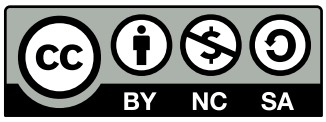
Aalto University
School of Science

Bachelor's programme in Engineering Physics and Mathematics

Molecular descriptor engineering for machine learning predictions in atmospheric science

Linus Lind

Bachelor's Thesis
2024



Aalto University
School of Science

Author Linus Lind

Title Molecular descriptor engineering for machine learning predictions in atmospheric science

Degree programme Engineering Physics and Mathematics

Major Engineering Physics **Code of major** SCI3028.A

Teacher in charge Prof. Patrick Rinke

Advisor Dr. Hilda Sandström

Date 16 May 2024 **Number of pages** 25+9 **Language** English

Abstract

Atmospheric organic compounds form complex mixtures, where new compounds may form via chemical reactions, increasing the number of unique molecular species and their complexity. These compounds may exist in gas phase and in particulate matter phase, suspended as aerosols. Organic aerosols are a particular focus of research in atmospheric science, since they affect radiative forcing, air quality, and other chemical and physical processes such as cloud formation in the atmosphere. Key physico-chemical properties of a molecule that determine aerosol formation are saturation vapour pressure and equilibrium partition coefficients, but unfortunately these are difficult to measure experimentally for atmospheric compounds. Thus, the use of machine learning methods for predicting these properties are on the rise, increasing the demand for new ways to store molecular data in a machine-readable format, in the form of a molecular descriptor. In this thesis, a new binary encoded molecular descriptor was developed for machine learning purposes: the BESPE-MACCS descriptor. It is based on the Molecular Access System (MACCS) descriptor and a molecular substructure enumeration method called binary encoded SMARTS pattern enumeration (BESPE). The aim of this thesis was to improve the applicability of the MACCS descriptor for predicting properties of atmospherically relevant organic compounds. The performance of the newly developed BESPE-MACCS descriptor was evaluated using a kernel ridge regression machine learning model, which was used in a previous study that compared prediction accuracies obtained by the model with different descriptors. In prediction accuracy, measured by mean absolute error, the BESPE-MACCS descriptor had similar or better performance compared to descriptors used in previous research within statistical significance, while having a much smaller file size and computational expense. Additionally, the BESPE-MACCS descriptor is human-interpretable, customisable, and simple, thus easy to develop further, which makes it a promising tool for studying organic aerosols in atmospheric science.

Keywords Saturation vapour pressure, Partition coefficient, Supervised machine learning, Kernel ridge regression, Organic aerosol, Molecular descriptor

Aalto-yliopisto
Perustieteiden
korkeakoulu

Tekijä Linus Lind

Työn nimi Molecular descriptor engineering for machine learning predictions in atmospheric science

Koulutusohjelma Teknillinen fysiikka ja matematiikka

Pääaine Teknillinen fysiikka **Pääaineen koodi** SCI3028.A

Vastuuopettaja Prof. Patrick Rinke

Työn ohjaaja PhD Hilda Sandström

Päivämäärä 16.5.2024 **Sivumäärä** 25+9 **Kieli** englanti

Tiivistelmä

Ilmakehässä esiintyvät orgaaniset aineet muodostavat moninaisia seoksia, jotka koostuvat useista eri yhdisteistä. Nämä aineet esiintyvät sekä kaasuina että pienhiukkasina aerosoleina. Aerosolit ovat vuorovaikutuksessa toistensa ja ilmakehän kaasujen kanssa, osallistuen erilaisiin kemiallisiin reaktioihin, joiden seurauksena muodostuu monimutkaisempia molekyylejä. Orgaanisten aerosolien tutkimus on yksi tärkeä ilmakehätieteen osa-alue, koska ne vaikuttavat ilmanlaatuun, auringon säteilyn sirontaan ja ilmakehässä tapahtuviin prosesseihin, kuten pilvien muodostumiseen.

Orgaanisten yhdisteiden fysiokemialliset ominaisuudet, kuten niiden höyryn kyllästymispaine ja jakaantumiskertoimet, määrittävät aerosolin muodostumisen ja sen käyttäytymisen ilmakehässä. Valitettavasti näiden ominaisuuksien mittaaminen kokeellisesti on hankalaa, koska ilmakehässä esiintyviä erilaisia orgaanisia aineita on paljon. Tämän vuoksi näiden ominaisuuksien ennustaminen koneoppimisella on lisääntynyt ilmakehätieteen tutkimuksessa, ja siksi tarve esittää molekyylien rakennetta tietokoneelle luettavassa muodossa eri tavoin on lisääntynyt. Tätä esitysmuotoa kutsutaan laajemmin keminformatiikassa deskriptoriksi.

Tässä kandidaatintyössä luotiin koneoppimista varten uusi binäärimuotoinen BESPE-MACCS-deskriptori, joka perustuu MACCS-deskriptoriin (engl. molecular access system) ja molekyylin alirakenteiden laskemismenetelmään (BESPE, engl. binary encoded SMARTS pattern enumeration). Työn tavoitteena oli parantaa MACCS-deskriptorin soveltuvuutta ilmakehässä esiintyvien orgaanisten yhdisteiden ominaisuuksien ennustamiseen. Työssä testattiin uuden deskriptorin soveltuvuutta koneoppimiseen kahdella eri tietojoukolla sovittamalla ne aikaisemmassa tutkimuksessa käytettyyn ydinfunktioregressiomalliin. Tätä kandidaatintyötä varten luodulla deskriptorilla saatiin tilastollisesti merkittävästi yhtä tarkkoja tai tarkempia ennustuksia muihin deskriptoreihin verrattuna, kun tarkkuutta mitataan absoluuttisten virheiden keskiarvolla. Uusi BESPE-MACCS-deskriptori on yksinkertaisempi, muokattavampi ja tiedostokooltaan huomattavasti pienempi muihin deskriptoreihin verrattuna. Siksi uusi deskriptori on lupaava työkalu orgaanisten aerosolien tutkimukselle ilmakehätieteessä.

Avainsanat höyryn kyllästymispaine, jakaantumiskerroin, ohjattu koneoppiminen, ydinfunktioregressio, orgaaninen aerosoli, deskriptori

Preface

I want to thank my supervisor Professor Patrick Rinke and my instructor Dr. Hilda Sandström for their help and guidance. I also wish to thank the Aalto Scientific Computing team for providing the computational resources for this thesis.

Otaniemi, 2 May 2024

Linus Lind

Contents

Symbols and abbreviations

Symbols

α	Regression weight coefficient
C	Saturation mass concentration (g m^{-3})
C^*	Effective saturation mass concentration (g m^{-3})
γ	Kernel width parameter
$\boldsymbol{I}$	Identity matrix
k	Kernel function
$\boldsymbol{K}$	Kernel matrix
$K_{\mathrm{WIOM/G}}$	Partition coefficient between water-insoluble organic matter and gas phases
$K_{\mathrm{W/G}}$	Partition coefficient between water and gas phases
$\boldsymbol{L}$	Loss function
λ	Ridge regression penalisation weight factor
M	Molar mass (g mol^{-1})
n	Amount of substance (mol) or number of data instances
P_{sat}	Saturation vapour pressure (kPa)
ϕ	Feature map
R	Gas constant = 8.314462618 (J K^{-1} mol^{-1})
T	Temperature (K)
V	Volume (m^3)
$\boldsymbol{x}$, $\boldsymbol{y}$	Feature vector & target vector
$\mathcal{X}$, $\mathcal{Y}$	Feature space & target space

Abbreviations

BESPE	Binary encoded SMARTS pattern enumeration
COSMO-RS	Conductor-like screening model for real solvents
ELVOC	Extremely low volatile organic compound
IVOC	Intermediate volatile organic compound
KRR	Kernel ridge regression
LVOC	Low volatile organic compound
MACCS	Molecular access system
MAE	Mean absolute error
MBTR	Many-body tensor representation
MSE	Mean squared error
OA	Organic aerosol(s)
SMARTS	SMILES arbitrary target specification
SMILES	Simplified molecular-input line-entry system
SOA	Secondary organic aerosol(s)
SVOC	Semi-volatile organic compound
TopFP	Topological fingerprint
VOC	Volatile organic compound

1 Introduction

Aerosols are suspensions of solid particles or liquid droplets in gas. In atmospheric science, aerosols are of particular interest, since they affect air quality and contribute to key atmospheric processes such as radiative forcing, formation of clouds or other clusters of particles and chemical reactions (Jimenez et al., 2009). Aerosols are an essential part of atmospheric physical chemistry, where properties of atmospherically relevant molecules are studied, so that atmospheric processes are better understood at a fundamental level.

While there may be many types of compounds suspended as aerosols in Earth's atmosphere, a specific topic of interest in atmospheric chemistry is organic aerosols (OA). In the atmosphere, OA may originate from biogenic sources, i.e., naturally occurring processes, or anthropogenic sources, such as combustion of fossil fuels, burning of biomass, and industrial processes (Jimenez et al., 2009). Organic compounds in air undergo various oxidation reactions in the atmosphere, from which secondary organic aerosols (SOA) may form as oxidation products (Srivastava et al., 2022).

Gas-particle partitioning is a key phenomenon in explaining SOA formation and behaviour (Zuend & Seinfeld, 2012). This partitioning in the atmosphere is mostly determined by the volatility (measured in e.g., saturation vapour pressure) and partition coefficients of the compound (Bilde et al., 2015; Lumiaro et al., 2021; Seinfeld & Pandis, 2016). However, experimentally measuring these properties proves to be a challenging problem, since organic compounds tend to chemically react forming more species, hence the number of species in the atmosphere is very large, and often a specific compound has a concentration below the detection limit of measurement instruments (Besel et al., 2023). Thus, synthesis of these compounds is often a prerequisite for performing experimental measurements. However, synthesising atmospheric compounds at the required scale is currently out of reach due to the sheer number of compounds and their complexity (Nozière et al., 2015).

To circumvent these practical issues, various computational methods have been developed with the aim of generating potentially significant atmospheric organic compounds and calculating their properties. However, accurately modelling important phenomena in the atmosphere, such as gas-particle partitioning, is computationally expensive, often requiring nonlocal computing such as computer clusters or supercomputers. Naturally, this brings a trade-off between computational expense and accuracy for different computational methods, which motivates the discovery of more efficient computational techniques.

Applying machine learning to predict molecular properties has recently gained popularity in atmospheric research (Besel et al., 2023; Dueben et al., 2022; Lumiaro et al., 2021), since machine learning is computationally lighter compared to state-of-the-art quantum chemistry simulations, while providing sufficient accuracy for most applications. However, machine learning requires feature engineering, i.e., in this

case, converting structural data of a molecule to machine-readable data via molecular descriptors. These descriptors encode information about molecules to numbers, which can then be used as the input data. A good descriptor encapsulates important information about many types of compounds, thus enabling accurate predictions with machine learning. However, for research applications, a descriptor should also be customisable and human-interpretable for adaptability and explainability.

In Lumiaro et al. (2021), several molecular descriptors were used in training and testing a Kernel Ridge Regression (KRR) machine learning model for predicting saturation vapour pressures, P_{sat}, and two equilibrium partition coefficients, $K_{\text{W/G}}$ and $K_{\text{WIOM/G}}$, on a data set sourced from Wang et al. (2017). To summarise, they found that the lowest prediction mean absolute errors (MAEs) were produced by the many-body tensor representation (MBTR), closely followed by the topological fingerprint (TopFP). Moreover, compared to the two best performing descriptors, the Molecular ACCess System (MACCS) descriptor produced prediction MAEs that were $18 - 36$ % larger across target variables P_{sat}, $K_{\text{W/G}}$, and $K_{\text{WIOM/G}}$.

