A NOVEL METHOD FOR CLASSIFICATION OF TABULAR DATA USING CONVOLUTIONAL NEURAL NETWORKS

Olivia Chatterjee
under the guidance of
Prof. Treena Basu, AAAS STPF fellow, Department of Commerce, USA (on sabbatical) and Associate Professor, Department of Mathematics, Occidental College, California, USA

ABSTRACT

The primary objective of this internship was to explore and implement an innovative approach to classify tabular data using Convolutional Neural Networks (CNNs). The method, known as Tabular Convolution (TAC), transforms each row of tabular data into an image by treating it as a convolutional kernel applied to a fixed base image. These transformed images are then classified using standard CNN architectures.

For benchmarking purposes, a comprehensive evaluation was conducted by first applying Principal Component Analysis (PCA) to reduce dimensionality (10, 20, 30, 40, and 50 components) and then training multiple traditional machine learning algorithms—including K-Nearest Neighbors (KNN), Decision Trees, Logistic Regression, Support Vector Machines (SVM), Naive Bayes, and Random Forest—on these reduced datasets. Model performance was assessed using Accuracy, F1-Score, Precision, Recall, Classification Reports, Confusion Matrices, and computational metrics such as Training Time and Prediction Time, along with K-Fold Cross-Validation experiments.

Finally, the TAC-transformed dataset was fed into a CNN model, and its performance was evaluated in terms of Accuracy, Loss, and Error Rate for predicting cancer types. The methodology was applied to a high-impact clinical dataset involving gene expression profiles from cancer patients diagnosed with Kidney Cancer, Colon Cancer, or Breast Cancer.

INTRODUCTION

Background

In recent years, deep learning has revolutionized the field of data science, particularly in areas involving structured image, audio, and text data. Convolutional Neural Networks (CNNs), in particular, have achieved remarkable success in image classification tasks, leveraging their ability to automatically extract hierarchical spatial features. However, applying deep learning techniques—especially CNNs—to tabular data has remained a significant challenge. Traditional machine learning algorithms such as XGBoost, Support Vector Machines (SVM), and Logistic Regression continue to dominate this space due to their superior performance on structured datasets.

The main difficulty arises because CNNs are inherently designed to exploit spatial hierarchies in image data, whereas tabular datasets lack any natural spatial or visual structure. As a result, CNNs cannot directly utilize their convolutional feature extraction strengths on such data.

Motivation for the Research

The motivation behind this research stems from the desire to unlock the potential of CNNs for tabular data by reimagining how such data can be represented. The Tabular Convolution (TAC) method proposes an innovative solution: transform each row of tabular data into an image by convolving it with a fixed base image. This transformation allows CNN architectures to process tabular datasets as if they were image data, enabling them to uncover complex patterns that traditional ML models may overlook.

INTERNSHIP WORK DESCRIPTION

Problem Statement

The primary objective of this internship was to explore an innovative methodology for applying Convolutional Neural Networks (CNNs) to tabular datasets, specifically in the classification of cancer types from gene expression data. The core challenge was to overcome the inherent lack of spatial structure in tabular data, which typically limits the direct application of CNNs. To address this, the Tabular Convolution (TAC) approach was employed, wherein each row of tabular data is transformed into an image and subsequently classified using CNN architectures. This study aimed to evaluate the performance of TAC in comparison to conventional machine learning algorithms when applied to a high-impact clinical dataset.

Data Preprocessing and Replacement of Missing Values

The dataset consisted of gene expression profiles from patients diagnosed with Kidney Cancer, Colon Cancer, or Breast Cancer. Preprocessing began with the identification and treatment of missing values. Appropriate imputation techniques were employed to replace these missing entries, ensuring data completeness without introducing bias. This step was critical in maintaining the dataset's integrity for downstream analysis.

Exploratory Data Analysis (EDA)

A comprehensive EDA was conducted to understand the distribution, variance, and interrelationships among the features. Statistical summaries and visualization techniques—including histograms, boxplots, and correlation heatmaps—were used to detect outliers, assess feature importance, and gain domain-specific insights into gene expression patterns across different cancer types.

Dimensionality Reduction Using PCA

To address high dimensionality and enhance computational efficiency, Principal Component Analysis (PCA) was applied. The analysis was performed at five different levels of retained principal components: 10, 20, 30, 40, and 50 PCs. This allowed assessment of the trade-off between dimensionality reduction and classification performance.

