AUTHOR QUERY FORM

AIP Publishing	**Journal: J. Chem. Phys.** **Article Number: JCP22-AR-CHAI2021-00840**	**Please provide your responses and any corrections by annotating this PDF and uploading it to AIP's eProof website as detailed in the Welcome email.**

Dear Author,

Below are the queries associated with your article. Please answer all of these queries before sending the proof back to AIP.

Article checklist: In order to ensure greater accuracy, please check the following and make all necessary corrections before returning your proof.
1. Is the title of your article accurate and spelled correctly?
2. Please check affiliations including spelling, completeness, and correct linking to authors.
3. Did you remember to include acknowledgment of funding, if required, and is it accurate?

Location in article	Query/Remark: click on the Q link to navigate to the appropriate spot in the proof. There, insert your comments as a PDF annotation.
Q1	Please check that the author names are in the proper order and spelled correctly. Also, please ensure that each author's given and surnames have been correctly identified (given names are highlighted in red and surnames appear in blue).
Q2	Please provide zip code for affiliations 1 and 2.
Q3	~~We have reworded the sentence beginning "The data are…" for clarity. Please check.~~
Q4	~~The Data Availability statement has been edited to follow AIPP style fordata openly available in a public repository that does not issue DOIs. Please check.~~
Q5	~~Please confirm the change in author's initials in Refs. 11, 74, and 113.~~
Q6	Ref. 11, 42, 50, 60, 67 and 80: Can these references be updated? If so, please provide the relevant information such as year, volume and page or article numbers as appropriate.
Q7	~~Please confirm the change in article title in Ref. 16.~~
Q8	~~Please confirm the content in Ref. 20, as we have inserted the required information.~~
Q9	We were unable to locate a digital object identifier (doi) for Refs. 46, 71, 78, 94, 106, 111, and 112. Please verify and correct author names and journal details (journal title, volume number, page number, and year) as needed and provide the doi. If a doi is not available, no other information is needed from you. For additional information on doi's, please select this link: http//www.doi.org/.
Q10	~~References 55 and 57 contain identical information. Please check and provide the correct reference or delete the duplicate reference. If the duplicate is deleted, renumber the reference list as needed and update all citations in the text.~~
Q11	~~Please confirm the change in year of publication in Ref. 57.~~
Q12	Please confirm the change in page number in Refs. 66, 100, and 110.
Q13	Please provide volume number in Ref. 67.
Q14	If e-print Refs. 69, 75, and 80 have subsequently been published elsewhere, please provide updated reference information (journal title, volume number, and page number).
Q15	Please provide publisher's name in Refs. 81, 119, 122, and 124.
Q16	~~Please confirm the change in journal title in Ref. 82.~~

Continued on next page

Continued from previous page	
Q17	Please provide page number in Ref. 87.
Q18	Please provide complete information in Ref. 94.
	Please confirm ORCIDs are accurate. If you wish to add an ORCID for any author that does not have one, you may do so now. For more information on ORCID, see https://orcid.org/. Ping Yang – 0000-0003-0105-6172 E. Adrian Henle – Xiaoli Z. Fern – Cory M. Simon – 0000-0002-8181-9178
	Please check and confirm the Funder(s) and Grant Reference Number(s) provided with your submission: National Science Foundation, Award/Contract Number 1920945 Please add any additional funding sources not stated above.

Thank you for your assistance.

The Journal of Chemical Physics ARTICLE scitation.org/journal/jcp

Classifying the toxicity of pesticides to honey bees via support vector machines with random walk graph kernels

Cite as: J. Chem. Phys. 156, 000000 (2022); doi: 10.1063/5.0090573
Submitted: 8 March 2022 • Accepted: 24 May 2022 •
Published Online: 9 99 9999

View Online Export Citation CrossMark

Ping Yang,[1] E. Adrian Henle,[1] Xiaoli Z. Fern,[2] and Cory M. Simon[1,a)]

AFFILIATIONS

[1] School of Chemical, Biological, and Environmental Engineering, Oregon State University, Corvallis, Oregon, USA
[2] School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, Oregon, USA

Note: This paper is part of the JCP Special Topic on Chemical Design by Artificial Intelligence.
[a)] **Author to whom correspondence should be addressed:** Cory.Simon@oregonstate.edu

ABSTRACT

Pesticides benefit agriculture by increasing crop yield, quality, and security. However, pesticides may inadvertently harm bees, which are valuable as pollinators. Thus, candidate pesticides in development pipelines must be assessed for toxicity to bees. Leveraging a dataset of 382 molecules with toxicity labels from honey bee exposure experiments, we train a support vector machine (SVM) to predict the toxicity of pesticides to honey bees. We compare two representations of the pesticide molecules: (i) a random walk feature vector listing counts of length-L walks on the molecular graph with each vertex- and edge-label sequence and (ii) the Molecular ACCess System (MACCS) structural key fingerprint (FP), a bit vector indicating the presence/absence of a list of pre-defined subgraph patterns in the molecular graph. We explicitly construct the MACCS FPs but rely on the fixed-length-L random walk graph kernel (RWGK) in place of the dot product for the random walk representation. The L-RWGK-SVM achieves an accuracy, precision, recall, and F1 score (mean over 2000 runs) of 0.81, 0.68, 0.71, and 0.69, respectively, on the test data set—with $L = 4$ being the mode optimal walk length. The MACCS-FP-SVM performs on par/marginally better than the L-RWGK-SVM, lends more interpretability, but varies more in performance. We interpret the MACCS-FP-SVM by illuminating which subgraph patterns in the molecules tend to strongly push them toward the toxic/non-toxic side of the separating hyperplane.

I. INTRODUCTION

A. Pesticide toxicity to bees

Pesticides (including insecticides, fungicides, and herbicides) are used in agriculture as an economical means to control weeds, pests, and pathogens. Thereby, pesticides increase expected crop yield and quality and contribute to food security.[1–4] However, widespread pesticide use has negative externalities on both aquatic and terrestrial ecosystems and human health.[5–9] For example, pesticides can harm agriculturally beneficial species not deliberately targeted, such as earthworms and bees.[10,11]

Although under debate, extensive pesticide use in agriculture may play a role[11–16] in the widespread decline[17–20] of bee populations (see Ref. 21 for a synopsis) via both lethal and sublethal toxicity.[11,21] Harms to bee populations are especially concerning because bees are valuable for agricultural production:[22] (1; primary value) bees serve as pollen vectors for many crops, including fruits, vegetables, nuts, oilseed, spices, and coffee.[23,24] Specifically, bees visit the flowers of plants (angiosperms) to collect pollen or nectar as a food source. In the process, bees (inadvertently) transfer pollen from the anther of one flower to the stigma of another flower, a necessary step in the production of seeds and fruits for many plants.[25] (2; secondary value[26]) Honey bees produce honey and beeswax. In addition, bees are ecologically valuable as pollen vectors for plants in natural habitats.[27]

Because insect,[28] weed,[29] and fungi[30] populations can develop resistance to an insecticide, herbicide, and fungicide, respectively, new pesticides must be continually discovered and deployed.[1] New pesticide development is also driven by the aim to reduce negative environmental impacts of incumbent pesticides.[31]

Virtual screenings can accelerate the discovery of new pesticides operating under a known mechanism. For example, suppose

an insect protein is a known target for insecticides. Then, computational protein-ligand docking[32] can score candidate compounds for insecticide activity, informing experimental campaigns.[33–39] However, newly proposed pesticides must also be assessed for toxicity to honey bees[40] (see the US EPA website[41]).

A computational model that accurately predicts the toxicity of pesticides to bees would be useful[42] (i) as a toxicity filter in virtual and experimental screenings of compounds for pesticide activity, (ii) in emergency situations where an immediate assessment of toxicity risk is needed, and (iii) to focus scrutiny and motivate more thorough toxicity assessments on existing and new pesticides predicted to be toxic. Generally, training machine learning models to predict the toxicity of compounds to biological organisms is an active area of research.[43,44] Indeed, open data from bee toxicity experiments[45–57] have been leveraged to train machine learning models to computationally predict the toxicity of pesticides to bees.[58–64]

B. Representing molecules for supervised machine learning tasks

A flurry of research activity is devoted to the data-driven prediction of the properties of molecules via supervised machine learning.[65] A starting point is to design a machine-readable representation of the molecule for input to the machine learning model.[66–68]

A vertex- and edge-labeled graph (vertices = atoms, edges = bonds, vertex label = element, and edge label = bond order) is a fundamental representation of the concept of a small molecule. For many classes of molecules, the mapping of the concept of a molecule to a molecular graph is one-to-one. If we wished to communicate a small molecule to an intelligent, extraterrestrial life form that has just arrived on Earth and does not know our language, we would likely sketch a vertex- and edge-labeled, undirected graph. Molecular graph representations break down for certain classes of molecules[66] and are invariant to the 3D structure and stereoisomerism.[69]

However, because classical machine learning algorithms operate in a Euclidean vector space, much research is devoted to the design of fixed-size, information-rich *vector* representations of molecules that encode their salient features.[66,70–72] Many molecular fingerprinting methods[72] extract topological features from the molecular graph[72] to produce a "bag of fragments" bit vector representation of the molecule.[73] For example, Molecular ACCess System (MACCS) structural key fingerprints[74] of a molecular graph are bit vectors indicating the presence or absence of a pre-defined list of subgraph patterns. Other hand-crafted molecular feature vectors include chemical, electronic, and structural/shape (3D) properties of the molecule as well.[66,73]

Two advanced supervised machine learning approaches circumvent explicit hand-crafting of vector representations of molecular graphs:

1. graph representation learning,[75] such as message passing neural networks (MPNNs)[76,77] that *learn* task-specific vector representations of molecular graphs for prediction tasks in an end-to-end manner and
2. graph kernels,[78–83] which (loosely speaking) measure the similarity between any two input graphs, allowing for the use of kernel methods,[84] such as support vector machines,[85] kernel regression/classification,[84] and Gaussian processes,[86] for prediction tasks.

That is, MPNNs and kernel methods operate directly on the molecular graph representation, bypassing engineering and explicit construction, respectively, of molecular feature vectors for machine learning tasks.

MPNNs are powerful models for molecular machine learning tasks[76] but require large training datasets. In contrast, kernel methods with graph kernels are likely more appropriate when training data are limited, as they are easier to train, possess fewer hyperparameters, and are less susceptible to overfitting.[87] Empirically, graph kernels give performance on par with MPNNs on a variety of molecular prediction tasks.[88]

C. Our contribution: Building a bee toxicity classifier of pesticides via the random walk graph kernel

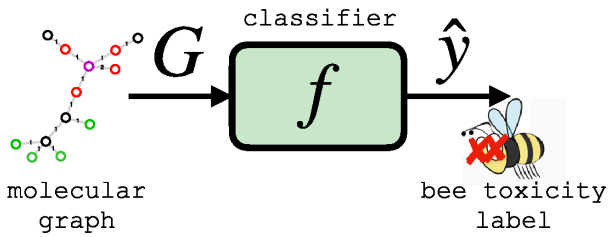
We train and evaluate a support vector machine (SVM) classifier for predicting the toxicity of pesticide molecules to bees. Enabling a machine learning approach, the BeeToxAI project[58] compiled labeled data from bee toxicity experiments, composed of 382 (pesticide molecule, bee toxicity outcome) pairs. We compare two constructions of a molecular vector space for the SVM: (1) a random walk feature space, describing pesticides by the set of vertex- and edge-label sequences along length-L walks on their molecular graphs, and (2) the MACCS fingerprint (FP) space, describing pesticides by the presence/absence of a list of pre-defined subgraph patterns in their molecular graphs. We explicitly construct the MACCS FPs but instead rely on the kernel trick and fixed-length-L random walk graph kernel (RWGK) for dot products in the random walk feature space. The L-RWGK-SVM achieves an F1 score (mean over 2000 runs) of 0.69 on the test dataset, and $L = 4$ is the mode optimal walk length to describe the molecular graphs. The MACCS-FP-SVM performs on par/marginally better but exhibits more variance in its performance. Finally, we illuminate subgraphs in the pesticide molecules that tend to most strongly push molecules in the MACCS FP space toward the toxic/non-toxic side of the separating hyperplane of the MACCS-FP-SVM.

