Dynamic Multi-Context Attention Networks for Citation Forecasting of Scientific Publications

Taoran Ji,[1, 2] **Nathan Self,** [1, 3] **Kaiqun Fu,** [1, 2] **Zhiqian Chen,** [4] **Naren Ramakrishnan,** [1, 3]
and Chang-Tien Lu [1, 2]

[1] Discovery Analytics Center, Virginia Tech, Arlington, VA 22203, USA
[2] Department of Computer Science, Virginia Tech, Falls Church, VA 22203, USA
[3] Department of Computer Science, Virginia Tech, Arlington, VA 22203, USA
[4] Department of Computer Science and Engineering, Mississippi State University, MS 39762, USA
{jtr, nwself, fukaiqun}@vt.edu, zchen@cse.msstate.edu, naren@cs.vt.edu, ctlu@vt.edu

Abstract

Forecasting citations of scientific patents and publications is a crucial task for understanding the evolution and development of technological domains and for foresight into emerging technologies. By construing citations as a time series, the task can be cast into the domain of temporal point processes. Most existing work on forecasting with temporal point processes, both conventional and neural network-based, only performs single-step forecasting. In citation forecasting, however, the more salient goal is n-step forecasting: predicting the arrival time and the technology class of the next n citations. In this paper, we propose Dynamic Multi-Context Attention Networks (DMA-Nets), a novel deep learning sequence-to-sequence (Seq2Seq) model with a novel hierarchical dynamic attention mechanism for long-term citation forecasting. Extensive experiments on two real-world datasets demonstrate that the proposed model learns better representations of conditional dependencies over historical sequences compared to state-of-the-art counterparts and thus achieves significant performance for citation predictions. The dataset and code have been made available online.[1]

Introduction

The evolution of technology is a coupling of prior work with new innovations in incremental or disruptive fashions. As such, as a paper or patent receives citations, their frequency and provenance can serve as a reflection of that evolutionary character. Citation-based bibliometrics analysis, such as g-index (Egghe 2006) and H-index (Hirsch 2005), have become well-accepted standard measures applied to individuals, high-tech companies, and institutions alike. The technological diversity of scientific documents, such as generality and originality, critical factors for decision-makers, can be measured via the technology class of the citing documents (Bessen 2008). Furthermore, patent citation statistics have been widely used for the tasks of technology impact analysis (Jang, Woo, and Lee 2017), patent quality assessment (Bessen 2008) (Lee et al. 2018), and identifying emerging technologies at an early stage. Citation forecasting is a field of growing importance due to the accelerating pace of technological change in increasingly competitive industrial and academic environments.

Many previous works (Yan et al. 2011; Acuna, Allesina, and Kording 2012; Yan et al. 2012) regard citation prediction problems as feature-driven regression tasks. Often, domain-specific, handcrafted features (e.g., domain keywords, topics, quality indicators, author information) are collected to formulate a deterministic model to predict the future citation count. However, these models cannot predict the technological categories of citing documents and thus are incapable of technological diversity assessment. Also, these models require prior domain knowledge and are hard to extend to different research areas. The performance of such models depends heavily on the quality of collected features. But, in real-world datasets, features like author lists and institution information (Kim et al. 2014) are often noisy, especially when considering papers from multiple disciplines. Furthermore, this category of models treats features as an accumulated view over a historical window, and thus ignores crucial patterns that evolve over time.

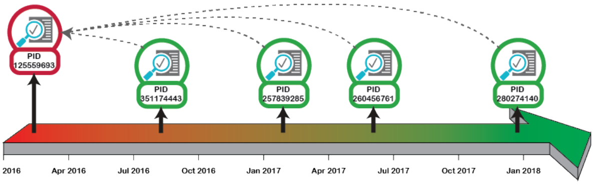

Figure 1: The citation chain for "The Essence of Wildlife Management" from the MAG dataset. The first 4 citations are shown along the timeline with their MAG ID attached by vertical line.

To address the above issues, point process based citation prediction models (Lee et al. 2012; Jang, Woo, and Lee 2017; Liu et al. 2017) have drawn growing attention in recent years. As shown in Figure 1, the sequence of citations that reference a given paper is naturally a time series. Consequently, it can be modeled as a temporal point process that modulates the temporal pattern in a series of points. In theory, the temporal point process is characterized by a conditional intensity function learned from observing points along the timeline. Conventional methods concen-

[1] https://github.com/TaoranJ/DMA-Nets

trate on designing a specific parametric form of the intensity function using heuristic assumptions specific to their application (Mishra, Rizoiu, and Xie 2016; Helmstetter and Sornette 2002). For instance, citation forecasting methods (Xiao et al. 2016; Liu et al. 2017) usually follow the paradigm of the general self-exciting process (Hawkes 1971) in which intensity spikes whenever a new citation arrives. This feature is used to simulate that a highly cited paper is more likely to receive more citations. These conventional methods have two notable drawbacks: (1) heuristic assumptions might not be able to reflect complicated temporal dependencies in real datasets; and, (2) in practice, the complexity of the intensity function is mathematically limited.

To address the challenges conventional models have in modeling intensity, more recent approaches use recurrent neural networks (RNNs) (Du et al. 2016; Mei and Eisner 2017) to approximate more complicated conditional intensity functions without heuristic assumptions or prior knowledge of dataset or application. Most existing RNN-based models (Wang et al. 2017; Xiao et al. 2019, 2017) have shown improved performance over conventional methods on both synthetic data and real-world datasets. RNN-based temporal point process models can be classified into two families: intensity-based models and end-to-end models. Intensity-based models (Wang et al. 2017; Du et al. 2016) use the neural network to implicitly modulate the conditional intensity function which is then used to obtain the conditional density function for maximum likelihood estimation and prediction. This group of models is optimized for observation history but is prone to forecasting error propagation for the task of long-term prediction. The family of end-to-end models (Ji et al. 2019) combines the process of representing the intensity function with the process of prediction. The advantage of end-to-end models is that, with careful design, the model can be further optimized during the predication phase, instead of only from the observation sequence.

