Forecasting Trends in Technological Innovations with Distortion-Aware Convolutional Neural Networks

Krisztian Buza, M. Besher Massri, Marko Grobelnik
{krisztian.antal.buza,besher.massri,marko.grobelnik}@ijs.si
Artificial Intelligence Laboratory, Institute Jozef Stefan
Ljubljana, Slovenia

ABSTRACT

Predicting trends in technological innovations holds critical importance for policymakers, investors, and other stakeholders within the innovation ecosystem. This study approaches this challenge by framing it as a time series prediction task. Recent efforts have introduced diverse solutions utilizing convolutional neural networks, including distortion-aware convolutional neural networks. While convolutional layers act as local pattern detectors, conventional convolution matches local patterns in a rigid manner in the sense that they do not account for local shifts and elongations, whereas distortion-aware convolution incorporate the capability to identify local patterns with flexibility, accommodating local shifts and elongations. The resulting convolutional neural network, with distortion-aware convolution, has exhibited superior performance compared to standard convolutional networks in multiple time series prediction tasks. As a result, we advocate for the application of distortion-aware convolutional networks in forecasting technological innovation trends and compare their performance with conventional convolutional neural networks.

CCS CONCEPTS

• **Computing methodologies → Neural networks**.

KEYWORDS

trends, innovation ecosystem, time series forecasting, convolutional neural networks, distortion-aware convolution

1 INTRODUCTION

Forecasting trends in technological innovations is of high value for policy makers, investors and other actors of the innovation ecosystem. In this paper, we cast this task as a time series forecasting problem.

Approaches for time series forecasting range from the well-known autoregressive models [4] over exponential smoothing [12] to solutions based on deep learning [10, 11, 16–19, 24, 26]. Among the numerous techniques, a prominent family of methods include forecast with convolutional neural networks (CNNs) [3, 20].

The inherent assumption behind CNNs is that local patterns are characteristic to time series and future values of the time series may be predicted based on those local patterns. While the operation of convolution plays the role of a local pattern detector, it matches patterns in a rigid manner as it does not allow for local shifts and elongations within the patterns. This issue has been addressed by distortion-aware convolution and the resulting convolutional neural network has been shown to outperform conventional convolutional networks in case of several time series forecasting tasks [6].

For the aforementioned reasons, in this paper we propose to use distortion-aware convolutional networks for forecasting trends in technological innovations. We perform experiments on real-world time series of the number of patents related to selected topics. We compare the performance of distortion-aware convolutional networks with conventional convolutional neural networks.

The reminder of the paper is organized as follows. In Section 2, we provide a short discussion of related works. We review distortion-aware convolutional networks in Section 3, followed by the experimental results in Section 4. Finally, we conclude in Section 5.

2 RELATED WORK

As we cast our problem as a time series forecasting task, we focus our review of related works on time series forecasting. As mentioned previously, a prominent family of methods include forecast techniques based on convolutional neural networks, recent surveys about them have been presented by Lim et al. [17], Sezer et al. [21] and Torres et al. [24].

An essential component of distortion-aware convolution is dynamic time warping (DTW). While DTW is one of the most successful distance measures in the time series domain, see e.g. [25], recent approaches integrate it with neural networks. For example, Iwana et al. [14], Cai et al. [9] and Buza [5] used DTW to construct features. In contrast, Afrasiabi et al. [1] used neural networks to extract features and used DTW to compare the resulting sequences. Shulman [22] proposed "an approach similar to DTW" to allow for flexible matching in case of the dot product. DTW-NN [13] considered neural networks and replaced "the standard inner product of a node with DTW as a kernel-like method". However, DTW-NN only considered multilayer perceptrons (MLP), whereas we focus on convolutional networks. In the context of time series classification, Buza and Antal proposed to replace the dot product in the convolution operation by DTW calculations [7]. In distortion-aware convolution [6], DTW is used together with the dot product, but the dot product itself is not modified.

3 BACKGROUND

We begin this section with a formal definition of our task followed by a review of convolutional neural networks with distortion-aware convolution [6].

