Characterizing Quantum Classifier Utility in Natural Language Processing Workflows

Kathleen E. Hamilton, Mayanka Chandra Shekar, Eduardo Antonio Coello Pérez, Prasanna Date, John Gounley, In-Saeng Suh, Georgia Tourassi

Oak Ridge National Laboratory , Oak Ridge ,TN

Dhanvi Bharadwaj

University of Wisconsin

Madison , Madison ,WI

Motivation

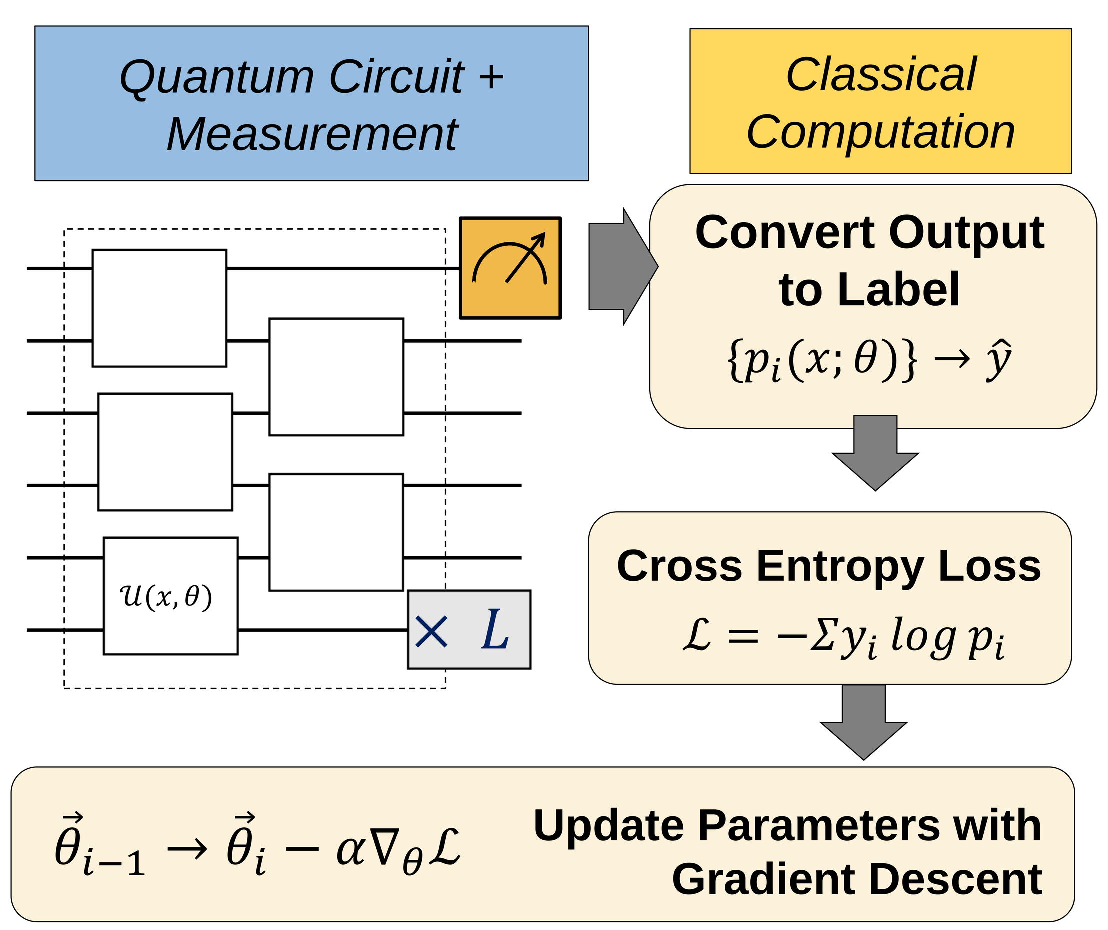
Quantum machine learning has benefitted from adapting neural network designs into quantum circuit analogues and leveraging classical training. But several challenges remain when designing QML models for training over **large**, **unbalanced**, and **high-dimensional** spaces.

We explore the effects of model complexity, feature selection, and task complexity on binary and multiclass classification performance.

As QML matures into a technology for real-world applications: our results provide empirical evidence that there are many open questions in feature selection, structure and embedding.

