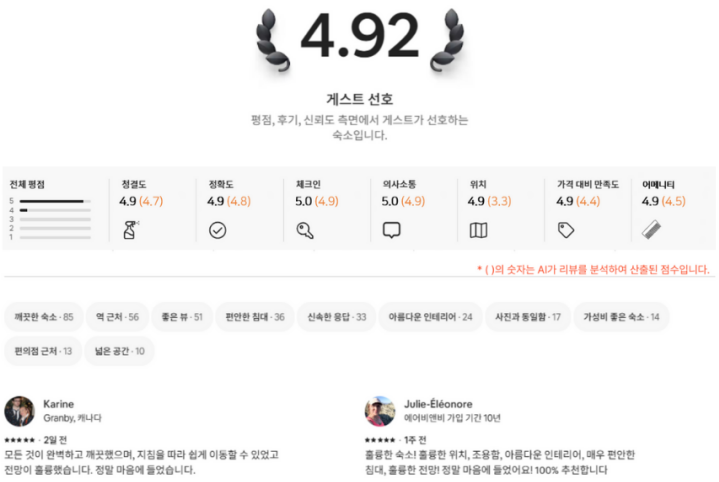


(Figure 1. 리뷰 별점 분포)

2. 별점이 비교기준으로서의 의미 저하.
3. 기대와 실제 경험 간의 괴리는 기대불일치로 만족도 및 신뢰성 저하.
4. 게스트는 다른 비교기준인 '리뷰'를 확인
    I. 많은 리뷰를 읽도록 하는 것은 탐색비용 증가로 이어짐.

[주제선정 및 프로젝트 목표]

1. 주제: 평점과 리뷰시스템 개편.
2. 프로젝트 목표: 리뷰 텍스트에서 감성점수를 구해 새로운 평점(AI score) 제시.
    I. 평점을 높게 주고 리뷰에 불만족스러운 경험 표출하는 경우를 확인.
    II. 리뷰에서 감성점수를 구해 새로운 평점(AI score)를 제시.
    → 긍정편향 완화 및 평점 신뢰성 회복.



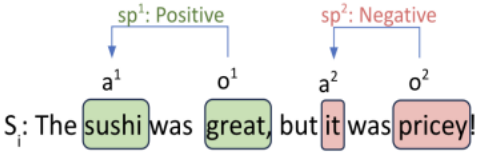
(Figure 2. AI score 구현 예시)

3. 또한, 게스트가 원하는 리뷰를 카테고리화 하여 리뷰 탐색비용을 줄이고자 함.

(Figure 3. 리뷰 카테고리화 예시)

[모델링]

1. ABSA task(프로젝트에서 사용할 NLP task)
    I. Aspect-Based Sentiment Analysis 의 약자로 항목별 감성분석을 뜻함.
    II. 리뷰 문장에서 '감성 대상', '극성값' 등을 추출하는 task.
    III. 현재 프로젝트에서는 '감성대상-속성-극성'을 뽑아내려고 함.
    IV. 예를 들어 '방이 깨끗해요'라는 리뷰는 '방-청결도-긍정'으로 추출.



Subtask	Input	Output
Aspect Term Extraction (AOOE)	S_i	a^1, a^2
Aspect Term Sentiment Classification (ATSC)	S_i+a^1, S_i+a^2	sp^1, sp^2
Aspect Sentiment Pair Extraction (ATSC)	S_i	(a^1, sp^1), (a^2, sp^2)
Aspect Oriented Opinion Extraction (ATSC)	S_i+a^1, S_i+a^2	o^1, o^2
Aspect Opinion Pair Extraction (AOPE)	S_i	(a^1, o^1), (a^2, o^2)
Aspect Opinion Sentiment Triplet Extraction (AOSTE)	S_i	(a^1, o^1, sp^1), (a^2, o^2, sp^2)
Aspect Category Opinion Sentiment Quadruplet Extraction (ACOSQE)	S_i	(a^1, c^1, o^1, sp^1), (a^2, c^2, o^2, sp^2)

Figure 1: Illustration of the six ABSA subtasks where S_i is the i^{th} sentence, a^i are the aspect terms, sp^i are the sentiment polarities and o^i is the opinion terms.

(Figure 4. ABSA Subtask 종류 예시)

2. 모델 선정 - InstructABSA
    I. 특징 요약
        A. Transformer 기반 Text-to-Text 모델.
        B. ABSA(Aspect-Based Sentiment Analysis, 특정 항목에 대한 감성분석) Task 에 맞도록 학습된 모델.
        C. Instruction(지시문)을 사용해 다양한 ABSA Task(ATE, ATSC, ASPE 등)를 단일 모델로 수행할 수 있기에 범용성/유연성이 특징.

