

# 한국인공지능올림피아드 (KOAI) 2026 예시문제

시행 | 2026년 6월 13일 (토), 10:00 ~ 16:00

주최·주관 | 한국정보기술진흥원

연계 | 국제인공지능올림피아드 국제학술위원회

## < 응시자 유의사항 >

- 온라인 응시는 응시자가 원하는 독립된 공간 (1인 1실, 자택 권고)에서 개인 컴퓨터로 응시하시면 됩니다. 응시프로그램에 접속하실 수 있는 로그인 정보는 수험표에서 확인하실 수 있으며, 수험표를 인쇄해서 준비해 주시기 바랍니다.
- 대회 시작 대기 및 응시 중 응시자 본인이 아닌 다른 사람이 화면에 보이거나 목소리가 들리는 경우 부정행위로 처리됩니다. 또한, 응시 중에는 화장실을 갈 수 없습니다.
- 대회 시간 동안 기기의 전원이 원활하며 안정적인 인터넷이 제공되는 응시용 기기(PC 또는 노트북 1대)와 감독용 기기(스마트폰 또는 태블릿 PC 1대, 노트북이나 PC는 불가능)가 필요합니다.
- 응시 중에는 2개 기기 모두 방해금지모드를 설정하는 것을 권장합니다.
- 응시용 기기 및 감독용 기기 모두 방전에 대비하여 충전기를 연결해 놓으시는 것을 권장합니다.
- 여러 인원이 한 온라인 시험장에서 응시하는 만큼 다른 참가자들을 방해하는 행동 (소란, 소음 등)은 감독관에게 경고받을 수 있으며, 응시자는 이에 대한 조치를 취해야 합니다. 지속적인 경고를 받았음에도 방해하는 경우, 부정행위로 처리될 수 있습니다.
- 대회 중 예기치 못한 돌발 상황으로 인해 감독용 기기 또는 응시용 기기에 (정전, 배터리 부족, 과열 등) 문제 발생 시 한국정보기술진흥원 카카오톡 채널로 학생이 직접 연락해야 하며 대처 방법을 안내해 드린 후, 대회 종료 후 소명 절차를 거쳐 최종 부정행위 여부를 판단합니다.
- 응시 프로그램 이용 중 오류 발생 시에는 손을 들어 기다려 주시면 감독관 확인 후에 적절한 조치를 취해드리도록 하겠습니다.

실제 KOAI는 인공물이 아닌 응시 프로그램에서 진행됩니다.

본 예시문제는 학습 참고용으로만 사용하시기 바라며, **정답은 공개하지 않습니다.**

본 예시문제는 IOAI 국제학술위원회에서 공시한 출제 기준에 따라 수록한 예시문항입니다. 출제 수준은 동 위원회에서 공시한 수준을 고려하여 **고등부를 기준으로 작성**되었습니다.

문제는 **난이도 순으로 정렬되어 있지 않습니다.**

실제 KOAI는 **고등부 선택형 80문항, 서술형 10문항 (6시간) / 중등부 선택형 40문항, 서술형 5문항 (3시간)**으로 구성되어 있습니다.

**고등부의 출제 범위는 1~4과목이며, 중등부는 1~2과목만 출제됩니다.**

## 선택형 (60문항)

### 1과목 기초 역량 및 고전 머신러닝

1. 다음 파이썬 코드의 실행 결과로 옳은 것은?

```
def f(x, lst=[]):  
    lst.append(x)  
    return lst  
  
a = f(1)  
b = f(2)  
print(b)
```

- [2]
- [1, 2]
- [1]
- TypeError 발생

2. numpy 배열  $a = \text{np.array}([[1, 2, 3], [4, 5, 6]])$ 에 대한 설명으로 옳지 않은 것은?

- $a.\text{shape}$ 는 (2, 3)이다.
- $a.\text{sum}(\text{axis}=0)$ 의 결과는 [5, 7, 9]이다.
- $a.\text{sum}(\text{axis}=1)$ 의 결과는 [6, 15]이다.
- $a.\text{reshape}(3, 2)$ 는 원본 배열의 메모리를 항상 새로 할당한다.

3. 다음 중 두 연속 변수 간의 분포와 관계를 동시에 시각화하기에 가장 적합한 seaborn 함수는?

- ① sns.barplot
- ② sns.lineplot
- ③ sns.jointplot
- ④ sns.countplot

4. scikit-learn의 estimator 인터페이스에 대한 설명으로 옳은 것은?