In this paper, we propose an RNN-based, end-to-end model for citation forecasting. This model introduces a hierarchical dynamic attention layer which uses two temporal attention mechanisms to enforce the model's ability to represent complicated conditional dependencies in real-world datasets and allow the model to automatically balance the learning process from the observation side and prediction side. Furthermore, the temporal prediction layer guarantees that the predicted citations are monotonically increasing along the time dimension. Specifically, the contributions and highlights of this paper are:

- Formulating a novel framework to provide long-term citation predictions in an end-to-end fashion by integrating the process of learning intensity function representations and the process of predicting future citations.
- Designing two novel temporal attention mechanisms to improve the model's ability to modulate complicated temporal dependencies and to allow the model to dynamically combine the observation and prediction sides during the learning process.
- Conducting extensive experiments on two real-world datasets to demonstrate that our model is capable of capturing the general shape of citation sequences and can consistently outperform other models for the citation forecasting task.
- Curating and releasing two large datasets from the United States Patent and Trademark Office (USPTO) and Microsoft Academic Graph (MAG), which can be used for citation prediction task and generalized point process task.

Problem Formulation

Let $\mathcal{C} = \{\mathcal{C}_i\}_{i=1}^{|\mathcal{C}|}$ be a set of collected citation sequences for scientific documents (e.g. a set of papers or patents). The ith sequence is denoted by $\mathcal{C}_i = \{(t_i, m_i)\}_{i=0}^{|\mathcal{C}_i|}$ where t_i and m_i refer to the published date and the technology class of the ith citation, and the 0th citation is the target document itself. The citation sequence can also be represented in terms of the inter-citation duration between two successive citations $\mathcal{C}_i = \{(\tau_i, m_i)\}_{i=1}^{|\mathcal{C}_i|}$ where $\tau_i = t_i - t_{i-1}$ refers to the time difference between the ith citation and the $(i-1)$th citation. These two representations are equivalent. In this paper, we use inter-citation duration notation because it makes it easier to constrain the end-to-end model to forecast citations correctly along the time dimension such that $t_{i+1} \geq t_i$.

Given data as described above, our problem is as follows: for a scientific document, using the first l citations $\{(\tau_i, m_i)\}_{i=1}^{l}$ as observations, can we forecast the sequence of the next n citations $\{(\tau_j, m_j)\}_{j=l+1}^{l+n}$? When $n = 1$, the problem is a one-step forecasting problem, which is simpler since learning the temporal point process depends only on the observation side and there is no error propagation on the prediction side. Because predicting only the next citation does not have much practical value, we focus only on the task when $n > 1$. In the case where $n > 1$, there are two challenges for the task of forecasting the next n citations. First, there is a trade-off of learning from the observation side or from the prediction side. On the one hand, observations are ground truth but they may be too few to provide enough information to modulate the temporal point process. On the other hand, predictions are less trustworthy but can provide extra information to the model for learning the temporal point process. Also, errors that occur early in the predication phase can be propagated into subsequent predictions. These challenges motivate us to adopt a sequence-to-sequence structure which takes into account both the observation side and prediction side during the training.

Models

By considering the arrival of a citation as an instant point on the timeline, we can study the entire citation sequence as a point process whose joint density function is represented as

$$f((\tau_1, m_1), (\tau_2, m_2), \ldots) = \prod_i f\left(\tau_i, m_i | \mathcal{H}_i^*\right) \qquad (1)$$

where the density function at ith step is conditioned by the information of historical citations up to point i, denoted by $\mathcal{H}_i^*$. In point process theory, this density function is usually

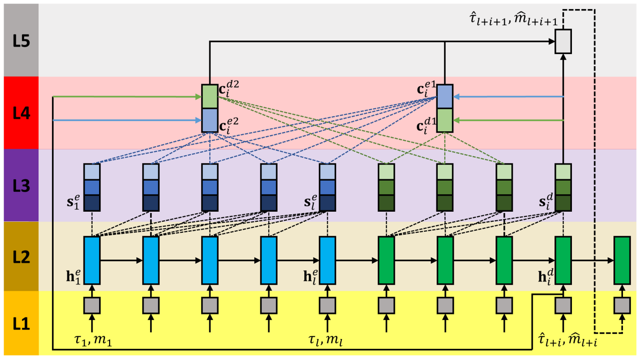

Figure 2: The architecture of DMA-Nets. L1 is the input layer. L2 is the recurrent representation layer. L3 refers to the local temporal attention (LTA) layer and L4 to the global multi-context temporal attention (GMTA) layer. Together, these comprise the dynamic hierarchical attention layer. L5 is the prediction layer.

learned through the conditional intensity function, which is used to predict future citations through a generative process. In this paper, we propose a novel framework that integrates the task of representing the conditional intensity function and predicting the arrival time and document class of the next n citations.

