Supporting Information

Expediting Field-Effect Transistor Chemical Sensor Design with Neuromorphic Spiking Graph Neural Networks

Rodrigo P. Ferreira[1, 2†], Rui Ding[1, 2†], Fengxue Zhang[3], Haihui Pu[1, 2], Claire Donnat[4*], Yuxin Chen[3*], Junhong Chen[1, 2*]

[1] Pritzker School of Molecular Engineering, University of Chicago, 5640 S Ellis Ave., Chicago, IL 60637, United States

[2] Chemical Sciences and Engineering Division, Physical Sciences and Engineering Directorate, Argonne National Laboratory, 9700 S Cass Ave., Lemont, IL 60439, United States

[3] Department of Computer Science, University of Chicago, 5730 S Ellis Ave., Chicago, IL 60637, United States

[4] Department of Statistics, University of Chicago, 5747 S Ellis Ave., Chicago, IL 60637, United States

[†]These authors contributed equally to this work

[*]Corresponding Authors: junhongchen@uchicago.edu, chenyuxin@uchicago.edu, cdonnat@uchicago.edu,

Methods

Literature Mining and DOI Compilation - Phase I

To initiate the process, we designed query strings for PubMed, Scopus, and Web of Science, targeting articles on FET sensors published between 2010 and 2024. These queries broadly scoped keywords like “field effect transistor,” “sensor,” “chemical sensors,” and “gas detectors,” focusing on peer-reviewed literature in English (“Phase_I” folder in the Github repository). The resulting DOI lists, obtained through these queries, were refined by removing duplicates. Subsequently, we investigated each DOI using an LLM-assisted template to systematically extract and organize critical data, forming the basis for the subsequent phases of the study.

Semi-Automated LLM-Assisted Data Extraction - Phase II

Building on the curated DOI list from Phase I, we retrieved full-text articles to extract relevant information using a question template specifically designed for LLM assistance. This structured template enabled ChatGPT to systematically parse each document and extract key parameters such as sensor type, detection target, LDL values in parts per million (ppm), probe materials, operational conditions, and mediums. Despite the efficiency gains provided by this semi-automated process, manual validation was integral to maintaining rigor and ensuring high data quality.

Manual checks focused on critical areas prone to complexity or ambiguity, such as converting LDL values from molar concentrations (e.g., μM) into ppm, requiring molecular weight calculations. Additionally, papers that described complex substances like enzymes (e.g., glucose oxidase) as probe materials were excluded, as these substances do not align with our focus on retrievable chemical or material properties, such as those available through RDKit or the Materials Project database. This “semi-LLM process” exemplifies a collaborative synergy between AI-driven acceleration and expert oversight. While some may view the necessity of manual intervention as a limitation, it ensures the integrity of the dataset, bridging the gap between automated efficiency and scientific rigor. This phase significantly expedited data extraction compared to fully manual methods while maintaining accuracy for subsequent phases. The question prompt template and the final curated table, comprising 1,433 data entries extracted from 1,192 publications, are available in the “Phase_II” folder of the GitHub repository.

Data Transformation and Physical/Chemical Properties Integration - Phase III

Building on the curated table of 1,433 data entries from Phase II, we advanced into a systematic process of transforming and enriching the dataset. The primary goal of this phase was to prepare the raw sensing performance data for downstream analyses by converting experimental records into property-enriched numerical datasets. The process encompassed three key stages:

(a) JSON Conversion and Deduplication

The initial step involved converting the Phase II table into structured JavaScript Object Notation (JSON) files, maintaining the original chemical and material entity names as reported. During this step, special attention was given to handle duplicate entries across different publications. In cases where multiple DOI entries described identical sensing conditions (e.g., same probe material, medium, and detection target) but with differing lower detection limits (LDL), we retained only the entry representing the most advanced performance—the lowest LDL. This process ensured that the dataset reflected the state-of-the-art performance for each sensor configuration. The resulting JSON files were stored in the repository folder Phase_III/Original_Raw_JSON.

(b) Data Augmentation

Given the relatively limited dataset size screened from the previous step, we applied a carefully designed augmentation strategy to increase the dataset's diversity. This process was guided by assumptions about the insensitivity of certain performance metrics (LDL) to minor variations in test conditions. Specifically:

Granular LDL Categorization:

The transformation of continuous LDL values into five discrete performance categories was a critical step to simplify and standardize the dataset, enabling consistent comparisons and improving compatibility with machine learning algorithms. This categorization process focused on reducing the granularity of the data while maintaining meaningful distinctions between performance levels. LDL values were analyzed and reassigned to discrete categories based on logarithmic scales, reflecting their scientific significance and widespread use in sensing performance metrics.

For LDL, the continuous values were categorized into five discrete ranges: Category 5 for values ≥1 ppm, Category 4 for 0.1–1 ppm, Category 3 for 0.001–0.1 ppm, Category 2 for 0.00001–0.001 ppm, and Category 1 for values <0.00001 ppm, with each category representing increasing sensitivity. This categorization not only preserved the inherent differences in detection performance but also facilitated a more manageable and interpretable dataset for analysis. The resulting granularity effectively balanced dataset simplicity with the retention of critical performance distinctions.

Controlled Variations in Test Conditions:

For gas-phase sensors, slight temperature perturbations within ±10% were applied without altering the LDL category.

For liquid-phase sensors that do not involve pH measurements, variations in temperature and pH within ±0.5 were introduced.

For gas sensors, replacing the carrier medium (e.g., nitrogen with argon) was assumed to have negligible impact on detection performance.

For pH sensors, small temperature adjustments (e.g., ±10%) were also considered irrelevant to LDL categorization.

This augmentation strategy significantly expanded the dataset, generating over 10,000 unique entries. To ensure rigorous evaluation of model performance, while the training set from augmented data was used for training, the final model assessment was conducted both on test set from the augmented data (as shown in **Figure 5** red bars), but also the original, non-augmented data points within (as shown in **Figure 5** blue bars). All data entries were stored in the repository folder Phase_III/Original_Raw_JSON. The augmentation approach balanced realism with the need for statistical diversity in subsequent analyses.

(c) Property Retrieval and Dataset Enrichment

The final stage of this phase focused on enriching the JSON files with relevant physical and chemical properties for each material and chemical entity.

