Sequential Pattern Mining Algorithms

Yoon Sang Cho

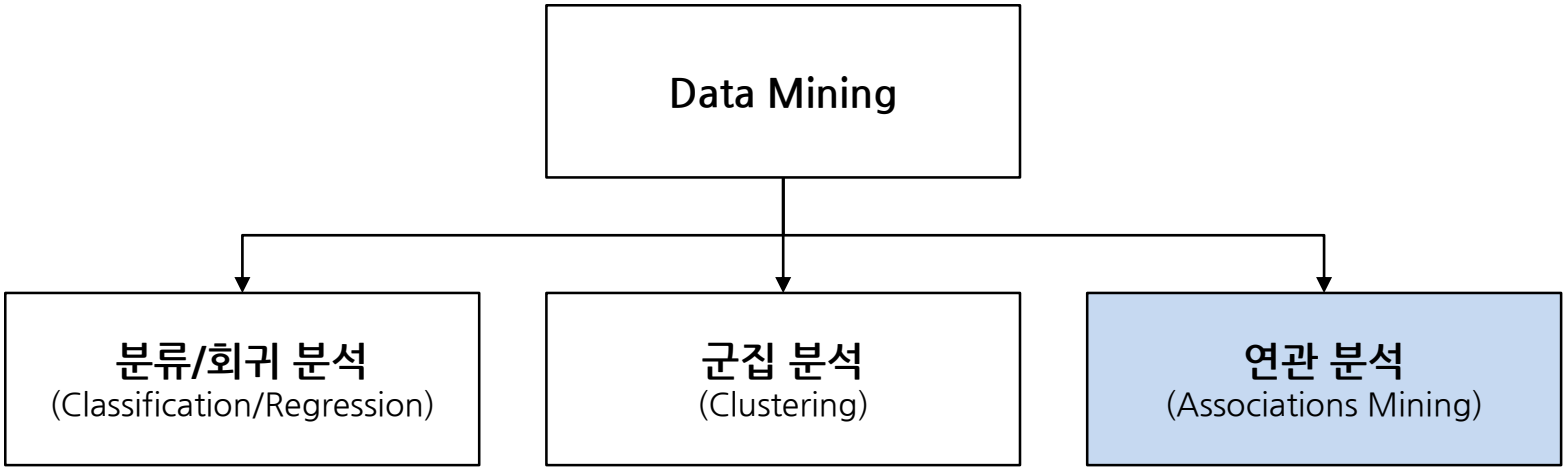
DMQA Seminar

목차

- 연관 분석
- 순차 패턴 분석
- 방법론
- 응용 사례
- 결론

연관 분석

❖ 연관 분석은 데이터 마이닝 분야 중 하나



연관 분석

- 연관 규칙(association rule)
  - '조건-결과' 식으로 표현되는 유용한 패턴(pattern)

- 이러한 규칙(패턴)을 발견해내는 것을 연관 분석(association analysis)
  - Apriori algorithm
    - 지지도, 신뢰도, 향상도를 기준으로 규칙 도출
    - 장바구니분석(market basket analysis)
      - 월마트사례

규칙1: 기저귀-맥주
규칙2: …

출처: lifeindigital.org

연관 규칙 측정 지표

❖ 지지도(support)

- ➢ 규칙: If A, then B
- ➢ 전체 관측치 중 특정 항목(예를 들어 A, B)이 동시에 포함되는 관측치 비율
- ➢ 전체적인 구매 경향 파악: 'A와 B가 같이 판매되고 있구나'

$$P(A \cap B) = \frac{A\text{와 } B\text{가 동시에 포함된 관측치 수}}{\text{전체 관측치수}}$$

트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

A와 B에 대한 지지도= $\frac{7}{10} = 0.7$

연관 규칙 측정 지표

❖ 신뢰도(confidence)

- ➢ 규칙: If A, then B
- ➢ 품목 A가 포함된 거래 중에서 품목 A, B를 동시에 포함하는 거래일 확률
- ➢ 연관성의 정도 파악: 'A를 샀던 사람이 B도 구매하더라'

$$\frac{P(A \cap B)}{P(A)} = \frac{A\text{와 } B\text{가 동시에 포함된 관측치 수}}{A\text{를 포함하는 거래수}}$$

트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

A와 B에 대한 신뢰도= $\frac{7}{7} = 1$

연관 규칙 측정 지표

❖ 향상도(lift)

- 규칙: If A, then B
- 품목 B를 구매한 고객 대비 품목 A를 구매한 후 품목 B를 구매한 고객에 대한 비율
- A와 B간 관계없는 경우 P(B|A)는 P(B)이므로 향상도 1

$$\frac{P(B|A)}{P(B)} = \frac{P(A \cap B)}{P(A)P(B)} = \frac{A\text{와 } B\text{를 포함하는 관측치} \times \text{전체 관측치 수}}{A \text{ 포함 관측치 수} \times B\text{포함 관측치수}}$$

트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

A와 B에 대한 향상도 $= \frac{7*10}{7*9} = \frac{70}{63} = 1.1$

향상도	의미
1	두 품목이 서로 독립
1보다 큼	두 품목이 서로 양의 관계
1보다 작음	두 품목이 서로 음의 관계

연관 규칙 측정 지표

- 연관 규칙 도출
  - Apriori rule: If A, then B
    - 지지도(0~1): 통계적 유의성(statistical significance)
    - 신뢰도(0~1): 연관 규칙의 강도(rule strength)
    - 향상도(> 1): 1보다 클 때 중요한 규칙
      - A와 B에 이어 다른 항목에 대해서도 탐색

트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

연관 규칙 측정 지표

❖ 최소 지지도(minimum support)

- ➢ 불필요한 탐색 시간을 줄이기 위해 사용자가 설정하는 하이퍼파라미터
- ➢ 처음에는 임의로 설정 후 계산 속도와 의미가 현실적인지 판단
- ➢ 높은 값에서 낮은 값으로 실행해야 효율적(10%, 5%, 2%, 1%)

최소 지지도: 50%(5)

트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

트랜잭션	지지도
A	**7**
B	**9**
C	1
D	1
E	1
F	1
G	1
…	1
Y	1
Z	1

연관 규칙 측정 지표

❖ 최소 지지도(minimum support)

- ➢ 불필요한 탐색 시간을 줄이기 위해 사용자가 설정하는 하이퍼파라미터
- ➢ 처음에는 임의로 설정 후 계산 속도와 의미가 현실적인지 판단
- ➢ 높은 값에서 낮은 값으로 실행해야 효율적(10%, 5%, 2%, 1%)

최소 지지도: 50%(5)

트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

트랜잭션	지지도
A	**7**
B	**9**
C	1
D	1
E	1
F	1
G	1
…	1
Y	1
Z	1

연관 규칙 측정 지표

❖ 연관 규칙 도출

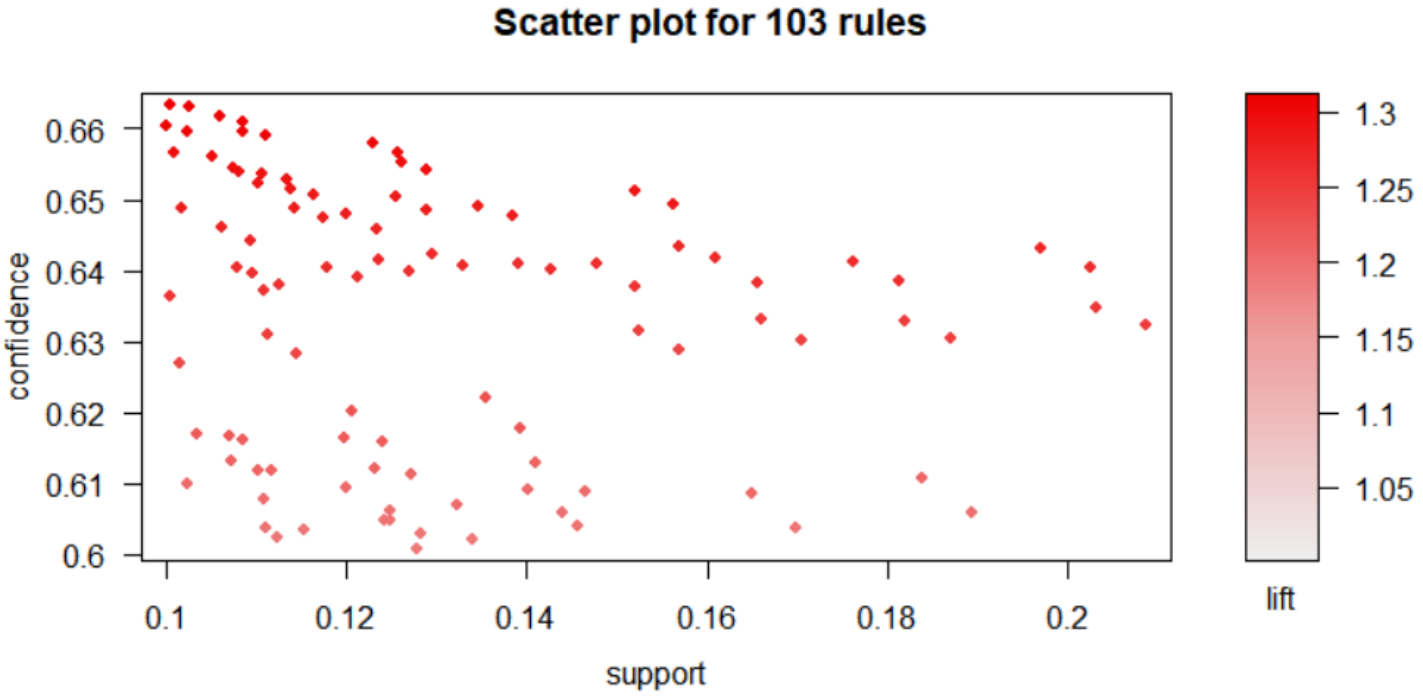
➢ 최소 지지도를 만족하고, 높은 수준의 신뢰도와 향상도를 갖는 규칙을 선정

```
> library(arules)
> data(Adult)
> rules = apriori(Adult,
+                 parameter=list(support = 0.1, confidence = 0.6),
+                 appearance = list(rhs=c('income=small', 'income=large'),
+                                   default='lhs'),
+                                   control=list(verbose=F))
> rules.sorted = sort(rules, by='lift')
> inspect(head(rules.sorted))
    lhs                                  rhs              support confidence     lift count
[1] {age=Young,
     workclass=Private,
     capital-loss=None}            => {income=small} 0.1005282  0.6633342 1.310622  4910
[2] {age=Young,
     workclass=Private}            => {income=small} 0.1025961  0.6630938 1.310147  5011
[3] {age=Young,
     marital-status=Never-married,
     capital-gain=None,
     capital-loss=None}            => {income=small} 0.1060563  0.6616426 1.307279  5180
[4] {age=Young,
     marital-status=Never-married,
     capital-gain=None}            => {income=small} 0.1084517  0.6609683 1.305947  5297
[5] {relationship=Own-child,
     capital-loss=None}            => {income=small} 0.1000983  0.6604972 1.305016  4889
[6] {relationship=Own-child}       => {income=small} 0.1023914  0.6596755 1.303393  5001
```