- ① fit() 메서드는 비지도 학습 모델에서는 사용할 수 없다.
- ② predict()는 모든 모델에서 확률값을 반환한다.
- ③ transform()은 데이터를 변환하는 변환기(transformer) 클래스에서 제공된다.
- ④ score() 메서드는 회귀 모델에서 항상 RMSE를 반환한다.

5. PyTorch의 autograd에 관한 설명으로 옳은 것은?

- ① 모든 텐서는 기본적으로 requires\_grad=True로 생성된다.
- ② loss.backward() 호출 후 .grad 속성에 누적되며, 학습 루프에서는 직접 zero\_grad()로 초기화해야 한다.
- ③ torch.no\_grad() 블록 내에서도 그래디언트가 자동으로 계산된다.
- ④ detach()는 텐서의 값을 0으로 만든다.

6. shape가 (4, 3, 32, 32)인 PyTorch 텐서 x에 대해, 0번 축과 1번 축을 교환하여 (3, 4, 32, 32) shape로 만드는 연산은?

- ① x.view(3, 4, 32, 32)
- ② x.reshape(3, 4, 32, 32)
- ③ x.transpose(0, 1)
- ④ x.squeeze(0)

7. GPU에서의 모델 학습과 관련된 설명으로 옳은 것은?

- ① "CUDA out of memory" 오류는 일반적으로 배치 크기가 너무 크거나 메모리 누수가 원인이다.
- ② "Expected all tensors to be on the same device" 오류는 PyTorch 버전 호환성 문제이다.
- ③ GPU 학습은 데이터 전송 비용이 무시할 수 있을 정도로 작아 항상 CPU보다 빠르다.
- ④ .to('cuda')는 모델 가중치를 GPU로 옮기지 않고 평가 모드로만 전환한다.

8. 선형회귀에서 정규방정식  $\theta = (X^T X)^{-1} X^T y$ 가 유일한 해를 갖기 위한 조건으로 가장 옳은 것은?

- ①  $X$ 의 행 수가 열 수보다 작아야 한다.
- ②  $X^T X$ 가 가역(invertible) 행렬이어야 한다.
- ③  $y$ 가 정규분포를 따라야만 한다.
- ④  $X$ 의 모든 원소가 양수여야 한다.

9. 이진 분류용 로지스틱 회귀에서 일반적으로 사용되는 손실 함수는?

- ① 평균제곱오차
- ② 이진 교차엔트로피
- ③ 힌지 손실
- ④ Huber 손실

10. L1 정규화(Lasso)와 L2 정규화(Ridge)의 차이로 옳은 것은?

- ① L1은 가중치를 정확히 0으로 만들 수 있어 특성 선택 효과가 있고, L2는 가중치를 작게 만들지만 0으로 만들기는 어렵다.
- ② L2는 가중치를 0으로 만드는 효과가 있고, L1은 작게 만들지만 한다.
- ③ L1과 L2 모두 가중치를 항상 0으로 만든다.
- ④ L1과 L2는 수학적으로 완전히 동일하다.

11. K-NN 분류기에서 K값에 따른 일반적인 동작에 대한 설명으로 옳은 것은?

- ① K가 작을수록 결정 경계가 부드러워진다.
- ② K=1일 때 학습 데이터에 대해 100% 정확도를 보이지만 과대적합 위험이 크다.
- ③ K가 매우 크면 과대적합이 심해진다.
- ④ K-NN은 학습 단계에서 모델 파라미터를 추정하는 매개변수적 방법이다.

12. 분류 의사결정 트리에서 분기 기준으로 일반적으로 사용되는 지표가 아닌 것은?

- ① 지니 불순도(Gini impurity)
- ② 엔트로피(Entropy) / 정보이득(Information Gain)
- ③ 분산(Variance)
- ④ 코사인 유사도(Cosine similarity)

13. 배깅(Bagging)과 부스팅(Boosting)에 대한 설명으로 옳지 않은 것은?

- ① Bagging은 약한 학습기를 병렬적으로 학습할 수 있다.
- ② Boosting은 이전 학습기의 오차를 다음 학습기가 보완하도록 순차적으로 학습한다.
- ③ Random Forest는 Bagging과 특성 무작위 추출을 결합한 방법이다.
- ④ Gradient Boosting은 모든 트리를 동시에 학습시켜 평균을 낸다.

