TECD: A Transformer Encoder Convolutional Decoder for High-Dimensional Biomedical Data

Luca Zedda, Alessandra Perniciano, Andrea Loddo(✉), and Barbara Pes

Department of Mathematics and Computer Science, University of Cagliari, via Ospedale 72, 09124 Cagliari, Italy
{luca.zedda,alessandra.pernician,andrea.loddo,pes}@unica.it

Abstract. In recent years, machine learning and deep learning methods have been increasingly explored in a variety of application domains, including healthcare. Despite the rapid advances in this field, several challenges still need to be addressed to properly model complex biomedical datasets, such as genomic datasets or physiological signals from wearable sensors, that exhibit a very high dimensionality, i.e., a high number of variables or features which can be mutually related. As evidenced by the literature, the induction of reliable predictive models becomes intrinsically harder as the data dimensionality increases. To give a contribution to this field, this paper explores a new deep learning approach that leverages the emerging paradigm of *Transformers*, which can capture long-range dependencies among the input features and combines them with a *Convolutional Neural Network*, which is suited for capturing local patterns and dependencies. The resulting architecture has shown very promising results on six biomedical datasets with high dimensionality (several thousands of features), paving the way for further research in this area.

Keywords: Biomedical data · High-dimensional data · Machine learning · Deep learning · Transformers

1 Introduction

Machine learning (ML) and deep learning (DL) have revolutionized the way data is analyzed and interpreted and have, indeed, been applied to a wide range of fields such as computer vision (CV), natural language processing (NLP), and sensor-based human activity recognition (HAR). These approaches are also increasingly being exploited in the context of healthcare data analytics, e.g., for improving the clinical decision-making process and diagnostic purposes, with a fast-growing number of reported applications [8,16].

However, as the dimensionality of data increases, the performance of traditional machine learning algorithms can degrade significantly, making it quite

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023 Workshops, LNCS 14104, pp. 225–240, 2023.
https://doi.org/10.1007/978-3-031-37105-9_16

challenging to develop robust predictive models that can be reliably deployed in the real world. More specifically, high dimensionality refers to the presence in the data of a considerable number of variables (or *features*), which can even greatly exceed the number of data records available, as is often the case in healthcare applications [2]. This can lead to the issue of *overfitting*, i.e., to the development of models that, while perfectly describing the data used to train the machine learning algorithm, do not generalize well when applied to new, previously unseen data.

In recent years, a large amount of research has focused on devising suitable approaches to deal with high-dimensional biomedical datasets, investigating learning methods that scale well on high-dimensional feature spaces [17,43] as well as adequate dimensionality reduction techniques, such as *feature extraction* and *feature selection* [30]. Furthermore, high dimensionality often comes with other issues embedded in the nature of the data, including class imbalance and noise, which require proper strategies at both pre-processing and learning stages [14,26]. Several approaches have been proposed to deal with such complex datasets, including hybrid learning strategies that combine feature selection with data-balancing or cost-sensitive approaches [27,40]. As well, methods have been devised for enriching data by ontological annotations that can reveal valuable semantic relationships among the features in order to improve the variable selection process and the classification itself [9,13].

To address the challenges associated with high-dimensional biomedical data, some works have also started to explore deep learning methods that have the potential to learn very complex non-linear relationships among the features [33, 35], albeit at a higher computational cost. In the presence of datasets with many more features than instances, hybrid models have also been investigated based on a suitable combination of feature selection and deep neural networks [19].

In this paper, we propose a new deep learning approach for classifying high-dimensional biomedical data based on a *Transformer Encoder Convolutional Decoder* architecture. Our approach, hereinafter referred to as TECD, combines the strengths of *transformers* and *convolutional neural networks* (CNNs) to effectively capture complex relationships that may be hidden in datasets containing a large number (on the order of thousands) of features.

The proposed TECD architecture has been validated in two different biomedical case studies. Specifically, the first study involved five genomic datasets for cancer diagnosis, while the second study focused on a high-dimensional dataset for Covid-19 detection. Our method obtained promising results in both studies, showing satisfactory performance also in the presence of imbalanced data distributions.

The remainder of this paper is organized as follows. In Sect. 2, an overview of the background concepts used in this study and an exploration of related work are provided. Following this, Sect. 3 presents the proposed architecture and its functional components. Section 4 illustrates the materials and methods used in this work, focusing on the presentation of the genomic and Covid-19 detection datasets. Next, Sect. 5 presents the experimental setup, the results obtained with

the associated findings, and a comparison with state-of-art machine learning methods. Finally, Sect. 6 gives the concluding remarks and outlines future work.