Furthermore, these descriptors have their own set of advantages and disadvantages. MBTR is interpretable with visualisation (Lumiaro et al., 2021), but is quite complicated and uses a large amount of memory. TopFP is not interpretable at all, since it uses binary encoding and hashing for storing the topology of a compound (Lumiaro et al., 2021), but uses moderate amounts of memory. MACCS is generated by storing answers to 166 predefined boolean questions (e.g. "Is there a silicon atom?") about the molecule as a 166-bit binary vector (Lumiaro et al., 2021). Thus, MACCS is human-interpretable, memory-efficient and easy to engineer further, albeit having worse performance than MBTR and TopFP prediction-wise (Lumiaro et al., 2021). The memory efficiency of these descriptors is presented in more detail in section 3.4.

Still, none of these descriptors have all the desirable properties for molecular property prediction with machine learning, especially for atmospheric organic compounds. Furthermore, research on these descriptors is still lacking, especially in the scope of machine learning predictions in atmospheric science. To tackle these issues, the aim of this thesis is to engineer a new molecular descriptor that provides accurate machine learning predictions for gas-particle partitioning properties, while being customisable and human-interpretable.

Lumiaro et al. (2021) establishes the foundation for the goals of this thesis, and to maintain comparability of results, most of the methodology is conducted in a similar fashion. More specifically, the aim of this thesis is to: (1) Develop a new interpretable descriptor - a modified MACCS descriptor tailored towards atmospheric organic compounds. (2) Benchmark the descriptor with applying the KRR model used in Lumiaro et al. (2021) to perform the predictions. (3) Improve and optimise the descriptor to obtain predictions with a MAE that is similar to the TopFP and MBTR descriptors. (4) Test the developed descriptor on a different data set, 'GeckoQ', from Besel et al. (2023) and refine the descriptor to enhance its generalisability.

2 Literature review

2.1 Secondary organic aerosol formation

Secondary organic aerosols (SOA) form from atmospheric organic compounds, which originate from oxidation of primary emissions that are sufficiently volatile (Srivastava et al., 2022). Further oxidation may occur in gas-phase or particulate phase forming lower volatility compounds (Donahue et al., 2012). The volatility of the precursor organic compound largely determines if it will form a SOA. (Donahue et al. 2011; Bianchi et al. 2019). Volatility describes the tendency for a substance to vaporise; a high volatility implies more vaporisation, and low volatility implies more condensation. Volatility is commonly measured in effective saturation mass concentration C^* or saturation vapour pressure P_{sat}, which are proportional by a factor that depends the temperature, molar mass and activity coefficient of the compound; this is later shown in section 3.1 in more detail. In atmospheric chemistry, organic compounds are often categorised into volatility classes. For example, Donahue et al. (2012) present a classification as shown in Table 1.

Table 1: Volatility classes (Donahue et al., 2012)

Volatility class	**Volatility** C^* (μg m^{-3})
Extremely low volatility organic compound (ELVOC)	$C^* < 3 \times 10^{-4}$
Low volatility organic compound (LVOC)	$3 \times 10^{-4} < C^* < 0.3$
Semi-volatile organic compound (SVOC)	$0.3 < C^* < 300$
Intermediate volatility organic compound (IVOC)	$300 < C^* < 3 \times 10^6$
Volatile organic compound (VOC)	$C^* > 3 \times 10^6$

Furthermore, volatility classes dictate the key processes of atmospheric organic compounds, namely condensation into clusters or vaporisation. Bianchi et al. (2019) summarise the behaviour of molecules in volatility classes as defined in Donahue et al. (2012): ELVOCs are essentially, without exception, in the condensed phase, thus mostly participate to new-particle formation with other ELVOCs; LVOCs are likely to condense onto other particulate matter, except for the smallest particles due to the Kelvin effect; SVOCs may exist in both the gas and the condensed phase at significant proportions; IVOCs exist almost exclusively in the gas phase in the atmosphere despite having a relatively low volatility; VOCs are volatile under all circumstances and contribute most to atmospheric gas-phase oxidation reactions.

SOA formation, in the atmosphere, is difficult to model analytically in detail. On a per molecule basis, a rigorous and fundamental approach essentially requires applying many-body quantum field theory, which is unfeasible for larger molecules as of now. Moreover, simulating SOA formation at this level of detail is computationally very expensive. This has spurred the research of methods that simplify this problem.

2.2 Saturation vapour pressure

Assessing the phases in which a compound can exist in the atmosphere is a key component in understanding the formation of aerosols (Bilde et al., 2015). As mentioned in section 2.1, for research of OA, the volatility of the organic compound is pivotal. A common measure of this is the saturation vapour pressure, P_{sat}, which is a pure compound property not affected by the surrounding solvent or the presence of other substances (Seinfeld & Pandis, 2016, p. 582; Lumiaro et al., 2021).

A concise definition of P_{sat} is "The pressure exerted by a pure substance (at a given temperature) in a system containing only the vapour and condensed phase (liquid or solid) of the substance." (IUPAC, 2019). The behaviour of a compound at P_{sat} at temperature T is determined by thermodynamics. At P_{sat}, the gas phase of the compound is at a dynamic equilibrium with the condensed phase; at the microscopic level, the molecules are constantly condensing and vaporising, but the rate of condensation and vaporisation within the system is equal (North, Erukhimova, 2009, p. 98-100). Furthermore, isothermally exerting work to shift this dynamic equilibrium changes the proportions of the gas and condensed phases of the compound.

In a qualitative sense, when the P_{sat} of a compound is high, it is very likely to vaporise into the gas phase and as a result the compound has a higher volatility. As such, the P_{sat} is the single most important property for determining gas-particle or gas-liquid droplet partitioning of a compound in the atmosphere (Seinfeld & Pandis, 2016, p. 582). However, the P_{sat} of a compound only describes its interaction with itself, not between other compounds.

2.3 Equilibrium partition coefficients

Equilibrium partition coefficients are a part of determining SOA behaviour and formation (Bilde et al., 2015). Together with P_{sat}, gas-particle partitioning is determined more accurately with equilibrium partition coefficients. The presence of other compounds affects the gas-particle partitioning of the compound depending on their chemical composition and properties (Seinfeld & Pandis, 2016, p. 594-596). Partitioning is influenced by complex interactions between compounds, which are challenging to analytically model in detail.

A simpler approach is to measure the partitioning ratio directly by measuring the concentrations in both phases at equilibrium. This is the rudimentary idea behind equilibrium partition coefficients. However, as mentioned in section 1, empirical measurements are extremely impractical for atmospheric compounds. Thus in atmospheric science, determining these coefficients is heavily reliant on computational methods.

In atmospheric science, a compound's partitioning is often described with two coefficients: one explaining the hydrophilic properties (water-solubility) and the other explaining interactions with water-insoluble matter. In this thesis, the partition coefficient between the gas phase and infinitely diluted water solution phase, $K_{\text{W/G}}$, is used.

Assuming an infinitely diluted solution isolates the system to a non-self-interacting compound, which simplifies the hydrophilic interactions to between water and the compound. For describing interactions with water-insoluble matter, the partition coefficient between the gas phase and the water-insoluble organic matter phase $K_{\mathrm{WIOM/G}}$ is used. This describes the compound's tendency to form clusters with other particulate matter via adsorption.

In atmospheric science, partition coefficients are usually determined using quantum chemistry simulations. One method is to use an appropriate parameterisation of the conductor-like screening model for real solvents (COSMO-RS), for instance, the COSMOtherm software, which has been used in several studies for various thermodynamic property prediction tasks (Hyttinen & Prisle, 2020; Besel et al., 2023; Wang et al., 2017). In fact, molecular data sets that are used in this thesis originate from such simulation models, but this is discussed in more detail in section 3.2.

2.4 Molecular descriptors as features

Physico-chemical properties such as saturation vapour pressure and equilibrium partition coefficients arise from the structure of a molecule. However, using machine learning models requires such structural data to be converted to machine-readable features. In machine learning predictions, the choice of features is crucial for achieving accurate predictions. Extracting and transforming raw data to meaningful features is referred to as feature engineering. For molecules, specifically, this representation of the data is referred to as a descriptor. Here, feature engineering is conducted by developing a new molecular descriptor from the MACCS descriptor (presented in detail in sections 3.5 & 3.6).

To compare the new descriptor, two types of descriptors are considered in this thesis: physical descriptors and cheminformatics descriptors. Physical descriptors encode physical distances and angles of atoms in a molecule three dimensionally along with their charge or atom numbers. Conversely, cheminformatics descriptors (e.g., MACCS) may encode a wide variety of chemical information about molecules such as, the existing substructures, topology or physicochemical properties, or some combination of different information.

While predictive performance is arguably the most important quality of a molecular descriptor in machine learning, it is not the only criterion by which descriptors are benchmarked in research applications. For example, having distinct features that have a semantic meaning allows for applying domain knowledge to feature selection. Then, feature importance analysis may be performed, which reveals statistical dependencies between the features and the predicted variable. This is a major disadvantage for physical descriptors and topological cheminformatics descriptors, since performing this analysis is not trivial and this makes them essentially a black box model, where it is hard to encapsulate which factors contributed most to the predictive performance.

2.5 Supervised learning with kernel ridge regression

In this thesis, the machine learning model used is kernel ridge regression (KRR), a supervised learning method. The model is trained with an input of a set of features and an output of the target variable. In simpler terms, the idea is that a collection of properties of a molecule are given to the model and the model estimates what the target number should be. This estimate is then compared to the real value, and with this feedback, the model adjusts its future estimates. A more rigorous explanation of KRR requires mathematical details that are provided in section 3.7.

Unlike most regression models, KRR is a non-linear regression model, which allows the model to learn complicated non-linear relationships in a high dimensional feature space (Witten et al., 2017, ch. 7). Linearity is too strong of an assumption for complex problems such as molecular property prediction, thus non-linear models such as KRR are often used. KRR has been used in several studies for molecular property prediction (Fabregat et al., 2022; Li & Rangarajan, 2019; Lumiaro et al., 2021; Stuke et al., 2019; Stuke et al., 2021).

3 Research material and methods

Research methods used in this thesis are centred around molecular descriptor development and training and testing the machine learning model. In practice, the data processing and computations were implemented in Python. Most notable libraries that were used for data processing were Pandas and NumPy, which were used in almost every script file for organising and manipulating data in an efficient manner. OpenBabel (Hutchison et al., 2011), an open source chemistry toolkit, was used for calculating molar masses of the molecules, converting molecular string representations and other miscellaneous data preprocessing steps. The specific versions of all software used are listed in the source code repositories (Appendix A.1).

3.1 Volatility classification

Categorising the molecules to volatility classes, as presented in Table 1, is achieved by converting saturation vapour pressures P_{sat} to effective saturation mass concentrations C^* by invoking the ideal gas law

$$P_{\text{sat}}V = nRT, \tag{1}$$

where V, n, and T are the volume, amount of substance and temperature of the gas, and R is the gas constant. The mass concentration C is given by

$$C = \frac{Mn}{V}, \tag{2}$$

where M is the molar mass. Note that the effective mass concentration is defined as (Donahue et al. 2012)

$$C = \lambda C^*, \tag{3}$$

where λ is the activity coefficient, which is a correction term accounting for non-ideal behaviour. For simplicity, approximately ideal behaviour may be assumed s.t. $C \approx C^*$. Combining equations (1) & (2), rearranging and approximating $C \approx C^*$ yields

$$P_{\text{sat}} = \frac{C}{M}RT \approx \frac{C^*}{M}RT \tag{4}$$

This is the closed form relation between P_{sat} and (effective) saturation mass concentration that will be used for classifying volatility based on P_{sat} predictions.

