Time-Aware Logical Pattern Mining

for Temporal Knowledge Graph Reasoning

시간인지 논리 패턴 마이닝을 통한 시간 지식그래프 추론

2026.07.15.

발표자 : 이동천

Graph & Language Intelligence Laboratory
Department of Computer Science and Engineering
Konkuk University

CONTENTS

Background

Temporal Knowledge Graph (TKG)

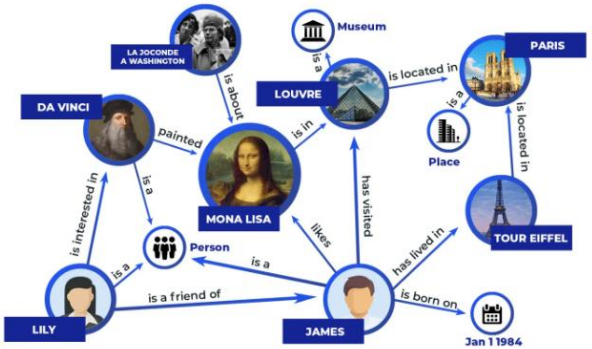
- Knowledge Graph (Static)
  - ✓ Entity 간 관계를 Triple 형태로 표현

$$\mathcal{G} = \{(s, r, o) \mid s, o \in \mathcal{E}, r \in \mathcal{R}\}$$

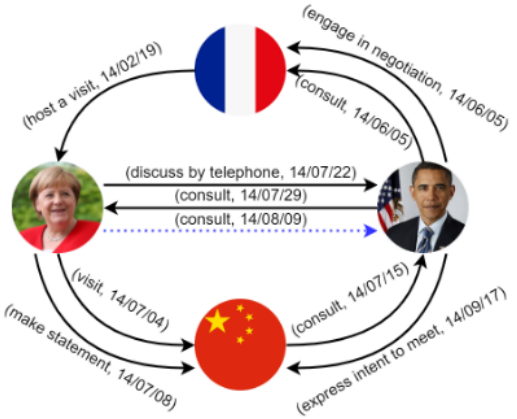
- Temporal Knowledge Graph
  - ✓ Timestamp 또는 Time Interval을 추가하여 동적인 사실을 표현 같은 (s, r, o)라도 시간에 따라 다름

$$\mathcal{G} = \{(s, r, o, \mathrm{t}) \mid s, o \in \mathcal{E}, r \in \mathcal{R}, t \in \mathcal{T}\}$$

$\mathcal{T}$: Timestamp or Time Interval



KG



TKG

출처 : https://yashuseth.wordpress.com/2019/10/08/introduction-question-answering-knowledge-graphs-kgqa/

TLogic: Temporal Logical Rules for Explainable Link Forecasting on Temporal Knowledge Graphs, AAAI 2022

Background

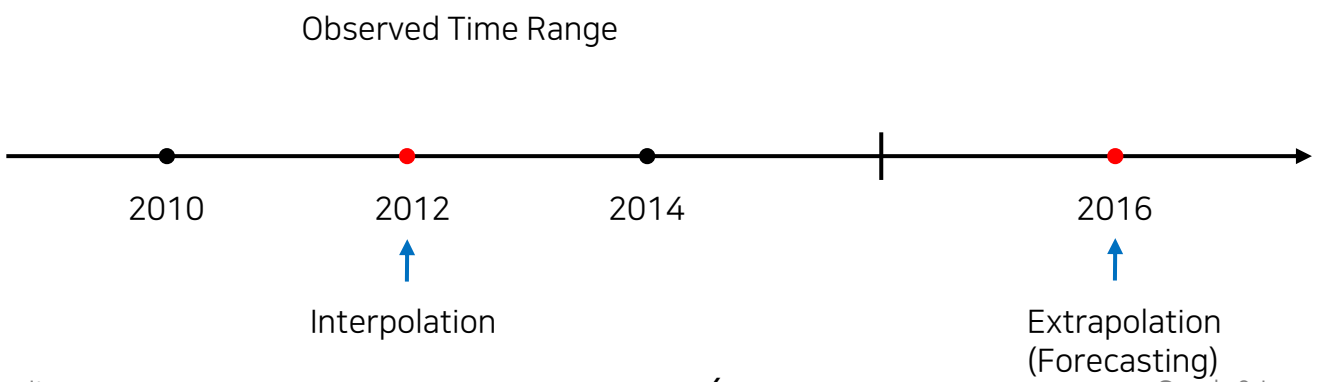
Temporal Knowledge Graph Reasoning

- 관측된 과거의 Temporal Fact를 기반으로 누락되거나 미래의 Fact를 추론
  - ✓ Interpolation Reasoning (보간)
    - 관측된 시간 범위 내에서 누락된 Fact를 추론

$$(s, r, ?, t), \qquad (?, r, o, t)$$

  - ✓ Extrapolation Reasoning (외삽)
    - 과거의 Temporal Fact를 기반으로 미래 Fact를 예측

$$(s, r, ?, t_{future}), \qquad (?, r, o, t_{future})$$



Background

Temporal Pattern in TKG

Temporal Order

사건의 발생 순서

$$t_1 < t_2$$

Recurrence / Periodicity

반복적 · 주기적으로 발생하는 관계

$$(s, r, o, t) \rightarrow (s, r, o, t + \Delta)$$

과거에 일어난 사건이 다시 발생

Temporal Interval

사건(관계) 간 시간 간격

$$\Delta t = t_2 - t_1$$

Duration

Fact가 유효한 시간 범위(구간)

$$[t_s, t_e]$$

언제부터 언제까지 참인지

Background

Existing Approaches for Temporal KG Reasoning

- Neural/Embedding-based Method
  - ✓ Entity, Relation, Time을 연속 벡터 공간에서 표현

$$Score(s,r,o,t) = f(s,r,o,t)$$

  - ✓ 장점
    - 효율적이고 확장 가능
    - 대규모 그래프에 적용 용이
  - ✓ 단점
    - 복잡한 구조적 의존성을 포착하기 어려움
    - 해석 가능한 추론이 불가능
  - ✓ Ex)
    - TTransE
    - TNTComplEx
    - ChronoR

Background

Existing Approaches for Temporal KG Reasoning

- Rule-Based Method
  - ✓ 설명 가능성이 부족한 기존 방법들과 달리, 사람이 이해할 수 있는 규칙을 사용
    - 사람이 직접 해석할 수 있기 때문에, 모델의 예측 결과에 대해 설명 가능한 근거 제공
  - ✓ Query entity에서 예측 Target entity까지 이어지는 Path를 시간 정보와 함께 표현

$$(\mathrm{E}_1, r_3, \mathrm{E}_3, \mathrm{T}_3) \leftarrow (\mathrm{E}_1, r_1, \mathrm{E}_2, \mathrm{T}_1) \wedge (\mathrm{E}_2, r_2, \mathrm{E}_3, \mathrm{T}_2)$$

Rule Head　　Rule Body

  - ✓ 장점
    - 예측 근거를 명시적으로 제공 (설명 가능성)
  - ✓ 단점
    - 규칙 템플릿 밖의 의존성은 표현 불가능
    - 규칙 탐색 공간이 폭발적으로 증가
  - ✓ Ex)
    - TLogic
    - TR-Rules
    - TILP

