28 An Optimized Cloud Computing Method for Extracting Molecular Descriptors

Christos Didachos, Dionisis Panagiotis Kintos,
Manolis Fousteris, Phivos Mylonas, and Andreas Kanavos

Abstract

Extracting molecular descriptors from chemical compounds is an essential preprocessing phase for developing accurate classification models. Supervised machine learning algorithms offer the capability to detect "hidden" patterns that may exist in a large dataset of compounds, which are represented by their molecular descriptors. Assuming that molecules with similar structure tend to share similar physicochemical properties, large chemical libraries can be screened by applying similarity sourcing techniques in order to detect potential bioactive compounds against a molecular target. However, the process of generating these compound features is time-consuming. Our proposed methodology not only employs cloud computing to accelerate the process of extracting molecular descriptors but also introduces an optimized approach to utilize the computational resources in the most efficient way.

Keywords

Molecular descriptors · Computational drug design · Computing performance · Dask · Ligand-based virtual screening · Chemical big data

28.1 Introduction

Machine learning algorithms can play a crucial role in solving problems related with classification [3] and object detection [23]. The precise capability of these algorithms to detect essential motifs in data and classify them in a meaningful way makes them applicable to different scientific fields. Artificial Intelligence and machine learning are highly bound with the demanding process of discovering and developing new drugs [6, 10, 13, 20].

Knowing the structure of a molecular target and a chemical compound is a prerequisite for studying their potential interactions [7]. To this purpose, mathematical approaches and computational methods are being used for the determination of quantitative relationships between the structural features of chemical compounds and their biological activities. This approach, known as Quantitative Structure Activity Relationship (QSAR), could be applied for numerous

C. Didachos
Computer Engineering and Informatics Department,
University of Patras, Patras, Greece
e-mail: christosdidachos@upatras.gr

D. P. Kintos · M. Fousteris
Department of Pharmacy, University of Patras, Patras,
Greece
e-mail: dpkintos@upatras.gr; manolisf@upatras.gr

P. Mylonas · A. Kanavos (✉)
Department of Informatics, Ionian University, Corfu,
Greece
e-mail: fmylonas@ionio.gr; akanavos@ionio.gr

P. Vlamos (ed.), *GeNeDis 2022*, Advances in Experimental Medicine and Biology 1424,
https://doi.org/10.1007/978-3-031-31982-2_28

purposes, such as the prediction of bioactivity of new compounds [9, 14]. However, its drawback is that it is time-consuming and usually requires high availability of computational resources.

Building a QSAR model is frequently based on the use of molecular descriptors. As it has been defined earlier, a molecular descriptor is “the final result of a mathematical procedure which transforms chemical information encoded within a symbolic representation of a molecule into a useful number or the result of some standardized experiment” [19]. The selection of the suitable physicochemical properties as well as theoretical molecular descriptors in a QSAR study is of paramount importance to maximize the accuracy of prediction [18].

Machine learning algorithms are very efficient and effective in recognizing patterns in a given dataset. As a part of a virtual screening study [17, 21], “hidden patterns” in a dataset of compounds, represented by molecular descriptors, could offer valuable information for their classification in sub-classes based on their estimated binding affinity on a molecular target [11]. Usually, virtual screening studies involve large chemical datasets consisting of thousands of compounds. Consequently, extracting meaningful descriptors for datasets of this size may be not only an extremely time-consuming process but also a key preliminary step of a new drug discovery campaign [8].

In our initial methodology [5], we employed Dask framework, which is based on Python for distributed computing and Amazon Web Services (AWS) as the cloud services provider. Our approach successfully accelerated the process of extracting molecular descriptors; in some cases, approximately 73 times faster. The initial approach proved that using a cluster with multiple nodes is more performant for large dataset, but for medium sized dataset a cluster with less nodes is more efficient. However, the exact number of nodes that are required for different data sizes was not clearly defined.

In this study, we aim to prove that the execution time of extracting descriptors is dependent on the size of the compound, which is relative to the length of the compound’s SMILES representation. A dataset with a predefined SMILES length is used as a template. We initially extract the descriptors for a different number of rows of this template using a different number of nodes. The required time for the cluster formation and the extraction of the descriptors is then calculated for various combinations of data sizes and cluster sizes. These calculations are used as a template for the identification of the suitable, based on their performance, number of nodes that should be used for a dataset given different sizes of dataset. It has been proven that our new optimized approach can maximize the performance of the initially proposed process.