II. PROBLEM SETUP: CLASSIFYING THE TOXICITY OF A PESTICIDE TO HONEY BEES

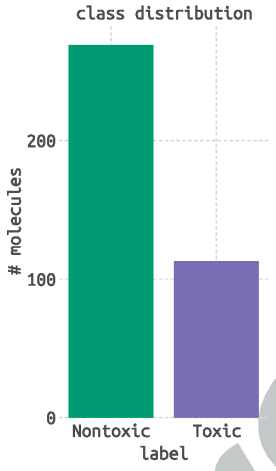
The pesticide toxicity classification task. We wish to construct a classifier $f : \mathcal{G} \to \{-1, 1\}$ that maps any molecular graph G (see Sec. III A) representing a pesticide molecule to a predicted binary label $\hat{y} = f(G)$, where $\hat{y} = 1$ is toxic to honey bees (*Apis mellifera*) and $\hat{y} = -1$ is nontoxic. The classifier f is valuable as a cheap-to-evaluate "surrogate model" of an expensive bee toxicity experiment [see Fig. 1(a)].

The labeled bee toxicity dataset.[58] From the BeeToxAI project,[58] we took labeled data $\{(G_n, y_n)\}_{n=1}^{N}$ composed of $N = 382$ examples of (i) a molecular graph $G_n \in \mathcal{G}$ representing a pesticide or pesticide-like molecule and (ii) its experimentally-determined acute contact bee toxicity label $y_n \in \{-1, 1\}$ (1: toxic, −1: nontoxic).

Figure 1(b) shows the class (im)balance; 113 of the molecules are labeled toxic, and 269 are labeled nontoxic.



(a)



(b)

FIG. 1. Problem setup. (a) Our objective is to train a classifier f that maps a molecular graph G to a binary prediction $\hat{y}$ of the bee toxicity of the pesticide molecule it represents. (b) The label distribution in the BeeToxAI[58] dataset.

The outcome of a bee exposure experiment was mapped to a toxicity label on the pesticide following US EPA guidelines:[89] the pesticide was labeled as toxic if the median lethal dose (LD_{50}) after 48 h to an adult honey bee was greater than 11 μg/bee—and nontoxic otherwise.

The dataset includes neonicotinoid, pyrethroid, organophosphate, carbamate, pyridine azomethine, phenylpyrazole, and organochlorine insecticides,[45–47,49–54,56,57] herbicides,[56] miticides,[50–52] and fungicides.[47,48,52,53,56] Any molecules bearing tetrahedral chiral centers are not labeled with stereochemical configuration.

The machine learning approach: data-driven prediction of bee toxicity. Our objective is to leverage the labeled bee toxicity dataset to train an SVM as the toxicity classifier $f(G)$. An SVM is a versatile supervised machine learning model that aims to find the maximum-margin separator (a hyperplane) between the positive and negative training examples in a mapped feature (vector) space. The mapped feature space does not need to be explicitly constructed. Instead, kernel functions can be used to (implicitly) perform the needed operation (dot product) in the mapped feature space. We compare two constructions of a molecular vector space by representing pesticide molecules with (1) the fixed-length random walk feature vector and (2) the MACCS fingerprint. We explicitly construct the latter, while for the former we rely on the fixed-length random walk graph kernel for the dot product.

III. METHODS

A. The vertex- and edge-labeled graph representation of a molecule

A fundamental representation of a molecule is as a vertex- and edge-labeled, undirected graph $G = (\mathcal{V}, \mathcal{E}, \ell_v, \ell_e)$:

- $\mathcal{V} = \{v_1, \ldots, v_N\}$ is the set of vertices representing its N atoms, excluding hydrogen atoms. We exclude H atoms in the molecular graph to avoid redundancy. For example, the hydrogen-excluding molecular graphs of ethane, ethylene, and acetylene can be distinguished by the order (an edge label) of the C–C bond (edge).
- $\mathcal{E}$ is the set of edges representing chemical bonds; $\{v_i, v_j\} \in \mathcal{E}$ iff the atoms represented by vertices $v_i \in \mathcal{V}$ and $v_j \in \mathcal{V}$ are bonded.
- $\ell_v : \mathcal{V} \to \{\mathrm{C}, \mathrm{N}, \mathrm{O}, \mathrm{S}, \mathrm{P}, \mathrm{F}, \mathrm{Cl}, \mathrm{I}, \mathrm{Br}, \mathrm{Si}, \ \mathrm{As}\}$ is the vertex-labeling function that provides the chemical element of each vertex (atom).
- $\ell_e : \mathcal{E} \to \{1, 2, 3, a\}$ ("a" for aromatic) is the edge-labeling function that provides the bond order of each edge (bond).

For example, see Fig. 2. This *molecular graph* representation of a molecule describes its topology and is invariant to translations and rotations of the molecule and to bond stretching, bending, and rotation.

Let $\mathcal{G}$ be the set of possible molecular graphs, so $G \in \mathcal{G}$.

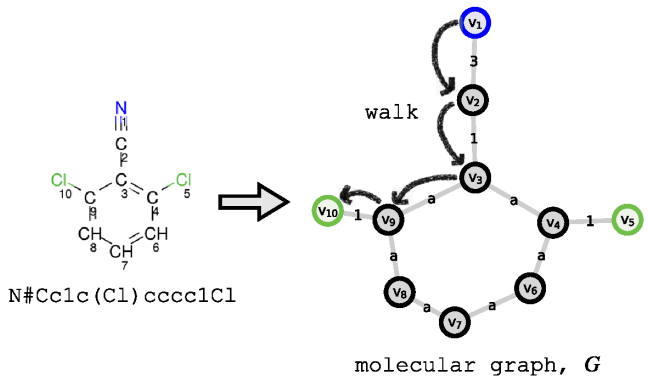


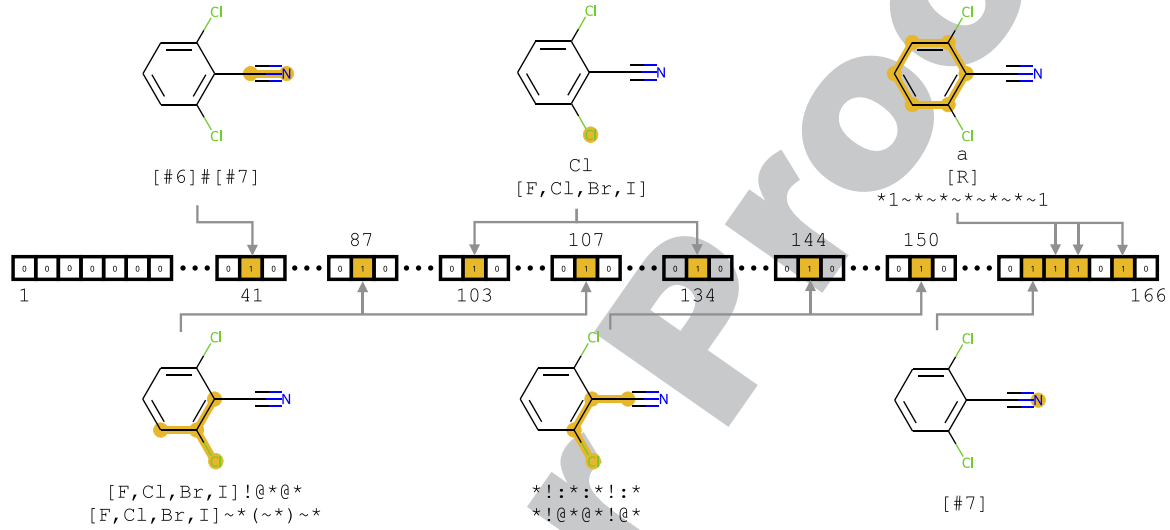
FIG. 2. The molecular graph representation of 2,6-dichlorobenzonitrile (SMILES string shown). Nodes are labeled by atomic species (indicated by color). Edges are labeled by bond order. A length $L = 4$ walk $(v_1, v_2, v_3, v_9, v_{10})$ on the molecular graph is indicated by the arrows. The label sequence of this walk is $(N, 3, C, 1, C, a, C, 1, Cl)$.

B. Two molecular vector spaces

We explore two *feature maps* $\phi : \mathcal{G} \to \mathbb{R}^F$ that map a molecular graph $G \in \mathcal{G}$ to a feature vector $\phi(G) \in \mathbb{R}^F$, with $\mathbb{R}^F$ being the molecular vector space in which the SVM operates: (1) the MACCS structural key fingerprint and (2) the fixed-length random walk feature vector.

