Feature Selection Methods for Predicting the Emotional Impact of Videos

Johannes Madl
Chair for Data Processing, Technical University of Munich
ga58qaf@mytum.de

***Abstract*—A previous study has examined the performance of an Echo State Network (ESN) for predicting the emotional impact of videos using the LIRIS ACCEDE dataset. By using a prior feature transformation, based on a high order tensor decomposition, I try to create more valuable inputs for the ESN or an Gated Recurrent Unit (GRU). One major problem that has been tackled is the varying video length of the dataset, which can be eliminated by using the Parafac2 decomposition. Unfortunately the feature transformation with tensor decompositions could not show significant improvements on the prediction performance.**

***Keywords*—Tensor Decomposition, Feature Selection, Video Affective Content Analysis, Emotion Prediction, Reservoir Computing.**

I. Introduction

During the rapid growth of social media and video platforms, video affective content analysis has gained more and more attention [1]. Choosing the right content that will be appealing to the users current mood is very challenging for many companies in the entertainment industry. Understanding and even predicting the emotional impact of videos can be very helpful in creating awarness and adressing advertisements more efficiently. A recent study [2] has investigated the prediction of emotional impact of videos with Echo State Networks (ESN) [3]. Unfortunately the approach could not provide any improvements in the prediction performance. One major part of the prediction is the choice of features for feeding the ESN, since the raw dataset contains not only features that contribute to a good prediction. In this case, the given features describe audio-visual characteristics of each video clip of the LIRIS ACEEDE dataset [4]. Some examples of the visual features are color energy, saturation and colorfulness, wherease audio features describe attributes like loudness or the zero-crossing-rate. However, reducing the number of input variables can improve the performance when developing a predictive model [5]. This leads to the idea to use a prior feature selection method, in order to achieve an potential increase in the prediction performance.

Feature selection with the principal component analysis (PCA) is a popular technique for getting insight into a dataset and it's structure, based on the singular value decomposition (SVD). It can be used for selecting good features of a dataset in order to achieve more accurate predictions. Unfortunately, the traditional PCA approach is limited to two dimensions, which will be exceeded very quickly when dealing with video data or other multidimensional datasets faced in data mining, neuroscience and elsewhere. However, there are higher order tensor decompositions for N-way arrays with $N \geq 3$ like: Candecomp/Parafac(CP) [6] decomposition, Tucker decomposition [7] and many more. But due to the structure of our LIRIS ACCEDE [4] dataset, there is one promising decomposition that is able to deal with varying lengths in the time domain, the Parafac2 decomposition [7]. The Parafac2 decomposition considers the non-cuboid structure of our dataset, which could lead to a better feature selection and preprocessing of the dataset. The original structure of the dataset can be seen in Figure 1, where it is displayed as a set of video slices. Each slice $X_i \in \mathbb{R}^{l_i \times m}$ with $i \in \{1, 2, ..., n\}$ is constructed out of l_k timesteps and a corresponding set of m features. The varying l_k timesteps of our dataset occur very natural and are just based on the fact that the collected and evaluated video clips have different durations.

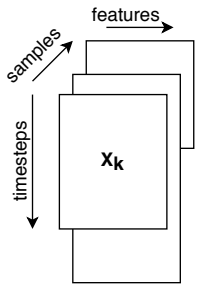

Fig. 1. Structure of the given LIRIS ACCEDE dataset [7].

The overall goal is to predict the emotional impact of videos by using reservior computing with an Echo State Network (ESN) or with a Gated Recurrent Unit Network (GRU) [8]. In general, both networks are fed by the audio-visual features of each clip and should predict their emotional impact, which is measured by two labeled numbers, valence and arousal. Both labels describe the axes of a two dimensional model for classifying emotions [2]. The mapping can be seen in Figure 2, from the previous study [2]. The valence values represents whether an excitement is good or bad, wherease the arousal values describe the strenght of the excitement. With both values, it is possible to address a big variety of emotions with only two numbers. Unfortunately a manual feature selection

in the prior study did not result in a very good performance of the ESN [2]. This ultimately raises the question whether a further improvement in the prediction of the emotional impact can be achieved. Is it possible to improve the prediction of the emotional impact of videos with the use of prior feature seletion methods?

To answer this question, we will firstly discuss possible alternatives beside the Parafac2 for the prior feature selection in our prediction task and how scores and loadings can be developed to reduce a training and validation dataset. After taking a closer look on every network of our prediction task, we will compare the effect of each feature selection method onto the prediction performance based on two experiments.

