부합 분석에서 오토마타를 이용한 분석값 표현법 + α

김세원
PLRG @ KAIST

부합 분석 소개

다루는 것

- 대상 프로그램: JSP, PHP, JavaScript (innerHTML, document.write, eval ...)
- 질문:
  - 문자열 표현식의 값이
  - 항상 주어진 문법(참조 문법 e.g. HTML)에 맞는가?

예제

```
x := [a
while ... do
  x := [ . x . ]
od
x := x . ]
print x
```

S → a | [S]

이 x의 값이 항상 이 문법에 맞습니까?

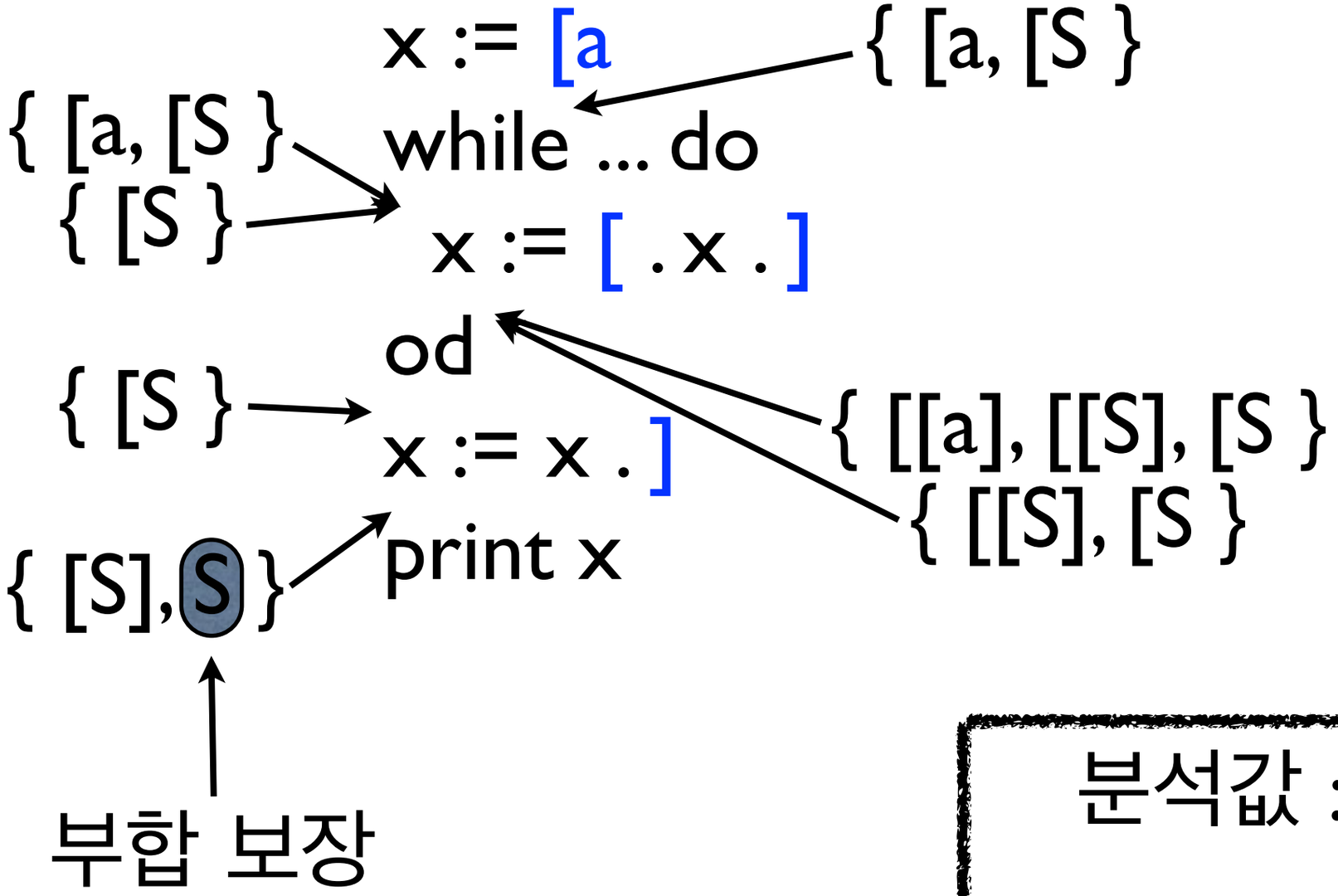
왜 어려운가?