Figure 2 presents the overall encoder-decoder architecture of DMA-Nets where the encoder is supplied with the sequence of observed citations $\zeta_e = \{(\tau_i, m_i)\}_{i=1}^{l}$ and the decoder aims to recurrently predict the sequence of the next n citations $\zeta_d = \{(\hat{\tau}_j, \hat{m}_j)\}_{j=l+1}^{l+n}$. The input layer (L1 in Fig. 2) encodes temporal and category information into dense vectors. The recurrent representation layer (L2 in Fig. 2) captures the hidden dependencies of the current citations over all previous citations. The learned representations enter the attention layer (L3 and L4 in Fig. 2) which consists of two modules. The local temporal attention layer compiles the history dependences between each pair of historical citations and generates intra-encoder states and intra-decoder states. Next, on the decoder side, the global temporal attention layer fuses multiple contexts obtained by attending to different queries on the information embedded by the inner states of both encoder and decoder. Finally, the prediction layer (L5 in Fig 2) makes the time-aware prediction for the next n citations.

Seq2Seq Structure for Citation Prediction

Given the observation sequence ζ_e, at each step, the encoder aims to encode and to compile the hidden dependencies across observed historical citations, thus generating a sequence of hidden states $\mathbf{h}^e = \{\mathbf{h}_1^e, \ldots, \mathbf{h}_l^e\}, \mathbf{h}_i^e \in \mathbb{R}^{d_h}$. The calculation of the ith hidden state $\mathbf{h}_i^e$ is defined in Equation 2:

$$\mathbf{h}_i^e = g(\mathbf{a}_i^e, \mathbf{h}_{i-1}^e), \mathbf{a}_i^e = f_{emb}(\tau_i, m_i) \tag{2}$$

where g is a recurrent unit, such as LSTM (Hochreiter and Schmidhuber 1997), GRU (Cho et al. 2014), or vanilla RNN, which captures the dependency structure of the current input over the hidden state at the previous steps, and f_{emb} is the embedding layer concatenating the embedding of τ_i and m_i to a d_{emb}-dimension dense vector. In particular, for the ith input (τ_i, m_i), τ_i is first discretized on year, month, and day, and then is embedded into R^{d_τ} space, and m_i is embedded into a $\mathbb{R}^{d_m}$ space. Likewise, at each step, the decoder takes as input the previous hidden state and the prediction from the previous step

$$\mathbf{h}_i^d = g(\mathbf{a}_i^d, \mathbf{h}_{i-1}^d), \mathbf{a}_i^d = f_{emb}(\hat{\tau}_{l+i}, \hat{m}_{l+i}), \tag{3}$$

and predicts the waiting time and the document class of next citation:

$$\hat{\tau}_{l+i+1}, \hat{m}_{l+i+1} = p(\mathbf{h}_i^d), \tag{4}$$

where $p(\cdot)$ is a function that predicts the next citation based on the current hidden state. In this work we use an LSTM recurrent unit and employ $d_\tau = d_m = 32$, $d_{emb} = d_\tau + d_m = 64$ and $d_h = 256$.

Hierarchical Dynamic Attention Layer

Though recurrent neural networks have been successfully used in various time series prediction tasks (Du et al. 2016), the fact that the last hidden state holds the entire memory of the sequence poses a bottleneck in learning conditional dependencies across a long sequence of temporal points. As a result, we propose a hierarchical dynamic attention layer that explores pairwise temporal dependencies from both local and global perspectives and from the viewpoint of both observations and predictions.

Local Temporal Attention (LTA) Layer In this layer, we propose a local temporal attention mechanism to enhance the modulation of conditional dependencies by allowing the model to access and directly attend to previous hidden states. We illustrate the local temporal attention mechanism on the encoder. The decoder follows a similar process. Let $\mathbf{h}_i^e$ be the current hidden state of the encoder and $\mathcal{H}_i^e = [\mathbf{h}_1^e; \ldots; \mathbf{h}_i^e] \in \mathbb{R}^{d_h \times i}$ be the i previous hidden states available along the time dimension. The local temporal attention mechanism aims to generate a corresponding intra-encoder attentional hidden state $\mathbf{s}_i^e$ for hidden state $\mathbf{h}_i^e$

$$\mathbf{s}_i^e = \text{LTA}(\mathbf{h}_i^e, \mathcal{H}_i^e).$$

To further enhance the model's flexibility in representing conditional temporal dependencies, we use multiple *heads* (Vaswani et al. 2017) to calculate attentional hidden states in different semantic subspaces and concatenate all the results together as the final $\mathbf{s}_i^e$. The calculation for the kth head is defined as

$$\begin{aligned} \mathbf{s}_{i,k}^e &= \text{LTA}_k(\mathbf{W}_k^1 \mathbf{h}_i^e, \mathbf{W}_k^2 \mathcal{H}_i^e, \mathbf{W}_k^3 \mathcal{H}_i^e) \\ &= \text{LTA}_k(\tilde{\mathbf{h}}_i^e, \bar{\mathcal{H}}_i^e, \tilde{\mathcal{H}}_i^e) \\ &= \sum_j^i w_{ij} \tilde{\mathcal{H}}_{i,j}^e = \sum_j^i \frac{\exp(e_{ij})}{\sum_k^i \exp(e_{ik})} \tilde{\mathcal{H}}_{i,j}^e, \end{aligned} \tag{5}$$

where $\mathbf{s}_{i,k}^e \in \mathbb{R}^{d_q}$ is an attentional hidden state for head k, and $\mathbf{W}_k^1$, $\mathbf{W}_k^2$, $\mathbf{W}_k^3$ are three learnable $d_q \times d_h$ matrices which project $\mathbf{h}_i^e$, $\mathcal{H}_i^e$ into three different subspaces $\boldsymbol{\hbar}_i^e \in \mathbb{R}^{d_q}$, $\bar{\mathcal{H}}_i^e \in \mathbb{R}^{d_q \times i}$, and $\tilde{\mathcal{H}}_i^e \in \mathbb{R}^{d_q \times i}$, w_{ij} is normalized e_{ij} measuring the amount of attention $\boldsymbol{\hbar}_i^e$ should pay to $\tilde{\mathcal{H}}_{i,j}^e$ (the jth column of $\tilde{\mathcal{H}}_i^e$), and e_{ij} is calculated by the following score function