Inorganic Materials: Properties such as band gaps, formation energies, and crystallographic parameters and also MAGPIE fingerprint were retrieved from established materials science databases, including Materials Project, AFLOW, JARVIS, COD, and OQMD.

Small Molecules: Molecular properties such as fingerprints (Morgan), topological polar surface area, and quantum chemical descriptors were integrated from databases like PubChem.

Polymers: Molecular descriptors were computed using quaternary representations, capturing features like topological indices, molecular weight distributions, and other physicochemical characteristics relevant to sensing applications. Morgan fingerprints are also included.

Each sensing experiment was represented as a comprehensive JSON object, embedding both the original experimental parameters (e.g., operating temperature, pH, detection limit categories) and the enriched chemical properties. These finalized datasets were stored in two subfolders: Phase_III/Original_Properties_Retrieved_JSON: Enriched versions of the original dataset without augmentation. Phase_III/Data_Augmented_Properties_Retrieved_JSON: Property-enriched versions of the augmented dataset.

This phase culminated in a unique dataset where abstract chemical and physical concepts were transformed into numerical descriptors while preserving the original experimental context. The rigorous deduplication, augmentation, and enrichment processes ensured that the dataset was both comprehensive and representative of cutting-edge FET sensor performance. The combination of cheminformatics and materials informatics principles provided a robust foundation for machine learning analyses in subsequent phases, bridging the gap between experimental insights and predictive modeling.

Modeling - Phase IV

The modeling component of our work involved learning insights from the original and augmented datasets. Specifically, we were interested in classifying each sensing experiment – represented by a JSON object–into a discrete category ranging from 1 to 5. The classification procedure involves the following algorithms: Gradient Boosting Classifier, CatBoost, XGBoost, vanilla MLP, vanilla GNN, vanilla SNN, and a hybrid, multimodal spiking network integrated SGNN.

The first step was to represent data in a specific way for each algorithm. Each JSON object is a set of the following blocks: "detect_target", "probe_material", "testing_medium_electrolyte" – simply referred to as target (T), probe (P), and medium (M), respectively. There's also an extra block containing the test operating temperature, as well as minimum and maximum pH values– which was called the conditions (C) block.

In the sensing experiments we've analyzed, each one of the target, probe, and medium is made of one–and possibly more–of the following substance types: small molecule, inorganic solid, and polymer. Each substance type has its own specific parameters, which can be divided in global physicochemical properties (*e.g.,* molecular weight, volume, complexity, charge) and topological geometric structural descriptors (*e.g.,* Morgan fingerprint).

Given this JSON object description, we shall now discuss how this data structure has been represented for each classification algorithm. In total, we used three different representation paradigms: sequential, graph-based, and spike-based.

(a) Data Representation

(i) Type I: Sequential

The sequential naive representation was used for the gradient boosting classifier, CatBoost, XGBoost, and vanilla MLP. We essentially concatenated all blocks (T, P, M, C) together so that each JSON object was converted into a long simple numerical vector. We also performed data padding–adding zeros–to standardize and ensure that all samples had the same dimension before using them in our models. Corresponding schematic is shown in **Figure 3c.**

(ii) Type II: Graph-based

On the other hand, the graph-based paradigm consisted of creating a node for each block (T, P, M, C). In this context, a node was a set of the corresponding physicochemical and structural parameters. Once again, we employed data padding to ensure that all nodes had the same size. This representation was used for the vanilla GNN and partially for the hybrid, multimodal SGNN. After defining each node, the follow-up question was about the best way we could connect them. Even though a fully connected graph, *i.e.*, every node connected to the other, would be the most intuitive architecture *a priori*, initial classification experiments have revealed a sub-optimal performance.

Additionally, a deeper analysis of the problem allowed us to realize that the conditions node mostly affect the medium–since operating temperature and pH values refer to the electrolyte medium that has been used in the sensing experiment. Although there are chemical phenomena involving the target, probe material, and experimental conditions, we considered them to be negligible when compared to the direct link between conditions and medium in our graph modeling scheme. Therefore, we adopted an undirected graph representation in which T, P, and M are fully connected, while C connects only to M, as illustrated in **Figure 3d**. It's also worth noting that all edges have the same weight.

(iii) Type III: Spike-based

Finally, the spike-based representation was in the vanilla SNN and as a component of the hybrid SGNN model. The main idea was that instead of looking at the JSON object as a set of T, P, M, and C blocks, we could perceive it in a different way*, i.e.,* a set of physicochemical, global properties and a set of topological, structural, geometric descriptor.

The advantage of this approach is that both groups are fundamentally different with respect to the information they store. While the former is primarily dense (most global properties are not zero), the latter is predominantly sparse–Morgan fingerprint's geometric and connectivity description is a binary vector containing mostly zeros and only a handful of ones.

Due to this substantial difference in the way both sets of information are represented, it would be convenient to express them accordingly. Therefore, to be able to natively and easily represent the connections between all the relevant components, we shall keep a graph-based framework for the dense, physicochemical global properties. This graph data structure is used as input for a vanilla GNN as shown in the upper pipeline of **Figure 4c**.

Conversely, sparse, structural properties are better separately represented via spikes. Each sparse vector corresponding to the Morgan fingerprint is simply treated as a spike train, with mostly zeros and a handful of ones. We then used this spike representation as input for the vanilla SNN and part of the SGNN algorithm as shown in the bottom pipeline of **Figure 4c**. We chose the spike latency encoding to leverage the structural properties' sparsity as much as possible. Therefore, as this approach contains one spike per timestep (maximum), it would be a better choice than rate encoding, for instance[1].

Ultimately, we combined the output of both GNN and SNN via a fusion layer to arrive at the final SGNN model prediction—we concatenated both outputs into a vector (size 10) that acts as input for the fusion layer, producing a final output as a vector of size 5, *i.e.*, the number of LDL categories in this classification task.

(b) Model Overview

As a key innovation factor of this work, we shall discuss some SNN fundamentals. First of all, spikes are the natural way to store and process information in neuromorphic computers–non-von

Neumann (non-digital) computers inspired in the human brain. There are multiple differences between neuromorphic and digital architectures[2]. On top of using spikes instead of binary data, the former also differs from the latter in terms of operation (parallel *versus* sequential processing) and organization (processing and memory in the same place *versus* separated). An additional crucial difference refers to the processing timing: while digital computers are synchronous, clock-driven, neuromorphic computers are event-driven[2, 3]. For such reasons, neuromorphic computers could represent a more efficient computing platform–not only energy-wise, but also in terms of memory overhead[4].