- 가능한 문자열 값 갯수가 무한할 수도
- 튜링기계 언어(프로그램) ⊆ 문맥 자유 언어
  - 결정 불가능

내 요약 방법

- 문자열 집합을
  - 공통 축약(유도의 역방향)형 집합으로 요약
  - 축약형의 집합은 완벽하지 않을 수 있다

예제 풀이



S → a | [S]

분석값 : 가능한 문자열들의
공통 축약형 집합

형식화 & 안전

- 갈로아 연결, 요약 접합, ⊔ 등등
- 안전. 믿어도 됩니다.

문제는 분석값 표현

ε 생성규칙이 있으면

요약값이, 공집합(top) 빼고 모두 무한 집합

$$A \rightarrow \varepsilon$$

$$v \in U$$

$$\{v, Av, vA, AAv, AvA, AAv, AAAv, \ldots\} \subseteq U$$

축약에 대해 닫혀 있어야

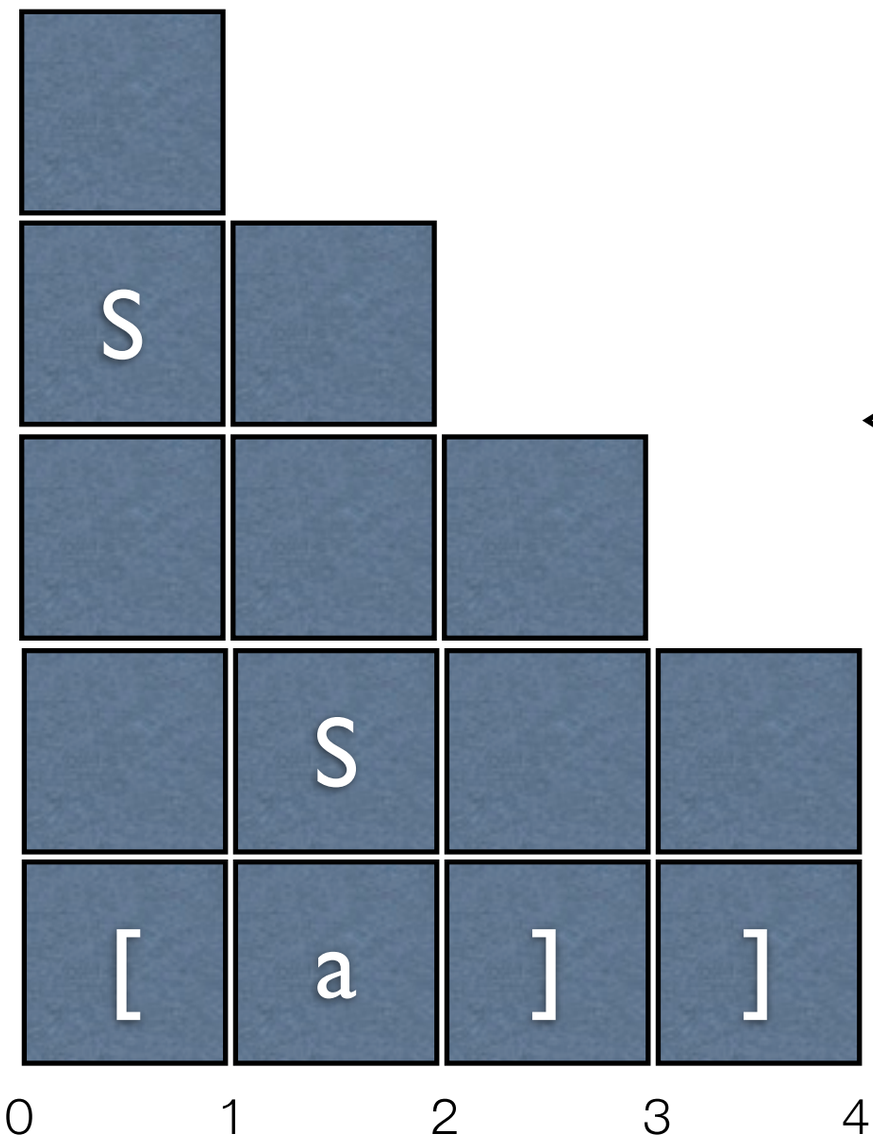
예전 해결책

- 분석값 집합에서 정확도에 공헌하지 않는 어휘 문자열 제외
  - 필요한 분석값을 모두 유한 집합으로 표현
- 단점
  - 직관적이지 않은 표현 / 알고리즘
  - 상수 문자열 요약의 집합 크기가 exponential < ∞

새로운 표현법

직관: CYK 파싱 테이블

S → a | [S]



α({ [a]] })
= { [a]], [S]], S] }

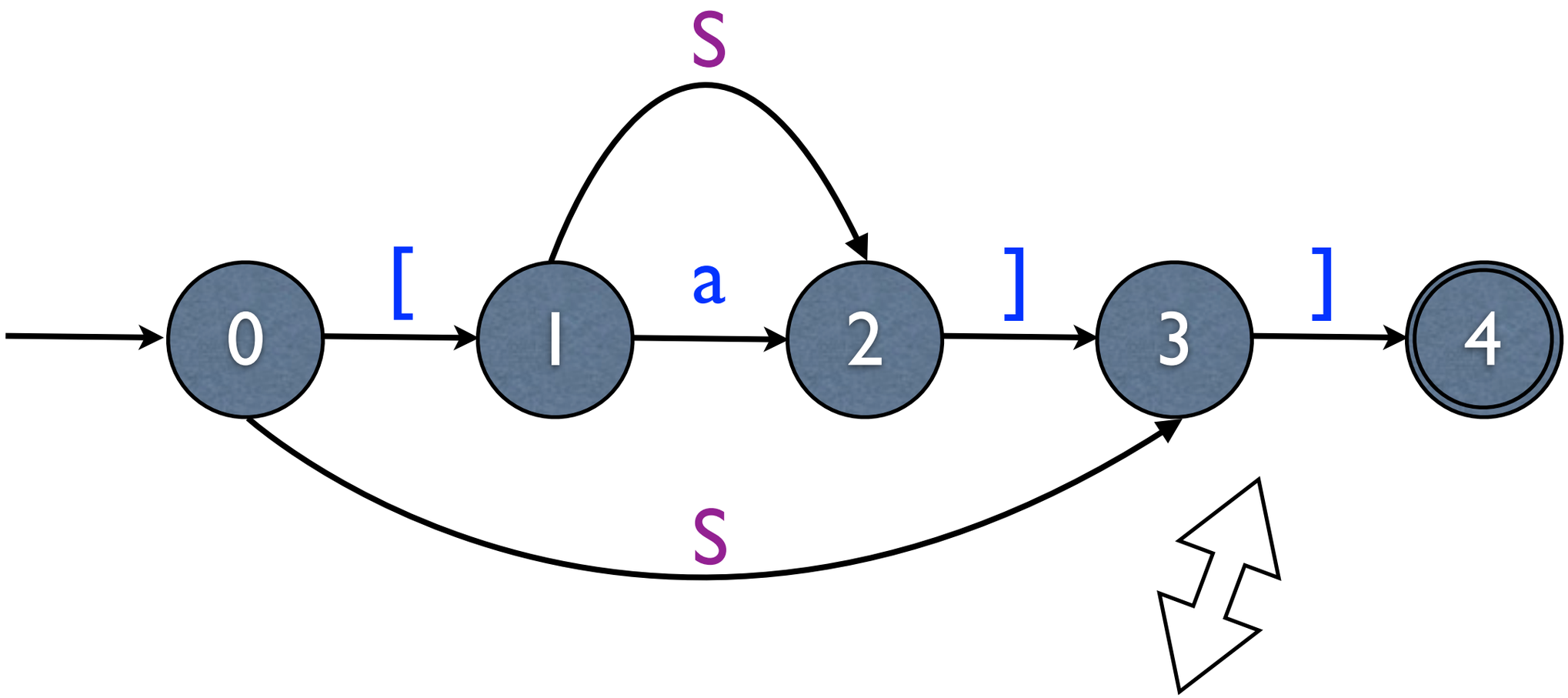
테이블로 분석값을?!

