The Graph Hawkes Neural Network for Forecasting on Temporal Knowledge Graphs

By **Zhen Han, Yunpu Ma, Yuyi Wang, Stephan Günnemann, Volker Tresp**

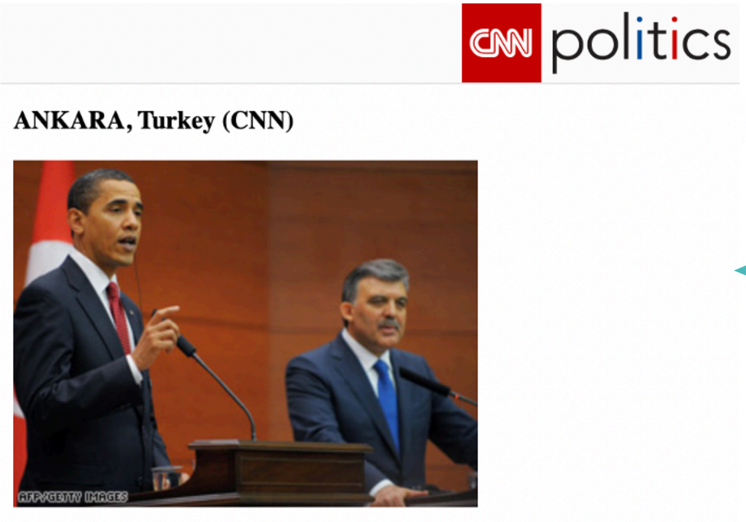
SIEMENS
Ingenuity for life

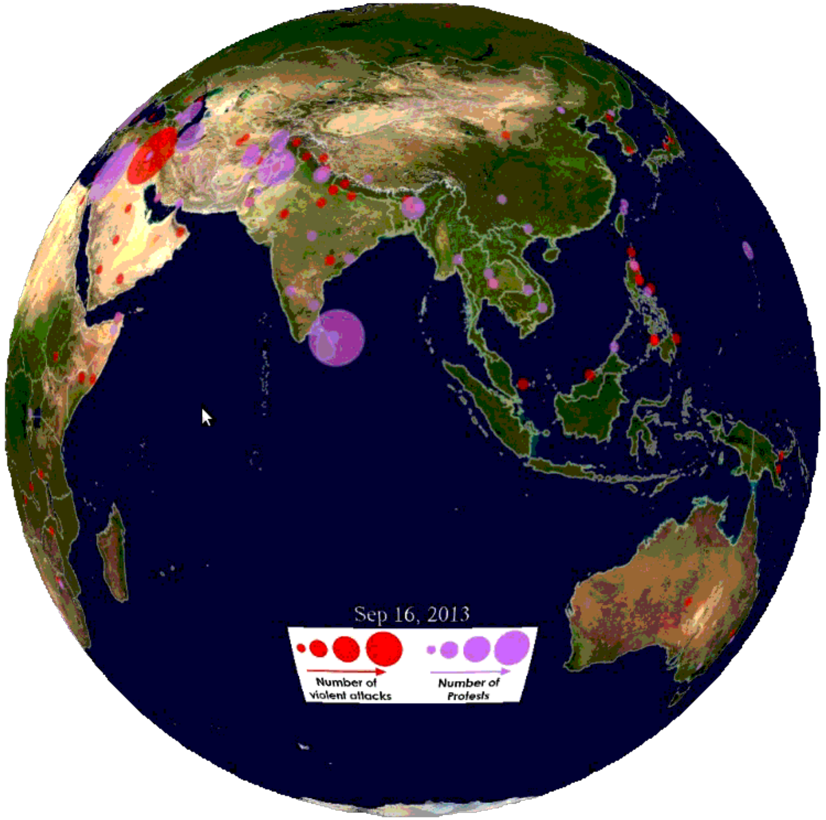
Temporal Knowledge Graph (tKG)

Each Quadruple represents an events:

(subject, predicate, object, timestamp)

(Obama, visit, Turkey, 2009-04-05)





Global Database of Events, Language, and Tone (GDELT)

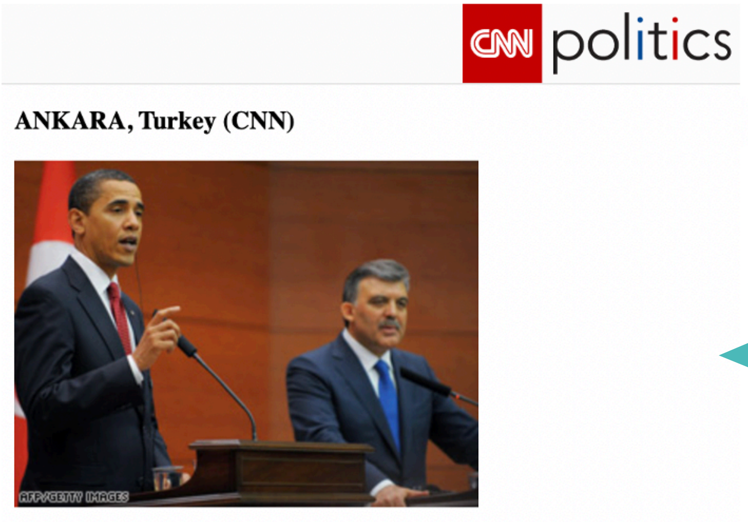
Graph View of a Temporal Knowledge Graph