$$e_{ij} = \frac{\left(\text{Sigmoid}(\mathbf{W}^e \boldsymbol{\hbar}_i^e)\right)^{\mathrm{T}} \tilde{\mathcal{H}}_{i,j}^e}{\sqrt{d_q}},$$

where $\mathbf{W}^e \in \mathbb{R}^{d_q \times d_q}$ is a learnable square matrix. Different from the vanilla dot-product score function, a non-linear projection of $\boldsymbol{\hbar}_i^e$ is used to avoid biased attention towards its neighbor hidden states (e.g. $\boldsymbol{\hbar}_i^e, \boldsymbol{\hbar}_{i-1}^e$). Also, the score is scaled to avoid values of large magnitude (Vaswani et al. 2017). Finally, $\mathbf{s}_i^e$ is obtained by concatenating $\mathbf{s}_{i,k}^e$ for each head:

$$\mathbf{s}_i^e = \text{LTA}(\mathbf{h}_i^e, \mathcal{H}_i^e) = \text{concat}(\mathbf{s}_{i,1}^e, \ldots, \mathbf{s}_{i,h}^e)\mathbf{W}^o, \tag{6}$$

where h is the number of heads used, $\mathbf{W}^o \in \mathbb{R}^{hd_q \times d_h}$ transforms the hd_q-dimension result back to d_h-dimension space. Likewise, the intra-decoder attentional hidden state $\mathbf{s}_i^d$ can be obtained by $\mathbf{s}_i^d = \text{LTA}(\mathbf{h}_i^d, \mathcal{H}_i^d)$, where $\mathcal{H}_i^d = [\mathbf{h}_1^d; \ldots; \mathbf{h}_i^d]$ refers to all currently available hidden states on the decoder. For this paper, we employ $h = 4$ and $d_q = 64$.

Global Multi-Context Temporal Attention (GMTA) Layer On top of the local temporal attention layer, we propose a global multi-context temporal attention mechanism with the several considerations in mind. First, the approach should allow the model to continue examining conditional temporal dependencies in the decoder phase. That is, in contrast to the traditional attention strategy (Luong, Pham, and Manning 2015) which attends only to encoder states, the proposed approach should consider the attentional hidden states on both sides. Second, the approach should let the model dynamically determine the combination of information from the encoder and decoder sides. Third, instead of learning attention weights based only on the state value, we argue that the temporal pattern of the temporal point process in the input should also be a decisive factor.

Here, we illustrate the computation process of attentional contexts on the encoder side. At the ith step of the decoder, let $\mathbf{s}_i^d$ be the decoder's current attentional hidden state. Let $\mathbf{S}^e = [\mathbf{s}_1^e; \ldots; \mathbf{s}_l^e]$ be the encoder's l attentional hidden states and $\mathcal{A}^e = [\mathbf{a}_1^e; \ldots; \mathbf{a}_l^e]$ be the encoder's l inputs. Two contexts, $\mathbf{c}_i^{e1}$ and $\mathbf{c}_i^{e2}$, are calculated. Again, we employ the multi-head strategy to calculate $\mathbf{c}_i^{e1}$ and $\mathbf{c}_i^{e2}$ in different semantic subspaces and use the concatenation for the final context. For the k-th head, both contexts are a weighted sum of projected $\mathbf{S}^e$. The difference is that for $\mathbf{c}_{i,k}^{e1}$ the attention weight e_{ij}^{e1} depends on the value of the attentional hidden states $\mathbf{s}_i^d$ and $\mathbf{s}_j^e$ while for $\mathbf{c}_{i,k}^{e2}$ the temporal pattern inputs $\mathbf{a}_i^d$ and $\mathbf{a}_j^e$ determine the attention weights e_{ij}^{e2}, that is

$$e_{ij}^{e1} = \frac{\left(\mathbf{Q}_k^{e1}\mathbf{s}_i^d\right)^{\mathrm{T}} \mathbf{V}_k^{e1}\mathbf{s}_j^e}{\sqrt{d_{c1}}}, \quad e_{ij}^{e2} = \frac{\left(\mathbf{Q}_k^{e2}\mathbf{a}_i^d\right)^{\mathrm{T}} \mathbf{V}_k^{e2}\mathbf{a}_j^e}{\sqrt{d_{c2}}}, \tag{7}$$

where $\mathbf{Q}_k^{e1}, \mathbf{V}_k^{e1} \in \mathbb{R}^{d_{c1} \times d_h}$ and $\mathbf{Q}_k^{e2}, \mathbf{V}_k^{e2} \in \mathbb{R}^{d_{c2} \times d_h}$ are learnable matrices for linear projection. Then we have $\mathbf{c}_{i,k}^{e1}$ and $\mathbf{c}_{i,k}^{e2}$ calculated as:

$$\mathbf{c}_{i,k}^{e1} = \sum_j^l \frac{\exp(e_{ij}^{e1})}{\sum_k^l \exp(e_{ik}^{e1})} \mathbf{U}_k^{e1}\mathbf{s}_j^e, \quad \mathbf{c}_{i,k}^{e2} = \sum_j^l \frac{\exp(e_{ij}^{e2})}{\sum_k^l \exp(e_{ik}^{e2})} \mathbf{U}_k^{e2}\mathbf{s}_j^e, \tag{8}$$