2 Background Concepts and Related Work

In the last decade, the impact of DL techniques has grown exponentially, revolutionizing several areas of machine intelligence, including CV [6], NLP [24], automatic speech recognition [18], drug discovery and toxicology [41], recommender systems [11], or medical analysis [4], to name a few.

Since the proposal of AlexNet's Convolutional Neural Network (CNN) [20], CNNs have begun to achieve outstanding performance in the ImageNet contest, making them one of the most notable DL approaches in the CV field. However, deep approaches are not limited to CNNs but include other techniques such as Multi-Layer Perceptron (MLP), Recurrent Neural Networks (RNN) and its variants, deep generative or unsupervised learning networks (e.g., GANs) [32].

In addition, one of the most recent approaches is certainly the transformer [37]. It uses the self-attention mechanism, differentially weighing the importance of each part of the input data. Transformer's success in NLP enabled researchers to explore its applicability to other fields, such as CV [10], using it for several tasks. Examples include object detection [5], image classification [29], or image segmentation [42].

In medicine and healthcare, pathologists have a keen interest in computer-aided diagnosis (CAD) systems because they can extract meaningful information from clinical and imaging data, provide personalized patient care by analyzing the patient's medical history, symptoms, and tests, or detect and classify diseases [7,39]. It is in this context that our approach is inspired. In fact, encoder-decoder architectures are widely used for various image segmentation tasks. FCN [23] introduces the encoder-decoder architecture and significantly improves the segmentation performance, while U-Net [31] and its evolution build on the idea of FCN by introducing a U-shaped network with lateral connections between the downsampling path and the upsampling path, which propagate context information for better localization. Although upsampling has been widely used in semantic segmentation to restore the low-resolution feature maps obtained from the downsampling path [21], more recently, attention mechanisms have improved the performance by weighting the channels before outputting them to the next layer.

Our work also exploits the concept of attention, but instead of focusing on the pixels of an image or using traditional feature selection approaches, it aims to refine tabular feature data within an encoding-decoding architecture while reducing computation through a feature selection strategy inherent to the architecture itself. Although a few works have recently tried out deep learning approaches to dealing with high-dimensional data matrices in the biomedical field [33], including methods based on CNNs [19,35], our approach is highly innovative as it jointly exploits the potential of transformers and CNNs. To our knowledge, no

previous work has investigated a similar DL approach to model complex high-dimensional biomedical data, such as the genomic datasets and the Covid-19 dataset used in this study.

3 The Proposed Architecture

This section presents the proposed approach and the basic building blocks. Specifically, Sect. 3.1 and Sect. 3.2 describe two key concepts underlying the proposed architecture, i.e., attention mechanisms and transformers, respectively, while the CNN is briefly illustrated in Sect. 3.3. Finally, the structure of the architecture proposal is presented in its entirety in Sect. 3.4.

3.1 Attention Mechanism

As mentioned above in Sect. 2, there has been considerable interest in attention mechanisms in recent years for NLP or CV tasks since the introduction of the transformer architecture and multi-head self-attention (MHSA) [37].

The primary goal of attention mechanisms is to focus on relevant parts of the input while ignoring irrelevant parts, allowing the model to selectively attend to specific parts of the input. More in detail, the MHSA formulation is represented by Eq. (1):

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W^O \tag{1}$$

where:

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{2}$$

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

Here, Q, K, and V represent the query, key, and value matrices. W_i^Q, W_i^K, and W_i^V represent the learned projection matrices for the ith head. W^O represents the learned output projection matrix. h is the number of heads. d_k is the dimensionality of the key vectors.

3.2 Transformers

Transformers [37] are a type of neural network architecture that aims to capture long-range dependencies between input sequences, allowing the model to better understand the context and meaning of the input.

Indeed, traditional neural networks, such as RNNs and CNNs, have been widely used for sequence modeling tasks. However, these networks have limitations in handling long sequences due to the vanishing gradient problem and the limited receptive field of CNNs. Transformers address these limitations using a self-attention mechanism to capture dependencies between all positions in the

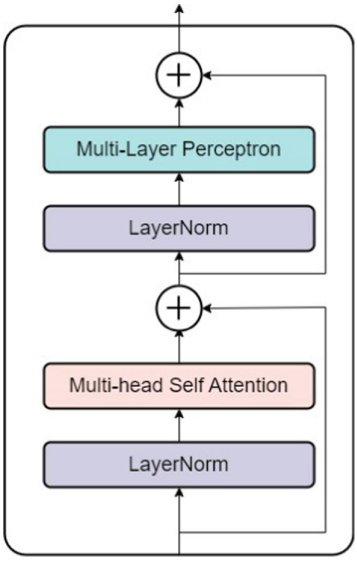

Fig. 1. A schematic illustration of a transformer block. The block takes a sequence of input tokens and produces a sequence of transformed tokens. A rectangle with its name represents each layer of the block. The block has three main components: a multi-head self-attention layer (in pink), a feed-forward layer (in turquoise), and two residue connections with layer normalization (in lilac). (Color figure online)

input sequence. The self-attention mechanism computes a weighted sum of the input based on the similarity between each input token and each other token, allowing the model to attend to relevant parts of the input at each processing step. A schematic representation of the transformer block architecture is depicted in Fig. 1.