14. SVM의 마진(margin)과 커널 트릭(kernel trick)에 대한 설명으로 옳은 것은?

- ① SVM은 결정 경계와 가장 가까운 데이터 포인트들 간의 거리를 최소화한다.
- ② 커널 트릭은 고차원 특성을 명시적으로 계산해야 한다.
- ③ RBF 커널은 무한 차원 특성 공간에 대응되는 효과를 낼 수 있다.
- ④ 소프트 마진 SVM에서 C값이 클수록 마진이 넓어지고 오분류 허용이 커진다.

15. K-평균(K-means) 알고리즘의 특성으로 옳지 않은 것은?

- ① 군집의 수 K를 사전에 지정해야 한다.
- ② 초기 중심점에 따라 결과가 달라질 수 있다 (k-means++ 등으로 완화 가능).
- ③ 비볼록(non-convex) 형태의 군집을 잘 찾아낸다.
- ④ 각 군집은 평균(centroid)으로 표현된다.

16. PCA(주성분 분석)에 대한 설명으로 옳은 것은?

- ① PCA의 주성분은 데이터 공분산 행렬의 고유벡터 중 큰 고유값에 해당하는 것이다.
- ② PCA는 비선형 차원 축소 기법이다.
- ③ PCA는 항상 분류 정확도를 향상시킨다.
- ④ PCA를 적용하기 전에 데이터를 표준화할 필요는 전혀 없다.

17. t-SNE와 UMAP에 대한 설명으로 옳지 않은 것은?

- ① 둘 다 고차원 데이터를 시각화 목적으로 저차원에 임베딩하는 데 자주 쓰인다.
- ② t-SNE는 국소 구조 보존에 강하지만 전역 구조 보존은 약할 수 있다.
- ③ UMAP은 t-SNE보다 일반적으로 빠르고 어느 정도 전역 구조도 보존한다.
- ④ t-SNE 임베딩 좌표의 절댓값과 군집 간 절대 거리는 항상 의미가 있다.

18. DBSCAN의 특성으로 옳은 것은?

- ① 군집의 수 K를 사전에 지정해야 한다.
- ② 노이즈 포인트(어떤 군집에도 속하지 않는 포인트)를 식별할 수 있다.
- ③ 비볼록 형태의 군집은 찾을 수 없다.
- ④ 데이터 밀도가 균일하지 않을수록 항상 가장 좋은 결과를 낸다.

19. 양성(positive) 표본이 매우 적은 클래스 불균형 이진 분류 문제에서, 단순 정확도(accuracy)보다 적절한 지표 조합은?

- ① 정확도와 학습 시간
- ② 정밀도, 재현율, F1 점수
- ③ MSE와 MAE
- ④  $R^2$ 와 결정계수

20. 다음 중 과대적합(overfitting)의 전형적인 신호로 가장 옳은 것은?

- ① 학습 손실은 낮지만 검증 손실이 학습 손실보다 훨씬 높다.
- ② 학습 손실과 검증 손실이 모두 매우 높다.
- ③ 학습 손실이 검증 손실보다 항상 더 높다.
- ④ 손실 곡선이 평탄하게 유지된다.

21. 다음 중 하이퍼파라미터 탐색 전략에 대한 설명으로 옳지 않은 것은?

- ① 그리드 서치(Grid Search)는 차원이 늘어날수록 비용이 기하급수적으로 증가한다.
- ② 랜덤 서치(Random Search)는 일반적으로 그리드 서치보다 적은 시도로도 좋은 영역을 찾는 경향이 있다.
- ③ 베이지안 최적화는 이전 시도 결과를 활용해 다음 시도를 선택한다.
- ④ 테스트 셋에서 직접 하이퍼파라미터를 튜닝하는 것이 가장 정확한 방법이다.

22. k-fold 교차검증(k-fold cross-validation)에 대한 설명으로 옳은 것은?

- ① 데이터를 k개의 폴드로 나누어 매번 1개를 검증, 나머지를 학습에 사용하고 k번 반복한다.
- ② k 값은 데이터 크기에 관계없이 항상 2가 최적이다.
- ③  $k=N$ (LOOCV)일수록 일반화 오차의 추정이 항상 더 정확해진다.
- ④ 시계열 데이터에서는 임의 분할이 가장 권장된다.

23. ROC 곡선과 AUC에 대한 설명으로 옳은 것은?

- ① ROC 곡선의 x축은 정밀도, y축은 재현율이다.
- ② AUC=0.5는 무작위 분류기와 동일한 수준임을 의미한다.
- ③ AUC는 항상 0.5에서 1.0 사이의 값을 갖는다.
- ④ ROC 곡선은 회귀 모델 평가에 표준적으로 사용된다.