연관 규칙 측정 지표

❖ 연관 규칙 도출

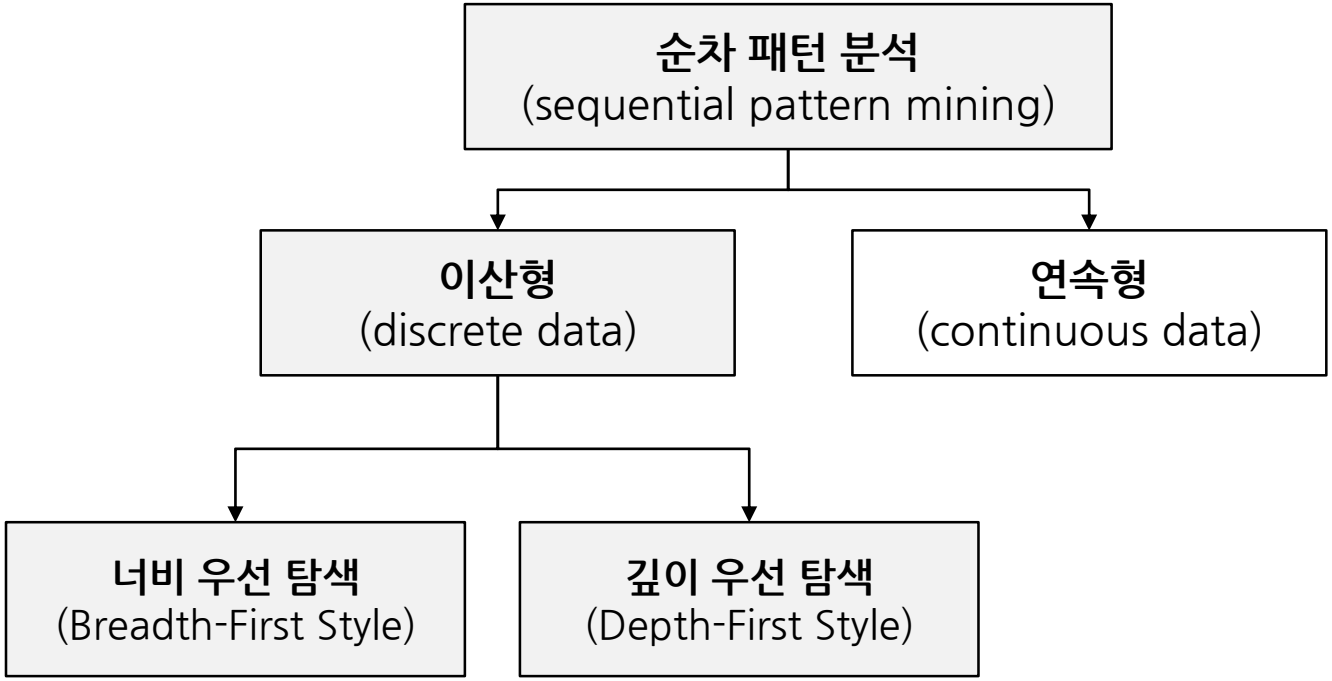
➢ 최소 지지도를 만족하고, 높은 수준의 신뢰도와 향상도를 갖는 규칙을 선정



순차 패턴 분석

❖ Sequential Pattern Mining

- ➢ 각 원소 사이에 순서가 있는 시퀀스 데이터 베이스에서 **순서를 고려**하여 **빈번**하게 나타나는 패턴(pattern)을 찾아주는 기술
- ➢ 지지도(support) 기준 탐색



순차 패턴 분석

- 시퀀스 데이터(categorical sequences)
  - 변수가 순서 의미를 갖는 데이터
  - 관측치 형태: event, item, element

Transaction

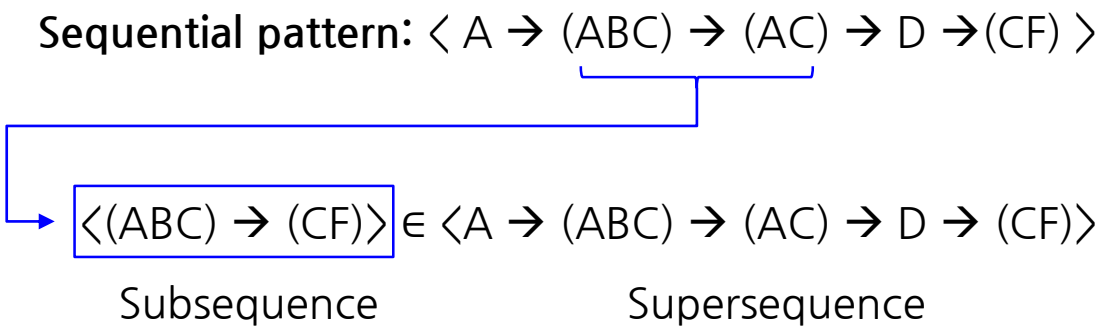
트랜잭션	구매 품목			
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

Sequence

트랜잭션	시점 1	시점 2	시점 3	시점 4
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z

순차 패턴 분석

- 시퀀스 데이터(categorical sequences)
  - 변수가 순서 의미를 갖는 데이터
  - 관측치 형태: event, item, element
  - *'어떤 패턴이 빈번하면 그 패턴의 일부분도 모두 빈번하다'*

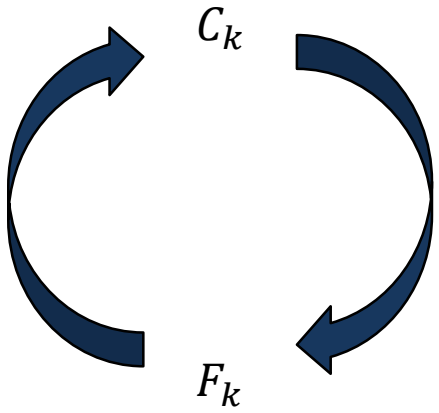


GSP: Generalized Sequential Pattern

- GSP algorithm: Apriori 기반
  - 크기가 1인 후보집합부터 시작해 빈번한 집합을 찾아가는 방식
  - 지지도를 기반으로 길이가 가장 긴 빈번한 순차 패턴 탐색
    - F_k: 길이가 k인 빈번한 패턴(frequent pattern)
    - C_k: 길이가 k인 후보 패턴(candidate pattern)
      - $\boldsymbol{F_k}$가 될 가능성이 큰 패턴

Sequence data

트랜잭션	시점 1	시점 2	시점 3	시점 4
고객 1	**A**	**B**	C	D
고객 2	**A**	**B**	E	F
고객 3	**B**	**A**	G	H
고객 4	**B**	**A**	I	J
고객 5	K	L	**A**	**B**
고객 6	M	N	**A**	**B**
고객 7	**A**	O	**B**	P
고객 8	Q	R	S	T
고객 9	U	V	**B**	W
고객 10	X	Y	**B**	Z



GSP: Generalized Sequential Pattern

- GSP algorithm: Apriori 기반
  - 지지도: 각 시퀀스에 해당 항목이 있는지 없는지 여부
  - 지지도 100%는 시퀀스 개수와 동일
  - C_1: 데이터 내 모든 아이템

최소 지지도 = 100%(4)

고객 ID	구매 기록(sequence)
1	〈 (맥주, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문) → (오징어) → (소주) → (맥주) → (신문, 땅콩) 〉
3	〈 (맥주, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (맥주, 오징어) → (신문, 양주) → (신문, 오징어)〉

