v.1

Apriori 알고리즘을 사용하여 confidence가 0.7 이상인 순서쌍을 찾기 위해 Python 코드를 작성해 보겠습니다. 주어진 코드를 기반으로 데이터를 MySQL에서 불러와서 처리하고, Apriori 알고리즘을 적용하여 원하는 결과를 추출할 것입니다.

1. 데이터베이스 연결 및 데이터 불러오기

기존 코드에서 데이터를 MySQL 데이터베이스에 저장한 후, 필요한 데이터를 불러오는 방식을 사용하겠습니다. 데이터베이스 연결 및 데이터 불러오기 코드는 다음과 같습니다:

2. Apriori 알고리즘 구현

이제 Apriori 알고리즘을 직접 구현하여 confidence가 0.7 이상인 순서쌍을 찾겠습니다. 여기서는 itertools를 사용하여 가능한 모든 조합을 생성하고, 이를 통해 support와 confidence를 계산할 것입니다.

설명

데이터는 MySQL에서 불러와서 각 transaction을 생성하고, reconnaissance, infection, action을 제외한 나머지 요소들을 'on' 또는 'off'로 변환하여 transactions 리스트에 저장합니다.

Apriori 알고리즘은 후보 생성과 지지도 계산을 반복하여 frequent itemset을 생성합니다.

각 frequent itemset에 대해 가능한 규칙을 생성하고, 이 규칙의 confidence를 계산하여 주어진 min\_confidence 이상인 규칙을 출력합니다.

이 코드는 주어진 데이터에서 Apriori 알고리즘을 구현하고, confidence가 0.7 이상인 연관 규칙을 찾아 출력하는 기능을 수행합니다.

v.2

코드 설명

MySQL 데이터 삽입 부분: output-dataset\_ESSlab.csv 파일을 읽어 MySQL 데이터베이스에 삽입하는 부분입니다. INSERT INTO 쿼리를 사용하여 각 행을 데이터베이스에 삽입합니다.

데이터 전처리 및 Apriori 알고리즘 적용: MySQL에서 데이터를 가져와서 Apriori 알고리즘을 적용하는 부분입니다. 데이터를 transactions 리스트로 변환하고, 이를 기반으로 지지도 계산 및 Apriori 알고리즘을 적용하여 연관 규칙을 추출합니다.

결과 출력: 추출된 연관 규칙을 출력합니다.

위 코드를 실행하기 전에 MySQL 데이터베이스가 실행 중이고, 정상적으로 접속할 수 있는지 확인해야 합니다. 또한 코드에서 사용하는 계정 정보나 포트 번호가 MySQL 설정과 일치하는지도 확인해야 합니다.

Mysql

확장자가 .sql인 파일은 일반적으로 SQL 쿼리와 명령어를 포함하고 있으며, MySQL 데이터베이스에 대해 실행해야 하는 경우가 많습니다. SQL 파일을 데이터베이스에 적용하려면 다음과 같은 방법을 사용하여 파일의 내용을 실행할 수 있습니다.

방법 1: MySQL 클라이언트에서 실행

MySQL 클라이언트를 사용하여 SQL 파일을 실행할 수 있습니다. 다음은 MySQL 클라이언트에서 .sql 파일을 실행하는 방법입니다:

명령 프롬프트 (Windows) 또는 터미널 (Linux/Mac) 열기

MySQL 클라이언트 접속

MySQL 클라이언트에 접속하려면 다음 명령어를 입력합니다:

bash

코드 복사

mysql -u your\_username -p

여기서 your\_username은 MySQL의 사용자 이름입니다. 명령어를 입력한 후, 비밀번호를 입력하라는 메시지가 나타납니다.

데이터베이스 선택

SQL 파일의 쿼리가 적용될 데이터베이스를 선택합니다. 예를 들어, sample 데이터베이스를 선택하려면 다음 명령어를 입력합니다:

sql

코드 복사

USE sample;

SQL 파일 실행

다음 명령어를 입력하여 .sql 파일을 실행합니다:

sql

코드 복사

SOURCE /path/to/your\_file.sql;

여기서 /path/to/your\_file.sql은 SQL 파일의 경로입니다.