28.2 Material and Methods

The molecular weight is a one-dimensional descriptor which describes a compound. Similarly, the Balaban index J and Bertz’s complexity index are presented. Both of these indexes are transformations of the available knowledge which is related to the structure of a compound. The Balaban index J [2] is a descriptor which is based on graph theory. The standard distance matrix of D of a graph G is a matrix $(D)_{ij}$ as described below:

$$(D)_{ij} = \begin{cases} \ell_{ij}, & \text{if} \quad i \neq j \\ 0, & \text{if} \quad i = j \end{cases} \tag{28.1}$$

where ℓ_{ij} is the shorter path which can be described as the minimum number of edges between the vertices i and j. Given a four-node cycle C_4, as it is illustrated in Fig. 28.1, the distance matrix D can be defined as the following matrix:

$$\begin{bmatrix} 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \end{bmatrix}$$

The Balaban index J is defined as

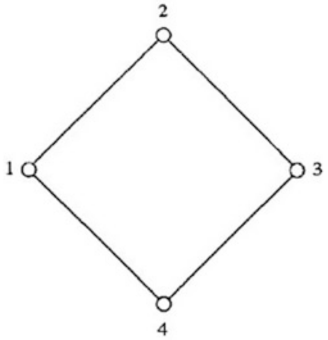

Fig. 28.1 Labeled four-membered cycle $C4$

$$J = \frac{E}{\mu + 1} \sum_{edges} (d_i d_j)^{-1/2} \qquad (28.2)$$

where E is the number of edges in a graph G, μ is the cyclomatic number of G, and d_i is the distance sum of vertex i (it is a sum of all entries in the ith row or column of the distance matrix). The cyclomatic number μ of a polycyclic graph G is equal to the minimum number of edges necessary to be removed from G in order to convert G into the related acyclic graph. As an example, the Balaban index J for the graph illustrated in Fig. 28.1 is equal to 2 [1].

Bertz's complexity index is a topological index which tries to quantify "complexity" of molecules. It consists of a sum of two terms: the first one representing the complexity of the bonding and the second representing the complexity of the distribution of heteroatoms [4, 15]. Bretz's molecular complexity index $C(n)$ is defined as follows:

$$C(n) = 2n \log_2 n - \sum n_i \log_2 n_i \qquad (28.3)$$

where n denotes a graph invariant and n_i is the cardinal number of the ith set of equivalent structural elements on which the invariant is defined. The summation goes over all sets of equivalent structural elements.

Additionally, molecular fingerprints are an essential category of descriptors. Molecular fingerprints try to encode a molecular structure. This structure is represented as a vector. Actually it is a sequence of binary digits which represents the 3D structure of the compound.

28.3 Implementation

28.3.1 Dataset

The dataset that was used in our study is a pandas dataframe of 80, 000 compounds. There are two different columns: the first one is the PubChem ID of the compound and the second is the SMILES representation of the compound [22]. The average SMILES length is 59.5 characters. Eight different datasets of 20, 001 rows were used at the phase of searching whether there is a relationship between the SMILES length and the execution time of extracting descriptors. For that purpose, each dataset consisted of the same compound. The comparison took place between two datasets having 25 and 118 SMILES length and two datasets of 26 and 112 SMILES length, respectively.

28.3.2 Dask Framework

Dask is a powerful Python framework which offers a pythonic way to execute code in parallel [16]. The code can be executed in multiple CPUs of a single machine or even in multiple nodes of a cloud cluster. Amazon Web Services were used as a cloud provider to scale up our computations and the coiled framework that enables the setting up of a cloud cluster.

28.3.3 Methodology

The extraction of molecular descriptors was achieved using the RDKIT framework.[1] RDKIT is a Python scientific framework related to computational chemistry. The proposed method generated four different groups–categories of descriptors (mostly 1D, 2D, 3D, Morgan Fingerprints) and saved the output in CSV format as a binary file (.pkl).

[1] https://zenodo.org/record/3732262.

In our approach, we calculated the time which is needed to extract molecular descriptors using a large combination of different cluster sizes and different dataset sizes. In more detail, we computed 170 different execution times in terms of extracting molecular descriptors using datasets having varying sizes, e.g., equal to 5, 000, 10, 000, 15, 000, 20, 000, 25, 000, 30, 000, 35, 000, 40, 000, 45, 000, 50, 000 compounds.