3.2 Properties of the data sets

The prediction accuracies of different vapour prediction models differ between volatility classes. Typically, the prediction accuracy is worst for ELVOCs and LVOCs, as most models tend to overestimate the volatility in said regions, but proper research on this is lacking due to scarce data on ELVOCs and LVOCs. Thus, the distribution of volatilities is essential for assessing the performance of predictions. Furthermore, supervised machine learning is sensitive to biases in the training data, and often tends to overfit to data that is similar to the most frequently occurring data.

The primary data set used for training and testing the model is sourced from Wang et al. (2017). It consists of 3414 atmospherically relevant molecules, whose properties were calculated using COSMOtherm. Molecules in the data set are represented using simplified molecular-input line-entry system (SMILES) encoding. SMILES is a string representation of the structure of a molecule. This is the largest dataset currently available for partition coefficients and saturation vapour pressures in the scope of atmospheric science (Lumiaro et al., 2021).

The Wang et al. (2017) data set was generated using COSMOtherm with temperature parameter $T = 288$ K. By rearranging equation (4), saturation mass concentrations can be calculated, and classified according to Table 1. The distribution of the volatility classes in the Wang et al. (2017) are presented in Table 2. The distribution shows that the data set consists of mostly IVOCs. Additionally, the data set has no ELVOCs, and LVOCs are only marginally present.

Table 2: Volatility class counts of Wang et al. (2017) and GeckoQ data sets

Class	**Volatility** (μg m^{-3})	**Counts** (Wang)	**% of data set** (Wang)	**Counts** (GeckoQ)	**% of data set** (GeckoQ)
ELVOC	$C^* < 3 \times 10^{-4}$	0	0.0 %	644	2.0 %
LVOC	$3 \times 10^{-4} < C^* < 0.3$	25	0.7 %	5921	18.7 %
SVOC	$0.3 < C^* < 300$	430	12.6 %	16378	51.8 %
IVOC	$300 < C^* < 3 \times 10^6$	2255	66.1 %	8528	27.0 %
VOC	$C^* > 3 \times 10^6$	704	20.6 %	166	0.5 %

The secondary data set used in this thesis, the GeckoQ data set from Besel et al. (2023), contains 31637 atmospherically relevant molecules. Similarly to the Wang et al. (2017) data set, P_{sat} values were calculated using COSMOtherm, but with a different temperature parameter T = 298 K. Contrary to the Wang et al. (2017) data set, GeckoQ does not contain partition coefficients. Additionally, molecules in GeckoQ are less volatile and have more than double the oxygen-to-carbon atom ratio (Table A5 & A6), thus molecules in GeckoQ are more oxygenated. The distribution of volatility classes of GeckoQ's molecules are also presented in Table 2.

3.3 Molecular descriptor generation

For generating the molecular descriptors, the scripts from Lumiaro et al. (2021) were used to first ensure correct functioning of the program. This ensures that the results are comparable to previous findings. Physical descriptors such as MBTR, are generated from position data of the atoms within the molecule. The position data encodes physical distances and angles in a 3D coordinate representation. MBTR was generated using DScribe library (Himanen et al., 2020) and ASE library (Larsen et al., 2017).

Cheminformatic descriptors, MACCS and TopFP, are generated from structural information of the molecule. TopFP implements a topological approach, where the atoms of the molecule are encoded as lists of straight paths. MACCS structural keys are generated by asking 166 boolean questions e.g. "Is there an fluorine atom?" and storing the answers as a vector of binary values. The structure of the MACCS descriptor is presented in Figure 1. For generating cheminformatic descriptors, the RDKit library (RDKit, 2023) was used.

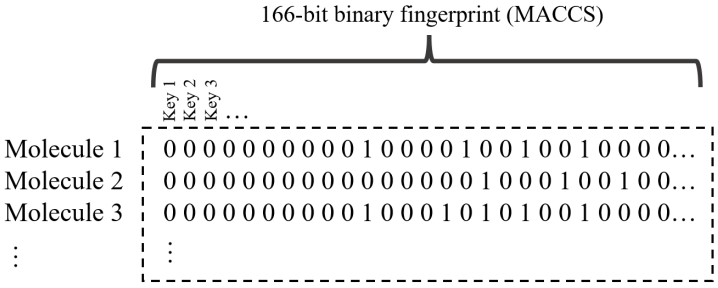


Figure 1: The structure of a MACCS descriptor file

3.4 Analysis on Molecular Access System descriptor

To improve the MACCS descriptor for atmospheric science, first it is good to understand why it performed poorly in Lumiaro et al. (2021). Analysing the distribution of MACCS structural keys present in the Wang et al. (2017) data set with descriptive statistical analysis shows that a large number of MACCS keys are always zero. This means that for example each of the 3414 molecules in the Wang et al. (2017) data set,

none of the molecules exhibit the properties that are queried via these MACCS keys. These keys shall be referred to as unused MACCS keys. For both data sets, a list of the unused MACCS keys and their corresponding questions are presented in the appendix in Table A1 and A2. Note that the GeckoQ data set had more unused keys, meaning that MACCS struggles to capture significant differences between these molecules. It is hypothesised that some of the questions may be irrelevant for atmospheric organic compounds, and thus replacing them is a viable option. To reiterate, the challenge is to select queries tailored for atmospheric organic compounds to enhance the applicability of the MACCS descriptor for predicting molecular properties with machine learning.

The MACCS descriptor was chosen to be improved because of its unique advantages. The benefit of using semantically meaningful binary keys as features for machines learning is that the data is in a human-interpretable form and easily customisable via the selection of binary questions. This allows for a more intuitive feature selection process where domain knowledge can be applied effectively. Additionally, such binary features are memory-efficient compared to other binary descriptors such as TopFP, which had an optimised bit-length of 8192 bits in Lumiaro et al. (2021), which is roughly 50 times larger compared to MACCS. Furthermore, generating MBTR and TopFP for the Wang et al. (2017) data set yields file sizes of 303 megabytes and 55 megabytes for the descriptors respectively. Generating the MACCS descriptor yields a file size of only one megabyte, making it the most compact descriptor of these three, and consequently it is fastest to generate and perform computations on; a smaller file size reduces the memory requirements for using machine learning models.

3.5 SIMPOL.1 groups and SMARTS pattern enumeration

Understanding atmospheric organic compounds requires some understanding of organic chemistry, namely about molecular substructures of organic compounds. Chemical properties of such molecules are often dominated by the functional groups present in a molecule, which suggests that an effective approach in predicting properties of atmospheric molecules is to analyse their functional groups.

A promising method is to use the SIMPOL.1 (SIMPOL) groups from Pankow & Asher (2008), which have been used in predicting vapour pressures of atmospheric compounds with a group contribution method. SIMPOL groups are molecular substructures, but not strictly functional groups. For example, the carbon number is just the number of carbon atoms present in the molecule. The group attribution method is based on enumerating molecular substructures with SMILES arbitrary target specification (SMARTS) patterns. SMARTS patterns are essentially string queries for finding molecular substructures from SMILES strings. This same substructure search method is used for generating the MACCS descriptor as well.

A Python program for SMARTS pattern enumeration has been developed by Ruggeri & Takahama (2016). They used Python 2 for implementing the substructure search scripts, and OpenBabel, an open-source cheminformatics tool kit developed by Hutchison et

al. (2011) for reading SMILES strings and SMARTS patterns. The key functionality of the program is simple: given an input SMILES string, the program counts the number of each selected SMARTS pattern present in the molecule. The program was ported to Python 3 by Dr. Hilda Sandström and it was originally used for enumerating SIMPOL functional groups for the SIMPOL group attribution method. In addition to the SIMPOL groups, the program included other SMARTS patterns, which were also included in the feature engineering process. The program includes the SIMPOL.1 group attribution method, which was calculated by Dr. Hilda Sandström for both data sets for results comparison.

3.6 Implementing SMARTS patterns as binary keys

To encode enumerated SMARTS patterns similarly to MACCS, they must be formulated to binary questions, for example, "Is there at least one carbonyl group?" or "Is the carbon number of the molecule larger than five?". This method shall be referred to as binary encoded SMARTS pattern enumeration (BESPE). The idea is to replace atmospherically irrelevant MACCS keys by applying the BESPE method on SMARTS patterns provided by the substructure search program.

Initially, improving MACCS with BESPE was achieved by replacing the 51 unused MACCS keys (Table A1) with BESPE features. The Wang et al. (2017) data set was analysed for SMARTS patterns that were provided by the substructure search program. The analysis shows that out of 40 SMARTS patterns, there were 27 patterns that were present and 13 patterns that were absent. Furthermore, of the 27 patterns, there was a subset of 20 patterns that could appear multiple times in a molecule. A list of all considered SMARTS patterns are listed in Table A4.

Note that not all patterns were considered initially. Nitrophenol, non-aromatic ring, aromatic ring, oxygen count, carbon number and ester all (ester or nitroester) were implemented in later versions of the descriptor. This was due to not being able to represent these substructures with SMARTS patterns alone, requiring some additional logic to be implemented in the substructure search program. Thus, initially, only 21 patterns were considered for the first two versions of BESPE-MACCS. The carbon number was considered separately from all other patterns in the third version, since it was hypothesised that a more specific binary enumeration of the carbon number could be an effective predictor of the target variables. This is because molecules with longer carbon chains generally tend to be less volatile (Pankow & Asher, 2008).

The first version of a BESPE-MACCS descriptor was implemented by replacing the first 21 unused keys by asking the analogous question "Does this pattern exist in the molecule?" for each of the patterns. With 30 unused keys left to be replaced, the second version replaces further 15 unused keys with questions "Does this molecule have two to four instances of this pattern?" for the 15 patterns that could appear multiple times (indicated by the max column of Table A5). Most of these patterns had a maximum occurrence ranging from two to four within the dataset. A more comprehensive list of

the SMARTS pattern statistics for both data sets are presented in the appendix A5 and A6. Having many of these patterns in a molecule is correlated with the molecules size, which in turn is inversely correlated with volatility, and will affect partitioning to condensed phase positively. This implicit information alone is expected to improve the predictive performance significantly.

The remaining 15 unused keys were replaced with a 15-bit one-hot encoding of the carbon number. Conveniently, the maximum number of carbon atoms for the Wang et al. (2017) data set was 15. A 15-bit one-hot encoding corresponds to the question "Is the carbon number X?", where $X = 1, 2, 3, ..., 15$. Note that while this may be effective for this particular data set, it is not effective for data sets, where the carbon number is higher than 15. With these features included, the third BESPE-MACCS version was created. The structure of this descriptor is presented in Figure 2.

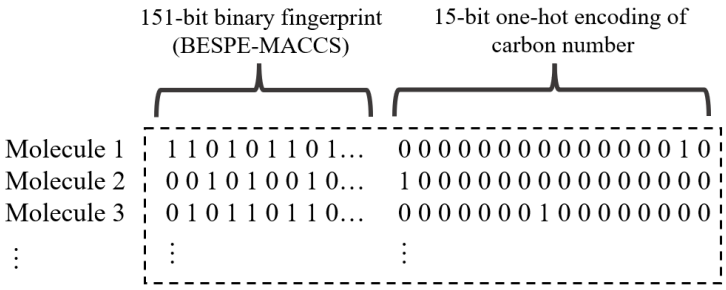


Figure 2: Structure of a BESPE-MACCS (3) descriptor file

While one-hot encoding accurately and simply represents the number of carbon atoms, it has some limitations. If the carbon number of a molecule exceeds the number of bits allocated for it, the descriptor is unable to store this information. Furthermore, one-hot encoding is a sparse representation of information; determining the number of carbon atoms of each molecule in a data set requires max(carbon number) number of bits, which clutters the feature space with lots of zeroes.