3. 모델 개요

I. 모델 계보

A. InstructABSA 는 Transformer → T5 → Tk-Instruct → InstructABSA 계보를 기반으로 개발.

B. InstructABSA 는 Seq2Seq 처리에 최적화된 Transformer 구조.

C. T5 는 "모든 NLP task 를 텍스트 입력-출력 형태로 통일"한 프레임워크로, InstructABSA 는 이 구조를 그대로 활용하여 다양한 ABSA task 에 적합시킴.

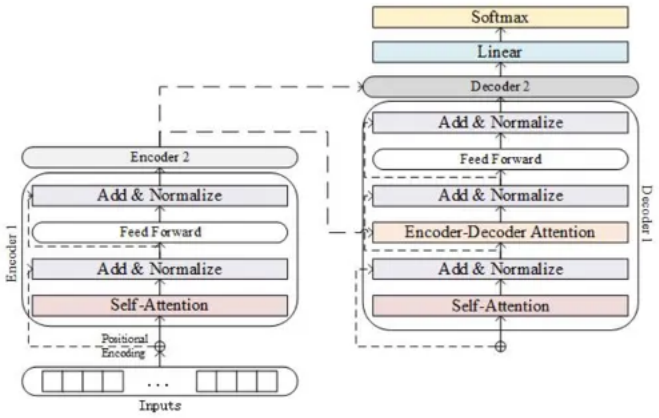
D. Tk-Instruct 는 다양한 NLP task 를 instruction 을 통해 학습하는 범용 모델로, T5 를 기반 수천 개의 instruction 을 학습시킨 것이 특징. InstructABSA 는 이 모델을 기반으로 ABSA 작업에 특화된 instruction 을 추가로 학습시킨 모델.

E. 최종적으로, InstructABSA 는 Tk-Instruct 의 instruction 기반 접근 방식을 계승하면서, ABSA 의 다양한 세부 작업(aspect extraction, sentiment classification 등)을 하나의 instruction 기반 모델로 통합 수행할 수 있도록 설계됨.

F. (요약) InstructABSA 는 Transformer 구조 위에 구축된 T5 를 기반으로 하며, Tk-Instruct 의 instruction 학습 패러다임을 차용하여 ABSA 모델로 발전한 구조.

II. 모델 구조

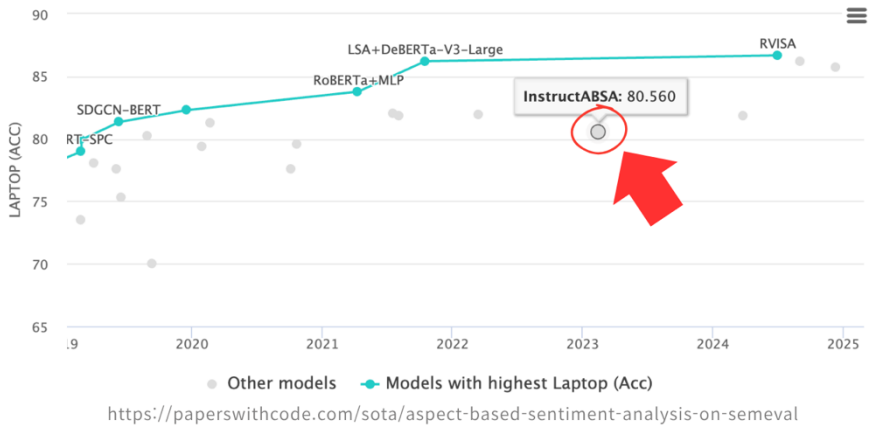
A. 모델 구조는 Transformer 에서 크게 변하지 않은 구조로 인코더 12 층, 디코더 12 층으로 구성.