where $\mathbf{U}_k^{e1} \in \mathbb{R}^{d_{c1} \times d_h}$ and $\mathbf{U}_k^{e2} \in \mathbb{R}^{d_{c2} \times d_h}$. Finally, contexts $\mathbf{c}_i^{e1}$ and $\mathbf{c}_i^{e2}$ are obtained by concatenating the results of all heads:

$$\begin{aligned}
\mathbf{c}_i^{e1} &= \text{GMTA}^{\mathbf{s}}(\mathbf{s}_i^d, \mathbf{S}^e) = \text{concat}(\mathbf{c}_{i,1}^{e1}, \ldots, \mathbf{c}_{i,m1}^{e2})\mathbf{W}^{e1}, \\
\mathbf{c}_i^{e2} &= \text{GMTA}^{\tau}(\mathbf{a}_i^d, \mathcal{A}^e, \mathbf{S}^e) = \text{concat}(\mathbf{c}_{i,1}^{e2}, \ldots, \mathbf{c}_{i,m2}^{e2})\mathbf{W}^{e2},
\end{aligned}$$

where m_1 and m_2 are the number of heads to use for $\mathbf{c}_i^{e1}$ and $\mathbf{c}_i^{e2}$, respectively, and both $\mathbf{W}^{e1} \in \mathbb{R}^{hd_{c1} \times d_h}$ and $\mathbf{W}^{e2} \in \mathbb{R}^{hd_{c2} \times d_h}$ are learnable projection matrices. Likewise, let $\mathbf{S}_i^d$ represent all of the decoder's previous attentional hidden states and $\mathcal{A}_i^d$ all the decoder's previous inputs. At the ith step, the decoder context $\mathbf{c}_i^{d1}$ and $\mathbf{c}_i^{d2}$ can be calculated by $\text{GMTA}^{\mathbf{s}}(\mathbf{s}_i^d, \mathbf{S}_i^d)$ and $\text{GMTA}^{\tau}(\mathbf{a}_i^d, \mathcal{A}_i^d, \mathbf{S}_i^d)$, respectively. In this work we employ $m_1 = 4$, $d_{c1} = 64$, $m_2 = 2$ and $d_{c2} = 32$.

Prediction Layer

The citation sequence has an implicit constraint that future citations always come after the most recent citation, that is, the predicted inter-citation duration should always be non-negative. Some previous work (Xiao et al. 2016) ignored this constraint. With this in mind, we design the prediction layer as below:

$$\begin{aligned}
&\mathbf{c}_i = \text{concat}(\mathbf{c}_i^{e1}, \mathbf{c}_i^{e2}, \mathbf{c}_i^{d1}, \mathbf{c}_i^{d2}, \mathbf{s}_i^d)\mathbf{W}^c, \\
&\hat{\tau}_{i+1} = \text{Softplus}(\mathbf{W}^{out}\mathbf{c}_i), \hat{m}_{i+1} = \text{Softmax}(\mathbf{c}_i)
\end{aligned} \tag{9}$$

where $\mathbf{W}^c \in \mathbb{R}^{5d_h \times d_h}$ and $\mathbf{W}^{out} \in \mathbb{R}^{1 \times d^h}$. Note that the prediction is constrained by the Softplus function to enforce the non-negative requirement.

Parameter Learning

The total loss is the sum of the time prediction loss and the cross-entry loss for document category prediction:

$$\sum_{i=l+1}^{l+n} \left(\left|\hat{t}_i - t_i\right| - \log(m_i)\right) = \sum_{i=l+1}^{l+n} \left|\sum_{j=l+1}^{i} (\hat{\tau}_j - \tau_j)\right| - \log(m_i).$$

Note that as the model is predicting inter-citation period instead of the citation time, and thus the temporal loss above is not the absolute pairwise difference between predictions and ground truth.

For regularization, we use dropout to the output of each sublayer with a dropout rate of 0.1. For optimization, we adopted the ADAM (Kingma and Ba 2014) optimizer for training with learning rate set to 0.0001 and weight decay of 0.0001.

USPTO									
Model	80% As Observations			50% As Observations			30% As Observations		
	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
RMTPP	83.0190	138.2183	0.4536	182.5732	271.2538	0.5866	258.9615	370.6494	0.5858
CYAN-RNN	71.8609	124.2318	0.8443	142.6116	221.6332	0.8421	191.9568	283.4559	**0.8431**
RPP	109.3605	175.8235	0.8206	234.2301	351.2669	0.7232	309.0723	429.7102	0.6837
Seq2Seq	62.0010	108.9615	0.7420	110.4464	176.9894	0.6816	148.8295	223.1447	0.5866
DotSeq2Seq	60.8083	105.7521	0.8191	106.9324	165.6998	0.7372	143.9769	215.8835	0.7561
PC-RNN	56.8930	100.3806	0.8121	**98.8260**	**156.5963**	0.7579	132.2665	199.2807	0.7739
DMA-Nets	**56.8676**	**98.3817**	**0.8570**	100.8460	157.6145	**0.8513**	**131.0907**	**196.7500**	0.8444

MAG									
Model	80% As Observations			50% As Observations			30% As Observations		
	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
RMTPP	26.2305	38.5251	0.2923	56.1802	85.0525	0.2073	85.8301	118.5709	0.3274
CYAN-RNN	20.3441	30.7344	0.6516	39.8326	56.7323	0.6509	71.8483	98.4990	0.6390
RPP	33.1357	48.2855	0.6658	76.5505	107.7528	0.6087	155.2278	234.0893	0.6027
Seq2Seq	21.6710	30.4907	0.6674	41.5486	57.4757	0.6397	59.8124	78.7138	**0.6487**
DotSeq2Seq	17.5930	25.9695	0.6320	32.4100	46.0663	0.5900	47.0133	63.1645	0.4987
PC-RNN	20.9865	30.4462	**0.6720**	38.3954	54.2841	0.6218	56.6671	73.7164	0.6167
DMA-Nets	**17.2212**	**25.5530**	0.6647	**28.8310**	**41.7167**	**0.6547**	**42.2267**	**58.3853**	0.6425

Table 1: Performance evaluation of our method (DMA-Nets) and peer methods. Timestamp predictions in days are evaluated using MAE and RMSE and document category predictions are evaluated using accuracy.