A more efficient approach is to encode the number of carbon atoms in binary form, reducing the clutter in the feature space. Additionally, since GeckoQ has many oxygen atoms per molecule, the oxygen count of molecules was encoded in a similar fashion at this stage. This marks the fourth version of the descriptor. Note that encoding the oxygen count in this way allows the removal of four MACCS keys (140, 146, 159 & 164, presented in Table A3), which are oxygen count keys that become semantically redundant after binary encoding, thus these were removed from MACCS.

The fifth version finalises the encoding of multiple patterns with the question "Does the molecule have more than four instances of this pattern?". Finally, the sixth version incorporates the nitrophenol group, non-aromatic rings and aromatic rings. For the sixth version, to keep the bit-length of the descriptor the same as MACCS at 166 bits,

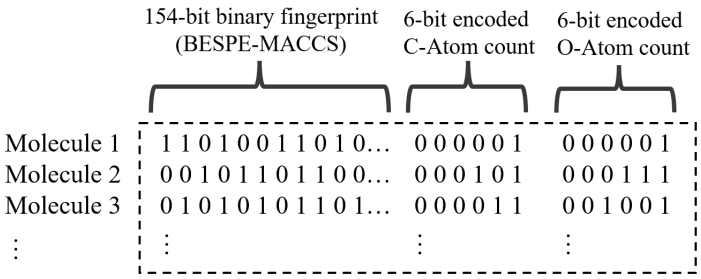


Figure 3: Structure of a BESPE-MACCS (4), (5) & (6) descriptor file

the hydroxyl group MACCS key (139) was removed, since it is included in BESPE features. Additionally, MACCS keys (100) and (17) were removed, since the only appeared once in the Wang et al. (2017) data set. These removed MACCS keys are presented in Table A3. Versions 4, 5 and 6 of the BESPE-MACCS descriptor have the structure shown in Figure 3. A summary of the development steps of the BESPE-MACCS descriptor are presented in Table 3.

Table 3: Steps of descriptor development

Descriptor version	Added binary features	New binary features
BESPE-MACCS (1)	Added single patterns (one or more instance)	21
BESPE-MACCS (2)	Added multiple patterns (2-4 instances)	15
BESPE-MACCS (3)	Added carbon number (15-bit one-hot encoding)	15
BESPE-MACCS (4)	Added carbon number and oxygen count encoding (6-bit binary encodings) (Removed one-hot encoding of carbon number)	12 (-15)
BESPE-MACCS (5)	Added multiple patterns ($>$ 4 instances)	5
BESPE-MACCS (6)	Added nitrophenol, ester (all), non-aromatic and aromatic rings	5

3.7 Kernel ridge regression

Understanding predictions that a kernel ridge regression model produces requires knowledge of the mathematical theory behind KRR. In this section, the KRR model is rigorously defined from ordinary least squares (OLS) regression. Key properties of KRR are presented in the intermediate steps of this derivation.

The mathematical formulation of KRR is based on adding a penalty term to the ordinary least squares fit of linear regression and then applying the kernel trick. Here, the KRR minimisation problem is derived from OLS linear regression. Let $\mathcal{X}$ denote the set of features and $\mathcal{Y}$ denote set of targets. In linear regression, first consider the plain mean squared loss (MSE) loss function

$$\mathrm{L}_{\text{linear}}(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{MSE}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j \boldsymbol{x_j} \cdot \boldsymbol{x_i} \right)^2, \tag{5}$$

where n is the number of data instances, $\boldsymbol{x_{i,j}} \in \mathcal{X}$ are feature vectors, $y_i \in \mathcal{Y}$ are the target values and α_j are the regression weight coefficients. Minimising this MSE loss function yields the regression weight coefficients, which form the best linear fit for the model. For ridge regression, a penalty term P is introduced to the loss function

$$\mathrm{L}_{\text{ridge}}(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{MSE} + \mathrm{P} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j \boldsymbol{x_j} \cdot \boldsymbol{x_i} \right)^2 + \frac{\lambda}{n} \sum_{i,j=1}^{n} \alpha_i \alpha_j \boldsymbol{x_j} \cdot \boldsymbol{x_i}, \tag{6}$$

where $\lambda \in \mathbb{R}$ is a weight factor that determines the magnitude of penalisation. This penalisation ensures that less emphasis is placed on individual data instances (Witten et al., 2017, ch. 7) by penalising large coefficients values. This makes ridge regression generally more robust against outliers, and regularises the weight coefficients closer to zero. Additionally, this regularisation reduces the variance of the weight coefficients, mitigating the negative impact of multicollinearity biases on predictions.

Next the kernel trick is performed: instead of the dot product $\boldsymbol{x_j} \cdot \boldsymbol{x}$, other inner products may be considered. By Mercer's theorem, a positive definite kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is associated with some inner product (Hoffman et al., 2008)

$$k(\boldsymbol{x_i}, \boldsymbol{x_j}) = \langle \phi(\boldsymbol{x_j}), \phi(\boldsymbol{x_i}) \rangle, \tag{7}$$

where ϕ is some feature map that maps into some inner product space. Now, a $n \times n$ kernel matrix may be defined as (Hoffman et al., 2008)

$$\boldsymbol{K} = (k(\boldsymbol{x_j}, \boldsymbol{x_i}))_{ij}. \tag{8}$$

The kernel matrix contains the pairwise similarity information of each data instance in the feature space transformed by the kernel. By substituting the dot products in eq. (6) with the kernel function in eq. (7) the loss function becomes

$$\mathrm{L}_{\text{KRR}}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j k(\boldsymbol{x_j}, \boldsymbol{x_i}) \right)^2 + \frac{\lambda}{n} \sum_{i,j=1}^{n} \alpha_i \alpha_j k(\boldsymbol{x_j}, \boldsymbol{x_i}), \tag{9}$$

introducing the ability to capture of non-linear relationships with the kernel trick. Finally, expressing the loss function more concisely via the kernel matrix from eq. (8) yields

$$\mathrm{L}_{\text{KRR}}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j k(\boldsymbol{x_j}, \boldsymbol{x_i}) \right)^2 + \frac{\lambda}{n} \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{K} \boldsymbol{\alpha}. \tag{10}$$

It follows that regression coefficients α_i satisfy the minimisation problem (Stuke et al., 2021, Lumiaro et al., 2021)

$$(10) \Rightarrow \arg\min_{\boldsymbol{\alpha}} \left(\mathrm{L}_{\mathrm{KRR}}\right) = \arg\min_{\boldsymbol{\alpha}} \left(\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{n} \alpha_j k(\boldsymbol{x_j}, \boldsymbol{x_i}) \right)^2 + \lambda \boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{K} \boldsymbol{\alpha} \right), \quad (11)$$

which has the analytical solution (Lumiaro et al., 2021)

$$\Rightarrow \boldsymbol{\alpha} = (\boldsymbol{K} - \lambda \boldsymbol{I})^{-1} \boldsymbol{y}. \quad (12)$$

The choice of kernel to use affects the accuracy of the model's predictions. In Lumiaro et al. (2021), it was determined that for the Wang et al. (2017) data set, the Gaussian kernel performed better than the Laplacian kernel. A Gaussian kernel with width parameter γ is defined as

$$k_G(\boldsymbol{x_i}, \boldsymbol{x_j}) = e^{-\gamma ||\boldsymbol{x_j} - \boldsymbol{x_i}||_2^2}. \quad (13)$$

It is important to note that equation (12) implies that solving the regression coefficients of the KRR model is inherently a matrix inversion operation on the $n \times n$ kernel matrix, where n is the number of data instances. From this remark, the computational time and space complexity for training and testing the KRR model is derived. Unfortunately with most implementations, the kernel matrix must be stored in memory, which has space complexity $O(n^2)$. Additionally, a matrix inversion algorithm has a time complexity of around $O(n^3)$ (Witten et al., 2017, ch. 7). This imposes a practical issue for training and testing with the GeckoQ data set, which is an order of magnitude larger than the Wang et al. (2017) data set.

3.8 Computational execution

Following the methodology of Lumiaro et al. (2021), the KRR model was run iteratively 10 times with hyperparameter optimisation for the kernel width γ and penalisation weight factor λ to ensure optimal hyperparameter values for the model. Between each run, different random seeding was used for shuffling the data set to obtain pseudorandom training and test sets. This was done to obtain statistically significant results. More iterations could be considered to further reduce statistical uncertainty, but ultimately the choice was to maintain consistency with the methodology of Lumiaro et al. (2021). First the baseline results for predictions were established by testing and training the model with the MACCS descriptor, to which the effect of subsequent addition of features may be compared to. Additionally, the TopFP descriptor was trained and tested as in Lumiaro et al. (2021). The random seeding was found to cause small differences in prediction MAE compared to those found in Lumiaro et al. (2021). The random seeds used were not reported in that study, which is the primary reason behind rerunning MACCS and TopFP for the Wang et al. (2017) data set.

For the Wang et al. (2017) data set, the KRR model was trained and tested with six different versions of the BESPE-MACCS descriptor with each subsequent version

having more features (Table 3). Additionally, for the sixth version of BESPE-MACCS, an additional experiment was conducted with only BESPE features included. All target variables $\log(P_{\text{sat}})$, $K_{\text{WIOM/G}}$ and $K_{\text{W/G}}$ were predicted with each version. To capture the learning curve for the model, six different training sizes were used: 500, 1000, 1500, 2000, 2500, and 3000, and for testing the model, a fixed test size of 414 was used for all training sizes. All computations with the Wang et al. (2017) data set were computed locally on a high-performance personal computer.

Computations for the GeckoQ data set required the use of a computer cluster due to the larger size of the data set, considering the time and space complexity of KRR presented in section 3.7. To save energy and computational resources, only the more advanced versions 3-6 were used for training and testing the KRR model on the GeckoQ data set. Similarly to the Wang et al. (2017) data set, the same additional experiment was performed by only including the BESPE features. Only the target variable $\log(P_{\text{sat}})$ was computed, since the data set did not include partition coefficients. To maintain consistency with both data sets, proportionally similar training sizes were used: 4633, 9267, 13900, 18534, 23167, and 27801, and a fixed test size of 3836.

4 Results & Discussion

Unless specified otherwise, all results presented here are averaged with an arithmetic mean of 10 iterations of shuffling the data, and then training and testing the KRR model. The most important performance metric that was used in comparing descriptors was the test MAE with the largest training size.

The forward selection methodology is guided by incremental improvements in the results; when addition of a set of features is found to significantly reduce the test MAE, it is likely that said features are effective predictors of the target variable. However, note that the converse is not necessarily true i.e. an addition of a set of features may not alone significantly reduce the test error, but may interplay with later added features positively. Furthermore, a feature's effectiveness is intrinsically data set dependent. These considerations are important for interpreting the results presented here.