- 길이 n 문자열 요약시, 공간 $O(n^2)$, 시간 $O(n^3)$
- 분석값 사이의 연산 필요
  - 파싱 테이블 사이의 요약 접합, ⊔, ▽, ⊑ ?

내려다 보기

- 한 차원 높은 자료 타입 / 알고리즘
- 문자열 ➡ 선형 오토마타
- CYK 파싱 ➡ CFL 도달 알고리즘

앞의 예제



α({ [a]] })
= { [a]], [S]], S] }

연산

분석값	⊔	⊓	⊑	⦿	▽
오토마타	∩	∪	⊇	접합 & CFL 도달	한쪽의 self cycle 삭제후 ∩ ➡ CFL 도달

구현

학부생 3명과 함께

- SAFE의 문자열 요약 공간 ➡ StringAutomata로 plugin
- 1차 검증 대상 : e.innerHTML = s
- 안정화 / 성능 올리기 진행 중

특징 1

- 기존 SAFE 분석의 +α
  - 문맥 구분, 객체 분석, 루프 풀기 등등등등 이용 가능

```
for (var i = 0; i < option.text.length; i++) {
    button_tag += '<div class="sec-ui-button-c">' + option.text[i] + '</div>';
}
button_tag += '<div class="sec-ui-button-r"></div>';

elem.innerHTML = button_tag;
```

특징 2

- 참조 문법을 확장 문맥 자유 문법 이용

```
DOCUMENT
DOCUMENT => `($PCDATA$)?($DOCTYPE$($PCDATA$)?)?($OHTML$($PCDATA$)?)?($HEAD$)?($BODY$)?($CHTML$($PCDATA
DOCTYPE => `\<!doctype( html)?( public)?$STRING$\>`
OHTML => `\<html\>`
HEAD => `($OHEAD$($PCDATA$)?)?$HELEM$($PCDATA$|$HELEM$)*($CHEAD$($PCDATA$)?)?`
OHEAD => `\<head\>`
HELEM => `$TITLE$|$SCRIPT$|$STYLE$|$ISINDEX$|$BASE$|$META$|$LINK$`
TITLE => `$OTITLE$($PCDATA$)?$CTITLE$`
OTITLE => `\<title\>`
CTITLE => `\</title\>`
SCRIPT => `$OSCRIPT$(~$CSCRIPT$)+$CSCRIPT$`
OSCRIPT => `\<script\>`
CSCRIPT => `\</script\>`
STYLE => `$OSTYLE$(~$CSTYLE$)+$CSTYLE$`
OSTYLE => `\<style($WS$$ATTR$*)?\>`
CSTYLE => `\</style\>`
ISINDEX => `\<isindex$WS$$ATTR$\>`
BASE => `\<base$WS$$ATTR$\>`
META => `\<meta$WS$$ATTR$+\>`
LINK => `\<link$WS$$ATTR$+\>`
CHEAD => `\</head\>`
BODY => `($OBODY$($PCDATA$)*)?$BCNP$$BC$+($CBODY$($PCDATA$)*)?`
OBODY => `\<body($WS$($ATTR$)*)?\>`
BCNP => `$BODYTAG$|$TEXTTAG$`
BC => `$BODYTAG$|$TEXT$`
TEXT => `$PCDATA$|$TEXTTAG$`
```

문자 수준 ➡잘린 token도 ok

?, *, + 등 정규식 표현

+ α ?

모양 / 의미 분석

어떻게?

