주어지는 예시 csv 데이터가 다음과 같아:

Vision,Indicator\_name,Metric\_Achievement\_H1,Metric\_Achievement\_H2,Evaluate\_Nature\_Metrics,Performance\_Metric\_Score\_H1,Performance\_Metric\_Score\_H2

2,"1,10,13,22",82.5,100,92.1,59.67,68.42

2,"1,9,13,20",100,100,89.07,67.81,67.81

2,"9,13,15,19",100,100,92,68.4,68.4

2,"1,3,13,17,29",100,100,84.43,66.89,66.89

1,"13,16,22",,68.8,87.93,,51.99

1,"5,13",,100,100,,70

1,"13,15,16,17,22,23",,100,100,,70

1,"13,17,22",,100,100,,70

1,"13,22,29",,100,100,,70

1,"2,13,22,28",,100,100,,70

2,"1,9,13,19",100,100,85.6,67.12,67.12

2,"1,9,13,19,23",80,100,80.8,56.16,66.16

2,"1,9,13,23,29",81.25,,88.67,58.36,

2,"1,13,15,17,19,23",83.5,,90.17,59.78,

2,"13,20,22,37",81,,87.17,57.93,

4,"1,2,13,26",74.61,73.17,87.93,54.89,54.17

4,"13,22,26,32",19.85,31.05,87.93,27.51,33.11

5,"7,13,23",,100,87.93,,67.59

5,"13,27",,59.21,87.93,,47.19

5,"2,13",,77.25,87.93,,56.21

먼저 이 데이터를 판다스로 받아서 Vision column에 따라 그룹을 나눌거야. 이 데이터는 NetworkX 프로그램으로 돌릴 edge csv를 만드는 데 기초가 될 거야. 나중에는 vision column에 따라 edge csv를 따로 만들 것이고. 저기서 시작 노드는 13번, 끝 노드는 Indicator\_name 내에 속하는 다른 각각의 숫자 노드(시작 노드의 번호는 저 리스트에서 제외해야 해)로 이루어진 가중치 edge를 만들 거야. 그걸 각 열마다 반복할 거야. 가중치 값은 Metric\_Achievement\_H1,Metric\_Achievement\_H2의 평균값과 Performance\_Metric\_Score\_H1,Performance\_Metric\_Score\_H2의 평균 값을 곱할 거야.

나중에 다른 source csv 파일도 계속 읽어올 것이니까 각 vision df에서 계속 내용을 전달받을 수 있도록 했으면 좋겠어.

부서 재편성을 효율적으로 하기 위해서는 네트워크 분석과 머신러닝 기법을 활용하는 접근이 매우 유용할 수 있습니다. 두 가지 네트워크(사업별 공무원 부서 협력 네트워크와 읍면동에서 부서별 협력 네트워크)에 대한 정보와 가중치를 바탕으로, 다양한 분석 방법을 통해 부서 재편성을 최적화할 수 있습니다. 여기에 활용할 수 있는 네트워크 정보와 머신러닝 기법을 아래와 같이 제안할 수 있습니다.

**1. 네트워크 정보 활용**

**a. 네트워크 중심성 분석 (Centrality Analysis)**

* **주요 지표**: Degree centrality, Betweenness centrality, Closeness centrality, Eigenvector centrality
* **설명**: 각 부서가 네트워크에서 얼마나 중요한 역할을 하는지 평가합니다. 예를 들어, 주관부서가 중심이 되는 네트워크에서 주관부서가 다른 부서들과 얼마나 긴밀하게 연결되어 있는지 분석할 수 있습니다. 이를 바탕으로 효율적인 재편성을 위한 핵심 부서들을 파악할 수 있습니다.
  + **Degree centrality**: 부서가 연결된 다른 부서의 수
  + **Betweenness centrality**: 네트워크 내에서 다른 부서들 간의 최단경로를 얼마나 자주 차지하는지
  + **Closeness centrality**: 네트워크 내에서 다른 부서들과의 평균 거리가 얼마나 가까운지
  + **Eigenvector centrality**: 연결된 부서의 중요성까지 반영한 중심성

**b. 커뮤니티 탐지 (Community Detection)**

* **주요 지표**: Modularity, Louvain method, Girvan-Newman algorithm
* **설명**: 네트워크 내에서 비슷한 특성을 가진 부서들이 어떻게 그룹화될 수 있는지 찾는 기법입니다. 커뮤니티 탐지를 통해 협력이 잘 이루어지는 부서들을 그룹화하고, 이들을 기반으로 재편성 방향을 제시할 수 있습니다.