CONTENTS

Temporal Logical Rule

Static Rule에서 Temporal Rule

- Static Logical Rule(AMIE, RuleN)
  - ✓ Ex) $\text{ally}(A,B) \wedge \text{hostile}(B,C) \rightarrow \text{hostile}(A,C)$
    - 관계의 구조적 연결만 표현
    - 사건의 발생 시점과 순서를 구분하지 않음

- TKG에서는 어떤 게 필요한가?
  - ✓ Body의 사건들이 Head보다 과거여야 함
    → Temporal Constraint
  - ✓ 동일한 근거라도 5년 전과 한달 전은 신뢰도가 달라야 함
    → Time Decay
  - ✓ 시대가 바뀌면 Rule이 무효화될 수 있음
    → Confidence Dynamics

→ 이러한 고려 사항을 Rule에 반영하는 것이 "Time-Aware"

Temporal Logical Rule

Temporal Logical Rule 정의

- Logical Rule에 Timestamp와 Temporal Constraint를 추가
  - ✓ Relation Path뿐 만 아니라 Event 간 Temporal Order를 함께 표현

- Definition

Rule Head　　Rule Body

$$r_h(X_1, X_{l+1}, T_q) \leftarrow \wedge_{i=1}^{l} r_i(X_i, X_{i+1}, T_i)$$

$$with \quad T_1 \leq T_2 \leq \ldots \leq T_l < T_q$$

  - ✓ E, T = entity, timestamp
  - ✓ Rule body : 과거에 관측된 근거 사건
  - ✓ Rule head : query 시점에서 예측하려는 사건
  - ✓ Ex) $Sanction(X, Y, T_3) \leftarrow Threaten(X, Y, T_1) \wedge Protest(Y, X, \mathrm{T}_2)$

Temporal Logical Rule

Temporal Logical Rule의 형태와 길이

- Rule의 형태

Cyclic Rule (순환)

Body의 시작 entity와 끝 entity가 Head와 동일 → 닫힌 경로

$r_h(X, Y, T_h) \leftarrow r_1(X, Z, T_1) \wedge r_2(Z, Y, T_2)$

Acyclic Rule (비순환) 인과 관계

Body가 Head의 Entity 쌍을 닫지 못함 → 열린 경로

$r_h(X, e_h, T_h) \leftarrow r_b(X, e_b, T_b)$

$\text{Job}(X, \text{Teacher}, T_h) \leftarrow \text{Major}(X, \text{Education}, T_b)$

- Rule Length

$l = 1$
Single Body Atom

+ 해석 쉬움 · Noise 적음
- 표현력이 낮음

$l = 2 \sim 3$
Multi-hop

+ 표현력 ↔ 비용 균형
- 탐색 공간 증가

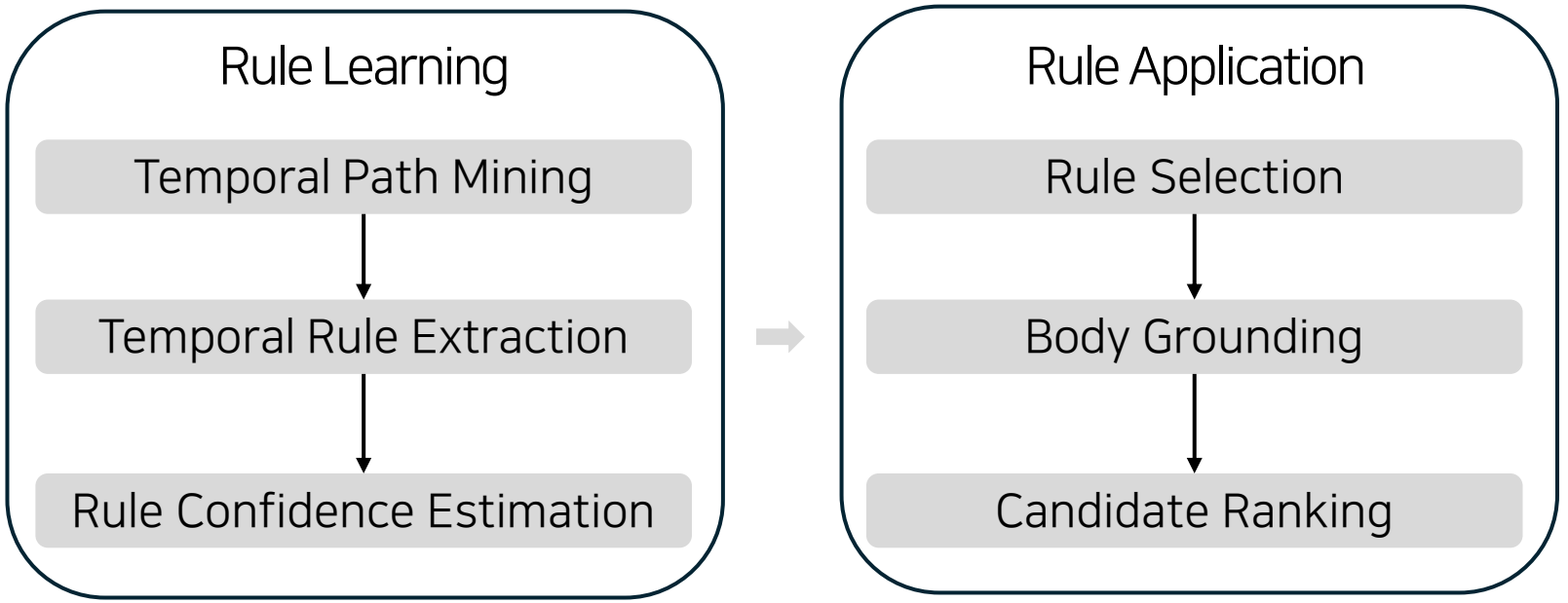
$l \geq 4$
Long Rule

+ 복잡한 multi-hop 시간적 의존 패턴 포착
- 조합 폭발 · Noise · 해석성 저하

Temporal Logical Rule

Temporal Rule Learning & Application

- TKG에서 반복되는 Temporal Path를 탐색하고 Rule로 일반화
- 학습된 Rule을 Query에 적용하여 Candidate Entity를 예측



Temporal Pattern을 명시적인 Rule로 추출하고, Historical Fact에 적용하여 예측

CONTENTS

관련 논문

Roadmap

Rule 탐색 · 생성	Confidence 추정	Rule 적용
TLogic (AAAI' 22)	TR-Rules (EMNLP' 23)	
LLM-DA (NIPS' 24)	TILP (ICLR' 23)	
TRKG-Miner (PAKDD' 23)	CountTRuCoLa (ICLR' 26)	

관련 논문

[2022][AAAI][TLogic]

- 기존 한계점
  - ✓ Embedding-based TKG reasoning은 예측 과정에 대한 설명 가능성이 부족
    - Temporal Pattern을 Latent representation으로 학습
    - 예측 결과에 대한 명시적인 Reasoning Chain 제공 어려움
  - ✓ Symbolic Rule Learning의 확장성 문제 및 수작업 Rule 정의 어려움

