Exposé Study Project: Evaluating Dilation in Time Series Classification

Michael Hirsch
Department of Computer Science
HU Berlin

July 1, 2022

Supervisor: Dr. rer. nat. Patrick Schäfer

1 Introduction

A time series is a sequence of values ordered by time. Time series classification is a form of supervised learning to learn from time series with labels and then predict the class label of time series with an unknown label. The main difference to general classification algorithms is the ordering of the values.

There are different approaches used in time series classification but many state-of-the-art classifiers are using (sliding) windows (subsequences) to process the time series. The goal is to extract specific local patterns in the time series which then helps to classify the time series.

ROCKET [DPW20] is a relatively new classifier that promises lower computational complexity with the same accuracy when compared to many state-of-the-art classifiers. ROCKET uses a sliding window combined with dilation and gets results faster with the same accuracy [DPW20]. Dilation describes the spacing between values inside the sliding window so the effect of dilation is smoothening a time series by deliberately ignoring some values. With increasing dilation, it is possible to capture the same pattern at different scales and look at a wider range of data without increasing the number of values inside the window, hence keep the memory and computation usage low.

In most state-of-the-art classifiers there is no dilation (meaning zero spacing) in between the values inside the sliding window, instead the size of the window hence the number of values is increased to capture a wider scale, consequently the memory and computation complexity rises too. Instead of taking an increased number of values to capture a wider scale, it is possible to increase only the dilation of the sliding window. With only increasing the dilation, the memory and computation complexity is kept low and the input time series gets sampled implicitly. The objective is of this thesis is to examine what effects the use of dilation has.

This study project is about applying a window with dilation to different time series classification algorithms that use a windowing approach and examine the difference in performance, memory and accuracy. The goal of this thesis is to figure out if the promising results from ROCKET are transferable to other time series classifiers by means of dilation.

2 Background and Definitions

Time series: A time series $T = (t_1, t_2, ..., t_n)$ represents a collection of n values obtained from sequential measurements over time, often sampled at equal time intervals. [EA12]

Sliding window: A sliding window is a 1-dimensional array of values of fixed length l that slides over the input time series, extracting subsequences of length l of each offset.

Convolutional neural network: A Convolutional Neural Network (CNN) is an artifical neural network with the purpose of detecting patterns in the input using convolutional kernels which act as a filter. A convolutional layer of a CNN receives the input, convolutes it with one or more convolutional kernels and passes the output to the next layer. Then a pooling layer reduces the spatial size of the convoluted features. This combination of convolution and pooling can be applied to the data multiple times. With the use of those layers, it is possible for the CNN to detect complex patterns in the input data. [AMAZ17]

Convolutional kernel: CNNs use convolutional kernels inside the convolutional layers to detect patterns in the input. In the application of time series, the kernel is a one dimensional vector of weights with a size, bias, dilation and padding. The weights are the skalar values stored inside the kernel, the size is the number of values inside the kernel and the bias is added to the result after the convolutional operation. The dilation describes the space between the values in the kernel itself. With dilation, it is possible to capture the same pattern at different scales and look at a wider range of data for the convolution without increasing the number of values inside the kernel hence keep the memory and computation usage low.

The padding handles the start and end of the input time series by appending a default value (typically zeros) to the time series so the middle of the kernel can start at the first real value of the time series.

After defining a kernel, it is applied to the time series through a sliding dot product to create a transformed time series. In this context, a 1-dimensional kernel is often referred to as a sliding window. [DPW20]

Pooling: The application of k random convolutional kernels to the input time series results in k feature maps with each as long as the input time series. To further work with the feature maps, pooling is used to reduce the dimensionality and achieve spatial invariance. [DPW20]

Pooling combines the outputs of the convolution into one data point. So the pooling output is a more compact representations with a better robustness to noise inside the data. It also reduces the memory usage and computational complexity in the CNN. [BPL10]