While transformers were initially developed for sequential modeling tasks in NLP, they have also been applied to tabular data, where they have achieved state-of-the-art performance in various machine learning tasks. Recently, novel deep learning approaches, such as TabNet [1] and TabFormer [25], have aimed to exploit the transformer architecture for tabular data applications.

3.3 Convolutional Neural Networks (CNNs)

CNNs are a class of artificial neural networks that rely on a mathematical operation called convolution to automatically extract key features from the input data. They also use a proper regularization method to deal with the problem of over-fitting, which may occur in a fully connected network [32].

CNNs were initially designed for 2D shapes and used extensively in image classification and visual recognition tasks. However, they can also be applied to tabular data, where they can effectively capture local dependencies between features.

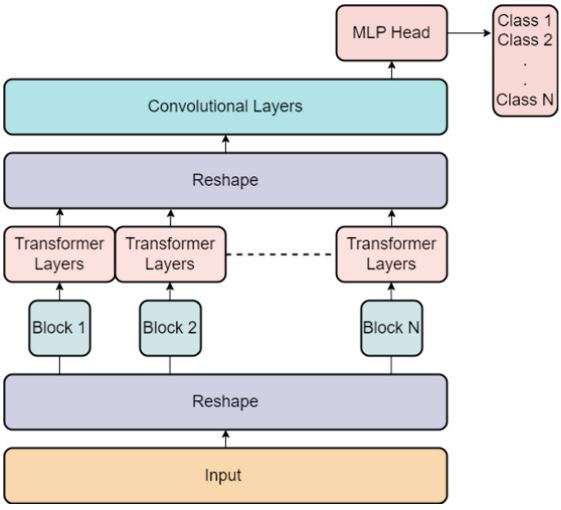

Fig. 2. A schematic illustration of the proposed TECD architecture that relies on an encoder-decoder structure. The encoder, composed of transformer layers, computes attention by reweighting features in relation to each other. At the same time, the convolutional decoder serves as a dimensional reduction tool or to enrich the input data through convolution operations.

In particular, 1D-CNNs are effective for tabular data classification [34]. 1D-CNNs apply a sequence of 1-dimensional convolutional operations to the input data, which can capture local patterns and dependencies between adjacent features. This makes them suitable for tasks where the order of features may be important.

3.4 TECD: Transformer Encoder Convolutional Decoder

Our proposed approach involves preprocessing the input data by padding it to form sequences divisible by a predetermined block size. Each block is passed through a transformer encoder, and the resulting outputs are merged. This merged output is then given to a CNN consisting of n residual blocks. Each residual block reduces the dimensionality of the input using an average pooling layer, and the final output is passed through a linear classification layer to generate predictions for the number of classes.

More specifically, the above process can be formalized as follows. Let X be the input data and B be the predetermined block size. We pad X so that its length is a multiple of B, resulting in a new sequence X'. We then divide X' into m B-sized blocks, denoted as $X_1, X_2, ..., X_m$, where m is the number of blocks. Each block X_i is then passed through a transformer encoder to generate an output tensor H_i. The output tensors $H_1, H_2, ..., H_m$ are then merged using a concatenation operation along the sequence dimension, resulting in a single output tensor H.

The output tensor H is then passed through a convolutional neural network (CNN) consisting of n residual blocks, denoted as $ResBlock_1$, $ResBlock_2$, ..., $ResBlock_n$. Each residual block reduces the dimensionality of the input using an average pooling layer. The final output is passed through a linear classification layer to generate predictions for the number of classes.

A schematic illustration of the proposed architecture is shown in Fig. 2.

4 Materials and Methods

We validated our innovative approach on two challenging case studies: the first refers to cancer classification from genomic data, while the second refers to Covid-19 detection from blood test data. For each case study, we conducted an extensive experimental investigation comparing the performance of our TECD architecture with that of state-of-the-art machine learning methods widely used in biomedical data analysis. In the following, we provide an overview of the datasets employed in our experiments (Sect. 4.1) and of the machine learning algorithms included in the comparative study (Sect. 4.2).