방법 2: MySQL Workbench에서 실행

MySQL Workbench와 같은 GUI 도구를 사용하는 경우, SQL 파일을 열고 실행할 수 있습니다:

MySQL Workbench 열기

SQL 파일 열기

MySQL Workbench에서 "File" 메뉴를 선택하고 "Open SQL Script"를 클릭하여 .sql 파일을 엽니다.

SQL 스크립트 실행

SQL 파일이 열린 후, 상단의 "Execute" 버튼을 클릭하여 스크립트를 실행합니다.

방법 3: 커맨드라인에서 직접 실행

커맨드라인에서 직접 SQL 파일을 실행할 수도 있습니다:

명령 프롬프트 (Windows) 또는 터미널 (Linux/Mac) 열기

다음 명령어를 입력하여 SQL 파일을 실행합니다:

bash

코드 복사

mysql -u your\_username -p your\_database < /path/to/your\_file.sql

여기서 your\_username은 MySQL의 사용자 이름, your\_database는 사용할 데이터베이스의 이름, /path/to/your\_file.sql은 SQL 파일의 경로입니다.

예시

SQL 파일을 C:\path\to\your\_file.sql에 저장했으며, MySQL 서버의 데이터베이스 sample에 적용하려면, 다음 명령어를 사용할 수 있습니다:

bash

코드 복사

mysql -u root -p sample < C:\path\to\your\_file.sql

이 명령어를 입력한 후 비밀번호를 입력하면 SQL 파일의 내용이 sample 데이터베이스에 적용됩니다.

요약

.sql 파일은 SQL 명령어를 포함하고 있으며, 데이터베이스에 적용하기 위해 실행해야 합니다.

MySQL 클라이언트, MySQL Workbench, 또는 커맨드라인에서 직접 실행할 수 있습니다.

데이터베이스에 적용하기 전에 SQL 파일의 내용을 확인하여 올바른 명령어가 포함되어 있는지 확인하는 것이 좋습니다.

V3

자료형 튜플로 변경

주요 수정 사항:

preprocess\_data 함수:

transactions를 튜플로 변환합니다. 리스트 대신 튜플을 사용하여 각 트랜잭션을 불변 데이터 구조로 만듭니다.

generate\_candidates 함수:

후보 생성 결과를 tuple로 반환합니다. set에서 후보를 생성하고 최종적으로 tuple로 변환하여 반환합니다.

apriori 함수:

itemset 및 frequent\_itemsets를 set으로 사용하여 중복을 자동으로 처리합니다. frequent\_itemsets를 업데이트한 후, tuple로 변환된 규칙을 반환합니다.

이렇게 수정된 코드는 튜플을 사용하여 데이터 구조의 불변성을 보장합니다.

v.5\_2 all

**변경사항 설명:**

1. **중간 그룹 추가**: current\_group이 업데이트될 때마다 그 상태를 related\_groups에 추가했습니다. 이로 인해 각 단계의 그룹을 모두 기록하게 됩니다.
2. **그룹 복사**: current\_group.copy()를 사용하여 현재 그룹의 상태를 복사하고 추가했습니다. 이를 통해 그룹의 상태가 계속 변경되더라도 이전 상태가 유지됩니다.

이제 코드가 중간 단계의 그룹도 포함하게 되어, 관련성 분석이 진행되는 모든 단계의 그룹을 추적할 수 있습니다.