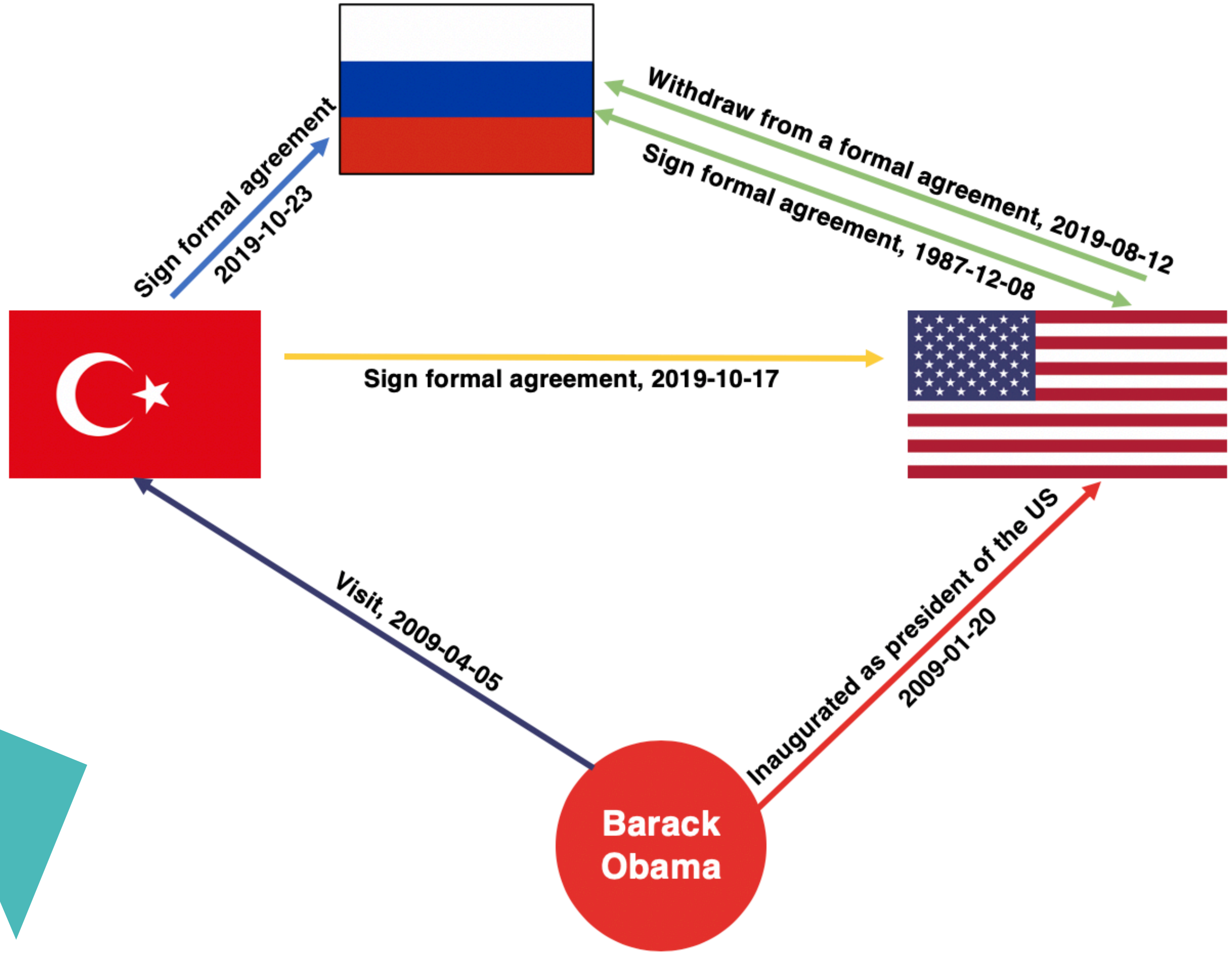
SIEMENS
Ingenuity for life

Each Quadruple represents an events:

(subject, predicate, object, timestamp)

(Obama, visit, Turkey, 2009-04-05)



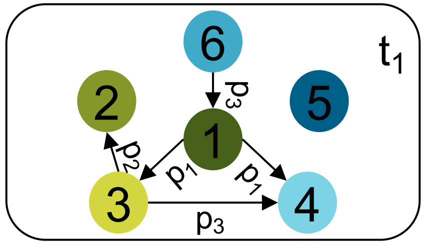


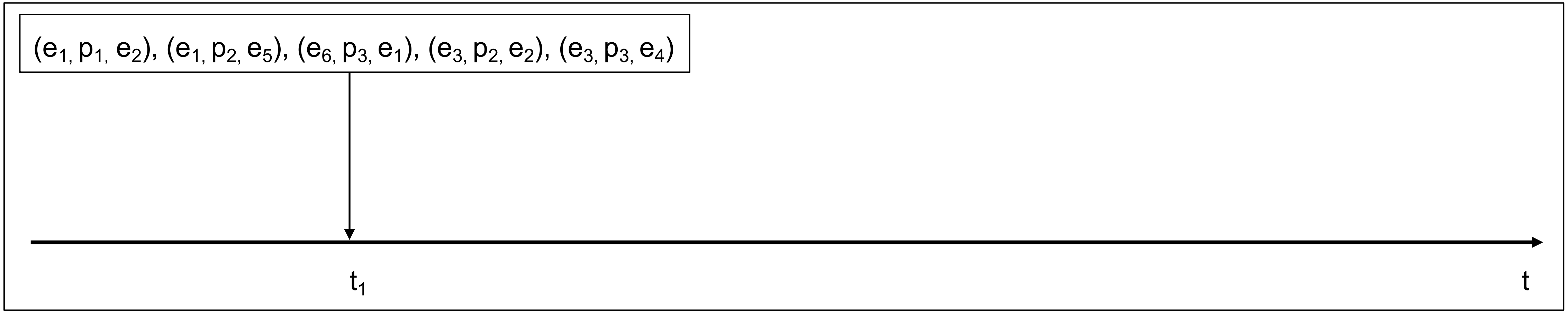
Zhen Han, The Graph Hawkes Neural Network for Forecasting on Temporal Knowledge Graphs, AKBC 2020, June 22th.