4.1 Datasets

Genomic Datasets. Gene expression data has proven to be very useful for various clinical applications, such as the diagnosis and prognosis of several types of cancer. However, one of the most serious problems in the analysis of such kind of data is the extremely low instances-to-features ratio, being the number of samples (i.e., biological tissues) very small compared to the number of features (i.e., genes). In this work, we exploit one of the largest publicly available repositories of gene expression data, namely the *Gene Expression Machine Learning Repository* (GEMLeR), that was collected by the International Genomics Consortium [36]. Specifically, we used the five datasets reported in Table 1, characterized by an instances-to-features ratio of 0.14: although low, this ratio is higher than that reported in most studies in the literature [15].

Each dataset refers to a binary classification task that discriminates a specific type of tumor (coded as a positive class) against other types of tumors (negative class). In particular, we chose five datasets that exhibit different levels of class imbalance, as shown in Table 1, to explore the performance of our approach across different data distributions. Precisely, the *Breast* and *Colon* datasets are moderately imbalanced, with 22% and 19% of minority instances, respectively. Instead, the *Uterus* and *Omentum* datasets present more skewed data distributions. In contrast, the *Ovary* dataset has a degree of imbalance that is in the middle of the others.

Covid-19 Detection Dataset. This dataset, which represents a very challenging case study [22], contains information from blood samples collected through a blood scanner. It was published by Hilab[1], a laboratory company based in

[1] https://hilab.com.br/.

Table 1. List of datasets from GEMLeR repository used in this work. The table indicates the number of instances and features of every dataset, along with the percentage of minority instances available.

Dataset	No. of Instances	No. of Features	Minority Instances
Breast	1545	10936	22%
Colon	1545	10936	19%
Ovary	1545	10936	13%
Uterus	1545	10936	8%
Omentum	1545	10936	5%

Brazil that operates thousands of blood scanner points throughout the country, primarily in hospitals and pharmacies.

The digitization technology of the devices and the large number of tests performed have generated enough data to create a significant benchmark for machine learning and deep learning tasks. Indeed, the dataset consists of 40,044 instances equally distributed into two classes representing positive or negative samples for Covid-19, as labeled by expert biomedical scientists. Each piece is described by 12,210 numerical features, resulting in a classification problem with very high dimensionality.

4.2 Machine Learning Methods

To validate our TECD architecture and demonstrate its effectiveness in biomedical data classification, we performed a comparative analysis involving the following machine learning algorithms:

- *Bayesian Network* (BayesNet);
- *Naïve Bayes* (NB);
- *Support Vector Machine* (SVM);
- *k-Nearest Neighbor* (k-NN);
- *Ripper* (JRip);
- *One Rule* (OneR);
- *Decision Tree* (J48);
- *Random Forest* (RF).

We observe that the methods considered are representatives of different families of classifiers [22]. Indeed, BayesNet and NB are probabilistic models that represent the conditional dependency relationships among the variables involved in the task. SVM, known to scale well to high-dimensional feature spaces, exploits a statistical learning framework to model linear decision boundaries and more complex boundaries in dependence on the adopted kernel function. The k-NN algorithm is an instance-based learner that assigns the class to unknown instances based on their similarity to the training records. JRip and OneR are rule-based

classifiers that induce simple yet effective models in the form of *if-then* prediction rules. Finally, J48 and RF are tree-based classifiers: in particular, J48 builds a single decision tree exploiting the well-known C4.5 method originally proposed by Quinlan [12]. At the same time, RF is an ensemble method combining multiple decision tree predictions.

5 Experimental Study

In this section, we summarize our experimental investigation, first presenting the experimental settings (Sect. 5.1) and then discussing the results obtained (Sect. 5.2).

5.1 Experimental Setup

The experiments were performed on a workstation with an NVIDIA GeForce RTX 3060 GPU and 32 GB of RAM. The configuration of both types of experiments is described below.

Experimental Setup for TECD. To ensure reproducibility, we set the random seed for data partitioning and model initialization. The experiment was repeated 5 times. In each run, the dataset was partitioned using the 80-10-10 rule: 80% for training, 10% for validation, and the remaining 10% for testing. In particular, in each run, the best model was selected based on the performance of the validation set; this model was then evaluated on the test of that particular run. The final performance was obtained by averaging the results of the five runs.

The training phase lasted a maximum of 250 epochs, with the following hyperparameters: learning rate of 1×10^{-5}, Lion optimizer [38], cross-entropy loss, a warmup learning rate of 1×10^{-7}, 25 warmup epochs, and batch size of 128. The Lion optimizer is a novel algorithm for training deep neural networks proposed by Google Brain using genetic algorithms. It is more memory efficient than Adam because it only keeps track of the moment.