C_1	지지도
맥주	4
오징어	4
신문	4
~~땅콩~~	~~3~~
~~소주~~	~~1~~
~~양주~~	~~1~~

GSP: Generalized Sequential Pattern

- GSP algorithm: Apriori 기반
  - C_{k+1}은 F_k로부터 생성
  - F_k로 생성된 항목의 모든 조합
    - 다른 시점(Temporal join)
    - 한 시점(Non-temporal join)

F_1
맥주
오징어
신문

F_1	맥주	오징어	신문
맥주	*	*	*
오징어	*	*	*
신문	*	*	*

F_1	맥주	오징어	신문
맥주		*	*
오징어			*
신문			

C_2	
다른 시점	**한 시점**
맥주 → 맥주	(맥주, 오징어)
맥주 → 오징어	(맥주, 신문)
맥주 → 신문	(오징어, 신문)
오징어 → 오징어	
오징어 → 맥주	
오징어 → 신문	
신문 → 신문	
신문 → 오징어	
신문 → 맥주	

GSP: Generalized Sequential Pattern

- GSP algorithm: Apriori 기반
  - C_{k+1}은 F_k로부터 생성
  - F_k로 생성된 항목의 모든 조합
    - 다른 시점(Temporal join)
    - 한 시점(Non-temporal join)

C_2	
다른 시점	**한 시점**
맥주 → 맥주	맥주, 오징어
맥주 → 오징어	맥주, 신문
맥주 → 신문	오징어, 신문
오징어 → 오징어	
오징어 → 맥주	
오징어 → 신문	
신문 → 신문	
신문 → 오징어	
신문 → 맥주	

C_2	지지도
맥주 → 맥주	1
맥주 → 오징어	3
맥주 → 신문	**4**
오징어 → 오징어	2
오징어 → 맥주	0
오징어 → 신문	**4**
신문 → 신문	3
신문 → 오징어	3
신문 → 맥주	2
(맥주, 오징어)	2
(맥주, 신문)	1
(신문, 오징어)	1

F_2
맥주 → 신문
오징어 → 신문

데이터 내 가장 빈번한 순차 패턴

GSP: Generalized Sequential Pattern

- GSP algorithm: Apriori 기반
  - C_{k+1}을 더 이상 만들 수 없을 때 종료
  - 후보 패턴을 만드는데 시간이 걸림

패턴 길이 (k)	후보 패턴 (C_k)	빈번 패턴 (F_k)
1	<(맥주)>, <(신문)>, <(오징어)>, <(땅콩)>, <(소주)>, <(양주)>	<(맥주)>, <(신문)>, <(오징어)>
2	<(맥주), (맥주)>, <(맥주, 신문)>, <(맥주), (신문)>, <(맥주, 오징어)>, <(맥주), (오징어)>, <(신문), (맥주)>, <(신문), (신문)>, <(신문, 오징어), <(오징어), (맥주)>, <(오징어), (신문)>, <(오징어), (오징어)>	**<(맥주), (신문)>, <(오징어), (신문)>**
3	없음	없음

PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan Tree를 만들어가며 빈번한 패턴 탐색
    - Prefix, Suffix(projected DB)
    - Prefix를 먼저 할 대상을 찾는 기준: 지지도

Prefix and Suffix?

SID	Sequence
1	A → B → C → D
2	A → C → D
3	B → A → C → D
4	D → A → B → C

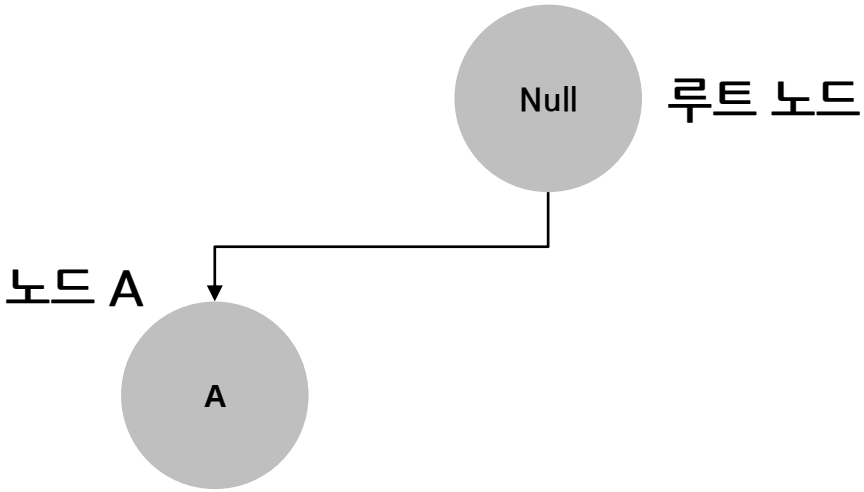
prefix
⟨(A)⟩

SID	Suffix(Sequence)
1	B → C → D
2	C → D
3	C → D
4	B → C

⟨(A)⟩-projected DB

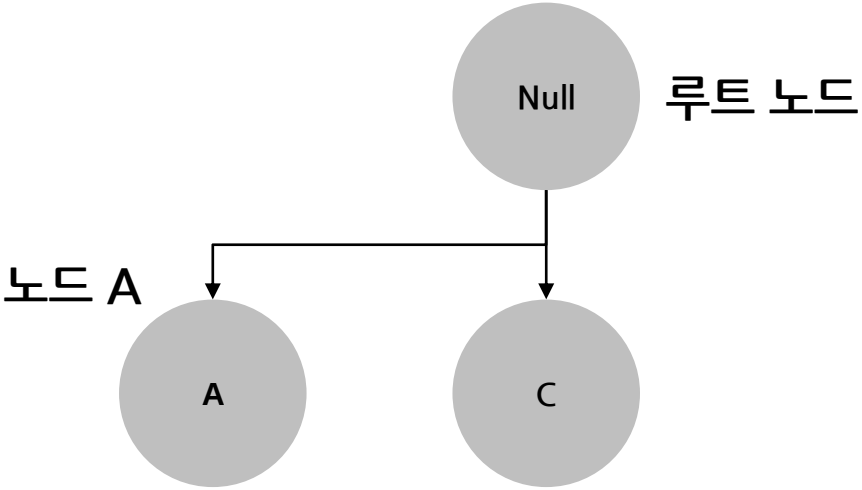
PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
    - PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색



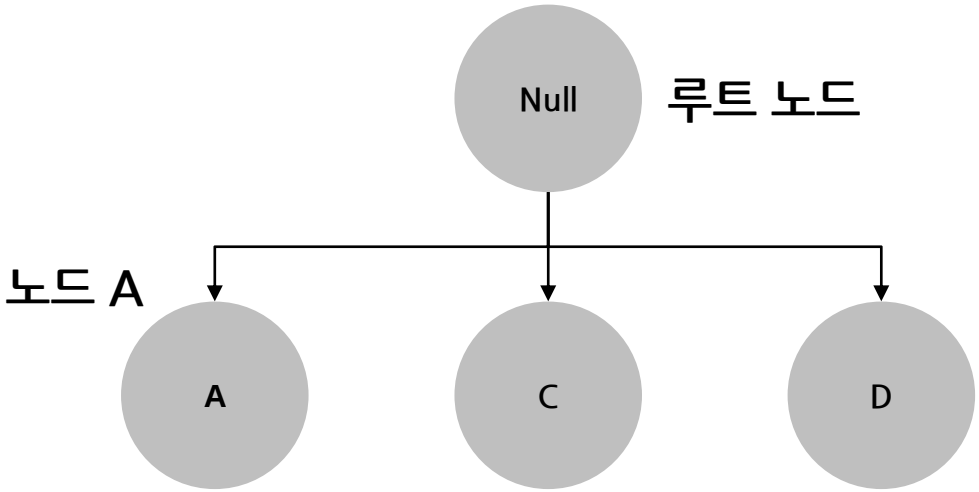
PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
    - PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색



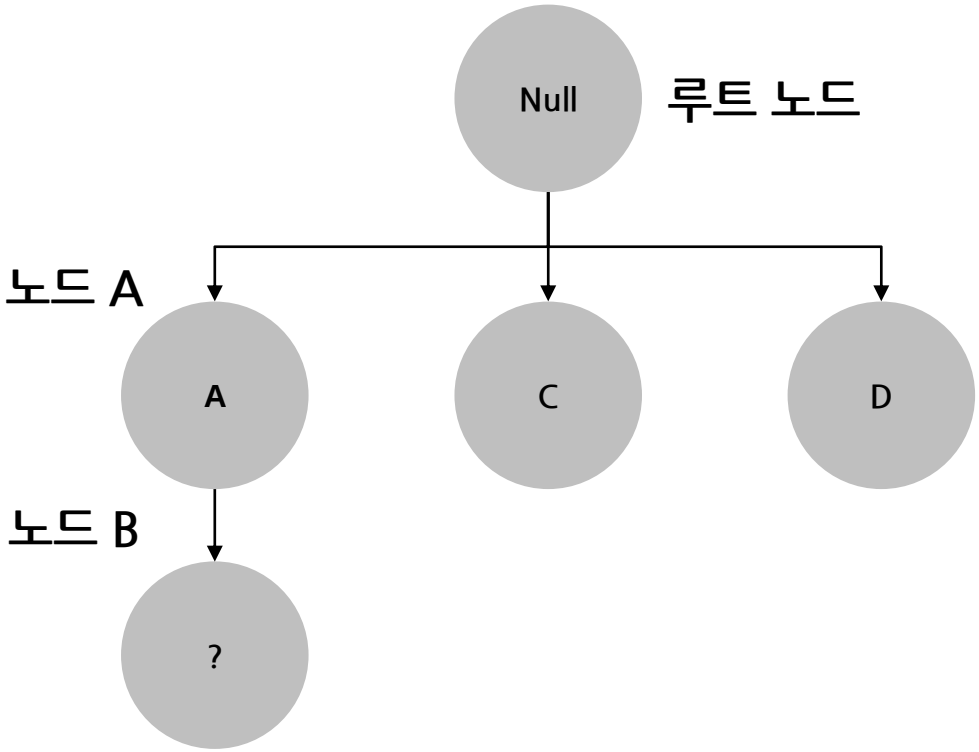
PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
    - PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색



