# Quantum Machine Learning for Anti-Microbial Classification in Essential Oil Components: Challenges and Advancements using QGANs

Anmol Chaure[1], Ashok Kumar Behera [2], Sudip Bhattacharya[3], Chaitali Choudhary[4], Aishwarya Chaure[5]
[1]*M.Tech. Scholar, CSE Department, BIT Durg*
[2,3]*Associate Professor, CSE Department, BIT Durg*
[4]*Assistant Professor, CSE Department, BIT Durg*
[5]*PhD Scholar, Faculty of Tropical Agrisciences, Czech University of Life Sciences Praha*

***Abstract*— Essential oils have gained widespread recognition for their ability to combat microorganisms, serving as natural alternatives to synthetic antimicrobial agents. However, the intricate nature of essential oil components and their interactions with microorganisms pose challenges in identifying the most potent anti-microbial elements. Traditional screening methods are costly, time-consuming, and yield limited success. This study delves into the potential and limitations of utilizing quantum machine learning in the field of drug discovery. The use of quantum machines stems from the fact that chemical systems are governed by quantum mechanics, which are challenging to simulate with classical computers By leveraging this approach, it becomes feasible to uncover potential anti-microbial agents within essential oils, thus contributing to the development of novel and efficient natural products with antimicrobial properties. This research paper, examines the feasibility of loading essential oil data using Generative Adversarial Networks into quantum machine learning algorithms for classifying the essential oil components based on their anti-microbial properties.**

***Index Terms*— *Quantum machine learning, Generative adversarial networks, Ethnobotany, Bioinformatics***

## I. INTRODUCTION

Dependency on plant-based medicines is on the rise as we are entering the era of antimicrobial resistance crisis. The reason for this can be traced to both the excessive and inappropriate use of drugs, as well as a decline in the development of new medications by the pharmaceutical industry due to lower financial incentives and more rigorous regulatory demands. [12]

The complexity of essential oil components and their interactions with microorganisms makes it challenging to identify the most effective anti-microbial agents. Traditional methods for screening essential oils are often time-consuming and expensive, and can result in limited success in identifying potential antimicrobial components. The process usually starts with the collection of plants for the process of oil extraction, which amounts to a time and labor-intensive task. Following this already long procedure, the process of oil extraction (known as distillation) is performed which involves the use of extensive machineries and expensive lab setup, that are not readily available. After going through these efforts, the results can have a huge deviation from the desirable one. [3]

In recent years, generative adversarial networks (GANs) have emerged as a promising approach for solving complex classification problems in various fields, including bioinformatics and drug discovery. QGANs are a type of GAN that leverage the power of quantum computing to generate synthetic data that is representative of the true data distribution. By training a QGAN on a dataset of essential oil components, we can generate synthetic samples that can be used to classify the anti-microbial properties of essential oil components.

The choice of a quantum model is motivated by the fact that the behavior of chemical systems is fundamentally determined by the principles of quantum mechanics, which are notoriously difficult to simulate using classical computers. To address this challenge, researchers have estimated the computational resources required to solve typical

problems in quantum chemistry, such as running time and qubit cost. Chemistry provides an ideal context for exploring the theoretical and practical implications of insights from quantum information. Moreover, researchers have identified a connection between the molecular properties of substances like essential oils and the cost of quantum simulation.[5]. Estimating these resources is a critical initial stage for an fair comparison between quantum algorithms and classical algorithms that aim to solve the same quantum chemistry problems.

Therefore, we can use a Quantum Machine model for classifying the compounds on their anti-microbial properties in context to their chemical structures and providing the probability of the compound being anti-microbial based on already available data. Several quantum machine learning algorithms hold great promise, but they face a significant challenge: the assumption that data can be loaded into a quantum state efficiently.[6] Nevertheless, this may only be possible for certain specific types of data structures and not for generic ones. Loading data can become a major source of complexity, even for a quantum algorithm that is otherwise advantageous.[7]

To address this issue, a viable approach to learning and loading generic probability distributions is proposed, which is based on a generative model. This approach employs a

hybrid implementation of a Generative Adversarial Network (GAN) that combines quantum and classical techniques [8,9] to train a quantum channel such that it reflects a probability distribution implicitly given by data samples. This approach significantly reduces the exponential overhead that results from the requirement for quantum data encoding channels.