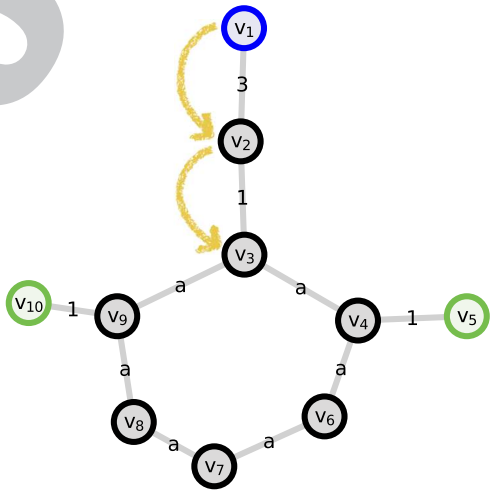
1. MACCS structural key fingerprint

The Molecular ACCess System (MACCS) structural key fingerprint (FP) of a molecular graph is a bit vector whose entries indicate the presence (1) or absence (0) of a list of $F = 166$ predefined subgraph patterns (molecular substructures/fragments).[72,74] The number of "on" (1) bits in the MACCS FP is equal to the



(a)

label sequence, s	# L=2 walks
(N, 3, C, 1, C)	1
(N, 3, C, 3, N)	1
(Cl, 1, C, 1, Cl)	2
(Cl, 1, C, a, C)	4
(C, 3, N, 3, C)	1
(C, 1, Cl, 1, C)	2
(C, a, C, 1, Cl)	4
(C, 1, C, 3, N)	1
(C, a, C, a, C)	24
(C, a, C, 1, C)	2
(C, 1, C, a, C)	2
(C, 1, C, 1, C)	2



(b)

FIG. 3. Illustrating the two molecular vector representations we employ for pesticides. (a) The MACCS fingerprint (FP) is a length-166 bit vector indicating the presence/absence of a predefined list of 166 subgraphs in a molecular graph. Shown here is the MACCS FP ("on" bits orange) of 2,6-dichlorobenzonitrile. Subgraphs of the molecular graph that activate bits of the MACCS FP are highlighted orange. The pattern(s) each subgraph matches is/are indicated in the SMARTS language. (b) The fixed-length random walk feature vector contains counts of label sequences encountered along all fixed-length walks on a molecular graph. The table lists all label sequences encountered along length $L = 2$ walks on the molecular graph of 2,6-dichlorobenzonitrile. The numbers in the second column, giving the number of walks having each label sequence, comprise the nonzero entries of the random walk feature vector of 2,6-dichlorobenzonitrile.

number of these subgraphs with presence in the molecule. The list of molecular patterns defining the MACCS feature map $\phi^{MACCS}(G): \mathcal{G} \to \mathbb{R}^{166}$ was curated by a company, Molecular Design Limited, Inc. (MDL), for drug discovery tasks.[74] Thus, we hypothesize that the MACCS fingerprint encodes biologically-relevant information about pesticide molecules for predicting their toxicity to bees. Indeed, MACCS fingerprints have been found to be predictive of toxicity in other studies.[64,90–93] We use the MACCS fingerprint implementation in the RDKit,[94] whose source code lists the subgraph patterns (described by SMARTS strings) corresponding to each keybit.[95] For example, keybits 29, 134, 125, and 154 indicate the presence of phosphorus, a halogen, an aromatic ring, and a carbonyl group, respectively. Figure 3(a) illustrates further.

2. The fixed-length random walk feature vector

a. Walks on a molecular graph and label sequences along them. The random walk feature map describes a molecular graph by the set of label sequences along walks on it.

A walk. A *walk* w of length L on a molecular graph G is a sequence of vertices such that consecutive vertices are joined by an edge,

$$w = (v_1, \ldots, v_{L+1}) \text{ such that } \{v_i, v_{i+1}\} \in \mathcal{E} \quad \text{for } i \in \{1, \ldots, L\}. \quad (1)$$

The length L refers to the number of edges (not necessarily unique) traversed along the walk (for example, see Fig. 2).

Let $\mathcal{W}_L(G)$ be the set of all possible walks of length L on a graph G.

The label sequence of a walk. The *label sequence* $s = \ell_w(w)$ of a walk $w = (v_1, \ldots, v_{L+1})$ gives the progression of vertex and edge labels along the walk,

$$\ell_w(w) = [\ell_v(v_1), \ell_e(\{v_1, v_2\}), \ldots, \ell_e(\{v_L, v_{L+1}\}), \ell_v(v_{L+1})] =: s \quad (2)$$

(for example, see Fig. 2).

Let $\mathcal{S}_L = \{s_1, \ldots, s_{S_L}\}$ be the set of all possible label sequences among length-L walks on all molecular graphs $G \in \mathcal{G}$—so $|\mathcal{S}_L| = S_L$.

b. The fixed-length random walk feature map. The fixed-length random walk feature vector of a molecular graph lists the number of fixed-length walks on the graph with each possible vertex- and edge-label sequence. Thereby, the molecular graph is described by the distribution of label sequences along fixed-length (equipoise[96]) random walks on it.[97–99]

Precisely, the fixed-length-L feature map $\phi^{(L)}: \mathcal{G} \to \mathbb{R}^{S_L}$ constructs a vector representation of a graph $G \in \mathcal{G}$ whose element i is a count of length-L walks on G with label sequence s_i,

$$\phi^{(L)}(G) := [\phi^L_1(G), \ldots, \phi^{(L)}_{S_L}(G)], \quad (3)$$

$$\text{where } \phi^{(L)}_i(G) := |\{w \in \mathcal{W}_L(G) : \ell_w(w) = s_i\}|. \quad (4)$$

As a length $L = 0$ walk constitutes an atom, $\phi^{(0)}(G)$ lists counts of atom types in the molecule. As a length $L = 1$ walk constitutes two (ordered) atoms joined by a bond, $\phi^{(1)}(G)$ lists counts of each particular (ordered) pairing of atoms joined by a particular bond type in the molecule.

Figure 3(b) illustrates by listing (in an arbitrary order) the nonzero elements of $\phi^{(L=2)}(G)$ for an example molecular graph G. In both Eq. (4) and Fig. 3(b), we dodge the task of explicitly imposing an ordering of the set $\mathcal{S}_L$, since we never explicitly construct $\phi^{(L)}(G)$.

3. Comparing and contrasting the MACCS FP and random walk feature vector

Both the MACCS FP and random walk feature vector characterize a molecular graph by looking for a list of "patterns" in it—subgraph patterns for the MACCS FP and label sequences along walks for the random walk feature vector. Distinctions are as follows: (1) the MACCS FP looks for *variable-size* subgraphs, whereas the random walk feature vector looks at *fixed-size* (length) label sequences along walks; (2) the random walk feature vector *counts* patterns, whereas the MACCS FP only *indicates* the presence of patterns; (3) the random walk feature vector exhaustively counts *all possible* walk patterns, while the MACCS FP non-exhaustively looks for a pre-defined, curated *subset* of the possible subgraph patterns; (4) owing to the variably-sized list of subgraph patterns, including wildcard atoms/bonds, in the MACCS FP, multiple subgraphs can activate the same bit, and a single subgraph can activate multiple bits; (5) the MACCS FPs $\phi^{MACCS}(G) \in \mathbb{R}^{166}$ are feasible to explicitly construct and store in memory, while the fixed-length random walk feature vectors $\phi^{(L)}(G) \in \mathbb{R}^{S_L}$ are not for large L, owing to the large number of possible label sequences S_L present in length-L walks on molecular graphs;[98] given V possible vertex labels and E possible edge labels, theoretically $|\mathcal{S}_L| = V^{L+1}E^L$ label sequences are possible in length-L walks, although many of these will not be observed in any plausible chemical system.

C. The fixed-length random walk graph kernel

The fixed-length random walk kernel[98] $k^{(L)}(G, G') = \phi^{(L)}(G) \cdot \phi^{(L)}(G')$ allows us to circumvent explicit construction of $\phi^{(L)}(G)$ when employing a kernel method of machine learning, which can be cast to rely only on dot products $\phi^{(L)}(G) \cdot \phi^{(L)}(G')$ of pairs of vector representations of molecular graphs G, G'.

1. Definition and explanation of the L-RWGK

The fixed length-L random walk graph kernel[97,100] (L-RWGK) $k^{(L)}: \mathcal{G} \times \mathcal{G} \to \mathbb{R}$ is a (symmetric, positive semidefinite) function such that evaluating $k(G, G')$ is implicitly equivalent to (i) mapping the two input graphs G and G' into the random walk vector space $\mathbb{R}^{S_L}$ via the feature map $\phi^{(L)}$ and then (ii) taking the inner product of these two vectors,

$$k^{(L)}(G, G') = \phi^{(L)}(G) \cdot \phi^{(L)}(G'). \quad (5)$$

As seen from Eq. (4), term i of $k^{(L)}(G, G')$ in Eq. (5) is the number of pairs of length-L walks—one in graph G and the other in graph G'—with the label sequence $s_i \in \mathcal{S}_L$. Hence, $k^{(L)}(G, G')$ sums counts of pairs of length-L walks on the two graphs G, G' sharing a label sequence,

$$k^{(L)}(G, G') = \sum_{s \in \mathcal{S}_L} |\{w \in \mathcal{W}_L(G) : \ell_w(w) = s\} \times |\{w' \in \mathcal{W}_L(G') : \ell'_w(w') = s\}|. \quad (6)$$

As the term associated with a label sequence s is nonzero, iff *both* graphs G and G' possess a length-L walk with label sequence s, this

sum may be restricted to be over the subset of label sequences in common between length-L walks on the two graphs, $\ell_w(\mathcal{W}_L(G)) \cap \ell'_w(\mathcal{W}_L(G'))$.

Intuitively, the 0-RWGK $k^{(0)}(G, G')$ sums counts of pairs of atoms of a particular type between the two graphs G, G'. The 1-RWGK $k^{(1)}(G, G')$ sums counts of pairs of two particular (ordered) atoms joined by a particular bond.

To evaluate the L-RWGK $k^{(L)}(G, G')$ without explicitly constructing the random walk feature vectors $\phi^{(L)}(G), \phi^{(L)}(G')$, we leverage the direct product graph to count pairs of label sequences in common between walks on two graphs G, G'.

2. The direct product graph to compute RWGKs

Given two input graphs $G, G' \in \mathcal{G}$, we construct a new graph, the direct product graph $G_\times = G \times G' = (\mathcal{V}_\times, \mathcal{E}_\times, \ell_{v,\times}, \ell_{e,\times})$, to evaluate the L-RWGK $k^{(L)}(G, G')$ between G and G'. The direct product graph $G_\times$ is constructed to give a one-to-one mapping between (i) walks in $G_\times$ and (ii) pairs of walks—one on G and one on G'—with the same label sequence.

Definition of the direct product graph. Each vertex of the direct product graph $G_\times = G \times G'$ is an ordered pair of vertices—the first in G and the second in G'. The vertices of the direct product graph are constituted by the subset of pairs of vertices between G and G' with the same vertex label,

$$\mathcal{V}_\times := \{(v, v') \in \mathcal{V} \times \mathcal{V}' \mid \ell_v(v) = \ell'_v(v')\}. \tag{7}$$

An undirected edge joins two vertices of the direct product graph $G_\times = G \times G'$ iff (i) the two involved vertices of G are joined by an edge in $\mathcal{E}$ and (ii) the two involved vertices of G' are joined by an edge in $\mathcal{E}'$ and (iii) these two edges in $\mathcal{E}$ and $\mathcal{E}'$ have the same label,

$$\mathcal{E}_\times := \big\{\{(u, u'), (v, v')\} \mid (u, u') \in \mathcal{V}_\times \wedge (v, v') \in \mathcal{V}_\times \wedge \times \{u, v\} \in \mathcal{E} \wedge \{u', v'\} \in \mathcal{E}' \wedge \ell_e(\{u, v\}) = \ell'_e(\{u', v'\})\big\}. \tag{8}$$

We equip the direct product graph $G_\times = G \times G'$ with vertex- and edge-labeling functions that give the (same) label of the involved vertices and edges in G and G',

$$\ell_{v,\times}((v, v')) := \ell_v(v) = \ell'_v(v'), \tag{9}$$

$$\ell_{e,\times}(\{(u, u'), (v, v')\}) := \ell_e(\{u, v\}) = \ell'_e(\{u', v'\}). \tag{10}$$