The idea of modeling neurons and synapses from the human brain greatly evolved over the past century. Starting with the simplest Integrate-and-Fire (IF) neuron in 1907, scientists have proposed a number of improvements that led to models with higher biological fidelity (Hodgkin-Huxley, Morris-Lecar) and computational efficiency (Izhikevich, AdEx IF). In our work, we used the Leaky-Integrate-and-Fire (LIF) neuron–one of the most popular models due to its unique combination of simplicity and accuracy[1].

The LIF neuron assumes that a spiking neuron behaves like a low-pass filter circuit made of a resistor R and a capacitor C. This assumption–which has been biologically validated–can be written mathematically as:

$$\tau dU(t)/dt = -U(t) + RI(t),$$

where $U(t)$ is the neuron's membrane potential, $\tau = RC$ is the time constant of the circuit, and $I(t)$ is the current flowing through the circuit at a given time *t.*

If the current is constant, the solution to the differential equation above becomes:

$$U(t) = RI + (U_0 - RI)e^{-t/\tau},$$

where U_0 is the membrane potential at $t = 0$. Using the decay rate $\beta = e^{-1/\tau}$, we can write the $U(t)$ solution via the forward Euler method and discretizing time:

$$U[t] = \beta U[t-1] + (1-\beta)I[t].$$

In the context of deep learning, it is useful to write $I[t]$ as $WX[t]$, where W is a weight matrix and $X[t]$ is a vectorized input completely decoupled from the effect of β. This representation's advantage lies in the fact that we can now separate $U[t]$ into three terms: decay ($\beta U[t-1]$), input ($WX[t]$), and reset ($S[t-1]\theta$). Mathematically,

$$U[t] = \beta U[t-1] + WX[t] - S[t-1]\theta,$$

where $S[t] = 1$ if $U[t] > \theta$, and $S[t] = 0$ otherwise.

In practical terms, this means that the membrane potential increases whenever an input spike arrives, and it decays over time according to the β factor. If the potential increases enough to

reach the potential threshold θ, the neuron fires an output spike, and the membrane potential resets to zero. **Figure S2** represents this behavior for a set of input and output spikes over time.

In our model, the decay rate β and the hidden size–*i.e.,* number of neurons in the hidden layer– are customizable. Since both hyperparameters are crucial for the SNN performance, we fine tune them using BALLET: a Bayesian Optimization framework that adaptively filters the search space for a high confidence region of interest until it finds the hyperparameter values associated with the highest accuracy.

Even though neuromorphic has many advantages over von Neumann architecture, spiking neurons are inherently non-differentiable due to their event-driven behavior. For this reason, gradient-based optimization methods–*e.g.,* backpropagation–cannot be used directly to train SNNs. Among the many approaches to this problem, we shall focus on the surrogate gradient descent (SGD) technique[2, 4]

As previously discussed, the spiking neuron fires according to the step function σ that includes a comparison between the membrane potential $U[t]$ and the threshold θ, which does not affect the forward pass. Conversely, in the backward pass, we cannot directly differentiate the step function. To address this issue, we can approximate the step function derivative σ' in a couple of different ways, including the sigmoid function $\sigma'(x) = \sigma(x)(1 - \sigma(x))$ and the Gaussian approximation[5] $\sigma'(x) = \exp(-x^2/2)$. This approach retains the temporal dynamics of spiking neurons while allowing gradient-based learning.

In our problem, the training mechanism involved a cross-entropy loss function to compare targets to our model's outputs, while the backward pass applied a sigmoid SGD, enabling the SNN training. On that note, we chose cross entropy over mean squared error (MSE) due to a higher alignment with probabilistic outputs, larger gradients for wrong predictions[6], and an observed faster convergence and overall higher classification accuracy in preliminary numerical experiments.

(c) Training Environment

Once we have defined the data representation for each algorithm, we moved on to the training environment characteristics. Even though these algorithms are significantly different from each other, we can establish some basic, pre-defined parameters that shall be shared by all of them. For instance, for all algorithms, we used an 80:20 training-test dataset ratio. As mentioned previously, to ensure a fair setting and to verify that the augmented data is consistent with the original dataset, we only used 80% of the augmented samples to train the model. We then evaluated the model in the remaining 20% of the augmented dataset–represented as "Test Set (Augmented)"–and in the original dataset within, which is referred to as "Test set (Original)". This training and evaluation procedure was the same for all algorithms in this work.

Additionally, we have applied the BALLET optimization framework for determining the optimum set of hyperparameters for each algorithm. Specific values for each algorithm can be found in the repository folder Phase_IV. We carried out most of the ML model training experiments locally in a computer with an Apple M2 chip, 16 GB RAM, and 256 GB Macintosh

HD. For certain computationally intensive tasks, the scripts were separately run on professional workstation and the supercomputer clusters as mentioned in the acknowledgement section.

(d) Relevant Classification Metrics

There are many ways to measure the classification performance of our models. Some of the most common metrics include accuracy, F1 score, precision, and recall. Even though classification metrics related to false positives and negatives may still be relevant in multi-category problems, in our specific case, the model was supposed to correctly identify the exact category (1-5) of a given sensing experiment. Therefore, accuracy was deemed to be the most important metric in our analysis, *i.e.*, the number of correct classification instances divided by the total number of samples. To determine whether or not a sample was correctly classified, we simply took each label from the test dataset and compared it to the model's prediction for that same sample.

Best Model Explanation - Phase V

After running all these models, we selected the one with the highest accuracy to obtain additional insights on our dataset. As illustrated in the Results section, the hybrid, multimodal SGNN outperformed the other classification algorithms by a significant margin. Therefore, we used it to address two main questions: feature selection and perfluorooctane sulfonic acid (PFOS) detection probe material screening. The corresponding code for this phase can be found in the repository folder Phase_V.