One pooling approach that is also used in ROCKET is global max pooling. The output is the maximum value from the input feature map. The second pooling approach used by ROCKET is called proportion of positive values (ppv). The output is the proportion of positive values from the feature map. [DPW20]

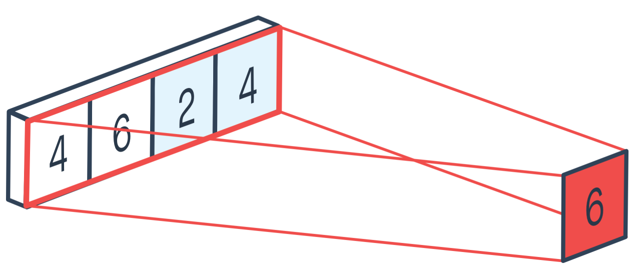

Figure 1: 1D Global max pooling [Glo]

Sliding window with dilation: In this project the window has a length l and a dilation size d. The dilation describes the spacing between values inside the sliding window (see Figure 2 for an example). The length is defined as the number of values inside the sliding window + the empty spaces between them (defined by the dilation size d).

In the context of most state-of-the-art time series classifiers, a sliding window is used to extract subseries of the time series. These samples are then used to analyse the patterns inside the input time series e.g. the frequence of specific patterns. To capture patterns at a wider scale, the length of the window is increased by means of increasing the number of values. There is no dilation used.

In the classifier ROCKET, the sliding window is a 1-dimensional dilated convolutional kernel, so the time series gets dilated and convoluted to create transformed time series. In contrast to most state-of-the-art time series classifiers to increase the window size, the dilation factor is increased. This keeps the number of values inside the sliding window the same while increasing the overall length of the sliding window and capturing patterns at a wider scale.

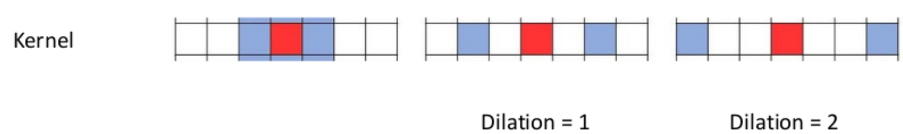

Figure 2: Example of a dilated convolutional kernel

3 State-of-the-art Classifiers

There are many different approaches to classify time series. Bagnall et al. [BLB+17] categorized them. One approach for each dilation applicable category was chosen, to examine the dilation approach applied to different types of classification algorithms:

1. **Distance based / Whole Series:**
   Distance based classifiers compare two time series as a vector and calculates the smallest distance between them. The classification is then based on the distance between the input time series and already labeled

time series. These classifiers don't use sliding windows so it is not possible to apply dilation. Therefore this category is not considered in this project.

2. **Interval based:**
   Interval based classifiers select random intervals of the time series, compute summary statistics over these intervals and classify's them based on these extracted features. This approach uses windows to select values for computing the summary statistics so dilation can be applied to extract new features at multiple resolutions. In this project the interval based approach is represented by the Time Series Forest Classifier (TSF). [DRTV13]

3. **Shapelets:**
   Shapelet based classifiers try to detect characteristic subsequences of the time series that represents a class and use the presence or absence of these subsequences to classify unknown time series. A single shapelet is a subsequence in a time series. To find and extract these characteristic shapelets a sliding window is used. So dilation can be applied directly to the window for shapelet extraction and different dilated shapelets can be used for classification. In this project the shapelet based approach is represented by the Shapelet Transform classifier. [LDHB12] Guillaume et. al. implemented random dilation in Shapelet Transform that improved global performance, especially scalability, in the state-of-the-art shapelet algorithm. [GVE22] This paper is also used as a reference to try good parameters for the benchmark of this project.