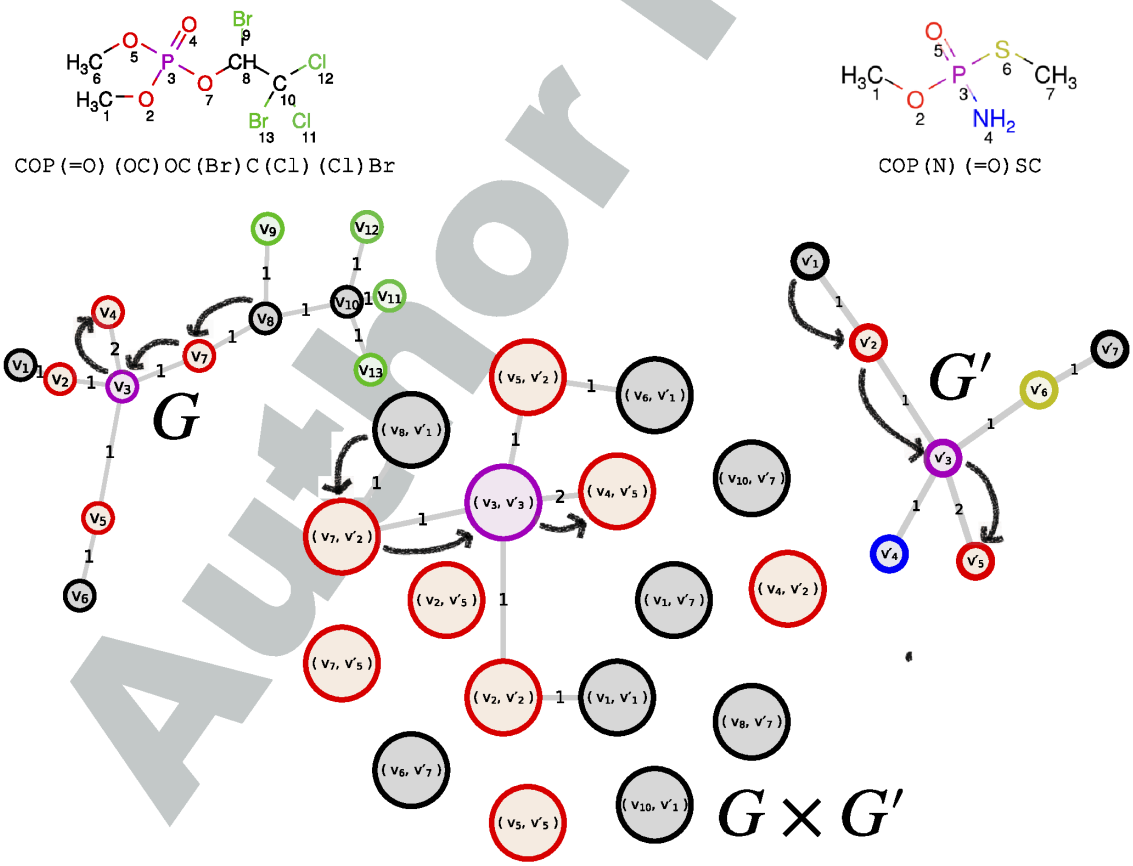


FIG. 4. Illustrating the direct product graph $G_\times = G \times G'$ of two molecular graphs G and G' representing two molecules (shown above their SMILES strings) in the BeeToxAI dataset. Vertex labels in the graphs are indicated by color. Note the one-to-one correspondence between (i) a walk on $G_\times$ and (ii) two walks on G and G' with the same label sequence. We indicate one such correspondence with the black arrows.

Figure 4 shows the direct product graph of two molecular graphs as an example.

Utility of the direct product graph for evaluating the *L*-RWGK. By construction, any given length-L walk $w_\times$ on the direct product graph $G_\times = G \times G'$ with label sequence $\ell_{w,\times}(w_\times)$ corresponds to a unique pair of walks $\{w, w'\}$, with $w \in \mathcal{W}_L(G), w' \in \mathcal{W}_L(G')$, possessing the label sequence $\ell_w(w) = \ell'_w(w') = \ell_{w,\times}(w_\times)$, and vice versa (giving a bijection). This is illustrated in Fig. 4. Therefore, all three of the following quantities are equivalent:

- the number of length-L walks on the direct product graph $G_\times = G \times G'$,
- the number of pairs of length-L walks on G and G' with the same label sequence, and
- through Eq. (6), the value of the L-RWGK $k^{(L)}(G, G')$.

The key to counting length-L walks on $G_\times = G \times G'$—and thus to evaluating $k^{(L)}(G, G')$—lies in its $|\mathcal{V}_\times| \times |\mathcal{V}_\times|$ adjacency matrix $A_\times$ whose entry (i,j) is one if vertices $v_{\times,i}, v_{\times,j} \in \mathcal{V}_\times$ are joined by an edge and zero otherwise. The number of walks of length L from vertex $v_{\times,i}$ to vertex $v_{\times,j}$ is given by element (i,j) of $A_\times^L$. Summing over all possible starting and end vertices of walks,

$$k^{(L)}(G, G') = \sum_{i=1}^{|\mathcal{V}_\times|} \sum_{j=1}^{|\mathcal{V}_\times|} [A_\times^L]_{i,j}. \tag{11}$$

Summary of evaluating the *L*-RWGK. Computing the L-RWGK $k^{(L)}(G, G')$, therefore, involves (i) constructing the direct product graph $G_\times = G \times G'$, (ii) building the adjacency matrix $A_\times$ of $G_\times$, (iii) computing the Lth power of $A_\times$, $A_\times^L$, and then (iv) summing its entries.

D. The linear kernel between two MACCS structural key fingerprints

For comparison to the L-RWGK, note the (linear) kernel applied to the MACCS FPs of a pair of molecular graphs,

$$k^{MACCS}(G, G') := \phi^{MACCS}(G) \cdot \phi^{MACCS}(G'), \tag{12}$$

gives the number of subgraph patterns in the MACCS library that are exhibited by *both* graphs G and G'.

E. Support vector machines (SVMs) as classifiers

A support vector machine (SVM)[84,85,101,102] is a supervised machine learning model for binary classification. To train an SVM using a labeled training dataset $\{(G_n, y_n)\}_{n=1}^N$, with $G_n \in \mathcal{G}$ and $y_n \in \{-1, 1\}$, we rely on a feature map $\phi : \mathcal{G} \to \mathbb{R}^F$ to represent graphs in a vector space. Such feature maps can be constructed explicitly (the case with the MACCS FP) or implicitly via the use of a kernel function $k(G, G') = \phi(G) \cdot \phi(G')$ between pairs of data (the case with the random walk feature vector). We briefly explain the SVM here. For more details, see Refs. 84 and 101.

The decision boundary. Ultimately, an SVM classifier $f(G)$ employs a hyperplane $w \cdot \phi(G) + b = 0$ in the feature space $\mathbb{R}^F$ as the decision boundary,

$$\hat{y} = f(G) = \text{sign}(w \cdot \phi(G) + b), \tag{13}$$

with $w \in \mathbb{R}^F$ normal to the hyperplane, pointing in the direction of (most) of the positive examples, and $b \in \mathbb{R}$ specifying the offset of the hyperplane from the origin. Training an SVM constitutes using the training data to find the "optimal" hyperplane described by parameters w, b.

The primal optimization problem. The (soft margin) SVM seeks a hyperplane that separates most of the training data with a large margin defined by the thickness of the region $|w \cdot \phi(G) + b| \leq 1$. The primal optimization problem associated with training an SVM is

$$\min_{w,b} \left(\frac{1}{2}\|w\|^2 + C \sum_{n=1}^N \xi_n \right), \tag{14}$$

$$\text{such that } \xi_n \geq 0 \quad \text{for } n \in \{1, \ldots, N\}, \tag{15}$$

$$y_n(w \cdot \phi(G_n) + b) \geq 1 - \xi_n \quad \text{for } n \in \{1, \ldots, N\}. \tag{16}$$

The slack variable ξ_n associated with data vector $\phi(G_n)$ allows, if it is nonzero, violation of the constraint $y_n(w \cdot \phi(G_n) + b) \geq 1$ that it lies (i) on the correct side of the decision boundary and (ii) outside of or on the boundary of the margin. The first term in the objective function describes the size of the margin; the second term penalizes constraint violations. The hyperparameter $C \geq 0$ trades a large margin for constraint violations.

The dual optimization problem. The Lagrangian dual of the primal optimization problem is in N Lagrange multipliers $\{\alpha_1, \ldots, \alpha_N\}$,

$$\max_\alpha \left(\sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \phi(G_i) \cdot \phi(G_j) \right), \tag{17}$$

$$\text{such that } 0 \leq \alpha_n \leq C \quad \text{for } n \in \{1, \ldots, N\}, \tag{18}$$

$$\sum_{n=1}^N \alpha_n y_n = 0, \tag{19}$$

where the solution to the dual problem α and the solution to the primal problem w satisfy

$$w = \sum_{n=1}^N \alpha_n y_n \phi(G_n). \tag{20}$$

The kernel trick. The objective of the dual problem in Eq. (17) depends only on the dot products $\phi(G_i) \cdot \phi(G_j)$ of the training data. The kernel trick is to replace $\phi(G_i) \cdot \phi(G_j)$ with a kernel function $k(G_i, G_j) = \phi(G_i) \cdot \phi(G_j)$ to bypass the explicit mapping of the graphs G_i and G_j into the vector space $\mathbb{R}^F$ to compute the dot product $\phi(G_i) \cdot \phi(G_j)$. Indeed, we use the L-RWGK $k^{(L)}(G_i, G_j)$ in Eq. (5) in place of constructing $\phi^{(L)}(G)$ and $\phi^{(L)}(G')$ and taking their dot product.

Using Eq. (20), we can also rewrite the decision rule in Eq. (13) for a new graph G in terms of the kernel between it and the graphs in the training dataset,

$$f(G) = \text{sign}\left(\sum_{n=1}^N \alpha_n y_n k(G_n, G) + b \right), \tag{21}$$

with α_n being the solution to the dual problem. Equation (21) allows us to also bypass mapping new molecular graphs G into the feature space $\mathbb{R}^F$ via ϕ when classifying them with the trained SVM.

The support vectors. An SVM is a *sparse* kernel machine;[101] the decision rule in Eq. (21) will depend on a *subset* of the training data, the *support vectors* $\phi^{(L)}(G_n)$ with $\alpha_n > 0$ that lie inside or on the boundary of the margin or outside the margin but on the wrong side of the decision boundary.

The Gram matrix. When, in practice, invoking the kernel trick, we store the inner products between all pairs of molecular graphs in a $N \times N$ Gram matrix K, whose element (i,j) gives the kernel $k(G_i, G_j)$ between molecular graphs G_i and G_j.

Centering. SVMs tend to perform better if the feature vectors $\{\phi(G_1), \ldots, \phi(G_N)\}$ are first centered.[103] Again to avoid explicit construction of them, the double-centering trick[84] allows us to obtain the inner products of the centered feature vectors from the inner products of the uncentered feature vectors in the Gram matrix K. Particularly, the centered Gram matrix $\tilde{K} := CKC$ with centering matrix $C = I - \frac{1}{N} oo^\top$ (I the identity matrix, o a vector of ones).[104]

F. Classification performance metrics

The performance metrics of a classifier $\hat{y} = f(G)$ include the following (measured over a labeled test dataset):

- Accuracy: fraction of examples classified correctly.
- Precision: among the examples classified as toxic ($\hat{y}_n = 1$), the fraction that are truly toxic ($y_n = 1$).
- Recall: among the examples that are truly toxic ($y_n = 1$), the fraction that are correctly predicted as toxic ($\hat{y}_n = 1$).

