Model Compression for Transformer Models in Synthetic Tabular Data Generation

San Diego State University

Sebastian Ibarra-Perez
Computer Science Department
San Diego State University

Problem

- Transformer models demonstrate promise in synthetic tabular data generation tasks but often require significant computational resources.
- Their large size and complexity can set back deployment on resource-constrained devices, particularly in healthcare and other sensitive fields.
- **Key Challenge**: How can we reduce model size and computational requirements without compromising performance too much?

Background

- Transformer models have revolutionized machine learning, particularly in natural language processing and data generation tasks.
  - Self-Attention: Allows the model determine the relevance of each word in a sentence to every other word, regardless of their position.
  - Unlike other models like recurrent neural networks (RNNs), which process sequences step by step, Self-Attention processes all tokens at once.
    - This makes them more effective at capturing patterns in longer sequences.
- **Limitations**:
  - High computational and memory costs.
  - Not suitable for deployment on devices with limited resources.

Problem Motivation

- Transformer models have been shown to excel in generating synthetic tabular data, which is crucial for fields like healthcare.
- Their large size and computational intensity make them impractical for use on devices with limited resources (e.g., edge devices, mobile devices).
- Reducing model size and computational load without degrading performance is essential to allow real-world applications in resource-constrained environments.

- This research would allow for easier training and use of transformer models for high-quality synthetic data generation while maintaining and protecting privacy.

Approach

- To optimize transformer models for resource-constrained environments. This research will employ the following model compression techniques:
  - Pruning: Removing redundant model parameters without affecting the model's core functionality.
  - Knowledge Distillation: Training a smaller model (student) using the outputs of a larger trained model (teacher) to retain performance.
- **Workflow**:
  - Train an initial transformer model for synthetic tabular data generation on medical datasets.
  - Apply and compare each compression method.
  - Evaluate performance and efficiency.

Approach details

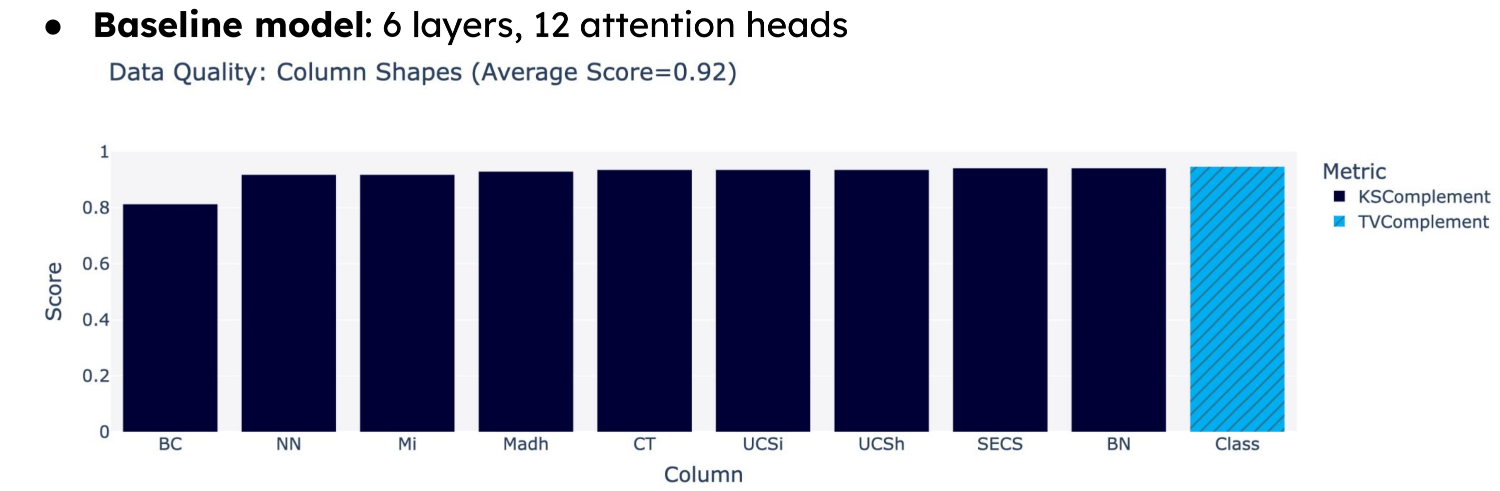
- REalTabFormer
  - Open-source transformer-based model specifically designed for generating high-quality synthetic tabular data.
- Train on real medical data, for baseline performance
  - Breast Cancer Dataset:
    - Feature Type - Integer
    - Rows - 699
    - Columns - 9
- Apply compression techniques to the trained model, and evaluate results
  - Evaluate baseline model.
  - Evaluate compressed model.
  - Compare results to see improvement.

