Inductive Future Time Prediction on Temporal Knowledge Graphs with Interval Time

Roxana Pop[1,*], Egor V. Kostylev[1]

[1]*University of Oslo*

Abstract

Temporal Knowledge Graphs (TKGs) are an extension of Knowledge Graphs where facts are temporally scoped. They have recently received increasing attention in knowledge management, mirroring an increased interest in temporal graph learning within the graph learning community. While there have been many systems proposed for TKG learning, there are many settings to be considered, and not all of them are yet fully explored. In this position paper we identify a problem not yet approached, inductive future time prediction on interval-based TKGs, and formalise it as a machine learning task. We then outline several promising approaches for solving it, focusing on a neurosymbolic framework connecting TKG learning with the temporal reasoning formalism DatalogMTL.

Keywords

Temporal Knowledge Graphs, Time prediction, Time intervals, Inductive KG completion

1. Introduction

Knowledge graphs (KGs) are a simple yet powerful formalism for representing semi-structured data, where nodes are entities of interest and directed edges are relations between entities [1]. A common KG format is RDF [2], where facts are triples (s, r, o) with s called the subject, r the relation, and o the object. *Temporal Knowledge Graphs* (TKGs) are an extension of KGs where the validity of each fact is contextualised by temporal information, which shows when the fact is true. TKGs can be classified by the types of temporal scopes they use into *point-based* and *interval-based* TKGs [3]. In point-based TKGs, temporal annotations of facts are points in time, and such facts are suitable for representing instantaneous events; for example, a temporal fact *(Obama, Visits, Canada)@2009* states that Barak Obama visited Canada in 2009. In turn, interval-based TKGs allow for interval temporal annotations, and their facts can represent continuous actions; for example, *(Obama, IsPresidentOf, USA)@[2009,2017]* represents Obama's presidency. Note that each point-based TKG can be seen as interval-based. Similarly to other temporal graphs, TKGs can be classified as *discrete* and *continuous*, depending on the timeline (i.e., set of time points) considered; however, discrete TKGs can always be seen as continuous [4].

KG completion is an important problem for static KGs [1], which aims to extend a presumably incomplete KG with missing facts. This problem can be adapted to TKGs in two possible ways:

dynamic link prediction and *time prediction* [5, 6]. Dynamic link prediction answers the question *'What?'*—that is, fills '*?*' in incomplete temporal facts as *(?, Visits, Canada)@2009*—while time prediction answers *'When?'*—that is, fills '*?*' in, for example, *(Obama, Visits, Canada)@?*. The time prediction task is the less researched one, though arguably more challenging; moreover, systems developed for time prediction can usually also address the dynamic link prediction (see Section 2 for an overview) .

There are several settings in which both the dynamic link prediction and time prediction tasks can be addressed as ML tasks, specified by the way in which the training and validation/test data relate to each other. The *interpolation*/*extrapolation* distinction [7] is made regarding time scopes: if an ML model is restricted to the time points or intervals seen while training, it works under interpolation, but if it can adapt to unseen times (e.g., future ones, relevant for *forecasting*), it works under extrapolation. The *transductive*/*inductive* distinction [5], borrowed from the static graph learning literature [8], is similar in spirit but concerns how the ML model deals with unseen entities: if it can adapt to unseen entities it is *inductive*, and otherwise it is *transductive*.

In short, interval-based TKGs generalize point-based TKGs, time prediction is more challenging than dynamic link prediction, and the extrapolation and inductive settings are more general than the interpolation and transductive ones. This motivates us to introduce and study the ML task of *inductive future time prediction on interval-based TKGs (ITKGs)*. We currently develop neural architectures for this problem, as well as explore connections of them to a recent symbolic temporal reasoning language, DatalogMTL [9]. This position paper outlines our current progress towards the design and evaluation of this neurosymbolic approach.

2. Related work

There are many systems developed for ML tasks on TKGs, though, as we will highlight in the following, few of these systems consider ITKGs, few of them approach the time prediction task and few of them work in the inductive setting—with no overlap that we are aware of.