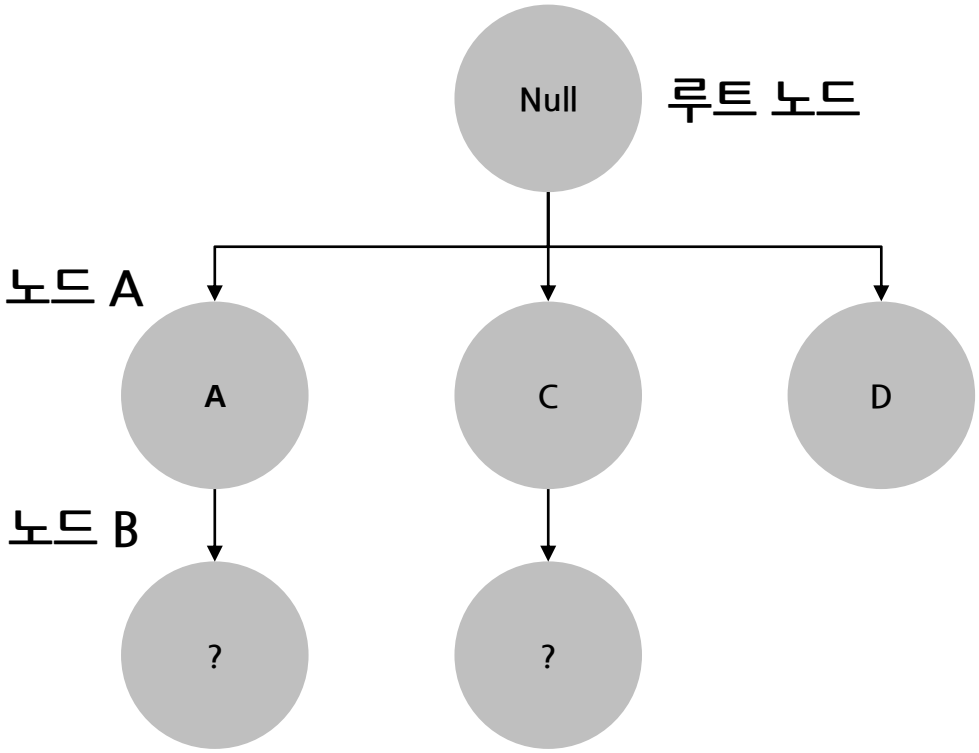
PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
    - PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색



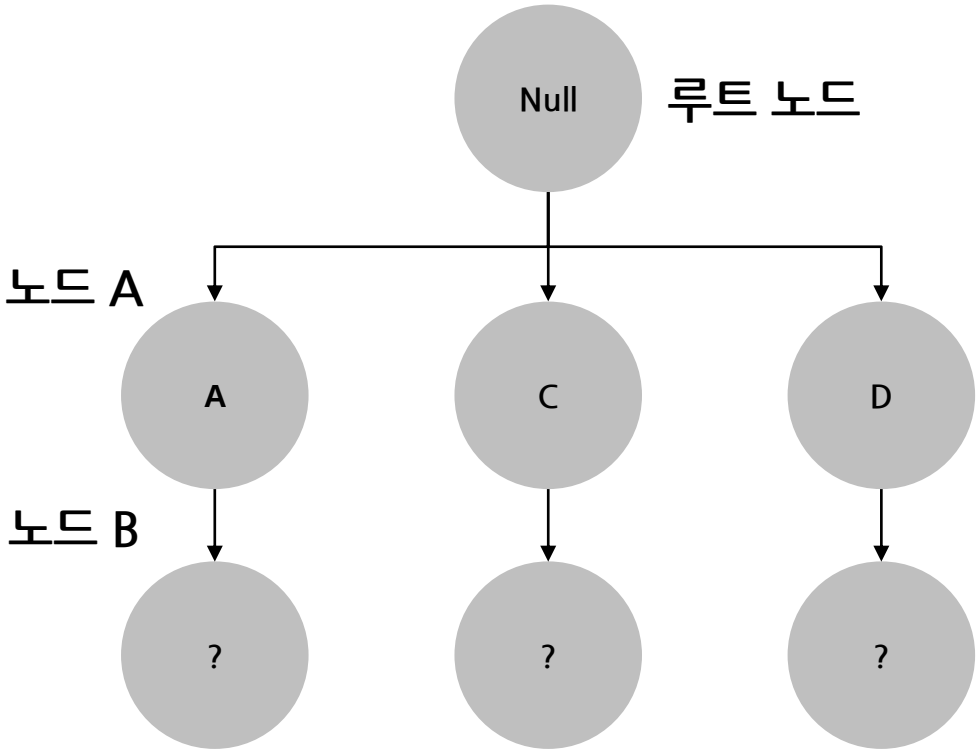
PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
    - PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색



PrefixSpan

- 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법
  - 상대적으로 GSP보다 대용량 데이터에 적합
  - PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
    - PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색



PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (맥주, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문) → (오징어) → (소주) → (맥주) → (신문, 땅콩) 〉
3	〈 (맥주, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (맥주, 오징어) → (신문, 양주) → (신문, 오징어)〉

항목	지지도
맥주	**4**
오징어	4
신문	4
땅콩	3
소주	1
양주	1

F_1

최소 지지도: 4(100%)

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (**맥주**, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문) → (오징어) → (소주) → (**맥주**) → (신문, 땅콩) 〉
3	〈 (**맥주**, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (**맥주**, 오징어) → (신문, 양주) → (신문, 오징어)〉

항목	지지도
맥주	**4**
오징어	**4**
신문	**4**
땅콩	3
소주	1
양주	1

F_1

최소 지지도: 4(100%)

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (_, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문) → (오징어) → (소주) → (_) → (신문, 땅콩) 〉
3	〈 (_, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (_, 오징어) → (신문, 양주) → (신문, 오징어)〉

항목	지지도
맥주	**4**
오징어	**4**
신문	**4**
땅콩	3
소주	1
양주	1

F_1

최소 지지도: 4(100%)

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (_, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문, 땅콩) 〉
3	〈 (_, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (_, 오징어) → (신문, 양주) → (신문, 오징어)〉

항목	지지도
맥주	**4**
오징어	**4**
신문	**4**
땅콩	3
소주	1
양주	1

F_1

최소 지지도: 4(100%)

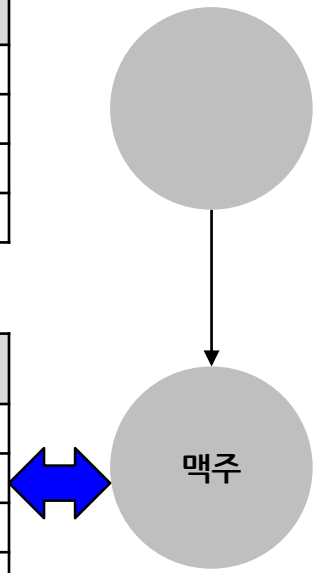
PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색

고객 ID	구매 기록(sequence)
1	〈 (**맥주**, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문) → (오징어) → (소주) → (**맥주**) → (신문, 땅콩) 〉
3	〈 (**맥주**, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (**맥주**, 오징어) → (신문, 양주) → (신문, 오징어)〉

고객 ID	구매 기록(suffix sequence)
1	〈 (_, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문, 땅콩) 〉
3	〈 (_, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (_, 오징어) → (신문, 양주) → (신문, 오징어)〉



〈(맥주)〉-projected DB

'〈(맥주)〉-노드'

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (_, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문, 땅콩) 〉
3	〈 (_, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (_, 오징어) → (신문, 양주) → (신문, 오징어)〉

항목	지지도
맥주	1
오징어	3
신문	**4**
땅콩	3
소주	0
양주	1

F_2

최소 지지도: 4(100%)

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (_, 땅콩) → (맥주, 오징어) → (**신문**)〉
2	〈 (**신문**, 땅콩) 〉
3	〈 (_, 신문) → (오징어) → (오징어, 땅콩) → (**신문**)〉
4	〈 (_, 오징어) → (**신문**, 양주) → (신문, 오징어)〉

항목	지지도
맥주	1
오징어	3
신문	**4**
땅콩	3
소주	0
양주	1

F_2

최소 지지도: 4(100%)

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	〈 (_, 땅콩) → (맥주, 오징어) → (_)〉
2	〈 (_, 땅콩) 〉
3	〈 (_, 신문) → (오징어) → (오징어, 땅콩) → (_)〉
4	〈 (_, 오징어) → (_, 양주) → (신문, 오징어)〉

항목	지지도
맥주	1
오징어	3
신문	**4**
땅콩	3
소주	0
양주	1

F_2

최소 지지도: 4(100%)

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	Null
2	〈 (_, 땅콩) 〉
3	Null
4	〈(_, 양주) → (신문, 오징어)〉

항목	지지도
맥주	1
오징어	3
신문	**4**
땅콩	3
소주	0
양주	1

F_2

최소 지지도: 4(100%)

더 이상 최소 지지도를 만족하는 패턴(F_3) 0개, Stop!