(Figure 5. T5 모델 아키텍쳐 - InstructABSA도 구조는 동일)

III. 모델성능

A. 2025 년 6 월 기준 SemEval Dataset benchmark ABSA task 에서 13 등 기록.



(Figure 6 . SemEval Dataset benchmark ABSA)

[모델 Fine-tuning & 라벨링]

1. Fine-tuning

I. 공개된 모델 중 숙박 도메인으로 학습된 모델이 없고 term-category-polarity 를 추출 task 모델이 없음 → 따라서 모델 Fine-tuning 필요.

II. 학습 시 Instruction 은 [문제정의 + 긍정/부정/중립 각각 두 개의 예시]로 구성.

```
bos_instruction = """Definition: The output will be the aspect terms, their predefined categories, and sentiment polarity.
Format each as term(category):polarity.
The category must be one of the following: Cleanliness, Communication, Location, Accuracy, Check-in, Amenity, or Value.
In cases where no aspect exists and no category exists, output should be noaspectterm(noaspectcategory):none.

# Positive example 1-
# input: The room was clean and I slept very well.
# output: room(Cleanliness):positive

# Positive example 2-
# input: The host was super responsive and they provide new TV.
# output: host(Communication):positive, TV(Amenity):positive

# Negative example 1-
# input: The place was dirty and check-in was delayed.
# output: place(Cleanliness):negative, check-in(Check-in):negative

# Negative example 2-
# input: The listing photos were inaccurate.
# output: photos(Accuracy):negative

# Neutral example 1-
# input: The price was reasonable for a one-night stay.
# output: price(Value):neutral

# Neutral example 2-
# input: It would take around 10 to 15 mins walk to the subway station.
# output: subway station(Location):neutral

# Now complete the following example-
# input: """
```

Figure 7. 프로젝트에 사용된 Instruction

III. 학습 파라미터 구성

```
training_args = {
    'output_dir':model_out_path,       # 모델 경로
    'learning_rate':5e-5,              # 학습률
    'lr_scheduler_type':'cosine',      # 학습 진행 중 학습률 감소(cosine)
    'per_device_train_batch_size':8,   # 학습 배치 사이즈
    'per_device_eval_batch_size':16,   # 평가 배치 사이즈
    'num_train_epochs':8,              # 학습 에폭 수
    'weight_decay':0.01,               # 가중치 감쇠 계수 (오버피팅 방지용)
    'warmup_ratio':0.1,                # 학습 step 중 일정 비율은 학습률을 선형으로 증가
    'logging_strategy': 'epoch',       # 로그 기록 (에폭마다)
}
```

(Figure 8. 학습 파라미터)

IV. 리뷰 데이터 일부를 라벨링하여 Dataset 을 만든 후 모델 tuning.

2. 라벨링

I. 리뷰를 카테고리화 후 하여 리뷰 탐색비용을 줄이자 → 카테고리 추출 필요.

II. 따라서 현재 프로젝트는 리뷰에서 Term-category-polarity 를 추출.

III. term: 감성대상, category: 대상속성, polarity: 극성

1. term 예시: 방, 화장실, 지하철 역, 호스트, 가격…
2. category 예시: 청결도, 체크인, 의사소통, 위치, 가격대비만족도, 편의시설
3. polarity 예시: 긍정, 부정, 중립
4. [{term, category, polarity}] 형식으로 라벨링

예) The room is clean → [{term: room, category: Cleanliness, polarity: positive}]

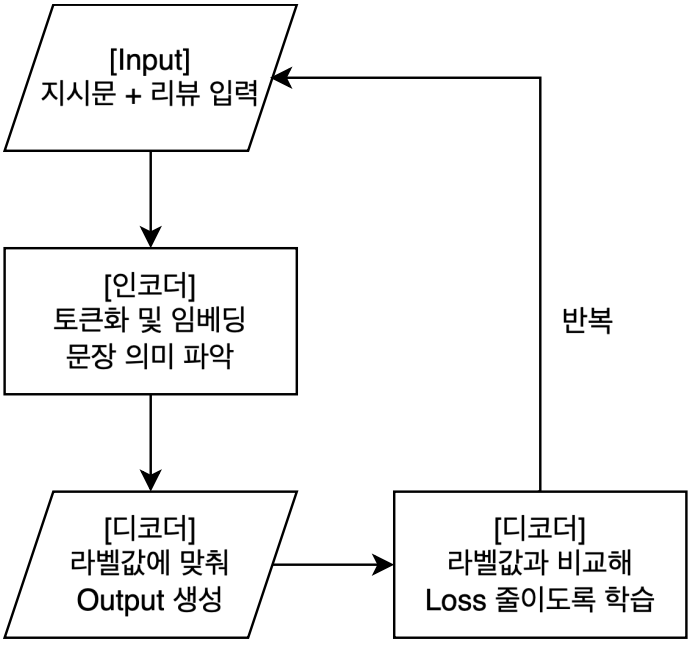
IV. 1000 개의 리뷰를 수작업으로 라벨링 진행.

1. 결측값 제거, 이모티콘 제거 등.
2. 전체 평점 분포를 반영 & 리뷰 5 자 이상인 1000 개로 구성.