The first one is about determining the most relevant features for obtaining a very low detection limit (high sensitivity) sensor. As we know, each block (T, M, P, C) has multiple physicochemical features, and each one of them can affect the final LDL classification. On that note, we ran integrated gradients and SHAP values experiments and obtained a list of the most recurrent features associated with the actual LDL determination. This is an important insight that can be used for guiding future research on FET sensor design–once we know which features are, on average, strongly correlated with the LDL, we can choose materials more effectively.

It is worth mentioning that SHAP values and integrated gradients are highly effective for feature selection in tabular data, which is not the representation we have in the GNN, SNN, and SGNN algorithms. In this case, we considered only the physicochemical properties within each node as the features to be selected. To further validate the feature selection results, we used some numerical techniques derived from random matrix theory (RMT) analysis[7]. After globally normalizing the features–ensuring zero-mean and unit-standard deviation–we used the Marcenko-Pastur law[8] to obtain the bulk distribution of eigenvalues of the corresponding covariance matrix. We then applied a sparse principal component analysis (PCA) considering the eigenvalues outside of the bulk spectrum, as they were the most relevant feature candidates[9]. The RMT analysis led us to the same set of most relevant global features for LDL multi-category classification.

On top of that, we also employed the SGNN model for screening the best possible probe materials for detecting PFOS. Given fixed conditions (T = 25 °C, pH max = pH min = 7) commonly found in our dataset, we consider all probe materials in our dataset as potential candidates for high sensitivity and selectivity toward PFOS. We then ran all possible combinations of those probe materials, PFOS as target, and fixed conditions using the best performing SGNN to determine which probe materials led to the lowest LDL values (high sensitivity).

Validation on Real Application - Phase VI

To simulate the binding interactions of various probe-target combinations under periodic boundary conditions, we conducted ab initio calculations using parameters optimized for precision and computational efficiency. The plane-wave energy cutoff was set to 450 eV, and spin-polarized DFT with the Perdew-Burke-Ernzerhof (PBE) functional was employed to describe exchange-correlation interactions. The Brillouin zone integration utilized a Gamma-centered 2 × 2 × 2 k-point grid and van der Waals corrections were included using the DFT-D3 method to capture dispersion effects. Ionic relaxations were performed with a convergence criterion of 0.02 eV/Å for the forces, and the electronic structure iterations were set to converge to an energy difference of $10e^{-4}$ eV.

For the modeling of small molecules such as β-CD, FcCOOH, and o-PD, the structures were placed in the center of a cubic vacuum box with appropriate dimensions to eliminate spurious interactions. Graphene was modeled as a supercell containing 128 carbon atoms with a vacuum thickness of 25 Å, while SWNT comprised 120 carbon atoms with chirality indices N=M=6, a repeat unit of 5, and the same vacuum thickness of 25 Å. For ZnO and Al_2O_3, the simulations utilized the most common (001) facet, modeled as supercells containing 192 and 120 atoms, respectively, with unsaturated metal (Zn or Al) surfaces exposed. A uniform vacuum thickness of 25 Å was applied to all periodic models to avoid artificial interactions between periodic images.

Binding energies (ΔE) were calculated using the formula:

$$\Delta E = |E_{probe\&target} - (E_{target} + E_{probe})|$$

where $E_{probe\&target}$ is the total energy of the coupled system, E_{probe} and E_{target} are the energies of the isolated probe and target, respectively. This approach enabled the quantification of binding interactions and selectivity for various probe-target pairs under realistic simulation conditions. The larger the value is, the more energy favorable the binding would be.

For quantum chemistry simulations, binding interactions between probes and targets were evaluated under three different environmental scenarios: vacuum, implicit solvent, and explicit solvent. Initially, the structures of all configurations, including the probe, target, and coupled systems, were optimized using a hybrid density functional theory approach with the B3LYP functional and the 6-31G(d) basis set. This step included vibrational frequency calculations to

confirm the stability of the optimized geometries and ensure that all structures corresponded to true minima on the potential energy surface.

Following structural optimization, single-point energy calculations were performed at a notably higher level of theory using the B3LYP functional and the def2-TZVP basis set. Tight SCF convergence criteria: extended quadratic convergence was employed to achieve robust and accurate electronic energy values. This higher-level calculation ensures better precision in describing the electronic interactions within the system, which is critical for evaluating binding energies.

For simulations incorporating implicit solvent effects, the solvent environment was modeled using the Self-Consistent Reaction Field (SCRF) method with water as the dielectric medium. This approach captures the influence of solvation on binding interactions without explicitly adding water molecules. In the explicit solvent scenario, a cluster of 12 water molecules was introduced around the system to mimic the solvation shell. Pre-optimization of this cluster was performed at a lower level of theory using the long-range corrected hybrid functional wb97xd/3-21G with relaxed convergence criteria to obtain an initial configuration. Subsequently, the system underwent re-optimization at the b3lyp/6-31G(d) level with tighter convergence settings to refine the geometry further. Binding energies (ΔE) in all scenarios were computed using the single-point energy differences.