4.1 Saturation vapour pressure predictions

4.1.1 Wang et al. (2017) data set

The prediction errors of P_{sat} predictions for the compared descriptors are presented in Table 4. In first three versions of the BESPE-MACCS descriptor, a significant decrease of the test MAE was achieved compared to the MACCS descriptor. The changes in the test MAE were similar in magnitude across the first three versions. The third version outperforms the topological fingerprint by 0.0244 logarithmic units (kPa). However, with different unknown random seeds, Lumiaro et al. (2021) found 0.01 logarithmic units smaller prediction MAEs on average for MACCS and TopFP. In latter versions of BESPE-MACCS, changes in MAE are insignificantly small, and for the fifth and

Table 4: Training and testing metrics of saturation vapour pressure P_{sat} predictions with the largest training size of 3000 on Wang et al. (2017) data set

Descriptor	**Training MAE** $\log_{10}$(kPa)	**Test MAE** $\log_{10}$(kPa)	**Test** R^2
TopFP	0.1379	0.3249	0.9515
MACCS	0.2090	0.4355	0.9136
BESPE-MACCS (1)	0.2417	0.3902	0.9322
BESPE-MACCS (2)	0.1607	0.3484	0.9436
BESPE-MACCS (3)	0.1824	0.3025	0.9591
BESPE-MACCS (4)	0.1773	0.2979	0.9602
BESPE-MACCS (5)	0.1803	0.3013	0.9594
BESPE-MACCS (6)	0.1821	0.3029	0.9591
BESPE (6)	0.3327	0.4514	0.9108

sixth versions, the prediction error begins to increase. Training MAE for versions 2-6 are very similar, with no clear signs of considerable overfitting. Interestingly, in the first version of BESPE-MACCS, the training MAE is disproportionately larger compared to the other descriptors.

The learning curves for TopFP, MACCS and the first four versions of the BESPE-MACCS descriptor are presented in Figure 4. BESPE-MACCS versions 3-6 outperform the MACCS descriptor by a large margin and the TopFP descriptor by a considerable margin for all training sizes. The training curve gradient is similar to TopFP for larger training sizes, which suggests that the BESPE-MACCS could scale similarly to TopFP

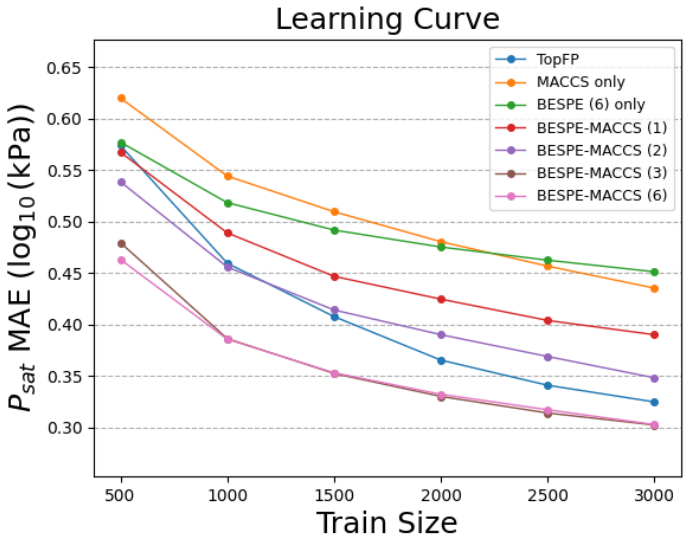


Figure 4: Test error training curve of saturation vapour pressure $\log_{10}(P_{sat})$ predictions for the Wang et al. (2017) data set for different descriptors. BESPE-MACCS versions 4 & 5 omitted due to curve overlap with version 6 (Table A7).

for larger data sets. For all training sizes, the prediction MAE are greatest for MACCS and BESPE separately, although at lower training sizes, BESPE outperforms MACCS, but quickly plateaus as the training size is increased.

With the additional experiment of training and testing the KRR model with just BESPE features, it is evident that the performance is subpar, even worse than that of the MACCS descriptor. This is an important finding, since this suggests that the KRR model establishes relationships between MACCS features and BESPE features, although showing this rigorously would require further research. Although MACCS is not explicitly a physical descriptor, it does include some features that contain information about relative placement of molecular substructures, e.g. key 95: "Is there a nitrogen and oxygen atom separated by two atoms?" It is possible that this relative placement information interplaying with BESPE features is significantly contributing to the decrease in MAE.

Compared to traditional methods for P_{sat} prediction, e.g., the SIMPOL.1 group attribution method, the KRR machine learning method approach with BESPE-MACCS descriptor yields a much lower MAE. For the entire Wang et al. (2017) data set, the SIMPOL method provides an MAE of 1.0632 logarithmic units. In standard units this corresponds to roughly a twelvefold over or underestimation making it essentially an order of magnitude prediction. In contrast, the BESPE-MACCS (6) provides, on average, a twofold over or underestimation. However for most applications, even an order of magnitude prediction is useful for volatility classification according to Donahue et al. (2012) classification presented in Table 1.

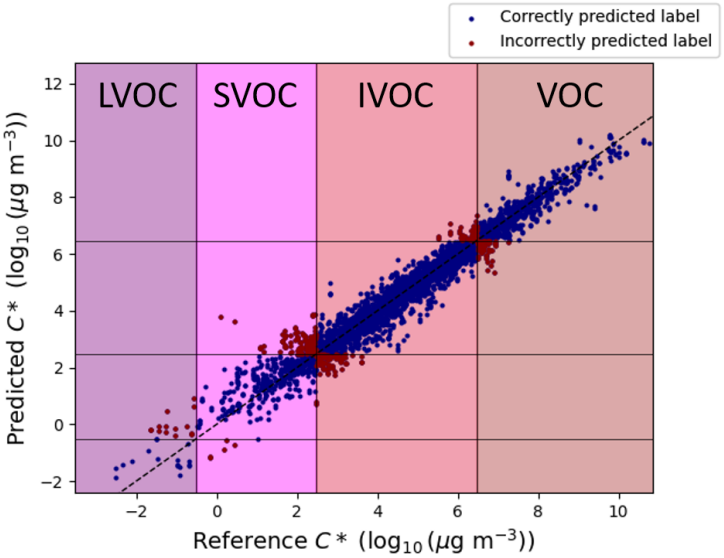


Figure 5: Scatter plot depicting reference effective saturation mass concentration values values against predicted values generated using the BESPE-MACCS (6) descriptor with the largest training size for the Wang et al. (2017) data set.

Figure 5 displays the incorrect and correct predictions on a scatter plot, where the reference volatility classes are coloured. The scatter plot shows few outliers, and most incorrect labels are near the edges of the bins. However, it is important to note that the edges for these volatility classes are non-strict approximations; the behaviour of the compound determines which class it ultimately belongs to. Thus, these edges should be interpreted as a general reference for volatility classification. These predictions in the LVOC volatility region are most inaccurate, and most accurate in the IVOC region, which was the most common class in the data set.

4.1.2 GeckoQ data set

Saturation vapour pressures predictions on the GeckoQ data set were generally less accurate with two to three times larger MAE values in logarithmic units compared to predictions on the Wang et al. (2017) data set. This is expected, since molecules in GeckoQ were larger, more complex, had a larger range of possible volatility values (Besel et al., 2023). Moreover, this is consistent with previous findings as comparing results of Lumiaro et al. (2021) and Besel et al. (2023) arrives to a similar conclusion, although in the latter, a Gaussian process regression (GPR) model was used instead of KRR. The learning curves are presented in Figure 6, and surprisingly the shapes of the curves are quite similar across all descriptors.

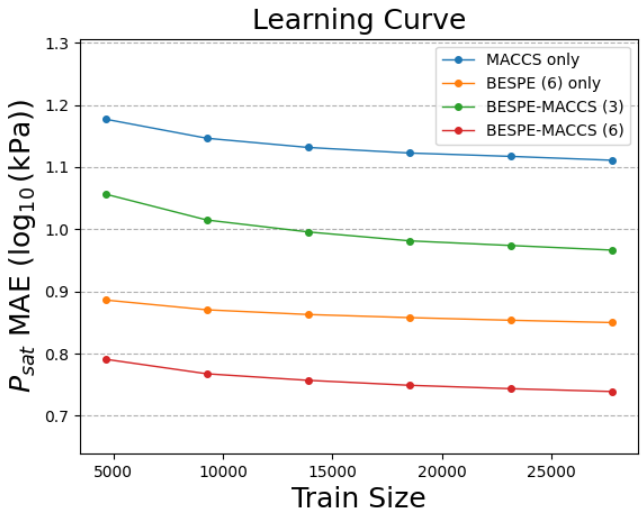


Figure 6: Test error training curve of saturation vapour pressure $\log_{10}(P_{\text{sat}})$ predictions for the GeckoQ data set for different descriptors. BESPE-MACCS versions 4 & 5 omitted due to curve overlap with version 6 (Table A8).

Besel et al. (2023) achieved a MAE of 0.82 logarithmic units with a GPR model using the TopFP descriptor on GeckoQ. Applying the SIMPOL group attribution method to GeckoQ yields a MAE of 2.30 logarithmic units. The most accurate result is attained with the BESPE-MACCS (6) descriptor and KRR, yielding a MAE of 0.74 logarithmic units, albeit not being directly comparable without testing the descriptor with GPR.

The properties of the model and the characteristics of GeckoQ are best encapsulated by the scatter plot presented in Figure 7. Predictions have more outliers, they have higher variance, and clearly are systematically overestimated in the ELVOC and LVOC regions and underestimated in the IVOC and VOC regions. Unsurprisingly, predictions in the SVOC region were less skewed in this regard, as most of the data set were SVOCs. This shows the general tendency for supervised learning models to favour predictions around the most common data points.

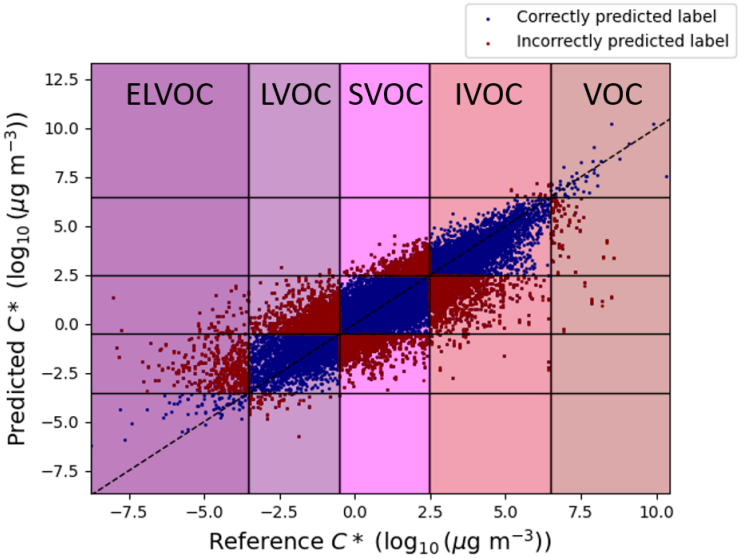


Figure 7: Scatter plot depicting reference effective saturation mass concentration values against predicted values generated using the BESPE-MACCS (6) descriptor with the largest training size for the GeckoQ data set from Besel et al. (2023).

To summarise P_{sat} predictions across both data sets, Table 5 shows the rate of correct predictions. Comparing to the ordering of percentages in Table 2, the rate of correct predictions are ordered according to their proportions in the data sets. Since only a small part of the data is from the ELVOC and LVOC regions, it is difficult to assess if the systematic overestimation in said regions is an inherent property of the BESPE-MACCS descriptor. This is an important open research question in future studies on the newly developed descriptor.