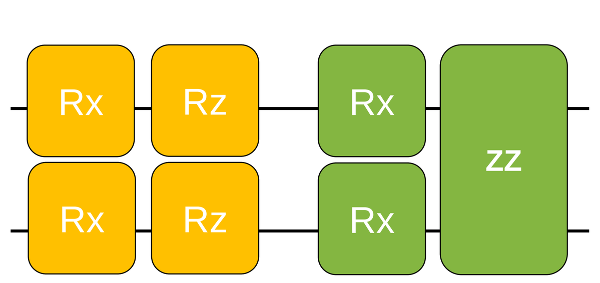
Quantum Classifier Design and Model Complexity

Circuit simulation and model training workflow implemented in PennyLane

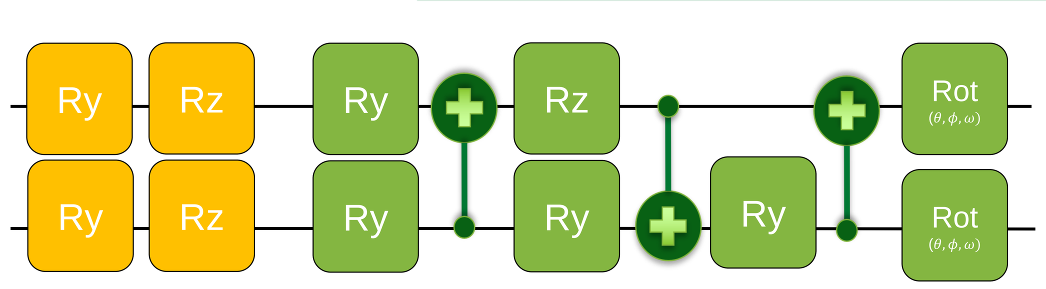


- Each circuit on **(n)** qubits is built from **(L)** layers of **(n-1)** two-qubit unitaries
- Each circuit uses data re-uploading [1]
- Measurements used to predict labels
- Supervised learning trains parameters via gradient descent using ADAM ($\alpha = 0.05$)
- Number of epochs: 16, 32

Unitary Designs



IsingZZ: 4 embedding, 3 trainable parameters



SU(4): 4 embedding [yellow], 11 trainable parameters [green], CNOT gates fixed

hamiltonke@ornl.gov **dhanvibharad@gmail.com**

[1] Pérez-Salinas, Adrián, et al. Quantum 4 (2020): 226.

[2] Guyon, Isabelle. (2008). *Madelon*. UCI Machine Learning Repository.

[3] Yoon, Hong-Jun, et al. 2019 IEEE International Conference on Big Data (Big Data).

The research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

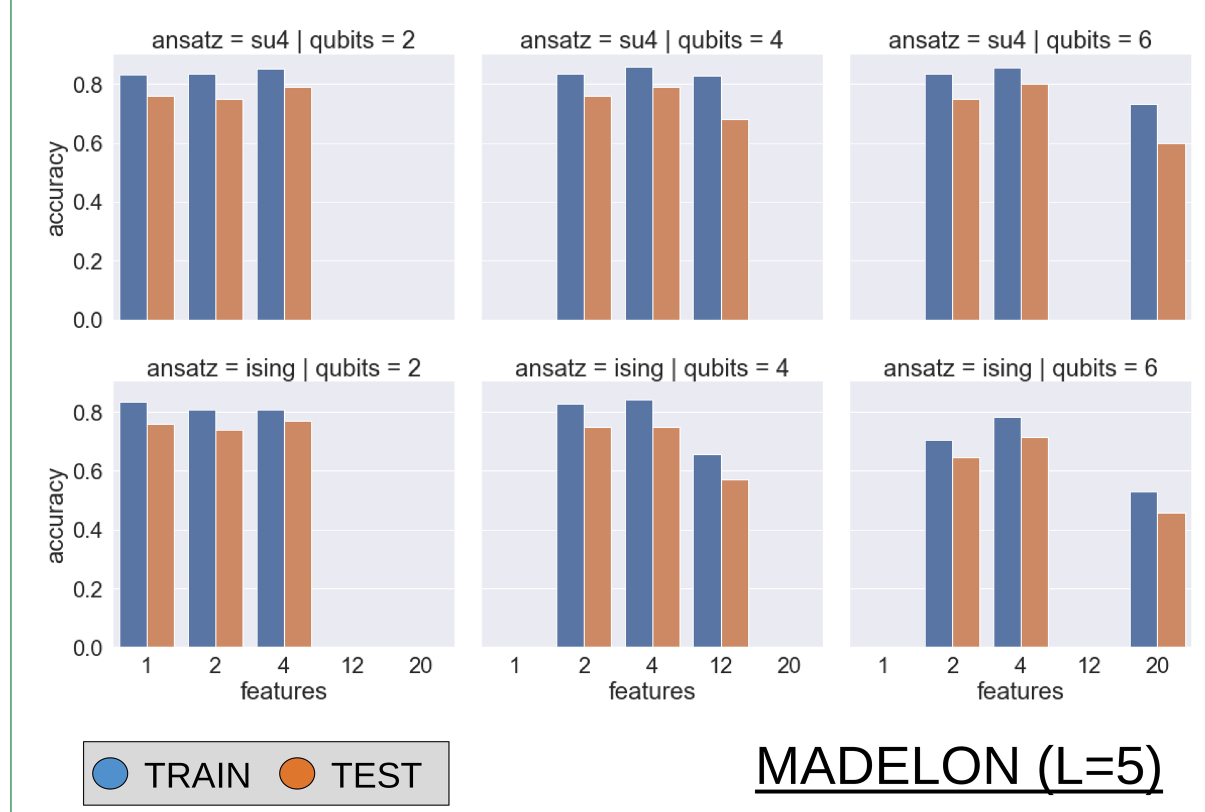
This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User supported under Contract DE-AC05-00OR22725.