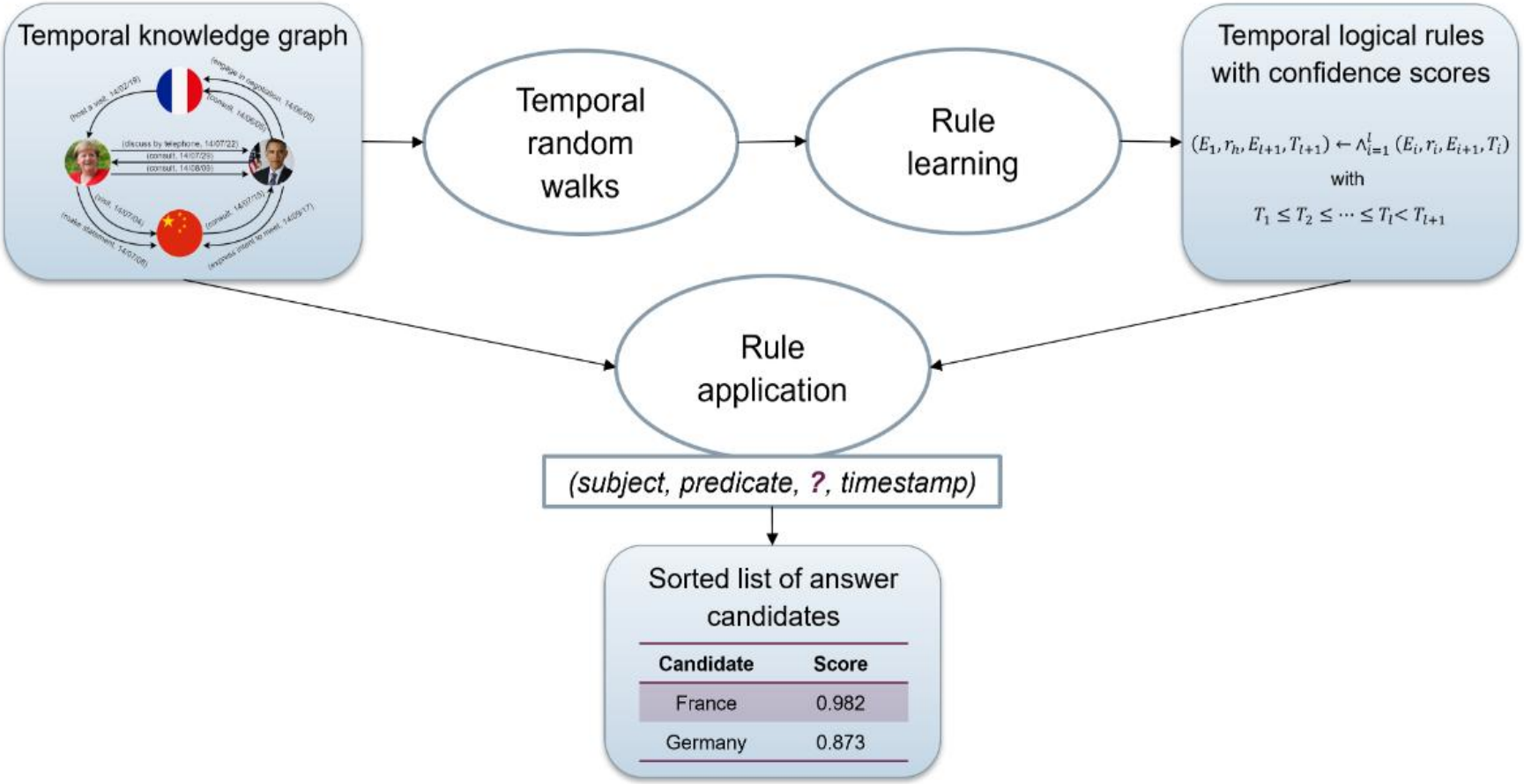
- 제안 방법
  - ✓ Temporal Random Walk 기반 Temporal Logical Rule Mining
    - 과거 방향의 Temporal Random Walk를 통해 Time-consistent Path 탐색
    - Cyclic Temporal Walk를 Temporal Logical Rule로 일반화
    - Rule Support / Body Support 기반 Rule Confidence 계산
    - Count-based Rule Confidence 계산 및 Link Forecasting

TLogic은 시간 순서를 만족하는 cyclic walk를 rule로 변환

TLogic: Temporal Logical Rules for Explainable Link Forecasting on Temporal Knowledge Graphs, AAAI 2022

관련 논문

[2022][AAAI][TLogic]



TLogic: Temporal Logical Rules for Explainable Link Forecasting on Temporal Knowledge Graphs, AAAI 2022

관련 논문

[2022][AAAI][TLogic]

- Temporal Random Walk

Inverse Relation 추가로 양방향 탐색 허용

  - ✓ Target relation을 가진 edge에서 출발하여 시간을 거슬러 올라가며 Walk 수행
  - ✓ 다음 edge 선택 시, 시간적으로 가까운 edge에 지수적으로 큰 확률 부여

$$\mathbb{P}(\mathrm{u};\ \mathrm{m}, \mathrm{e}_{\hat{\mathrm{m}}}, \mathrm{t}_{\hat{\mathrm{m}}}) = \frac{exp(t_u - t_{\hat{m}})}{\sum_{\hat{u} \in \mathcal{A}(m, e_{\hat{m}}, t_{\hat{m}})} exp(t_{\hat{u}} - t_{\hat{m}})}$$

- Rule Generation
  - ✓ 샘플링된 Walk의 구체적인 entity를 변수로 치환 → Rule 후보 획득

Walk : $(\mathrm{US}, \mathrm{Threaten}, \mathrm{Iran}, T_1) \rightarrow (\mathrm{Iran}, \mathrm{Protest}, \mathrm{US}, T_2) \rightarrow (\mathrm{US}, \mathrm{Sanction}, \mathrm{Iran}, T_3)$

Rule : $(E_1,\ \mathrm{Sanction}, E_2,\ T_3) \leftarrow (E_1,\ \mathrm{Threaten}, E_2, T_1) \wedge (E_2, \mathrm{Protest}, E_1, T_2)$

관련 논문

[2022][AAAI][TLogic]

- Count-based Rule Confidence
  - ✓ Body Support : 시간 제약을 만족하는 rule body grounding의 수
  - ✓ Rule Support : 해당 grounding 이후에 rule head가 실제로 발생한 grounding의 수

$$\mathrm{Conf(R)} = \mathrm{Rule\ Support} \,/\, \mathrm{Body\ Support}$$

- Application

$$\mathbf{F}(R, \mathbf{c}) = \alpha \cdot conf(R) + (1-\alpha) \cdot exp\left(-\lambda \cdot \left(t_q - t_l(B(R,c))\right)\right)$$

  - ✓ λ 가 클수록 오래된 grounding의 점수를 더 빠르게 감소시킴
  - ✓ $t_1\big(B(R,c)\big)$: body groundin의 가장 빠른 timestamp

TLogic: Temporal Logical Rules for Explainable Link Forecasting on Temporal Knowledge Graphs, AAAI 2022

관련 논문

[2023][PAKDD][TRKG-Miner]

- 기존 한계점
  - ✓ TLogic은 Cyclic Temporal Rule만 Mining
    - Walk가 시작 entity로 되돌아와야 하는 닫힘 조건이므로, Cyclic closure 조건을 만족하지 못한 sampled walk는 rule로 변환되지 못함
      - Sparse 영역의 정보 포착 불가
  - ✓ 모든 Event에 Strict Temporal Order 강제
    - 시간 간격이 짧아 순서가 무의미한 Event 쌍도 Walk에서 배제
      - 유용한 Rule 다수 손실