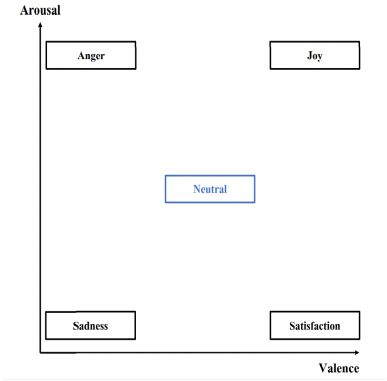

Fig. 2. Mapping categorical emotions to the two-dimensional valence-arousal space [2].

II. Feature Selection Methods

For reducing the features of the dataset, which will be fed into the recurrent networks, traditional feature selection methods as well as tensor based feature selection methods will be discussed in this section.

A. PCA

In order to apply the traditional Principle Component Analysis (PCA) onto the given dataset structure, several preprocessing tasks have to be executed. Firstly, the dataset needs to get in a cuboid shape, which can be done by selecting the longest sequence of the dataset and fill all other slices with elements towards l_{max}. For this expansion there are three noteworthy options: (1) filling up with zeros, (2) filling up with the mean value of the features over time, (3) continuing the last given value in time. After applying one of these options, the result will be a tensor $X \in \mathbb{R}^{l_{max} \times m \times n}$. Tests with 800 samples have shown that the reconstruction error does not change significantly due to different filling methods in combination with the PCA approach. By using the filling method before taking the mean value, a distortion in the mean values will be created, which will be calculated later on. With the distortion, the PCA approach can be better compared to the CP approach, which requires a cuboid shape and therefore a filling method as well. In the following implemetations I will continue the last value given in time to fill up towards an cuboid shape. Secondly I am taking the mean over all timesteps l_{max} in order to get $\bar{X} = (x_1, x_2, ..., x_m) \in \mathbb{R}^{n \times m}$, which can be expressed as followed

$$\bar{X} = \frac{\sum_{i=0}^{l_{max}-1} X_{i::}}{l_{max}} \approx U_k S_k V_k^T. \tag{1}$$

Hereby $X_{i::}$ denotes to the horizontal slice of the third-order tensor X. After taking the mean over all timesteps l_{max}, the PCA can be used to compress training and validation set with the rank $k << m$. A rank k approximation of $\bar{X}$ can be formulated by the singular value decompositoin (SVD) $U_k \in \mathbb{R}^{n \times k}$, $S_k \in \mathbb{R}^{k \times k}$, $V_k^T \in \mathbb{R}^{k \times m}$. The following equations describe the PCA and how the corresponding scores and loadings need to be calculated [9]

$$P = V_k \tag{2}$$

$$T = U_k S_k. \tag{3}$$

The matrix $T \in \mathbb{R}^{n \times k}$ contains the scores and the matrix $P \in \mathbb{R}^{m \times k}$ contains the first k right singular vectors of $\bar{X}$, which is refered as the loadings. The scores can be seen as reduced dataset of $\bar{X}$ and the loadings P will be used to project onto the subspace of T. After reducing the dimensions of the dataset, the reduced set can be used to select the best features of the original tensor. With this method, the same input structure with all feature selection methods for the ESN or the GRU is ensured. This is particularly done by calculating the pearson correlation between the reduced matrices after applying the PCA and the full matrix right before applying the PCA. The pearson correlation measures the linear relationship between two vectors [10]. All k features responsible for the highest magnitues in the pearson correlation will be selected in order to reduce the tensor slices. A short illustration of the whole PCA feature selection can be seen in Figure 3.

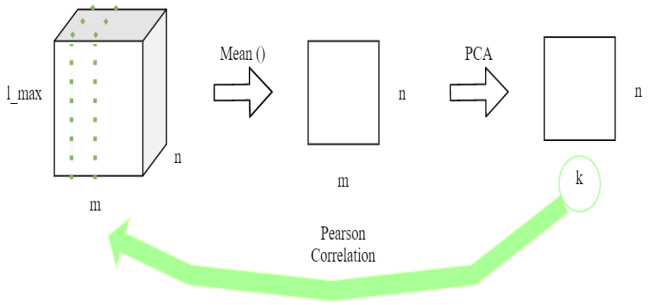

Fig. 3. PCA approach for feature selection

B. Candecomp Parafac Decomposition

The Candecomp Parafac (CP) [7] decomposition factorizes a tensor into a sum of component rank-one tensors. There