Experimental Setup for Machine Learning Algorithms. For the machine learning algorithms, we leveraged the implementations provided by the WEKA library [12]. For a fair comparison, we used the same experimental protocol for the TECD architecture, i.e., 5 training-testing runs (with 10% of the data reserved for testing at each run).

The settings adopted for the classifiers correspond, in most cases, to the default parameters provided by the WEKA library. In particular, we used a linear kernel for the SVM method. At the same time, the RF classifier was implemented with 100 trees and $log_2(N) + 1$ random features (where N is the dimensionality, i.e., the number of features, of the dataset at hand). For the k-NN approach, we considered two versions of the method: with and without instance weighting; in both cases, the number of nearest neighbors (i.e., the k parameter) was set to 5 to reduce the risk of overfitting. In what follows, we will refer to the weighted version of k-NN as k-NNw.

Furthermore, since the performance of machine learning algorithms can degrade in the presence of a large number of features, we also conducted experiments in which the data dimensionality was reduced by applying proper feature selection methods. Specifically, we used a ranking-based selection approach [3] to assign weights to the features based on their correlation with the target class; then, a subset containing the first 110 top-ranked features (1% of the original dimensionality) was employed for model induction. For a broad comparison, we experimented with different feature ranking criteria, as implemented by the WEKA library, namely *Information Gain* (IG), *Symmetrical Uncertainty* (SU), *Gain Ratio* (GR), *Pearson's Correlation* (Corr), and *ReliefF*, which have proven to be effective in the context of biomedical data analysis [3,28].

5.2 Experimental Results and Discussion

For our evaluation, we considered two metrics: the F-measure and the accuracy. Specifically, the F-measure is the harmonic mean of precision and recall (see Eq. (4)), where precision is the ratio of true positives to the total predicted positives and recall is the ratio of true positives to the total actual positives.

$$F - measure = \frac{2 * precision * recall}{precision + recall}, \tag{4}$$

The F-measure is commonly used in information retrieval and binary classification in the presence of imbalanced datasets. For this reason, we considered the F-measure for evaluating the experiments conducted on the genomic datasets, which present an imbalanced class distribution, as shown in Table 1.

On the other hand, accuracy is defined as the ratio of correct predictions (true positives and true negatives) to the total number of cases, as reported in Eq. (5):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{5}$$

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives. Since the accuracy considers both classes equally, it is well suited for performance evaluation in the presence of balanced data distributions, as in the Covid-19 detection dataset considered here.

Results on the GEMLeR Datasets. Table 2 summarizes the results of the first comparative analysis conducted on the five GEMLeR datasets included in our study (*Breast*, *Colon*, *Ovary*, *Uterus*, and *Omentum*). Specifically, the performance of our TECD architecture is compared with that of the nine machine learning methods described in Sect. 4.2.

As can be seen, our proposed approach performs robustly across all genomic benchmarks considered. Precisely, in four out of five cases (i.e., in *Breast*, *Colon*, *Uterus*, and *Omentum* datasets), it achieved the highest F-measure value, while in the *Ovary* dataset it was outperformed only by SVM and JRip. Thus, overall,

Table 2. Results obtained on the GEMLeR datasets in terms of F-measure.

Dataset	TECD	BayesNet	NB	SVM	KNN	KNNw	OneR	JRip	J48	RF
Breast	**0.93**	0.91	0.90	0.90	0.82	0.81	0.79	0.81	0.85	0.87
Colon	**0.96**	0.85	0.88	0.89	0.87	0.87	0.85	0.88	0.88	0.85
Ovary	0.68	0.60	0.58	0.72	0.50	0.50	0.61	**0.74**	0.67	0.47
Uterus	**0.72**	0.47	0.57	0.60	0.62	0.62	0.18	0.48	0.50	0.24
Omentum	**0.53**	0.44	0.49	0.44	0.39	0.39	0.37	0.41	0.38	0.00

our TECD architecture has proved to be effective in classifying different cancer types based on the patient's genetic profile, despite the low instances-to-features ratio of this kind of data (see Sect. 4.1).

Table 3. Accuracy results obtained on the Covid-19 dataset.