Result:

[['action'], ['action', 'flow\_iat\_max'], ['action', 'flow\_iat\_mean', 'flow\_iat\_max'], ['action', 'flow\_iat\_mean', 'flow\_iat\_min', 'flow\_iat\_max'], ['action', 'flow\_iat\_max', 'flow\_iat\_min', 'flow\_iat\_mean', 'flow\_iat\_std'], ['flow\_iat\_total', 'action', 'flow\_iat\_max', 'flow\_iat\_min', 'flow\_iat\_mean', 'flow\_iat\_std'], ['flow\_iat\_total', 'action', 'flow\_iat\_max', 'flow\_iat\_min', 'flow\_iat\_mean', 'forward\_iat\_max', 'flow\_iat\_std'], ['flow\_iat\_max', 'flow\_iat\_mean', 'forward\_iat\_max', 'flow\_iat\_std', 'flow\_iat\_total', 'action', 'forward\_iat\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'flow\_iat\_mean', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_std', 'flow\_iat\_total', 'action', 'forward\_iat\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'forward\_iat\_std', 'flow\_iat\_mean', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_std', 'flow\_iat\_total', 'action', 'forward\_iat\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'forward\_iat\_std', 'forward\_iat\_total', 'flow\_iat\_mean', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_std', 'flow\_iat\_total', 'action', 'forward\_iat\_mean', 'flow\_iat\_min'], ['backward\_iat\_max'], ['backward\_iat\_mean'], ['backward\_iat\_min'], ['backward\_iat\_std'], ['backward\_iat\_total'], ['backward\_packet\_length\_max'], ['backward\_packet\_length\_mean'], ['backward\_packet\_length\_min'], ['backward\_packet\_length\_std'], ['bpkts\_per\_second'], ['flow\_ack'], ['flow\_ack', 'infection'], ['flow\_cwr'], ['flow\_ece'], ['flow\_fin'], ['flow\_packets\_per\_second'], ['flow\_packets\_per\_second', 'flow\_iat\_max'], ['flow\_packets\_per\_second', 'flow\_iat\_min', 'flow\_iat\_max'], ['flow\_iat\_total', 'flow\_packets\_per\_second', 'flow\_iat\_min', 'flow\_iat\_max'], ['flow\_iat\_total', 'flow\_packets\_per\_second', 'flow\_iat\_max', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_iat\_total', 'flow\_packets\_per\_second', 'flow\_iat\_max', 'flow\_iat\_min', 'flow\_protocol', 'forward\_iat\_max'], ['flow\_iat\_total', 'flow\_packets\_per\_second', 'flow\_iat\_max', 'flow\_iat\_min', 'flow\_protocol', 'forward\_iat\_max', 'forward\_iat\_min'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_total', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_total', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_total', 'forward\_packet\_length\_mean', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_total', 'forward\_packet\_length\_mean', 'forward\_packet\_length\_min', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'fpkts\_per\_second', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_total', 'forward\_packet\_length\_mean', 'forward\_packet\_length\_min', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'total\_bhlen', 'fpkts\_per\_second', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'flow\_iat\_total', 'forward\_packet\_length\_mean', 'forward\_packet\_length\_min', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['flow\_packets\_per\_second', 'flow\_iat\_max', 'total\_bhlen', 'fpkts\_per\_second', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'total\_forward\_packets', 'flow\_iat\_total', 'forward\_packet\_length\_mean', 'forward\_packet\_length\_min', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['total\_length\_of\_forward\_packets', 'flow\_packets\_per\_second', 'flow\_iat\_max', 'total\_bhlen', 'fpkts\_per\_second', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_iat\_min', 'total\_forward\_packets', 'flow\_iat\_total', 'forward\_packet\_length\_mean', 'forward\_packet\_length\_min', 'flow\_iat\_min', 'flow\_protocol', 'forward\_packet\_length\_max'], ['flow\_psh'], ['forward\_packet\_length\_std', 'flow\_psh'], ['flow\_rst'], ['flow\_syn'], ['reconnaissance', 'flow\_syn'], ['flow\_urg'], ['total\_backward\_packets'], ['total\_fhlen'], ['total\_length\_of\_backward\_packets']]