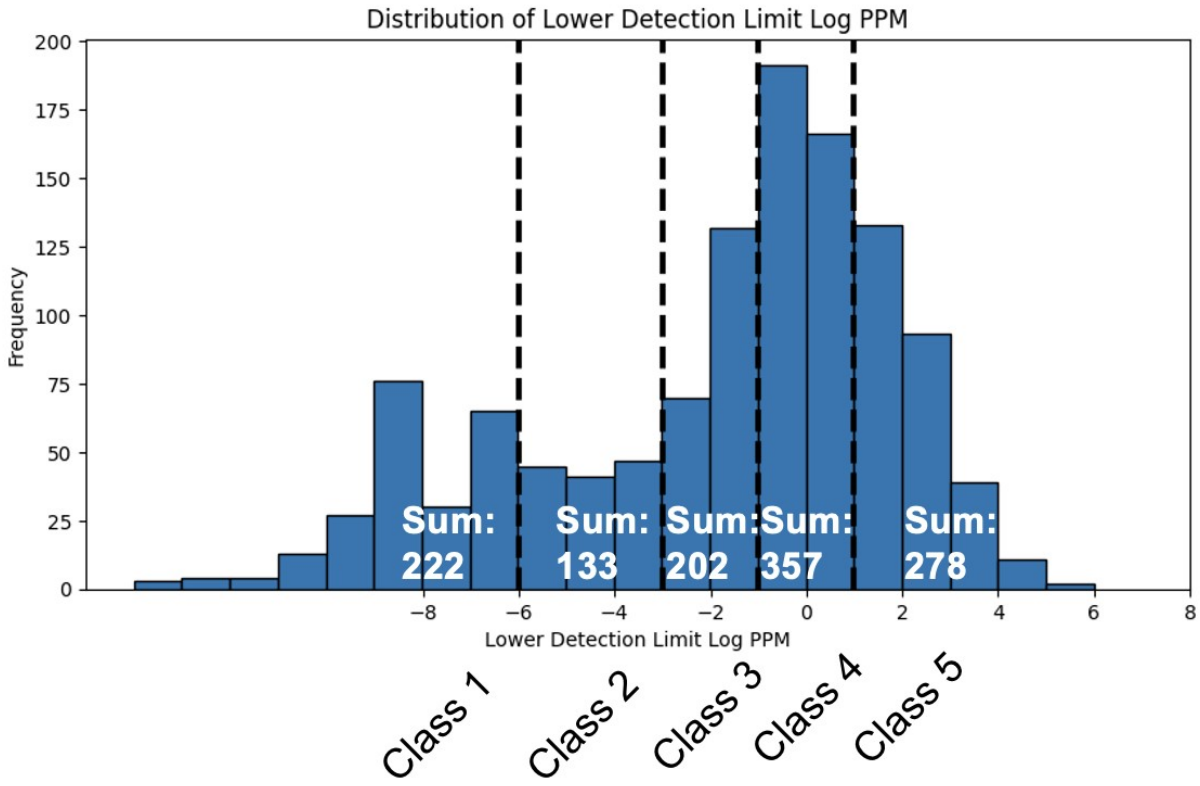

Figure S1 Distribution of LDL and also threshold of the five classes as ML modeling output.

Log (LDL[ppm])	Category	Number of original samples
$\log(LDL) <- 6$	1 (Very High Sensitivity)	222
$-6 \leq \log(LDL) <- 3$	2 (High Sensitivity)	133
$-3 \leq \log(LDL) <- 1$	3 (Medium Sensitivity)	202
$-1 \leq \log(LDL) < 1$	4 (Low Sensitivity)	357
$\log(LDL) > 1$	5 (Very Low Sensitivity)	278

Table S1 Category description in terms of the LDL (in ppm) and the total amount of original data samples per category.

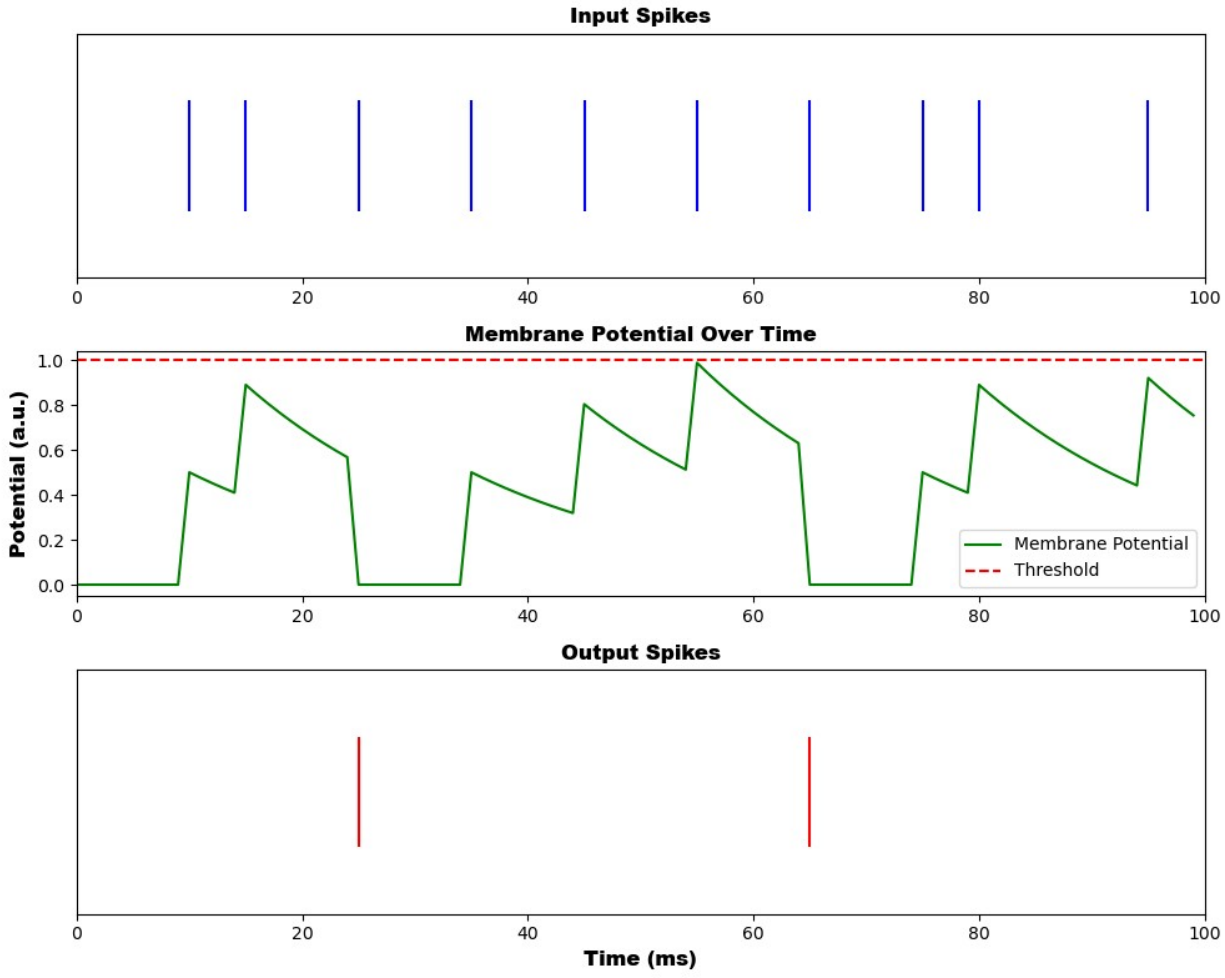

Figure S2 Simulation of a LIF neuron dynamics with membrane potential, input and output spikes over time.