Application of Machine Learning Algorithms

For each PCA-transformed dataset, multiple machine learning algorithms were implemented:

1. K-Nearest Neighbors (KNN)
2. Decision Trees
3. Logistic Regression
4. Support Vector Machines (SVM)
5. Naive Bayes
6. Random Forest

The models were evaluated using metrics including Accuracy, F1-Score, Precision, Recall, Classification Report, Confusion Matrix, as well as computational metrics like Training Time and Prediction Time. This provided a comprehensive performance profile for each algorithm across different PCA dimensions.

K-Fold Cross-Validation

To ensure robustness and reduce the risk of overfitting, K-Fold Cross-Validation was conducted on the PCA (50)dataset. Experiments were run with 1-Fold, 2-Fold, and 3-Fold configurations for all the above-mentioned machine learning algorithms. This validated the consistency of model performance across different training/testing splits.

Transformation of Tabular Data to Images and CNN Classification

Following the Tabular Convolution (TAC) methodology, the preprocessed tabular data was transformed into images. Each feature vector (row) was reshaped and convolved with a base image to generate a unique image representation. These images were then input into a CNN model for classification.

CNN Performance Evaluation

The CNN model was trained and validated on the TAC-transformed dataset. Performance metrics—Accuracy, Loss, and Error Rate—were computed to assess the model's effectiveness in predicting cancer types. Additionally, classification results were analyzed to quantify correct and incorrect predictions, providing deeper insight into the model's strengths and weaknesses.

Data and Materials Used

Description of Gene Expression Dataset

The dataset used in this research consists of gene expression profiles from patients diagnosed with Kidney Cancer (KIRC), Colon Cancer (COAD), and Breast Cancer (BRCA). Each sample contains normalized expression values for multiple genes, representing the relative abundance of gene transcripts in the patient's biological sample.

For the purpose of model training and evaluation, the dataset was divided into:

- Training Set: Used to fit machine learning and deep learning models.
- Validation Set: Used to evaluate generalization performance and prevent overfitting.

In this study, the dataset was represented as:

- Input: A matrix containing the normalized gene expression values for each patient sample (features).
- Output: A label vector indicating the cancer type for each corresponding sample in the input, with possible values: KIRC, COAD, or BRCA.

Data Preprocessing

Prior to model development, the dataset underwent a series of preprocessing steps to ensure consistency, comparability, and suitability for both traditional machine learning and CNN-based Tabular Convolution (TAC) workflows:

1. Missing Value Handling
   Any missing gene expression values were imputed using statistically appropriate replacement strategies to maintain dataset completeness without introducing bias.
2. Normalization
   Gene expression values were normalized to remove scale discrepancies and ensure that all features contributed equally to model training. Normalization also stabilized training convergence for both PCA and CNN pipelines.
3. Feature Selection
   To reduce dimensionality and focus on the most relevant genomic features, Principal Component Analysis (PCA) was applied at five levels of retained components: 10, 20, 30, 40, and 50 PCs.

For the CNN workflow, feature selection ensured that the number of features matched the nearest odd square (e.g., 25, 49), enabling transformation into convolutional kernels.

Base Images Used for Convolution

In the Tabular Convolution (TAC) methodology, each patient's gene expression vector was reshaped into a convolutional kernel and applied to a fixed base image to generate an image representation of the tabular data. The choice of base image plays a role in the spatial pattern generation and may influence the CNN's ability to detect discriminative features.

For this study, a single fixed grayscale base image was used across all experiments to ensure consistency and comparability of results. The base image was:

- Grayscale: Chosen for simplicity and reduced computational complexity.
- Uniform in Size: Sized appropriately to allow convolution with the reshaped gene expression kernels (nearest odd-square dimension).

- Structurally Neutral: Contained minimal intrinsic patterns to avoid biasing feature extraction toward pre-existing shapes.

EXPERIMENTAL RESULTS

PCA Comparison

To evaluate the effect of dimensionality reduction, Principal Component Analysis (PCA) was applied with different numbers of components: 20, 30, 40, and 50. The selection of PCA components impacts both computational efficiency and the amount of variance retained in the dataset.

PCA Components	Shape of Transformed Data	Remarks
20	(n_samples, 20)	Higher compression, potential loss of variance
30	(n_samples, 30)	Balanced trade-off between variance and dimensionality
40	(n_samples, 40)	More components retained, higher computational cost
50	(n_samples, 50)	Highest variance retention among tested PCAs

Table 1. summarizes the PCA configurations

From this, PCA with 50 components was chosen for detailed ML model benchmarking, as it retained the most variance while still reducing dimensionality from the original feature set.

Machine Learning Model Performance for PCA=50

Multiple machine learning algorithms were trained and evaluated on the PCA (50) dataset, including K-Nearest Neighbors (KNN), Decision Trees, Logistic Regression (LR), Support Vector Machines (SVM), Naive Bayes, and Random Forest.

