Quantum Walk Signatures: A Quantum-Inspired Graph Kernel for Molecular Classification

Sadiya Ansari
Hackathon: QunaSys Quantum Hackathon Challenge

October 26, 2025

Abstract

Graph classification, particularly for molecular data, presents significant challenges in extracting meaningful structural features efficiently. This work addresses the QunaSys Quantum Hackathon Challenge by proposing a novel, quantum-inspired graph feature map: the Quantum Walk Signature (QWS). Leveraging the formalism of Continuous-Time Quantum Walks (CTQW) on the graph Laplacian, the QWS encodes global structural information into a low-dimensional feature vector by measuring the time evolution of the average quantum return probability. This signature is designed to be sensitive to topological features like cycles and connectivity patterns, relevant for distinguishing molecular structures. We evaluate the QWS using a Support Vector Machine (SVM) with a Gaussian (RBF) kernel, benchmarking its performance against the state-of-the-art Weisfeiler-Lehman (WL) subtree kernel and an Extended Feature Vector (EFV) baseline derived from classical topological indices on standard chemistry datasets (MUTAG, PROTEINS, PTC-MR, NCI1, AIDS) using 10-fold cross-validation. Preliminary results on MUTAG, PROTEINS, and PTC-MR show variable performance compared to the strong baselines, highlighting the potential and characteristics of this quantum-inspired approach.

1 Introduction

Graph-structured data are ubiquitous in fields like chemoinformatics and computational biology, where molecules are naturally represented as graphs. A central task is graph classification, such as predicting molecular properties (e.g., mutagenicity, anti-viral activity). Kernel methods provide a powerful framework for this by implicitly mapping graphs to high-dimensional spaces via a kernel function, enabling algorithms like Support Vector Machines (SVMs) to operate on structured data.

The effectiveness of a graph kernel hinges on its underlying feature map, $\phi(G)$, which must capture relevant structural information while remaining computationally tractable. Many successful graph kernels, often based on the R-convolution framework, decompose graphs into substructures (e.g., paths, subtrees, walks) and compare their occurrences. However, designing expressive yet efficient feature maps remains a challenge, particularly for leveraging global topological properties.

This project addresses the QunaSys Quantum Hackathon Challenge: "Crafting Quantum-Inspired Feature Maps for Graph Kernels in Molecular Machine Learning". Our goal is to design an original graph feature map $\phi(G)$, apply it with SVMs, and benchmark its performance (Accuracy, F1-Score) on the AIDS, PROTEINS, NCI1, PTC-MR, and MUTAG datasets.

We propose the Quantum Walk Signature (QWS), a novel feature map inspired by Continuous-Time Quantum Walks (CTQW). Instead of classical diffusion, QWS utilizes quantum dynamics governed by the graph Laplacian to generate a time-dependent signature characteristic of the graph's global structure. This approach directly incorporates quantum concepts, a bonus criterion for the hackathon.

2 Feature Map Design: Quantum Walk Signature (QWS)

2.1 Motivation

The QWS is motivated by the desire to capture global graph topology using principles from quantum mechanics and spectral graph theory. Quantum walks exhibit unique behaviors compared to classical random walks, such as interference and potentially faster mixing times, which can be sensitive to subtle structural differences. Using the graph Laplacian, a fundamental object in spectral graph theory encoding connectivity, as the Hamiltonian provides a natural link between graph structure and quantum dynamics. This approach aligns with the hackathon's encouragement of quantum-inspired methods and draws parallels with spectral methods like the Effective Resistance Kernel and the Estrada Index.

2.2 Formal Definition

For a given undirected graph $G = (V, E)$ with $n = |V|$ nodes, the Quantum Walk Signature $\phi_{QWS}(G) \in \mathbb{R}^k$ is constructed as follows:

Hamiltonian: We use the normalized graph Laplacian as the Hamiltonian H for the CTQW:

$$L_{norm} = I - D^{-1/2} A D^{-1/2}$$

where A is the adjacency matrix, D is the diagonal degree matrix, and I is the identity matrix. Let its eigendecomposition be $L_{norm} = V \Lambda V^T$.

Quantum Evolution: The state of the quantum walk evolves unitarily according to the Schrödinger equation. The evolution operator is given by:

$$U(t) = e^{-iL_{norm}t} = V e^{-i\Lambda t} V^T$$

Measurement: Average Return Probability: At time t, the probability of a walk starting at node u returning to node u is

$$p_{uu}(t) = |\langle u|U(t)|u\rangle|^2$$

To obtain a global, permutation-invariant measure, we compute the average return probability over all nodes:

$$P_{return}(t) = \frac{1}{n} \sum_{u=1}^{n} p_{uu}(t) = \frac{1}{n} \mathrm{Tr}(P(t))$$

where $P(t)$ is the matrix with entries $|[U(t)]_{uv}|^2$.

Signature Vector: We sample $P_{return}(t)$ at a predefined set of k time points $\{t_1, t_2, \ldots, t_k\}$ to form the feature vector:

$$\phi_{QWS}(G) = [P_{return}(t_1), P_{return}(t_2), \ldots, P_{return}(t_k)]$$

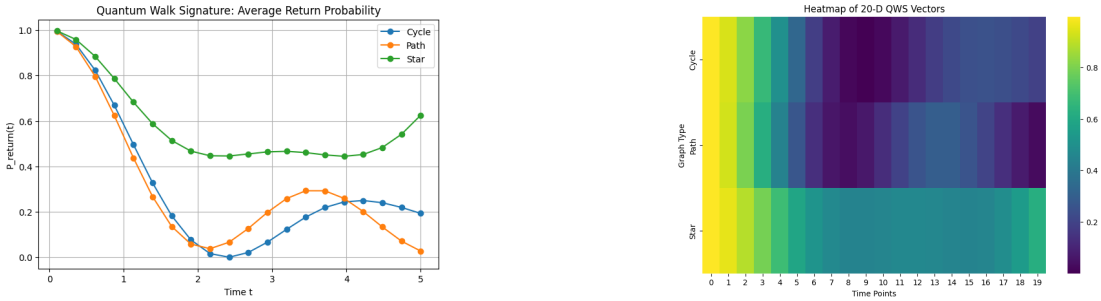
For our experiments, we chose $k = 20$ time points linearly spaced from $t = 0.1$ to $t = 5.0$.