Experiments

We compare our DMA-Nets experimentally to state-of-the-art methods on two large real-world datasets compiled from the United States Patent and Trademark Office (USPTO) and the Microsoft Academic Graph (MAG).

Dataset Description and Experiment Setup

Dataset: USPTO is a premier patent database documenting U.S. patents. We adopted the patent citation collection published in (Ji et al. 2019), which consists of 15,000 sequences with a length within 20 to 100. Each patent has a granted date and a category. Furthermore, to better validate the model's capacity, we extend the original dataset to 25,000 sequences following the same procedure. MAG (Sinha et al. 2015) is a paper database maintained by Microsoft. For each paper, we construct a citation chain using its publish date and category from the database. We likewise remove papers with chains shorter than 20 and trim chains longer than 100, and then sample 25,000 sequences for the experiment. For both dataset, we used 17,500 sequences as the training set, 5,000 sequences as the test set, and the remaining 2,500 sequences as the validation set. All the datasets used in the paper are available for download[2].

Metrics: Following similar procedures in (Du et al. 2016; Wang et al. 2017; Xiao et al. 2017; Ji et al. 2019), we use *mean absolute error* (MAE) and *root mean squared error* (RMSE) as evaluation metrics for citation time predictions, and *accuracy* for document category prediction.

Compared Baselines: We compare DMA-Nets with state-of-the-art point process baselines including two intensity-based models and four end-to-end based models:

- **RMTPP (Du et al. 2016)**: RMTPP uses recurrent units to learn the intensity function for general point process analysis and is able to predict point arrival time and type in a sequence.
- **CYAN-RNN (Wang et al. 2017)**: CYAN-RNN uses GRU-based recurrent units and attention mechanisms to learn the intensity function for a general information resharing process and can forecast the arrival time and type of next resharing behavior.
- **RPP (Xiao et al. 2017)**: RPP is similar to RMTPP, but it uses a fully connected layer to map the embedded hidden state directly to time and type predictions.
- **Seq2Seq (Sutskever, Vinyals, and Le 2014)**: Seq2Seq represents a traditional sequence-to-sequence model which consists of one recurrent representation layer.
- **DotSeq2Seq (Luong, Pham, and Manning 2015)**: DotSeq2Seq represents a Seq2Seq model with a traditional static attention mechanism. We used a dot-product score function in the experiments.
- **PC-RNN (Ji et al. 2019)**: PC-RNN is an end-to-end point process model for patent citation forecasting which is able to integrate multiple observation sequences and have a static attention mechanism equipped on the prediction side. On the USPTO dataset, we used three sequences of patent citations, assignee citations, and inventor citations.

[2] https://github.com/TaoranJ/DMA-Nets

				USPTO					
Model	80% As Observations			50% As Observations			30% As Observations		
	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
DMA-Nets$_g$	58.8086	100.1425	0.8496	104.2577	163.2522	0.8303	134.8752	200.0412	0.8401
DMA-Nets$_l$	57.9564	102.1665	0.8539	102.5050	158.4229	0.8480	134.2213	201.2891	**0.8455**
DMA-Nets	**56.8676**	**98.3817**	**0.8570**	**100.8460**	**157.6145**	**0.8513**	**131.0907**	**196.7500**	0.8444
				MAG					
Model	80% As Observations			50% As Observations			30% As Observations		
	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
DMA-Nets$_g$	17.4309	25.6951	0.6473	30.1178	42.6512	0.6347	44.6062	60.8219	0.6271
DMA-Nets$_l$	17.5243	26.0156	**0.6702**	30.3576	43.4257	0.6484	44.0911	59.9982	0.6264
DMA-Nets	**17.2212**	**25.5530**	0.6647	**28.8310**	**41.7167**	**0.6547**	**42.2267**	**58.3853**	**0.6425**

Table 2: Performance evaluation of variants of DMA-Nets.

On the MAG dataset, the observation side has only paper citation sequences available.

Performance Comparison

By restricting the length of the observation window to 30%, 50%, and 80% of the entire sequence, we conducted three groups of experiments to test each model's performance on the USPTO dataset and the MAG dataset. In real-world applications, short observation windows are preferred over long observation windows. All results are reported in Table 1.

Our model consistently outperforms RMTPP, CYAN-RNN, and RPP for time prediction and category prediction in all experiments. RMTPP, CYAN-RNN, and RPP deliver long-term prediction in one-step forecasting fashion. That is, they modulate the citation process based only on the observations and then use it as the generator during the prediction process by directing the ith prediction to the $(i + 1)$th input. In the introduction section, we argued that this type of model has two major shortcomings: 1) their capacity for modulating the point process will diminish as the number of available observations decreases, and 2) they are prone to the accumulated time prediction error as the number of predictions made increases. Our observation in Table 1 reinforces these assumptions. As the observation window shrinks from 80% to 30%, against the best of these three models, our model's MAE improvement increases from 20.86% to 31.71% on the USPTO dataset and from 15.35% to 41.23% on the MAG dataset. Similar results are also observed for the RMSE metric. This observation also demonstrates that our model has a better capacity to modulate the citation process with information from the prediction side and mitigate error propagation in the time prediction. In terms of document class prediction, we observed that our model significantly outperforms RMTPP and RPP and is competitive with CYAN-RNN. We argue this is because both our model and CYAN-RNN are empowered by the attention mechanism, which allows the model to look back at previous document categories during the prediction process.

