DeepVATS: A tool for Deep Learning Visual Analytics on Large Time Series Data

Keynote paper

Inmaculada Santamaria-Valenzuela
Department of Computer Systems Engineering
Universidad Politécnica de Madrid
Madrid, Spain
mi.santamaria@upm.es

Victor Rodriguez-Fernandez
Department of Computer Systems Engineering
Universidad Politécnica de Madrid
Madrid, Spain
victor.rfernandez@upm.es

Jong Hyuk Park
Department of Computer Science
and Engineering
Seoul National University of Science
and Technology (SeoulTech)
Seoul, Korea
jhpark1@seoultech.ac.kr

David Camacho
Department of Computer Systems Engineering
Universidad Politécnica de Madrid
Madrid, Spain
david.camacho@upm.es

***Abstract*—DeepVATS is a Deep Learning Visual Analytics tool that integrates Deep Learning and Visual Analytics for the analysis of extensive time series data. This paper presents the main components of DeepVATS, its execution pipeline, and future research directions.**

***Index Terms*—DeepVATS, Visual Analytics, shiny**

I. Introduction

DeepVATS is a tool inspired by TimeCluster [1], which integrates Visual analytics (VA) and Deep Learning (DL). It uses dimensionality reduction (DR) techniques for efficient analysis and interaction with large time series. Its main tasks are: learn good embeddings of the time series; cluster and project them in 2 dimensions; and finally, provide interactive visualizations to explore different perspectives of the projected embeddings to detect spot patterns and anomalies.

This keynote paper is based on [2], [3], that show DeepVATS' tool using different datasets and cases of use. The code of the tool is publickly available on https://github.com/vrodriguezf/deepvats.

The rest of the paper is organized as follows. Section II. DeepVATS description provides a basic description of the tool. Section III. Execution pipeline describes more precisely the execution pipeline. Finally, section IV. Conclusions and future research directions outlines the main conclusions and the future guidelines for the tool developement.

This research was supported by Strategic Networking & Development Program funded by the Ministry of Science and ICT through the National Research Foundation of Korea (RS-2023-00267476), by Grant PLEC2021-007681 (XAI-DisInfodemics) and PID2020-117263GB-100 (FightDIS) funded by MCIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe", by the "European Union" or by the "European Union NextGenerationEU/PRTR", by European Commission under IBERIFIER Plus - Iberian Digital Media Observatory (DIGITAL-2023-DEPLOY-04-EDMO-HUBS, Project Nº: 101158511), by Calouste Gulbenkian Foundation, under the project MuseAI - Detecting and matching suspicious claims with AI and by "Convenio Plurianual with the Universidad Politécnica de Madrid in the actuation line of Programa de Excelencia para el Profesorado Universitario"

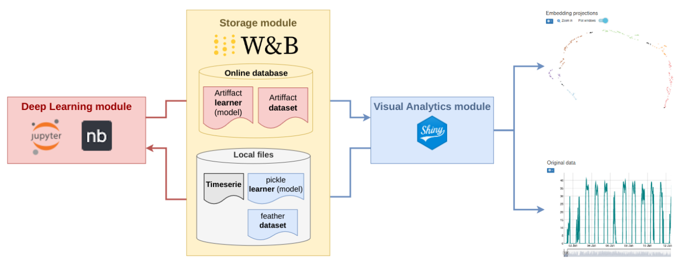

Fig. 1. DeepVATS' global workflow

II. DeepVATS description

DeepVATS integrates three main modules: Deep Learning Module, Storage Module and Visual Analytics Module. The backbone of DeepVATS is the Masked Time Series Autoencoder (MTSAE) [3]. This deep learning architecture encodes, through a self-supervised masking strategy, the time series data, enabling powerful visual analytics on the encoder's embedding space. The MTSAE is implemented with the `tsai` library [4], and forms the basis of DeepVATS' deep learning module (See Fig. 1)

The Storage module uses Weights & Biases (W&B, wandb.ai) as a "database" for saving the dataset and encoder artifacts as well as any other information that may be used by both the Deep Learning and the Visual modules or that is simply too heavy to be stored within the local memmory.

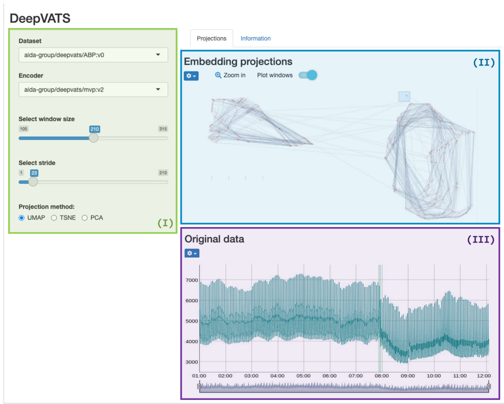

Fig. 2. Graphical User Interface of DeepVATS. Note that there's a section of the embedding projections selected, which is in turn highlighted in the time series plot, showing the two-way communication between the plots.