PFAS Name	Probe 1	Probe 2	Probe 3	Probe 4	Probe 5
Perfluorooctanoic acid	graphene	zinc oxide	carbon nanotube	aluminum oxide	gold
Perfluorohexanesulfonamide	graphene	aluminum oxide	zinc oxide	gold	carbon nanotube
Perfluoroundecanoic acid	graphene	aluminum oxide	zinc oxide	tin dioxide	phenol
Perfluoropropanesulfonic acid	graphene	zinc oxide	aluminum oxide	carbon nanotube	tin dioxide
Perfluorooctadecanoic acid	graphene	zinc oxide	hafnium oxide	phenol	ethylene oxide
Perfluorooctanesulfonic acid	graphene	zinc oxide	aluminum oxide	carbon nanotube	tin dioxide
Perfluorohexanesulfonic acid	graphene	aluminum oxide	zinc oxide	carbon nanotube	silicon dioxide

Table S2 Probe substances predicted by the SGNN model for different PFAS target analytes. In this case, we performed the probe material screening using the fully-converged SGNN model after training and testing with the augmented data.

PFAS Name	Probe 1	Probe 2	Probe 3	Probe 4	Probe 5
Perfluorooctanoic acid	graphene	silicon dioxide	gold	zinc oxide	tin dioxide
Perfluorohexanesulfonamide	graphene	zinc oxide	silicon dioxide	carbon nanotube	aluminum oxide
Perfluoroundecanoic acid	graphene	gold	phenol	silicon dioxide	carbon nanotube
Perfluoropropanesulfonic acid	graphene	zinc oxide	carbon nanotube	aluminum oxide	tin dioxide
Perfluorooctadecanoic acid	graphene	gold	tin dioxide	titanium dioxide	ethylene oxide
Perfluorooctanesulfonic acid	graphene	zinc oxide	aluminum oxide	indium oxide	tin dioxide
Perfluorohexanesulfonic acid	graphene	zinc oxide	silicon dioxide	carbon nanotube	aluminum oxide

Table S3 Probe substances predicted by the SGNN model for different PFAS target analytes. In this case, we performed the probe material screening using the fully-converged SGNN model after training and testing with the original data, only.

Note: The consistency of the probe predictions across all PFAS substances—comparing Table S2 and S3—suggests that our data augmentation approach not only increases the dataset size, but also introduces realistic variations that mirror intrinsic uncertainties in real-world sensor measurements. This analysis indicates that the SGNN's probe screening prediction would still be effective if we had used only the original dataset.

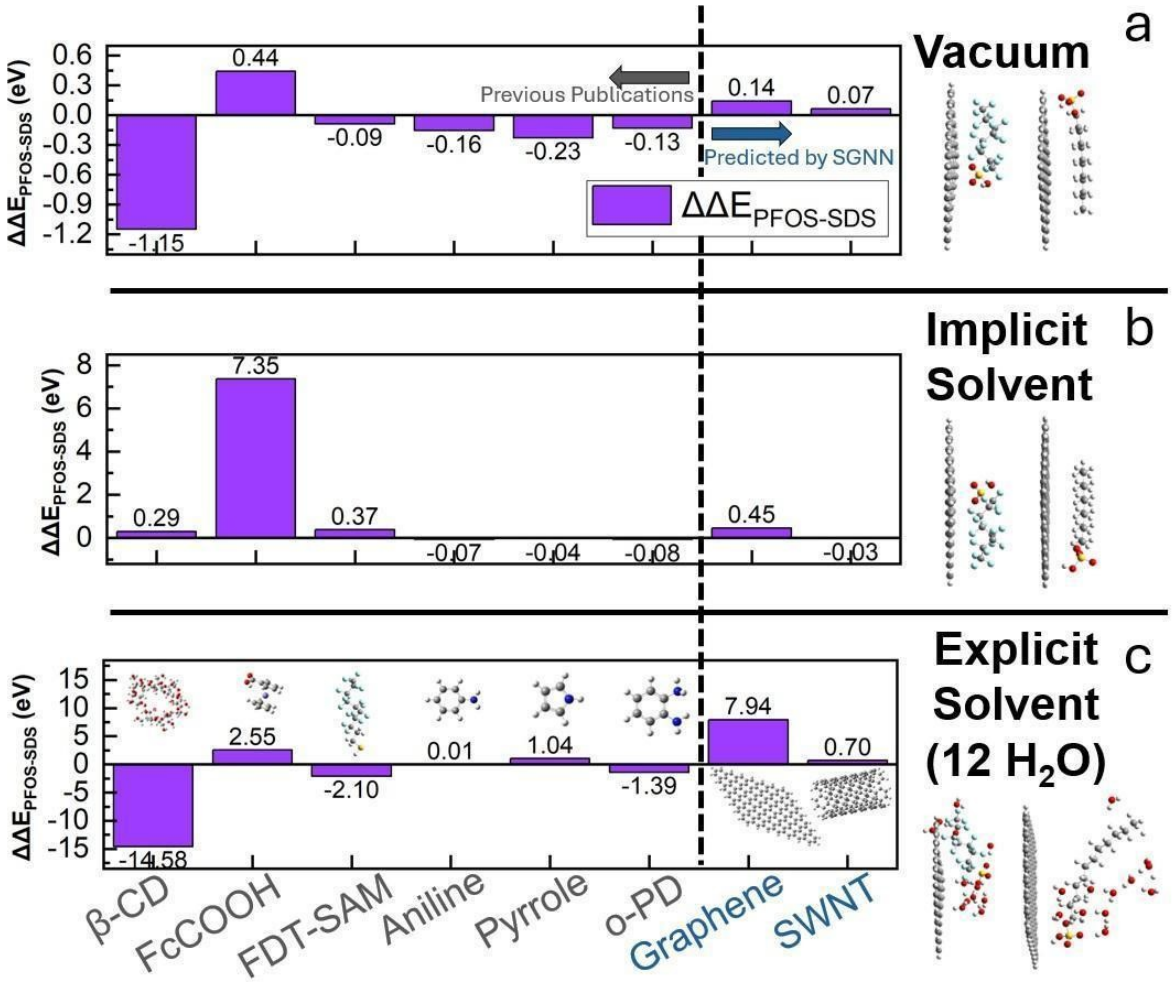

Figure S3 Histogram summary plots indicating the values of $\Delta\Delta E_{PFOS\text{-}SDS}$ in (a) vacuum, (b) implicit solvent field of water, and (c) explicit solvent water cluster.