Table 5: Fraction of correct volatility classifications from P_{sat} predictions with BESPE-MACCS (6)

Class	Wang et al. (2017)	GeckoQ
ELVOC	Not present	17.2 %
LVOC	54.8 %	63.7 %
SVOC	75.8 %	85.5 %
IVOC	95.2 %	75.2 %
VOC	92.1 %	40.9%

4.2 Equilibrium partition coefficient predictions

The learning curves for predictions on the infinitely diluted water-gas equilibrium partition coefficient $K_{W/G}$ for the different descriptors are presented in Figure 8. BESPE alone is by far the worst performing descriptor with the largest MAEs for all training sizes. Versions 3-6 of the BESPE-MACCS descriptor outperform the TopFP descriptor by a small but significant margin, and the gradient is roughly similar.

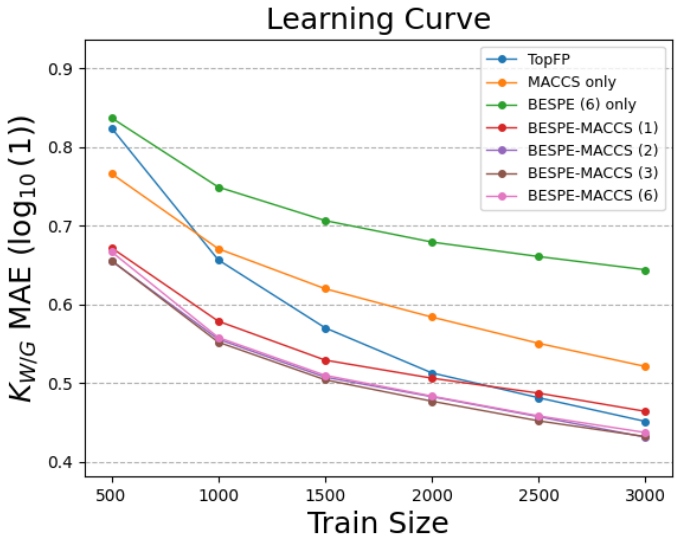


Figure 8: Test error training curve of infinitely diluted water-gas equilibrium partition coefficient $K_{W/G}$ for the Wang et al. (2017) data set for different descriptors. BESPE-MACCS versions 4 & 5 omitted due to curve overlap with version 6 (Table A9).

Similarly, the learning curves for the water insoluble organic matter-gas equilibrium partition coefficient $K_{WIOM/G}$ are presented in Figure 9. However, MACCS is outperformed by BESPE for the smaller training sizes, but is quickly reaching similar values for the largest training sizes, and likely would outperform BESPE with a larger training size due to the gradient being steeper. The $K_{WIOM/G}$ predictions are very similar to $K_{W/G}$, except that they are more accurate for each training size roughly by a constant value.

For equilibrium partition coefficient predictions on the Wang et al. (2017) data set, the results are largely similar to P_{sat} predictions, with considerable improvements in BESPE-MACCS versions 1-3 and insignificant changes in later versions. The results for equilibrium partition coefficients also show a similar phenomenon that MACCS and BESPE alone perform much worse than any versions of the BESPE-MACCS descriptor, and that the performance of versions 4-6 are slightly better than TopFP. This shows that BESPE-MACCS in unison is better able to determine partitioning at equilibrium than the other considered descriptors.

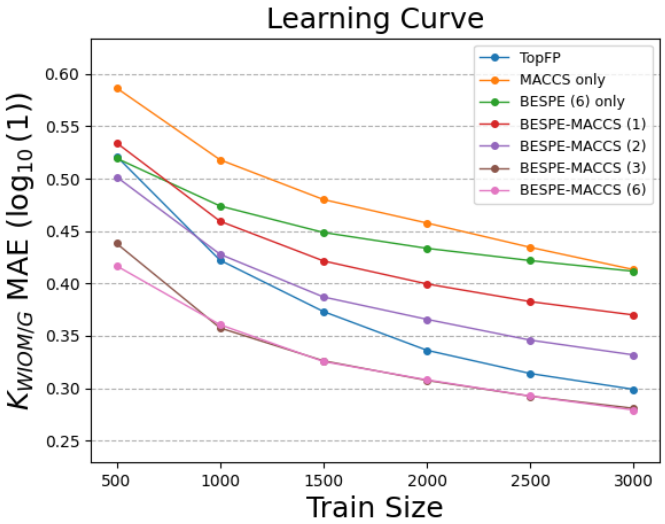


Figure 9: Test error training curve of water insoluble organic matter-gas equilibrium partition coefficient $K_{\mathrm{WIOM/G}}$ for the Wang et al. (2017) data set for different descriptors. BESPE-MACCS version 4 & 5 omitted due to curve overlap with version 6 (Table A10).

Furthermore, with only BESPE features included, the prediction MAE is by far the worst for $K_{\mathrm{W/G}}$ predictions. Interestingly, this phenomenon is not as pronounced in $K_{\mathrm{WIOM/G}}$ predictions, where MACCS and BESPE on their own are performing similarly at the largest training size. As described in section 3.6, the predictive success of BESPE is also due to implicitly capturing information of the size of a molecule via counting of the number of SMARTS patterns present. $K_{\mathrm{W/G}}$ is associated to water solubility, which may generally be less affected by the size of the molecule than $K_{\mathrm{WIOM/G}}$. It is also possible that SMARTS patterns included in BESPE are simply not enough to accurately describe the relevant intermolecular interactions such as dipole-dipole interactions (i.e., hydrogen bonding for the most part for organic compounds) that are crucial for accurately determining water solubility, since the relative positions of the patterns are completely unknown without MACCS keys.

5 Conclusions

In this thesis, a new molecular descriptor, the BESPE-MACCS descriptor, was developed and tested with a KRR machine learning model. The descriptor was iteratively improved and optimised with two atmospherically relevant data sets of molecules, the Wang et al. (2017) and GeckoQ data sets, leading to predictions that were on par or slightly better than other descriptors used in previous research (Lumiaro et al., 2021; Besel et al., 2023). Additionally, the descriptor offers several unique advantages such as being human-interpretable and customisable to the data set via choosing appropriate SMARTS patterns. This allows for easier feature importance analysis, which is useful for further research and development.

Compared to the original MACCS structural key descriptor, the addition of BESPE features lead to a significant increase in predictive performance for all target variables (P_{sat}, $K_{W/G}$, & $K_{WIOM/G}$) for both data sets when tested with a KRR model. However, the different versions of the BESPE-MACCS descriptor had varying incremental effects on the prediction MAEs, but the overall trend was that incremental changes became smaller with later versions. Versions 1-4 incrementally decreased the MAE significantly, while versions 5 & 6 showed insignificant differences. Thus, it is inferred that enumerated SMARTS patterns up to four instances, with binary encoding of carbon number and oxygen count provided the greatest positive impact on the prediction accuracy, while the specific substructures nitrophenol, ester (all), non-aromatic and aromatic rings and enumerating more than four instances of SMARTS patterns impacted the later versions of the BESPE-MACCS descriptor insignificantly.

There are many open research questions on the performance of BESPE-MACCS descriptor in machine learning predictions. Specifically for saturation vapour pressure predictions, volatility biases of the descriptor are still largely unknown. This question may be explored by introducing a data set with uniformly distributed volatilities in each of the Donahue et al. (2012) inspired classes. In addition, feature dependency on partition coefficients $K_{W/G}$ and $K_{WIOM/G}$ should be established to determine the cause of poor performance of BESPE features alone. Ultimately, the results of the newly developed descriptor successfully aligned with the goals of this thesis in a precise manner, providing a new tool for atmospheric science research.

References

Besel V., Todorović M., Kurtén T., Rinke P., & Vehkamäki H. (2023) *Atomic structures, conformers and thermodynamic properties of 32k atmospheric molecules* Sci Data 10, 450
https://doi.org/10.1038/s41597-023-02366-x

Bianchi F., Kurtén T., Riva M., Mohr C., Rissanen M. P., Roldin P., Berndt T., Crounse J. D., Wennberg P. O, Mentel T. F., Wildt J., Junninen H., Jokinen T., Kulmala M., Worsnop D. R., Thornton J. A., Donahue N., Kjaergaard H. G., & Ehn M. (2019) *Highly Oxygenated Organic Molecules (HOM) from Gas-Phase Autoxidation Involving Peroxy Radicals: A Key Contributor to Atmospheric Aerosol.* Chem. Rev. 2019, 119, 3472-3509
https://doi.org/10.1021/acs.chemrev.8b00395

Bilde M., Barsanti K., Booth M., Cappa C. D., Donahue N. M., Emanuelsson E. U., McFiggans G., Krieger U. K., Marcolli C., Topping D., Ziemann P., Barley M., Clegg S., Dennis-Smither B., Hallquist M., Hallquist Å. M., Khlystov A., Kulmala M., Mogensen D.,... Riipinen I. (2015) *Saturation vapor pressures and transition enthalpies of low-volatility organic molecules of atmospheric relevance: from dicarboxylic acids to complex mixtures.* Chem. Rev., 115, 10, 4115–4156
https://doi.org/10.1021/cr5005502

Donahue N. M., Epstein S. A., Pandis S. N., & Robinson A. L. (2011) *A two-dimensional volatility basis set: 1. organic-aerosol mixing thermodynamics*, Atmos. Chem. Phys., 11, 3303–3318
https://doi.org/10.5194/acp-11-3303-2011

Donahue N. M., Kroll J. H., Pandis S. N., & Robinson A. L. (2012) *A two-dimensional volatility basis set – Part 2: Diagnostics of organic-aerosol evolution*, Atmos. Chem. Phys., 12, 615–634
https://doi.org/10.5194/acp-12-615-2012

Dueben P. D., Schultz M. G., Chantry M., Gagne II D. J., Hall D. M., & McGovern A. (2022) *Challenges and Benchmark Datasets for Machine Learning in the Atmospheric Sciences: Definition, Status, and Outlook.* Artificial Intelligence for the Earth Systems, 1(3), e210002
https://doi.org/10.1175/AIES-D-21-0002.1

Fabregat R., van Gerwen P., Haeberle M., Eisenbrand F., & Corminboeuf C. (2022) *Metric learning for kernel ridge regression: assessment of molecular similarity*, Mach. Learn.: Sci. Technol. 3 035015
https://doi.org/10.1088/2632-2153/ac8e4f

Himanen L., Jäger M. O. J., Morooka E. V., Canova F. F., Ranawat Y. S., Gao D. Z., Rinke P., & Foster A. S. (2020) *DScribe: Library of descriptors for machine learning in materials science*,Computer Physics Communications, Vol. 247, 106949, ISSN 0010-4655
https://doi.org/10.1016/j.cpc.2019.106949

Hofmann T., Schölkopf B., & Smola A. J. *Kernel methods in machine learning*, Ann. Statist. 36 (3) 1171 - 1220
https://doi.org/10.1214/009053607000000677

Hutchison G. R., O'Boyle N. M., Banck M., James C. A., Morley C. & Vandermeersch T. (2011) *Open Babel: An open chemical toolbox*, J Cheminform 3, 33
https://doi.org/10.1186/1758-2946-3-33

Hyttinen N. & Prisle N. L. (2020) *Improving Solubility and Activity Estimates of Multifunctional Atmospheric Organics by Selecting Conformers in COSMOtherm*
https://doi.org/10.1021/acs.jpca.0c04285

International Union of Pure and Applied Chemistry (2019) *Compendium of Chemical Terminology: 'Saturation vapour pressure'*, UIPAC, Online version 3.0.1
https://doi.org/10.1351/goldbook.S05479