The existing literature focuses predominantly on point-based TKGs [10, 11, 12, 13, 14, 15, 7, 16, 17, 18, 6], though some works consider interval-based TKGs [3, 19, 20, 21]. As for the timeline type, there are some works viewing TKGs as snapshots of static graphs sampled at equidistant time points, most notably RE-GCN [14] and RE-NET [7], thus working with a discrete timeline. Yet, there are various works, both specifically for TKGs [11, 10, 3, 19, 18, 6], and in the larger temporal graph learning community [4, 22, 23] which focus on continuous time.

Most of the existing TKG learning systems address the dynamic link prediction task [24, 11, 12, 13, 14, 15, 25, 26, 27, 28, 7, 18, 20], and only a few approach also time prediction [10, 3, 19, 16, 21, 29, 6], of which some are limited to time points [10, 16, 6], while others can predict intervals [3, 19, 29]. Some time prediction methods, such as those employed by EvoKG [10], GHNN [16] and Know-Evolve [6] for TKGs, and DyRep [22] for temporal networks, are based on Temporal Point Processes, while the more recent systems that can predict time intervals, such as TIMEPLEX [19] and TIME2BOX [3], use the *greedy coalescing* method [19].

As for the settings, there are some works focusing on interpolation [30, 31, 3, 18, 29], though most systems target extrapolation [32, 10, 11, 33, 12, 13, 14, 15, 25, 7, 16, 17]. Yet, there are not many inductive TKG systems, and their approaches are varied: TLogic [11] is based on temporal

graphs, FILT [34] on concept-aware mining, and TANGO [25] on neural ODEs [35]. If we look at the broader static and temporal graph learning areas, inductive capabilities are often achieved by using architectures based on *Graph Neural Networks* (*GNNs*) [22, 23, 36, 37, 8].

Most of the aftermentioned methods are neural in nature, with the notable exception of TLogic [11], which mines temporal logical rules. Yet, the rules in TLogic are limited to time points. On the symbolic side, there exist temporal logics that can deal with time intervals, such as DatalogMTL [9]—a recently introduced formalism extending Datalog [38] to the temporal dimension. Datalog is a rule-based logical language which can be used for static KG reasoning and which has been utilised in neurosymbolic methods in KG learning [37]. While the connections of DatalogMTL and ITKG learning have not yet been explored, a DatalogMTL program can generate new temporal facts via reasoning and could hence be seen as a predictor on ITKG data. This predictor could be used for both dynamic link prediction and time prediction, could work in an inductive setting (similar to Datalog for static KGs [37]), and could be restricted to only generate facts with future temporal annotations — working in the extrapolation setting.

3. Problem formalisation

In this section, we formalise the problem that we study, starting from basic notions such as temporal knowledge graphs and concluding with its cast as an ML task.

Let $\mathcal{T}$ and $\mathcal{R}$ be finite sets of *types* and *relations*, respectively, collectively called *predicates* $\mathcal{P}$, and let $\mathcal{E}$ be an infinite set of *entities*, also known as *constants*. Let $\mathbb{T}$ be a *timeline*—that is, a set of timepoints; in our context, it is either integers $\mathbb{Z}$ or rationals $\mathbb{Q}$. We are interested in intervals over $\mathbb{T}$, and concentrate on the set $\mathsf{Int}_{\mathbb{T}}$ of non-empty closed intervals $[t_1, t_2] \subset \mathbb{T}$ with $t_1 \leq t_2$. An interval of the form $[t_1, t_1]$ is *punctual*, and we may write it just t_1.

A *fact* is a triple of the form (e, type, T), where $e \in \mathcal{E}$ and $T \in \mathcal{T}$, or of the form (e_1, R, e_2), where $e_1, e_2 \in \mathcal{E}$ and $R \in \mathcal{R}$. Then, a *temporal fact* is $\lambda @ \rho$, where λ is a fact and $\rho \in \mathsf{Int}_{\mathbb{T}}$.

Definition 1. *An* interval-based temporal knowledge graph *(ITKG) over* $\mathbb{T}$ *is a set of facts (which we call* atemporal *in this context) and temporal facts. An ITKG is a* point-based temporal knowledge graph *(PTKG) if all the intervals in its temporal facts are punctual.*

For an ITKG G, let $\mathsf{Pred}(G)$ and $\mathsf{Const}(G)$ denote the predicates and entities appearing in G, respectively, and let $\mathsf{Sig}(G) = \mathsf{Pred}(G) \cup \mathsf{Const}(G)$.