Method	Accuracy
RF [22]	0.74
ANN [22]	0.84
TECD (Our)	**0.86**

Results on the Covid-19 Dataset. In a previous work [22], we already explored the same Covid-19 dataset examined here by conducting an extensive comparative analysis of different machine learning methods, including those considered in this paper, and of different deep learning methods, including 1D-CNN, TabNet, Node as well as an ANN network specially designed for that purpose. Among the machine learning methods, RF was the best-performing one in [22], with an overall accuracy of 0.74. However, the ANN model turned out to be significantly more accurate, with an overall accuracy of 0.84. Such a model, designed after exploring different neural network architectures and settings, is used here as a reference method to evaluate the performance of our new TECD architecture. As seen in Table 3, TECD shows superior performance, confirming it can be a suitable approach for handling complex high-dimensional datasets.

Discussion. Although not exhaustive, the results of our experimental analysis showed that the proposed encoding-decoding architecture has the potential to capture meaningful patterns across different types of biomedical datasets, even in the presence of imbalanced data distributions, with an inherent capability of reducing the input dimensionality. High-dimensional feature spaces represent indeed a primary challenge for most machine learning algorithms, which often require a preliminary preprocessing aimed at reducing the number of input features, especially in the presence of a low instances-to-features ratio, as in the case of the genomic datasets.

On these datasets, as mentioned in Sect. 5.1, we conducted further experiments to evaluate the impact of a preliminary dimensionality reduction on the nine machine learning methods involved in our comparative analysis. Specifically, each classifier was evaluated in conjunction with different feature selection methods, namely IG, Corr, SU, GR, and ReliefF, for a total of 45 selector-classifier combinations. As shown in Table 4, in most cases, the experiments with feature selection showed better performance for the machine learning algorithms. However, the final F-measure may strongly depend on the adopted selection approach, and no single selector-classifier combination performs consistently better than the others.

Table 4. F-measure performance obtained on the GEMLeR datasets using machine learning classifiers in conjunction with different feature selection methods (IG, Corr, SU, GR, ReliefF).

Dataset	IG	Corr	SU	GR	ReliefF	BayesNet	NB	SVM	KNN	KNNw	OneR	JRip	J48	RF
Breast	✓					0.92	0.89	0.91	0.91	0.90	0.79	0.90	0.86	0.92
		✓				0.88	0.80	0.90	0.89	0.88	0.79	0.87	0.85	0.89
			✓			0.90	0.88	0.91	0.90	0.90	0.79	0.89	0.88	0.92
				✓		0.88	0.79	0.89	0.89	0.88	0.79	0.89	0.86	0.91
					✓	0.90	0.85	0.91	0.88	0.88	0.79	0.88	0.86	0.90
Colon	✓					0.87	0.84	0.90	0.89	0.89	0.85	0.91	0.84	0.88
		✓				0.86	0.82	0.89	0.88	0.88	0.85	0.90	0.88	0.88
			✓			0.86	0.84	0.90	0.88	0.88	0.85	0.89	0.86	0.88
				✓		0.86	0.84	0.90	0.89	0.88	0.85	0.90	0.85	0.89
					✓	0.86	0.82	0.89	0.86	0.85	0.85	0.92	0.91	0.91
Ovary	✓					0.71	0.65	0.79	0.68	0.67	0.61	0.77	0.74	0.84
		✓				0.66	0.67	0.77	0.56	0.56	0.61	0.73	0.69	0.81
			✓			0.75	0.66	0.79	0.66	0.65	0.61	0.78	0.75	0.86
				✓		0.76	0.68	0.80	0.54	0.54	0.61	0.80	0.69	0.81
					✓	0.66	0.63	0.82	0.59	0.59	0.61	0.79	0.78	0.79
Uterus	✓					0.55	0.51	0.52	0.42	0.42	0.18	0.47	0.51	0.46
		✓				0.53	0.54	0.51	0.39	0.39	0.18	0.46	0.41	0.50
			✓			0.57	0.52	0.51	0.42	0.42	0.22	0.47	0.54	0.45
				✓		0.56	0.53	0.49	0.43	0.43	0.24	0.55	0.43	0.42
					✓	0.52	0.52	0.45	0.32	0.32	0.21	0.56	0.52	0.48
Omentum	✓					0.50	0.42	0.37	0.43	0.43	0.37	0.53	0.34	0.48
		✓				0.48	0.41	0.31	0.56	0.56	0.37	0.50	0.40	0.42
			✓			0.46	0.40	0.32	0.37	0.37	0.37	0.46	0.38	0.45
				✓		0.47	0.41	0.23	0.29	0.29	0.37	0.41	0.34	0.37
					✓	0.39	0.33	0.39	0.34	0.34	0.37	0.65	0.37	0.53

Overall, comparing the results in Table 4 with those previously shown in Table 2, we can still observe that our TECD architecture turns out to be quite effective overall. Indeed, in three out of five datasets (*Breast*, *Colon*, and *Uterus*) it outperformed each of the 45 selector-classifier combinations, while

in the *Omentum* dataset, it outperformed 40 combinations out of 45; the *Ovary* dataset in the only one, among those examined, in which TECD did not perform significantly better.