Jimenez J. L., Canagaratna M. R., Donahue N. M., Prevot A. S. H., Zhang Q., Kroll J. H., DeCarlo P. F., Allan J. D., Coe H., Ng N. L., Aiken A. C., Docherty K. D., Ulbrich I. M., Grieshop A. P., Robinson A. L., Duplissy J., Smith J. D., Wilson K. R., Lanz V. A.,... Worsnop D. R. (2009) *Evolution of Organic Aerosols in the Atmosphere*, Science 326, 1525-1529
https://doi.org/10.1126/science.1180353

Larsen A. H., Mortensen J. J., Blomqvist J., Castelli I. E., Christensen R., Dułak M., Friis J., Groves M. N., Hammer B., Hargus C., Hermes E. D, Jennings P. C., Jensen P. B., Kermode J., Kitchin J. R., Kolsbjerg E. L., Kubal J. Kaasbjerg K., Lysgaard S.,... Jacobsen K. W. (2017) *The atomic simulation environment—a Python library for working with atoms*, J. Phys.: Condens. Matter Vol. 29 273002
https://doi.org/10.1088/1361-648X/aa680e

Li B. & Rangarajan S. (2019) *Designing compact training sets for data-driven molecular property prediction through optimal exploitation and exploration.* Molecular Systems Design & Engineering
https://doi.org/10.48550/arXiv.1906.10273

Lumiaro E., Todorović M., Kurtén T., Vehkamäki H., & Rinke P. (2021) *Predicting gas–particle partitioning coefficients of atmospheric molecules with machine learning.* Atmos. Chem. Phys., 21, 13227–13246
https://doi.org/10.5194/acp-21-13227-2021

Nozière B., Kalberer M., Claeys M., Allan J., D'Anna B., Decesari S., Finessi E., Glasius M., Grgić I., Hamilton J. F., Hoffmann T., Iinuma Y., Jaoui M., Kahnt A., Kampf C. J., Kourtchev I., Maenhaut W., Marsden N., Saarikoski S.,... Wisthaler A. (2015) *The molecular identification of organic compounds in the atmosphere: state of the art and challenges*, Chem Rev., 115(10):3919-83.
https://doi.org/10.1021/cr5003485

North G. R. & Erukhimova T. L. (2009) *Atmospheric thermodynamics: elementary physics and chemistry*, Cambridge, UK: Cambridge University Press.
https://doi.org/10.1017/CBO9780511609695

Pankow J. F. & Asher W. E. (2008) *SIMPOL.1: a simple group contribution method for predicting vapor pressures and enthalpies of vaporization of multifunctional organic compounds*, Atmos. Chem. Phys., 8, 2773–2796
https://doi.org/10.5194/acp-8-2773-2008

RDKit version 2023.9.1 (2023) *Open-source cheminformatics*, www.rdkit.org
https://doi.org/10.5281/zenodo.8413907

Ruggeri G. & Takahama S. (2016) *Technical Note: Development of chemoinformatic tools to enumerate functional groups in molecules for organic aerosol characterization* (ported to Python 3 by Sandström H.), Atmos. Chem. Phys., 16, 4401–4422
https://doi.org/10.5194/acp-16-4401-2016

Seinfeld J. H. & Pandis S. N. (2016) *Atmospheric Chemistry and Physics: From Air Pollution to Climate Change* (3rd ed.) USA: John Wiley & Sons, Incorporated

Srivastava D., Vu T. V., Tong S., Shi Z., & Harrison R. M. (2022) *Formation of secondary organic aerosols from anthropogenic precursors in laboratory studies.*, npj Clim Atmos Sci 5, 22
https://doi.org/10.1038/s41612-022-00238-6

Stuke A., Todorović M., Rupp M., Kunkel C., Ghosh K., Himanen L., & Rinke P. (2019) *Chemical diversity in molecular orbital energy predictions with kernel ridge regression* J. Chem. Phys. 150, 204121
https://doi.org/10.1063/1.5086105

Stuke A., Rinke P., & Todorović M. (2021) *Efficient hyperparameter tuning for kernel ridge regression with Bayesian optimization.* Mach. Learn.: Sci. Technol. 2 035022
https://doi.org/10.1088/2632-2153/abee59

Wang C., Yuan T., Wood S. A., Goss K., Li J., Ying Q., & Wania F. (2017) *Uncertain Henry's law constants compromise equilibrium partitioning calculations of atmospheric oxidation products*, Atmos. Chem. Phys., 17, 7529–7540
https://doi.org/10.5194/acp-17-7529-2017,2017.

Witten I. H., Frank E., Hall M. A., & Pal C. J. (2017) *Data mining: practical machine learning tools and techniques* (4th ed.). Amsterdam: Elsevier

Zuend A. & Seinfeld J. H. (2012) *Modeling the gas-particle partitioning of secondary organic aerosol: the importance of liquid-liquid phase separation*, Atmos. Chem. Phys., 12, 3857–3882
https://doi.org/10.5194/acp-12-3857-2012

A Appendix

A.1 Source code

BESPE-MACCS descriptor generation

https://github.com/kuulia/BESPE-MACCS

SMARTS pattern enumeration

https://github.com/kuulia/SMARTS_pattern_enumeration

A.2 Supplemental tables

Table A1: List of unused MACCS keys in the Wang et al. (2017) data set

key	Question
0	empty key
1	Are there isotopes
2	Is there an atom with an atomic number above 103?
3	Are there atoms belonging to groups IVa, Va, Via, rows 4-6?
4	Is there an actinide in the molecule?
5	Does the molecule contain atoms from group IIIB or IVB?
6	Does the molecule contain lanthanides?
7	Does the molecule contain atoms from group VB, VIB or VIIB?
8	Is there a heteroatom at a 3 bond distance from a ring fragment?
9	Is there an atom from Group VIII?
10	Is there an atom from Group IIa?
12	Is there an atom from Group IB, IIB?
13	Does the molecule contain a hydroxylamine-derivative?
14	Does the molecule have a disulfide bond?
18	Is there an atom from Group VIII?
20	Is there an silicon atom?
23	Does the molecule contain an aminomethanediol?
25	Does the molecule contain an methanetriamine?
26	Is there a double-bonded carbon that has three ring bond?
27	Is there an iodine atom?
29	Is there an phosphor atom?
30	Is there a heteroatom at bonded to four atoms three of which are carbons?
31	Does the molecule have any heteroatom bonded to a halogen
32	Does the molecule have a aminothiomethane group?
33	Is there any bond between nitrogen and sulphur?
35	Is there alkali metal?
36	Is there a sulphur containing heterocycle?
37	Does the molecule have a diaminocarbinol group?
38	Does the molecule have a diaminoethyl group?
41	Does the molecule have a cyano group?
42	Is there an flourine atom?
44	Are there any atoms other than H, C, N, O, Si, P, S, F, Cl, Br, and I?
47	Is there sulphur separated by two bonds to nitrogen?
52	Are there two nitrogens bonded together?
59	Is the molecule lacking sulphur atoms?
62	Are there two rings separated by a bond?
64	Is there a carbon connected to an atom in a ring?
65	Is there an aromatic nitrogen or carbon?
68	Are there two bonded heteroatoms, both bonded to a hydrogen atom?
75	Is there a nitrogen in a ring that is connected to an atom outside the ring?
77	Are there two nitrogens bonded to the same atom?
78	Is there a nitrogen double bonded to a carbon?
81	Is there an atom with three bonds, one of which is connected to a sulphur atom?
84	Does the molecule have a primary amine?
85	Is there a nitrogen bonded to three carbons?
87	Is there a halide bonded to a ring?
98	Is there a heteroatom separated by five atoms to a ring?
121	Is there a nitrogen containing ring?
125	Is there more than one aromatic ring?
151	Is there a nitrogen bonded to at least one hydrogen?
166	Does the molecule have more than one fragments?

Table A2: List of unused MACCS keys in the Besel et al. (2023) GeckoQ data set

key	Question
0	empty key
1	Are there isotopes
2	Is there an atom with an atomic number above 103?
3	Are there atoms belonging to groups IVa, Va, Via, rows 4-6?
4	Is there an actinide in the molecule?
5	Does the molecule contain atoms from group IIIB or IVB?
6	Does the molecule contain lanthanides?
7	Does the molecule contain atoms from group VB, VIB or VIIB?
8	Is there a heteroatom at a 3 bond distance from a ring fragment?
9	Is there an atom from Group VIII?
10	Is there an atom from Group IIa?
12	Is there an atom from Group IB, IIB?
13	Does the molecule contain a hydroxylamine-derivative?
14	Does the molecule have a disulfide bond?
17	Does the molecule have 2 triplebonded carbons?
18	Is there an atom from Group VIII?
20	Is there an silicon atom?
25	Does the molecule contain an methanetriamine?
26	Is there a double-bonded carbon that has three ring bond?
27	Is there an iodine atom?
29	Is there an phosphor atom?
30	Is there a heteroatom at bonded to four atoms three of which are carbons?
31	Does the molecule have any heteroatom bonded to a halogen
32	Does the molecule have a aminothiomethane group?
33	Is there any bond between nitrogen and sulphur?
35	Is there alkali metal?
36	Is there a sulphur containing heterocycle?
37	Does the molecule have a diaminocarbinol group?
38	Does the molecule have a diaminoethyl group?
39	Is there a sulphur atom bonded to three oxygens?
40	Does the molecule have a thioperoxide (single bond between a sulphur and oxygen atom)?
41	Does the molecule have a cyano group?
42	Is there an flourine atom?
44	Are there any atoms other than H, C, N, O, Si, P, S, F, Cl, Br, and I?
46	Is there a bromine atom?
47	Is there sulphur separated by two bonds to nitrogen?
51	Is there a carbon bonded to sulphur to an oxygen?
52	Are there two nitrogens bonded together?
55	Is there a sulphur bonded to two oxygens?
58	Is there a sulphur atom bonded to two heteroatoms?
59	Is the molecule lacking sulphur atoms?
60	Does the molecule contain a double bond between a sulphur and oxygen atom?
61	Is there a sulphur atom bonded to three atoms?
62	Are there two rings separated by a bond?
64	Is there a carbon connected to an atom in a ring?
65	Is there an aromatic nitrogen or carbon?
67	Is there a heteroatom bonded to sulphur atom?
68	Are there two bonded heteroatoms, both bonded to a hydrogen atom?
73	Does the molecule have a double bonded sulphur?
75	Is there a nitrogen in a ring that is connected to an atom outside the ring?
77	Are there two nitrogens bonded to the same atom?
78	Is there a nitrogen double bonded to a carbon?
81	Is there an atom with three bonds, one of which is connected to a sulphur atom?
84	Does the molecule have a primary amine?
85	Is there a nitrogen bonded to three carbons?
86	Are there two methylene groups bonded to the same heteroatom?
87	Is there a halide bonded to a ring?
88	Is there a sulphur atom?
93	Is there a heteroatom bonded to a methyl group?
100	Is there a methylene bridge directly connected to a nitrogen atom?
103	Is there a chlorine atom?
107	Is there and atom with three bonds, one of which is to a halogen?
121	Is there a nitrogen containing ring?
125	Is there more than one aromatic ring?
134	Is there a halogen atom?
151	Is there a nitrogen bonded to at least one hydrogen?
166	Does the molecule have more than one fragments?

Table A3: List of additionally removed MACCS keys

key	Question
17	Does the molecule have 2 triplebonded carbons?
100	Is there a methylene bridge directly connected to a nitrogen atom?
139	Is there an hydroxyl group?
140	Is there more than three oxygens?
146	Is there more than 2 oxygen atoms?
159	Are there more than one oxygen atoms?
164	Is there a oxygen atom?