Model Complexity

Increasing $\mathcal{U}(x,\theta)$ complexity does not guarantee improved training performance, but **Increasing Depth** often does

Increasing (n) increases the embedding capacity up to 4(n-1) unique features per layer

However: this entangles random and informative features- a likely cause of performance drop when many unique features are used



MADELON (L=5)

Feature Selection

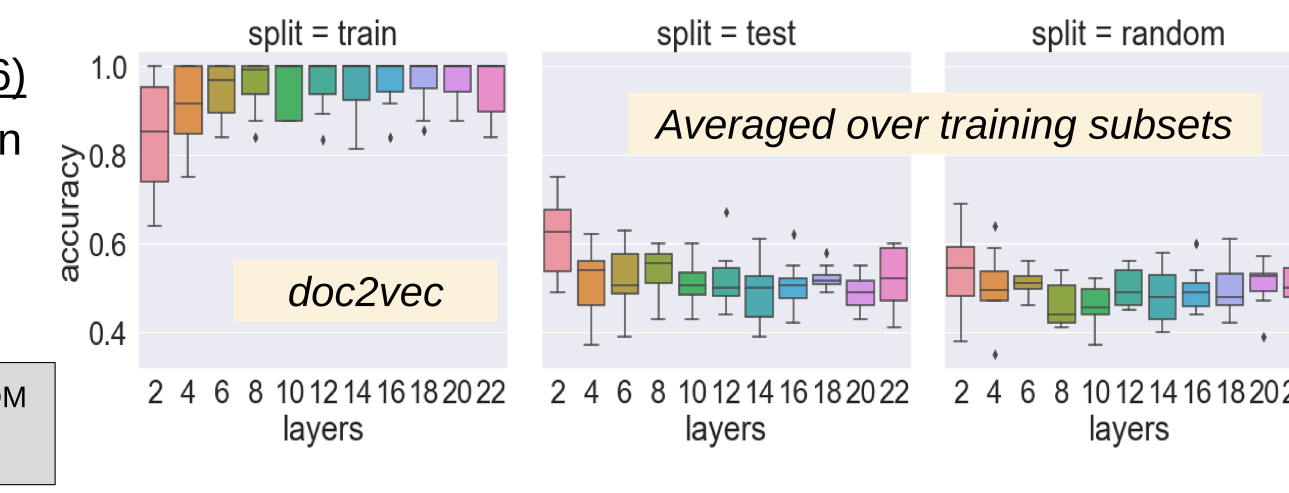
- Mutual information methods extract the top **K** important features, rescaled to $[0, 2\pi]$ or $[0, \pi]$
- **Rescaling** tokenized P3B3 features has low overhead, **but** will discard semantic information
- **Trained embeddings** (doc2vec) maintains complex features, **but** this may be lost by naïve reshaping of feature vectors for data re-uploading