- 제안 방법
  - ✓ Temporal Relaxation 기반 Rule Mining
  - ✓ Link-star Rule 추가
    - Acyclic Rule의 특수 클래스
  - ✓ Acyclic to Cyclic Ratio로 두 Rule 유형의 생성 비율 제어

관련 논문

[2023][PAKDD][TRKG-Miner]

- Relaxed Temporal Random Walk
  - ✓ 연속 link 간 시간 조건을 $t_i \geq t_{i-1} - \delta$로 완화
    - 다음 edge가 이전 edge보다 최대 δ만큼 이른 경우까지 temporal walk로 허용
  - ✓ 현재 timestamp와 가까운 candidate edge에 지수적으로 높은 sampling probability 부여

$$\mathbb{P}(t_c; T_c, t_u) = \frac{\exp(-|t_u - t_c|)}{\sum_{t_{c_i} \in T_c} \exp(-|t_u - t_{c_i}|)}$$

- Rule Generation
  - ✓ Cyclic Rule
    - Head relation을 마지막 link로 샘플링한 뒤 역방향 walk 수행
    - 시작·종료 entity가 동일한 cyclic walk 생성
    - 마지막 link의 inverse relation을 rule head로 설정
  - ✓ Link-star Rule
    - 길이 3의 acyclic walk에서 가운데 link를 head로 먼저 샘플링
    - 가운데 link 양옆의 두 link를 rule body로 구성
    - Cycle closure 조건이 없어 거의 모든 sampled acyclic walk를 rule로 활용

- Cyclic Rule Generation

$$Robin \underset{T_1}{\overset{\text{visit}}{\rightarrow}} Spain \underset{T_2}{\overset{\text{talk}}{\rightarrow}} France \underset{T_3}{\overset{\text{visit}^{-1}}{\rightarrow}} Robin$$

  - ✓ 마지막 edge를 Rule Head로 변환하여 Rule Head로 사용
    - $(\text{France}, \text{visit}^{-1}, \text{Robin}, T_3) = (\text{Rovin}, \text{visit}, \text{France}, T_3)$
  - ✓ 나머지 edge를 Rule body로 사용
    - $(E_1, visit, E_3, T_3) \leftarrow (E_1, visit, E_2, T_1) \wedge (E_2, talk, E_3, T_2)$

- Link-star Rule Generation
  - ✓ *길이 3의 Acyclic Walk 탐색*

$$\underset{T_0}{\overset{\text{visit}^{-1}}{\text{Belgium} \rightarrow}} \text{Fred} \underset{T_1}{\overset{\text{talk}}{\rightarrow}} \underset{T_2}{\overset{\text{visit}}{\text{Robin} \rightarrow}} \text{Spain}$$

  - ✓ 가운데 *edge*를 *Rule head*로 선택
    - 가운데 edge인 $(Fred, talk, Robin, T_1)$ 을 예측 대상으로 사용하고, 양옆 edge를 근거로 사용
  - ✓ Rule로 일반화

$$(E_1, talk, E_2, T_1) \leftarrow (E_0, visit^{-1}, E_1, T_0) \wedge (E_2, visit, E_3, T_2)$$

관련 논문

[2023][ICLR][TILP] Time Interval 기반

- 기존 한계점
  - ✓ 통계적 Rule-based 방법은 count-based rule confidence 계산
    - 각 Rule을 독립적으로 평가 → Rule 간 Interaction을 반영하기 어려움
    - 구조적으로 유사한 Rule이라도 희소하면 낮은 confidence를 받음

- 제안 방법
  - ✓ Temporal Logical Rule을 End-to-End로 학습하는 Differentiable Framework
    - Constrained random walk와 Temporal Operator를 이용
    - RNN 기반 Attention으로 Predicate/temporal 제약의 중요도 학습
    - Temporal feature를 결합하여 후보 점수 계산

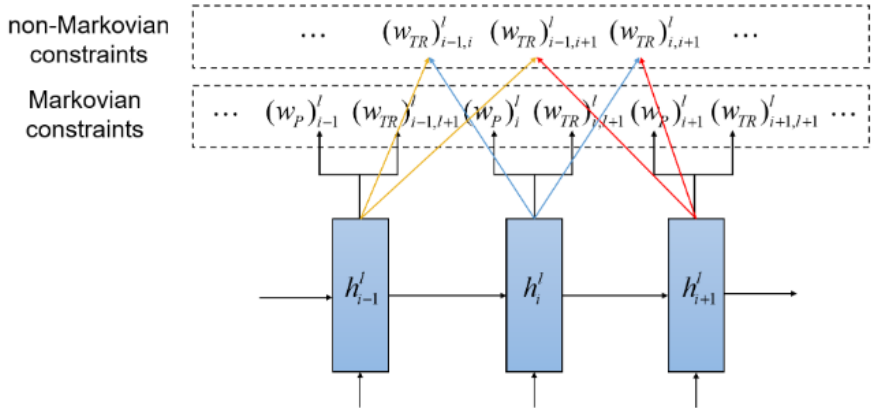
TILP는 시간 제약을 포함한 논리 규칙을 미분가능하게 학습

TILP: Differentiable Learning of Temporal Logical Rules on Knowledge Graphs, ICLR 2023

관련 논문

[2023][ICLR][TILP]

- Temporal Rule Extraction
  - ✓ Constrained Random walk 기반 Temporal Path 탐색
    - Markovian Constraint 하에 Random walk 수행 후, Non-Markovian Constraint로 필터링
    - Path를 Predicate Sequence + Temporal Relation 형태의 Rule로 변환



Markovian : 다음 상태가 오직 현재 상태에만 의존

Non-Markovian : 다음 상태가 이전 상태들의 정보에도 의존

TILP: Differentiable Learning of Temporal Logical Rules on Knowledge Graphs, ICLR 2023

관련 논문

[2023][ICLR][TILP]

- Differentiable Rule Learning
  - ✓ Rule 구성 요소의 중요도(Attention)를 학습하여 각 규칙의 신뢰도 계산
    - Predicate 및 Temporal relation에 대한 attention vector를 RNN 기반으로 생성
      - 이를 통해 미분 가능하게 계산
    - Rule frequency를 단순 counting하지 않고 differentiable하게 rule confidence 학습

- Temporal Feature Modeling
  - ✓ TKG의 시간적 특성을 명시적으로 모델링하여 rule evidence를 보완
    - Recurrence
    - Temporal Order
    - Relation Interval
    - Duration

각 특성은 확률 분포로 모델링 한 후 Score와 가중합

TILP: Differentiable Learning of Temporal Logical Rules on Knowledge Graphs, ICLR 2023

관련 논문

[2023][EMNLP-F][TR-Rules]

- 기존 한계점
  - ✓ TLogic은 Cyclic Temporal Rule 중심의 Rule Mining
    - 다양한 Temporal Dependency 표현에 한계
    - Count-based Confidence는 Temporal Redundancy의 영향을 받을 수 있음
      - 동일한 Triple이 다른 timestamp와 함께 반복적으로 나타날 수 있음

- 제안 방법
  - ✓ Temporal Redundancy를 고려한 Rule-based Link Forecasting
    - Acyclic Temporal Rule을 추가하여 Rule Structure 확장
    - Timeline을 여러 Temporal Window로 분할
    - 각 Window 내 Valid Body Instance와 Head Event의 발생 여부를 기반으로 Window Confidence 계산