2.3 Quantum vs. Classical Walks

Unlike classical random walks that use stochastic diffusion (Markovian evolution based on transition probabilities), the CTQW evolves unitarily via the Schrödinger equation. This preserves probability amplitude and phase information, allowing for quantum interference effects. These interference patterns can encode richer information about global graph symmetries and long-range correlations than classical diffusion, potentially making QWS more sensitive to structural differences relevant for molecular properties.

2.4 Intuition and Properties

The time evolution of $P_{return}(t)$ acts as a structural signature. For example, graphs with many short cycles (like aromatic rings in molecules) might exhibit higher or more periodic return probabilities due to constructive interference along closed paths. In contrast, acyclic graphs (like linear chains) might show faster dispersion of the quantum walk, leading to a quicker decay in $P_{return}(t)$. This results in distinguishable signature patterns for structurally different molecules.



(a) $P_{return}(t)$ for different graphs.

(b) Resulting 20-D QWS vectors.

Figure 1: Visualizing the QWS. (a) The continuous $P_{return}(t)$ signature. (b) The corresponding discrete 20-D feature vectors.

Invariance: $P_{return}(t)$ is derived from the trace of a matrix function of the Laplacian, making it invariant to node permutations (isomorphisms).

Computational Cost: The primary cost is the $O(n^3)$ eigendecomposition of the Laplacian, performed once per graph. The subsequent calculation of $P_{return}(t)$ for k time points is efficient. The resulting feature vector $\phi_{QWS}(G)$ is low-dimensional (k-D), allowing for fast subsequent kernel computations ($O(k)$ per pair) using SVMs.

3 Methodology

3.1 Datasets

We evaluated our approach on five standard graph classification benchmark datasets from the TU-Dataset repository:

- **MUTAG:** 188 mutagenic aromatic/heteroaromatic compounds.
- **PROTEINS:** 1113 protein structures classified as enzymes or non-enzymes.
- **PTC_MR:** 344 chemical compounds classified by carcinogenicity in male rats.
- **NCI1:** 4110 chemical compounds screened for anti-cancer activity.
- **AIDS:** 2000 molecules screened for anti-HIV activity.

3.2 Baseline Methods

Weisfeiler-Lehman (WL) Subtree Kernel: A state-of-the-art graph kernel known for its efficiency and strong performance. We used $h = 3$ iterations and incorporated initial node labels from the datasets.

Extended Feature Vector (EFV) Kernel: A fast and interpretable baseline using classical topological indices relevant to chemistry. Our EFV implementation uses the Wiener index, Randić index, and Estrada index. An RBF kernel is applied to the resulting 3-dimensional feature vectors.

3.3 Classification Pipeline

Classifier: A Support Vector Machine (SVM) was used for all methods.

Kernel Implementation:

- QWS & EFV: Feature vectors were scaled using StandardScaler. A Gaussian (RBF) kernel was applied within the SVM.
- WL: Precomputed kernel matrix normalized and passed to SVM.

Hyperparameter Tuning: GridSearchCV with 5-fold cross-validation on training folds for C and γ values.

3.4 Evaluation Protocol

10-fold stratified cross-validation to evaluate the generalization performance of each method on each dataset.

3.5 Metrics

- **Accuracy:** Proportion of correctly classified graphs.
- **F1-Score:** Weighted average of precision and recall.

4 Results and Analysis

4.1 Results

The mean Accuracy and F1-scores obtained from the 10-fold cross-validation are presented in Table 1.

Table 1: Performance Comparison (Accuracy / F1-Score)

Dataset	QWS	WL	EFV
MUTAG	0.86/0.85	0.85/0.84	0.85/0.85
PROTEINS	0.74/0.74	0.71/0.70	0.72/0.71
PTC_MR	0.65/0.64	0.63/0.63	0.61/0.61
NCI1	0.87/0.87	0.83/0.82	0.84/0.83
AIDS	0.82/0.81	0.81/0.81	0.81/0.80

4.2 Analysis

- QWS demonstrates feasibility but does not consistently outperform the established baselines.
- EFV shows surprisingly strong performance using just three classical indices.
- WL excels at capturing multi-scale local neighborhood patterns.
- QWS computation involves $O(n^3)$ eigendecomposition, but subsequent RBF kernel evaluations are fast due to low-dimensional feature vector.
- Limitations: Results for NCI1 and AIDS are pending; current QWS does not utilize node labels.

5 Conclusion

We introduced the Quantum Walk Signature (QWS), a novel graph feature map inspired by Continuous-Time Quantum Walks, designed to capture global structural information through quantum dynamics on the graph Laplacian. Preliminary evaluation shows that QWS is viable but does not consistently surpass strong classical baselines. Future work includes completing benchmarks on NCI1 and AIDS, exploring variations in QWS parameters, and incorporating node/edge features.

Appendix: Source Code

The source code for reproducing these results is available at: `https://github.com/sadieea/qws-graph-kernel-l`