The F1 score is the harmonic mean of precision and recall. Owing to class imbalance [see Fig. 1(b)], the F1 score is a better performance metric of the classifier than accuracy.[105]

IV. RESULTS

We now train and evaluate the performance of a support vector machine (SVM) to classify the toxicity of pesticide molecules to honey bees using two different molecular representations:

- the fixed-length-L random walk feature vector and
- the MACCS structural key fingerprint (FP).

We explicitly construct the MACCS FPs but invoke the kernel trick and rely on the fixed-length-L random walk graph kernel (L-RWGK) in place of a dot product in the random walk feature space.

A. Machine learning procedures

Data preparation. The data are prepared from the SMILES strings representing the pesticide molecules in the BeeToxAI dataset.[58]

MACCS fingerprints. We used RDKit[94] to explicitly construct the MACCS FPs of the pesticide molecules, $\{\phi^{MACCS}(G_1), \ldots, \phi^{MACCS}(G_N)\}$. Then, we computed the dot product $\phi^{MACCS}(G_i) \cdot \phi^{MACCS}(G_j)$ between each pair of MACCS FPs and stored them in the Gram matrix K^{MACCS}.

Fixed-length random walk graph kernel. We used MolecularGraph.jl to obtain the molecular graphs $\{G_1, \ldots, G_N\}$ representing the pesticide molecules. For each pair of graphs (G_i, G_j), we constructed their direct product graph $G_i \times G_j$, evaluated the L-RWGK $k^{(L)}(G_i, G_j)$ via Eq. (11) (using our own code), and then stored it in a Gram matrix $K^{(L)}$ for $L \in \{0, \ldots, 12\}$.

A train-test run. For both the MACCS FP and fixed-length random walk representations, a "train-test run" of an SVM comprises the following procedure. First, we randomly shuffle and then split the examples into a 80%/20% train/test split. We stratify the split to preserve the distribution of class labels in the two splits. Second, using only the training split, we use stratified $K = 3$-fold cross-validation to determine the optimal hyperparameter(s). For the MACCS fingerprint, the hyperparameter is the C parameter of the SVM. For the fixed-length random walk representation, the hyperparameters are both C and L, the length of the random walks. Through grid search, we choose the optimal hyperparameter(s) as the one(s) providing the K SVMs (each trained on $K-1$ folds) with the maximal mean F1 score on the validation sets (one fold each). The hyperparameter grid comprises (i) $\log_{10} C \in \{-6, \ldots, 1\}$ and (ii) $L \in \{0, \ldots, 12\}$. Finally, we train a deployment SVM with the optimal hyperparameter(s) on all training data and evaluate its performance (precision, recall, accuracy, and F1 score) on the hold-out test set.

Note that, for each SVM trained, we center the Gram matrix K pertaining only to the training graphs via the double-centering trick.[84] We adopt a similar centering trick[104] for the Gram matrix

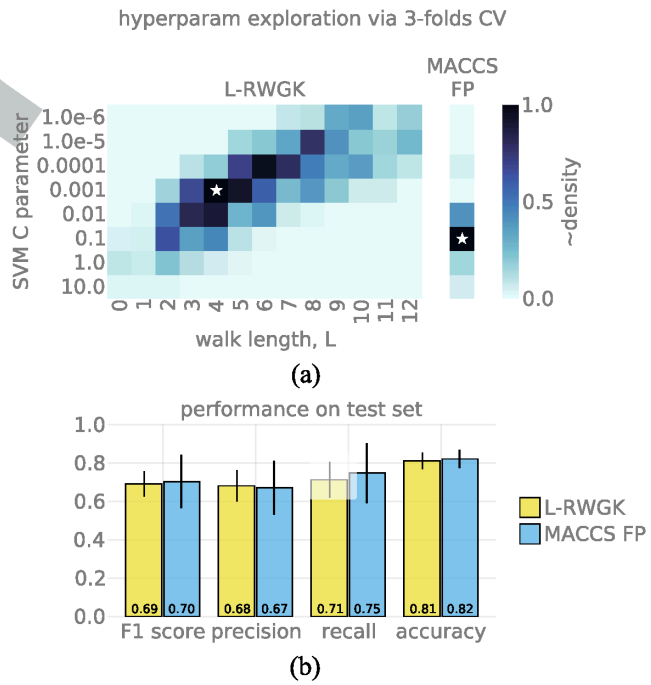


FIG. 5. Average results of the SVM toxicity classifier over 2000 (stochastic) runs of test/train splits using the (i) L-RWGK and (ii) MACCS FP with a linear kernel. (a) The empirical ~~joint~~ distribution of the optimal hyperparameters during the threefold cross-validation procedure to determine the optimal SVM C parameter and also, in the case of the L-RWGK, length L of the walks. The ⋆ marks the mode of the optimal hyperparameters. (b) Toxicity classification performance of the deployment SVM (with the optimal C, L from cross-validation) on hold-out test data. Bars show standard deviation.

giving the similarity of the test graphs with the training graphs when we feed it as input to the SVM for predictions on the test split.

We used the SVC implementation and Gram matrix centerer in scikit-learn.[106] We scaled the C parameter in Eq. (14) seen by the slack variables pertaining to each class to balance penalization of constraint violations for each class.

Overall procedure. For both the MACCS fingerprint and fixed-length random walk representations, we conducted 2000 (stochastic, owing to the random train/test and K-folds splits) train-test runs; for each run, we evaluated the performance of a hyperparameter-optimized, trained SVM classifier on the hold-out test set. Conducting multiple train-test runs allows us to report both expected performance and variance in the performance.

B. Cross-validation results

Figure 5(a) shows the empirical distribution of optimal hyperparameters during the $K = 3$-folds cross-validation routine. The mode of the distribution of the optimal C parameter for the MACCS-FP-SVM is 0.1. The mode of the joint distribution for the optimal walk length L and SVM C parameter for the L-RWGK-SVM is $L = 4$ and $C = 0.001$. In conclusion, the pesticide molecules were best described by random walks of length $L = 4$ for bee toxicity prediction. The optimal C parameter tended to decrease with the walk length, consistent with the view of the inverse of C as a regularization parameter expected to increase when the representation of the examples is more complex.

C. Classification performance on the test set

Figure 5(b) shows the mean and standard deviation of the accuracy, precision, recall, and F1 score of the L-RWGK-SVM and MACCS-FP-SVM on the hold-out test set of pesticide molecules. The performance of the L-RWGK-SVM is on par with/slightly lower than that of the MACCS-FP-SVM but has the advantage of a lower variance (see error bars). The L-RWGK-SVM achieves, on average, an F1 score, precision, recall, and accuracy of 0.69, 0.68, 0.71, and 0.81, respectively.

D. Interpreting the MACCS-FP-SVM

Explaining the predictions of and interpreting a molecular machine learning model can give chemical insights and foster trust—or distrust, by uncovering "Clever Hans" predictions—in the model.[107,108]

In contrast to the L-RWGK-SVM that leverages the kernel trick, the MACCS-FP-SVM lends interpretability because we may explicitly construct w via Eq. (20).

We interpret a MACCS-FP-SVM toxicity classifier by inspecting the vector $w \in \mathbb{R}^{166}$ normal to its separating hyperplane. Weight $w_i \in \mathbb{R}$ in w is associated with MACCS keybit i, which looks for

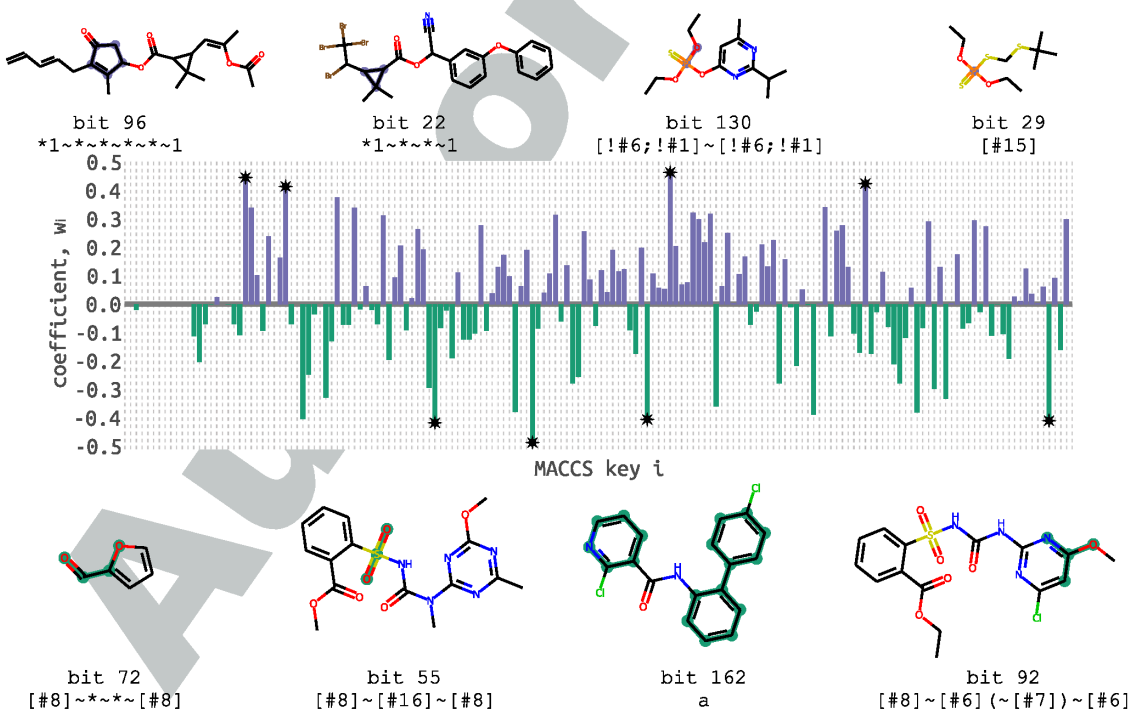


FIG. 6. Interpreting the SVM toxicity classifier operating on un-standardized MACCS fingerprints. The bar plot visualizes the $w \in \mathbb{R}^{166}$ vector normal to the separating hyperplane and pointing in the direction of the toxic examples [see Eq. (13)]. If a coefficient w_i of w is positive (negative), the presence of the corresponding molecular subgraph pattern (indicated by MACCS key i) correlates with a prediction of toxicity (non-toxicity). The MACCS keys and examples of molecules in the BeeToxAI dataset exhibiting those patterns (highlighted) for the four most positive and four most negative w_i are shown in the top and bottom, respectively (top: all toxic, bottom: all non-toxic).

a particular subgraph pattern in the molecular graph. For ease of interpretability, here we do not standardize the input MACCS FPs and instead retain them as bit vectors; consequently, a positive (negative) coefficient w_i implies the presence of the subgraph pattern described by MACCS keybit i tends to produce a prediction of toxicity (non-toxicity).