Element count 1: 23 groups

Element count 2: 5 groups

Element count 3: 2 groups

Element count 4: 2 groups

Element count 5: 2 groups

Element count 6: 2 groups

Element count 7: 2 groups

Element count 8: 2 groups

Element count 9: 2 groups

Element count 10: 2 groups

Element count 11: 2 groups

Element count 12: 1 groups

Element count 13: 1 groups

Element count 14: 1 groups

Element count 15: 1 groups

Nomal\_v.1

Result:

[['action'], ['backward\_iat\_max'], ['backward\_iat\_mean', 'backward\_iat\_max'], ['backward\_iat\_mean', 'backward\_iat\_max', 'backward\_iat\_min'], ['backward\_iat\_mean', 'backward\_iat\_max', 'backward\_iat\_std', 'backward\_iat\_min'], ['backward\_iat\_min', 'backward\_iat\_max', 'backward\_iat\_std', 'backward\_iat\_total', 'backward\_iat\_mean'], ['backward\_iat\_min', 'backward\_iat\_max', 'backward\_packet\_length\_max', 'backward\_iat\_std', 'backward\_iat\_total', 'backward\_iat\_mean'], ['backward\_iat\_min', 'backward\_iat\_max', 'backward\_packet\_length\_max', 'backward\_iat\_std', 'backward\_iat\_total', 'backward\_iat\_mean', 'backward\_packet\_length\_mean'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean'], ['backward\_iat\_max', 'backward\_iat\_std', 'flow\_ack', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean'], ['flow\_iat\_max', 'backward\_iat\_max', 'backward\_iat\_std', 'flow\_ack', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean'], ['flow\_iat\_max', 'backward\_iat\_max', 'backward\_iat\_std', 'flow\_ack', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'flow\_iat\_mean', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean'], ['flow\_iat\_max', 'backward\_iat\_max', 'backward\_iat\_std', 'flow\_ack', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'flow\_iat\_mean', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'backward\_iat\_max', 'flow\_iat\_std', 'backward\_iat\_std', 'flow\_ack', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'flow\_iat\_mean', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'backward\_iat\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'backward\_iat\_max', 'flow\_iat\_std', 'backward\_iat\_std', 'flow\_ack', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'bpkts\_per\_second', 'flow\_iat\_mean', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_iat\_mean', 'flow\_iat\_min'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'flow\_iat\_max', 'flow\_psh', 'flow\_ack', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_iat\_std', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['forward\_iat\_total', 'backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_iat\_std', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['forward\_iat\_total', 'backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'forward\_iat\_std', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['forward\_iat\_total', 'backward\_iat\_max', 'backward\_iat\_std', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'forward\_iat\_std', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['forward\_iat\_total', 'backward\_iat\_max', 'backward\_iat\_std', 'total\_fhlen', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'forward\_iat\_std', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['forward\_iat\_total', 'total\_length\_of\_backward\_packets', 'backward\_iat\_max', 'backward\_iat\_std', 'total\_fhlen', 'backward\_packet\_length\_min', 'forward\_iat\_mean', 'backward\_iat\_min', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_iat\_total', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'forward\_iat\_std', 'bpkts\_per\_second', 'flow\_iat\_mean', 'flow\_iat\_std', 'backward\_iat\_mean', 'flow\_iat\_min'], ['backward\_packet\_length\_std'], ['flow\_cwr'], ['flow\_ece'], ['flow\_fin'], ['flow\_packets\_per\_second'], ['flow\_packets\_per\_second', 'backward\_packet\_length\_max'], ['flow\_packets\_per\_second', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean'], ['flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean'], ['bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min'], ['bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min'], ['flow\_iat\_max', 'bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min'], ['flow\_iat\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_iat\_min'], ['flow\_iat\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_iat\_max', 'flow\_psh', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_iat\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_iat\_max', 'flow\_psh', 'forward\_iat\_total', 'forward\_iat\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_min', 'flow\_protocol'], ['flow\_iat\_max', 'flow\_psh', 'forward\_iat\_total', 'forward\_iat\_max', 'forward\_packet\_length\_max', 'flow\_ack', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'bpkts\_per\_second', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_iat\_min', 'flow\_protocol'], ['forward\_iat\_total', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'flow\_psh', 'forward\_packet\_length\_max', 'forward\_iat\_max', 'flow\_ack', 'bpkts\_per\_second', 'flow\_iat\_min'], ['forward\_iat\_total', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'flow\_psh', 'forward\_packet\_length\_max', 'forward\_iat\_max', 'flow\_ack', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['forward\_iat\_total', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'flow\_psh', 'forward\_packet\_length\_max', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['forward\_iat\_total', 'fpkts\_per\_second', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['forward\_iat\_total', 'fpkts\_per\_second', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['total\_bhlen', 'forward\_iat\_total', 'fpkts\_per\_second', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['total\_bhlen', 'forward\_iat\_total', 'fpkts\_per\_second', 'total\_fhlen', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['total\_bhlen', 'forward\_iat\_total', 'fpkts\_per\_second', 'total\_fhlen', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'total\_forward\_packets', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['total\_bhlen', 'forward\_iat\_total', 'total\_length\_of\_backward\_packets', 'fpkts\_per\_second', 'total\_fhlen', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'total\_forward\_packets', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['total\_bhlen', 'forward\_iat\_total', 'total\_length\_of\_backward\_packets', 'fpkts\_per\_second', 'total\_fhlen', 'flow\_packets\_per\_second', 'backward\_packet\_length\_min', 'total\_forward\_packets', 'forward\_packet\_length\_mean', 'flow\_iat\_total', 'backward\_packet\_length\_max', 'backward\_packet\_length\_mean', 'forward\_iat\_min', 'flow\_protocol', 'total\_length\_of\_forward\_packets', 'flow\_iat\_max', 'forward\_packet\_length\_max', 'flow\_psh', 'forward\_iat\_max', 'flow\_ack', 'forward\_packet\_length\_std', 'total\_backward\_packets', 'bpkts\_per\_second', 'forward\_packet\_length\_min', 'flow\_iat\_min'], ['flow\_rst'], ['flow\_syn'], ['flow\_urg'], ['infection'], ['reconnaissance']]