Slovenian KDD Conference 2023 , 9–13 October 2023, Ljubljana, Slovenia

3.1 Problem Formulation

Given an observed time series $x = (x_1, \ldots, x_l)$ of length l, in our case each x_i represents the number of patents related to a given topic in a month, we aim at predicting its subsequent h values $y = (x_{l+1}, \ldots, x_{l+h})$, i.e., the number of patents in the subsequent h months. We say that h is the forecast horizon and y is the target. Furthermore, we assume that a dataset D is given which contains n time series with the corresponding target:

$$D = \{(x^{(i)}, y^{(i)})_{i=1}^{n}\}. \tag{1}$$

We use D to train neural networks for the aforementioned prediction task. We say that $x^{(i)}$ is the input of the neural network.

In our experiments, we assume that an independent dataset D^* is given which can be used to evaluate the predictions of our model. Similarly to D, dataset D^* contains pairs of input and target time series. D^* is called the test set.

3.2 The Distortion-aware Convolutional Block

The main idea behind distortion-aware convolution [6] is to calculate, besides the dot products (or inner products), DTW distances between the kernel and time series segments as well. This is illustrated in Fig. 1. Our distortion-aware convolutional block has two output channels: one for dot products and another channel for the DTW distances.

While in case of the dot product, higher similarity between the time series segment and the pattern corresponds to higher values, the opposite is true for the DTW distances. In case of DTW, high similarity between the time series segment and the pattern is reflected by a distance close to zero. Therefore, to make sure that the activations on both channels are consistent, the activations of the DTW channel of our distortion-aware convolutional block are calculated as follows:

$$out_{DTW}(t) = \frac{1}{1 + DTW(in[t : t+s], w)}, \tag{2}$$

where out_{DTW} denotes the activation of the DTW channel of the distortion-aware convolutional block, $in[t : t+s]$ is the segment of the block's input between the t-th and $(t+s)$-th position[1], s is the size of the filter, w are the weights of the filter representing a local pattern and $DTW(.,.)$ is a function that calculates the DTW distance between two time series segments.

Training neural networks with distortion-aware convolution may be challenging because of the backpropagation of gradients through the DTW calculations. The basic idea of training is to train the network with conventional convolution instead of distortion-aware convolution initially and add DTW-computations once the weights of the convolutional layer have already been determined. For details, see [6].

4 EXPERIMENTAL EVALUATION

The goal of our experiments is to examine whether the neural networks with distortion-aware convolution are more suitable for forecasting technological trends compared to their counterparts with conventional convolution.

[1] In Eq. (2) we use a Python-like syntax: the lower index, t is inclusive, the upper index, $t + s$ is exclusive in $in[t : t + s]$.

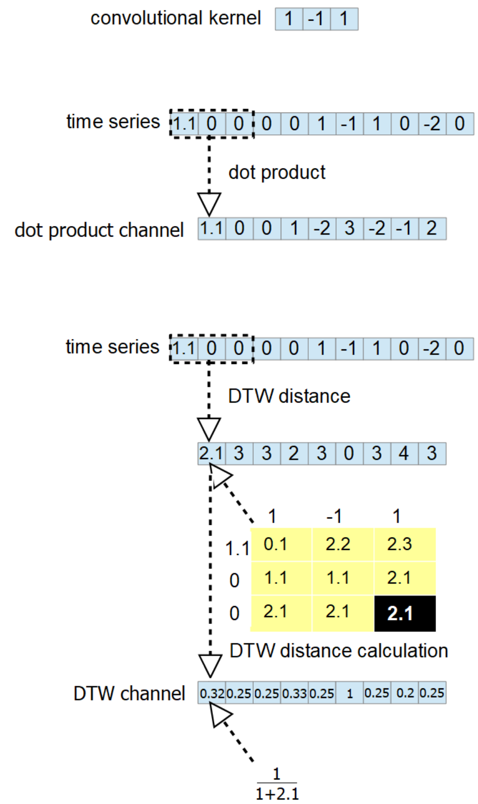

Figure 1: In case of distortion-aware convolution, additionally to the dot product (top), DTW distances between the kernel and time series segments are calculated (bottom). Thus, our distortion-aware convolutional block has two output channels: one for dot products and another channel for the DTW distances scaled according to Eq. (2).

4.1 Data

Lens is a web-based service that offers global access to patent information, academic articles, regulatory databases, and additional relevant materials.[2] The platform is designed to simplify the exploration and evaluation of intellectual property information while promoting research and inventive activities. Lens grants complimentary access to patent databases from more than 100 nations and includes sophisticated search functionalities and analytical tools for diverse research and analysis needs.