24. 다음 중 피처 엔지니어링 기법으로 보기 가장 어려운 것은?

- ① 범주형 변수의 원-핫 인코딩(one-hot encoding)
- ② 시계열에 대한 슬라이딩 윈도우(sliding window) 통계량 추출
- ③ PCA를 이용한 차원 축소 후 결과를 새로운 피처로 사용
- ④ 학습률(learning rate) 스케줄링

25. 가변 길이 시퀀스를 미니배치로 학습할 때 일반적으로 사용하는 전처리 기법은?

- ① 모든 시퀀스를 가장 짧은 길이에 맞춰 자른다.
- ② 패딩(padding)과 마스크(masking)를 사용해 같은 길이의 텐서로 묶고, 손실 계산에서 패딩 토큰을 무시한다.
- ③ 시퀀스 길이의 평균값을 가중치로 곱한다.
- ④ 시퀀스를 이미지처럼 2D로 강제 변환한다.

2과목  
신경망 및 딥러닝

26. 단일 퍼셉트론(선형 분류기)으로 학습할 수 없는 문제는?

- ① AND
- ② OR
- ③ NOT
- ④ XOR

27. 경사하강법에서 학습률(learning rate)이 너무 큰 경우 일반적으로 나타나는 현상은?

- ① 손실이 단조 감소하여 빠르게 수렴한다.
- ② 손실이 발산하거나 진동하며 수렴하지 못한다.
- ③ 학습 속도와 학습률은 무관하다.
- ④ 모델 파라미터가 항상 0이 된다.

28. 역전파(backpropagation)에 대한 설명으로 옳은 것은?

- ① 출력층에서 입력층 방향으로 연쇄법칙(chain rule)을 사용해 그래디언트를 계산한다.
- ② 입력층에서 출력층 방향으로 그래디언트를 계산한다.
- ③ 미분 없이 무작위 탐색만으로 가중치를 갱신한다.
- ④ 활성화 함수가 비미분 가능해도 항상 잘 작동한다.

29. ReLU 활성화 함수의 특징으로 옳지 않은 것은?

- ① 양의 입력에 대해 그래디언트가 1이라 깊은 신경망에서 그래디언트 소실 문제가 시그모이드보다 덜하다.
- ② 음의 입력에서 출력이 0이며, 일부 뉴런이 영구적으로 비활성화되는 "Dying ReLU" 문제가 발생할 수 있다.
- ③ 출력이 (0, 1) 범위로 제한된다.
- ④ 계산이 매우 단순하여 효율적이다.

30. 다음 중 손실 함수의 일반적인 사용처로 옳은 것은?

- ① 회귀: Cross Entropy
- ② 다중 클래스 분류: Softmax + Categorical Cross Entropy
- ③ 이진 분류: MSE
- ④ 이상치(outlier)에 매우 민감한 회귀: MAE

31. MLP에서 비선형 활성화 함수가 없으면 발생하는 현상으로 가장 옳은 것은?

- ① 깊이를 아무리 깊게 해도 단일 선형 변환과 동등해진다.
- ② 항상 더 좋은 성능을 낸다.
- ③ 그래디언트 소실 문제가 사라진다.
- ④ 자동으로 합성곱 신경망과 동등해진다.

32. 임베딩(embedding)의 핵심 아이디어로 가장 옳은 것은?

- ① 이산 토큰이나 고차원 입력을 의미 정보를 담은 저차원 연속 벡터로 사상한다.
- ② 데이터를 항상 1차원 스칼라로 압축한다.
- ③ 결측치를 처리하기 위한 통계적 대체값이다.
- ④ 손실 함수의 한 종류이다.

33. CNN에서 풀링(pooling)의 역할로 옳지 않은 것은?

- ① 특징 맵(feature map)의 공간 해상도를 줄여 계산량을 감소시킨다.
- ② 작은 평행이동(translation)에 대한 어느 정도의 불변성을 제공한다.
- ③ 학습 가능한 가중치를 추가한다.
- ④ 과대적합을 어느 정도 완화하는 효과를 줄 수 있다.

34. Scaled Dot Product Attention  
 $\text{softmax}(QK^T/\sqrt{d_k})V$  에서  $\sqrt{d_k}$ 로 나누는 이유  
가장 옳은 것은?

- ① 출력의 차원을 줄이기 위해서이다.
- ② 키 벡터의 차원에 비례해 내적값이 너무 커져 softmax가 포화되는 것을 방지하기 위해서이다.
- ③ 그래디언트를 0으로 만들기 위해서이다.
- ④ 행렬 곱이 정의되도록 차원을 맞추기 위해서이다.