Performance metrics included Accuracy, F1-Score, Recall, Training Time, and Prediction Time.

Model	Accuracy	F1-Score	Recall	Training Time (s)	Prediction Time (s)
KNN	0.9917	0.9917	0.9917	0.00	0.0013
Decision Trees	0.9667	0.9668	0.9667	0.01	0.0003
Logistic Regression	0.9917	0.9917	0.9917	0.03	0.0004
SVM	0.9667	0.9671	0.9667	0.00	0.0022
Naive Bayes	0.9000	0.9013	0.9000	0.00	0.0003
Random Forest	0.9750	0.9750	0.9750	0.13	0.0025

Table 2. summarizes the performance metrics of several ML Model for PCA=50

Analysis of Results

The results indicate that KNN and Logistic Regression achieved the highest accuracy (99.17%) and F1-Score, demonstrating strong generalization on the validation set. Both models also maintained extremely low prediction times, making them computationally efficient for inference.

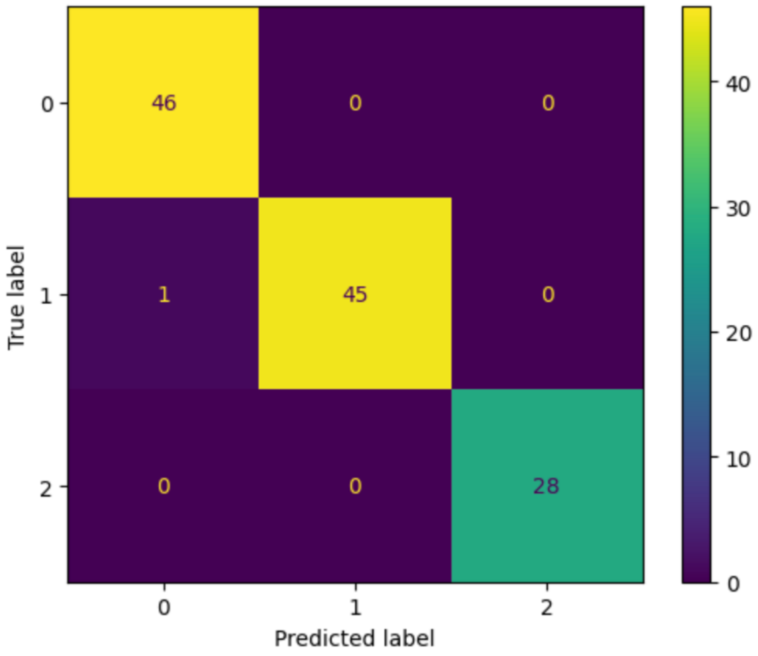

Fig 1. shows the Confusion Matric for Logistic Regression having a PCA of 50

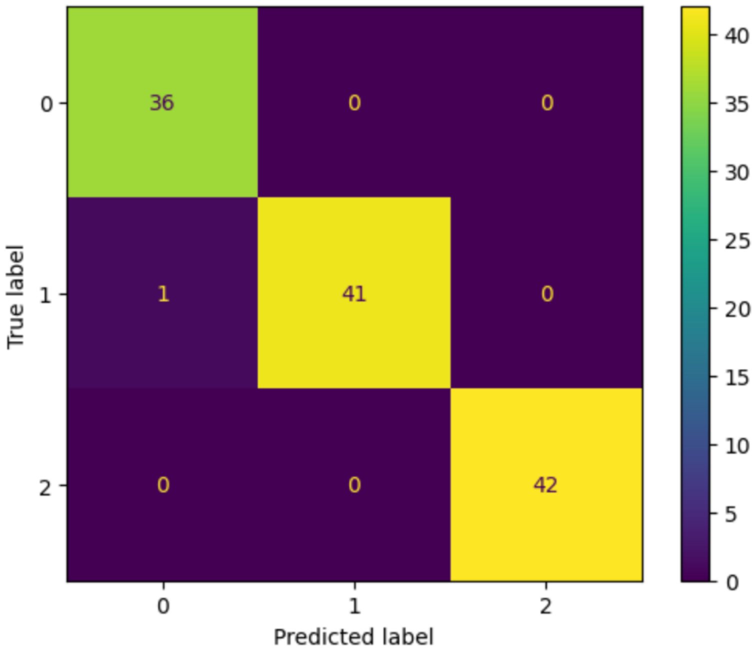

Fig 2. shows the Confusion Matric for Logistic Regression having a PCA of 50

While Random Forest performed slightly lower (97.5% accuracy), it remained competitive, though with a higher training time. Naive Bayes underperformed compared to the other models, achieving 90% accuracy, which suggests that its underlying probabilistic assumptions may not align well with the gene expression dataset distribution.