is no finite algorithm for determing the rank of a tensor, consequently the model will fit the CP decomposition with different ranks multiple times until the best setup will be received. For the feature selection with CP decomposition, the original tensor slices need to be transfered into a cuboid shape again, by filling up all the slices with continuing the last given value in time. After that, an approximation of the tensor $X \in \mathbb{R}^{l_{max} \times m \times n}$ with the following rank k can be expressed as [7]

$$\hat{X} = \sum_{i=1}^{k} a_i \circ b_i \circ c_i. \tag{4}$$

Hereby $\circ$ denotes to the Khatri-Rao product, also called the matching columnwise Kronecker product. All the previously shown rank-one vectors can be summarized into the factor matrices $A \in \mathbb{R}^{n \times k}$, $B \in \mathbb{R}^{l_{max} \times k}$ and $C \in \mathbb{R}^{m \times k}$. With these matrices it is possible to express the scores and loadings of the CP decomposition for an approximation of the mode-0 unfolding $\hat{X}_{n \times l_{max} m}$ as

$$\hat{X}_{n \times l_{max} m} = TP^T \tag{5}$$

$$P = (C \circ B)^T \tag{6}$$

$$T = A. \tag{7}$$

With the scores $T \in \mathbb{R}^{n \times k}$ representing the reduced dataset and the corresponding loadings $P \in \mathbb{R}^{k \times l_{max} m}$ we are dealing with a similar situation like before in the case of PCA [9]. By calculating the pearson correlation between the reduced dataset of dimension $\mathbb{R}^{n \times k}$ and the averaged cuboid shaped dataset of dimension $\mathbb{R}^{n \times m}$, the model is able to take the k features which result in a high correlation and use them for reducing the original dataset.

C. Parafac2

Parafac2 is not strictly a completly different tensor decomposition, it is more a variant of CP that can be applied towards a collection of matrices with different lengths like our original dataset. Compared to the CP, Parafac2 applies the same factor along one mode and allows the other factor matrices to vary. This can be seen as a relaxation of the constraints given by CP. Until now we filled up the original dataset with the last given values in time in order to receive a cuboid shaped tensor. With the Parafac2 decomposition it is possible to decompose the tensor slices directly without the previous mentioned preprocessing steps. According to Kolda [7], each slice of our original tensor X_i with $i \in \{1, 2, ..., n\}$ can be approximated by the expression

$$X_i = B_i \; diag(a_i) \; C^T. \tag{8}$$

Hereby is each slice $X_i \in \mathbb{R}^{l_i \times m}$, $B_i \in \mathbb{R}^{l_i \times k}$, $diag(a_i) \in \mathbb{R}^{k \times k}$ and $C \in \mathbb{R}^{k \times m}$. The multiplication of $B_i \; diag(a_i)$ can be seen as the scores of the slice X_i with the corresponding loadings $P = C$. In this context, the term $diag(a_i)$ represents the expansion of the vector a_i onto a diagonal matrix $diag(a_i)$ of dimension $k \times k$, where all elements of a_i are contained in the diagonal. Of course, the reduction has to be done for each slice of the tensor, which is containing the features and has the individual length l_i. There may be an advantage of the Parafac2 decomposition in feature selection due to the consideration of the original data structure. Moreover, there is no need to expand the video slices and therefore it is not necessary to create any additional distortions. A brief graphical interpretation of the ongoing Parafac2 decomposition can be seen in Figure 4.

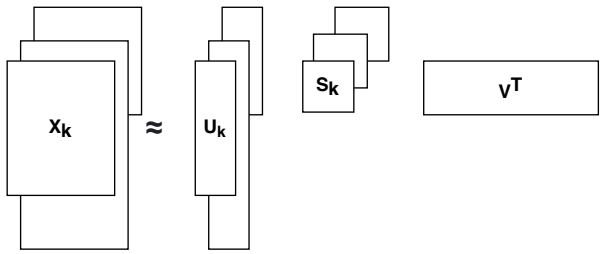

Fig. 4. Parafac2 decomposition structure [7].

III. Recurent Neural Networks

After reducing the amount of features significantly with one of the mentioned methods, I try to predict the emotional impact of the videos based on their valence and arousal features with an Echo State Network (ESN) and the Gated Recurrent Unit Network (GRU). Therefore we feed the reduced video slices into the networks, which should compress the data further on. After feeding the data into the ESN, I predict the valence and arousal values with a ridge regression model. In case of the GRU, the ridge regression model is not needed due to the structure of the GRU. Before going into detail about the experiments, I would like to shortly introduce the most important characteristics of the ESN and the GRU.