TR-Rules는 Cyclic/Acyclic Rule를 추출하고
Temporal Redundancy를 Window Confidence로 해결

TR-Rules: Rule-based Model for Link Forecasting on Temporal KnowledgeGraph Considering Temporal Redundancy, EMNLP-F 2023

관련 논문

[2023][EMNLP-F][TR-Rules]

- Temporal Rule Extraction
  - ✓ Cyclic Rule Mining
    - Constrained Random Walk 기반 Temporal Path 탐색
      - Rule head를 샘플링 한 뒤, Head의 객체에서 출발해 시간 역순으로 Random walk 수행
      - 지수 가중치 분포를 사용하여 가까운 timestamp를 가진 edge 선택하여 전이

$$(E_1, r_h, E_{n+1}, T_{n+1}) \leftarrow \bigwedge_{i=1}^{n} (E_i, r_{b_i}, E_{i+1}, T_i)$$

  - ✓ Acycilc Rule Mining (길이 1)
    - Rule head(Quadruple)를 무작위로 샘플링한 뒤, head entity에 인접한 edge 중 시간적으로 빠른 edge 선택

$$(E_a, r_h, e_h, T_h) \leftarrow (E_a, r_b, e_b, T_b) \quad (T_h > T_b)$$

TR-Rules: Rule-based Model for Link Forecasting on Temporal KnowledgeGraph Considering Temporal Redundancy, EMNLP-F 2023

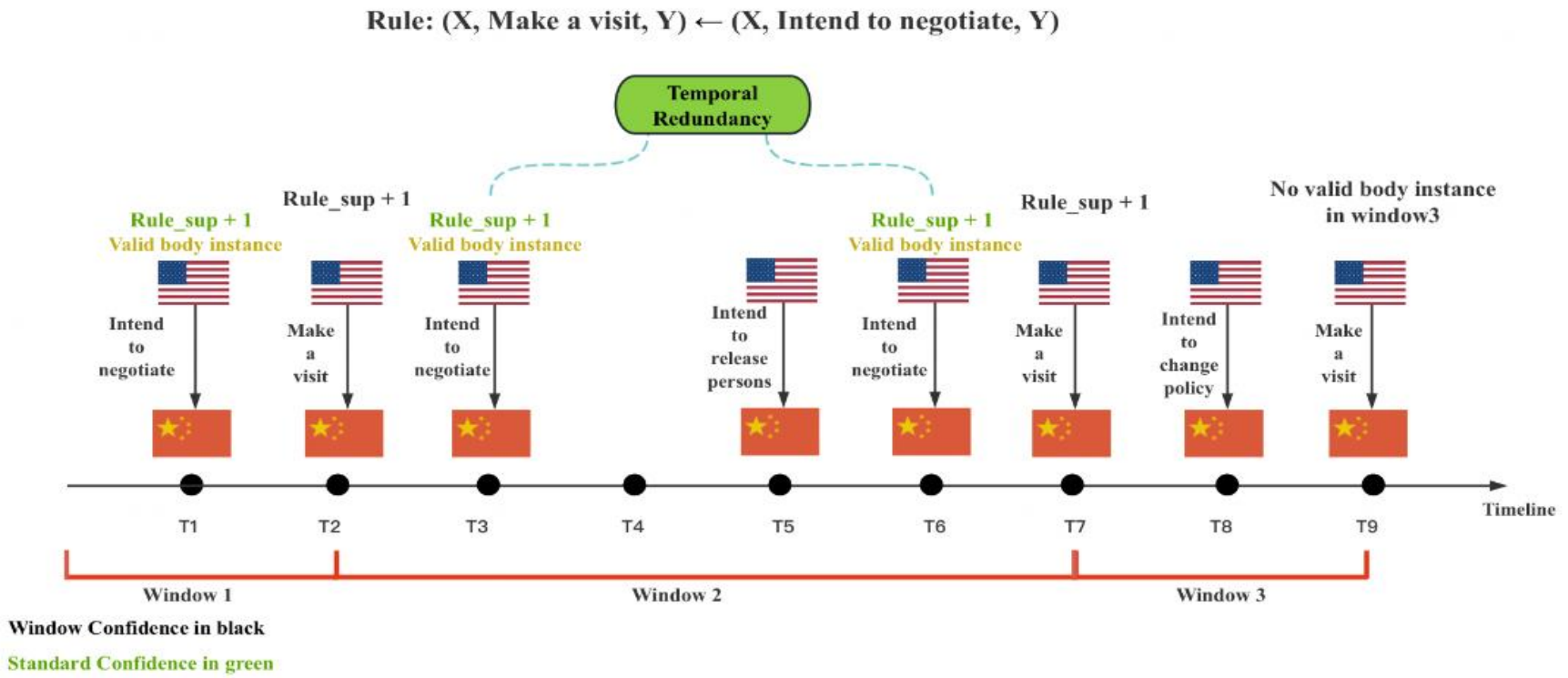
관련 논문

[2023][EMNLP-F][TR-Rules]

- Confidence Estimation – Window Confidence

Temporal Redundancy

  - ✓ 동일 fact가 여러 시점에서 반복되어 Counting으로 rule support가 과대평가됨
  - ✓ Rule의 모든 Head Timestamp를 정렬하여 Timeline을 window로 나눔
    - 각 window에 body instance가 하나라도 있으면 그 window에서 rule support를 1로 집계



TR-Rules: Rule-based Model for Link Forecasting on Temporal KnowledgeGraph Considering Temporal Redundancy, EMNLP-F 2023

관련 논문

[2024][NIPS][LLM-DA]

- 기존 한계점
  - ✓ 해석 가능성과 Dynamic Adaptation 한계
    - Deep learning 기반 방법들은 예측 과정에 대한 해석이 어려움
    - Rule-based 방법은 Temporal Pattern을 효과적으로 Rule로 추출 / 갱신하기 어려움
    - TKG가 시간에 따라 변화하여 Temporal Distribution Shift 발생