4. **Dictionary Based:**
   Dictionary-based classifiers transform time series into a sequence of discrete "words". Classification is then based on the frequency of these words. To extract these discrete "words" a sliding window is used. With dilation inside the sliding window it is possible to detect dilated "words" at mutliple resolutions. In this project the dictionary based approach is represented by the Bag of SFA Symbols (BOSS) classifier. [Sch15]

4 Objective

ROCKET shows that with dilation, it is possible to capture same patterns at different scales and look at a wider range of data without increasing the number of values inside the kernel hence keep the memory and computation usage low. [DPW20] The objective of this project is to see, if this dilation approach can also improve other state-of-the-art classifiers. So dilation is used to sample the input time series and then the dilated time series gets classified with those different state-of-the-art classifiers mentioned above.

Performance, memory usage and accuracy are going to be measured and compared to the classifications without the first step of applying the dilated sliding window. In addition the effect of varying the parameters of the sliding window on the measurements are going to be examined. So the goal is to look at performance, memory and accuracy to see if dilation inside the state-of-the-art classifiers can improve any of it without impairing another one.

5 Methods: Using Dilation in state-of-the-art classifiers

A simple mapping is used, to avoid the need of re-implementing the whole algorithms from each approach: The dilation will be applied beforehand on the time series. In essence this dilation mapping basically only rearrange the time series to mimic the dilation before the actual windowing process inside the classifiers happens. For the dilation mapping there is one parameter, the dilation size d, i.e. the spacing between the values.

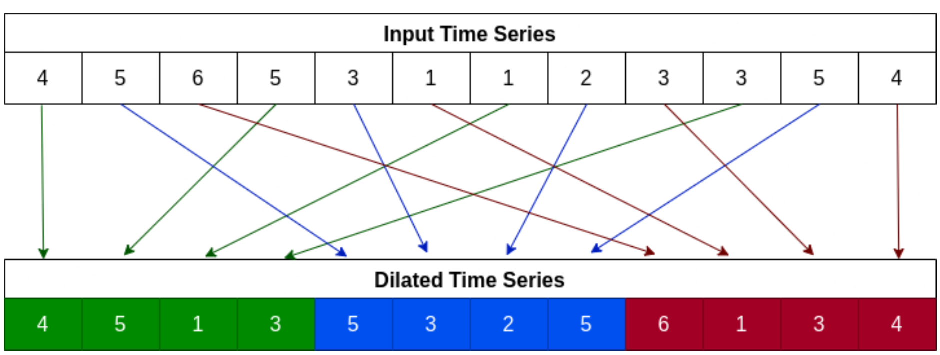

Figure 3: Example of the dilation mapping with d = 2

The dilation mapping splits a time series into $d + 1$ sequences and concatenate them. It starts at the first element of the input time series and takes every dth element to create a new sequence. The next sequence of elements starts at the second element and takes every dth element from there. This process is done $d + 1$ times, so every element is inside one sequence. At the end all sequences are concatenated to the dilated time series. The dilated sliding window approach is split into two steps. The dilation is applied with the dilation mapping (parameter: dilation size d) and the sliding window without dilation is applied inside the state-of-the-art classifiers. The produced samples in the dilated sliding window method (from ROCKET) and the two step method (this project) are the same, except d additional samples are in the two step method. Those are the result from values of two sequences. However with a relatively small dilation size compared to the number of values inside the time series, it is expected to not have a big effect on the result. For an example to see that this mapping works, see Figure 4. On the left side is our dilation mapping and the result is passed to a state-of-the-art classifer. On the right side is the application of dilation directly inside the sliding window. The resulting samples are the same except the one sample that is the result from two sequences.