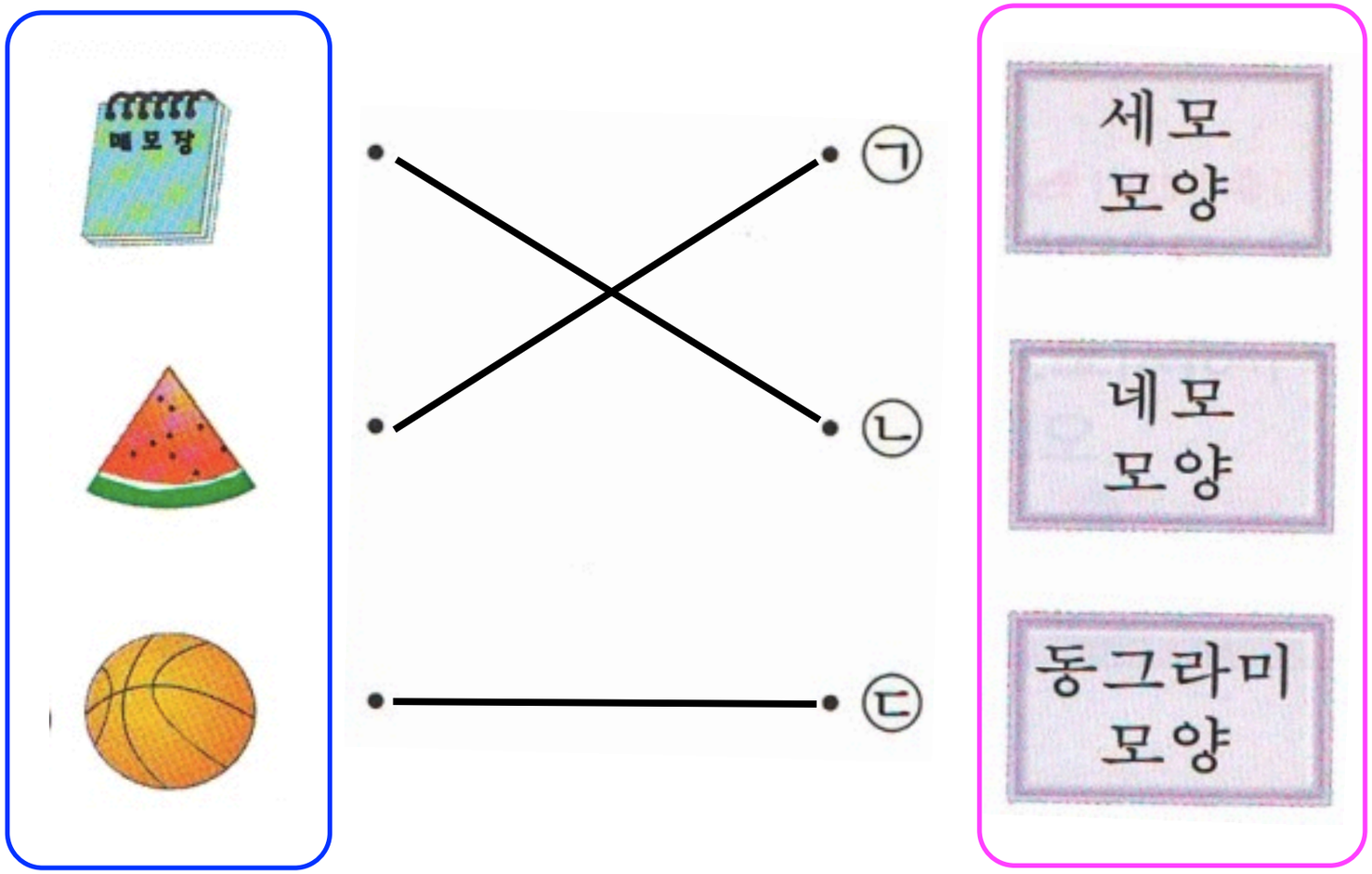
형식 개념 분석 + 속성 문법

↓

모양 / 의미 분석을 위한 Galois 연결

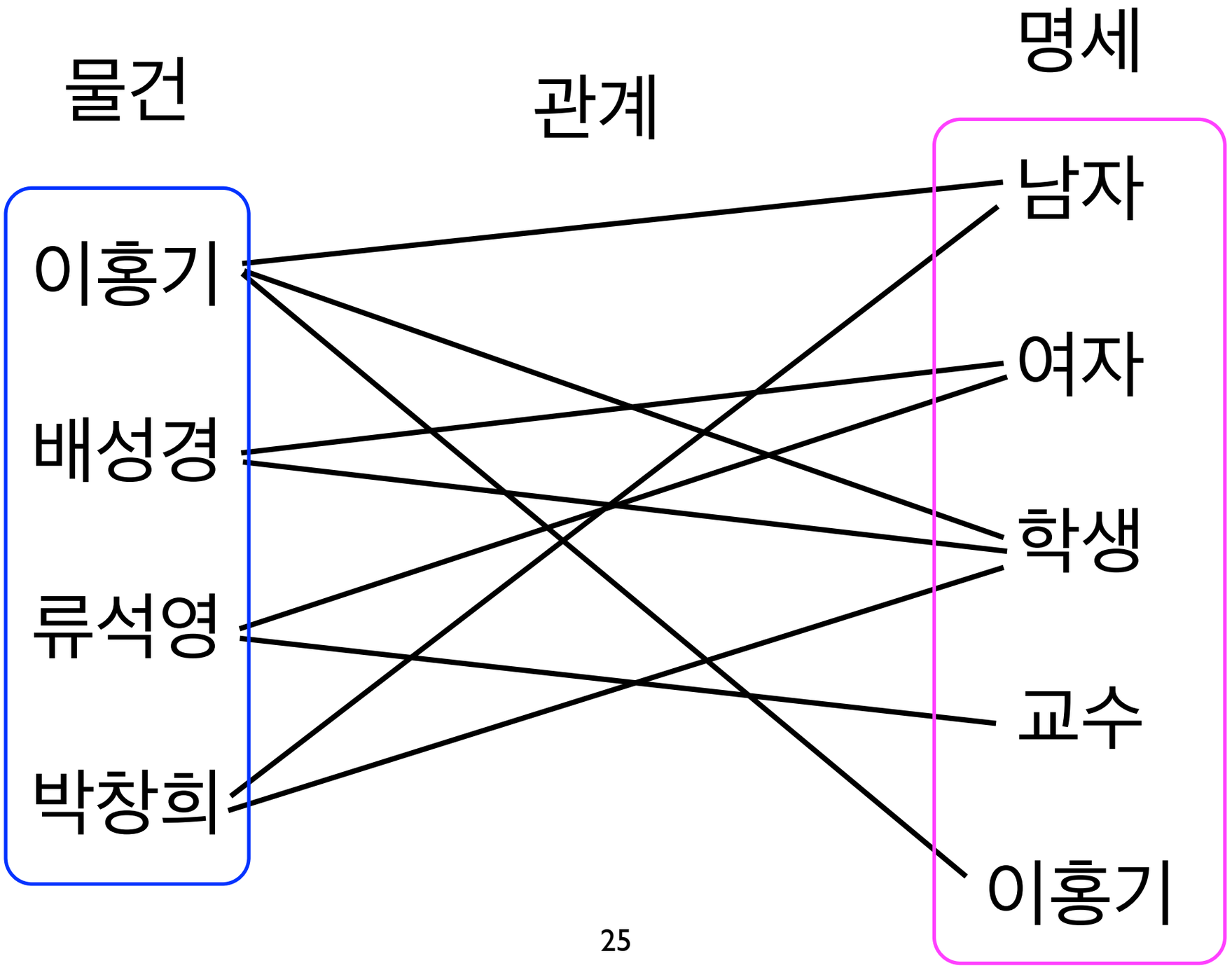
형식 개념 분석?

- 초등학교 1학년 수학 문제



물건 관계 명세

여럿-여럿도 가능



강력한 도구

- 물건과 명세 사이의 관계로부터,
- 물건 집합(⊆)과 명세 집합(⊇) 사이의 Galois 연결 도출

부합 분석

물건: 문자열, 명세: 어휘문자열, 관계: 유도관계

모양 / 의미 분석

- 물건: 문자열
- 명세: 속성 달린 어휘문자열
- 관계: 속성 할당 패턴 고려한 유도관계

속성의 선택

- tree automata ➡ 모양 분석
- (타입 환경, 타입) ➡ 타입 안전성 분석
- 정의되는 변수 / 자유 변수 ➡ 부수 효과 분석
- ……

결론

- 부합 분석값의 자연스러운 표현법/계산법 발견
  - SAFE에 plugin, 확장 문맥자유 문법 지원
- 부합 분석에서 [문맥자유 문법 ➡ 속성 문법]
  - 모양 / 의미 분석 가능, 연구 진행

감사합니다