35. 표준 트랜스포머 인코더 구조의 구성 요소가 아닌 것은?

- ① 멀티 헤드 셀프 어텐션
- ② 위치 정보(positional encoding/embedding)
- ③ 잔차 연결(residual connection)과 LayerNorm
- ④ 합성곱 풀링(convolutional pooling) 레이어

36. 오토인코더(autoencoder)에 대한 설명으로 옳은 것은?

- ① 입력을 잠재 공간으로 압축했다가 다시 복원하는 비지도 학습 모델이다.
- ② 분류기 학습을 위해 항상 레이블이 필요하다.
- ③ 디코더 없이 인코더만으로 정의된다.
- ④ 잠재 차원을 입력보다 항상 크게 설정해야 한다.

37. 미니배치 경사하강법(mini-batch SGD)이 전체 배치 경사하강법보다 일반적으로 갖는 장점은?

- ① 메모리 효율성과 어느 정도의 노이즈로 인한 일반화 향상 효과
- ② 항상 더 작은 학습 손실
- ③ 결정론적 그래디언트 추정
- ④ 학습률 조정이 불필요해짐

38. Adam과 AdamW의 차이로 가장 옳은 것은?

- ① Adam은 모멘텀을 사용하지 않는다.
- ② AdamW는 가중치 감쇠(weight decay)를 그래디언트에 더하지 않고 파라미터 업데이트 시 분리해서 적용한다.
- ③ AdamW는 일차 모멘트만 추정한다.
- ④ Adam과 AdamW는 완전히 동일한 알고리즘이다.

39. 학습률 스케줄링(learning rate scheduling)에 대한 설명으로 옳지 않은 것은?

- ① 학습 초반에 워밍업(warmup)을 두어 큰 학습률에 의한 불안정성을 줄일 수 있다.
- ② 코사인 스케줄(cosine schedule)은 학습 후반부에 학습률을 점진적으로 감소시킨다.
- ③ 학습률을 단계적으로 줄이는 step decay는 흔한 방법이다.
- ④ 모든 모델에서 학습률은 항상 고정된 값으로 두는 것이 가장 좋다.

40. 드롭아웃(Dropout)에 대한 설명으로 옳은 것은?

- ① 학습 시 일부 뉴런을 확률적으로 비활성화하여 과대적합을 줄인다.
- ② 추론 시에도 동일한 확률로 뉴런을 비활성화한다.
- ③ 완전연결층에는 적용할 수 없다.
- ④ 학습 손실을 항상 낮춘다.

41. 신경망의 가중치 초기화에 대한 설명으로 옳은 것은?

- ① 모든 가중치를 0으로 초기화하면 대칭성 문제로 같은 층의 뉴런들이 동일한 함수를 학습한다.
- ② Xavier(Glorot) 초기화는 ReLU에 가장 적합하다.
- ③ He 초기화는 시그모이드에 가장 적합하다.
- ④ 초기화 방식은 학습에 거의 영향을 미치지 않는다.

42. 배치 정규화(Batch Normalization)에 대한 설명으로 옳지 않은 것은?

- ① 미니배치 단위로 평균과 분산을 계산해 활성화화를 정규화한다.
- ② 학습 안정성과 수렴 속도를 향상시키는 경향이 있다.
- ③ 추론 시에는 학습 중 계산해 둔 이동평균(running mean/var)을 사용한다.
- ④ 배치 크기에 무관하며 매우 작은 배치에서 더 잘 작동한다.

43. LoRA와 같은 PEFT(parameter-efficient fine-tuning)의 핵심 아이디어로 가장 옳은 것은?

- ① 사전학습 모델의 모든 파라미터를 항상 새로 학습한다.
- ② 큰 가중치 행렬의 변화량을 저랭크 행렬로 근사하여 일부 파라미터만 학습한다.
- ③ 학습 데이터를 압축하는 기법이다.
- ④ 모델의 출력층 자체를 제거한다.

### 3과목 컴퓨터 비전

44. 합성곱(convolution) 레이어의 특성으로 옳지 않은 것은?