The different "architectures" of the cloud cluster were set equal to 25, 35, 45, 55, 65, 75, 85, 95, 105, 115, 125, 135, 145, 155, 165, 175, 185 nodes. Each of these datasets consisted of the same SMILES with length equal to 57 characters as this was the median SMILES length of the initial dataset. We had to take into consideration the length of the SMILES as we proved that there is a relationship between the SMILES length and the execution time of generating molecular descriptors (more time is required for larger compounds).

It is usually known that the SMILES length of a drug is between 20 and 90 characters [12]. As a result, all these 170 calculations were used as a template. Our proposed method used this template to estimate the best number of nodes related to the size of the dataset. Finally, we compared the execution time needed using the proposed method (using the number of nodes based on the template) and the best performance, which was observed from the initial approach in [5]. Our proposed method was proved to be the optimized one as it overpasses the initial approach. Based on the template, it offers the capability to estimate the best number of nodes related to the size of the dataset and the median number of SMILES length.

28.4 Results

As we can observe in Fig. 28.2, the time needed to extract molecular descriptors is larger for compounds with bigger SMILES length compared to compounds with smaller SMILES length. Although in some cases the SMILES length is the same, each dataset consists of different chemical compounds.

In Tables 28.1 and 28.2, the execution time is displayed for a variety of nodes for different cluster sizes. These calculations are used as a template in order to estimate the number of nodes that could be the most performant choice. The template that is displayed in these tables is compatible with the results of our previous study [5]. In more detail, when the number of rows in the dataset is relatively small, a cluster with a small number of nodes is the most efficient option. However, when the number of compounds is being increased, the solution of using more cloud nodes tends to be the most efficient. For a given number of compounds, our approach proposes a number of nodes that could have the maximum impact of extracting the molecular descriptors as fast as possible.

In our initial approach [5], for a dataset of 10, 000 compounds and a cluster of 180 nodes, the required time to extract molecular descriptors was 198.9 seconds. The proposed method (using the template) proposed the usage of 145 nodes, and the result of this cloud infrastructure was to extract the descriptors in 152.2 seconds. Sequentially, for a dataset of 20, 000 compounds, the initial method used a cluster of 100 nodes, and the execution time was 263.6 seconds. However, the proposed approach, for the same dataset, used a cluster of 145 nodes, and the execution time was 183.1 seconds. Finally, for a dataset of 50, 000 compounds, the initial method used a cluster of 180 nodes, and the execution time was 445.3 seconds. The proposed method used a cluster of 185 nodes, and the execution time was 307 seconds.

In all cases, the use of the template in our proposed method offered the capability to use a number of nodes that tend to extract the molecular descriptors much faster compared to the initial approach. All these comparisons are illustrated in Fig. 28.3.