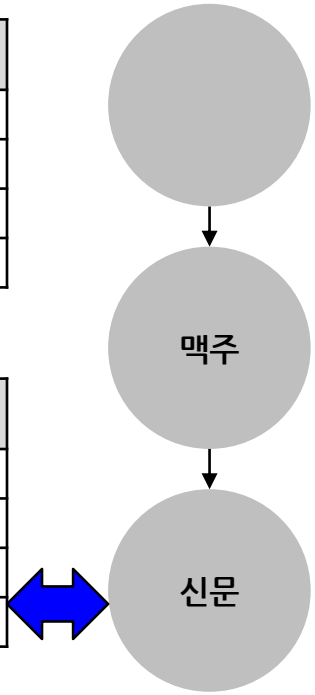
PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색

고객 ID	구매 기록(sequence)
1	〈 (**맥주**, 땅콩) → (맥주, 오징어) → (신문)〉
2	〈 (신문) → (오징어) → (소주) → (**맥주**) → (신문, 땅콩) 〉
3	〈 (**맥주**, 신문) → (오징어) → (오징어, 땅콩) → (신문)〉
4	〈 (**맥주**, 오징어) → (신문, 양주) → (신문, 오징어)〉

고객 ID	구매 기록(suffix sequence)
1	Null
2	〈 (_, 땅콩) 〉
3	Null
4	〈(_, 양주) → (신문, 오징어)〉



〈(맥주), (신문)〉-projected DB

‘〈(맥주) , (신문)〉-노드’

PrefixSpan

❖ 후보 패턴을 만들지 않으면서 빈번한 패턴을 찾는 방법

- ➢ 상대적으로 GSP보다 대용량 데이터에 적합
- ➢ PrefixSpan-Tree를 만들어가며 빈번한 패턴 탐색
  - ➢ PrefixSpan-Tree: 각 노드의 projected-DB에서 F_k를 탐색

고객 ID	구매 기록(sequence)
1	⟨ (맥주, 땅콩) → (맥주, 오징어) → (신문)⟩
2	⟨ (신문) → (오징어) → (소주) → (맥주) → (신문, 땅콩) ⟩
3	⟨ (맥주, 신문) → (오징어) → (오징어, 땅콩) → (신문)⟩
4	⟨ (맥주, 오징어) → (신문, 양주) → (신문, 오징어)⟩

항목	지지도
맥주	4
오징어	**4**
신문	4
땅콩	3
소주	1
양주	1

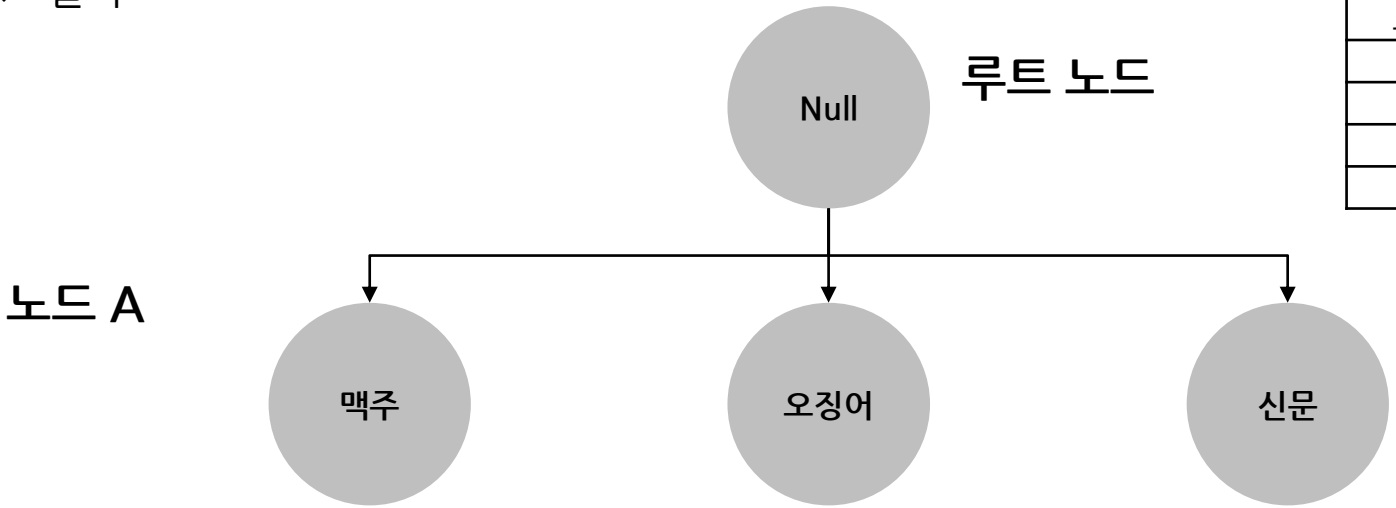
F_1

최소 지지도: 4(100%)