Element count 1: 12 groups

Element count 2: 2 groups

Element count 3: 2 groups

Element count 4: 2 groups

Element count 5: 2 groups

Element count 6: 2 groups

Element count 7: 2 groups

Element count 8: 2 groups

Element count 9: 2 groups

Element count 10: 2 groups

Element count 11: 2 groups

Element count 12: 2 groups

Element count 13: 2 groups

Element count 14: 2 groups

Element count 15: 2 groups

Element count 16: 2 groups

Element count 17: 2 groups

Element count 18: 2 groups

Element count 19: 2 groups

Element count 20: 2 groups

Element count 21: 2 groups

Element count 22: 2 groups

Element count 23: 2 groups

Element count 24: 2 groups

Element count 25: 2 groups

차집합 코드

**코드 설명:**

* **집합 사용**: group1\_set과 group2\_set을 생성하여 각 그룹의 원소를 집합으로 변환합니다. 집합은 원소의 순서를 고려하지 않으므로, 두 CSV 파일의 원소 순서가 다르더라도 문제없이 차집합을 계산할 수 있습니다.
* **차집합 계산**: difference = group1\_set - group2\_set를 통해 차집합을 계산합니다. 이 경우, group1의 원소 중 group2에 없는 원소들만 포함됩니다.

결과값 판정 -> list 제외

# 'anomal'이 1인 열들 선택, 'reconnaissance', 'infection', 'action' 열 제외 anormal\_columns = anormal\_rows.columns.difference(['anomal', 'reconnaissance', 'infection', 'action'])

이것들을 합쳐서 평가 점수에 따라 persent(confidence)를 변경하는 코드를 짜고 싶어. 총점 평가 비율은 Accuracy 100%, Precision 100%, Recall 100%, F1 score 100% 기준으로 점수 총점 100% 중 Accuracy 40%, Precision 20%, Recall 30% F1 score 10% 차지 비율로 점수를 평가할거야.