- ① 동일한 필터를 입력 위치에 걸쳐 공유함으로써 평행이동 등변성(translation equivariance)을 갖는다.
- ② 동일한 입력 크기에서 완전연결층보다 일반적으로 파라미터 수가 적다.
- ③ 필터(커널) 크기, 스트라이드, 패딩으로 출력 크기를 조절할 수 있다.
- ④ 모든 입력 위치마다 별도의 필터를 사용한다.

45. ImageNet 등 사전학습 모델을 작은 이미지 데이터셋에 적용할 때 일반적으로 가장 권장되는 방법은?

- ① 처음부터 무작위 가중치로 학습한다.
- ② 사전학습 가중치로 초기화한 후 마지막 분류기 또는 일부 레이어를 미세조정한다.
- ③ 사전학습 모델은 사용하지 않고 더 큰 모델을 만든다.
- ④ 데이터 증강은 사용하지 않는다.

46. YOLO 계열과 DETR의 차이로 가장 옳은 것은?

- ① YOLO는 그리드 기반의 단일 단계(single-stage) 탐지기이며, DETR은 트랜스포머와 집합 예측(set prediction)을 사용한다.
- ② DETR은 그리드 기반이고 YOLO는 트랜스포머 기반이다.
- ③ 두 모델 모두 영역 제안 네트워크(RPN)를 핵심 구성 요소로 사용한다.
- ④ YOLO는 항상 비교 불가능할 정도로 정확도가 낮다.

47. U-Net의 핵심 구조로 옳은 것은?

- ① 인코더-디코더 구조와 동일 해상도 단위 간 스킵 연결(skip connection)
- ② 트랜스포머 디코더만으로 구성된 구조
- ③ 풀링 레이어가 전혀 없는 구조
- ④ 출력이 항상 단일 스칼라 값인 구조

48. ResNet의 핵심 아이디어로 가장 옳은 것은?

- ① 잔차 연결(residual connection)을 통해 매우 깊은 네트워크의 학습을 용이하게 한다.
- ② 합성곱 없이 완전연결층만 쌓는다.
- ③ 풀링을 일절 사용하지 않는다.
- ④ 활성화 함수로 시그모이드만 사용한다.

49. 다음 중 일반적인 이미지 증강(image augmentation) 기법이 아닌 것은?

- ① 무작위 좌우 반전(random horizontal flip)
- ② 무작위 자르기 및 크기 조정(random resized crop)
- ③ 색상 지터(color jitter)
- ④ 이미지에 픽셀별 정답 레이블을 직접 표시(annotation)

50. GAN(Generative Adversarial Network)에 대한 설명으로 옳은 것은?

- ① 생성자(G)와 판별자(D)가 서로 경쟁하며 학습하는 미니맥스 게임으로 정의된다.
- ② 판별자는 진짜 데이터를 생성하는 역할을 한다.
- ③ GAN은 판별자 없이 생성자만 학습한다.
- ④ GAN은 항상 안정적으로 수렴한다.

51. 비전 분야의 자기지도 학습(self-supervised learning)에 해당하는 방법은?

- ① ImageNet 클래스 라벨을 이용한 분류 학습
- ② SimCLR, MoCo와 같은 대조 학습(contrastive learning) 또는 MAE와 같은 마스크 기반 학습
- ③ 분할 마스크를 직접 라벨로 학습하는 지도 학습
- ④ K-NN 기반 분류 학습

52. CLIP에 대한 설명으로 가장 옳은 것은?

- ① 이미지와 텍스트를 공유 임베딩 공간으로 사상하도록 대조 학습으로 학습되어 제로샷 분류에 활용 가능하다.
- ② 이미지 분류만 가능하며 텍스트는 다루지 않는다.
- ③ 픽셀 수준 분할에 특화된 모델이다.
- ④ 항상 지도 학습용 이미지 라벨이 필요하다.

53. 확산 모델(diffusion model)의 일반적인 학습 방식으로 가장 옳은 것은?

- ① 데이터에 점진적으로 가우시안 노이즈를 추가한 뒤, 그 역과정(노이즈 제거)을 학습한다.
- ② GAN과 동일하게 판별자와 경쟁한다.
- ③ 분류 손실만으로 학습된다.
- ④ 한 번의 forward pass로 무작위 이미지를 그대로 출력한다.

4과목  
자연어 처리 및 오디오

54. 텍스트 분류 파이프라인의 일반적인 흐름으로 가장 옳은 것은?