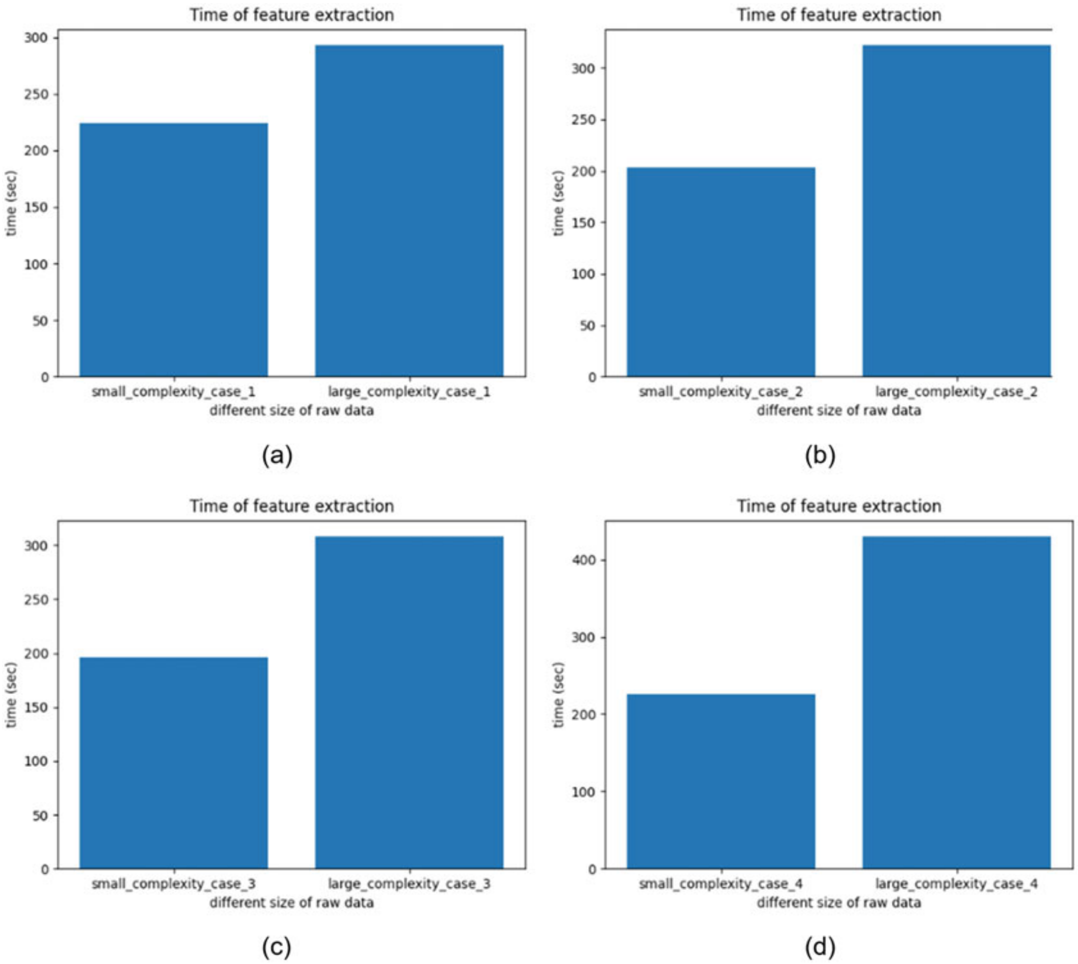

Fig. 28.2 Both datasets consist of 20,001 compounds and SMILES length equal to (**a**) first dataset = 25 characters, second dataset = 118 characters (**b**) first dataset = 26 characters, second dataset = 112 characters (**c**) first dataset = 25 characters, second dataset = 118 characters (although the SMILES length is the same as in case (**a**), each dataset consists of a different compound) (**d**) first dataset = 26 characters, second dataset = 112 characters (although the SMILES length is the same as in case (**b**), each dataset consists of a different compound)

28.5 Conclusions and Future Work

Our initial approach [5] is highly efficient in extracting RDKIT molecular descriptors. This offers researchers the capability to handle even a bigger number of compounds in computational chemistry approaches [21]. More to the point, the proposed method offers an optimized approach that extracts molecular descriptors in a more efficient way, e.g., in terms of time that is required.

Regarding future work, the proposed methodology could be enriched using a larger number of templates for different SMILES lengths. This could give the capability to the researchers to extract molecular descriptors of a dataset based on the median SMILES length of the dataset. As the execution time of extracting molecular descriptors varies regarding the SMILES length, the usage of a large number of templates could offer the capability to extract descriptors using the most performant cloud infrastructure based on SMILES length.

Table 28.1 Execution time (sec) for a variety of cloud infrastructures 1/2

Number of nodes	5K rows	10K rows	15K rows	20K rows	25K rows
25	169,55	211,74	371,77	476,84	568,92
35	171,38	212,75	288,31	288,19	397,3
45	170,92	211,78	209,89	250,09	382,3
55	151,84	199,87	200,64	239,43	296,24
65	133,91	187,78	180,44	238,09	251,37
75	130,62	165,74	183,49	236,41	250,26
85	134	155,7	215,73	238,37	254,56
95	137,81	158,37	189,36	217,29	255,59
105	142,98	155,15	176,36	198,76	253,42
115	147,65	155,93	179,09	195,91	253,2
125	152,56	152,81	198,74	192,62	242,59
135	154,38	153,03	193,77	187,45	236,16
145	140,89	152,2	184,07	183,19	228,92
155	146,92	161,05	186,25	201,25	229,86
165	145,26	170,19	178,08	211,94	222,94
175	149,1	170,79	176,77	216,76	214,26
185	159,8	170,61	178,14	220,62	212,91