그래서 그 평가 점수가 높은 방향으로 confidence를 변경할 건데, 10% 단위로 나눠서 점수가 가장 높은 쪽에 가까운 쪽으로 confidence를 변경하고, 뒤에는 다시 앞서 한 10% 내에서 또 1% 단위로 나눠서 평가, 그 다음은 0.1% 단위로 나눠서 평가.... 이걸 반복해서 0.000001%까지 작성해줘.

요약

단계 1: 데이터셋을 로드하고 anomal 열을 생성하여 related\_groups를 계산합니다.

단계 2: 각 related\_groups에 대한 메트릭을 평가하고, 이를 기반으로 confusion matrix를 계산합니다.

단계 3: 메트릭을 기반으로 최종 점수를 계산하고, 이 점수를 기반으로 confidence를 조정합니다.

네, 맞아요! 해당 코드에서 confidence (결합 비율) 기준에 따라 그룹을 확장하는 방식에서는, confidence가 낮은 그룹이 confidence가 높은 그룹에 포함될 수 있는 구조가 됩니다.

문제의 원인:

코드에서 related\_groups는 반복적으로 그룹을 확장하는 방식으로 생성됩니다. 만약 한 그룹에서 특정 열과의 관련성(공유 비율)을 기준으로 다른 열을 추가할 때, 처음 만들어진 작은 그룹이 나중에 큰 그룹으로 합쳐질 수 있습니다.

예시를 통해 설명:

첫 번째 그룹이 ['A']라고 가정하고, 두 번째 그룹은 ['A', 'B']로 확장됩니다.

이때 A와 B 사이의 결합 비율이 70% 이상이라면 A와 B는 관련성이 있다고 판단됩니다.

만약 세 번째 그룹이 ['A', 'C']라면, A와 C 사이의 결합 비율이 70% 이상이라면 C는 그룹에 포함될 수 있습니다.

그런데 두 번째 그룹에서 A와 B가 이미 확장된 그룹이라면, 세 번째 그룹인 ['A', 'C'] 또한 ['A', 'B', 'C']로 확장될 수 있다는 문제가 발생할 수 있습니다.

왜 그런가?

새로운 그룹을 발견하는 과정 (while new\_group\_found:)에서 각 열은 그 기준을 만족할 때마다 계속해서 그룹에 추가되며, 기존 그룹에서 겹치는 부분이 있는 다른 열들이 반복적으로 추가되기 때문입니다.

따라서 confidence가 낮은 그룹이 confidence가 높은 그룹에 포함되는 것은 자연스러운 현상입니다.

해결 방법:

related\_groups에서 각 그룹을 고유한 항목만 포함하도록 하고, 중복되지 않도록 하는 방법은 다음과 같습니다:

그룹 내에 이미 존재하는 항목을 추가하지 않기: 새로 추가되는 열이 이미 그룹에 포함되어 있지 않다면 추가해야 합니다.

그룹의 확장 조건을 다시 확인: 그룹을 확장할 때, related\_groups에서 추가되는 그룹이 기존 그룹에 포함되어 있지 않으면 추가하지 않도록 조건을 강화합니다.