PrefixSpan

- 다시 깊이 우선 탐색에 따라 루트노드에서 다른 아이템들로 노드를 확장
- 결과

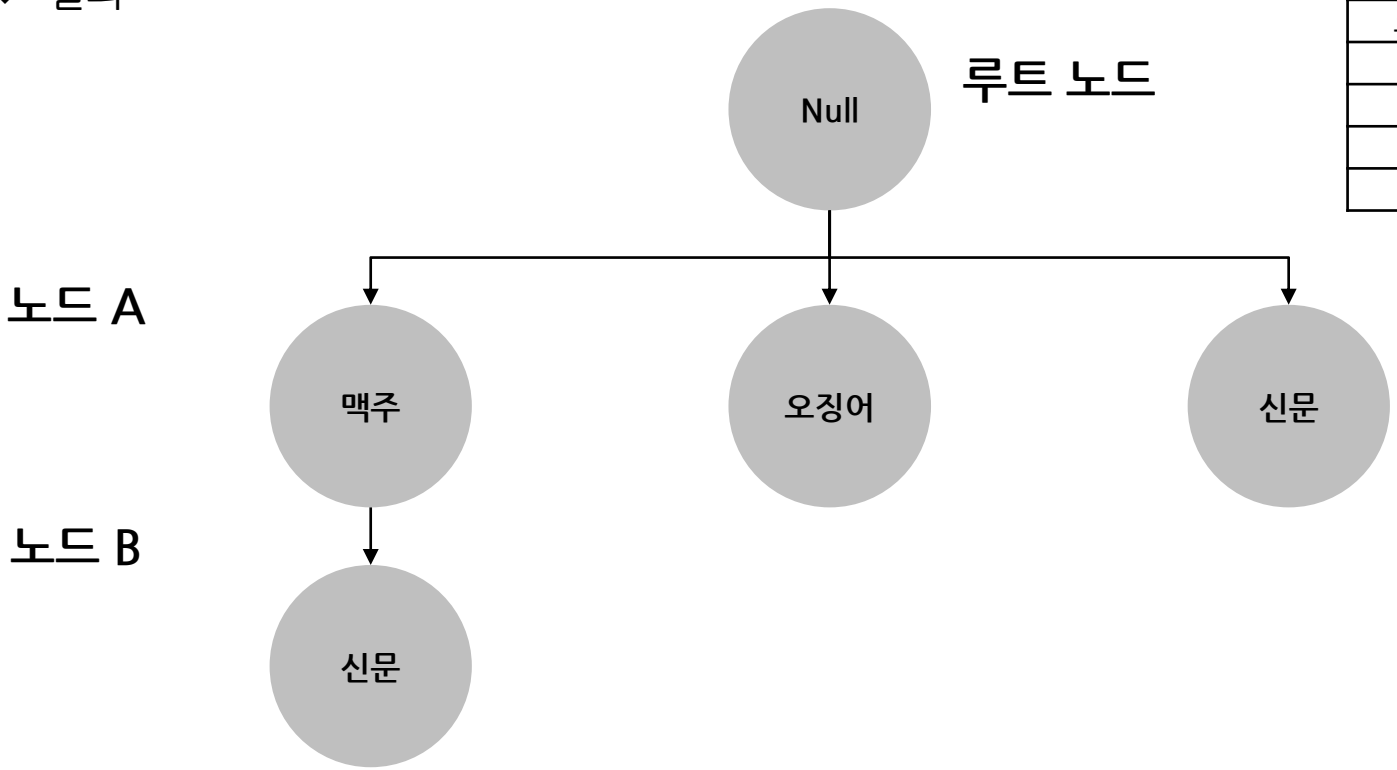
항목	지지도
맥주	4
오징어	4
신문	4
땅콩	3
소주	1
양주	1



PrefixSpan

- 다시 깊이 우선 탐색에 따라 루트노드에서 다른 아이템들로 노드를 확장
- 결과

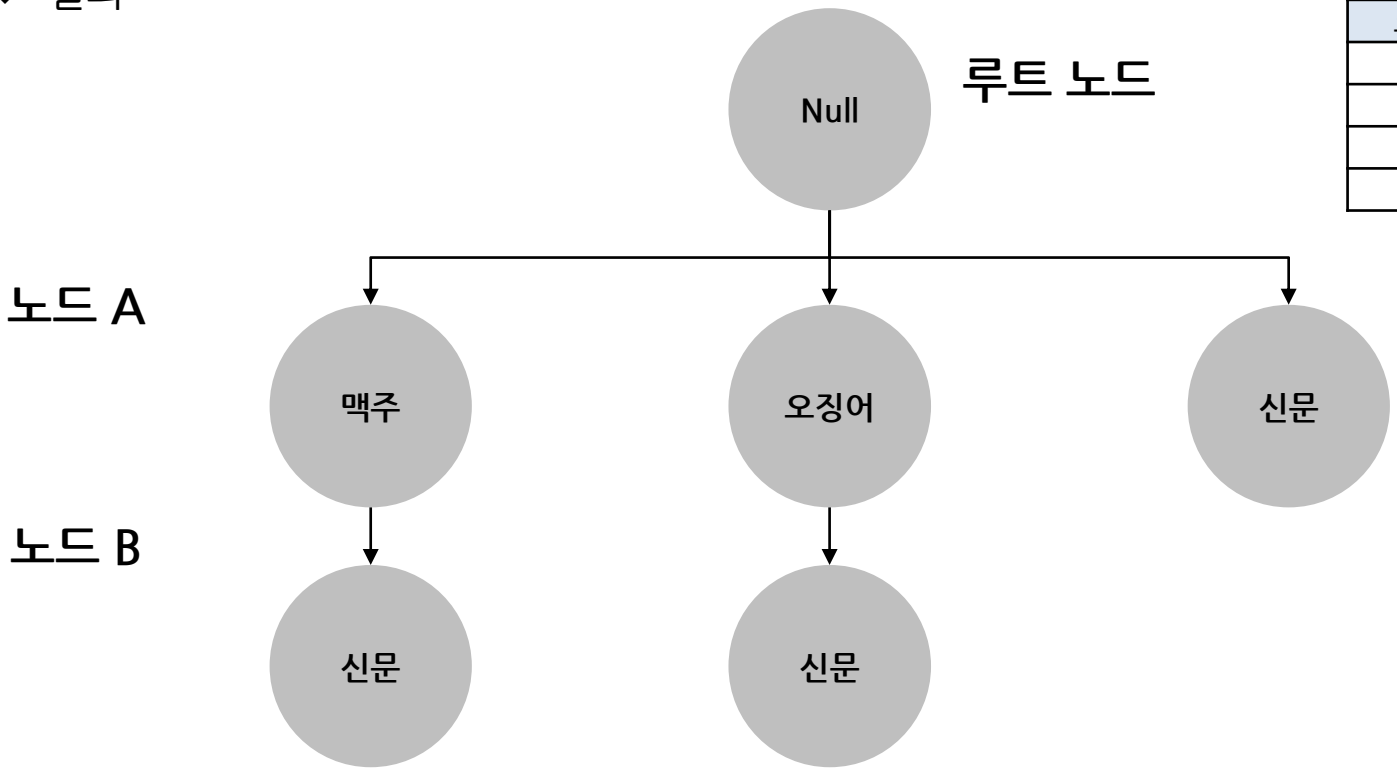
항목	지지도
맥주	4
오징어	4
신문	4
땅콩	3
소주	1
양주	1



PrefixSpan

- 다시 깊이 우선 탐색에 따라 루트노드에서 다른 아이템들로 노드를 확장
- 결과

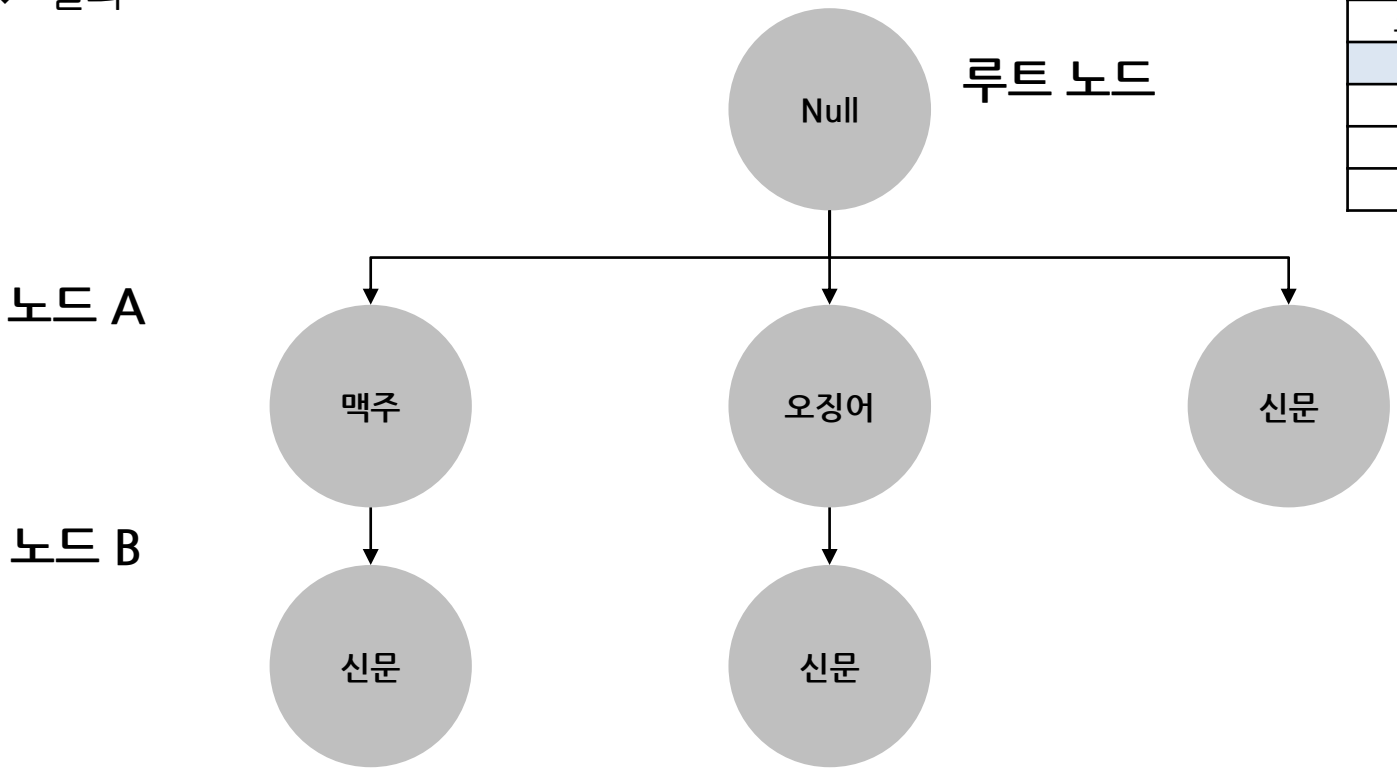
항목	지지도
맥주	4
오징어	4
신문	4
땅콩	3
소주	1
양주	1



PrefixSpan

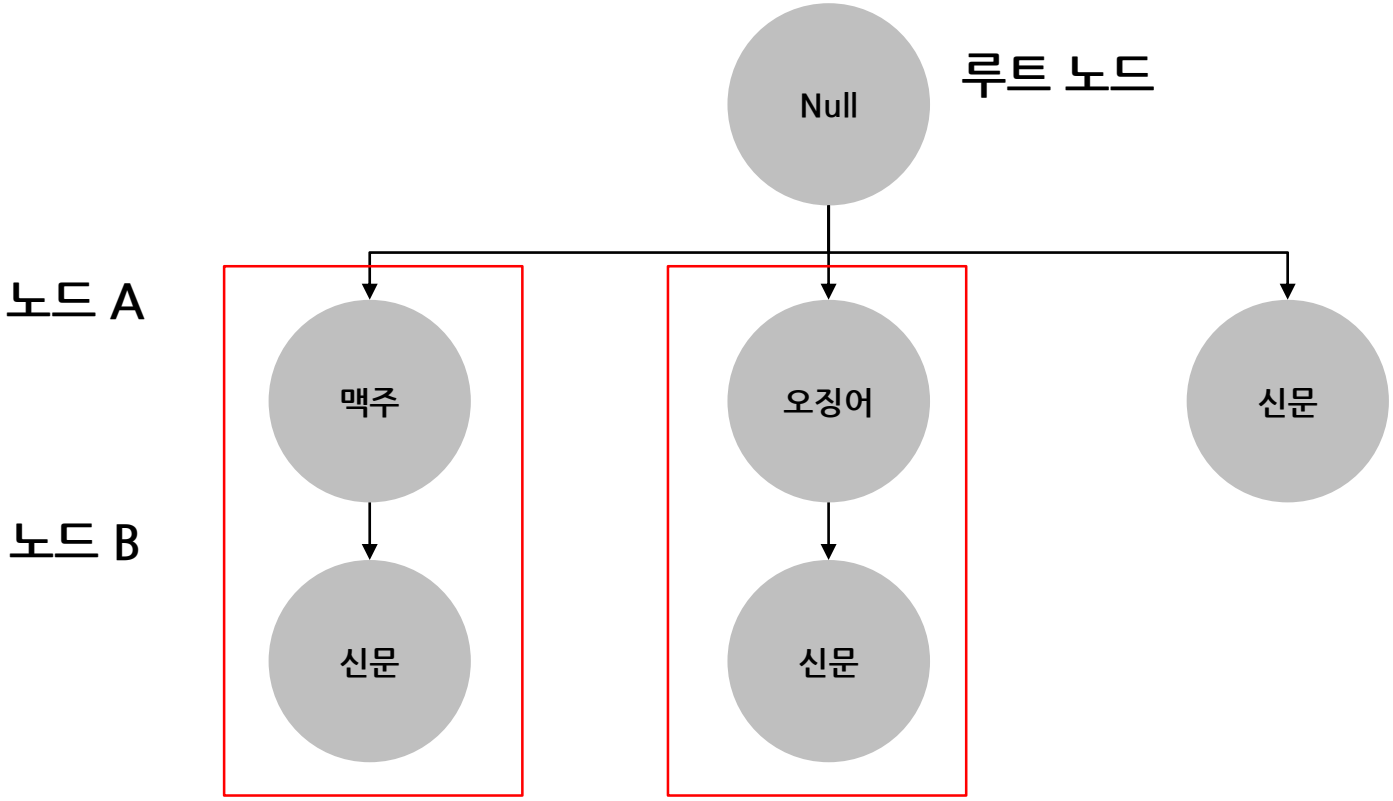
- 다시 깊이 우선 탐색에 따라 루트노드에서 다른 아이템들로 노드를 확장
- 결과