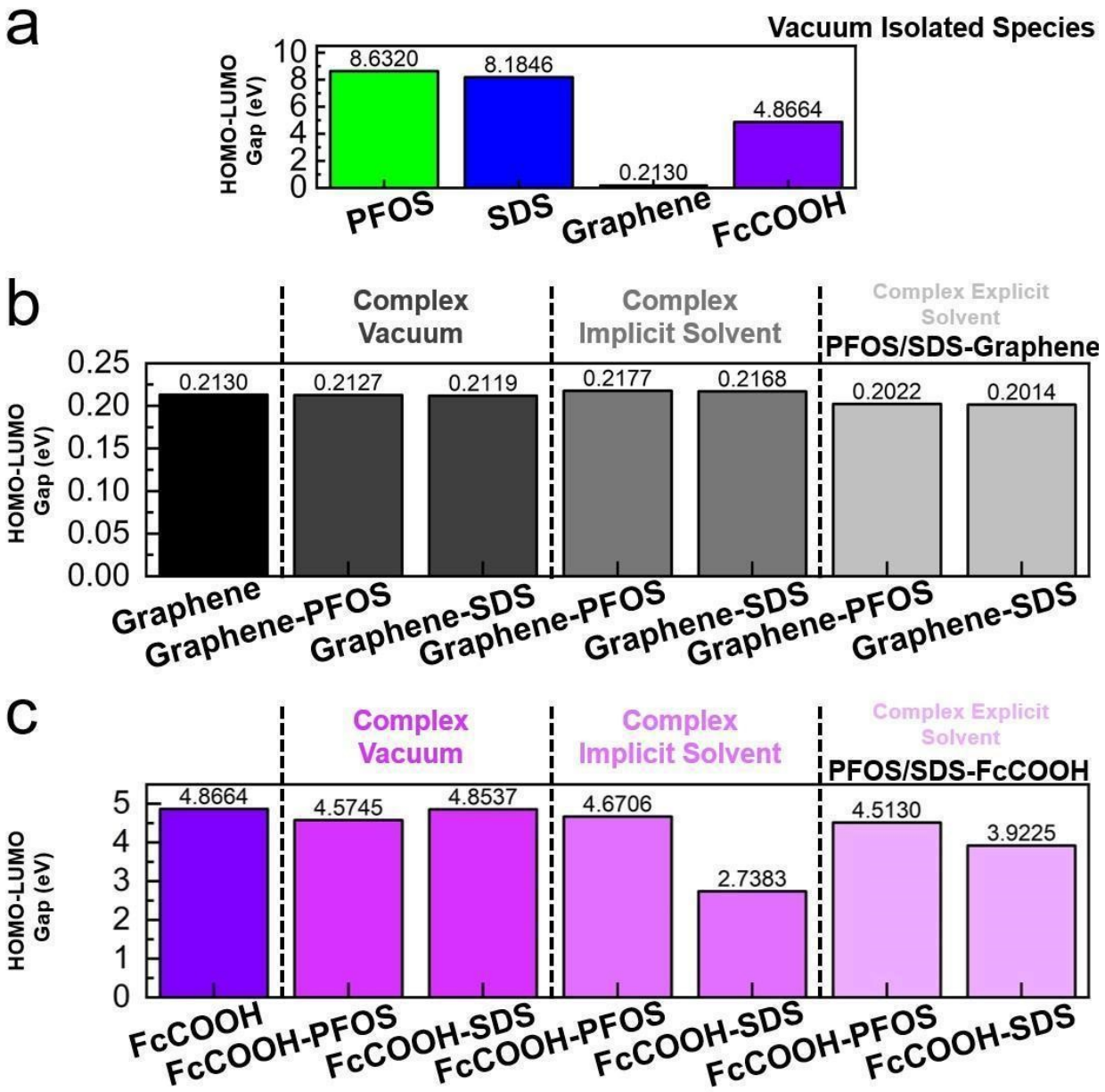

Figure S4 (a) HOMO-LUMO gap values calculated for different isolated molecule species in vacuum. (b) HOMO-LUMO gap values calculated for different scenarios of graphene before and after binding with PFOS and SDS. (c) similar to (b), gap values for FcCOOH's binding behavior.

Note:

In our study, the isolated standard graphene moiety exhibits a HOMO-LUMO gap of 0.2130 eV, highlighting its highly conductive and metal-like nature. In contrast, FcCOOH has a significantly larger gap of 4.8664 eV. Upon binding with PFOS or SDS, FcCOOH shows comparatively larger changes in the HOMO-LUMO gap, whereas graphene demonstrates only slight variations. This observation aligns with our conclusion in the main text that FcCOOH undergoes stronger electronic coupling and orbital interactions with PFOS, as evidenced by the drastically changed

gap values. Conversely, graphene primarily interacts through weak π-π stacking or physisorption, leading to minimal alterations in the HOMO-LUMO gap. Despite it is still secondary compared to FcCOOH, graphene's meaningful binding interactions still justify itself as a potential probe material. Combining these probes, such as grafting FcCOOH on a graphene channel, could exploit their unique strengths for enhanced sensing performance in practical applications.

Further, under both vacuum and explicit solvation (12 water molecules), the HOMO-LUMO gap reduces upon binding either PFOS or SDS to graphene or FcCOOH. However, under implicit solvation, a slight increase in the HOMO-LUMO gap is observed for graphene. This discrepancy arises because implicit solvent models approximate the solvent as a uniform dielectric medium, which can fail to account for specific solute-solvent interactions or solvent-induced polarization effects[10]. These limitations can result in an inaccurate depiction of how solvation affects the electronic structure, particularly for systems like graphene, where subtle electronic rearrangements play a key role. Hence, while the inclusion of explicit solvation with 12 water molecules is computationally more expensive, it provides a higher accuracy and a more realistic representation of the interaction environment, making it critical for reliably interpreting binding energetics and electronic behavior.