A. Echo State Network

Echo State Networks provide an architecture and the supervised learning principles for Recurrent Neural Networks (RNN), which are suitable for time series processing [11]. They are variants of RNN's but belong to the reservior computing framework [12]. Traditional neural networks suffer from the vanishing gradient problem, where parameter either do not change that much in the hidden layers or lead to chaotic behaviour. ESN do not suffer from this problems and are well adapted for handling chaotic time series [13]. Each state $x(n) \in \mathbb{R}^{N_x}$ can be described as

$$x(n) = f(W_{in} u(n) + W_r x(n-1)). \tag{9}$$

The recurrent character can be clearly seen in the formula since each state depends on the previous step. The non-linear function f is usually chosen to be the symmetric $tanh$ or a sigmoid function. Weights between the input layer and the

reservoir are refered as W_{in} and the weights of the reservoir as W_r. All weights W_r are assigned randomly and are not trainable. Only the weights of the output layer are trainable. For a rich set of dynamics the resevoir should also be sparsely interconnected [14].

B. Gated Recurrent Unit

The Gated Recurrent Unit (GRU) can be considered as a variation of the Long Short-Term Memory (LSTM), because of their similar design [8]. As it is already mentioned in the name of the GRU, it uses a gating mechanism. The implemented gates influence the flow of information inside a unit without having a seperate memory cell. Compared to the LSTM, the GRU uses only two gates, one update and one reset gate, instead of three. The update gate tells the model how much information from the past needs to be maintained and passed along to the future. The reset gate decides how much past information the model needs to forget. This structure results into the advantage of the GRU that it has fewer parameters compared to the LSTM. Like the ESN, the GRU is capable of solving the vanishing gradient problem, which occurs in regural neural networks.

IV. Experiments

The LIRIS ACCEDE [4] dataset consists of 9.800 video clips each labeled with numerical values for valence with range 1.3 to 3.6 and arousal with range 1.3 to 4.3. For each video clip 291 features describe the audio-visual attributes of each clip, like mentioned before. For the following experiments I am using 3000 video samples splitted into 75 % training data, 10 % validation data and 15 % test data. The rank k ranges from 1 to 30 with a stepsize of 3. For simplicity I have focused on the prediction of the valence values instead of taking both into account. Before I insert our datasets into the networks, I use the mentioned feature selection methods in order to select the most valuable input features with the help of the pearson correlation. After applying the PCA, CP, Parafac2 or none of them, I try to predict the valence values with the ESN and later on within a second experiment the GRU. A short overview of the whole process can be seen in Figure 5. The individual parameters for both networks will be discussed seperately in the following subsections for ESN and GRU.

A. Echo State Network Parameters

For adapting our data input in form of video slices with different length l_i towards the required input shape of the ESN, I fill up all slices towards a unitary length l_{max} with zeros. Additionally we neutralize this operation in our implementation by defining the zeros as mask values, which means they don't affect the ESN calculations or the result. Before transfering the input data into the ESN, I scale the input data with a MinMax-Scaler into the range of [0,1]. Based on the tests on the validation set I try to find the best setup before testing the final model onto our test data. I vary the number of neurons inside the range of [700,1200] with stepsize 100 and the spectral radius in the range of [0.7, 0.9] with stepsize

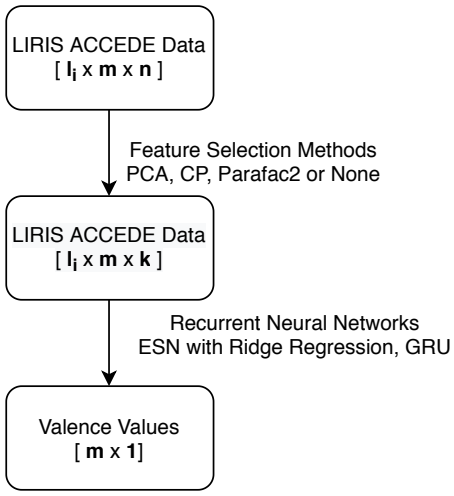

Fig. 5. Overview of all experiments in one chart.