We extracted time series from the Lens patent database as follows. For selected topics identified by their Cooperative Patent Classification (CPC) codes, we extracted the number of granted patents as well as the number of patent applications per month between January 1980 and December 2022. We considered the following topics: (a) “image or video recognition” (G06V), (b) “neural networks” (G06N3/02), (c) “natural language processing” (G06F40) and (d) all topics related to artificial intelligence. We considered the number of patents separately for the most significant jurisdictions, i.e., (a) United States of America, (b) China, (c) Korea, (d) Japan and

[2] http://lens.org

Table 1: Mean absolute error (MAE) and root mean squared error (RMSE) for forecasting the time series of granted patents in case of our approach (DCNN) and the baseline (CNN). Lower values indicate better performance.

topic	juris-diction	RMSE CNN	RMSE DCNN	MAE CNN	MAE DCNN
image or	US	165.9	106.0	131.2	92.7
video	China	405.8	320.9	323.87	217.6
recognition	Korea	13.9	27.7	12.4	19.9
	Japan	55.9	49.8	39.9	37.8
	Europe	34.5	34.7	32.3	32.9
	ALL	494.7	399.6	416.8	341.3
neural	US	10.7	9.1	9.4	7.9
networks	China	5.6	5.5	3.8	3.7
	Korea	6.3	2.3	5.4	2.1
	Japan	3.5	2.9	2.5	2.0
	Europe	2.7	1.6	2.2	1.2
	ALL	7.6	8.3	6.3	6.7
natural	US	19.7	15.1	14.8	12.0
language	China	57.1	47.0	41.6	41.7
processing	Korea	14.2	8.5	13.1	7.3
	Japan	11.8	10.7	9.5	7.3
	Europe	4.8	3.0	3.5	2.7
	ALL	67.0	45.7	59.5	35.5
ALL	US	270.2	216.9	224.1	196.4
	China	870.2	1108.8	763.2	998.1
	Korea	56.6	138.3	53.8	129.4
	Japan	124.8	132.0	81.4	89.9
	Europe	85.8	69.2	82.1	65.9
	ALL	1045.1	1129.1	929.2	964.6

Table 2: Mean absolute error (MAE) and root mean squared error (RMSE) for forecasting the time series of patent applications in case of our approach (DCNN) and the baseline (CNN). Lower values indicate better performance.

topic	juris-diction	RMSE CNN	RMSE DCNN	MAE CNN	MAE DCNN
image	US	188.2	177.1	170.2	163.3
or video	China	3405.0	1061.7	3375.4	1042.3
recognition	Korea	128.9	70.8	99.7	69.4
	Japan	103.8	106.4	87.1	66.1
	Europe	51.9	55.5	45.0	49.4
	ALL	3641.9	2110.5	3627.3	2027.8
neural	xUS	79.8	15.3	76.9	12.7
networks	China	21.2	20.8	16.8	19.0
	Korea	44.6	6.8	43.7	6.2
	Japan	13.9	7.1	13.5	4.8
	Europe	15.8	5.9	14.9	4.4
	ALL	267.7	45.6	262.7	38.6
natural	US	64.1	68.7	55.5	64.6
language	China	418.9	318.2	363.6	289.3
processing	Korea	35.1	23.4	29.7	21.0
	Japan	16.7	18.7	10.5	10.8
	Europe	11.2	14.3	9.7	11.2
	ALL	298.1	543.0	226.9	489.3
ALL	US	532.3	329.1	458.9	311.3
	China	6443.7	2784.2	6239.0	2386.5
	Korea	405.4	216.8	340.2	180.8
	Japan	224.8	228.1	159.1	128.6
	Europe	130.0	163.5	97.5	121.3
	ALL	5445.1	3355.8	5009.0	2547.0

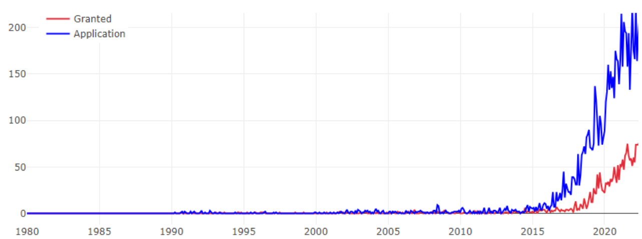

Figure 2: Total number of granted patents (red) and patent applications (blue) for all the jurisdictions in the Lens database related to "neural networks" (CPC: G06N3/02).