From Temporal Knowledge Graph to Event Sequence

SIEMENS

Ingenuity for life

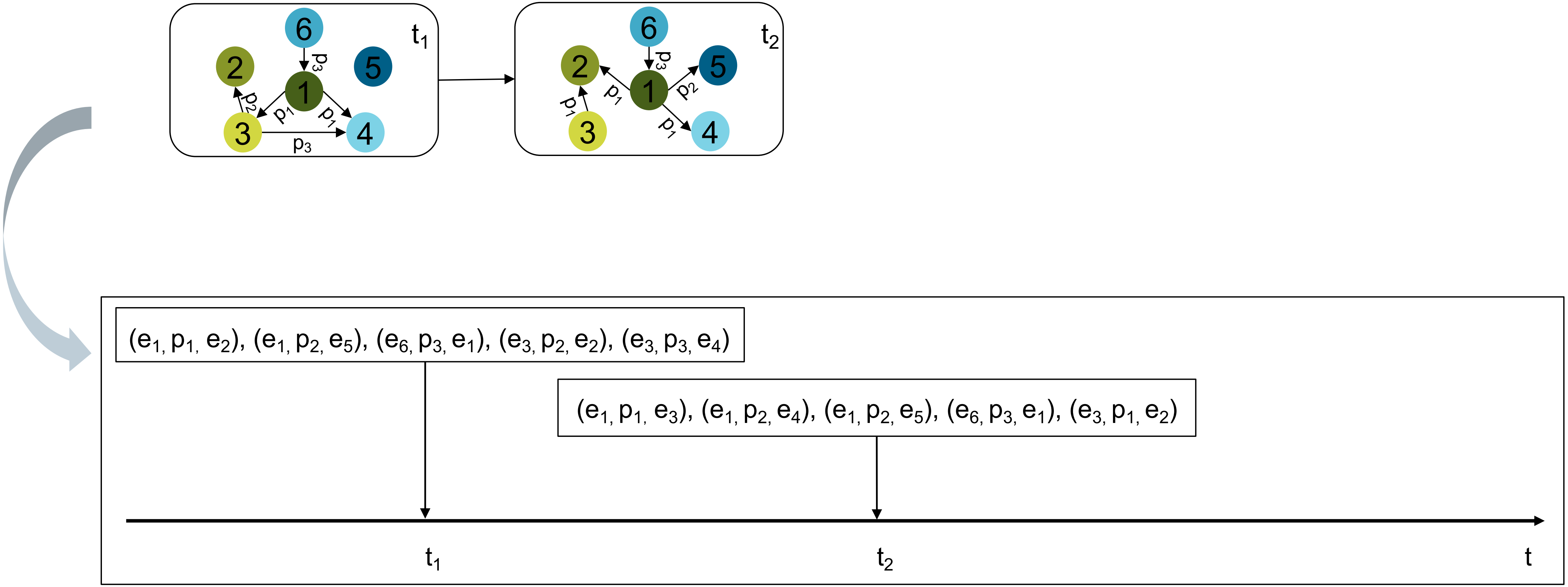




Timeline of a Sequence of Events.

From Temporal Knowledge Graph to Event Sequence

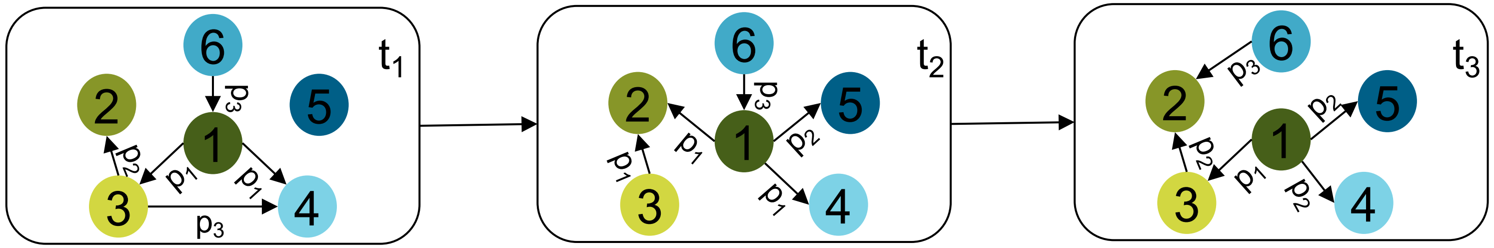
SIEMENS
Ingenuity for life



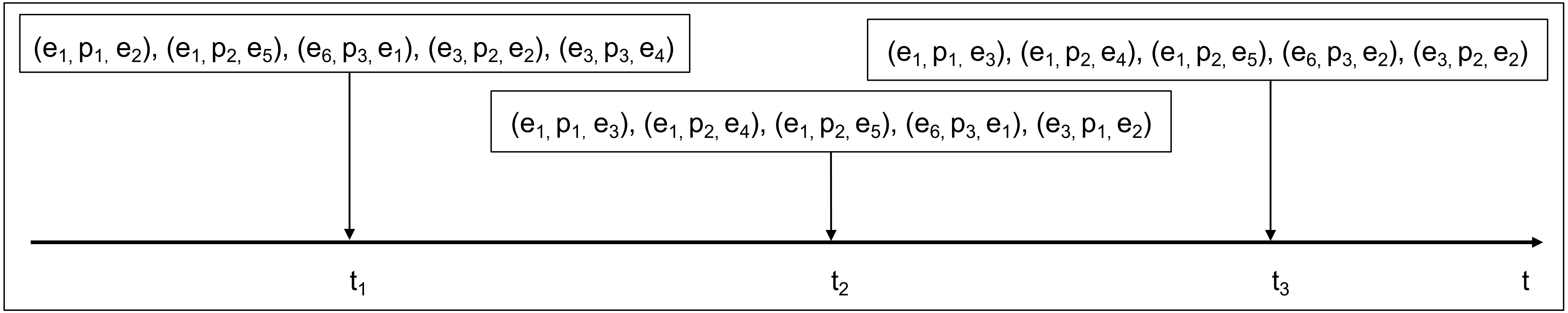
Timeline of a Sequence of Events.

From Temporal Knowledge Graph to Event Sequence

SIEMENS
Ingenuity for life



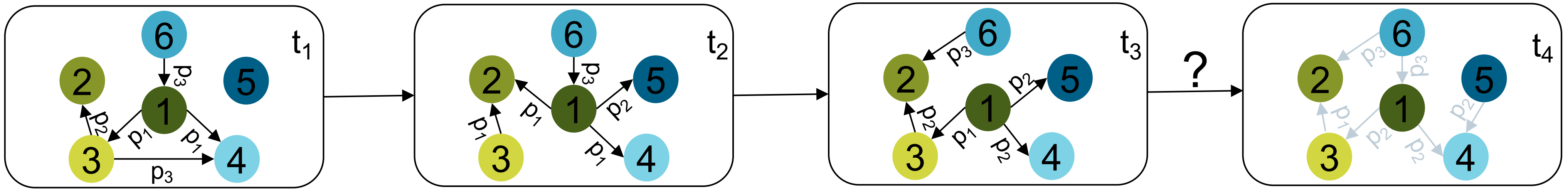
Slices of a Discrete-time Temporal Knowledge Graph.