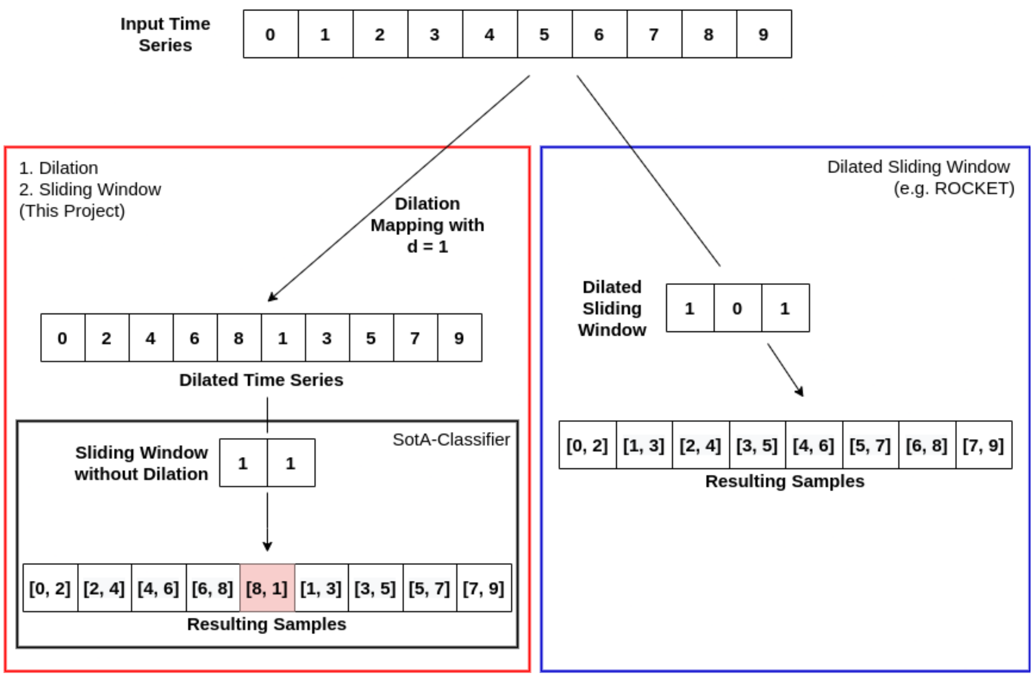

Figure 4: Dilation + Sliding Window vs. Dilated Sliding Window

There are multiple parameters to set for different results. The dilation size and the parameters of each classifier. Determining how these values are set the best, is part of the project progress as most classifiers vary the length of the window, we expect to adapt this parameter. For training, validation and testing, the UCR dataset [DKK+18] is used. The following is the rough procedure of the project.

1. Obtain baseline results for the different time series classifiers with the UCR dataset
2. Implement the dilation step prior to classification
3. Evaluate the different time series classifiers with the UCR dataset and learn optimal parameters with dilation
4. Compare results to each other and get to a conclusion

References

[AMAZ17] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)*, pages 1–6. Ieee, 2017.

[BL14] Anthony Bagnall and Jason Lines. An experimental evaluation of nearest neighbour time series classification. *arXiv preprint arXiv:1406.4757*, 2014.

[BLB+17] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data mining and knowledge discovery*, 31(3):606–660, 2017.

[BPL10] Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118, 2010.

[DKK+18] Hoang Anh Dau, Eamonn Keogh, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, Yanping, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, Gustavo Batista, and Hexagon-ML. The ucr time series classification archive, October 2018. https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

[DPW20] Angus Dempster, François Petitjean, and Geoffrey I Webb. Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34(5):1454–1495, 2020.

[DRTV13] Houtao Deng, George Runger, Eugene Tuv, and Martyanov Vladimir. A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153, 2013.

[EA12] Philippe Esling and Carlos Agon. Time-series data mining. *ACM Computing Surveys (CSUR)*, 45(1):1–34, 2012.

[Glo] Global max pooling 1d. https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/global-max-pooling-1d. (Accessed on 05/28/2022).

[GVE22] Antoine Guillaume, Christel Vrain, and Wael Elloumi. Random dilated shapelet transform: A new approach for time series shapelets. In *International Conference on Pattern Recognition and Artificial Intelligence*, pages 653–664. Springer, 2022.

[LDHB12] Jason Lines, Luke M Davis, Jon Hills, and Anthony Bagnall. A shapelet transform for time series classification. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 289–297, 2012.

[Sch15] Patrick Schäfer. The boss is concerned with time series classification in the presence of noise. *Data Mining and Knowledge Discovery*, 29(6):1505–1530, 2015.