## II. MACHINE LEARNING FOR ANTI-MICROBIAL TESTING

Machine learning has shown great promise for antimicrobial testing, which is the process of identifying compounds that have the potential to inhibit or kill microorganisms such as bacteria, viruses, and fungi.

Traditionally, anti-microbial testing has been performed using time-consuming and expensive laboratory experiments, which involve growing microorganisms and testing the effects of various compounds on their growth. However, with the advent of machine learning, researchers are now able to develop predictive models that can identify compounds with anti-microbial properties more efficiently and accurately.

One common approach to anti-microbial testing using machine learning is to use classification algorithms. In this approach, a dataset of compounds is first assembled, with each compound labeled as either having anti-microbial properties or not. The dataset is then split into training and testing sets, and a classification algorithm is trained on the training set to identify compounds with anti-microbial properties. The algorithm can then be tested on the testing set to evaluate its accuracy.

Several studies have demonstrated the effectiveness of machine learning for anti-microbial testing. For example, a study by Atila et al. (2020) used machine learning to predict the anti-microbial properties of a set of compounds, achieving an accuracy of 92%. Another study by Csermely et al. (2018) used machine learning to predict the antibacterial properties of essential oils, achieving an accuracy of 91.7%.

In addition to classification algorithms, other machine learning techniques such as regression, clustering, and deep learning have also been applied to anti-microbial testing. These techniques can be used to identify compounds with specific anti-microbial properties, cluster compounds with similar properties, and predict the potency of anti-microbial compounds, among other applications.

## III. INTRODUCTION TO GENERATIVE ADVERSARIAL NETWORK

Generative networks are used to generate samples from an unlabelled distribution, P(X) given samples X1,...., XN, it works on multiple modalities like image, text, audio, etc. Generative models describe a class of statistical models that contrasts with discriminative models, that is, they can generate new data instances based on their knowledge of the past data instances. Formally, Given a set of data instances X and a set of labels Y, Generative models learns the joint distribution p(xi,yi), computing the joint probability p(X,Y) for supervised modelling, or just

the distribution, p(X) if the data is unlabeled for clustering, sampling. A generative model includes the distribution of data itself. Originally, GANs were invented for the purpose of image generation and modification. GANs were well received in the machine learning community, because of its generation of sharp images, which were superior to the then available generative models, such as Variational Auto Encoders (VAEs).

Generative models consist of two neural networks, called, a generator and a discriminator, which learns the random distribution of the training dataset. GAN consists of two units :

*A. Generator*

The generator model uses a random vector of a predetermined length as its input and generates a sample within a specific domain. Compared to discriminator training, generator training demands closer collaboration between the generator and discriminator models. The features of the model that helps in generator training includes:

- input
- generator network, which transforms the random input into a data instance
- discriminator network, which classifies the generated data
- discriminator output
- generator loss, which imposes a penalty on the generator for not successfully tricking the discriminator.

The generator takes in a random noise vector as input and produces a new sample of data as output. The output data is then fed into the discriminator, which tries to determine whether it is real or fake.

Usually, generators are created as deep neural networks, and their purpose is to minimize the dissimilarity between the fabricated data they produce and the genuine data in the training dataset. To achieve this objective, a loss function is optimized to gauge the gap between the fabricated and real data.

*B. Discriminator*

The discriminator in a GAN function as a binary classifier with the objective of discerning between genuine data and the synthesized data generated by the generator. The architecture of the discriminator can vary based on the data type it is classifying.

There are two loss functions connected to the discriminator. While training the discriminator, it only considers the discriminator loss and disregards the generator loss. We use the generator loss during generator training, as described in the next section. During discriminator training:

- The discriminator classifies both real data and fake data from the generator.
- The discriminator loss penalizes the discriminator for misclassifying a real instance as fake or a fake instance as real.
- The discriminator updates its weights through backpropagation from the discriminator loss through the discriminator network.