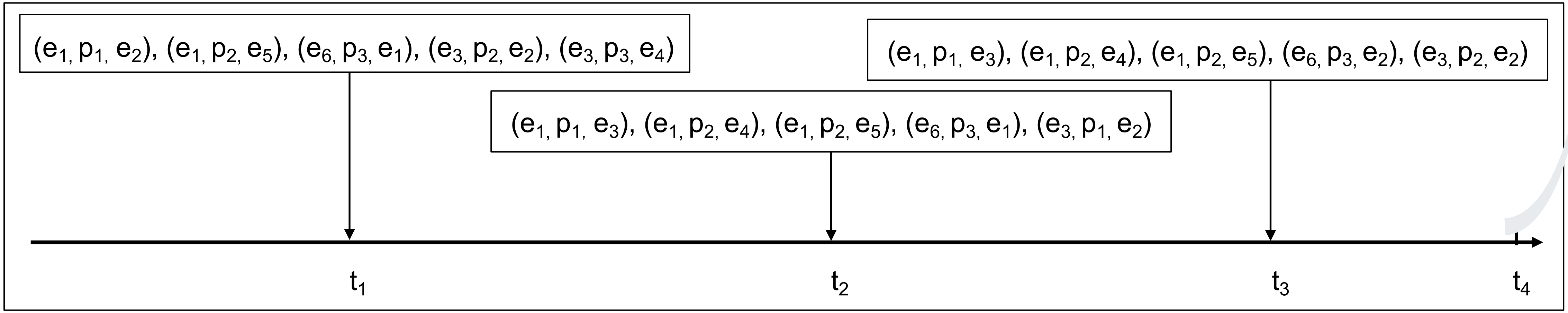
Timeline of a Sequence of Events.

From Temporal Knowledge Graph to Event Sequence

SIEMENS
Ingenuity for life



Slices of a Temporal Knowledge Graph.



Timeline of a Sequence of Events.

Hawkes Process & Neural Hawkes Process

Hawkes Process[2]

$$\lambda_k(t) = \mu_k + \sum_{h:t_h<t} \alpha_{k_h,k} \exp\left(-\delta_{k_h,k}(t - t_h)\right).$$

Intensity function of event type k

Base intensity

Mutual excitation

Exponential decaying with time

Hawkes Process & Neural Hawkes Process

Hawkes Process[2]

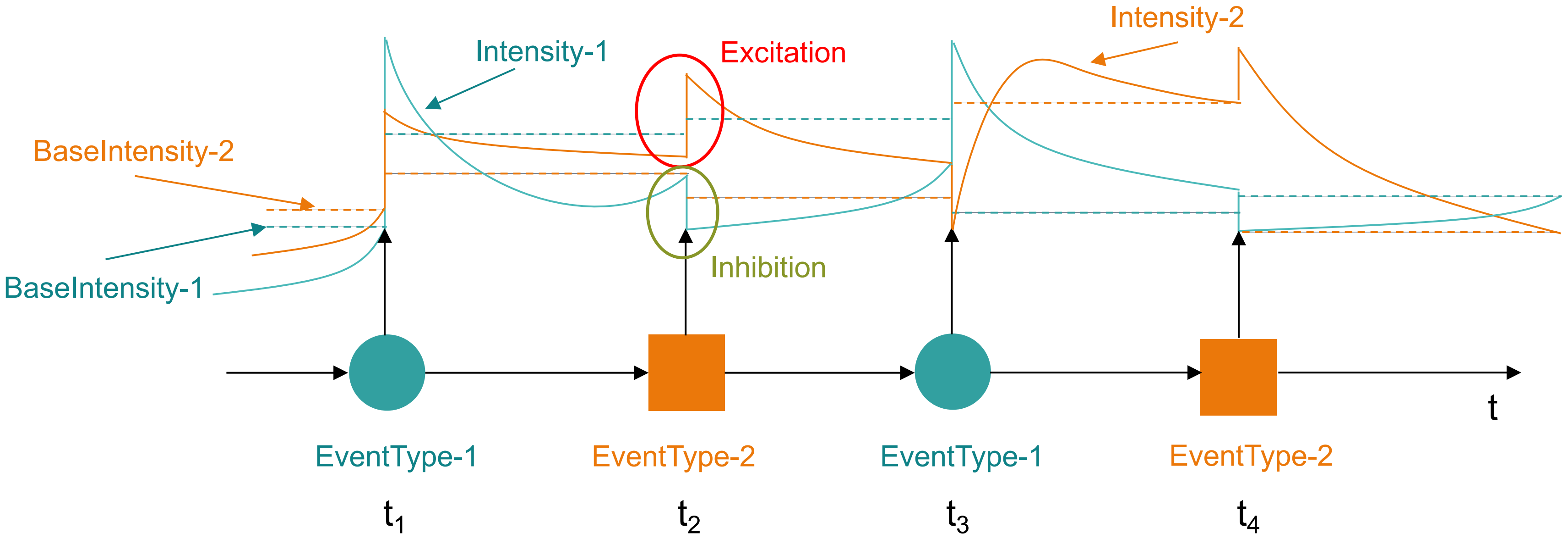
$$\lambda_k(t) = \mu_k + \sum_{h:t_h<t} \alpha_{k_h,k} \exp\left(-\delta_{k_h,k}(t - t_h)\right).$$

Intensity function of event type k Base intensity Mutual excitation Exponential decaying with time

Neural Hawkes Process[3]

$$\lambda_k(t) = f(\boldsymbol{w}_k^T \boldsymbol{h}(t)).$$

Intensity function of event type k Activation function Event-specific weight vector Hidden state vector