수정된 코드 예시:

그룹을 확장할 때, 새로 추가된 그룹이 이미 이전 그룹에 포함되어 있지 않도록 추가 로직을 넣을 수 있습니다.

python

코드 복사

import pandas as pd

data = pd.read\_csv('./output-dataset\_ESSlab.csv')

# 'reconnaissance', 'infection', 'action' 열 중 하나라도 1인 경우

data['anomal'] = data[['reconnaissance', 'infection', 'action']].any(axis=1).astype(int)

# 'reconnaissance', 'infection', 'action' 열을 제외한 열 선택

cols\_to\_change = data.columns.difference(['reconnaissance', 'infection', 'action'])

# 0보다 큰 값을 1로 변경

data[cols\_to\_change] = data[cols\_to\_change].applymap(lambda x: 1 if x > 0 else 0)

# 'anomal' 값이 1인 행만 필터링

anormal\_rows = data[data['anomal'] == 1]

# 'anomal'이 1인 열들 선택, 'reconnaissance', 'infection', 'action' 열 제외

anormal\_columns = anormal\_rows.columns.difference(['anomal', 'reconnaissance', 'infection', 'action'])

# 각 column에서 값이 1인 row의 인덱스 리스트 만들기

anormal\_lists = {col: anormal\_rows.index[anormal\_rows[col] == 1].tolist() for col in anormal\_columns}

# 퍼센트 기준 설정

percent = 70 / 100

# 관련성 있는 리스트를 저장할 리스트

related\_groups = []

considered = set() # 이미 고려된 열들을 저장할 집합

# 모든 열에 대해 반복

for a in list(anormal\_lists.keys()):

if a in considered:

continue

current\_group = {a} # 현재 그룹 초기화

considered.add(a) # 현재 열을 고려된 집합에 추가

related\_groups.append(current\_group.copy()) # 초기 그룹 추가

new\_group\_found = True

while new\_group\_found:

new\_group\_found = False

for b in list(anormal\_lists.keys()):

if b not in current\_group:

# A와 B의 관련성 확인

a\_count = len(anormal\_lists[a])

b\_count = len(anormal\_lists[b])

combined\_count = len(set(anormal\_lists[a]) & set(anormal\_lists[b]))

# a\_count 또는 b\_count가 0이 아닐 경우에만 조건 확인

if a\_count > 0 and b\_count > 0:

if combined\_count / a\_count >= percent and combined\_count / b\_count >= percent:

# 그룹 확장 시, 이미 고려된 그룹인지 확인

if not any(b in group for group in related\_groups):

current\_group.add(b)

considered.add(b) # 추가된 열을 고려된 집합에 추가

new\_group\_found = True

related\_groups.append(current\_group.copy()) # 중간 그룹 추가

# 결과 확인

print([list(group) for group in related\_groups])

# 그룹의 요소 개수별로 세기 위한 딕셔너리 초기화

count\_by\_size = {}

# related\_groups의 각 그룹에 대해 요소 개수 세기

for group in related\_groups:

size = len(group) # 그룹의 요소 개수

if size in count\_by\_size:

count\_by\_size[size] += 1

else:

count\_by\_size[size] = 1

# 결과 출력

sorted\_counts = sorted(count\_by\_size.items())

# 출력

for size, count in sorted\_counts:

print(f"Element count {size}: {count} groups")

# related\_groups의 각 그룹을 DataFrame으로 변환

related\_groups\_df = pd.DataFrame(related\_groups)

# 현재 Python 파일과 동일한 위치에 CSV 파일로 저장

related\_groups\_df.to\_csv('related\_groups\_anomal\_2.csv', index=False, header=False)

주요 수정 사항:

그룹 확장 시, 기존 그룹을 고려한 조건 추가:

if not any(b in group for group in related\_groups): 조건을 추가하여, 새로운 그룹을 기존 그룹에 포함시키지 않도록 했습니다.

이 조건은 새로운 열이 다른 그룹에 포함되지 않은 경우에만 해당 그룹에 추가하도록 합니다.

결과:

이 수정된 코드는 confidence가 낮은 그룹이 높은 그룹에 포함되지 않도록 보장합니다. 즉, 각 그룹은 고유한 항목들만 포함하며, 서로 중복되지 않게 됩니다.

이제 confidence가 낮은 그룹이 confidence가 높은 그룹의 요소를 포함하는 문제를 방지할 수 있습니다.