Figure 6 visualizes the w vector of our interpretable MACCS-FP-SVM, trained on all of the data and with the optimal C hyperparameter found in Fig. 5(a) ($C = 0.1$). We inspect the molecular patterns corresponding with the four most positive (top) and four most negative (bottom) coefficients (bars decorated with *)—associated with predictions of toxicity and non-toxicity, respectively. Their MACCS keybits and SMARTS strings specifying the molecular pattern they look for are shown. We also show an example molecule in the dataset exhibiting that pattern (see highlight); the molecules on the top (bottom) were correctly predicted to be toxic (non-toxic).

We caution against mistaking *association* for *causality* in our interpretation of the SVM as (i) the two random variables indicating the presence of two subgraphs (described by two MACCS keybits) in a molecule are generally not independent and (ii) anthropogenic biases[109–111] could be involved in the generation and curation of the training dataset.

E. Run times.

The majority of the computational run time for generating our results was in computing the 382×382 Gram matrices $K^{(L)}$ involving the L-RWGK. Using four cores, the run time ranged from less than five minutes ($L = 0, 1$) to ~20–25 minutes for $L \geq 7$ (see the supplementary material).

V. DISCUSSION

We trained and evaluated a support vector machine classifier that predicts the toxicity of pesticides to honey bees. We compared two molecular vector representations: (1) MACCS fingerprints listing the presence/absence of a set of pre-defined subgraph patterns in the molecular graph and (2) a random walk feature vector listing counts of label sequences along all fixed-length walks on the molecular graph. While we explicitly construct the fingerprints, we relied on the fixed-length random walk graph kernel for dot products in the random walk vector space. The classifier using the MACCS fingerprints (a) gave a slightly higher mean F1 score (0.70 vs 0.69) than the classifier using the random walk feature, (b) grants a degree of interpretability, but (c) exhibits a higher variance in performance.

Graph kernels have been previously used with SVMs, Gaussian processes, and kernel regression for molecular machine learning tasks,[79,112] such as to classify proteins,[113] score protein–protein interactions,[114] predict methane uptake in nanoporous materials,[115] predict atomization energy of molecules,[116,117] and predict thermodynamic properties of pure substances.[118]

A Gaussian process model[86] using the L-RWGK would enable uncertainty quantification in the prediction.

As the BeeToxAI dataset[58] was collected from many sources, including the scientific literature, anthropogenic biases could be present, e.g., in the choices of which molecules to be tested for bee toxicity. As a result, the training and testing data distributions could differ, and the performance measure on the hold-out set (sampled from the training distribution) may not reflect the generalization error when the model is deployed. Reference 111 articulates various types of this "dataset shift." Anthropogenic biases in datasets in chemistry have been uncovered and shown to cause machine learning algorithms trained on them to exhibit poorer generalization performance.[109,110]

Disadvantages of the L-RWGK include (i) its compute- and memory-intensity to evaluate, hence poor scalability to large molecules and large datasets,[79] and (ii) tottering. Expanding on (ii), by definition, the vertices in a walk [see Eq. (1)] may not be distinct (then, it would be a *path*). Thus, long walks that totter back and forth between the same few vertices—e.g., at the extreme: $w = (u, v, u, v, \ldots, u, v)$—are accounted for in the L-RWGK. These walks do not contribute extra information about the similarity of two graphs—e.g., for our extreme example, no more information beyond the length-2 walk (u, v). Tottering could thus lead to a "dilution" of the similarity metric expressed by the random walk kernel.[100] Modification of the random walk kernel can prevent tottering walks[119] from contributing to the similarity metric.

The L-RWGK can be generalized further by defining a kernel between two walks w and w' as a product of the kernel between the edges and vertices along the walk.[79,98,113] A (non-Dirac) kernel between vertices could account for similarity of chemical elements.

In addition to the fixed-length-L random walk kernel, ~~we employed~~ the (i) max-length-L random walk kernel and (ii) geometric random walk kernel count pairs of length-ℓ walks with a shared label sequence (i) for $\ell \in \{0, \ldots, L\}$ and (ii) $\ell \in \{0, \ldots\}$.[98,100]

In addition to random walk kernels, other graph kernels can be used to express the similarity of molecular graphs:[79] shortest-path,[120] graphlet,[121] tree- and cyclic-pattern,[122,123] and optimal assignment kernels.[124]

SUPPLEMENTARY MATERIAL

The supplementary material includes (i) a comparison of the (a) linear kernel between MACCS FPs and (b) the $L = 4$ random walk graph kernel and (ii) the run times for computing the random walk graph kernel.

ACKNOWLEDGMENTS

We thank Jana Doppa and Aryan Deshwal for stimulating Cory's interest in graph kernels. We acknowledge support from the National Science Foundation (Award No. 1920945).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Ping Yang: Data curation (lead); Formal analysis (lead); Investigation (lead); Methodology (lead); Software (lead); Writing – original draft (equal); Writing – review & editing (equal). **E. Adrian Henle:** Formal analysis (equal); Methodology (equal); Software (equal); Validation (equal); Writing – review & editing (equal). **Xiaoli Z. Fern:**

Methodology (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal). **Cory M. Simon:** Conceptualization (equal); Software (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data and code to fully reproduce this study are available in GitHub (Ref. 125).

REFERENCES

[1] F. P. Carvalho, "Agriculture, pesticides, food security and food safety," Environ. Sci. Policy **9**(7–8), 685–692 (2006).

[2] E.-C. Oerke, "Crop losses to pests," J. Agric. Sci. **144**(1), 31–43 (2006).

[3] J. Popp, K. Pető, and J. Nagy, "Pesticide productivity and food security. a review," Agron. Sustainable Dev. **33**(1), 243–255 (2013).

[4] J. Cooper and H. Dobson, "The benefits of pesticides to mankind and the environment," J. Crop Prot. **26**(9), 1337–1348 (2007).

[5] P. Nicolopoulou-Stamati, S. Maipas, C. Kotampasi, P. Stamatis, and L. Hens, "Chemical pesticides and human health: The urgent need for a new concept in agriculture," Front. Public Health **4**, 148 (2016).

[6] C. Wilson and C. Tisdell, "Why farmers continue to use pesticides despite environmental, health and sustainability costs," Ecol. Econ. **39**(3), 449–462 (2001).

[7] I. Mahmood, S. R. Imadi, K. Shazadi, A. Gul, and K. R. Hakeem, "Effects of pesticides on environment," in *Plant, Soil and Microbes* (Springer, 2016), pp. 253–269.

[8] D. Tilman, J. Fargione, B. Wolff, C. D'Antonio, A. Dobson, R. Howarth, D. Schindler, W. H. Schlesinger, D. Simberloff, and D. Swackhamer, "Forecasting agriculturally driven global environmental change," Science **292**(5515), 281–284 (2001).

[9] S. Stehle and R. Schulz, "Agricultural insecticides threaten surface waters at the global scale," Proc. Natl. Acad. Sci. U. S. A. **112**(18), 5750–5755 (2015).

[10] S. Johnson and G. Preetha, *Pesticide Toxicity to Non-target Organisms* (Springer, 2016).

[11] L. W. Pisa, V. Amaral-Rogers, L. P. Belzunces, J. M. Bonmatin, C. A. Downs, D. Goulson, D. P. Kreutzweiser, C. Krupke, M. Liess *et al.*, "Effects of neonicotinoids and fipronil on non-target invertebrates," Environ. Sci. Pollut. Res. **22**(1), 68–102 (2015).

[12] B. P. Oldroyd, "What's killing American honey bees?," PLoS Biol. **5**(6), e168 (2007).

[13] B. A. Woodcock, J. M. Bullock, R. F. Shore, M. S. Heard, M. G. Pereira, J. Redhead, L. Ridding, H. Dean, D. Sleep, P. Henrys *et al.*, "Country-specific effects of neonicotinoid pesticides on honey bees and wild bees," Science **356**(6345), 1393–1395 (2017).

[14] A. J. Vanbergen and the Insect Pollinators Initiative, "Threats to an ecosystem service: Pressures on pollinators," Front. Ecol. Environ. **11**(5), 251–259 (2013).

[15] R. J. Gill, O. Ramos-Rodriguez, and N. E. Raine, "Combined pesticide exposure severely affects individual-and colony-level traits in bees," Nature **491**(7422), 105–108 (2012).

[16] D. Goulson, "REVIEW: An overview of the environmental risks posed by neonicotinoid insecticides," J. Appl. Ecol. **50**(4), 977–987 (2013).

[17] D. VanEngelsdorp, J. Hayes, Jr., R. M. Underwood, and J. Pettis, "A survey of honey bee colony losses in the U.S., fall 2007 to spring 2008," PloS One **3**(12), e4071 (2008).

[18] I. Koh, E. V. Lonsdorf, N. M. Williams, C. Brittain, R. Isaacs, J. Gibbs, and T. H. Ricketts, "Modeling the status, trends, and impacts of wild bee abundance in the United States," Proc. Natl. Acad. Sci. U. S. A. **113**(1), 140–145 (2016).

[19] S. A. Cameron, J. D. Lozier, J. P. Strange, J. B. Koch, N. Cordes, L. F. Solter, and T. L. Griswold, "Patterns of widespread decline in north American bumble bees," Proc. Natl. Acad. Sci. U. S. A. **108**(2), 662–667 (2011).

[20] M. Spivak, E. Mader, M. Vaughan, and N. H. Euliss, Jr., "The plight of the bees," Environ. Sci. Technol. **45**, 34 (2011).

[21] D. Goulson, E. Nicholls, C. Botías, and E. L. Rotheray, "Bee declines driven by combined stress from parasites, pesticides, and lack of flowers," Science **347**(6229), 1255957 (2015).

[22] N. Gallai, J.-M. Salles, J. Settele, and B. E. Vaissière, "Economic valuation of the vulnerability of world agriculture confronted with pollinator decline," Ecol. Econ. **68**(3), 810–821 (2009).

[23] R. Winfree, N. M. Williams, H. Gaines, J. S. Ascher, and C. Kremen, "Wild bee pollinators provide the majority of crop visitation across land-use gradients in New Jersey and Pennsylvania, USA," J. Appl. Ecol. **45**(3), 793–802 (2008).

[24] A.-M. Klein, B. E. Vaissiere, J. H. Cane, I. Steffan-Dewenter, S. A. Cunningham, C. Kremen, and T. Teja, "Importance of pollinators in changing landscapes for world crops," Proc. R. Soc. B **274**(1608), 303–313 (2007).

[25] P. H. Raven, G. B. Johnson, J. B. Losos, K. A. Mason, and S. R. Singer, *Biology* (McGraw-Hill Higher Education, 2008).

[26] M. D. Levin, "Value of bee pollination to U.S. agriculture," Am. Entomol. **29**(4), 50–51 (1983).

[27] K. J. Hung, J. M. Kingston, M. Albrecht, D. A. Holway, and J. R. Kohn, "The worldwide importance of honey bees as pollinators in natural habitats," Proc. R. Soc. B **285**(1870), 20172140 (2018).

[28] T. C. Sparks and R. Nauen, "IRAC: Mode of action classification and insecticide resistance management," Pestic. Biochem. Physiol. **121**, 122–128 (2015).

[29] M. D. K. Owen and I. A. Zelaya, "Herbicide-resistant crops and weed resistance to herbicides," Pest Manage. Sci **61**(3), 301–311 (2005).