Table 28.2 Execution time (sec) for a variety of cloud infrastructures 2/2

Number of nodes	30K rows	35K rows	40K rows	45K rows	50K rows
25	469,3	829,06	607,89	1043	794,42
35	467,2	533	500,8	637,9	706,5
45	458,55	361,52	395,54	414,51	681,63
55	389,11	358,62	389,12	413,23	580,6
65	362,46	355,71	386,91	408,91	435,98
75	290,99	314,75	368,41	407,4	429,4
85	259,93	297,48	307,4	409,1	383,37
95	256,14	295,27	295,36	396,36	357,57
105	246,72	269,87	280,3	378,58	349,17
115	249,75	269,88	279,69	378,69	343,35
125	249,03	265,58	263,66	382,69	328,69
135	251,47	265,41	360,07	315,47	319,07
145	248,68	263,3	251,3	297,45	315,45
155	253,3	258,86	256,22	313,12	319,97
165	242,96	239,73	253,36	337,2	315,7
175	242,88	239,73	253,56	332,46	309,42
185	245,25	239,94	256,17	332,04	307,57

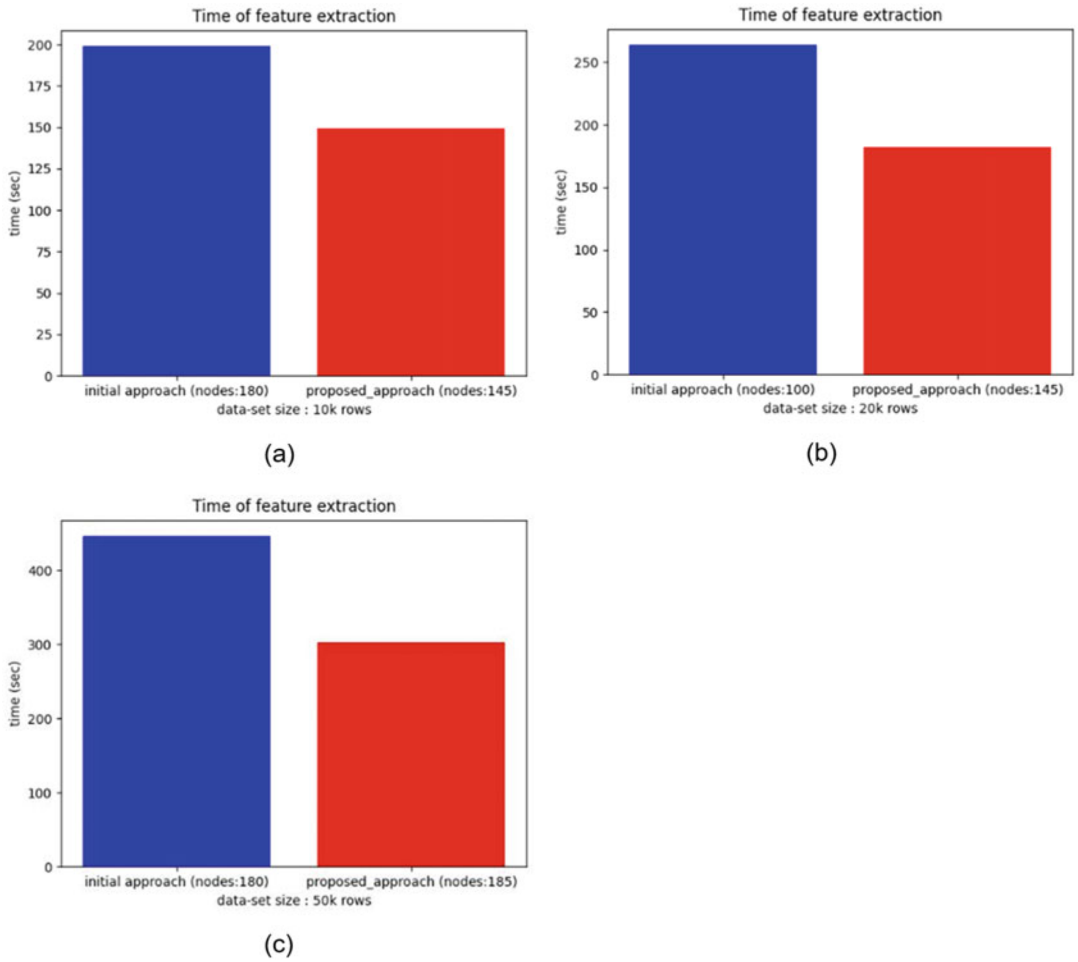

Fig. 28.3 Initial vs Proposed Approach with characteristics: (**a**) Dataset size = 10K rows, Initial Approach = 180 nodes, Proposed Approach = 145 nodes (**b**) Dataset size = 20K rows, Initial Approach = 100 nodes, Proposed Approach = 145 nodes (**c**) Dataset size = 50K rows, Initial Approach = 180 nodes, Proposed Approach = 185 nodes