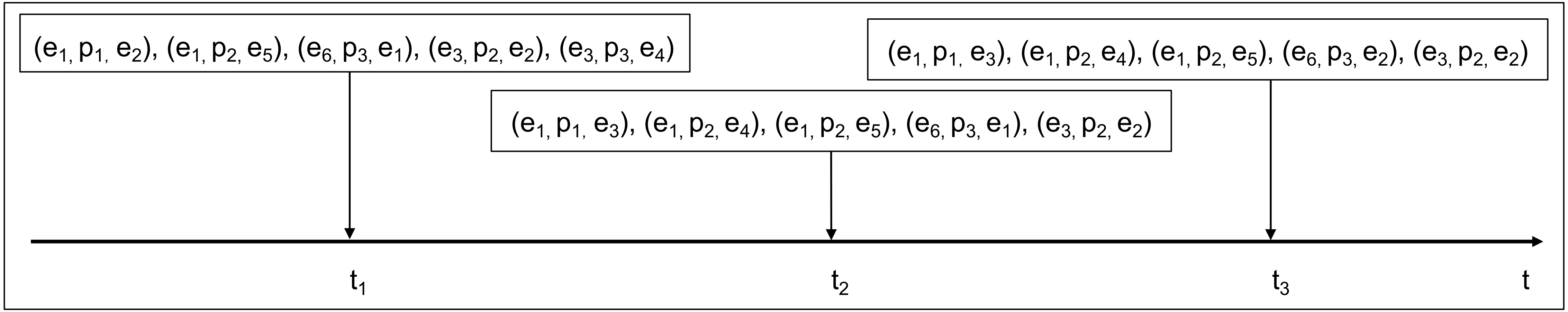
An Event Stream from the Neural Hawkes Process.

Challenge: Characteristics of Temporal Knowledge Graphs

SIEMENS
Ingenuity for life

- **Scalability: a huge amount of event types in tKGs.**
  - Number of **probable** event types in our tKG dataset: $1.4 \cdot 10^{10}$

    (subject, predicate, object)
  - **Existing** event types in our dataset: $1.2 \cdot 10^{6}$



Event Sequence Extracted from a Temporal Knowledge Graph

How to improve the scalability of Hawkes process?

- Considering an **object prediction query** ($e_{1,}$ $p_{1,}$ $?_{,}$ t_4).

How to improve the scalability of Hawkes process?

SIEMENS
Ingenuity for life

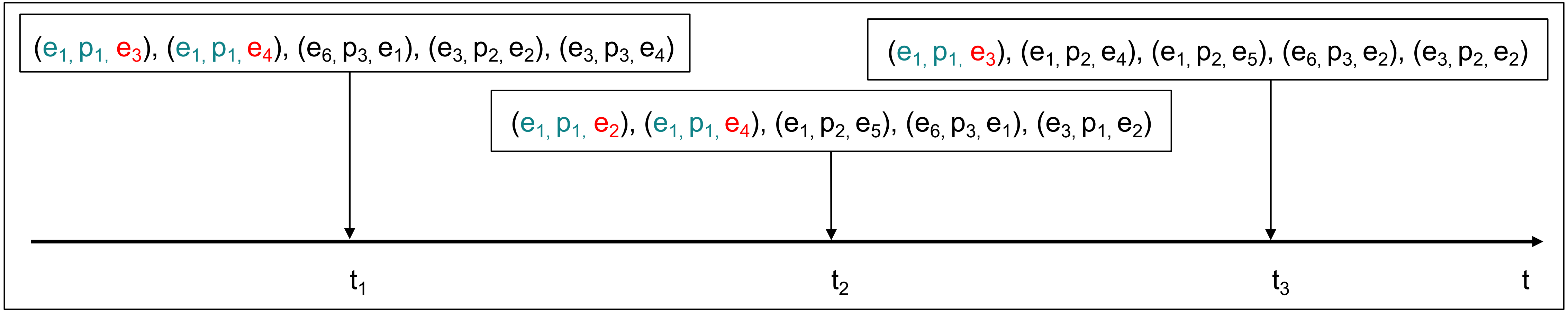
- Considering an **object prediction query** (e_1, p_1, ?, t_4).
- Modelling intensity functions inspired by score functions of KGs

How to improve the scalability of Hawkes process?

SIEMENS
Ingenuity for life

- Considering an **object prediction query** $(e_1, p_1, ?, t_4)$.
- Modelling intensity functions inspired by score functions of KGs
- Investigating the influence of the following historical event sequence:

$$e^{h,sp}(e_1, p_1, t_4) = \{(e_1, p_1, e_3, t_1), (e_1, p_1, e_4, t_1), (e_1, p_1, e_2, t_2), (e_1, p_1, e_4, t_2), (e_1, p_1, e_3, t_3)\}.$$



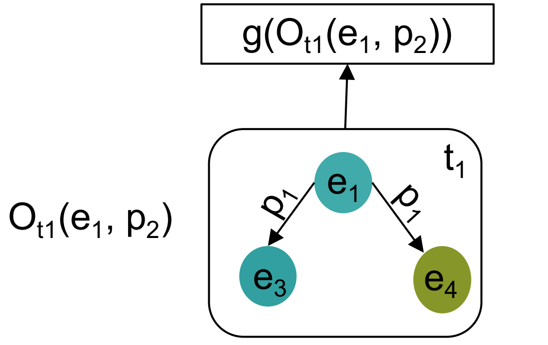
Event Sequence Extracted from a Temporal Knowledge Graph

Neighborhood Aggregation

SIEMENS
Ingenuity for life

- Considering an **object prediction query** $(e_1, p_1, ?, t_4)$.

- Neighborhood Aggregation Module[1] :



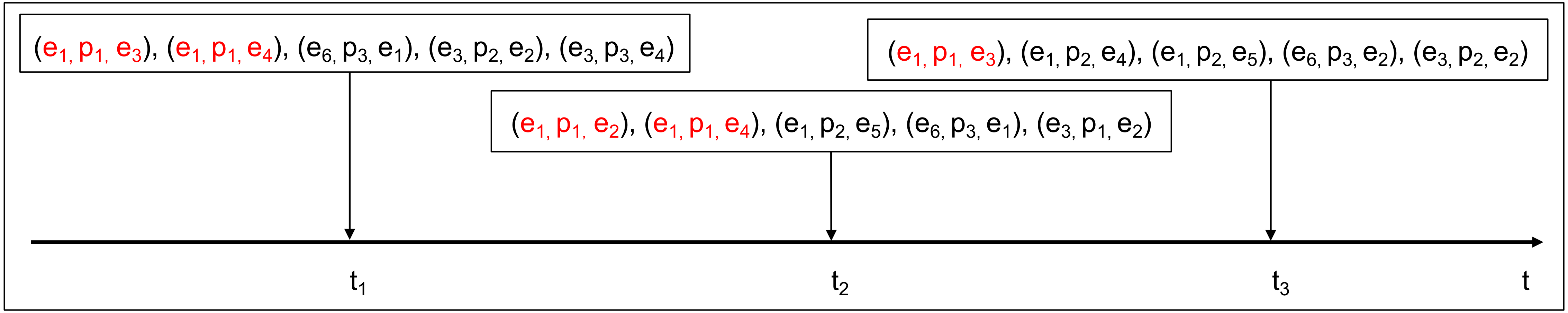
Neighborhood Aggregation

$$g\left(O_{t_1}(e_1, p_2)\right) = \frac{1}{\left|O_{t_1}(e_1, p_2)\right|}(\mathbf{e}_3 + \mathbf{e}_4)$$