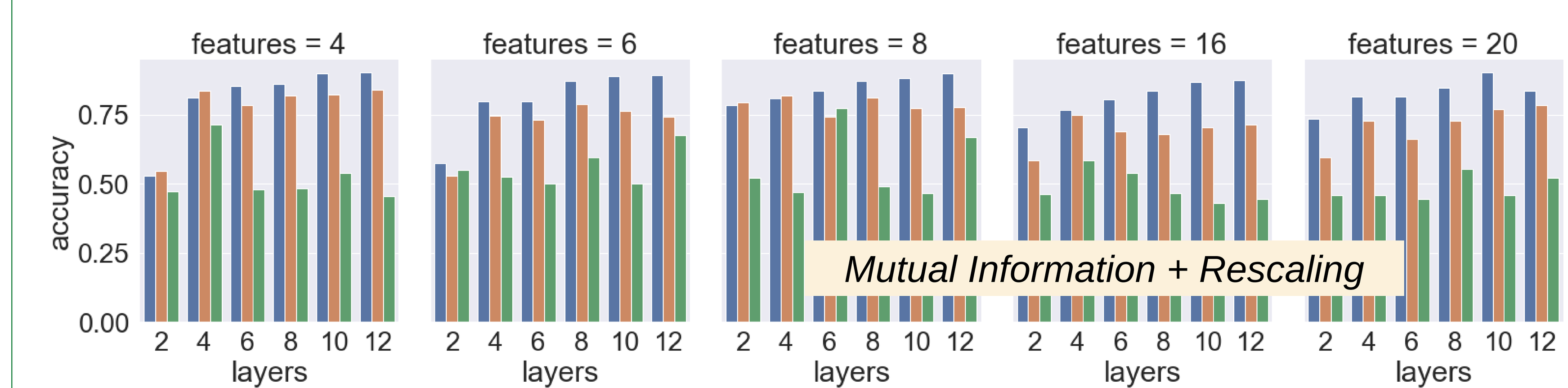
P3B3: IsingZZ (n=6)

Binary classification

$|X_{train}| = 661$

$|X_{test}| = 97$





Task Complexity

MADELON [2]

Balanced, binary labels

- **Unstructured:** 2 informative, 2 redundant, 16 uninformative features

Generated in scikit-learn

Pilot3 (P3B3) [3]

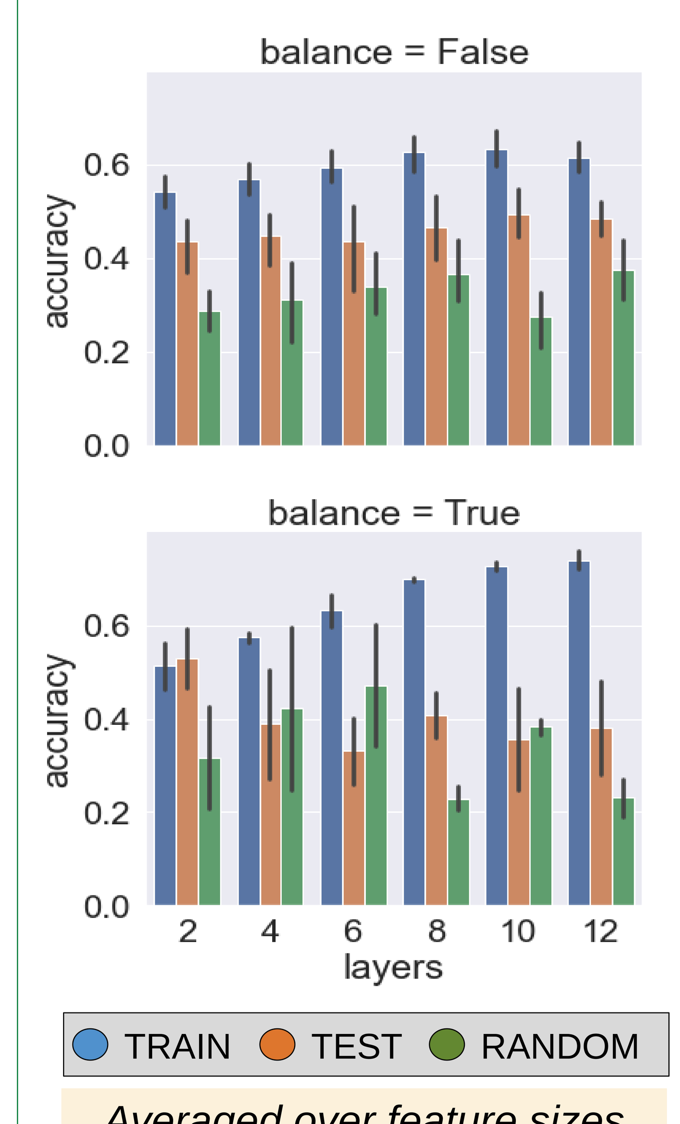
- Unbalanced, multiple labels
- 1500 total features
- **Structured:** enumerated and padded token list of features