Rating	
5.0	84.0
4.0	13.0
3.0	2.0
2.0	1.0
1.0	0.0

(Figure 9 . 리뷰 평점 비율)

3. Fine-tuning 학습과정



(Figure 10 . 모델학습 프로세스다이어그램)

[Input] 지시문 + 리뷰 문장 입력
ex) Definition: The output will be the aspect Terms, their predefined categories, and sentiment polarity… + The location is good
↓

[encoder] 토큰화 및 임배딩
ex) [Definition, :, The, output, … The, location, is…]
↓

[encoder] 문장 의미 파악 & context vector 디코더로 전달
↓

[decoder] 감성분석 결과 출력(Output)
ex) location(Location):positive
↓

[decoer] 정답과 비교해 loss 줄이도록 학습

[Fine-tuned 모델 성능]

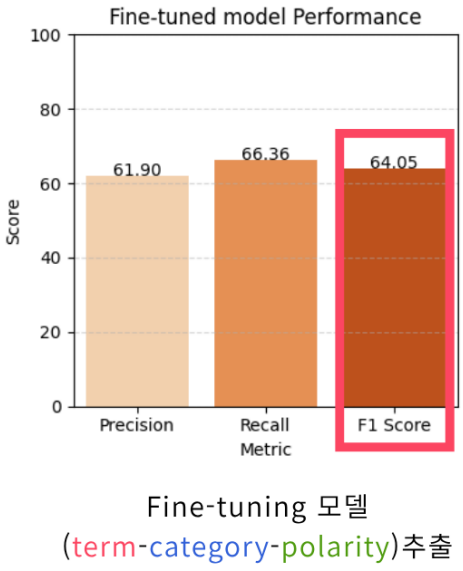
1. F1-score 기준 64.05 점.
2. 기존 InstructABSA 모델 중 프로젝트와 가장 유사한 task 성능(오른쪽 아래 그래프)과 비교 시 두 데이터 셋의 벤치마크 사이에 위치.
3. 적은 라벨링에도 유사한 성능 기록.

```
p, r, f1, _ = t5_exp.get_metrics(id_tr_labels, id_tr_pred_labels)
print('Train Precision: ', p)
print('Train Recall: ', r)
print('Train F1: ', f1)

p, r, f1, _ = t5_exp.get_metrics(id_te_labels, id_te_pred_labels)
print('Test Precision: ', p)
print('Test Recall: ', r)
print('Test F1: ', f1)
```

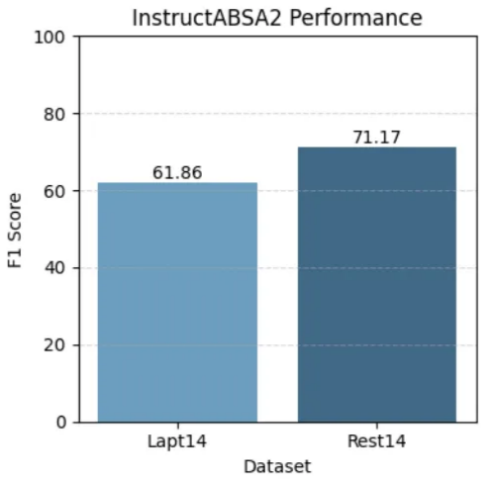
```
Train Precision:  0.813953488372093
Train Recall:  0.8309591642924976
Train F1:  0.8223684210526315
Test Precision:  0.6190476190476191
Test Recall:  0.6635730858468677
Test F1:  0.6405375139977603
```

(Figure 11. Fine-tuning 성능)