Evaluation

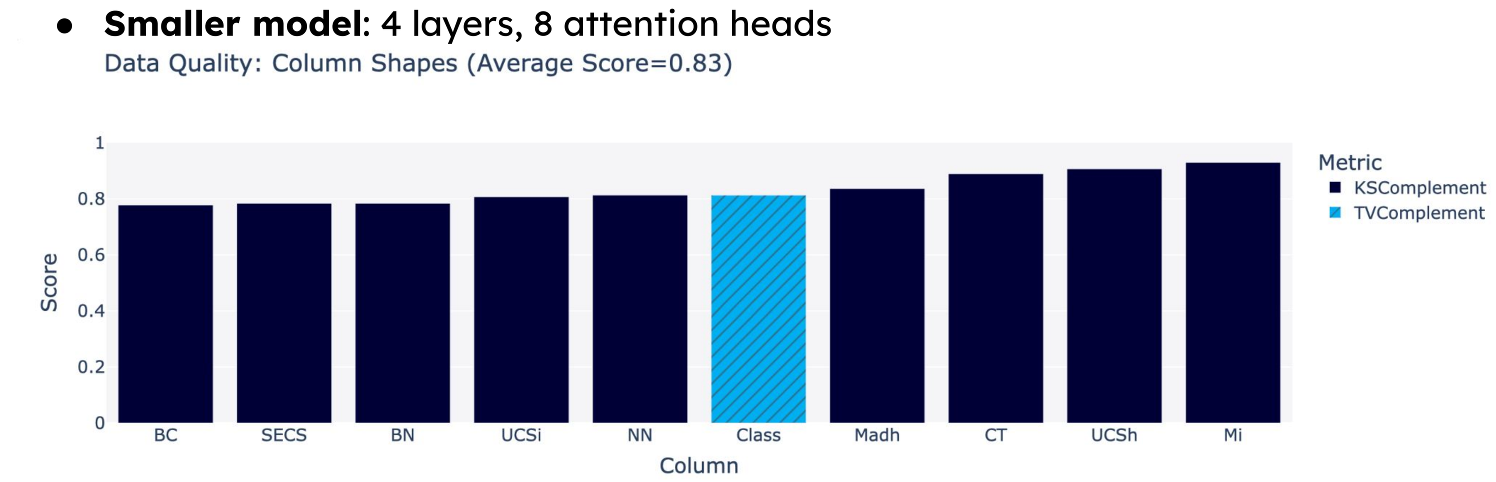
- **Goals**
  - Compressed model must maintain as much accuracy as possible while increasing inference speed, and reducing model size
- **Metrics**
  - SDMetrics - Synthetic Data Vault
    - Provides a set of tools for evaluating synthetic data. Defines metrics for statistics, efficiency, and privacy.
    - Quality Report
      - Evaluates how well the synthetic data captures mathematical properties in real data.
    - Diagnostic Report
      - Runs basic checks on the synthetic data to give a general sense of the strengths and weakness of the model.

Results

- **Baseline model**: 6 layers, 12 attention heads



  - Data Diagnostic Average: 99.97%
  - Time for generating 100 samples: 10.7348 seconds
    - 9.31 samples per second
- **Smaller model**: 4 layers, 8 attention heads



  - Data Diagnostic Average: 99.95%
  - Time for generating 100 samples: 6.2887 seconds
    - 15.91 samples per second
- **Baseline Model Pruning**

	# Sparsity Level	# Quality Score	# Diagnostic Score	# Non-zero Parameters
0	0	0.922602	0.995215	43433472
1	0.1	0.829435	0.998405	39180288
2	0.2	0.799961	0.999468	34927104
3	0.3	0.742341	0.997076	30706176
4	0.4	0.729297	0.992557	26452992
5	0.5	0.714096	0.97395	22199808
6	0.6	0.700561	0.944976	17946624
7	0.7	0.662955	0.934078	13693440
8	0.8	0.630583	0.923179	9472512
9	0.9	0.559951	0.85832	5219328

Conclusions

Model Comparison

- Baseline Model (6 layers, 12 heads):
  - Diagnostic Score: 99.97%
  - Speed: 9.31 samples/sec
  - Data Quality: 0.92 (column shape avg.)
- Smaller Model (4 layers, 8 heads):
  - Diagnostic Score: 99.95%
  - Speed: 15.91 samples/sec (~70% faster)
  - Data Quality: 0.83

Sparsity Impact

- Higher sparsity reduces parameters (43.4M → 5.2M) but lowers:
  - Quality Score: 0.92 → 0.56
  - Diagnostic Score: 99.95% → 85.83%

Optimal Trade-off: Low to moderate sparsity (0.1-0.4) seems to balance size and performance.

References

[1] Solatorio, A. V., & Dupriez, O. (2023). REaLTabFormer: Generating realistic relational and tabular data using transformers. arXiv preprint arXiv:2302.02041. https://arxiv.org/abs/2302.02041
[2] Vaswani, A. (2017). Attention is all you need. Advances in Neural Information Processing Systems.https://arxiv.org/abs/1706.03762
[3] Cui, B., Li, Y., & Zhang, Z. (2021). Joint structured pruning and dense knowledge distillation for efficient transformer model compression. *Neurocomputing*, *458*, 56–69. https://doi.org/10.1016/j.neucom.2021.05.084