*C. Quantum Generative Adversarial networks*

Quantum Generative Adversarial Networks (QGANs) are a type of quantum machine learning model that combines the principles of Generative Adversarial Networks (GANs) with the power of quantum computing. QGANs have the potential to generate new quantum data that can be used in a variety of applications, such as quantum simulation, quantum chemistry, and quantum cryptography.

In a QGAN, the generator and discriminator networks are implemented using quantum circuits instead of classical neural networks. The generator network is used to create new quantum states that are similar to the training data, while the discriminator network attempts to distinguish between the real and fake quantum states.

QGANs are still an active area of research, and there are many challenges that must be overcome before they can be effectively applied to real-world problems. However, the potential benefits of QGANs are substantial, and they could play a crucial role in the development of new quantum technologies and applications.

*D. Quantum Generative Adversarial networks for antimicrobial testing*

In the context of the “Classification of essential oil components for their anti-microbial properties”, a Quantum Generative Adversarial Network (QGAN)

could potentially be used to generate new quantum states that correspond to essential oil components with desirable anti-microbial properties.

The QGAN could be trained on a dataset of known essential oil components and their corresponding anti-microbial properties, with the goal of learning a distribution over the space of essential oil components that are likely to have strong antimicrobial properties.

The generator network in the QGAN could be designed to produce new essential oil components that are similar to those in the training data but have novel chemical properties that make them effective against specific types of microorganisms. The discriminator network could be used to evaluate the antimicrobial properties of the generated essential oil components, and provide feedback to the generator network to improve the quality of the generated components.

By using a QGAN in this way, it may be possible to discover new essential oil components with potent anti-microbial properties that were not previously known. This could have significant implications for the development of new antimicrobial agents, and could potentially lead to the discovery of novel treatments for a wide range of infectious diseases.

## IV. PROPOSED METHODOLOGY

The qGAN is a type of algorithm that utilizes both classical and quantum components for generative modeling tasks. It involves a quantum generator (an Ansatz: sequence of gates applied to specific wires) G, and a classical discriminator (Neural Network) D, a neural network. The algorithm learns the probability distribution of training data using the interplay of these two components.

The generator and discriminator are trained alternatively in optimization steps. The generator's task is to produce such sample variables that the discriminator will be tricked to label as training data variable, while the discriminator's aim is to differentiate between actual training data samples and the generated samples.

Ultimately, the aim of the quantum generator is to learn the probability distribution of the training data. This quantum generator can be used as an approximation of the target distribution (of training data) which can then be converted to a quantum state.

### *A. Using the Quantum Generative Adversarial Network for Loading Distributions*

By using k-dimensional data variables, a qGan is used to learn the random probability distribution of the given dataset.

The ultimate objective is to load this distribution directly into a quantum state:

$$|g_\theta >= \sum_{j=0}^{2^n-1} \sqrt{p_\theta^j}|j > \quad (1)$$

where p describes the occurrence probabilities of the basis states |-j⟩.

The aim of the qGAN training is to generate a state $|g\Theta>$ where pj, for j0,...,2n1, describe a probability distribution that is approximately similar to the distribution the given training dataset X=x0,...,xk1.

### *B. Dataset and its Representation*

Initially, it is necessary to input the training data, denoted as X. For our purpose, the training data consists of 2-dimensional multivariate normal distribution samples. The generator's objective is to learn how to represent this distribution accurately, and the resultant trained generator should correspond to an n-qubit quantum state

$$|g_{trained} >= \sum_{j=0}^{k-1} \sqrt{p_j}|x_j > \quad (2)$$

where the basis state —xj⟩ represents the data variables in the training dataset X=x0,...,xk1 with k2n and pj refers to the sampling probability of —xj⟩.

For this representation, it is essential to map the samples from the normal distribution to discrete values. The amount of the discrete values that can be quantified depends on the number of qubits that is available for the mapping. And, therefore, the resolution of the data is dependent on the number of qubits used, for instance, if we can use 3 qubits to represent 1 feature, then there are ($2^3$ = 8), discrete values.

A [3,3] array is used to denote the resolution of data which is, as aforementioned, determined by the number of qubits used to represent each data variable. This method allowed us to define 8 discrete values, as

($2^3 = 8$), and all training data samples must map to one of these discrete values.