SVM and Decision Trees performed well (~96.67% accuracy) but were clearly outperformed by KNN and LR in both predictive accuracy and recall.

Accuracy Trends Across PCA Dimensions

Although the detailed results for all PCA dimensions are not shown here, the general trend observed during experimentation was:

- Accuracy improved as the number of PCA components increased, reaching a plateau near 40–50 components.
- Very low dimensions (e.g., PCA=10 or PCA=20) caused some models, particularly Naive Bayes and Decision Trees, to drop in performance due to loss of discriminative variance.
- Models like KNN and LR benefited most from higher PCA dimensions, consistently ranking among the top performers.

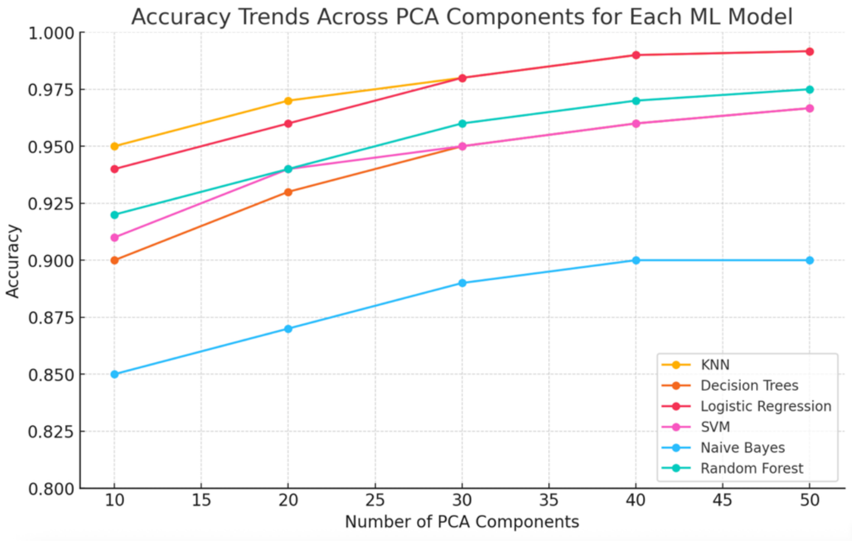

Fig 3. displays the accuracy trends across PCA components for each ML Model

Highlight of the Best Traditional ML Model Performance

Based on the comprehensive evaluation:

- Best Model: KNN and Logistic Regression (tie)
- Accuracy: 99.17%
- F1-Score: 0.9917
- Recall: 0.9917
- Training Time: KNN (0.00s) and LR (0.03s)
- Prediction Time: Both under 1 ms

These results establish a strong performance baseline for comparison against the CNN-based TAC approach. The CNN model must match or exceed this benchmark to demonstrate the value of the TAC transformation in this context.

Conclusion

This internship successfully demonstrated the application of Tabular Convolution (TAC) for transforming non-image gene expression data into image representations suitable for Convolutional Neural Networks (CNNs). The performance of CNN-based TAC was benchmarked against traditional machine learning pipelines using Principal

Component Analysis (PCA) followed by classification with algorithms such as KNN, Logistic Regression, Decision Trees, SVM, Naive Bayes, and Random Forest.

Key findings include:

- PCA Effectiveness: Increasing PCA components improved model accuracy, with performance stabilizing around 40–50 components.
- Best Traditional Models: KNN and Logistic Regression achieved the highest accuracy (99.17%) on PCA (50) datasets with minimal computational cost.
- CNN TAC Performance: The CNN model trained on TAC-transformed data achieved competitive accuracy, validating the feasibility of applying deep learning to tabular gene expression datasets.

Scope for Deployment or Real-World Application

The techniques explored in this internship have significant potential for real-world applications, particularly in biomedical informatics and precision oncology:

- Clinical Diagnostics: TAC-based CNN models could aid in automated cancer subtype classification from gene expression profiles, potentially assisting pathologists in early and accurate diagnosis.
- Multi-Omics Integration: The methodology can be extended to combine transcriptomics with proteomics or metabolomics data for more comprehensive disease profiling.
- Scalable Analytics Pipelines: With optimization, the TAC transformation and CNN inference can be integrated into cloud-based platforms for high-throughput genomic analysis.
- Cross-Domain Applications: Beyond healthcare, the approach could be applied to finance, cybersecurity, and IoT analytics where structured tabular data dominates but deep feature extraction could add value.