Intuitively, an atemporal fact in an ITKG represents something that holds all the time, so it is redundant to have a temporal version of this triple in the same ITKG; moreover, overlaps of intervals for the same triple are also redundant. This motivates the following notion: an ITKG G is in *normal form* if there is no $\lambda @ \rho$ in G with λ in G (as an atemporal triple), and there are no $\lambda @ \rho_1$ and $\lambda @ \rho_2$ in G with $\rho_1 \cap \rho_2 \neq \emptyset$. It is straightforward to reduce an ITKG to an ITKG in normal form in a unique way, and the resulting ITKG is semantically equivalent to the original one. So, in the rest of this paper, we silently concentrate on normal ITKGs.

Every time point $t \in \mathbb{T}$ limits the *past* subgraph $G_{\leq t}$ of an ITKG G over $\mathbb{T}$ that contains

- every atemporal fact λ in G;
- every fact $\lambda @ [t_1, t_2']$ with $t_2' = \min(t_2, t)$ for a fact $\lambda @ [t_1, t_2] \in G$.

Intuitively, future time prediction on ITKGs is the problem of predicting future temporal facts of an ITKG G on the base of its past counterpart $G_{\leq t}$. To formalise this problem as an ML task, we assume that every ITKG $G_{\leq t}$ with t the maximal time point in an interval of $G_{\leq t}$ has the (most probable) *temporal completion* G with $\mathsf{Sig}(G) = \mathsf{Sig}(G_{\leq t})$ such that $G_{\leq t}$ is the past graph of G limited by t. In the following definition we will concentrate on *time prediction*—that is, on predicting the nearest to t maximal future interval for a given tuple or the absence of such an interval. We also consider the general *inductive prediction*—that is, the setting where the prediction function applies to any ITKG over the given predicates $\mathcal{P}$, while the entities may be arbitrary. In particular, an inductive ML model trained on ITKGs with one set of entities should be applicable to ITKGs with any other entities.

Definition 2. *The inductive* next interval *function* $f_{\mathsf{next\text{-}int}}(G_{\leq t}, \lambda)$ *maps an ITKG $G_{\leq t}$ over $\mathbb{T}$ with* $\mathsf{Pred}(G_{\leq t}) \subseteq \mathcal{P}$ *and temporal completion G, and a triple λ over $\mathsf{Sig}(G_{\leq t})$ to the smallest interval $[t_1, t_2]$ such that $t_1 \geq t$, $t_2 > t$, and $\lambda@[t_1, t_2] \in G$, if such an interval exists, and to a special symbol $\emptyset$ otherwise; here, an interval $[t_1, t_2]$ is* smaller *than another interval $[t_1', t_2']$ if $t_1 < t_1'$ (note that, due to normalisation, we need not compare overlapping intervals).*

Thus, the ML task of *inductive future time prediction on ITKGs* for the time domain $\mathbb{T}$ is to learn (in a supervised way) the next interval function $f_{\mathsf{next\text{-}int}}$.

4. Proposed approaches

The main approach we would like to investigate is neurosymbolic in nature. We would like to develop a framework in which we train a neural architecture for time interval prediction and then extract a temporal logical program from the trained model that can generate the future time intervals through the means of temporal reasoning. As baselines we will use purely neural methods to make sure the neurosymbolic method has at least comparable empirical results.

4.1. Neurosymbolic architecture

Monotonic GNNs (*MGNNs*) [37] are a class of GNNs introduced for KG completion, which generate the same facts on an input KG as the application of a set of Datalog [38] rules. Moreover, for each trained MGNN model, the equivalent Datalog rules can be automatically extracted [37], resulting in a neurosymbolic architecture that allows for a smooth switch between the two paradigms. We are currently generalising this architecture to ITKGs, moving from Datalog to its temporal counterpart, DatalogMTL. One of the key insights of the MGNN-based (static) KG completion system is to encode the original graph into a different graph in which each (potential) edge becomes a node, and the existence of a certain type or relation is given by a feature attached to such a node. We exemplified in Figure 1 how this encoding could be expanded to ITKGs (with some technical details omitted for simplicity). The nodes of the encoding are pairs of constants in the original graph, edges link nodes that share constants, and the node features are indexed by types and relations (which are *Human*, *IsPresidentOf*, *Visits*, *IsPresidentOf*$^{-1}$, *Visits*$^{-1}$ in our example). However, while in the static case [37] the features indicate through Booleans the truth values of types and relations (e.g. $[0, 0, 0, 0, 1]$ for (*Canada*, *Obama*)), in our case they