The Visual Analytics module, the interactive component of DeepVATS, uses the DeepLearning Module to get encoder's embeddings, clusters them using hdbscan [5], and projects them via UMAP [6], TSNE, or PCA [7]). Then it shows two interactive plots, namely the embeddings' projection plot (PP) and the time series plot (TSP). By selecting points in the PP, their corresponding data windows are displayed in the TSP and vice versa. Additional features like zoom functionality aid in the detailed analysis and better understanding of the embeddings. DeepVATS presents three key advantages over its competitors (e.g. TimeCluster [1]). First, its MTSAE backbone model enables versatile pattern recognition beyond segmentation. Second, the model is trained across different windows sizes, reducing sensitivity to its choice. Last, its modular open-source design facilitates easy adaptation to new tasks, such as outliers detection [3].

III. Execution pipeline

This section explains more deeply deepVATS' execution pipeline. The first step is to load the dataset and training the encoder. To do so, the user must configure the experiment and W&B information in *deepvats/nbs_pipeline/config/base.yml* and *config/02b-encoder_mvp.yaml* files. Then, he/she has to execute the notebooks *deepvats/nbs_pipeline/01_dataset_artifact* and *nbs_pipeline/02b_encoder_MVP.ipybn*. This will result in a dataset and an encoder artifacts, with the encoder model being trained using the configurated parameters. Then, they can be used for the analysis or time series in the VA module, including all its metadata.

Now, the user can run de VA module for the analysis of the loaded dataset. The embeddings of a time series dataset are computed using one encoder trained. Those embeddings are projected using dimensionality reduction techniques, and visualised as an interactive connected scatter plot in tandem with an interactive time series plot, whose interactions affect the view of each other (See Fig. 2). This has been divided into 3 sections: the control panel (Mark 1 in the figure), the embeddings projection (Mark 2), which contains the central connected scatter plot, and the time series plot (Mark 3).

In the Control Panel of the GUI, the user can configure a variety of controls for the selection of datasets, encoder, algorithms and aesthetics. These controls include (among others): the Dataset selector, to select the dataset to be analysed from the list of artifacts logged in the Storage Module by the DL module (in the future, the upload of dataset will be directly available from the VA module); the Encoder selector, to select the encoder artifact; encoder parameters selectors (Window size and stride selectors); and Projection method selector, that allows to select the dimensional reduction algorithm between Uniform Manifold Approximation and Projection (UMAP) [6], t-Distributed Stochastic Neighbor Embedding (t-SNE) [8], and Principal Component Analysis (PCA) [9]. As UMAP lacks stability in the GPU version [10], [11], UMAP is being changed to use PCA followed by UMAP to ensure similar clusters in all executions.

Once the configuration has been selected in the Control Panel, the embeddings projection is computed and shown in the upper plot, and the time series plot is reloaded accordingly.

IV. Conclusions and future research directions

DeepVATS is a powerful tool that can be used to visually analyze univariate and multivariate time series in an easy and interactive way. However, some scalability problems have been shown. To improve the tool we are working on: improving the use of cache in the shiny app to ensure no recalculations; adding explainability techniques (feature selection AI techniques such as Layer-wise Relevance Propagation and feature adition); adding interactive MPlots for extra information.

References

[1] M. Ali, M. W. Jones, X. Xie, and M. Williams, "Timecluster: dimension reduction applied to temporal data for visual analytics," *The Visual Computer*, vol. 35, no. 6-8, pp. 1013–1026, 2019.

[2] I. Santamaria-Valenzuela, V. Rodriguez-Fernandez, J. H. Park, and D. Camacho, "Scalability analysis of deepvats: A framework for deep learning visual analytics on large time series data," in *Proceedings of the 2024 World Congress on Information Technology Applications and Services (World IT Congress 2024)*, (Location), World IT Congress, 2024.

[3] V. Rodriguez-Fernandez, D. Montalvo-Garcia, F. Piccialli, G. J. Nalepa, and D. Camacho, "Deepvats: Deep visual analytics for time series," *Knowledge-Based Systems*, vol. 277, p. 110793, 2023.

[4] I. Oguiza, "tsai - a state-of-the-art deep learning library for time series and sequential data." Github, 2023.

[5] scikit-learn contrib, "scikit-learn-contrib/hdbscan." https://github.com/scikit-learn-contrib/hdbscan, 2024. Accessed: Jan. 04, 2024.

[6] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[7] rapidsai, "Rapids cuml api documentation." https://docs.rapids.ai/api/cuml/stable/api/. Accessed: Jan. 04, 2024.

[8] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008.

[9] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[10] rapidsai, "[bug] different outputs for umap on cpu vs. gpu · issue #5474 · rapidsai/cuml." https://github.com/rapidsai/cuml/issues/5474. Accessed: Jan. 04, 2024.

[11] rapidsai, "[bug] accuracy of lanczos solver. · issue #313 · rapidsai/raft." https://github.com/rapidsai/raft/issues/313. Accessed: Jan. 04, 2024.