Fine-tuning 모델
(term-category-polarity)추출

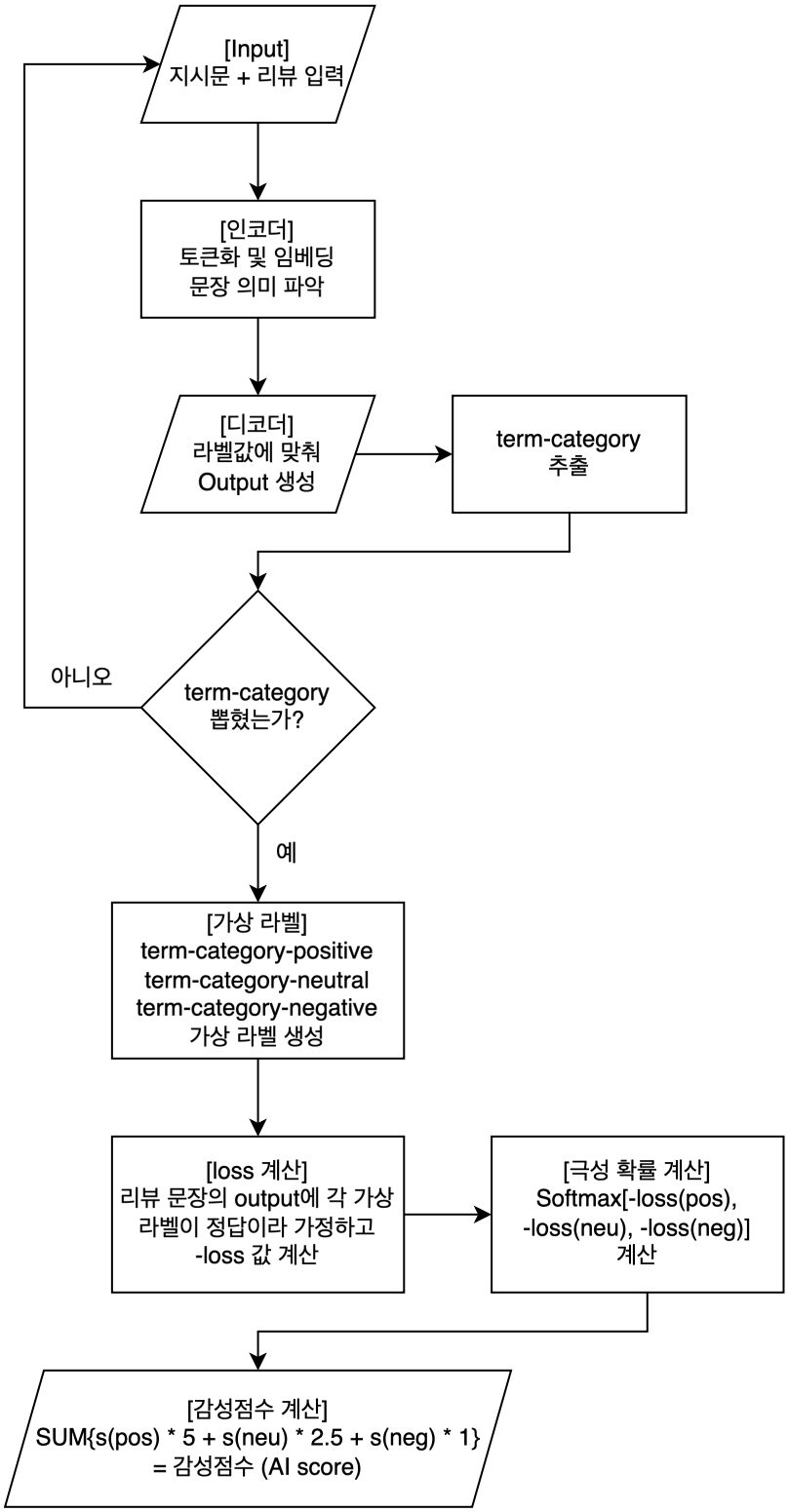
(Figure 12. Fine-tuning 성능 그래프)



InstructABSA
(term-opinion-polarity)추출

(Figure 13 . (프로젝트 유사 task) InstructABSA 성능)

[감성점수(AI score) 산출]



(Figure 14. 감성점수(AI score) 산출 프로세스다이어그램)

(감성점수 산출과정 예시)

[Input] 지시문 + 리뷰 문장 입력
ex) Definition: The output will be the aspect Terms, their predefined categories, and sentiment polarity… + The location is good
↓

[encoder] 토큰화 및 임배딩
ex) [Definition, :, The, output, … The, location, is…]
↓

[encoder] 문장 의미 파악 & context vector 디코더로 전달
↓

[decoder] 감성분석 결과 출력(Output)
ex) location(Location):positive
↓

감성분석 결과 중 term-category 추출
ex) location(Location)
↓

추출한 쌍에 positive, negative, neutral 을 붙혀 가상 라벨값을 생성
ex) location(Location):positive
location(Location):neutral
location(Location):negative
↓

각 가상 라벨이 정답이라 가정하고 리뷰 문장의 output 과 비교 ⇒ -loss 값 계산
ex) -loss(location(Location):positive) = -0.3
-loss(location(Location):neutral) = -1.6
-loss(location(Location):negative) = -2.3
↓

각 -loss 값에 Softmax 를 씌워 확률 추출
ex) Softmax[-0.3, -1.6, -2.3] = [0.7, 0.2, 0.1]
각 확률의 의미 = positive 일 확률 0.7, neutral 일 확률 0.2, negative 일 확률 0.1
↓