$=\{e_3, e_4\}$

Embedding of the 3-th entity

Embedding of the 4-th entity



Event Sequence Extracted from a Temporal Knowledge Graph

Graph Hawkes Process

- Object prediction query $(e_{s_i}, e_{p_i}, ?, t_i)$.

- Hidden state computed by a **continuous-time LSTM** (cLSTM) network[3]

$$\mathbf{h}_{\mathrm{sub}}\left(e_{s_i}, e_{p_i}, t_i, e_i^{h,sp}\right) = \mathrm{cLSTM}\left(\mathbf{e}_{s_i}, \mathbf{e}_{p_i}, \cup_{j=1}^{i} g\left(O_{t_j}(e_{s_i}, e_{p_i})\right)\right)$$

Historical event sequence Subject embedding Predicate embedding Neighborhood aggregation module

Graph Hawkes Process

- Object prediction query $(e_{s_i}, e_{p_i}, ?, t_i)$.

- Hidden state computed by a **continuous-time LSTM** (cLSTM) network[3]

$$\mathbf{h}_{sub}\left(e_{s_i}, e_{p_i}, t_i, e_i^{h,sp}\right) = \mathrm{cLSTM}\left(\mathbf{e}_{s_i}, \mathbf{e}_{p_i}, \cup_{j=1}^{i} g\left(O_{t_j}(e_{s_i}, e_{p_i})\right)\right)$$

Historical event sequence Subject embedding Predicate embedding Neighborhood aggregation module

Inner product

- Subject-centric intensity function

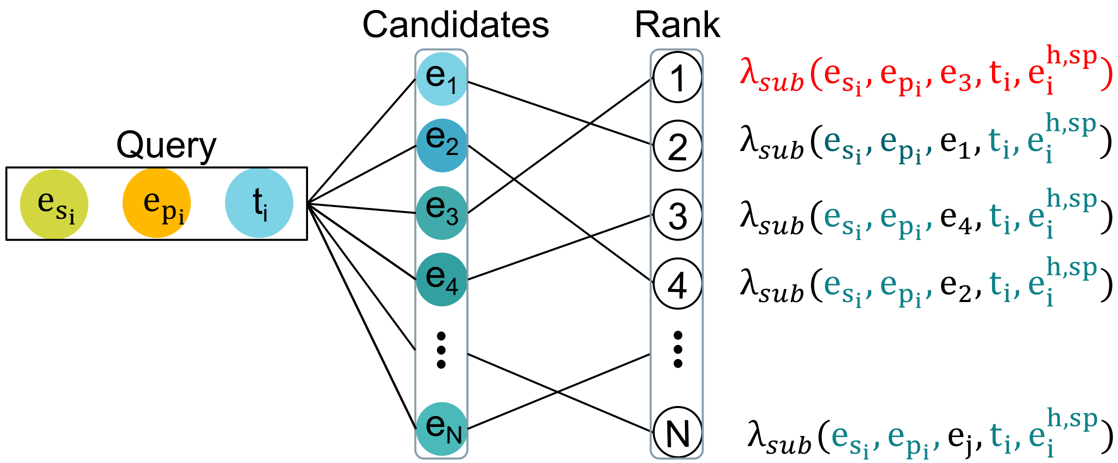
$$\lambda_{sub}\left(e_o | e_{s_i}, e_{p_i}, t_i, e_i^{h,sp}\right) = f\left(\mathbf{W}_\lambda\left(\mathbf{e}_{s_i} \oplus \mathbf{W}_h \mathbf{h}_{sub}\left(e_{s_i}, e_{p_i}, t_i, e_i^{h,sp}\right) \oplus \mathbf{e}_{p_i}\right) \cdot \mathbf{e}_o\right)$$

Historical event sequence Subject embedding Hidden state vector Predicate embedding Object embedding

Link Prediction Task

- Consider an object prediction query $(e_{s_i}, e_{p_i}, ?, t_i)$ and the corresponding $e_i^{h,sp}$.

- Choose the object candidate with the highest intensity.



Time Prediction Task

- Given a time prediction query $(e_{s_i}, e_{p_i}, e_{o_i}, t = ?)$ for $t > t_L$.

Last occurrence time of the given event type

Time Prediction Task

SIEMENS
Ingenuity for life

- Given a time prediction query $\left(e_{s_i}, e_{p_i}, e_{o_i}, t = ?\right)$ for $t > t_L$.

Last occurrence time of the given event type

- Computing conditional probability density that the given event type (e_{si}, e_{pi}, e_{oi}) occurs at time t based on the **survival analysis theory**:

$$p(t|e_{s_i}, e_{p_i}, e_{o_i}, e_i^{h,sp}, e_i^{h,op}) = \lambda_t\left(e_{s_i}, e_{p_i}, e_{o_i}, t, e_i^{h,sp}, e_i^{h,op}\right) \exp\left(-\int_{t_L}^{t} \lambda_t\left(e_{s_i}, e_{p_i}, e_{o_i}, \tau, e_i^{h,sp}, e_i^{h,op}\right) d\tau\right)$$

Intensity function

Historical event sequences

Last occurrence time of the given event type

Time Prediction Task

SIEMENS
Ingenuity for life

- Given a time prediction query $\left(e_{s_i}, e_{p_i}, e_{o_i}, t = ?\right)$ for $t > t_L$.