(e) Europe. Additionally, we considered the time series of the total number of patents for all the jurisdictions of the database. Thus, we considered in 48 time series in total, see also the first two columns of Tab. 1 and Tab. 2. Two example time series are shown in Fig. 2.

For each time series, we trained the neural networks to predict the number of granted patents (or patent applications, respectively) for each month of a 6-monthly period, i.e., the forecast horizon was $h = 6$. As input, we used the number of granted patents (or patent applications, respectively) in the previous 36 months. The data related to the years 1980...2019 was used as training data, while the data from 2019...2022 was used as test data.

From the long time series corresponding years 1980...2019, we extracted training instances with a moving window. This resulted in 10496 training instances in total which corresponds to 427 training instance for each time series.

When evaluating the network on the test data, we used the data from 2019...2021 as input data and the task was to predict the number of granted patents (or patent applications, respectively) for the first six month of 2022.

4.2 Experimental Settings

In order to assess the contribution of distortion-aware convolution, for each time series, we trained two versions of the neural network: *with* and *without* distortion-aware convolution, and compared the results. In the former case, the first hidden layer was a distortion-aware convolutional layer (with both dot product and DTW calculations), whereas in the later case, we used conventional convolution (with dot product only).

For simplicity, we considered a convolutional network containing a single convolutional layer with 25 filters, followed by a max pooling layer with window size of 2, and a fully connected layer with 100 units. We set the size of convolutional filters to 9. The number of units in the output layer corresponds to the forecast horizon, as each unit is expected to predict one of the numeric values of the target time series. We trained the networks for 1000 epochs with the Adam optimizer [15] with learning rate of 10^{-5} and batch size of 16. The loss function was mean squared error.

We implemented our neural networks in Python using the PyTorch framework. In order to support reproduction of our work, we made the implementation of our model publicly available in a github repository. The code illustrates training and evaluation of our model on standard benchmark datasets.[3]

We evaluated the predicted time series both in terms of mean absolute error (MAE) and root mean squared error (RMSE). In particular, we calculated MAE (and RMSE, respectively) for each forecast time series.

As the goal of our experiments is to assess the contribution of distortion-aware convolution, our baseline, denoted as CNN, is the aforementioned neural network with conventional convolution instead of distortion-aware convolution.

4.3 Results

Tab. 1 and Tab. 2 show our results in terms of MAE and RMSE. Our approach, convolutional neural network with distortion-aware convolution is denoted by DCNN, while CNN denotes the neural network with conventional convolution. As one can see, in the majority of the examined cases, DCNN outperforms CNN both in terms of MAE and RMSE. In those cases when CNN performs better, typically, both models are rather accurate (the error is low for both models) or the difference is very small compared to the magnitude of the error.

5 CONCLUSIONS AND OUTLOOK

In this paper, we focused on forecasting technological trends and cast this task as a time series forecasting problem. We considered a recent approach, convolutional neural networks with distortion-aware convolution, which has not been used for this task previously.

We performed experiments on real-world time series representing the number of granted patents and patent applications related to selected topics. Our observations show that convolutional neural networks with distortion-aware convolution are promising for this task. Furthermore, combination of conventional convolutional networks and neural networks with distortion-aware convolution may be an interesting target of future works.

Last, but not least, we mention that time series are prominent in various real-world applications [2, 23] and our approach can be extended to handle other types of time series, such as multivariate time series (or series of vectors) that can be compared with a more general version of DTW, see e.g. [8].

ACKNOWLEDGMENTS

This work was supported by the European Union through enRich-MyData EU HE project under grant agreement No 101070284 and Slovenian National grants (CRP V2-2272; V5-2264; CRP V2-2146.

[3] https://github.com/kr7/dcnn-forecast

REFERENCES

[1] Mahlagha Afrasiabi, Muharram Mansoorizadeh, et al. 2019. DTW-CNN: time series-based human interaction prediction in videos using CNN-extracted features. *The Visual Computer* (2019), 1–13.

[2] Margit Antal and László Zsolt Szabó. 2016. On-line verification of finger drawn signatures. In *11th international symposium on applied computational intelligence and informatics*. IEEE, 419–424.

[3] Anastasia Borovykh, Sander Bohte, and Cornelis W Oosterlee. 2017. Conditional time series forecasting with convolutional neural networks. *arXiv preprint arXiv:1703.04691* (2017).