각 극성에 가중치 부여 및 가중합 계산 ⇒ 감성점수 산출
ex) 극성값 가중치(positive: 5, neutral: 2.5, negative:1)
SUM(0.7 * 5 + 0.2 * 2.5 + 0.1 * 1) ⇒ 감성점수: 4.1
↓

모든 리뷰 문장에 대해 반복하여 감성점수(AI score) 추출

```
input_sentence = "The location is terrible but host is really responsive"
result = analyze_aspects(input_sentence)

pprint(result)
```

```
{'host.Communication': {'probs': {'negative': 0.2351,
                                  'neutral': 0.2647,
                                  'positive': 0.5002},
                        'score': 4.03},
 'location.Location': {'probs': {'negative': 0.5177,
                                 'neutral': 0.2618,
                                 'positive': 0.2205},
                       'score': 2.63}}
```

(Figure 15. 감성점수 산출 예시)

[결과]

1. Fine-tuned 모델로 총 22 만개 리뷰에 대해 감성점수 산출.
   ↓(22 만개 리뷰 선정 방식)

   I. 기존 리뷰 데이터에서 결측값 제거, 이모티콘 제거, 리뷰 길이 5 이상 등 전처리.
   II. 기존 리뷰 데이터의 세부 별점(청결도, 위치, 정확도…)은 결측값이었음.
   → 리뷰 데이터 숙소에 대해 세부 별점을 크롤링 진행한 후 기존 리뷰와 merge.(30 만개 리뷰)
   III. 리뷰 100 개 이상인 숙소만 선정.(→ 총 22 만개 리뷰)

2. (숙소기준) AI_score vs 기존 별점 분포 비교

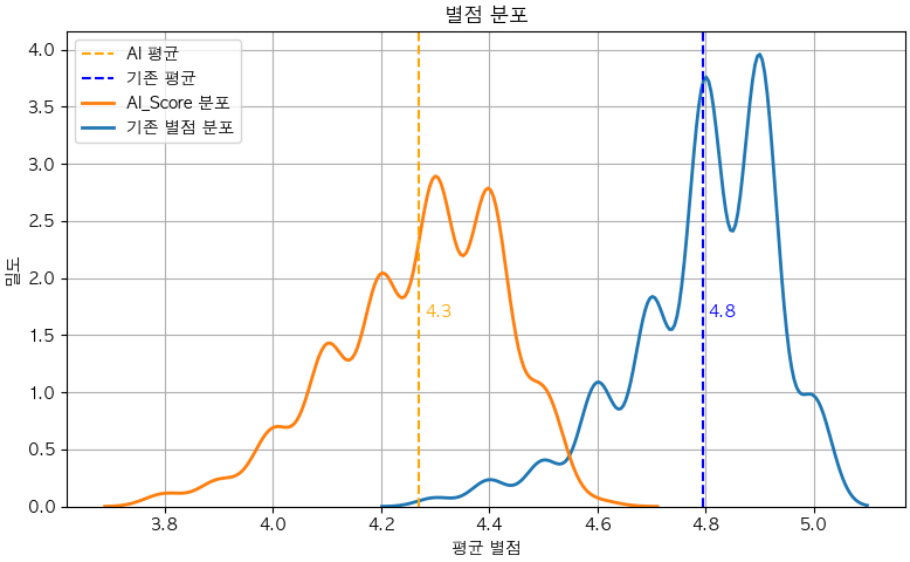
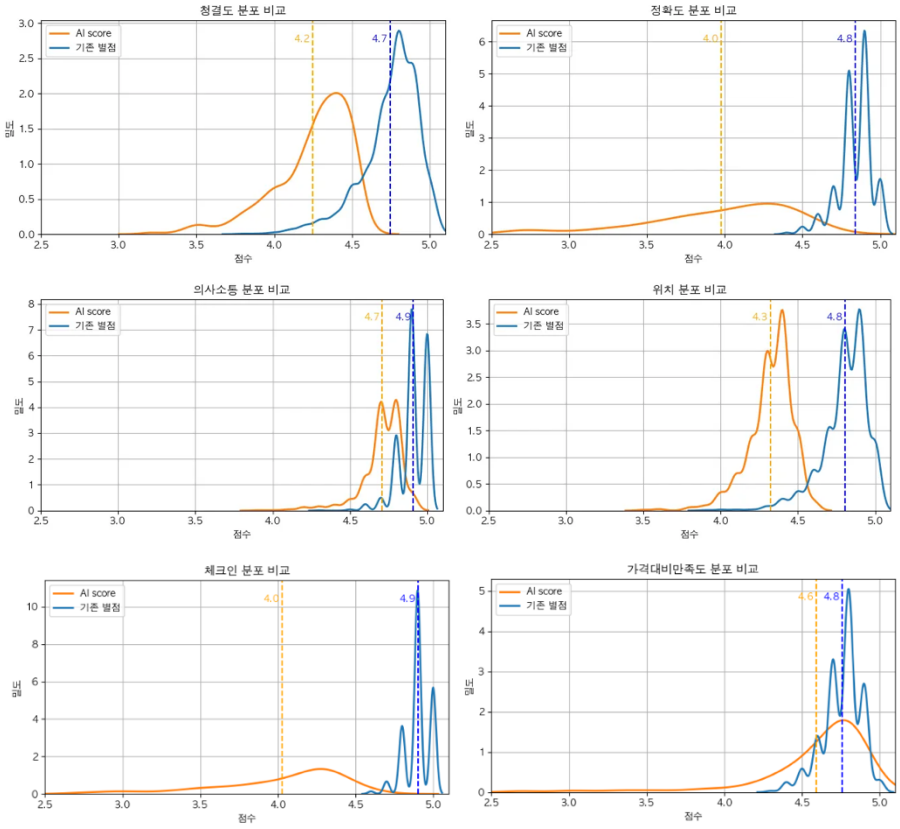