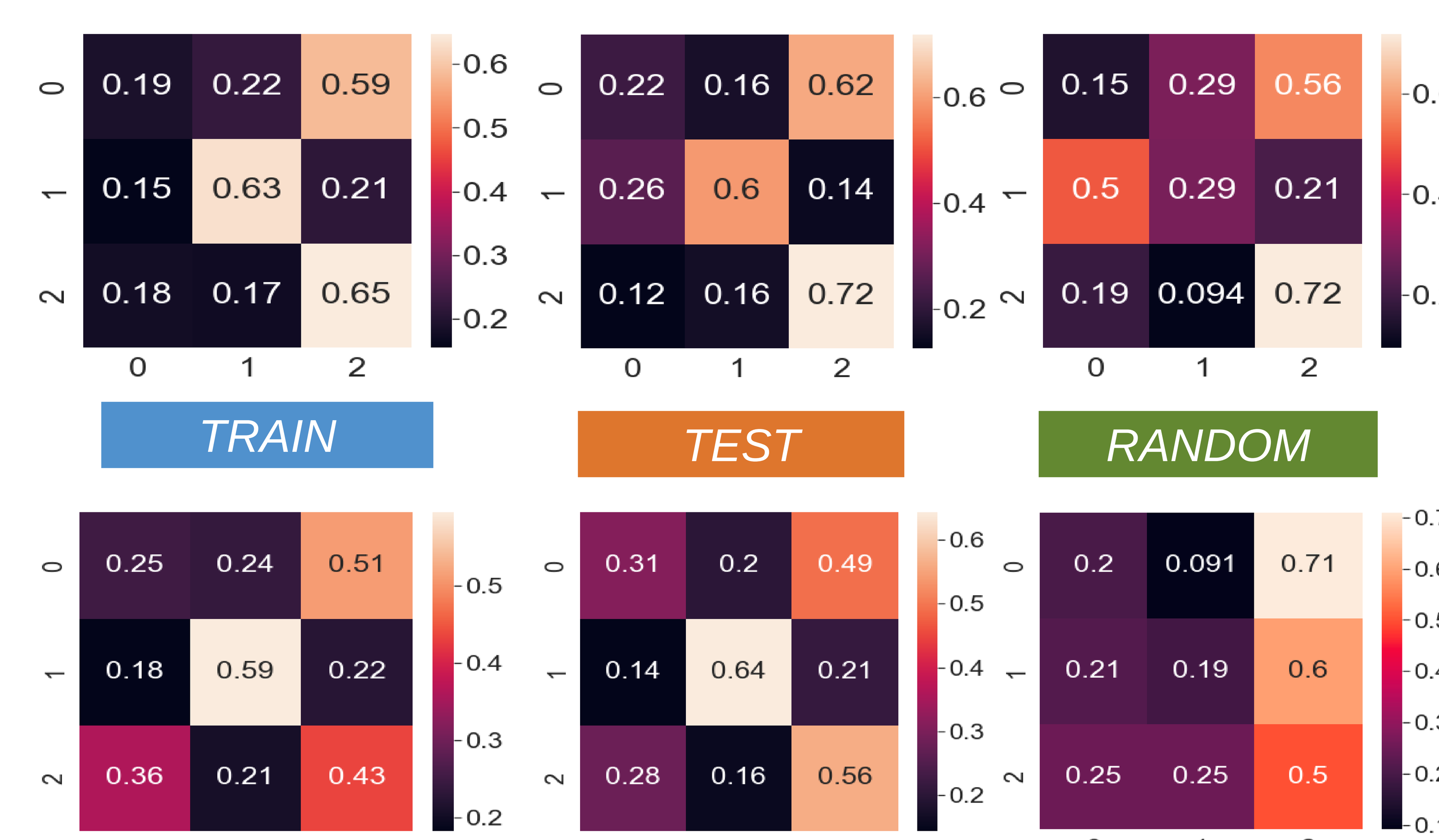
Synthetic cancer pathology reports

Binary Classification expands to Multiclass by using one-hot encoded labels, implemented by down-selecting on low-weight bitstrings – which may explain the random model performance

Class imbalance addressed by: relative sample weighting, or under-sampling to construct balanced classes

(Top): *Unbalanced training data* $|X_{train}| = 1148\ (487{:}\,375{:}\,286)$





(Bottom): *100 samples per class (balanced)*

P3B3: IsingZZ (L=12, n=6)

Multiclass Classification $|X_{test}| = 127\ (55{:}\,42{:}\,32)$

20 features rescaled to **half-circle** $[0, \pi]$