0.1. The spectral radius is the maximum of all eigenvalues of the reservoir weights and it should be between 0 and 1 to ensure that the network has the echo state property [15]. For predicting the final valence values I use a ridge regression with the range of the regularization parameter alpha between 0.7 and 1.2 with stepsize 0.1. The prediction performance is measured with the R^2 score, defined as

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)}{\sum(y_i - \bar{y})}. \tag{10}$$

Hereby represents y_i one sample of the true values and $\hat{y}_i$ the prediction. The best possible score in this metric is 1.0 and it has no lower bound because the model can be arbitarily worse. The overall performances of the ESN can be seen in the following Figure 6. Table 1 does contain the optimal rank according to the best R^2 score of the validation set, developed while training the ESN. The selected rank will be used for the feature selection of the test set after training.

	PCA	CP	Parafac2
ESN Rank:	16	16	22

TABLE I. Chosen rank after training the network, which results in the highest R^2 score onto the validation set.

Beside the fact that the Parafac2 has the worst fit on the training set, it is able to create a positive R^2 score on the validation set for selecting the best setup. Additionally it is able to perform as the best feature selection method for the unknown test data compared to the PCA, CP or none. Unfortunately the R^2 scores onto the test set are still negative, which represents a very bad prediction. Moreover, not a single

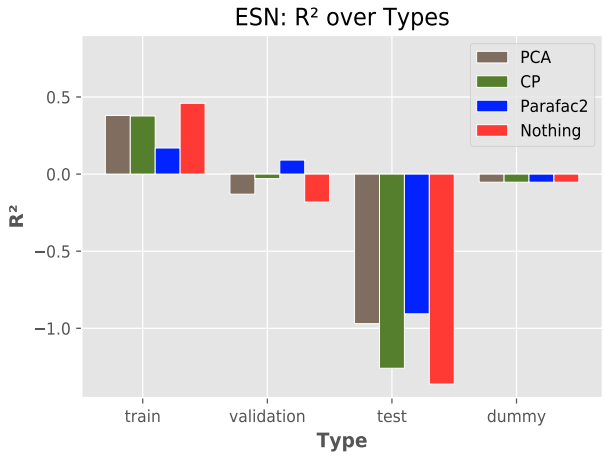

Fig. 6. ESN R^2 scores over different sets.

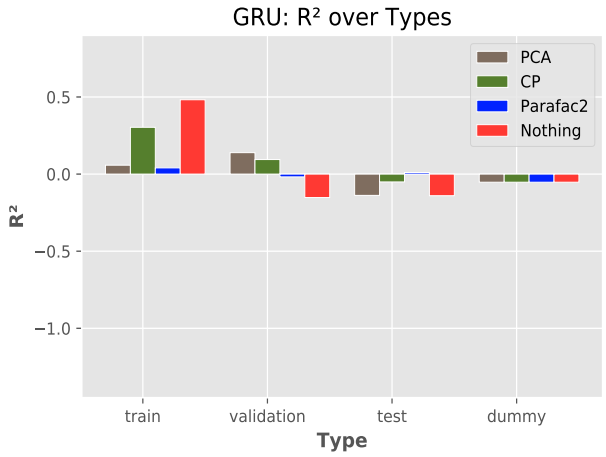

Fig. 7. GRU R^2 scores over different sets.

method could beat the dummy regressor, which always predicts the mean value of the given training labels. It is the cheapest prediction model to apply onto our problem and our model was not able to beat its performance.

B. Gated Recurrent Unit Parameters

Before feeding the data into the GRU, I adapt our reduced video slices by filling up the slices and applying a MinMax-Scaler, similar to the ESN preprocessing. Instead of using the $tanh$, I am now using the $sigmoid$ function for the GRU. Inside the GRU framework of keras, it is also possible to make us of mask values to avoid the impact of our extension of the video slices. During the GRU simulation I am using a batch size of 300, 200 epochs and numbered in the range of 128 to 640 with the stepsize of 128. Additionally to the 200 epochs, I am using the early stopping algorithm from keras in order to shorten the simulation time. The training will stop as soon as no more significant changes in the validation loss occur from training the model. Due to the structure of the GRU, an additional ridge regression of the output of the network is no longer needed, compared to the experiment with the ESN. The results of the valence prediction can be seen in Figure 7. Similiar to the Table 1, Table 2 does contain the optimal rank of each feature selection method combined with the GRU, developed during the trainig of the network.