[30] J. A. Lucas, N. J. Hawkins, and B. A. Fraaije, "Chapter two - The evolution of fungicide resistance," in *Advances in Applied Microbiology*, edited by, S. Sariaslani and G. Michael Gadd (Academic Press, 2015), Vol. 90, pp. 29–92.

[31] N. Umetsu and Y. Shirai, "Development of novel pesticides in the 21st century," J. Pestic. Sci. **45**(2), 54–74 (2020).

[32] S. F. Sousa, P. A. Fernandes, and M. J. Ramos, "Protein–ligand docking: Current status and future challenges," Proteins **65**(1), 15–26 (2006).

[33] F. Gang, X. Li, C. Yang, L. Han, H. Qian, S. Wei, W. Wu, and J. Zhang, "Synthesis and insecticidal activity evaluation of virtually screened phenylsulfonamides," J. Agric. Food Chem. **68**(42), 11665–11671 (2020).

[34] T. Harada, Y. Nakagawa, T. Ogura, Y. Yamada, T. Ohe, and H. Miyagawa, "Virtual screening for ligands of the insect molting hormone receptor," J. Chem. Inf. Model. **51**(2), 296–305 (2011).

[35] X. Hu, B. Yin, K. Cappelle, L. Swevers, G. Smagghe, X. Yang, and L. Zhang, "Identification of novel agonists and antagonists of the ecdysone receptor by virtual screening," J. Mol. Graphics Modell. **81**, 77–85 (2018).

[36] T.-T. Yao, S.-W. Fang, Z.-S. Li, D.-X. Xiao, J.-L. Cheng, H.-Z. Ying, Y.-J. Ying, J.-H. Zhao, and X.-W. Dong, "Discovery of novel succinate dehydrogenase inhibitors by the integration of in silico library design and pharmacophore mapping," J. Agric. Food Chem. **65**(15), 3204–3211 (2017).

[37] S. Horoiwa, T. Yokoi, S. Masumoto, S. Minami, C. Ishizuka, H. Kishikawa, S. Ozaki, S. Kitsuda, Y. Nakagawa, and H. Miyagawa, "Structure-based virtual screening for insect ecdysone receptor ligands using MM/PBSA," Bioorg. Med. Chem. **27**(6), 1065–1075 (2019).

[38] G. J. Correy, D. Zaidman, A. Harmelin, S. Carvalho, P. D. Mabbitt, V. Calaora, P. J. James, A. C. Kotze, C. J. Jackson, and N. London, "Overcoming insecticide resistance through computational inhibitor design," Proc. Natl. Acad. Sci. U. S. A. **116**(42), 21012–21021 (2019).

[39] K. Teralı, "An evaluation of neonicotinoids' potential to inhibit human cholinesterases: Protein–ligand docking and interaction profiling studies," J. Mol. Graphics Modell. **84**, 54–63 (2018).

[40] A. Decourtye, M. Henry, and N. Desneux, "Overhaul pesticide testing on bees," Nature **497**(7448), 188 (2013).

[41] See https://www.epa.gov/pollinator-protection/pollinator-risk-assessment-guidance for United States Environmental Protection Agency. Pollinator risk assessment guidance; accessed 20 February 2022.

[42] G. J. Myatt, E. Ahlberg, Y. Akahori, D. Allen, Amberg A., L. T. Anger, A. Aptula, S. Auerbach, L. Beilke, P. Bellion, R. Benigni, J. Bercu, E. D. Booth, D. Bower, A. Brigo, N. Burden, Cammerer Z., M. T. D. Cronin, K. P. Cross, L. Custer, M. Dettwiler, K. Dobo, K. A. Ford, M. C. Fortin, S. E. Gad-McDonald, N. Gellatly, V. Gervais, K. P. Glover, Glowienke S., J. Van Gompel, S. Gutsell, B. Hardy, J. S. Harvey, J. Hillegass, M. Honma, J.-H. Hsieh, C.-W. Hsu, K. Hughes, C. Johnson, R. Jolly, D. Jones, R. Kemper, M. O. Kenyon, M. T. Kim, N. L. Kruhlak, S. A. Kulkarni, K. Kümmerer, P. Leavitt, B. Majer, S. Masten, S. Miller, J. Moser, M. Mumtaz, W. Muster, L. Neilson, T. I. Oprea, G. Patlewicz, A. Paulino, E. Lo Piparo, M. Powley, D. P. Quigley, M. V. Reddy, A.-N. Richarz, P. Ruiz, B. Schilter, R. Serafimova, W. Simpson, L. Stavitskaya, R. Stidl, D. Suarez-Rodriguez, D. T. Szabo, A. Teasdale, A. Trejo-Martin, J.-P. Valentin, A. Vuorinen, B. A. Wall, P. Watts, A. T. White, J. Wichard, K. L. Witt, A. Woolley, D. Woolley, C. Zwickl, and C. Hasselgren, "*In silico* toxicology protocols," Regul. Toxicol. Pharmacol. **96**, 1–17 (2018).

[43] A. B. Raies and V. B. Bajic, "*In silico* toxicology: Computational methods for the prediction of chemical toxicity," Wiley Interdiscip. Rev.: Comput. Mol. Sci. **6**(2), 147–172 (2016).

[44] F. A. Quintero, S. J. Patel, F. Muñoz, and M. Sam Mannan, "Review of existing QSAR/QSPR models developed for properties used in hazardous chemicals classification system," Ind. Eng. Chem. Res. **51**(49), 16101–16115 (2012).

[45] I. Takao, N. Motoyama, J. T. Ambrose, and R. M. Roe, "Mechanism for the differential toxicity of neonicotinoid insecticides in the honey bee, *Apis mellifera*," J. Crop Prot. **23**(5), 371–378 (2004).

[46] D. Laurino, M. Porporato, A. Patetta, A. Manino *et al.*, "Toxicity of neonicotinoid insecticides to honey bees: Laboratory tests," Bull. Insectology **64**(1), 107–113 (2011).

[47] F. Sanchez-Bayo and K. Goka, "Pesticide residues and bees – A risk assessment," PLoS One **9**(4), e94482 (2014).

[48] E. L. Atkins and D. Kellum, "Comparative morphogenic and toxicity studies on the effect of pesticides on honeybee brood," J. Apic. Res. **25**(4), 242–255 (1986).

[49] H. M. Thompson, "Assessing the exposure and toxicity of pesticides to bumblebees (*Bombus* sp.)," Apidologie **32**(4), 305–321 (2001).

[50] Y.-T. Hu, T.-C. Wu, E.-C. Yang, P.-C. Wu, P.-T. Lin, and Y.-L. Wu, "Regulation of genes related to immune signaling and detoxification in *Apis mellifera* by an inhibitor of histone deacetylation," Sci. Rep. **7**(1), 41255–41314 (2017).

[51] K. Pohorecka, T. Szczęsna, M. Witek, A. Miszczak, and P. Sikorski, "The exposure of honey bees to pesticide residues in the hive environment with regard to winter colony losses," J. Apic. Sci. **61**(1), 105–125 (2017).

[52] C. A. Mullin, M. Frazier, J. L. Frazier, S. Ashcraft, R. Simonds, D. VanEngelsdorp, and J. S. Pettis, "High levels of miticides and agrochemicals in North American apiaries: Implications for honey bee health," PLoS One **5**(3), e9754 (2010).

[53] A. Decourtye, J. Devillers, E. Genecque, K. L. Menach, H. Budzinski, S. Cluzeau, and M. H. Pham-Delègue, "Comparative sublethal toxicity of nine pesticides on olfactory learning performances of the honeybee *Apis mellifera*," Arch. Environ. Contam. Toxicol. **48**(2), 242–250 (2005).

[54] T. S. Bovi, R. Zaluski, and R. O. Orsi, "Toxicity and motor changes in Africanized honey bees (*Apis mellifera* L.) exposed to fipronil and imidacloprid," An. Acad. Bras. Cienc. **90**, 239–245 (2018).

[55] M. E. I. Badawy, H. M. Nasr, and E. I. Rabea, "Toxicity and biochemical changes in the honey bee *Apis mellifera* exposed to four insecticides under laboratory conditions," Apidologie **46**(2), 177–193 (2015).

[56] N. Tsvetkov, O. Samson-Robert, K. Sood, H. S. Patel, D. A. Malena, P. H. Gajiwala, P. Maciukiewicz, V. Fournier, and A. Zayed, "Chronic exposure to neonicotinoids reduces honey bee health near corn crops," Science **356**(6345), 1395–1397 (2017).

[57] M. E. I. Badawy, H. M. Nasr, and E. I. Rabea, "Toxicity and biochemical changes in the honey bee *Apis mellifera* exposed to four insecticides under laboratory conditions," Apidologie **46**(2), 177–193 (2015).

[58] J. T. Moreira-Filho, R. C. Braga, J. M. Lemos, V. M. Alves, J. V. V. B. Borba, W. S. Costa, N. Kleinstreuer, E. N. Muratov, C. H. Andrade, and B. J. Neves, "BeeToxAI: An artificial intelligence-based web app to assess acute toxicity of chemicals to honey bees," Artif. Intell. Life Sci. **1**, 100013 (2021).

[59] F. Wang, J.-F. Yang, M.-Y. Wang, C.-Y. Jia, X.-X. Shi, G.-F. Hao, and G.-F. Yang, "Graph attention convolutional neural network model for chemical poisoning of honey bees' prediction," Sci. Bull. **65**(14), 1184–1191 (2020).

[60] E. Carnesecchi, A. A. Toropov, A. P. Toropova, N. Kramer, C. Svendsen, J. L. Dorne, and E. Benfenati, "Predicting acute contact toxicity of organic binary mixtures in honey bees (*A. mellifera*) through innovative QSAR models," Sci. Total Environ. **704**, 135302 (2020).

[61] M. Hamadache, O. Benkortbi, S. Hanini, and A. Amrane, "QSAR modeling in ecotoxicological risk assessment: Application to the prediction of acute contact toxicity of pesticides on bees (*Apis mellifera* L.)," Environ. Sci. Pollut. Res. **25**(1), 896–907 (2017).

[62] F. Como, E. Carnesecchi, S. Volani, J. L. Dorne, J. Richardson, A. Bassan, M. Pavan, and E. Benfenati, "Predicting acute contact toxicity of pesticides in honeybees (*Apis mellifera*) through a k-nearest neighbor model," Chemosphere **166**, 438–444 (2017).

[63] X. Xu, P. Zhao, Z. Wang, X. Zhang, Z. Wu, W. Li, Y. Tang, and G. Liu, "In silico prediction of chemical acute contact toxicity on honey bees via machine learning methods," Toxicol. In Vitro **72**, 105089 (2021).

[64] X. Li, Y. Zhang, H. Chen, H. Li, and Y. Zhao, "Insights into the molecular basis of the acute contact toxicity of diverse organic chemicals in the honey bee," J. Chem. Inf. Model. **57**(12), 2948–2957 (2017).

[65] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, "Machine learning for molecular and materials science," Nature **559**(7715), 547–555 (2018).

[66] L. David, A. Thakkar, R. Mercado, and O. Engkvist, "Molecular representations in AI-driven drug discovery: A review and practical guide," J. Cheminformatics **12**(1), 56 (2020).