On both the USPTO and the MAG dataset, our model generally performs better than Seq2Seq, DotSeq2Seq, and PC-RNN. These three models all practice the seq2seq structure while the Seq2Seq model modulates historical dependencies via the recurrent layer and the other two models adopt both the recurrent and static attention layers to model conditional dependencies. We gained several insights by observing performance results in Table 1. First, we observed that seq2seq based models are better at the time prediction task, which again strengthens our hypothesis that the long-term prediction is a benefit of the seq2seq structure. Second, as the observation window shrinks, our model's improvement in MAE and RMSE gradually increases in most cases. Intuitively, with fewer observations available, forecasting performance relies more heavily on the prediction phase. So we argue that, unlike other models which can only rely on the current hidden state or encoder's history states, our model has increased performance because the LTA and GMTA layers allow our model to benefit from modeling historical dependencies on the prediction side (i.e., the area with green blocks in Fig. 2). Note that even in the USPTO dataset, where PC-RNN uses three sequences (patent, assignee, and inventor citation sequences) as input on the encoder side, our model's performance is still competitive. Third, we observed that on the MAG dataset, the Seq2Seq and DotSeq2Seq model could only achieve competing results on either the time prediction tasks or the category predictions task. In contrast, our model's performance on both time prediction and category prediction is competitive and balanced. We argue this is attributed to the flexibility and expression capability empowered by the hierarchical dynamic attention layer.

Ablation Study

Global Multi-Context Temporal Attention (GMTA) Layer Analysis We first analyze the contributions of the global multi-context temporal attention layer (GMTA). In this ablation test, we remove the GMTA layer from DMA-Nets and create one variant, named DMA-Nets$_g$. For DMA-Nets$_g$, at each step of the decoder, we drop the calculation

	Hyper-parameters									USPTO		MAG	
	d_{emb}	d_h	h	d_q	m_1	d_{c1}	m_2	d_{c2}	dropout	MAE	ACC	MAE	ACC
base	64	256	4	64	4	64	2	32	0.1	56.8676	**0.8570**	17.2212	0.6647
d_h	64	128	4	32	4	32	2	32	0.1	58.3457	0.8516	17.2143	0.6656
	64	64	4	16	4	16	2	32	0.1	58.3930	0.8373	17.3194	0.6576
d_{emb}	32	256	4	64	4	64	2	16	0.1	57.3704	0.8512	17.1048	0.6681
	16	256	4	64	4	64	2	8	0.1	57.7617	0.8536	17.1164	0.6561
h, m_1	64	256	8	32	8	32	2	32	0.1	57.5771	0.8521	**17.0623**	0.6641
	64	256	2	128	2	128	2	32	0.1	**56.8441**	0.8555	17.1831	**0.6687**
m_2	64	256	4	64	4	64	4	16	0.1	57.5239	0.8521	17.0971	0.6675
dropout	64	256	4	64	4	64	2	32	0.3	57.4690	0.8536	17.1240	0.6660

Table 3: Hyper-parameter analysis for DMA-Nets.

of the encoder's contexts $\mathbf{c}_i^{e1}$ and $\mathbf{c}_i^{e2}$ and the decoder's contexts $\mathbf{c}_i^{d1}$ and $\mathbf{c}_i^{d2}$ (L4 in Fig. 2) and instead use only the current attentional hidden state $\mathbf{s}_i^d$ as the input for the prediction layer. Consequently, the calculation of the encoder's attentional hidden states $[\mathbf{s}_1^e; \ldots; \mathbf{s}_l^e]$ is also removed. In this variant, decoder's states $\mathbf{h}_i^d$ and $\mathbf{s}_i^d$ carry the burden of holding information of previous states. The performance of DMA-Nets$_g$ is reported in Table 2. As expected, DMA-Nets outperforms DMA-Nets$_g$. Intuitively, the GMTA layer is most beneficial in cases where the model relies more on observations. This is because the GMTA layer provides a global view of citation sequences and captures the dynamics on both the observation side and the prediction side. When the GMTA layer is missing, the prediction side dynamics can be carried by both the recurrent unit and the local attentional state but the historical observations can be encoded only by the RNN backbone. On the USPTO dataset, we observed that the performance gain brought by the GMTA layer is most significant when 80% of sequence used as observations. On the MAG dataset, GMTA layer is most beneficial when 50% of sequence used as observations.

Local Temporal Attention (LTA) Layer Analysis Next, we analyze the contributions of the local temporal attention layer (LTA). We created an ablation named DMA-Nets$_l$ by removing the local temporal attention layer from DMA-Nets. As a result, on both observation and prediction side, instead of calculating local attentional hidden states $\{\mathbf{s}_i^e\}_{i=1}^l$ and $\{\mathbf{s}_i^d\}_{i=1}^n$ we used the corresponding RNN hidden states $\{\mathbf{h}_i^e\}_{i=1}^l$ and $\{\mathbf{h}_i^d\}_{i=1}^n$ as the input for the subsequent global multi-context temporal attention (GMTA) layer. The performance of DMA-Nets$_l$ is also reported in Table 2. The fully fledged DMA-Nets outperforms DMA-Nets$_l$ on both datasets which indicates that the LTA layer improves model performance.