	PCA	CP	Parafac2
GRU Rank:	22	13	22

TABLE II. CHOSEN RANK AFTER TRAINING THE NETWORK, WHICH RESULTS IN THE HIGHEST R^2 SCORE ON THE VALIDATION SET.

It strikes the fact that CP results into the best fit of the training data, compared to the other decompositoins. But similar to the ESN, all methods do result in a worse fit of the training data than applying 'Nothing'. Nevertheless, the R^2 scores onto the test set, created by the CP and PCA, are again worse than the dummy predictor. Parafac2 is the only feature selection method that results in a slightly better R^2 score on the test set than the dummy regressor. Unfortunately the prediction is still not near to a good predictive performance.

V. Discussion

Discussing the results of both experiments, there is a slight increase of the R^2 scores onto the test sets with prior feature selection, compared to the case 'Nothing', where there is no prior transformation applied. This leads to the assumption that non valuable information has been removed with the prior algorithms. Unfortunately, most results of the test set are worse than the dummy regressor, except of the GRU with Parafac2. The dummy regressor is the cheapest solution and should be seen as lowest benchmark. The fact that the model itself could not beat the performance of the dummy regressor in most cases is questioning what essence can actually be derived from our experiments.

VI. Conclusion

I have evaluated the performance of several features selection method as preprocessing of the emotion prediction with an Echo State Network and a Gated Recurrent Unit. With the different feature selection methods, I was still not able to enhance the prediction performance of each framework significantly. One reason for this could be the fact that the experiments ran with only 3000 samples. The relatively low number of video samples were chosen due to the corresponding size of the tensor and to shorten the overall processing time. Nevertheless, the Parafac2 decomposition fulfilled the first impression, that considering the varying lenghts of our LIRIS ACCEDE dataset within the feature selection, may result in a better performance. In both cases, whether it be with the GRU or with the ESN, the Parafac2 R^2 scores of the test set are slightly better than it's competitors. But this might also be the case, because the Parafac2 decomposition result into the worst reconstruction error, considering the rank range between 1 and

30. This results into a worse fit of the training and a better fit onto the test data. This could also be the reason for the slightly better results of Parafac2, of course relative to it's competitors. However, in both cases, with the ESN and with the GRU, the model ends up with bad predictions of the valence values and therefore emotion prediction with the LIRIS ACEEDE dataset.

REFERENCES

[1] S. Wang and Q. Ji, "Video affective content analysis: A survey of state-of-the-art methods," *IEEE Transactions on Affective Computing*, vol. 6, no. 4, pp. 410–430, 2015.

[2] A. Hennerkes, "Predicting the emotional impact of videos using echo state networks," Chair of Data Processing, Technical University of Munich, Tech. Rep., 2019.

[3] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, vol. 3, no. 3, pp. 127–149, 2009.

[4] Y. Baveye, J.-N. Bettinelli, E. Dellandréa, L. Chen, and C. Chamaret, "A large video database for computational models of induced emotion," in *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*. IEEE, 2013, pp. 13–18.

[5] N. Kwak and C.-H. Choi, "Input feature selection for classification problems," *IEEE transactions on neural networks*, vol. 13, no. 1, pp. 143–159, 2002.

[6] A. Stegeman and N. D. Sidiropoulos, "On kruskal's uniqueness condition for the candecomp/parafac decomposition," *Linear Algebra and its applications*, vol. 420, no. 2-3, pp. 540–552, 2007.

[7] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[8] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[9] C. Keimel, "Design of video quality metrics with multi-way data analysis," Ph.D. dissertation, Chair of Data Processing, Technical University of Munich, 2013.

[10] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise reduction in speech processing*. Springer, 2009, pp. 1–4.

[11] C. Gallicchio, A. Micheli, and L. Pedrelli, "Comparison between deepesns and gated rnns on multivariate time-series prediction," *arXiv preprint arXiv:1812.11527*, 2018.

[12] F. M. Bianchi, S. Scardapane, S. Løkse, and R. Jenssen, "Reservoir computing approaches for representation and classification of multivariate time series," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[13] H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks-with an erratum note," *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, vol. 148, no. 34, p. 13, 2001.

[14] D. Prokhorov, "Echo state networks: appeal and challenges," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 3. IEEE, 2005, pp. 1463–1466.

[15] G. K. Venayagamoorthy and B. Shishir, "Effects of spectral radius and settling time in the performance of echo state networks," *Neural Networks*, vol. 22, no. 7, pp. 861–863, 2009.