항목	지지도
맥주	4
오징어	4
신문	4
땅콩	3
소주	1
양주	1



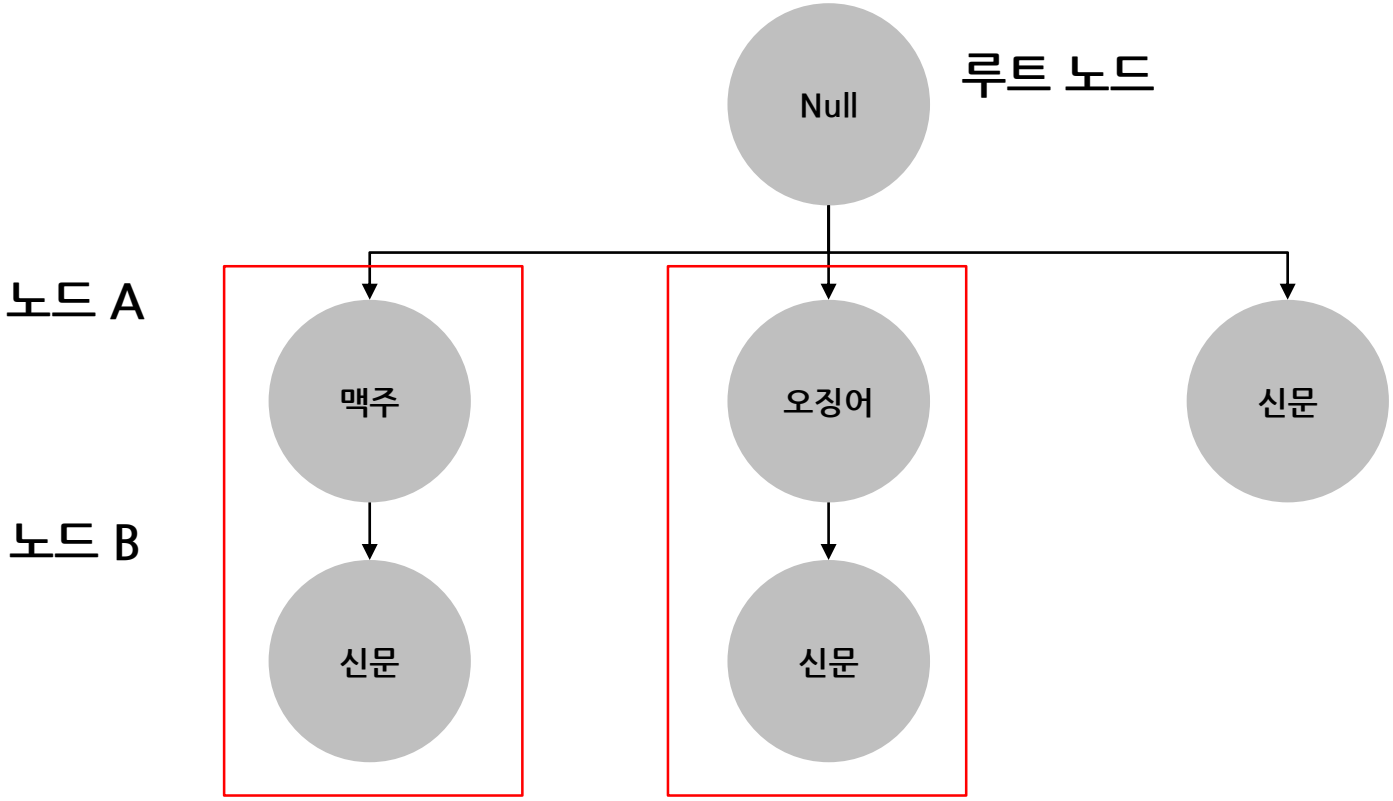
PrefixSpan

❖ 다시 깊이 우선 탐색에 따라 루트노드에서 다른 아이템들로 노드를 확장



PrefixSpan

❖ Big Mistake



Others

❖ **SPIRIT 알고리즘**

- 알고리즘 적용 결과 도출되는 시퀀스 중 특정 시퀀스를 포함하는 패턴들에게만 특별히 관심이 있는 경우 외의 패턴들은 불필요한 정보이며 계산량만 늘어나게 됨
- 이러한 문제를 해결하는 방법으로 후보 패턴을 만들 때 사용자가 관심있는 시퀀스를 정규식(regular expression) 형태로 나타내 그 조건을 만족하는 패턴들만 고려

❖ **SPADE 알고리즘**

- Sequential pattern mining in **Vertical Format Database**

Vertical sequence data

Sequence ID	Event ID	Events
1	1	A
1	2	ABC
1	3	AC
1	4	D
1	5	CF
2	1	AD
2	2	C
2	3	BC
2	4	AE
3	1	EF
3	2	AB
3	3	DF
3	4	C
3	5	B
4	1	E
4	2	G
4	3	AF
4	4	C
4	5	B
4	6	C

Others

❖ Hidden Markov Model(HMM)

- ➢ 순차패턴 인식에 효과적인 생성모델 계열의 기계학습 모델
- ➢ 시퀀스가 진행됨에 따라 시퀀스의 중요도를 확률로 표현
- ➢ 사용자가 설정하는 임계값보다 높은 길이의 시퀀스 도출

HMM 기반 순차 패턴 인식

Seq ID	시점1	시점2	시점3	…	시점t
1	A1	B4	C6	…	T6
2	A5	B1	C4	…	T5
3	A7	B2	C2	…	T2
…	⋮	⋮	⋮	⋮	⋮
146	A2	B2	C6	…	T7
147	A3	B3	C3	…	T5
148	A3	B3	C3	…	T3
149	A3	B3	C3	…	T3
150	A2	B5	C8	…	T6

Observation		확률
시퀀스 ID	이벤트 ID	
1	A16	0.0001
1	A17	0.0001
1	A18	0.0001
1	**A19**	**0.0659**
1	A20	0.0001
2	B0	0.0001
2	B32	0.0001
2	**B5**	**0.0911**
2	B10	0.0001
2	B11	0.0000
…	…	…
150	T22	0.0001
150	T23	0.0001
150	**T24**	**0.0999**
150	T25	0.0001
150	T26	0.0001

순차 패턴

응용 사례

❖ **수면 이벤트 시퀀스 데이터**

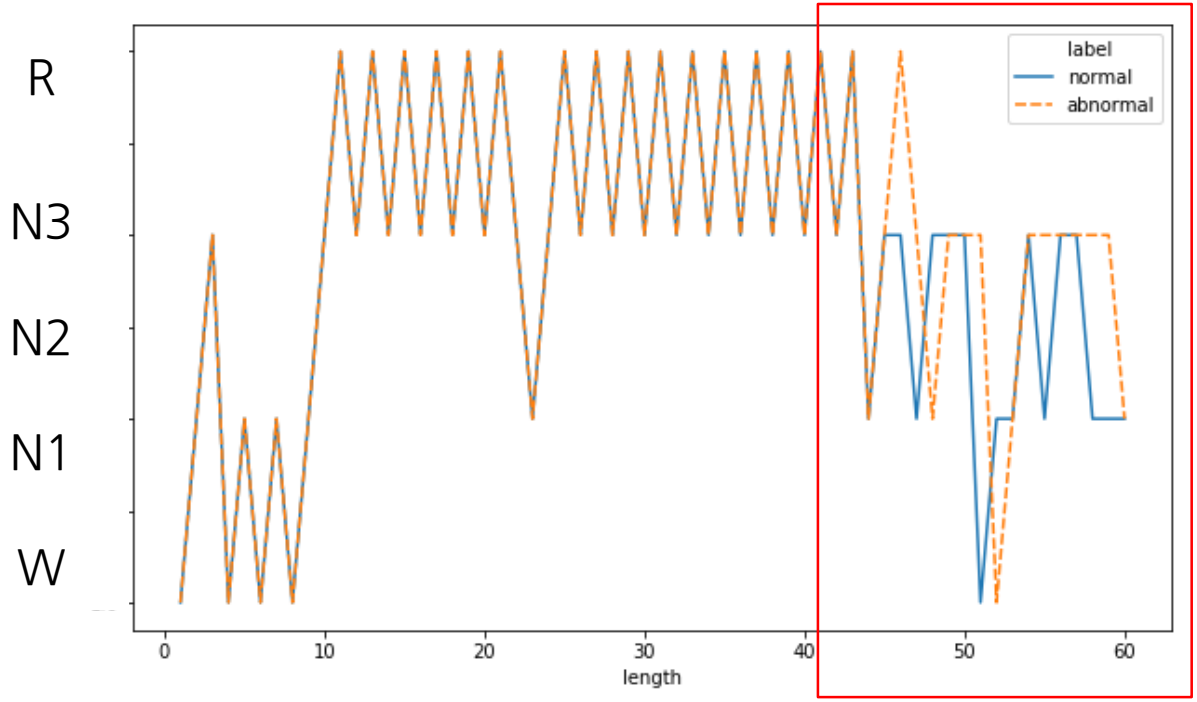
- PrefixSpan 적용
- 수면상태: W(얕은 수면), N1, N2, N3, R(깊은 수면) 순으로 수면 정도를 의미
- 정상인과 수면질환 환자의 수면 패턴 차이 분석