Last occurrence time of the given event type

- Computing conditional probability density that the given event type (e_{si}, e_{pi}, e_{oi}) occurs at time t based on the **survival analysis theory**:

$$p(t|e_{s_i}, e_{p_i}, e_{o_i}, e_i^{h,sp}, e_i^{h,op}) = \lambda_t\left(e_{s_i}, e_{p_i}, e_{o_i}, t, e_i^{h,sp}, e_i^{h,op}\right) \exp\left(-\int_{t_L}^{t} \lambda_t\left(e_{s_i}, e_{p_i}, e_{o_i}, \tau, e_i^{h,sp}, e_i^{h,op}\right) d\tau\right)$$

Intensity function

Historical event sequences

Last occurrence time of the given event type

- The expectation of the next happening time:

$$\hat{t_i} = \int_{t_L}^{\bowtie} \tau \cdot p(\tau|e_{s_i}, e_{p_i}, e_{o_i}, e_i^{h,sp}, e_i^{h,op})\, d\tau$$

Last occurrence time of the given event type

Probability density function

Historical event sequences

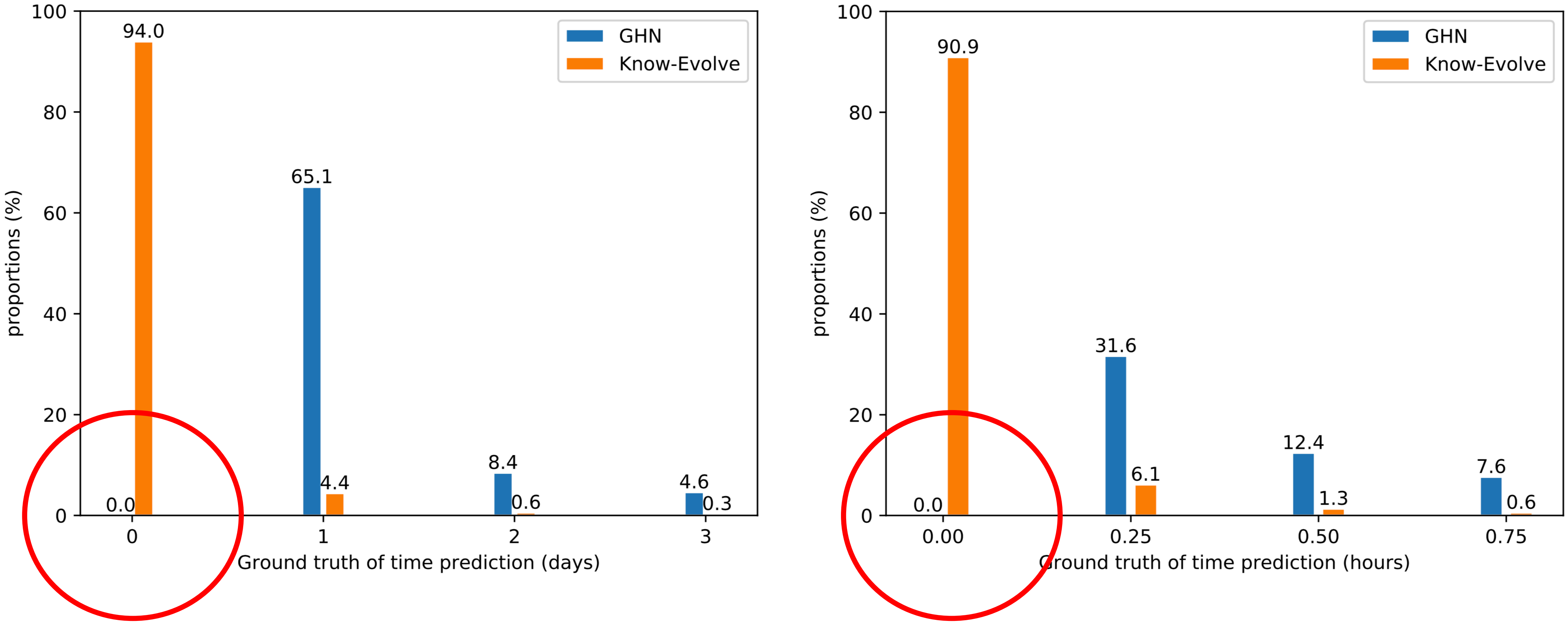
Experimental Results - Link Prediction

SIEMENS
Ingenuity for life

Datasets	GDELT – filtered				ICEWS14 – filtered			
Models	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
T-TransE	5.45	0.44	4.89	15.10	7.15	1.39	6.91	18.93
TA-TransE	9.57	0.00	12.51	27.91	11.35	0.00	15.23	34.25
TA-Dismult	10.28	4.87	10.29	20.43	10.73	4.86	10.86	22.52
LiTSEE	6.64	0.00	8.10	18.72	6.45	0.00	7.00	19.40
GHN	**23.55**	**15.66**	**25.51**	**38.92**	**28.71**	**19.82**	**31.59**	46.47

Table 1: Link prediction results: Mean Reciprocal Rank (MRR, %) and Hits@1/3/10 (%).