- ① 토큰화 → 임베딩/인코딩 → 분류기 헤드
- ② 분류기 헤드 → 토큰화 → 임베딩
- ③ 임베딩 → 토큰화 → 분류기 헤드
- ④ 분류기 → 분류기 → 분류기

55. BERT의 사전학습 목표(pre-training objective)로 옳은 것은?

- ① Masked Language Modeling (MLM)
- ② 좌→우 단방향 다음 단어 예측만 사용하는 자기회귀 언어 모델링
- ③ 분류 정확도 직접 최적화
- ④ 강화학습 보상 최대화

56. 다음 중 자기회귀(autoregressive) 언어 모델의 학습 방식으로 옳은 것은?

- ① 이전 토큰들이 주어졌을 때 다음 토큰의 확률 분포를 예측하도록 학습한다.
- ② 미래 토큰을 보고 과거 토큰을 예측하도록 학습한다.
- ③ 토큰 순서와 무관하게 분류만 수행한다.
- ④ 손실 함수로 MSE를 사용한다.

57. 기계번역에서 인코더-디코더 구조의 역할로 옳은 것은?

- ① 인코더는 원문(source)을 의미 표현으로 변환하고, 디코더는 그 표현으로부터 목표 언어 문장을 생성한다.
- ② 인코더와 디코더는 모두 분류만 수행한다.
- ③ 디코더는 항상 입력 문장 자체를 그대로 출력한다.
- ④ 인코더-디코더 구조는 어텐션을 사용할 수 없다.

58. 대규모 사전학습 언어 모델(LLM)을 응용 과제에 적용하는 방법으로 적절하지 않은 것은?

- ① 제로샷 또는 퓨샷 프롬프팅
- ② 지시 튜닝(instruction tuning)
- ③ LoRA 등 PEFT를 이용한 도메인 특화 미세 조정
- ④ 모든 응용에서 항상 처음부터 새로 학습(from scratch)하는 것이 가장 효율적이다.

59. HuBERT와 같은 자기지도 오디오 인코더의 학습 방식으로 가장 옳은 것은?

- ① 오디오의 일부 프레임을 마스킹하고, 사전 클러스터링으로 얻은 의사 레이블을 예측하도록 학습한다.
- ② 오디오에 대한 사람의 텍스트 라벨이 모두 필요하다.
- ③ 출력은 항상 단일 분류 라벨이다.
- ④ 오디오 신호를 직접 화소(pixel) 단위로 예측한다.

60. OpenAI의 Whisper에 대한 설명으로 가장 옳은 것은?

- ① 다국어 오디오를 입력으로 받아 전사(transcription) 및 번역을 수행하는 인코더-디코더 트랜스포머 모델이다.
- ② 텍스트만 입력으로 받는 모델이다.
- ③ 화자 분리(speaker diarization)만을 위한 모델이다.
- ④ 합성 음성 생성(TTS) 전용 모델이다.

## 서술형 (5문항)

1. 머신러닝 모델을 학습시킬 때 자주 마주치는 두 가지 문제는 과대적합(overfitting)과 과소적합(underfitting)이다. 다음 물음에 답하시오.

(1) 과대적합과 과소적합이 각각 무엇인지 1~2 문장으로 설명하시오. 그리고 학습 데이터와 검증 데이터의 정확도(또는 손실)를 보았을 때, 어떻게 둘을 구분할 수 있는지 적으시오.

(2) 어떤 모델이 학습 데이터에서는 정확도 99%, 검증 데이터에서는 정확도 70%를 보였다. 이 모델은 과대적합과 과소적합 중 어느 쪽에 해당하는가? 그 이유를 함께 설명하시오.

(3) (2)의 모델을 개선할 수 있는 방법을 두 가지 제시하고, 각 방법이 왜 도움이 되는지 한두 문장으로 설명하시오.

2. NumPy만 사용하여 K-최근접 이웃(K-NN) 분류기의 예측 함수를 작성하시오. 학습 단계는 따로 없고, 예측 시점에 학습 데이터에서 가장 가까운 k 개의 이웃을 찾아 그들의 레이블을 다수결로 모아 클래스를 결정한다.

거리 척도는 유클리드 거리(Euclidean distance)를 사용한다.