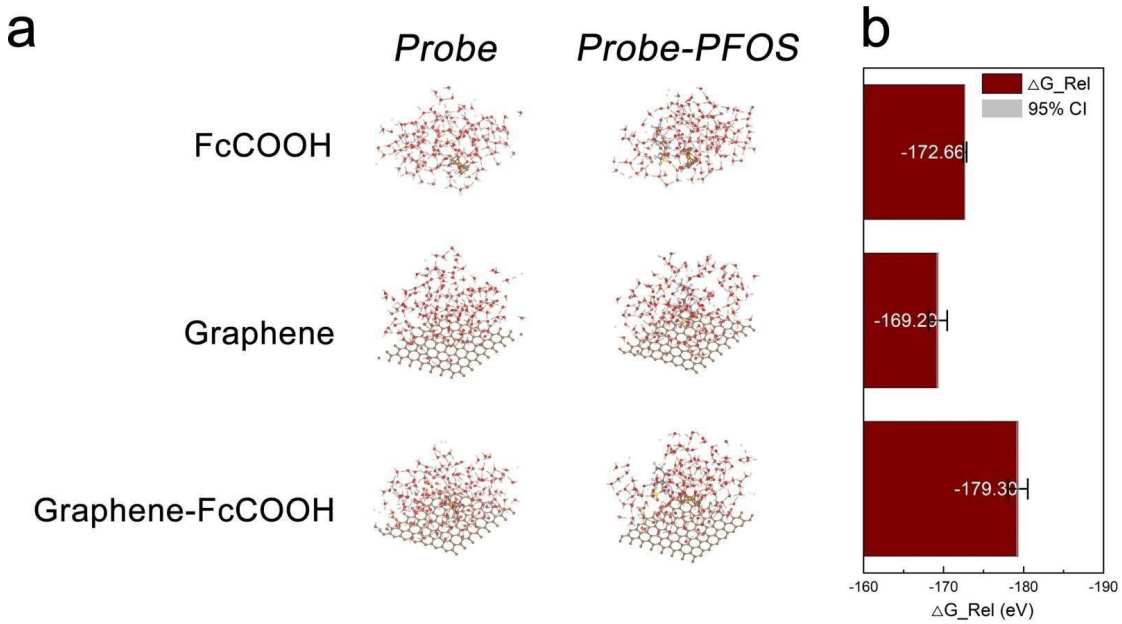

Figure S5 (a) Snapshots of the different investigated probe and probe-PFOS systems with explicit water molecules. (b) corresponding ΔG_rel statistical values.

To test our hypothesis that graphene and FcCOOH may synergistically enhance PFOS binding, we performed *ab initio* molecular dynamics (AIMD) simulations on pre-optimized configurations of three probe systems: FcCOOH, graphene, and graphene–FcCOOH. For each probe system, both the isolated probe and its corresponding probe–PFOS complex were embedded in a simulation cell containing 168 explicit water molecules (with the water density maintained based on the cell volume) and a 15 Å vacuum layer along the z direction to minimize spurious interactions. The AIMD simulations were conducted using the same electronic and ionic settings as our previous static calculations, with the only modification being that the systems dynamically evolved in the NVT ensemble. An initial pre-optimization of 1000 AIMD steps (1 ps) was performed, followed by a production run of an additional 1000 AIMD steps for production. A sliding-window procedure was then employed to identify a representative equilibrium window of 100 consecutive frames, defined by an average absolute energy change between successive frames below a threshold of (total number of atoms × 0.5 meV). In this study, we report the free energy (G) because G—incorporating the electronic entropy via Fermi–Dirac smearing—offers a more thermodynamically realistic description under constant temperature and volume conditions. The relative binding free energy was defined as

$$\Delta G_{rel} = G(\text{probe–PFOS}) - G(\text{probe}).$$

Because the additional energy term corresponding to an isolated PFOS molecule is invariant across the different probe systems and is prohibitively expensive to obtain via AIMD, it was omitted from the relative analysis. Statistical analysis of the 100-frame equilibrium window yielded the mean, variance, and 95% confidence interval for ΔG_rel for each system. This approach enables a

quantitative ranking of the PFOS binding propensities of the different probes while significantly reducing computational cost compared to a full thermodynamic cycle evaluation.

As shown in **Figure S5**, our hypothesis is well-validated by the results in these realistic simulation environments. FcCOOH alone, as expected, could show a relatively 3.37 eV advantage compared to pure graphene. Impressively, when FcCOOH is mounted on the graphene sheet, a 6.64 eV free energy advantage is observed. This phenomenon has well qualitatively supported our hypothesis that in a practical 2D FET graphene sensor where graphene is used as channel material, grafting FcCOOH might increase the performance.

References in Supporting Information

1. J. K. Eshraghian, M. Ward, E. O. Neftci, X. Wang, G. Lenz, G. Dwivedi, M. Bennamoun, D. S. Jeong and W. D. Lu, *Proceedings of the IEEE %@ 0018-9219*, 2023.
2. C. D. Schuman, S. R. Kulkarni, M. Parsa, J. P. Mitchell, P. Date and B. Kay, *Nature Computational Science*, 2022, **2**, 10-19.
3. S. R. Kulkarni, M. Parsa, J. P. Mitchell and C. D. Schuman, *Neurocomputing*, 2021, **447**, 145-160.
4. J. Li, H. Zhang, R. Wu, Z. Zhu, B. Wang, C. Meng, Z. Zheng and L. Chen, *arXiv preprint arXiv:2305.19306*, 2023.
5. E. O. Neftci, H. Mostafa and F. Zenke, *IEEE Signal Processing Magazine*, 2019, **36**, 51-63.
6. A. Mao, M. Mohri and Y. Zhong, 2023.
7. G. Livan, M. Novaes and P. Vivo, *Monograph Award*, 2018, **63**, 54-57.
8. H. Liu, A. Aue and D. Paul, 2015, **675-712**.
9. D. Paul and A. Aue, *Journal of Statistical Planning and Inference*, 2014, **150**, 1-29.
10. J. Dziedzic, H. H. Helal, C. K. Skylaris, A. A. Mostofi and M. C. Payne, *Europhysics Letters*, 2011, **95**, 43001.