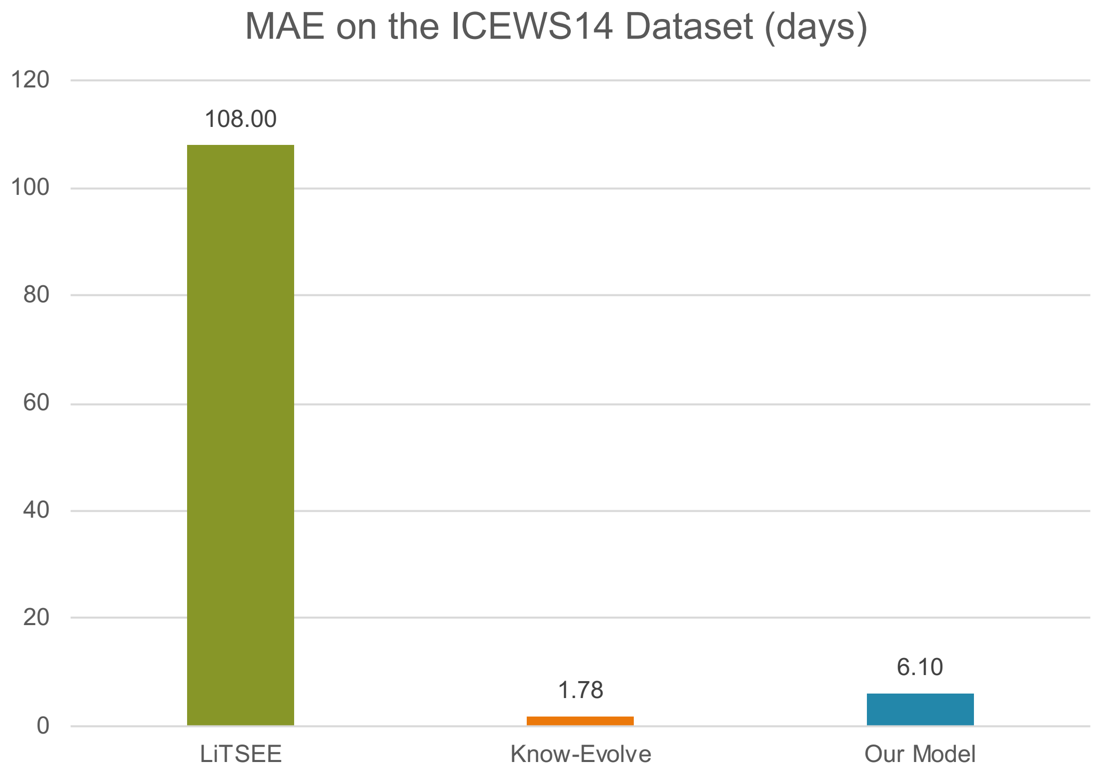
How to Fairly Compare the Time Prediction Performance?

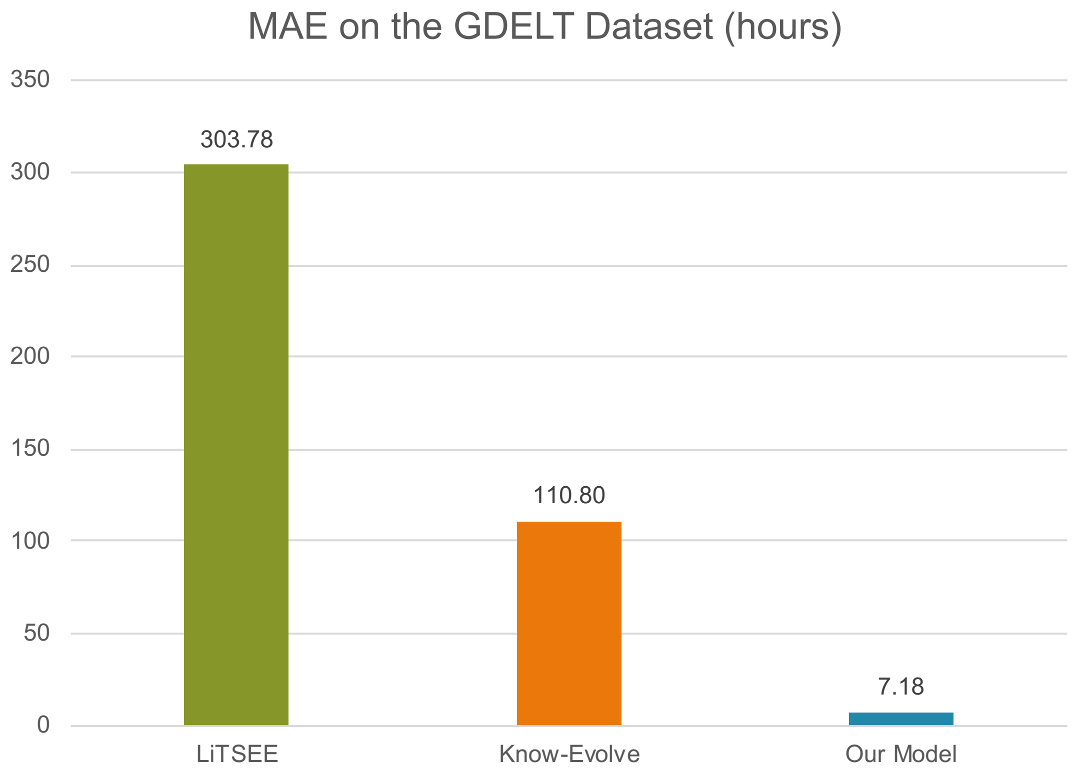


Our model (GHN) is nontrivial for time prediction.

Experimental Results - Time Prediction

SIEMENS
Ingenuity for life





Applications

Integrated conflict early warning

Supporting clinical decisions

in terms of personalized healthcare

Conclusion

- Solving the challenge of massive event types.
- Proposing the Graph Hawkes Process for forecasting on temporal knowledge graphs.
- Define new evaluation metrics on temporal knowledge graph reasoning tasks.

Conclusion

SIEMENS
Ingenuity for life

- Solving the challenge of massive event types.
- Proposing the Graph Hawkes Process for forecasting on temporal knowledge graphs.
- Define new evaluation metrics on temporal knowledge graph reasoning tasks.

Future Work

- Enabling induction on new nodes.
- Explainability.

SIEMENS
Ingenuity for life

Thank you!

Link to our paper: https://openreview.net/forum?id=kXVazet_cB

Reference

SIEMENS
Ingenuity for life

[1] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In Advances in Neural Information Processing Systems, pages 1024–1034, 2017.

[2] Alan G Hawkes. Spectra of some self-exciting and mutually exciting point processes. Biometrika, 58(1):83-90, 1971.

[3] Hongyuan Mei and Jason Eisner. The Neural Hawkes Process: A Neurally Self-Modulating Multivariate Point Process. In Advances in Neural Information Processing Systems, 2017.

[4] Woojeong Jin and Changlin Zhang and Pedro Szekely and Xiang Ren. Recurrent Event Network for Reasoning over Temporal Knowledge Graphs. arXiv preprint arXiv:1904.05530, 2019.

[5] M. S. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling. Modeling relational data with graph convolutional networks. In *ESWC*, 2018.

[6] Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. Know-Evolve: Deep temporal reasoning for dynamic knowledge graphs. In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, pp. 3462–3471, 2017.

[7] Odd Aalen, Ornulf Borgan, and Hakon Gjessing. Survival and event history analysis: a process point of view. Springer Science & Business Media, 2008.