References

1. Babić D, Klein D, Lukovits I, Nikolić S, Trinajstić N (2002) Resistance-distance matrix: A computational algorithm and its application. International Journal of Quantum Chemistry 90(1):166–176
2. Balaban AT (1982) Highly discriminating distance-based topological index. Chemical Physics Letters 89(5):399–404
3. Bazan JG, Nguyen HS, Nguyen SH, Synak P, Wróblewski J (2000) Rough set algorithms in classification problem. In: Rough Set Methods and Applications, pp 49–88
4. Bertz SH (1981) The first general index of molecular complexity. Journal of the American Chemical Society 103(12):3599–3601
5. Didachos C, Kintos DP, Fousteris M, Gerogiannis VC, Son LH, Kanavos A (2022) A cloud-based distributed computing approach for extracting molecular descriptors. In: 6th International Conference on Algorithms, Computing and Systems (ICACS)
6. Hessler G, Baringhaus KH (2018) Artificial intelligence in drug design. Molecules 23(10):2520
7. Hwang H, Dey F, Petrey D, Honig B (2017) Structure-based prediction of ligand–protein interactions on a genome-wide scale. Proceedings of the National Academy of Sciences 114(52):13685–13690

8. Kombo DC, Tallapragada K, Jain R, Chewning J, Mazurov AA, Speake JD, Hauser TA, Toler S (2013) 3d molecular descriptors important for clinical success. Journal of Chemical Information and Modeling 53(2):327–342
9. Kubinyi H (1997) QSAR and 3D QSAR in drug design part 1: Methodology. Drug Discovery Today 2(11):457–467
10. Lavecchia A (2015) Machine-learning approaches in drug discovery: Methods and applications. Drug Discovery Today 20(3):318–331
11. Lionta E, Spyrou G, Vassilatis DK, Cournia Z (2014) Structure-based virtual screening for drug discovery: Principles, applications and recent advances. Current Topics in Medicinal Chemistry 14(16):1923–1938
12. Liu P, Li H, Li S, Leung KS (2019) Improving prediction of phenotypic drug response on cancer cell lines using deep convolutional network. BMC Bioinformatics 20(1):1–14
13. Mak KK, Pichika MR (2019) Artificial intelligence in drug development: Present status and future prospects. Drug Discovery Today 24(3):773–780
14. Mauri A, Consonni V, Todeschini R (2017) Molecular descriptors. In: Handbook of Computational Chemistry, pp 2065–2093
15. Randić M, Plavšić D (2002) On the concept of molecular complexity. Croatica Chemica Acta 75(1): 107–116
16. Rocklin M (2015) Dask: Parallel computation with blocked algorithms and task scheduling. In: 14th Python in Science Conference, 130–136
17. Shoichet BK (2004) Virtual screening of chemical libraries. Nature 432(7019):862–865
18. Stahura FL, Bajorath J (2005) New methodologies for ligand-based virtual screening. Current Pharmaceutical Design 11(9):1189–1202
19. Todeschini R, Consonni V (2010) Molecular descriptors. Recent Advances in QSAR Studies pp 29–102
20. Vamathevan J, Clark D, Czodrowski P, Dunham I, Ferran E, Lee G, Li B, Madabhushi A, Shah P, Spitzer M, Zhao S (2019) Applications of machine learning in drug discovery and development. Nature Reviews Drug Discovery 18(6):463–477
21. Walters WP, Stahl MT, Murcko MA (1998) Virtual screening - an overview. Drug Discovery Today 3(4): 160–178
22. Weininger D (1988) Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. Journal of Chemical Information and Computer Sciences 28(1):31–36
23. Zou Z, Shi Z, Guo Y, Ye J (2019) Object detection in 20 years: A survey. CoRR abs/1905.05055