contain the time intervals where the facts are true. In case of multiple time intervals we have multiple node features; see features for (*Canada*, *Obama*). How and if MGNNs or other GNNs can be modified to work in the temporal case is something we are currently researching.
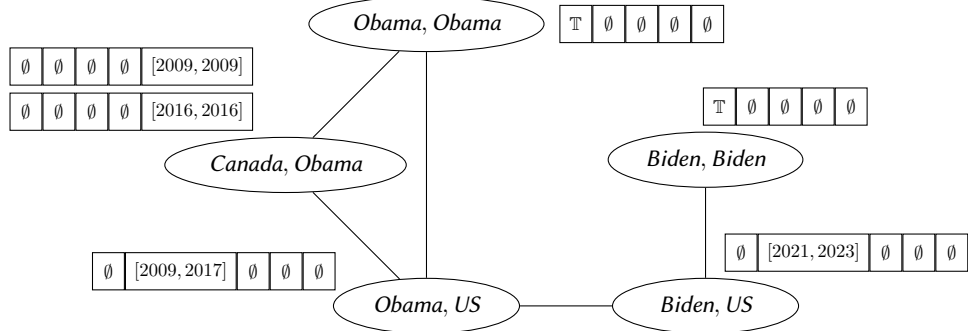

Figure 1: Edge-based graph transformation of the ITKG {(*Obama*, type, *Human*), (*Biden*, type, *Human*), (*Obama*, *IsPresidentOf*, *US*)@[2009, 2017], (*Biden*, *IsPresidentOf*, *US*)@[2021, 2023], (*Obama*, *Visits*, *Canada*)@2009, (*Obama*, *Visits*, *Canada*)@2016}

4.2. Benchmarks, baselines, and metrics

Existing works for time prediction on ITKGs [19, 3] evaluate time prediction performance on the YAGO11k [29], Wikidata12k [29], and Wikidata114K [3] datasets. We will investigate if these datasets can be turned into inductive benchmarks, as well as design new benchmarks from other relevant datasets.

Regarding baselines, we believe that GraphMixer [39], a recent system based on the MLP-Mixer architecture [40], is a good candidate due to its simplicity, and we plan to adapt it to time prediction on ITKGs. We will also investigate GNN-based architectures with inductive and continuous time capabilities such as DyRep [22], TGN [23], and EvoKg [10]. Some of these architectures have time prediction capabilities, but they are limited to time points. For the architectures where time interval prediction is not achievable through simple modifications, we will employ the greedy coalescing method [19]. With regards to evaluation metrics, two have been proposed for the interval time prediction task: *aeIOU* [19] and *gaeIOU* [3], of which *gaeIOU* has more desirebale properties [3] and it is the one we will therefore concentrate on.

5. Conclusions and future work

In this paper we highlighted the more general views on TKGs (continuous and interval-based), the different ML-based tasks approached in the literature (dynamic link and time prediction), and the more general ML settings (extrapolative and inductive). We then formalised the future time prediction task on interval-based TKGs, and proposed to extend a neurosymbolic framework from the static KG case to approach this task, as well as provided a way of extending the graph encoding from the static case. Our next steps are to adapt GNN-based architectures to work on the encoded graph and explore DatalogMTL programs extraction from the trained models.

References

[1] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. F. Sequeda, S. Staab, A. Zimmermann, Knowledge graphs, ACM Comput. Surv. 54 (2022) 71:1–71:37.

[2] F. Manola, E. Miller, RDF Primer, W3C Recommendation, 2004.

[3] L. Cai, K. Janowicz, B. Yan, R. Zhu, G. Mai, Time in a box: Advancing knowledge graph completion with temporal scopes, in: The Conference on Knowledge Capture Conference (K-CAP), 2021, pp. 121–128.