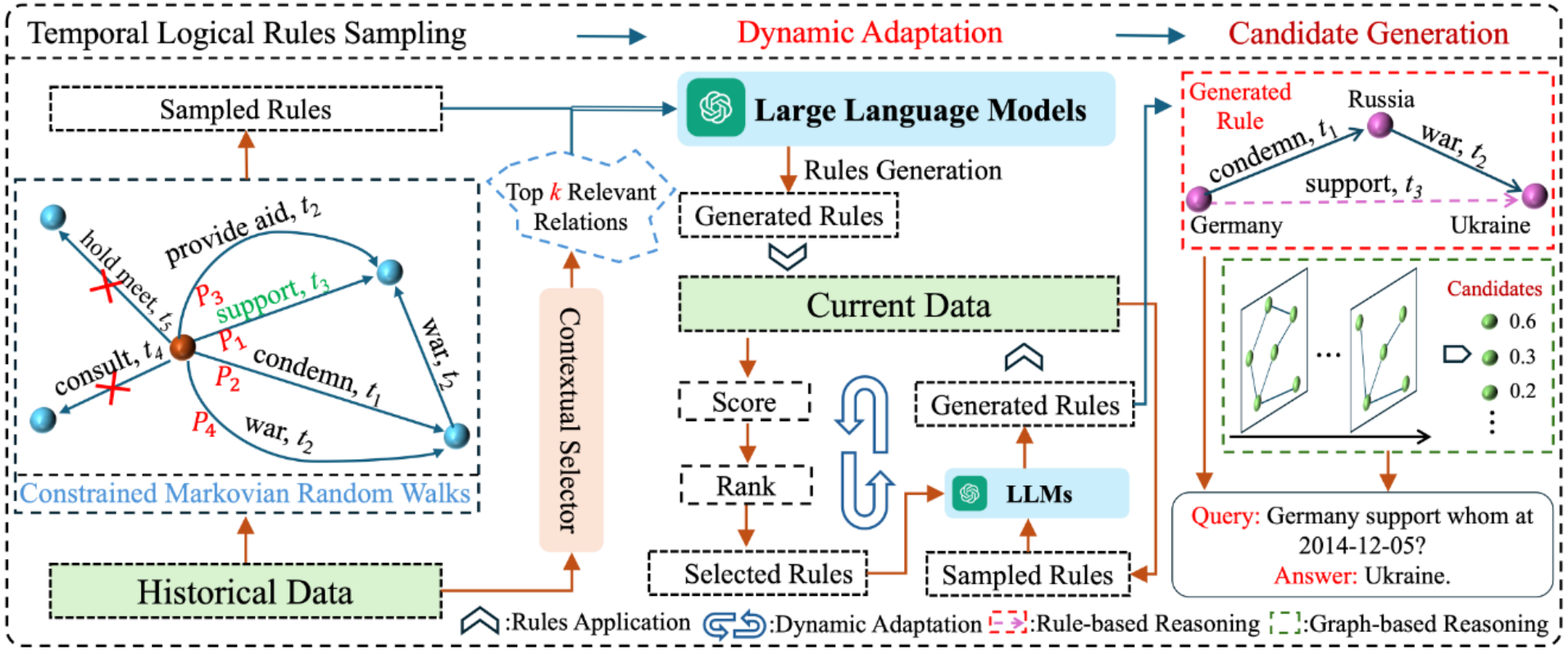
- 제안 방법
  - ✓ LLM 기반 Temporal Rule Generation 및 Dynamic Adaptation Framework
    - Historical Data에서 Temporal Logical Rule Sampling
    - LLM을 활용하여 General Temporal Rule 생성
    - Current Data 기반으로 Low-quality Rule을 동적 갱신
    - Rule-based + Graph-based Reasoning으로 Candidate 예측

LLM-DA는 LLM을 통해 Rule 생성 및 업데이트

관련 논문

[2024][NIPS][LLM-DA]

LLM 자체를 fine-tuning 하는 것이 아닌 Rule 업데이트



관련 논문

[2024][NIPS][LLM-DA]

- Temporal Rule Sampling
  - ✓ Historical data에서 Temporal Path와 초기 규칙을 추출
- LLM-based Rule Generation
  - ✓ Relation 의미와 주변 context를 이용하여 일반화된 Temporal Rule 생성
- Dynamic Adaptation
  - ✓ Current data를 이용하여 품질이 낮은 Rule을 반복적으로 업데이트
- Candidate Generation
  - ✓ Rule-based reasoning
  - ✓ Graph-based reasoning
  - ✓ 두 결과를 결합하여 후보 entity 예측

관련 논문

[2026][ICLR_Withdrawn][CountTRuCoLa]

- 기존 한계점
  - ✓ Temporal Rule 기반 방법은 복잡한 Multi-hop Rule Structure에 집중
    - 긴 Rule은 표현력을 높이지만 Noise 증가 및 해석 가능성 저하
    - 기존 Temporal confidence는 최근 발생 여부뿐 아니라 반복 빈도를 충분히 함께 반영하지 못함

- 제안 방법
  - ✓ Recency와 Frequency를 고려한 Temporal Rule Confidence Learning
    - Single Body Atom 기반 4가지 single rule type 학습
    - 가장 최근 Rule Body 발생 시점과 Query 간 Temporal Distance 고려
    - 일정 시간 범위 내 Body의 Occurrence Frequency 고려
    - Rule 별 Parameterized Confidence Function 학습

CountTRuCoLa는 과거 사실의 최근성과 반복 빈도를 활용해 미래 관계를 예측

관련 논문

[2026][ICLR_Withdrawn][CountTRuCoLa]

- 4가지 Rule Type 제안
  1. xy-rule

$$h(x, y, t^*) \leftarrow b(x, y, t), \qquad t^* > t$$

  - 동일한 entity pair 사이의 관계 변화 학습
  - 과거 relatio이 미래 relaton으로 이어지는 패턴 포착
  - 예시)

$$Consult(x, y, t^*) \leftarrow ExpressIntentToMeet(x, y, t)$$

  2. Recurrent xy-rule
  - 과거에 등장한 fact가 미래에 다시 반복되는 패턴 학습
  - 기존 Recurrency Baseline을 포함하는 가장 단순한 temporal rule

$$h(x, y, t^*) \leftarrow \mathrm{h}(x, y, t), \qquad t^* > t$$

  - 예시)

$$HostVisit(x, y, t^*) \leftarrow HostVisit(x, y, t)$$

관련 논문

[2026][ICLR_Withdrawn][CountTRuCoLa]

- 4가지 Rule Type 제안
  3. c-rule
     - 특정 constanct entity가 포함된 temporal rule 학습
     - 특정 지역, 조직, 국가 등 frequent entity와 연결된 패턴 포착

$$h(x, \mathrm{d}, t^*) \leftarrow b(x, \mathrm{d}', t) \wedge t^* > t$$

     - 예시)

$$studied(x, \mathrm{U}v\mathrm{A}, t^*) \leftarrow born(x, Amsterdam, t)$$

  4. z-rule / f-rule
     - Temporal transitio이 아닌 relation-level frequency priori을 학습
     - z-rule : 전체 entity 기준의 일반적인 objec분포
     - f-rule : 특정 subject 기준의 object 분포

$$p(x, d, t) \leftarrow \exists z\, p(x, z, t)$$
$$p(c, d, t) \leftarrow \exists z\, p(c, z, t)$$

관련 논문

[2026][ICLR_Withdrawn][CountTRuCoLa]

- Temporal Confidence Learning
  - ✓ Recency & Frequency 기반 Rule confidence
    - 각 xy-rule과 c-rule에 대해 rule-specific confidence function 학습
    - confidence는 두 요소의 합으로 계산

$$conf_r(\Delta) = f_r(\Delta) + g_r(\Delta)$$

  - Δ : body fact가 과거에 발생한 시점들과 예측 시점 사이의 시간 거리 집합

$$\Delta = \{t^* - t' \mid b(c, d, t') \in G\}$$

- Recency Function
  - ✓ 최근 window 안에서 body fact가 얼마나 자주 반복되었는지 반영

$$g_r(\Delta) = clip\left(\rho \frac{|\Delta_w|}{W} + \frac{\kappa_r}{min(\backslash \mathrm{Delat})}, -\gamma_r, \gamma_r\right)$$

CONTENTS

데이터셋

ICEWS (Integrated Crisis Early Warning System) – TimeStamp/Event

- 설명
  - ✓ 뉴스 기사에서 자동으로 추출한 정치·사회적 사건(event) 데이터를 구조화

> Quadruple 샘플
>
> (Bangladesh, Consult, Vietnam, 2018-06-01)
> (Angela Merkel, Consult, Francois Hollande, 2014-11-10)