**c. 가중치 기반 분석 (Weighted Network Analysis)**

* **주요 지표**: 가중치 합산, 최단 경로, 클러스터링 계수
* **설명**: 각 부서 간 협력의 강도(가중치)를 고려하여 분석합니다. 가중치를 고려하면, 단순히 연결만을 보는 것이 아니라, 협력의 강도가 중요한 부서나 업무를 더 잘 파악할 수 있습니다. 이를 통해 더 효율적인 협력 부서 구성을 할 수 있습니다.

**2. 머신러닝 기법 활용**

**a. 클러스터링 (Clustering)**

* **기법**: K-means, DBSCAN, Hierarchical clustering
* **설명**: 부서들을 비슷한 협력 특성을 가진 그룹으로 나누는 방법입니다. 예를 들어, 협력 강도가 높은 부서들을 하나의 그룹으로 묶고, 비슷한 업무나 목표를 가진 부서들을 한 곳에 재배치할 수 있습니다. 이때, 네트워크에서의 가중치를 특성으로 사용할 수 있습니다.

**b. 그래프 신경망 (Graph Neural Networks, GNN)**

* **기법**: GCN (Graph Convolutional Network), GraphSAGE
* **설명**: 네트워크 구조를 직접 모델링하는 머신러닝 기법입니다. GNN은 부서들 간의 관계를 그래프 형태로 모델링하고, 이를 학습하여 최적의 부서 재편성 방향을 제시할 수 있습니다. 이 방법은 부서 간의 협력 관계와 가중치를 반영하여, 협력 강도가 높은 부서들을 그룹화하거나, 역할 분담을 더 효율적으로 할 수 있습니다.

**c. 최적화 문제 (Optimization Problems)**

* **기법**: Integer Programming, Linear Programming
* **설명**: 부서 재편성을 수학적 최적화 문제로 모델링하여, 효율적인 부서 재배치를 도출할 수 있습니다. 예를 들어, 각 부서의 협력 강도와 중요도를 고려한 가중치를 목표로 최소화 또는 최대화하는 방식으로 재편성 방안을 제시할 수 있습니다.

**3. 네트워크 분석 기반 시나리오 제시**

**a. 효율적 부서 간 협력 강화**

* **목표**: 협력 강도가 높은 부서들 간의 업무를 통합하거나 상호 협력 구조를 강화하는 방식으로 재편성합니다. 커뮤니티 탐지나 중심성 분석을 통해 주요 부서들을 확인하고, 이들 간의 협력 강도를 최대화하는 방식으로 재편성을 고려합니다.

**b. 업무 분담 최적화**

* **목표**: 비슷한 유형의 업무를 처리하는 부서들을 그룹화하여 업무 효율성을 높입니다. 이 경우 클러스터링 기법을 활용하여 부서 간 협력 강도가 유사한 그룹을 찾아내고, 이를 바탕으로 부서별 역할을 재편성합니다.

**c. 부서 간 균형 잡힌 협력 네트워크 구축**

* **목표**: 각 부서가 협력하는 강도에 차이가 있을 때, 이를 균형 있게 맞추는 방식입니다. 가중치 기반 최단 경로 분석이나 그래프 신경망을 사용해 협력의 강도가 높은 부서들이 너무 집중되지 않도록 균형을 맞추는 방식으로 최적화할 수 있습니다.

**4. 부서 재편성 시 고려해야 할 요소**

* **부서의 역할과 중요도**: 각 부서의 중요도와 역할을 네트워크 분석을 통해 파악한 후, 이를 바탕으로 재편성해야 합니다.
* **협력의 지속성**: 부서 간 협력이 단기적인 성과만을 위한 것이 아니라 지속 가능한 협력 구조로 이어질 수 있도록 해야 합니다.
* **정책적 목표와 우선순위**: 재편성 시 정부의 정책 목표와 우선순위를 고려하여, 전략적으로 필요한 부서들 간의 협력 네트워크를 구축하는 것이 중요합니다.

**결론**

이러한 네트워크 분석과 머신러닝 기법을 결합하여 부서 간 협력 구조를 최적화하고, 부서 재편성을 효율적으로 진행할 수 있을 것입니다. 부서 간 협력 강도, 중요성, 네트워크 내 역할 등을 고려한 다양한 접근을 통해 더 나은 조직 구조를 만들어 갈 수 있습니다.