[4] A. H. Souza, D. Mesquita, S. Kaski, V. Garg, Provably expressive temporal graph networks, in: The Advances in Neural Information Processing Systems (NeurIPS), 2022.

[5] S. M. Kazemi, R. Goel, K. Jain, I. Kobyzev, A. Sethi, P. Forsyth, P. Poupart, Representation learning for dynamic graphs: A survey, 2020.

[6] R. Trivedi, H. Dai, Y. Wang, L. Song, Know-Evolve: Deep temporal reasoning for dynamic knowledge graphs, in: The International Conference on Machine Learning (ICML), 2017, pp. 3462–3471.

[7] W. Jin, M. Qu, X. Jin, X. Ren, Recurrent event network: Autoregressive structure inferenceover temporal knowledge graphs, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 6669–6683.

[8] W. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, in: The Advances in Neural Information Processing Systems (NeurIPS), 2017.

[9] S. Brandt, E. G. Kalaycı, V. Ryzhikov, G. Xiao, M. Zakharyaschev, Querying log data with metric temporal logic, J. Artif. Intell. Res. 62 (2018) 829–877.

[10] N. Park, F. Liu, P. Mehta, D. Cristofor, C. Faloutsos, Y. Dong, EvoKG: Jointly modeling event time and network structure for reasoning over temporal knowledge graphs, in: The ACM International Conference on Web Search and Data Mining (WSDM), 2022, p. 794–803.

[11] Y. Liu, Y. Ma, M. Hildebrandt, M. Joblin, V. Tresp, TLogic: Temporal logical rules for explainable link forecasting on temporal knowledge graphs, in: The AAAI Conference on Artificial Intelligence (AAAI), 2022, pp. 4120–4127.

[12] C. Zhu, M. Chen, C. Fan, G. Cheng, Y. Zhang, Learning from history: Modeling temporal knowledge graphs with sequential copy-generation networks, in: The AAAI Conference on Artificial Intelligence (AAAI), 2021, pp. 4732–4740.

[13] H. Sun, J. Zhong, Y. Ma, Z. Han, K. He, TimeTraveler: Reinforcement learning for temporal knowledge graph forecasting, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021, pp. 8306–8319.

[14] Z. Li, X. Jin, W. Li, S. Guan, J. Guo, H. Shen, Y. Wang, X. Cheng, Temporal knowledge graph reasoning based on evolutional representation learning, in: The International Conference on Research and Development in Information Retrieval (SIGIR), 2021, pp. 408–417.

[15] Z. Li, X. Jin, S. Guan, W. Li, J. Guo, Y. Wang, X. Cheng, Search from history and reason for future: Two-stage reasoning on temporal knowledge graphs, in: The Annual Meeting of the Association for Computational Linguistics (ACL) and the International Joint Conference on Natural Language Processing (ICNLP), 2021, pp. 4732–4743.

[16] Z. Han, Y. Wang, Y. Ma, S. Günnemann, V. Tresp, Graph hawkes neural network for future prediction on temporal knowledge graphs, in: The Automated Knowledge Base Construction (AKBC), 2020.

[17] Z. Han, P. Chen, Y. Ma, V. Tresp, Explainable subgraph reasoning for forecasting on temporal knowledge graphs, in: The International Conference on Learning Representations (ICLR), 2021.

[18] R. Goel, S. M. Kazemi, M. Brubaker, P. Poupart, Diachronic embedding for temporal knowledge graph completion, in: The AAAI Conference on Artificial Intelligence (AAAI), 2020, pp. 3988–3995.

[19] P. Jain, S. Rathi, Mausam, S. Chakrabarti, Temporal Knowledge Base completion: New algorithms and evaluation protocols, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 3733–3747.

[20] A. García-Durán, S. Dumančić, M. Niepert, Learning sequence encoders for temporal knowledge graph completion, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2018, pp. 4816–4821.

[21] J. Leblay, M. W. Chekol, Deriving validity time in knowledge graph, in: The Web Conference (WWW), 2018, pp. 1771–1776.