Finally, as regards the second case study considered in this paper, related to Covid-19 detection from blood test data, we observe that the instances-to-features ratio is not a critical concern in this case, as a vast number of instances is available along with many features. In a so different setting, as previously observed in Table 3, our TECD architecture also showed promising performance, prompting us to enlarge our experimentation in order to further validate our approach on multiple and heterogeneous scenarios.

6 Conclusions

Computer-aided diagnosis systems are becoming increasingly effective in assisting clinical staff with a wide range of diseases and cancers. Artificial intelligence methods have been successfully applied in this field in recent years. However, several issues still need to be addressed to cope with the inherent complexity of biomedical datasets, which pose multiple challenges, including high dimensionality.

In this work, we presented a new deep learning architecture designed to classify high-dimensional biomedical data. The main innovative contribution of this work is the realization of a novel approach able to combine the strengths of transformers and convolutional neural networks to effectively capture complex relationships that may be hidden in datasets containing a large number of features. In addition, within the encoding-decoding architecture, the concept of attention has been applied, which aims to refine the input data through a feature selection strategy inherent to the architecture itself.

Overall, the experiments performed on six public datasets produced results that are in line with or better than other state-of-the-art methods in the literature, making our proposed TECD architecture an effective contribution to deep learning models in addressing the challenges associated with high-dimensional datasets. In future work, we plan to further validate our approach on different real-world case studies, encompassing various problem settings, e.g., in terms of the number of instances, features, instances-to-features ratio, and class imbalance level.

Acknowledgements. This research was supported by the ASTRID project (Fondazione di Sardegna, L.R. 7 agosto 2007, n°7, CUP: F75F21001220007).

References

1. Arik, S., Pfister, T.: TabNet: attentive interpretable tabular learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 6679–6687 (2021)
2. Berisha, V., et al.: Digital medicine and the curse of dimensionality. NPJ Digit. Med. **4**(1), 153 (2021)

3. Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A.: Feature Selection for High-Dimensional Data. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21858-8
4. Budd, S., Robinson, E.C., Kainz, B.: A survey on active learning and human-in-the-loop deep learning for medical image analysis. Medical Image Anal. **71**, 102062 (2021)
5. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020, Part I. LNCS, vol. 12346, pp. 213–229. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_13
6. Chai, J., Zeng, H., Li, A., Ngai, E.W.: Deep learning in computer vision: a critical review of emerging techniques and application scenarios. Mach. Learn. Appl. **6**, 100134 (2021)
7. D'Ancona, G., et al.: Deep learning to detect significant coronary artery disease from plain chest radiographs AI4CAD. Int. J. Cardiol. **370**, 435–441 (2022)
8. Davenport, T., Kalakota, R.: The potential for artificial intelligence in healthcare. Future Healthc. J. **6**(2), 94–98 (2019)
9. Dessì, N., Pascariello, E., Pes, B.: Integrating ontological information about genes. In: 2014 IEEE 23rd International WETICE Conference, pp. 417–422. IEEE (2014)
10. Dosovitskiy, A., et al.: An image is worth 16×16 words: transformers for image recognition at scale. In: 9th International Conference on Learning Representations, ICLR 2021, 3–7 May 2021, Austria (2021)
11. Elkahky, A.M., Song, Y., He, X.: A multi-view deep learning approach for cross domain user modeling in recommendation systems. In: WWW 2015 (2015)
12. Frank, E., Hall, M.A., Witten, I.H.: The WEKA Workbench. Online Appendix for 'Data Mining: Practical Machine Learning Tools and Techniques', 4th edn. Morgan Kaufmann (2016)
13. Gillies, C.E., Siadat, M.R., Patel, N.V., Wilson, G.D.: A simulation to analyze feature selection methods utilizing gene ontology for gene expression classification. J. Biomed. Inform. **46**(6), 1044–1059 (2013)
14. Gupta, S., Gupta, A.: Dealing with noise problem in machine learning data-sets: a systematic review. Procedia Comput. Sci. **161**, 466–474 (2019)
15. Hambali, M.A., Oladele, T.O., Adewole, K.S.: Microarray cancer feature selection: review, challenges and research directions. Int. J. Cogn. Comput. Eng. **1**, 78–97 (2020)
16. Javaid, M., Haleem, A., Singh, R.P., Suman, R., Rab, S.: Significance of machine learning in healthcare: features, pillars and applications. Int. J. Intell. Netw. **3**, 58–73 (2022)
17. Kalina, J.: Classification methods for high-dimensional genetic data. Biocybern. Biomed. Eng. **34**(1), 10–18 (2014)
18. Kaur, A.P., Singh, A., Sachdeva, R., Kukreja, V.: Automatic speech recognition systems: a survey of discriminative techniques. Multim. Tools Appl. **82**(9), 13307–13339 (2023)
19. Kilicarslan, S., Adem, K., Celik, M.: Diagnosis and classification of cancer using hybrid model based on ReliefF and convolutional neural network. Med. Hypotheses **137**, 109577 (2020)
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. NIPS 2012, vol. 1, pp. 1097–1105 (2012)