Study No.	Event 1	Event 2	Event 3	Event 4	…	Event t-2	Event t-1	Event t	Label
AI0315	W	N1	N2	**R**	…	N2	N1	W	Normal
AI0316	W	N1	N2	**R**	…	N2	N1	W	Normal
AI0321	W	N1	N2	**R**	…	N2	N1	W	Normal
…	…	…	…	…	…	…	…	…	…
LE018	W	N1	N2	R	…	N2	N1	W	Abnormal
LE019	W	N1	N2	R	…	N2	N1	W	Abnormal
LE020	W	N1	N2	R	…	N2	N1	W	Abnormal

응용 사례

❖ **수면 이벤트 시퀀스 데이터**

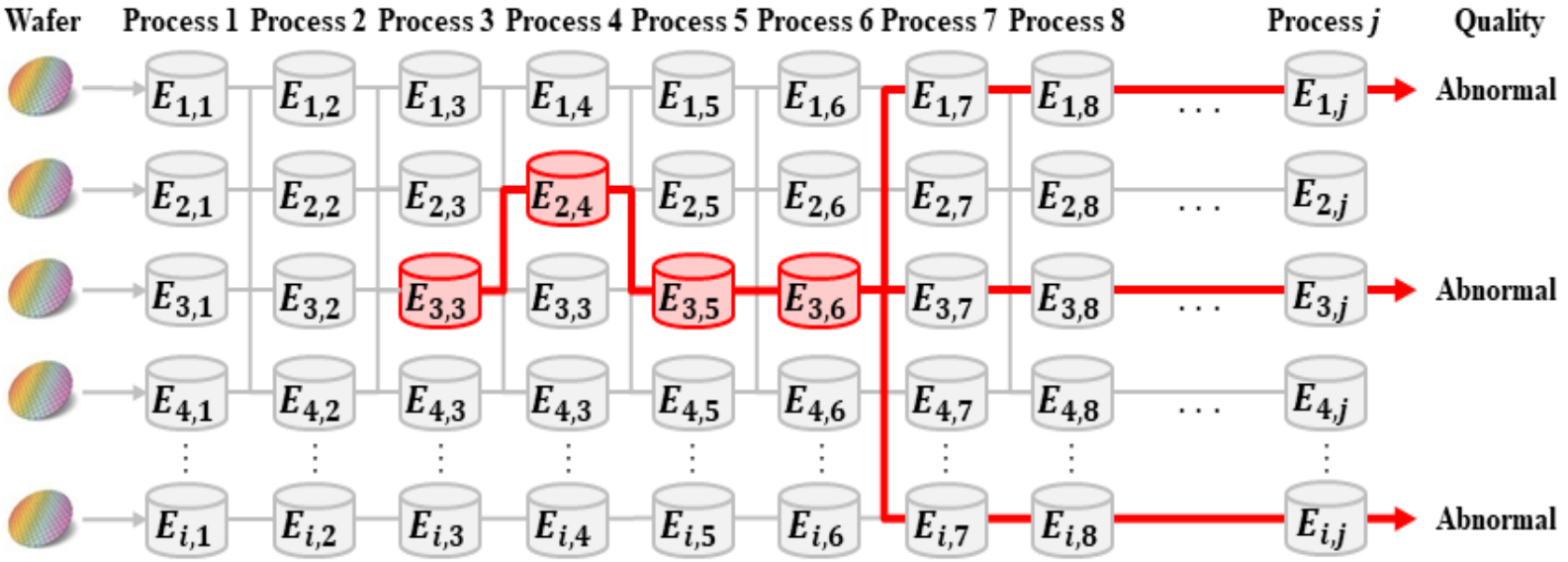
- ➢ PrefixSpan 적용
- ➢ 수면상태: W(얕은 수면), N1, N2, N3, R(깊은 수면) 순으로 수면 정도를 의미
- ➢ 정상인과 수면질환 환자의 수면 패턴 차이 분석



응용 사례

❖ **공정 설비 시퀀스데이터**

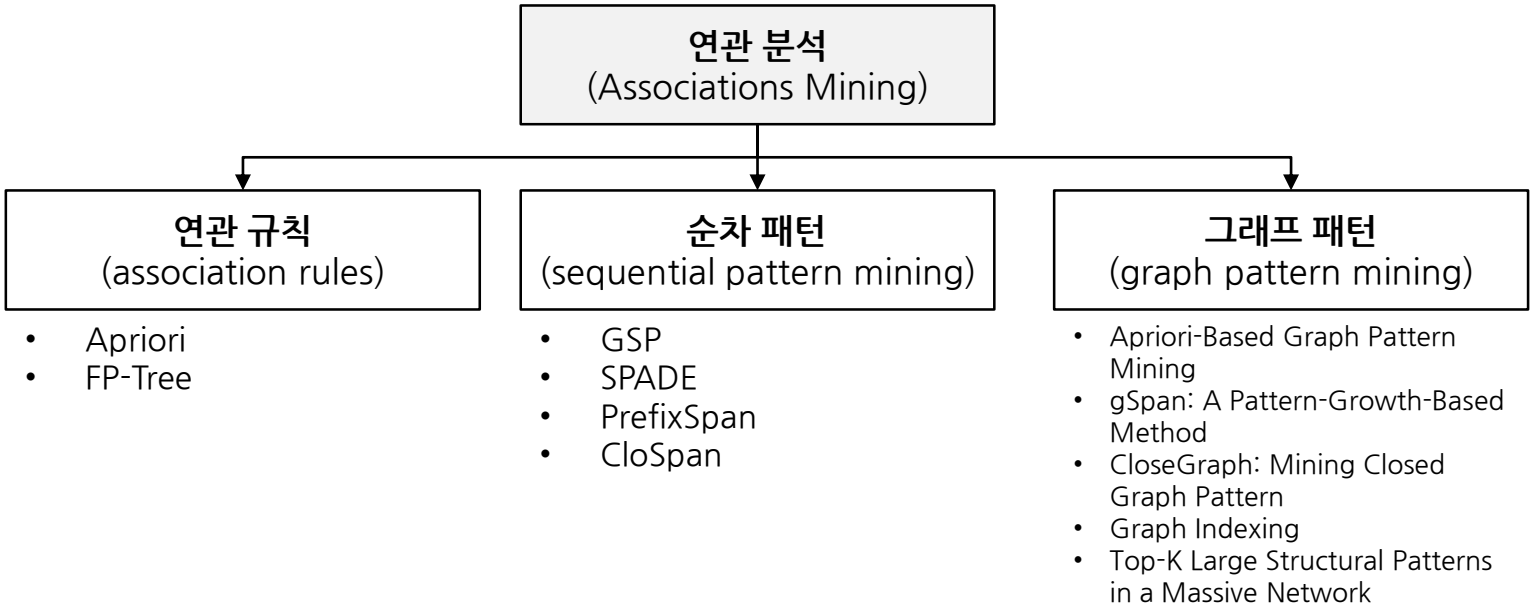
- ➢ Hidden Markov Model 적용
- ➢ 반도체 제조공정 내 불량 설비 시퀀스 탐지



결론

❖ 패턴 분석

- ➢ 연속형 시퀀스 데이터일 때 다양한 관측치를 이산화 하여 사용
- ➢ 이산화를 EVENT로 하면 순차 패턴 분석 기법 적용
- ➢ 이산화를 그래프로 할 수 있다면 그래프 패턴 분석 기법 적용



결론

- 연관 분석
  - 연관 규칙(Apriori rule)을 도출하는 분석 기법
  - 규칙 도출 기준 척도: 지지도, 신뢰도, 향상도
  - Apriori 기반 알고리즘: 계산이 쉬운 장점, 계산 시간이 방대한 단점

- 순차 패턴 분석
  - GSP: 지지도 기반 너비우선 탐색
  - Prefix: 지지도 기반 깊이 우선 탐색
  - HMM: 기계학습 모델 기반 순차 패턴 분석
  - 응용 사례

- 요약
  - 실제로는 방대한 데이터를 다루어야 하는 시간이 매우 오래 걸림
  - CPU 사용을 줄이기 위해 다양한 자료구조와 방법을 사용

감사합니다