[입출력 명세]

1) 함수 시그니처는 다음과 같다.

```
def knn_predict(X_train, y_train, X_test, k=5):
    # X_train: shape (n, d) 의 np.ndarray (학습 데이터)
    # y_train: shape (n,) 의 정수 np.ndarray (정수 클래스 레이블, 0 이상)
    # X_test: shape (m, d) 의 np.ndarray (예측 대상)
    # k: int, 이웃의 수
    # 반환: shape (m,) 의 정수 np.ndarray (각 테스트 샘플의 예측 레이블)
    ...
```

2) 거리: 유클리드 거리

3) 다수결: k 개 이웃의 레이블 중 가장 자주 등장한 레이블을 선택. (동률 시 작은 레이블 우선)

4) for 루프는 사용해도 무방하나, 거리 계산은 가능한 한 numpy의 broadcasting을 활용하여 벡터화하는 것을 권장한다.

[사용 예시]

```
import numpy as np

X_train = np.array([[0, 0], [1, 0], [0, 1],
                   [10, 10], [10, 11], [11, 10]],
                  dtype=float)
y_train = np.array([0, 0, 0, 1, 1, 1])
X_test = np.array([[0.5, 0.5], [10.5, 10.5]],
                  dtype=float)

print(knn_predict(X_train, y_train, X_test, k=3))
# 기대 출력: [0 1]
```

3. 트랜스포머(Transformer)는 "어텐션 (attention)" 메커니즘을 핵심으로 하는 신경망 구조이다. 다음 물음에 답하시오.

(1) 셀프 어텐션(self-attention)이 "어떤 일"을 하는지, 자신의 말로 설명하시오. (예: 한 토큰이 같은 시퀀스 내의 어떤 토큰들로부터 정보를 가져오는지 등)

(2) 스케일드 닷-프로덕트 어텐션 식  $\text{softmax}(Q \cdot K^T / \sqrt{d_k}) \cdot V$  에서  $\sqrt{d_k}$  로 나누는 이유는 무엇인가? (힌트:  $d_k$  가 크면 내적값의 크기가 어떻게 되고, 그것이 softmax에 어떤 영향을 주는지 생각해 보라.)

(3) 트랜스포머는 단일 어텐션 대신 "멀티 헤드 어텐션(multi-head attention)"을 사용한다. 그 이유를 한두 문장으로 설명하시오.

(4) 셀프 어텐션은 토큰의 "순서"를 스스로 구분하지 못한다. 트랜스포머는 이 문제를 어떻게 해결하는가? 대표적인 한 가지 방법을 적으시오.

4. 확산 모델(Diffusion Model)은 이미지에 노이즈를 조금씩 더했다가 다시 제거하는 과정을 학습하는 생성 모델로, 최근 이미지 생성 분야의 대표적인 접근 중 하나이다. 다음 물음에 답하시오.

(1) 정방향(forward) 과정에서는 어떤 일이 일어나는지 설명하시오. 이 과정은 학습이 필요한 과정인가, 아니면 미리 정해진 규칙으로 일어나는 과정인가?

(2) 역방향(reverse) 과정에서는 어떤 일이 일어나는지 설명하시오. 학습이 끝난 모델로 새 이미지를 "생성"하는 단계가 (1)과 (2) 중 어느 쪽에 해당하는가?

(3) 학습 시 신경망은 무엇을 입력으로 받고, 무엇을 예측하도록 학습되는가? (수식 없이 말로 설명해도 좋다.)

(4) 학습된 모델로 새 이미지를 만들 때, 한 번의 신경망 호출로 끝나지 않고 "여러 단계"의 디노이징을 반복해야 하는 이유를 설명하시오.

5. 사전학습 언어 모델은 크게 BERT 계열과 GPT 계열로 나눌 수 있다. 다음 물음에 답하십시오.

(1) BERT와 GPT를 다음 두 가지 측면에서 비교하십시오.

- ① 사용하는 트랜스포머 부분(인코더/디코더)
- ② 사전학습 목표(pre-training objective)

(2) "영화 리뷰가 긍정인지 부정인지 분류" 하는 같은 과제를 다음 두 가지 방식으로 풀 수 있다. 두 방식 중 어떤 상황에서 각각이 유리한지 한 가지씩만 적으시오.

- ① BERT-base 모델을 학습 데이터로 미세조정 (fine-tuning) 하기
- ② 대규모 LLM(예: GPT 계열)에 few-shot 프롬프트를 주고 풀게 하기

(3) 매우 큰 사전학습 모델의 모든 파라미터를 미세조정하기 어려운 환경에서는 LoRA 같은 "효율적 미세조정(PEFT, parameter-efficient fine-tuning)" 기법을 사용한다. PEFT가 무엇인지(어떤 아이디어로 작동하는지), 그리고 왜 메모리와 계산이 절감되는지 간단히 설명하십시오.