Table A4: SMARTS patterns used in BESPE

Pattern name	SMARTS pattern (or aprl_ssp substructure search query)
amine, primary	[C][NX3;H2;!$(NC=O)]([H])[H]
amine, secondary	[C][NX3;H;!$(NC=O)]([C])[H]
amine, tertiary	[C][NX3;H0;!$(NC=O);!$(N=O)]([C])[C]
alkane CH	[CX4][H]
alkene CH	[CX3;$(C=C)][H]
aromatic CH	[c][H]
carbonyl	[#6,#1][CX3](=O)[#6,#1]
hydroxyl (alkyl)	[C;!$(C=O)][OX2H][H]
carboxylic acid	[CX3](=O)[OX2H][H]
ester, all	[CX3H1,CX3](=O)[OX2H0][#6]
ester	ester, all-nitroester
ether	[OD2]([C;!R;!$(C=O)])[C;!R;!$(C=O)]
peroxide	[#6][OD2][OD2,OD1][#6]
nitro	[#6][$([NX3](=O)=O),$([NX3+](=O)[O-])](~[O])(~[O])
aromatic hydroxyl	[c;!$(C=O)][OX2H][H]
hydroperoxide	[#6;!$(C=O)][OD2][OX2H,OD1][#1]
amide	[CX3](=O)[NX3;!$(N=O)]([#6,#1])[#6,#1]
nitrate	[#6][O][NX3;$([NX3](=[OX1])(=[OX1])O), $([NX3+]([OX1-])(=[OX1])O)](~[O])(~[O])
organosulfate	[#6][O][SX4;$([SX4](=O)(=O)(O)O), $([SX4+2]([O-])([O-])(O)O)](~[O])(~[O])(~[O])
carbonylperoxynitrate	[C](=O)OO[N](~O)~[O]
ketone	[CX3;$(C([#6])(=[O])[#6])](=[O;!$([O][O])])
aldehyde	[CX3;$(C([#6,#1])(=[O])[#6,#1])]([#6,#1])(=[O;!$([O][O])])[H]
amide, primary	[CX3;$(C(=[O])[NX3;!$(N=O)])](=[O])[N]([#1])[#1]
amide, secondary	[CX3;$(C(=[O])[NX3;!$(N=O)]([#6])[#1])](=[O])[N][#1]
amide, tertiary	[CX3;$(C(=[O])[NX3;!$(N=O)]([#6])[#6])](=[O])[N]
carbonylperoxyacid	C(=O)O[O][H]
peroxy nitrate	[#6][O;!$(OOC(=O))][O;!$(OOC(=O))][N](~O)~[O]
carbon number	[#6]
ether, aromatic	c~[O,o]~[c,C&!$(C=O)]
ether (alicyclic)	[OD2;R]([C;!$(C=O);R])[C;!$(C=O);R]
amine, aromatic	[N;!$(NC=O);!$(N=O);$(Na)]
nitroester	[#6][OX2H0][CX3,CX3H1](=O)[C;$(C[N](~[O])~[O]), $(CC[N](~[O])~[O]),$(CCC[N](~[O])~[O]), $(CCCC[N](~[O])~[O]),$(CCCCC[N](~[O])~[O])]
C=C-C=O (non-aromatic)	[$(C=CC=O);A;R]
C=C (non-aromatic)	C=C
non-aromatic ring	eval userdef.count_nonaromatic_rings(molecule)
aromatic ring	eval userdef.count_aromatic_rings(molecule)
nC-OHside-a	[C;$(C[NX3][CH,CC](=O)),$(CC[NX3][CH,CC](=O)), $(CCC[NX3][CH,CC](=O)),$(CCCC[NX3][CH,CC](=O)), $(CCCCC[NX3][CH,CC](=O))]
carbon number^2	{carbon number}-{nC-OHside-a}-1 if ({amide, primary}{amide, secondary}+{amide, tertiary} > 0) else 0
oxygen count	[#8]
nitrophenol	userdef.count_nitrophenols(molecule, '{aromatic hydroxyl},'{nitro})

Table A5: SMARTS patterns statistics on Wang et al. (2017) data set

Group name	Counts	Min	Max	Mean	Weighted mean
amine, primary	0	0	0	0.000	0.000
amine, secondary	0	0	0	0.000	0.000
amine, tertiary	0	0	0	0.000	0.000
alkane CH	3345	0	26	8.927	9.112
alkene CH	662	0	6	0.340	1.752
aromatic CH	225	0	6	0.211	3.200
carbonyl	1848	0	5	0.890	1.644
hydroxyl (alkyl)	1688	0	4	0.647	1.309
carboxylic acid	264	0	2	0.079	1.023
ester, all	303	0	4	0.107	1.201
ester	303	0	4	0.107	1.201
ether	126	0	2	0.039	1.056
peroxide	185	0	1	0.054	1.000
nitro	119	0	2	0.046	1.328
aromatic hydroxyl	85	0	2	0.033	1.306
hydroperoxide	928	0	1	0.272	1.000
amide	0	0	0	0.000	0.000
nitrate	683	0	2	0.208	1.041
organosulfate	0	0	0	0.000	0.000
carbonylperoxynitrate	279	0	1	0.082	1.000
ketone	1433	0	5	0.588	1.400
aldehyde	907	0	3	0.303	1.139
amide, primary	0	0	0	0.000	0.000
amide, secondary	0	0	0	0.000	0.000
amide, tertiary	0	0	0	0.000	0.000
carbonylperoxyacid	278	0	1	0.081	1.000
peroxy nitrate	3	0	1	0.001	1.000
carbon number	3414	1	15	6.704	6.704
ether, aromatic	0	0	0	0.000	0.000
ether (alicyclic)	167	0	1	0.049	1.000
amine, aromatic	0	0	0	0.000	0.000
nitroester	0	0	0	0.000	0.000
C=C-C=O (non-aromatic)	78	0	4	0.046	2.013
C=C (non-aromatic)	751	0	2	0.225	1.021
nC-OHside-a	0	0	0	0.000	0.000
carbon number[1]	0	0	0	0.000	0.000
oxygen count	3333	0	12	4.052	4.151
aromatic ring	229	0	1	0.067	1.000
non-aromatic ring	808	0	3	0.312	1.319
nitrophenol	43	0	4	0.033	2.605

[1]Carbon number on the acid-side of an amide (asa)

Table A6: SMARTS patterns statistics on GeckoQ data set (Besel et al., 2023)

Group name	Counts	Min	Max	Mean	Weighted mean
amine, primary	0	0	0	0.000000	0.000000
amine, secondary	0	0	0	0.000000	0.000000
amine, tertiary	0	0	0	0.000000	0.000000
alkane CH	30733	0	22	5.487720	5.649139
alkene CH	2542	0	4	0.117932	1.467742
aromatic CH	29	0	4	0.002655	2.896552
carbonyl	25367	0	6	1.348137	1.681358
hydroxyl (alkyl)	18268	0	5	0.823213	1.425662
carboxylic acid	10073	0	3	0.345197	1.084185
ester, all	4145	0	2	0.169043	1.290229
ester	3960	0	2	0.156336	1.248990
ether	0	0	0	0.000000	0.000000
peroxide	8910	0	1	0.281632	1.000000
nitro	4838	0	2	0.155008	1.013642
aromatic hydroxyl	22	0	3	0.001106	1.590909
hydroperoxide	19751	0	4	0.773114	1.238368
amide	0	0	0	0.000000	0.000000
nitrate	17396	0	2	0.665076	1.209531
organosulfate	0	0	0	0.000000	0.000000
carbonylperoxynitrate	7330	0	2	0.241932	1.044202
ketone	18390	0	5	0.828302	1.424959
aldehyde	13657	0	4	0.519834	1.204218
amide, primary	0	0	0	0.000000	0.000000
amide, secondary	0	0	0	0.000000	0.000000
amide, tertiary	0	0	0	0.000000	0.000000
carbonylperoxyacid	7793	0	3	0.259411	1.053125
peroxy nitrate	0	0	0	0.000000	0.000000
carbon number	31637	1	10	6.859879	6.859879
ether, aromatic	0	0	0	0.000000	0.000000
ether (alicyclic)	6551	0	1	0.207068	1.000000
amine, aromatic	0	0	0	0.000000	0.000000
nitroester	396	0	2	0.012707	1.015152
C=C-C=O (non-aromatic)	361	0	2	0.013023	1.141274
C=C (non-aromatic)	2900	0	2	0.091823	1.001724
nC-OHside-a	0	0	0	0.000000	0.000000
carbon number[2]	0	0	0	0.000000	0.000000
oxygen count	31636	0	17	9.933306	9.933620
aromatic ring	29	0	1	0.000917	1.000000
non-aromatic ring	14912	0	2	0.518412	1.099852
nitrophenol	8	0	6	0.000885	3.500000

[2]Carbon number on the acid-side of an amide (asa)

Table A7: Test MAEs of P_{sat} predictions of different versions of BESPE-MACCS on Wang et al. (2017) data set

Training size	**BESPE-MACCS (4)** (Test MAE $\log_{10}$(kPa))	**BESPE-MACCS (5)** (Test MAE $\log_{10}$(kPa))	**BESPE-MACCS (6)** (Test MAE $\log_{10}$(kPa))
500	0.4615	0.4623	0.4633
1000	0.3821	0.3826	0.3859
1500	0.3481	0.3507	0.3532
2000	0.3277	0.3302	0.3324
2500	0.3115	0.3154	0.3173
3000	0.2979	0.3013	0.3029

Table A8: Test MAEs of P_{sat} predictions of different versions of BESPE-MACCS on GeckoQ data set (Besel et al., 2023)

Training size	**BESPE-MACCS (4)** (Test MAE $\log_{10}$(kPa))	**BESPE-MACCS (5)** (Test MAE $\log_{10}$(kPa))	**BESPE-MACCS (6)** (Test MAE $\log_{10}$(kPa))
4633	0.7886	0.7910	0.7909
9267	0.7673	0.7677	0.7675
13900	0.7575	0.7572	0.7570
18534	0.7503	0.7493	0.7490
23167	0.7460	0.7441	0.7435
27801	0.7394	0.7396	0.7389

Table A9: Test MAEs of $K_{W/G}$ predictions of different versions of BESPE-MACCS on Wang et al. (2017) data set

Training size	**BESPE-MACCS (4)** (Test MAE $\log_{10}(1)$)	**BESPE-MACCS (5)** (Test MAE $\log_{10}(1)$)	**BESPE-MACCS (6)** (Test MAE $\log_{10}(1)$)
500	0.6488	0.6555	0.6667
1000	0.5451	0.5499	0.5578
1500	0.4992	0.5037	0.5097
2000	0.4725	0.4780	0.4832
2500	0.4465	0.4523	0.4581
3000	0.4251	0.4312	0.4369

Table A10: Test MAEs of $K_{WIOM/G}$ predictions of different versions of BESPE-MACCS on Wang et al. (2017) data set

Training size	**BESPE-MACCS (4)** (Test MAE $\log_{10}(1)$)	**BESPE-MACCS (5)** (Test MAE $\log_{10}(1)$)	**BESPE-MACCS (6)** (Test MAE $\log_{10}(1)$)
500	0.4125	0.4161	0.4168
1000	0.3538	0.3569	0.3606
1500	0.3201	0.3220	0.3255
2000	0.3033	0.3057	0.3080
2500	0.2880	0.2906	0.2925
3000	0.2758	0.2780	0.2792