Hyper-parameter Analysis

We investigate the sensitivity of d_h, d_{emb}, h, m_1, m_2, and dropout rate and report the performance of DMA-Nets in MAE and ACC metrics on the 80% observation setting in Table 3. We observe that, in general, DMA-Nets is robust to different hyper-parameter settings. On the MAG dataset, the performance variation through different settings is insignificant. On the USPTO dataset, it is observed that reducing the model size d_h will decrease the model's performance on time prediction task.

Conclusion

In this paper, we present a novel framework for forecasting citations of scientific publications. On top of the seq2seq architecture, this model constructs a hierarchical dynamic attention layer which considers the citation process from both local and global perspectives and from the viewpoint of both observations and predictions. To enable the model to represent interconnected dependencies across observation sequences and prediction sequences, we employ a local temporal attention mechanism to allow the model to look back along the temporal dimension and fuse more complicated intra-encoder and intra-decoder hidden attentional states. Additionally, the global multi-context attention layer encourages the model to learn the temporal point process from a global viewpoint by considering not only observations but also the predictions that have already been made. We demonstrate the performance improvement of our model on two real-world datasets collected from USPTO and MAG. Experimental results demonstrate that our model can consistently outperform state-of-the-art point process modeling methods for the task of citation forecasting.

References

Acuna, D. E.; Allesina, S.; and Kording, K. P. 2012. Future impact: Predicting scientific success. *Nature* 489(7415): 201.

Bessen, J. 2008. The value of US patents by owner and patent characteristics. *Research Policy* 37(5): 932–945.

Cho, K.; Van Merriënboer, B.; Bahdanau, D.; and Bengio, Y. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259* .

Du, N.; Dai, H.; Trivedi, R.; Upadhyay, U.; Gomez-Rodriguez, M.; and Song, L. 2016. Recurrent marked temporal point processes: Embedding event history to vector. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1555–1564. ACM.

Egghe, L. 2006. Theory and practise of the g-index. *Scientometrics* 69(1): 131–152.

Hawkes, A. G. 1971. Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58(1): 83–90.

Helmstetter, A.; and Sornette, D. 2002. Subcritical and supercritical regimes in epidemic models of earthquake aftershocks. *Journal of Geophysical Research: Solid Earth* 107(B10): ESE–10.

Hirsch, J. E. 2005. An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences* 102(46): 16569–16572.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8): 1735–1780.

Jang, H. J.; Woo, H.-G.; and Lee, C. 2017. Hawkes process-based technology impact analysis. *Journal of Informetrics* 11(2): 511–529.

Ji, T.; Chen, Z.; Self, N.; Fu, K.; Lu, C.; and Ramakrishnan, N. 2019. Patent Citation Dynamics Modeling via Multi-Attention Recurrent Networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, 2621–2627. doi:10.24963/ijcai.2019/364. URL https://doi.org/10.24963/ijcai.2019/364.

Kim, J.; Diesner, J.; Kim, H.; Aleyasen, A.; and Kim, H.-M. 2014. Why name ambiguity resolution matters for scholarly big data research. In *2014 IEEE International Conference on Big Data (Big Data)*, 1–6. IEEE.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Lee, C.; Cho, Y.; Seol, H.; and Park, Y. 2012. A stochastic patent citation analysis approach to assessing future technological impacts. *Technological Forecasting and Social Change* 79(1): 16–29.

Lee, C.; Kwon, O.; Kim, M.; and Kwon, D. 2018. Early identification of emerging technologies: A machine learning approach using multiple patent indicators. *Technological Forecasting and Social Change* 127: 291–303.

Liu, X.; Yan, J.; Xiao, S.; Wang, X.; Zha, H.; and Chu, S. M. 2017. On predictive patent valuation: Forecasting patent citations and their types. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Luong, M.-T.; Pham, H.; and Manning, C. D. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025* .

Mei, H.; and Eisner, J. M. 2017. The neural hawkes process: A neurally self-modulating multivariate point process. In *Advances in Neural Information Processing Systems*, 6754–6764.

Mishra, S.; Rizoiu, M.-A.; and Xie, L. 2016. Feature driven and point process approaches for popularity prediction. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 1069–1078. ACM.

Sinha, A.; Shen, Z.; Song, Y.; Ma, H.; Eide, D.; Hsu, B.-j. P.; and Wang, K. 2015. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, 243–246. ACM.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 3104–3112.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wang, Y.; Shen, H.; Liu, S.; Gao, J.; and Cheng, X. 2017. Cascade dynamics modeling with attention-based recurrent neural network. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2985–2991. AAAI Press.

Xiao, S.; Yan, J.; Farajtabar, M.; Song, L.; Yang, X.; and Zha, H. 2019. Learning time series associated event sequences with recurrent point process networks. *IEEE transactions on neural networks and learning systems* .

Xiao, S.; Yan, J.; Li, C.; Jin, B.; Wang, X.; Yang, X.; Chu, S. M.; and Zha, H. 2016. On Modeling and Predicting Individual Paper Citation Count over Time. In *IJCAI*, 2676–2682.

Xiao, S.; Yan, J.; Yang, X.; Zha, H.; and Chu, S. M. 2017. Modeling the intensity function of point process via recurrent neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Yan, R.; Huang, C.; Tang, J.; Zhang, Y.; and Li, X. 2012. To better stand on the shoulder of giants. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, 51–60.

Yan, R.; Tang, J.; Liu, X.; Shan, D.; and Li, X. 2011. Citation count prediction: learning to estimate future citations for literature. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, 1247–1252.