- Statistic

Dataset	기간	Entity	Relation	Timestamp	Train / Valid / Test
ICEWS14	2014	7,128	230	365	63,685 / 13,823 / 13,222
ICEWS05-15	2005-2015	10,488	251	4,017	322,958 / 69,224 / 69,147
ICEWS18	2018	23,033	256	304	373,018 / 45,995 / 49,545

데이터셋

WIKIDATA/ YAGO – Time Interval

- 설명
  - ✓ Wikipedia 계열 지시 그래프에서 시간 정보가 포함된 Fact를 추출한 데이터셋
  - ✓ 하나의 Fact를 특정 시점이 아닌 유효한 시간 구간으로 표현
    - 소속/ 직책과 같은 일정 기간 유지되는 상태와 관계를 주로 표현

(Alice Bradley Sheldon, receiveAward, Nebula Award for Best Novelette, [1977, 1977])

- Statistic

Dataset	Entity	Relation	Time Point	Interval	Train / Valid / Test
WIKIDATA12k	12,544	24	237	2,564	32,497 / 4,062 / 4,062
YAGO11k	10,622	10	251	6,651	16,408 / 2,051 / 2,050

평가 지표

Evaluation Metric – Hit@K / MRR

- Ranking 문제로 평가
  - ✓ Query $(s, r, ?, t)$ → 후보 Entity에 점수를 매겨 정렬하여 순위
  - ✓ Time-aware Filtered: 동일 timestamp에 참인 다른 정답은 후보에서 제외
    - 같은 시점에 정답이 여러 개 존재 가능 → 맞는 예측이 감점되는 것을 방지

- Hit@K
  - ✓ 정답이 K개 안에 있는지
    - 순위와 상관 없이 동일

- MRR (Mean Reciprocal Rank)
  - ✓ 정답이 나온 순위의 역수를 평균 낸 값 정답이 몇 번째에 등장했는지
    - 정답이 1등이면 $\frac{1}{1} = 1$, 정답이 2등이면 $\frac{1}{2} = 0.5$ 정답에 가까울수록 성능이 좋음

$$\mathrm{MRR} = \frac{1}{|U|}\sum_{u=1}^{|U|}\frac{1}{k_u}$$

CONTENTS

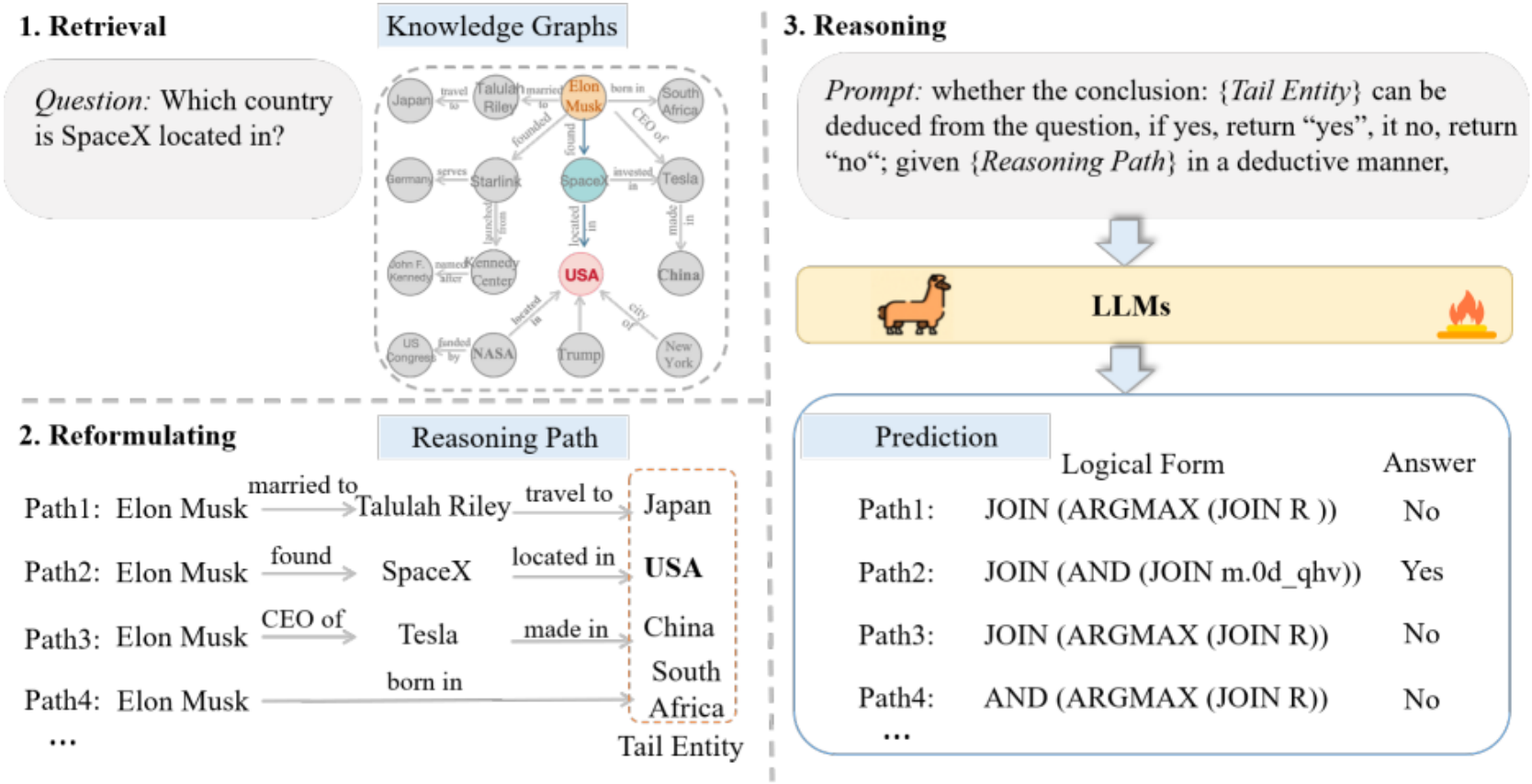
마무리

정리

Model	Rule	Rule 추출 방식	Rule Confidence	Contribution
TLogic (AAAI' 22)	Cyclic, $l \leq 3$	Temporal RW	Count + 고정 Decay	Search에 시간 정보를 넣음
TILP (ICLR' 23)	Temporal Operator	Constrained RW	Differentiable	Confidence를 학습으로 반영
TR-Rules (EMNLP' 23)	+ Acyclic Rule	RW	Window Confidence	Space 확장 + Redundancy 완화
LLM-DA (NIPS' 24)	Cyclic	LLM이 생성	Dynamic Update	LLM을 통해 검색 및 동적 업데이트
CountTRuCoLa (ICLR' 26)	$l = 1$	탐색 최소화	Recency + Frequency	Confidence가 중요함을 보임

연구 방향

[2026][ICLR][CoD]



CoD 모듈 별 설명 및 확장 시 발생하는 문제점

모듈 1 검색 (Retrieval)

- CoD
  - ✓ Topic entity로 부터 최대 hop 내 entity/relation을 통해 소규모 그래프 생성
  - ✓ 정답과 관련된 relation을 이진 분류 (정답과 관련된 relation=1, other = 0)
  - ✓ T5와 같은 경량 모델을 통해 학습

- Temporal CoD로 확장 시 문제점
  - ✓ TKG의 Quadruple에 대해서 Relation 단위 이진 분류가 맞지 않음 Temporal Information도 함께 고려해야 함
    - Ex. President_of 라는 relation은 질문과 관련이 있지만, 특정 시점의 Quadruple만 정답과 관련
  - ✓ Before/ After과 같은 시간 제약을 질문에서 해석하여 검색에 반영해야 함