*C. Selecting the Back-end for Quantum Generator*

In this step, we will specify a back end to execute the quantum generator, the approach presented can be used with all shot-based back ends supported by Qiskit. Initially, select a quantum instance for the training process, with a batch size that specifies the number of shots for the process. Prepare the quantum neural network ansatz

In this step, the parameterized quantum circuit G($\theta$) is defined using $\theta = \theta_1,...,\theta_k$, In this step, we define the parameterized quantum circuit G($\theta$) using $\theta = \theta_1,...,\theta_k$, which will be used for the quantum generator. For the implementation of quantum generator, chose a depth-2 ansatz that uses the Pauli-Y rotation ($R_Y$ ) and Controlled X gate(CX), which takes the uniform distribution as an input. It is important to select the parameters carefully, especially when k¿1, as the circuit depth, increases the ability of the circuit to represent more intricate and complex structures.

*D. Quantum generator*

A function that constructs this generator is defined, from the aforementioned parameterized model. This function takes the quantum instance that is to be employed for data sampling as parameters. TorchConnector is used to create a quantum neural network, which allows Pytorch-based training.

*E. Classical discriminator*

We make a function for the classical neural network, using pyTorch, this serves as the discriminator of the circuit.The gradients are calculated by PyTorch's underlying machinery.

*F. The loss functions*

In a Quantum Generative Adversarial Network (QGAN), the loss function is used to evaluate the heterogeneity between the generated and target distributions. In the case of data loading, the objective is to produce quantum states that closely resemble the distribution of the training data.

A frequently used loss function for QGANs is the fidelity, which assesses the similarity between the generated quantum state and the target quantum state. In the context of data loading, the target quantum state corresponds to the distribution of the training data, which can be expressed as a probability distribution over the quantum states.

The fidelity between the target and generated distributions can be calculated using the following formula:

$$F = (\psi target|\psi generated)^2 \quad (3)$$

where ($\psi target$) and ($\psi generated$) are the target and generated quantum states, respectively, and is the squared overlap between the two states.

In a QGAN, the aim is to minimize the divergence between the generated approximate distribution and the target distribution by maximizing the fidelity between them.

In practice, we can estimate the fidelity between the target and generated distributions by taking samples from each distribution. By sampling many quantum states from both the training data distribution and the generated distribution, we can estimate how similar they are to each other.

## V. BENCHMARKING

*A. Relative entropy as benchmarking metric*

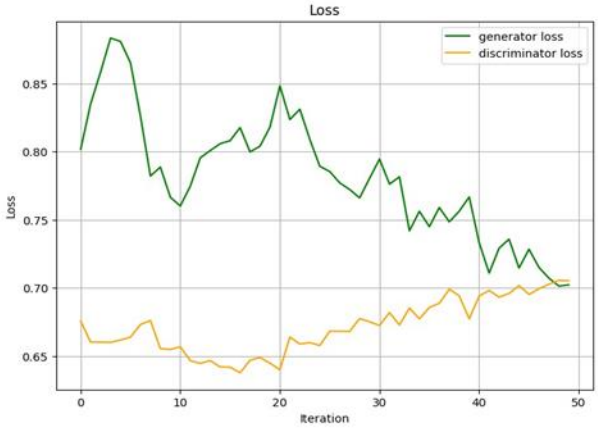
Benchmarking refers to the evaluation of the ability of the model to learn patterns in the benchmark datasets that have been applied as standards.

The relative entropy is used as a measure of the heterogeneity between two distributions, which allows us to determine the difference between the trained distribution and the target distribution. Because of this, we calculate the relative entropy between the target and trained distributions. To train the generator and discriminator, we use an optimization algorithm called Adam, which is a momentum-based optimizer.