[67] D. S. Wigh, J. M. Goodman, and A. A. Lapkin, "A review of molecular representation in the age of machine learning," Wiley Interdiscip. Rev.: Comput. Mol. Sci. ■, e1603 (2022).

[68] L. Pattanaik and C. W. Coley, "Molecular representation: Going long on fingerprints," Chem **6**(6), 1204–1207 (2020).

[69] L. Pattanaik, O.-E. Ganea, I. Coley, K. F. Jensen, W. H. Green, and C. W. Coley, "Message passing networks for molecules with tetrahedral chirality," arXiv:2012.00094 (2020).

[70] T. Le, V. C. Epa, F. R. Burden, and D. A. Winkler, "Quantitative structure – Property relationship modeling of diverse materials properties," Chem. Rev. **112**(5), 2889–2919 (2012).

[71] A. Simon, D. Schwalbe-Koda, S. Mohapatra, D. James, K. P. Greenman, and R. Gómez-Bombarelli, "Learning matter: Materials design with machine learning and atomistic simulations," Acc. Mater. Res. **3**, 343 (2022).

[72] A. Cereto-Massagué, M. J. Ojeda, Valls, C., M. Mulero, S. Garcia-Vallvé, and G. Pujadas, "Molecular fingerprint similarity search in virtual screening," Methods **71**, 58–63 (2015).

[73] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: Moving beyond fingerprints," J. Comput.-Aided Mol. Des. **30**(8), 595–608 (2016).

[74] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse, "Reoptimization of MDL keys for use in drug discovery," J. Chem. Inf. Comput. Sci. **42**(6), 1273–1280 (2002).

[75] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," arXiv:1709.05584 (2017).

[76] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning* (PMLR, 2017), pp. 1263–1272.

[77] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," IEEE Trans. Neural Networks Learn. Syst. **32**(1), 4–24 (2020).

[78] S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt, "Graph kernels," J. Mach. Learn. Res. **11**, 1201–1242 (2010).

[79] M. Rupp and G. Schneider, "Graph kernels for molecular similarity," Mol. Inf. **29**(4), 266–273 (2010).

[80] K. Borgwardt, E. Ghisu, F. Llinares-López, L. O'Bray, and B. Rieck, "Graph kernels: State-of-the-art and future challenges," arXiv:2011.03854 (2020).

[81] J. Ramon and T. Gärtner, "Expressivity versus efficiency of graph kernels," in *Proceedings of the First International Workshop on Mining Graphs, Trees and Sequences* (■, 2003), pp. 65–74.

[82]L. Ralaivola, S. J. Swamidass, H. Saigo, and P. Baldi, "Graph kernels for chemical informatics," Neural Networks **18**(8), 1093–1110 (2005).

[83]G. Nikolentzos, G. Siglidis, and M. Vazirgiannis, "Graph kernels: A survey," J. Artif. Intell. Res. **72**, 943–1027 (2021).

[84]K. P. Murphy, *Probabilistic Machine Learning: An Introduction* (MIT Press, 2022).

[85]C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learn. **20**(3), 273–297 (1995).

[86]C. E. Rasmussen and C. K. I. Williams, "Gaussian processes for machine learning," *Adaptive Computation and Machine Learning* (MIT Press, 2006).

[87]S. S. Du, K. Hou, R. R. Salakhutdinov, B. Poczos, R. Wang, and K. Xu, "Graph neural tangent kernel: Fusing graph neural networks with graph kernels," Adv. Neural Inf. Process. Syst. **32**, ■ (2019).

[88]Y. Xiang, Y.-H. Tang, G. Lin, and H. Sun, "A comparative study of marginalized graph kernel and message-passing neural network," J. Chem. Inf. Model. **61**(11), 5414–5424 (2021).

[89]See https://www.epa.gov/sites/default/files/2014-06/documents/pollinator_risk_assessment_guidance_06_19_14.pdf for Office of Pesticide Programs; United States Environmental Protection Agency. Guidance for assessing pesticide risks to bees, 2014; accessed 3 May 2022.

[90]Z. Yin, H. Ai, L. Zhang, G. Ren, Y. Wang, Q. Zhao, and H. Liu, "Predicting the cytotoxicity of chemicals using ensemble learning methods and molecular fingerprints," J. Appl. Toxicol. **39**(10), 1366–1377 (2019).

[91]X. Li, L. Chen, F. Cheng, Z. Wu, H. Bian, C. Xu, W. Li, G. Liu, X. Shen, and Y. Tang, "In silico prediction of chemical acute oral toxicity using multi-classification methods," J. Chem. Inf. Model. **54**(4), 1061–1069 (2014).

[92]D.-S. Cao, Y.-N. Yang, J.-C. Zhao, J. Yan, S. Liu, Q.-N. Hu, Q.-S. Xu, and Y.-Z. Liang, "Computer-aided prediction of toxicity with substructure pattern and random forest," J. Chemom. **26**(1–2), 7–15 (2012).

[93]C. Zhang, F. Cheng, W. Li, G. Liu, Pw. Lee, and Y. Tang, "*In silico* prediction of drug induced liver toxicity using substructure pattern recognition method," Mol. Inf. **35**(3–4), 136–144 (2016).

[94]G. Landrum *et al.*, "RDKit: A software suite for cheminformatics, computational chemistry, and predictive modeling," ■ ■, ■ (2013).

[95]See https://github.com/rdkit/rdkit/blob/master/rdkit/Chem/MACCSkeys.py for RDKit. RDKit source code for MACCS fingerprint; accessed May 23 2022.

[96]D. J. Klein, J. L. Palacios, M. Randić, and N. Trinajstić, "Random walks and chemical graph theory," J. Chem. Inf. Comput. Sci. **44**(5), 1521–1525 (2004).

[97]T. Gärtner, P. Flach, and S. Wrobel, "On graph kernels: Hardness results and efficient alternatives, in *Learning Theory and Kernel Machines* (Springer, 2003), pp. 129–143.

[98]N. M. Kriege, M. Neumann, C. Morris, K. Kersting, and P. Mutzel, "A unifying view of explicit and implicit feature maps of graph kernels," Data Min. Knowl. Discovery **33**(6), 1505–1547 (2019).

[99]H. Kashima, K. Tsuda, and A. Inokuchi, "Marginalized kernels between labeled graphs," in *Proceedings of the 20th International Conference on Machine Learning* (ICML-03, 2003), pp. 321–328.

[100]M. N. Kriege, F. D. Johansson, and C. Morris, "A survey on graph kernels," Appl. Networks Sci. **5**(1), 6 (2020).

[101]C. M. Bishop and N. M. Nasrabadi, *Pattern Recognition and Machine Learning* (Springer, 2006), Vol. 4.

[102]A. Ben-Hur and J. Weston, "A user's guide to support vector machines, in *Data Mining Techniques for the Life Sciences* (Springer, 2010), pp. 223–239.

[103]M. Meilă, "Data centering in feature space," in *International Workshop on Artificial Intelligence and Statistics* (PMLR, 2003), pp. 209–216.

[104]B. Schölkopf, A. Smola, and Müller, K.-R., "Nonlinear component analysis as a kernel eigenvalue problem," Neural Comput. **10**(5), 1299–1319 (1998).

[105]T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," PloS One **10**(3), e0118432 (2015).

[106]F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res. **12**, 2825–2830 (2011).

[107]W. Samek and K.-R. Müller, "Towards explainable artificial intelligence," in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (Springer, 2019), pp. 5–22.

[108]G. P. Wellawatte, A. Seshadri, and A. D. White, "Model agnostic generation of counterfactual explanations for molecules," Chem. Sci. **13**(13), 3697–3705 (2022).

[109]X. Jia, A. Lynch, Y. Huang, M. Danielson, I. Lang'at, A. Milder, A. E. Ruby, H. Wang, S. A. Friedler, A. J. Norquist, and J. Schrier, "Anthropogenic biases in chemical reaction data hinder exploratory inorganic synthesis," Nature **573**(7773), 251–255 (2019).

[110]D. P. Kovács, W. McCorkindale, and A. A. Lee, "Quantitative interpretation explains machine learning models for chemical reaction prediction and uncovers bias," Nat. Commun. **12**(1), 1695 (2021).

[111]S. Amos, "When training and test sets are different: Characterizing learning transfer," Dataset Shift Mach. Learn. **30**, 3–28 (2009).

[112]M. P. Preeja and K. P. Soman, "Walk-based graph kernel for drug discovery: A review," Int. J. Comput. Appl. **94**, 1–7 (2014).

[113]K. M. Borgwardt, C. S. Ong, S. Schönauer, S. V. Vishwanathan, A. J. Smola, and H. P. Kriegel, "Protein function prediction via graph kernels," Bioinformatics **21**(suppl 1), i47–i56 (2005).

[114]C. Geng, Y. Jung, N. Renaud, V. Honavar, A. M. J. J. Bonvin, and L. C. Xue, "iScore: a novel graph kernel-based function for scoring protein—protein docking models," Bioinformatics **36**(1), 112–121 (2019).

[115]H. Ohno and Y. Mukae, "Machine learning approach for prediction and search: Application to methane storage in a metal–organic framework," J. Phys. Chem. C **120**(42), 23963–23968 (2016).

[116]Y.-H. Tang and W. A. de Jong, "Prediction of atomization energy using graph kernel and active learning," J. Chem. Phys. **150**(4), 044107 (2019).

[117]G. Ferré, T. Haut, and K. Barros, "Learning molecular energies using localized graph kernels," J. Chem. Phys. **146**(11), 114107 (2017).

[118]Y. Xiang, Y.-H. Tang, H. Liu, G. Lin, and H. Sun, "Predicting single-substance phase diagrams: A kernel approach on graph representations of molecules," J. Phys. Chem. A **125**(20), 4488–4497 (2021).

[119]P. Mahé, N. Ueda, T. Akutsu, J.-L.Perret, and J.-P. Vert, "Extensions of marginalized graph kernels," in *Proceedings of the Twenty-first International conference on Machine learning* (■, 2004), p. 70.

[120]K. M. Borgwardt and H.-P. Kriegel, "Shortest-path kernels on graphs," in *Fifth IEEE International Conference on Data Mining (ICDM'05)* (IEEE, 2005), p. 8.

[121]N. Shervashidze, S. V. N. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt, "Efficient graphlet kernels for large graph comparison," in *Artificial Intelligence and Statistics* (PMLR, 2009), pp. 488–495.

[122]T. Horváth, T. Gärtner, and S. Wrobel, "Cyclic pattern kernels for predictive graph mining," in *Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (■, 2004), pp. 158–167.

[123]P. Mahé and J.-P. Vert, "Graph kernels based on tree patterns for molecules," Mach. Learn. **75**(1), 3–35 (2009).

[124]H. Fröhlich, J. K. Wegner, F. Sieker, and A. Zell, "Optimal assignment kernels for attributed molecular graphs," in *Proceedings of the 22nd International Conference on Machine Learning* (■, 2005), pp. 225–232.

[125]C. Simon, J. K. Wegner, F. Sieker, and A. Zell, "SimonEnsemble/graph-kernel-SVM-for-toxicity-of-pesticides-to-bees," https://github.com/SimonEnsemble/graph-kernel-SVM-for-toxicity-of-pesticides-to-bees (2022).