Figure 16 . (숙소기준) AI_score VS 기존 별점 분포 그래프

I. 기존 별점 평균: 4.8 점
II. AI score 평균: 4.3 점
III. AI score 분포는 기존 별점 분포보다 넓게 분산된 것을 확인.

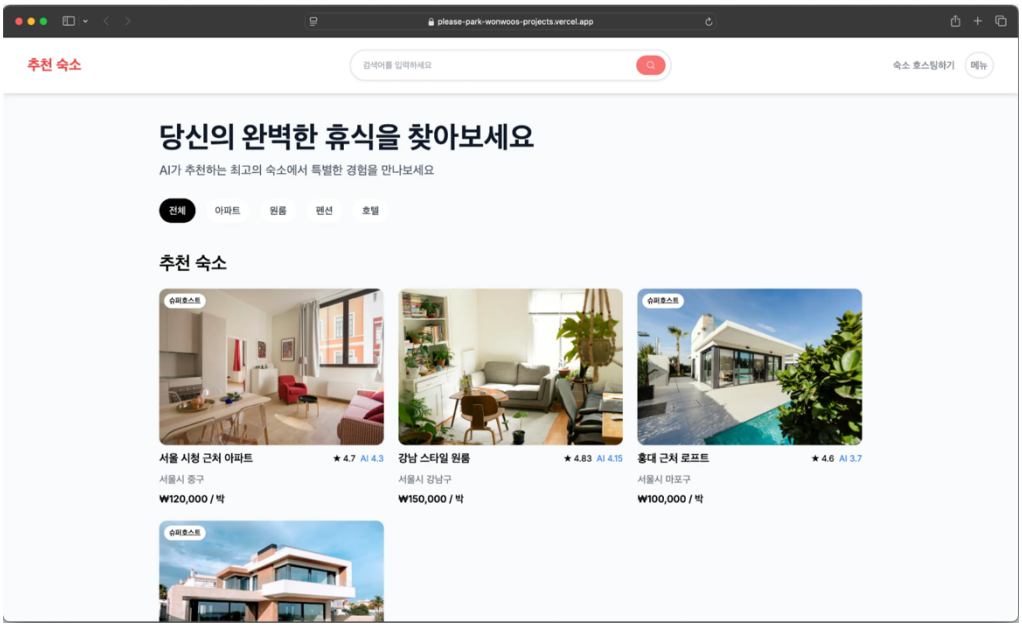
IV. 각 세부항목별 분포 비교



(Figure 17. 세부항목별 분포 비교)

