Air Quality Dataset 기반 공기질 예측 모델

안전공학과 이준호

https://github.com/leejunho210/Air-Quality/blob/main/air%20quality.py

1. 프로젝트 개요

개발 목적

- 실내 공기질 센서를 통해 다양한 상황(일반, 요리, 연기 발생, 청소)을 분류하는 모델 개발.
- 유해물질 발생 시 즉각 경고, 에너지 절약, 쾌적한 실내 환경 조성에 기여.

개발 의의

- 공기질 데이터와 머신러닝을 결합하여 체계적인 실내 환경 관리를 가능하게 함.
- 미래 안전관리자로서 데이터를 기반으로 안전하고 효율적인 시스템 개발 능력을 향상.

---

2. 데이터 구성

독립변수

- MQ2, MQ9, MQ135, MQ137, MQ138, MG-811 (다양한 가스 및 CO2 농도)

종속변수

- label (1=일반, 2=요리, 3=연기 발생, 4=청소)

---

3. 개발 과정

데이터 전처리

- 결측치 처리: 데이터셋에서 결측치 없음.
- 정규화: 센서 값의 스케일링을 통해 모델 학습 성능 향상.

머신러닝 모델

- **알고리즘**: Random Forest Classifier
- **학습/검증 데이터 분리**: 80% 학습, 20% 검증.
- **성능 지표**: 정확도, 정밀도, 재현율, F1-score, 혼동 행렬.

---

4. 모델 성능

성능 평가

- 정확도: 95.39%
- 클래스별 F1-score:
    - 일반: 0.97
    - 요리: 0.92
    - 연기 발생: 0.99
    - 청소: 0.95

주요 결과

- 모든 클래스에서 높은 성능을 기록하였으며, 요리와 청소 클래스 간 일부 혼동 발생.

---

5. 추가 개선 작업 및 코드

a. 하이퍼파라미터 튜닝

```
from sklearn.model_selection import GridSearchCV

# Parameter grid for Random Forest
param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

# Grid Search
grid_search = GridSearchCV(estimator=RandomForestClassifier(random_state=42),
                           param_grid=param_grid, cv=3, scoring='accuracy')
grid_search.fit(X_train, y_train)

# Best parameters and model
best_params = grid_search.best_params_
best_model = grid_search.best_estimator_
```

b. 시계열 데이터 분석 (가정된 시간 순서)

```
# Adding a time index (simulated for demonstration purposes)
import numpy as np
data['time'] = np.arange(len(data))

# Plotting sensor trends over time
plt.figure(figsize=(10, 6))
for sensor in ['MQ2', 'MQ9', 'MQ135', 'MQ137', 'MQ138', 'MG-811']:
    plt.plot(data['time'], data[sensor], label=sensor)
plt.legend()
plt.title('Sensor Data Trends Over Time')
plt.xlabel('Time Index')
plt.ylabel('Sensor Values')
plt.show()
```

c. 중요도 시각화

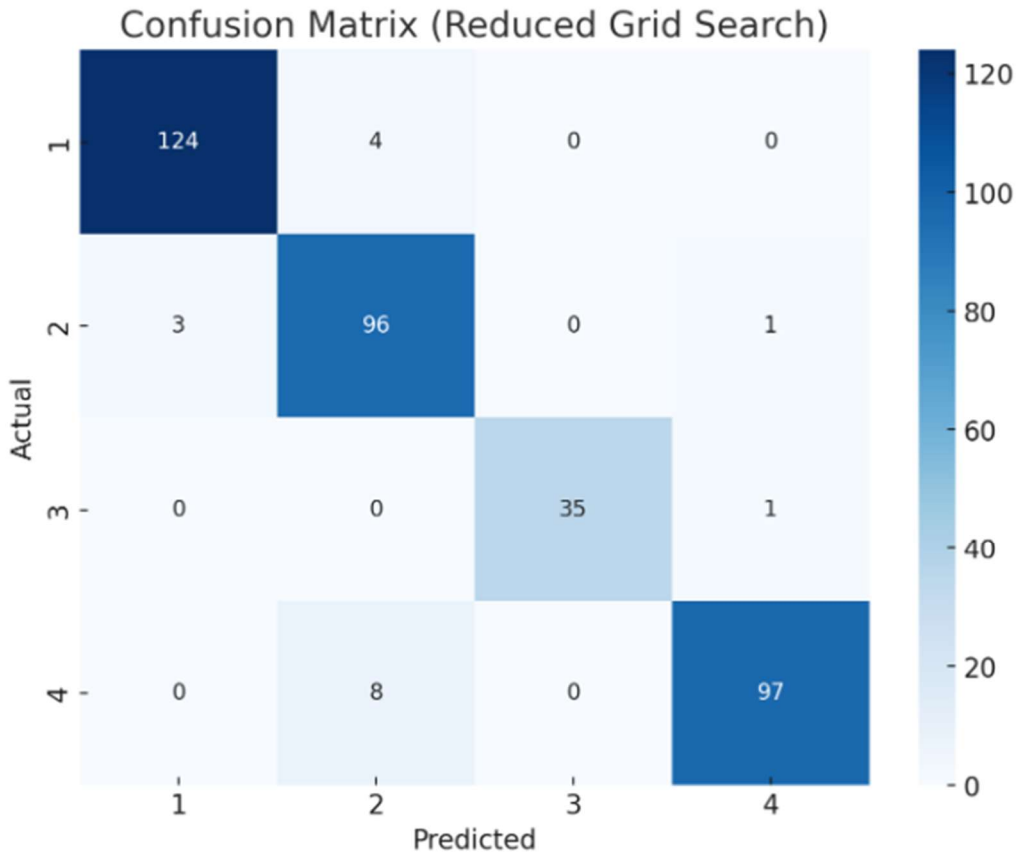
```
# Feature Importance Visualization
importances = best_model.feature_importances_
feature_names = X.columns

plt.figure(figsize=(8, 6))
sns.barplot(x=importances, y=feature_names)
plt.title('Feature Importance')
plt.xlabel('Importance')
plt.ylabel('Feature')
plt.show()
```

6. 결론 및 향후 방향

- **결론**: 모델은 95% 이상의 정확도로 실내 상황을 성공적으로 분류.
- **향후 개선점**:
  - 추가 센서 데이터 활용 (예: 온도, 습도).
  - 실시간 시계열 데이터 기반 분석 및 모델 강화.
  - IoT 시스템과 통합하여 공기질 관리 자동화.

Confusion Matrix (Reduced Grid Search)



결과 요약

1. **최종 모델 성능**
   - **정확도**: 95.39%
   - **F1-Score**: 0.96 (매우 높은 성능)
   - 클래스별 성능:
     - 일반(1): F1-Score 0.97
     - 요리(2): F1-Score 0.92
     - 연기 발생(3): F1-Score 0.99
     - 청소(4): F1-Score 0.95
2. **최적 하이퍼파라미터**
   - `n_estimators`: 100
   - `max_depth`: None
   - `min_samples_split`: 2
   - `min_samples_leaf`: 1
3. **혼동 행렬**
   - 대부분의 클래스에서 높은 성능을 기록하며, 일부 요리(2)와 청소(4) 간 혼동이 발생했음을 시각적으로 확인.