[4] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. 2015. *Time series analysis: forecasting and control.* John Wiley & Sons.

[5] Krisztian Buza. 2020. Asterics: Projection-based classification of eeg with asymmetric loss linear regression and genetic algorithm. In *14th International Symposium on Applied Computational Intelligence and Informatics*. IEEE, 000035–000040.

[6] Krisztian Buza. 2023. Time Series Forecasting with Distortion-Aware Convolutional Neural Networks. In *9th SIGKDD International Workshop on Mining and Learning from Time Series.*

[7] Krisztian Buza and Margit Antal. 2021. Convolutional neural networks with dynamic convolution for time series classification. In *International Conference on Computational Collective Intelligence*. Springer, 304–312.

[8] Krisztian Antal Buza. 2011. Fusion methods for time-series classification. *PhD thesis at the University of Hildesheim* (2011).

[9] Xingyu Cai, Tingyang Xu, Jinfeng Yi, Junzhou Huang, and Sanguthevar Rajasekaran. 2019. DTWNet: a dynamic time warping network. *Advances in neural information processing systems* 32 (2019).

[10] Zhengping Che, Sanjay Purushotham, Guangyu Li, Bo Jiang, and Yan Liu. 2018. Hierarchical deep generative models for multi-rate multivariate time series. In *International Conference on Machine Learning*. PMLR, 784–793.

[11] Marco Cuturi and Mathieu Blondel. 2017. Soft-dtw: a differentiable loss function for time-series. In *International conference on machine learning*. PMLR, 894–903.

[12] Everette S Gardner Jr. 2006. Exponential smoothing: The state of the art—Part II. *International journal of forecasting* 22, 4 (2006), 637–666.

[13] Brian Kenji Iwana, Volkmar Frinken, and Seiichi Uchida. 2020. DTW-NN: A novel neural network for time series recognition using dynamic alignment between inputs and weights. *Knowledge-Based Systems* 188 (2020), 104971.

[14] Brian Kenji Iwana and Seiichi Uchida. 2020. Time series classification using local distance-based features in multi-modal fusion networks. *Pattern Recognition* 97 (2020), 107024.

[15] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[16] Vincent Le Guen and Nicolas Thome. 2019. Shape and time distortion loss for training deep time series forecasting models. *Advances in neural information processing systems* 32 (2019).

[17] Bryan Lim and Stefan Zohren. 2021. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A* 379, 2194 (2021), 20200209.

[18] Linbo Liu, Youngsuk Park, Trong Nghia Hoang, Hilaf Hasson, and Luke Huan. 2023. Robust Multivariate Time-Series Forecasting: Adversarial Attacks and Defense Mechanisms. In *The Eleventh International Conference on Learning Representations.*

[19] Yao Qin, Dongjin Song, Haifeng Chen, Wei Cheng, Guofei Jiang, and Garrison Cottrell. 2017. A dual-stage attention-based recurrent neural network for time series prediction. *arXiv preprint arXiv:1704.02971* (2017).

[20] Rajat Sen, Hsiang-Fu Yu, and Inderjit S Dhillon. 2019. Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting. *Advances in neural information processing systems* 32 (2019).

[21] Omer Berat Sezer, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. 2020. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied soft computing* 90 (2020), 106181.

[22] Yaniv Shulman. 2019. Dynamic Time Warp Convolutional Networks. *arXiv preprint arXiv:1911.01944* (2019).

[23] Abdul Sittar and Dunja Mladenić. 2023. An approach to creating a time-series dataset for news propagation: Ukraine-war case study. In *Slovenian KDD Conference.*

[24] José F Torres, Dalil Hadjout, Abderrazak Sebaa, Francisco Martínez-Álvarez, and Alicia Troncoso. 2021. Deep learning for time series forecasting: a survey. *Big Data* 9, 1 (2021), 3–21.

[25] Xiaopeng Xi, Eamonn Keogh, Christian Shelton, Li Wei, and Chotirat Ann Ratanamahatana. 2006. Fast time series classification using numerosity reduction. In *Proceedings of the 23rd international conference on Machine learning*. 1033–1040.

[26] Tian Zhou, Ziqing Ma, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, Rong Jin, et al. 2022. Film: Frequency improved legendre memory model for long-term time series forecasting. *Advances in Neural Information Processing Systems* 35 (2022), 12677–12690.