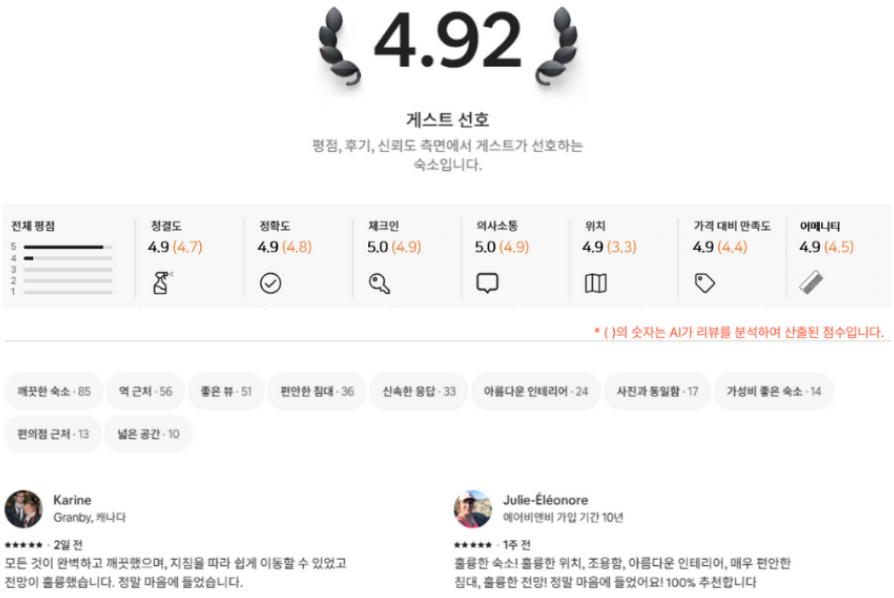
[웹페이지 구현]

1. 현재 구현 웹페이지



(Figure 18. 실제 구현 페이지 https://please-park-wonwoos-projects.vercel.app/)

2. 최종 구현 페이지 예시



(Figure 19. AI score 구현 예시)

(Figure 20. 리뷰 카테고리화 예시)

[기대효과 및 한계점]

1. 기대효과

   [게스트]

   1. 리뷰기반 신뢰성 있는 평점(편향 완화된)을 활용해 숙소 비교 가능.
   2. 방대한 리뷰를 전부 읽을 필요 없이 게스트가 세부 정보를 빠르게 파악할 수 있어 의사결정 피로도 감소.
   3. 세분화된 AI score 와 리뷰 필터링으로 게스트가 원하는 항목에 대한 리뷰를 쉽게 파악할 수 있어 만족도 증가.

   [호스트]

   4. AI score 로 객관적인 리뷰/평점을 확인해 빠르고 구체적인 정보로 피드백 가능.
   5. 강점은 새로운 마케팅 요소, 약점은 개선점.

   [에어비앤비]

   6. 차별화된 유용한 리뷰/평점 시스템 제공으로 플랫폼 신뢰도 및 경쟁력 확대.

2. 한계점
   1. 시간 & 자원 제약으로 적은 라벨링 데이터 확보 → Fine-tuned 모델이 일정 수준의 성능은 넘었지만, 더 많은 라벨링을 수행해 성능을 더 높이지 못한 아쉬움.
   2. 웹페이지 구현 시 기술적인 제약이 있어 A/B Test 를 실제 페이지로 진행하지 못함.
   3. 실제 모델 output 데이터와 웹페이지를 연동해 서비스를 구현하지 못함.

[참고문헌]

1. Zervas, Georgios and Proserpio, Davide and Byers, John, A First Look at Online Reputation on Airbnb, Where Every Stay is Above Average (December 28, 2020). Available at SSRN: A first look at online reputation on Airbnb, where every stay is above average(Schuckert et al., 2020
2. Zolbanin, H. M., & Wynn, D. (2022). From star rating to sentiment rating: using textual content of online reviews to develop more effective reputation systems for peer-to-peer accommodation platforms. Journal of Business Analytics, 6(2), 127-139. https://doi.org/10.1080/2573234X.2022.2122880
3. Bridges, J., & Vásquez, C. (2016). If nearly all Airbnb reviews are positive, does that make them meaningless? Current Issues in Tourism, 21(18), 2057-2075. https://doi.org/10.1080/13683500.2016.1267113
4. Rudolph, S. (2015). The Impact of Online Reviews on Customers' Buying Decisions [Infographic]. Business 2 Community. https://www.business2community.com/infographics/impact-online-reviews-customers-buying-decisions-infographic-01280945
5. Wang, Mingye, Pan Xie, Yao Du, and Xiaohui Hu. 2023. "T5-Based Model for Abstractive Summarization: A Semi-Supervised Learning Approach with Consistency Loss Functions" Applied Sciences 13, no. 12: 7111. https://doi.org/10.3390/app13127111
6. https://paperswithcode.com/sota/aspect-based-sentiment-analysis-on-semeval
7. https://www.airbnb.co.kr/rooms/682756408556877714?check_in=2025-08-01&check_out=2025-08-03&photo_id=1938686872&source_impression_id=p3_1749266293_P3OJJfuordThwVBK&previous_page_section_name=1000