⇒ Temporal Intent 추출하는 모듈 + Time-Aware Retriever 필요

CoD 모듈 별 설명 및 확장 시 발생하는 문제점

모듈 2 재구성 (Reformulation)

- CoD
  - ✓ 검색된 Triple set을 reasoning path 형태로 변환
    - Parameter-free, hand-craft rule을 통해 학습이 필요 없음
  - ✓ 알고리즘
    - i 번째 hop의 tail entity와 i + 1번째 hop의 Head entity인 연결성 원칙으로 Triple 연결
    - 연결된 Chain을 Path로 생성
  - ✓ Retriever가 잘 작동하면 최종 Path 수는 보통 10개 미만으로 나옴

- Temporal CoD로 확장 시 문제점
  - ✓ Path 조합 시 연결성 + 시간적 일관성으로 고려해야 함
    - Ex. 결혼 한 후에 쓴 책과 같은 질문은 결혼 < 저술 이라는 순서 제약을 만족하는 Path만 유효
  - ✓ 시간 제약을 규칙으로 처리하면, 시간 구간의 Allen's Interval algebra를 다루는 규칙 체계 필요하며 질문의 시간 의도(제약)을 파싱
  - ✓ 같은 entity 쌍에 시점만 다른 Quadruple이 여러 개 존재하기 때문에 후보 Path 폭발적

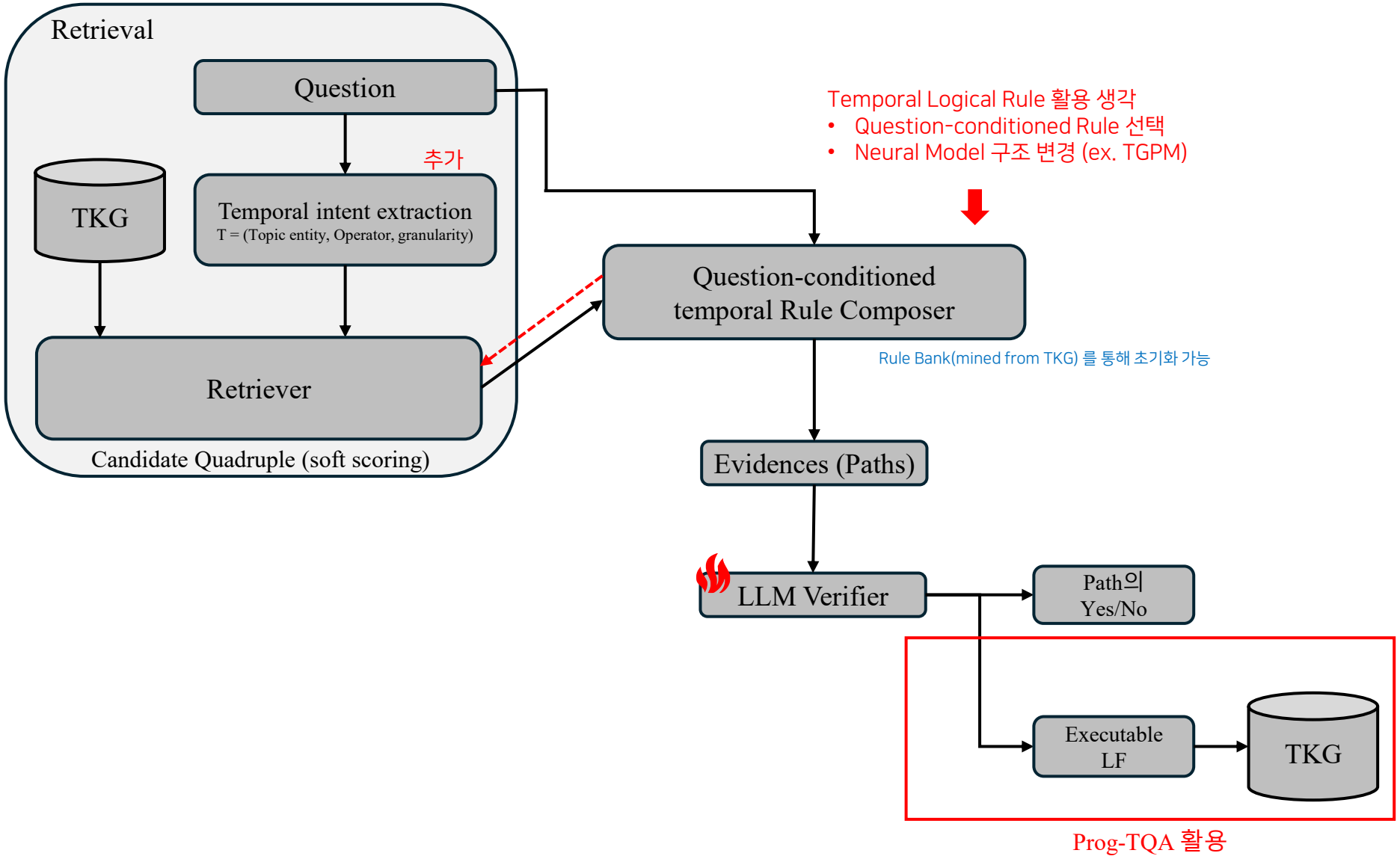
CoD 모듈 별 설명 및 확장 시 발생하는 문제점

모듈 3 추론 (Reasoning)

- CoD
  - ✓ LLM (Llama-2 7B, LoRA)를 파인튜닝하는 단계로 두 목적함수 결합
    - 답 검증 : 질문과 path가 주어졌을 때 Path의 tail entity가 정답으로 도출되는지, 이진 분류
      - Tail이 정답인 경우 Yes
    - Logical Form 생성 : Next-token Prediction으로 LF를 생성하도록 학습
  - ✓ 추론 시 Beam search로 후보를 생성하고 실행 가능한 LF 중 신뢰도 높은 것을 실행
    - LF 실행 불가능 시, Yes 중 Tail entity를 정답으로 활용

- Temporal CoD로 확장 시 문제점
  - ✓ 정답이 timestamp인 경우 Tail entity가 도출되는가?라는 프레임에 맞지 않음
  - ✓ First/last와 같은 질문은 Path를 독립적으로 Yes/No 판정이 아닌 여러 Path 간 비교 및 집계 필요
  - ✓ LF supervised signal 문제
    - Webqsp와 CWQ는 SPARQL이 제공되어 LF 라벨을 만들 수 있지만,
      TKG 벤치마크는 시간 연산자를 포함하는 별도 annotation 체계를 사용하는 경우가 많아 시간 연산자를 포함한 LF 문법과 실행 엔진을 설계해야함

아키텍처


Retrieval
Question
추가
TKG
Temporal intent extraction
T = (Topic entity, Operator, granularity)
Retriever
Candidate Quadruple (soft scoring)
Temporal Logical Rule 활용 생각
• Question-conditioned Rule 선택
• Neural Model 구조 변경 (ex. TGPM)
Question-conditioned
temporal Rule Composer
Rule Bank(mined from TKG) 를 통해 초기화 가능
Evidences (Paths)
LLM Verifier
Path의
Yes/No
Executable
LF
TKG
Prog-TQA 활용

Thank You for Listening