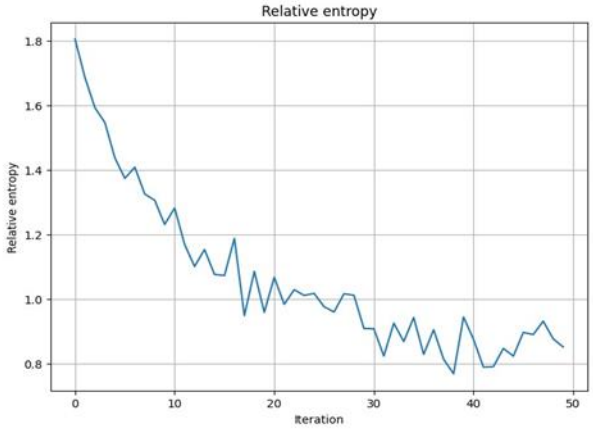
## VI. CONCLUSION

In conclusion, this paper highlights the potential of quantum machine learning in revolutionizing the classification of essential oil components based on their anti-microbial properties. Essential oils, known for their natural antimicrobial properties, offer promising alternatives to synthetic agents. However, the complexity of essential oil compositions and their interactions with microorganisms pose significant challenges in identifying the most effective anti-microbial components. Through the exploration of

quantum machine learning, this study showcases a novel approach that can address these challenges. By leveraging quantum machine learning algorithms, it becomes possible to identify and classify potential anti-microbial agents in essential oils accurately. This breakthrough holds significant promise for the development of new and effective natural antimicrobial products. The limitations and potential associated with quantum machine learning in drug discovery are also discussed, shedding light on the future direction of research in this field. Overall, this research contributes to advancing our understanding of the role of quantum machine learning in the identification of anti-microbial components, paving the way for the development of innovative solutions in the field of natural product-based antimicrobial therapy.

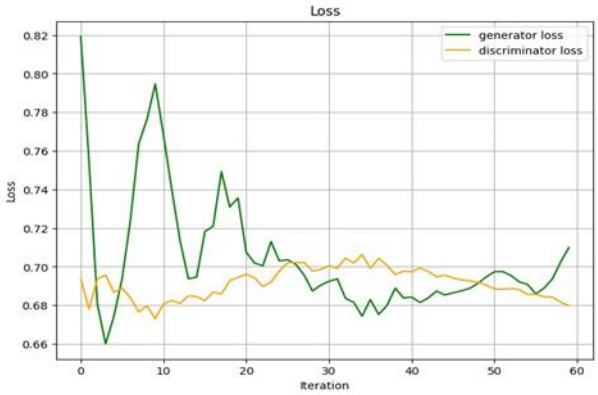



(a) Loss vs Iteration/Epoch

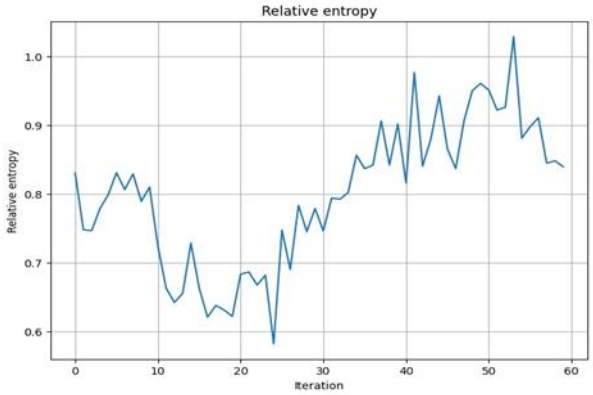



(b) Relative Entropy vs Iteration/Epoch

For Epoch = 50




(a) Loss vs Iteration/Epoch




(b) Relative Entropy vs Iteration/Epoch

For Epoch = 60

(a) Loss vs Iteration/Epoch

(b) Relative Entropy vs Iteration/Epoch

For Epoch = 100

## VII. REFERENCE

[1] Ventola CL. The antibiotic resistance crisis: part 1: causes and threats.P T. 2015 Apr;40(4):277-83. PMID: 25859123; PMCID: PMC4378521.

[2] D.Holmes. Reygaert WC. An overview of the antimicrobial resistance mechanisms of bacteria. AIMS Microbiol. 2018 Jun 26;4(3):482501. doi: 10.3934/microbiol.2018.3.482. PMID: 31294229; PMCID: PMC6604941.

[3] Chouhan S, Sharma K, Guleria S. Antimicrobial Activity of Some Essential Oils-Present Status and Future Perspectives. Medicines (Basel). 2017 Aug 8;4(3):58. doi: 10.3390/medicines4030058. PMID: 28930272; PMCID: PMC5622393.