[22] R. S. Trivedi, M. Farajtabar, P. Biswal, H. Zha, DyRep: Learning Representations over Dynamic Graphs, in: The International Conference on Learning Representations (ICLR), 2019.

[23] E. Rossi, B. Chamberlain, F. Frasca, D. Eynard, F. Monti, M. Bronstein, Temporal graph networks for deep learning on dynamic graphs, in: The ICML Workshop on Graph Representation Learning (GRL@ICML), 2020.

[24] P. Shao, D. Zhang, G. Yang, J. Tao, F. Che, T. Liu, Tucker decomposition-based temporal knowledge graph completion, Knowledge-Based Systems 238 (2022).

[25] Z. Han, Z. Ding, Y. Ma, Y. Gu, V. Tresp, Learning neural ordinary equations for forecasting future links on temporal knowledge graphs, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021, pp. 8352–8364.

[26] J. Wu, M. Cao, J. C. K. Cheung, W. L. Hamilton, TeMP: Temporal message passing for temporal knowledge graph completion, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 5730–5746.

[27] T. Lacroix, G. Obozinski, N. Usunier, Tensor decompositions for temporal knowledge base completion, in: The International Conference on Learning Representations (ICLR), 2020.

[28] J. Jung, J. Jung, U. Kang, Learning to walk across time for temporal knowledge graph completion, in: The Conference on Knowledge Discovery and Data Mining (SIGKDD), 2021, p. 786–795.

[29] S. S. Dasgupta, S. N. Ray, P. Talukdar, HyTE: Hyperplane-based temporally aware knowledge graph embedding, in: The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2018, pp. 2001–2011.

[30] Y.-C. Lee, J. Lee, D. Lee, S.-W. Kim, THOR: Self-supervised temporal knowledge graph embedding via three-tower graph convolutional networks, in: The International Conference on Data Mining (ICDM), 2022, pp. 1035–1040.

[31] A. Sadeghian, M. Armandpour, A. Colas, D. Z. Wang, ChronoR: Rotation based temporal knowledge graph embedding, in: The AAAI Conference on Artificial Intelligence (AAAI),

2021, pp. 6471–6479.

[32] S. Wang, X. Cai, Y. Zhang, X. Yuan, CRNet: Modeling concurrent events over temporal knowledge graph, in: The International Semantic Web Conference (ISWC), 2022, pp. 516–533.

[33] Z. Li, S. Guan, X. Jin, W. Peng, Y. Lyu, Y. Zhu, L. Bai, W. Li, J. Guo, X. Cheng, Complex evolutional pattern learning for temporal knowledge graph reasoning, in: The Annual Meeting of the Association for Computational Linguistics (ACL), 2022, pp. 290–296.

[34] Z. Ding, J. Wu, B. He, Y. Ma, Z. Han, V. Tresp, Few-shot inductive learning on temporal knowledge graphs using concept-aware information, in: The Conference on Automated Knowledge Base Construction (AKBC), 2022.

[35] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, D. K. Duvenaud, Neural Ordinary Differential Equations, in: The Advances in Neural Information Processing Systems (NeurIPS), volume 31, Curran Associates, Inc., 2018.

[36] S. Liu, B. Cuenca Grau, I. Horrocks, E. V. Kostylev, INDIGO: GNN-based inductive knowledge graph completion using pair-wise encoding, in: The Advances in Neural Information Processing Systems (NeurIPS), 2021, pp. 2034–2045.

[37] D. J. Tena Cucala, B. Cuenca Grau, E. V. Kostylev, B. Motik, Explainable GNN-based models over knowledge graphs, in: The International Conference on Learning Representations (ICLR), 2022.

[38] S. Abiteboul, R. Hull, V. Vianu, Foundations of Databases, Addison-Wesley, 1995.

[39] W. Cong, S. Zhang, J. Kang, B. Yuan, H. Wu, X. Zhou, H. Tong, M. Mahdavi, Do we really need complicated model architectures for temporal networks?, in: The International Conference on Learning Representations (ICLR), 2023.

[40] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, A. Dosovitskiy, MLP-Mixer: An all-MLP Architecture for Vision, in: The Advances in Neural Information Processing Systems (NeurIPS), 2021, pp. 24261–24272.