21. Li, Y., Cai, W., Gao, Y., Li, C., Hu, X.: More than encoder: Introducing transformer decoder to upsample. In: Adjeroh, D.A., et al. (eds.) IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2022, Las Vegas, NV, USA, 6–8 December 2022, pp. 1597–1602. IEEE (2022)
22. Loddo, A., Meloni, G., Pes, B.: Using artificial intelligence for COVID-19 detection in blood exams: a comparative analysis. IEEE Access **10**, 119593–119606 (2022)
23. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
24. Madsen, A., Reddy, S., Chandar, S.: Post-hoc interpretability for neural NLP: a survey. ACM Comput. Surv. **55**(8), 155:1–155:42 (2023)
25. Padhi, I., et al.: Tabular transformers for modeling multivariate time series. In: ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3565–3569. IEEE (2021)
26. Pes, B.: Learning from high-dimensional biomedical datasets: the issue of class imbalance. IEEE Access **8**, 13527–13540 (2020)
27. Pes, B.: Learning from high-dimensional and class-imbalanced datasets using random forests. Information **12**, 286 (2021)
28. Pes, B., Lai, G.: Cost-sensitive learning strategies for high-dimensional and imbalanced data: a comparative study. PeerJ Comput. Sci. **7**, e832 (2021)
29. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
30. Ray, P., Reddy, S.S., Banerjee, T.: Various dimension reduction techniques for high dimensional data analysis: a review. Artif. Intell. Rev. **54**(5), 3473–3515 (2021). https://doi.org/10.1007/s10462-020-09928-0
31. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
32. Sarker, I.H.: Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. SN Comput. Sci. **2**(6), 420 (2021)
33. Sevakula, R.K., Singh, V., Verma, N.K., Kumar, C., Cui, Y.: Transfer learning for molecular cancer classification using deep neural networks. IEEE/ACM Trans. Comput. Biol. Bioinf. **16**(6), 2089–2100 (2019)
34. Shwartz-Ziv, R., Armon, A.: Tabular data: deep learning is not all you need. Inf. Fusion **81**, 84–90 (2022)
35. Singh, R., Lanchantin, J., Robins, G., Qi, Y.: DeepChrome: deep-learning for predicting gene expression from histone modifications. Bioinformatics **32**(17), i639–i648 (2016)
36. Tiglic, G., Kokol, P.: Stability of ranked gene lists in large microarray analysis studies. J. Biomed. Biotechnol. **2010** (2010)
37. Vaswani, A., et al.: Attention is all you need. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4–9 December 2017, Long Beach, CA, USA, pp. 5998–6008 (2017)
38. Yazdani, M., Jolai, F.: Lion optimization algorithm (LOA): a nature-inspired metaheuristic algorithm. J. Comput. Design Eng. **3**(1), 24–36 (2016)

39. Zedda, L., Loddo, A., Di Ruberto, C.: A deep learning based framework for malaria diagnosis on high variation data set. In: Sclaroff, S., Distante, C., Leo, M., Farinella, G.M., Tombari, F. (eds.) ICIAP 2022. LNCS, vol. 13232, pp. 358–370. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-06430-2_30
40. Zhang, C., Zhou, Y., Guo, J., Wang, G., Wang, X.: Research on classification method of high-dimensional class-imbalanced datasets based on SVM. Int. J. Mach. Learn. Cybern. **10**, 1765–1778 (2019)
41. Zhavoronkov, A., et al.: Deep learning enables rapid identification of potent DDR1 kinase inhibitors. Nat. Biotechnol. **37**(9), 1038–1040 (2019)
42. Zheng, S., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, 19–25 June 2021, pp. 6881–6890. Computer Vision Foundation/IEEE (2021)
43. Zou, H.: Classification with high dimensional features. Wiley Interdisc. Rev.: Comput. Stat. **11**(1), e1453 (2019)