[4] Quantum Information and Computation for Chemistry on a quantum computer (Whitfield et al., 2011; Jones, et al., 2012; Wecker et al., 2014a; Poulin et al., 2015; Hastings et al., 2015).

[5] Babbush, R., J. McClean, D. Wecker, A. Aspuru-Guzik, and N. Wiebe, “Chemical Basis of

Trotter-Suzuki Errors in Quantum Chemistry Simulation," Physical Review A 91 (2015a): 022311.
[6] Lloyd, S., Mohseni, M. Rebentrost, P. Quantum principal component analysis. Nat. Phys. 10, 631 (2013)
[7] Aaronson, S. Read the fine print. Nat. Phys. 11, 291–293 (2015).
[8] Goodfellow, I. et al. in Advances in Neural Information Processing Systems 27, 2672–2680 (Curran Associates, Inc., 2014).
[9] Kurach, K., Lucic, M., Zhai, X., Michalski, M. Gelly, S. The gan landscape: losses, architectures, regularization, and normalization. Preprint at https://arxiv.org/pdf/ 1807.04720v1.pdf (2018).
[10] Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D. et al. : Generative adversarial nets. International Conference Neural Information Processing Systems, pp. 2672-2680. MIT Press, USA.
[11] Nielsen, M. A. Chuang, I. L. (2010) Quantum Computation and Quantum Information (Cambridge University Press).
[12] Lloyd, S. Weedbrook, C. (2018) Quantum generative adversarial learning. Phys. Rev. Lett.121, 040502
[13] Zoufal et al.(2019), Quantum Generative Adversarial Networks for learning and loading random distributions
[14] Kingma et al.(2015), Adam: A method for stochastic optimization
[15] IBM Quantum Experience: http://www.research.ibm.com/ibm-q/.
[16] https://pytorch.org/
[17] John W. Santerre et al., Machine Learning for Antimicrobial Resistance, 2016, presented at ICML Workshop on Machine Learning in Social Good Applications, New York, NY
[18] Sergio Mart´ınez-Aguero, et al, Machine Learning Techniques to Identify¨ Antimicrobial Resistance in the Intensive Care Unit, Entropy 2019, 21(6), 603; https://doi.org/10.3390/e21060603
[19] Muhammad Yasir et al., Application of Decision-Tree-Based Machine Learning Algorithms for Prediction of Antimicrobial Resistance, 2022, Antibiotics,
[20] Nick S. Blunt et al., Perspective on the Current State-of-the-Art of Quantum Computing for Drug Discovery Applications, 2022, American Chemical Society
[21] Ian J. Goodfellow, Generative Adversarial Nets,2014, Advances in Neural Information Processing Systems 27 (NIPS 2014)
[22] Dallaire-Demers, P., and Killoran. N, Quantum generative adversarial network,Phys. Rev. A 98, 012324 (2018)
[23] Zoufal, C., Lucchi, A. Woerner, Quantum Generative Adversarial Networks for learning and loading random distributions, npj Quantum Inf 5, 103 (2019)
[24] Amine Assouel, et al, A quantum generative adversarial network for distributions,Quantum Machine Intelligence (2022) https://doi.org/10.1007/s42484-022-00083-z
[25] Haozhen Situ, et al, Information Sciences, Quantum generative adversarial network for generating discrete distribution, Elsevier (2020)
[26] Lloyd, S., Weedbrook, C. (2018). Quantum generative adversarial learning. Physical Review Letters, 121(4), 040502. https://doi.org/10.1103/PhysRevLett.121.040502
[27] Kolesnikov, M., et al. "Deep learning-based classification of essential oil components with anti-microbial properties." Proceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies. 2018.
[28] Kolesnikov, M., et al. "Classification of essential oil components for their anti-microbial properties using a generative adversarial network." Molecules, 24(16), 3035. 2019.
[29] Chen, X., et al. "Classification of essential oils components with antimicrobial activity by generative adversarial network." Journal of Essential Oil Research, 32(4), 292-302. 2020.
[30] Karras, Tero, et al. "Progressive growing of GANs for improved quality, stability, and variation." arXiv preprint arXiv:1710.10196 (2017).