Knodle: A Support Vector Machines-Based Automatic Perception of Organic Molecules from 3D Coordinates

Maria Kadukova, Sergei Grudinin

▶ To cite this version:

Maria Kadukova, Sergei Grudinin. Knodle: A Support Vector Machines-Based Automatic Perception of Organic Molecules from 3D Coordinates. Journal of Chemical Information and Modeling, 2016, 56 (8), pp.1410-1419. 10.1021/acs.jcim.5b00512 . hal-01381010

HAL Id: hal-01381010

https://inria.hal.science/hal-01381010

Submitted on 8 Nov 2016

Knodle, a Support Vector Machines-Based Automatic Perception of Organic Molecules from 3D Coordinates

Maria Kadukova[1] and Sergei Grudinin*[2,3,4]

[1]Moscow Institute of Physics and Technology, Dolgoprudniy, Russia
[2]Univ. Grenoble Alpes, LJK, F-38000 Grenoble, France
[3]CNRS, LJK, F-38000 Grenoble, France
[4]Inria, France

Here we address the problem of the assignment of atom types and bond orders in low molecular weight compounds. For this purpose, we have developed a prediction model based on nonlinear Support Vector Machines (SVM), implemented in a KNOwledge-Driven Ligand Extractor called *Knodle*, a software library for the recognition of atomic types, hybridization states and bond orders in the structures of small molecules. We trained the model using an excessive amount of structural data collected from the PDBbindCN database. Accuracy of the results and the running time of our method is comparable with other popular methods, such as NAOMI, fconv, and I-interpret. On the popular Labute's benchmark set consisting of 179 protein-ligand complexes, *Knodle* makes five to six perception errors, NAOMI makes seven errors, I-interpret makes nine errors, and fconv makes thirteen errors. On a larger set of 3,000 protein-ligand structures collected from the PDBBindCN general data set (v2014), *Knodle* and NAOMI have a comparable accuracy of approximately 3.9 % and 4.7% of errors, I-interpret made 6.0 % of errors, while fconv produced approximately 12.8 % of errors. On a more general set of 332,974 entries collected from the Ligand Expo database, *Knodle* made 4.5 % of errors. Overall, our study demonstrates the efficiency and robustness of nonlinear SVM in structure perception tasks. *Knodle* is available at https://team.inria.fr/nano-d/software/Knodle.

Introduction

Information about chemical properties of atoms and bonds between them is very important for many computational methods in structural biology, medicine and bioinformatics. For example, the proper assignment of atom types and bond orders is crucial for the success of virtual screening methods in drug design [26] as well as for the performance of some knowledge-based potentials [19]. Experimentally determined molecular structures initially contain information only about atomic coordinates and their chemical elements. It is, generally, rather simple to describe chemical properties of nucleic and amino acids, but for many other molecules this task may become very challenging. Low molecular weight compounds (from now on called ligands), which are very important for the pharmaceutical industry, are among these hard-classified molecules, for example.

Several algorithms for the determination of chemical properties and bond orders have been proposed in the past years. The earliest ones were based on simple geometric considerations including bonds lengths and valence angles [4]. Later,

*sergei.grudinin@inria.fr

some functional groups were taken into account [13], as well as hybridization states and atomic charges [16, 28, 23, 27]. To make the algorithm less dependent on possible errors in the experimentally determined structures, several approaches have been proposed. These include the maximum weighted match search [15, 20], a search of a Lewis structure with the minimal charge on each atom by means of either linear programming [12], or by the maximization [22] or minimization [25, 9] of a special scoring function, etc.

All existing methods share the following general scheme of molecular structure perception. First, initial information about the valence states and the atomic hybridizations is perceived from the molecular geometry. Then, some bond orders are predicted. Finally, the results of the previous steps are refined using the functional groups patterns. In the end, atomic types can be set and bond orders are assigned. Although molecular geometry contains more information than just the bond lengths and valence angles, most often only these data are used for the perception of chemical molecular properties. Other geometrical data, such as the values of the torsion angles or the triple product of the bonds of an atom, which defines its space geometry, may also be used, but generally do not constitute the core of perception algorithms.

One particular exception is the NAOMI perception model [22], where different bond lengths, valence angles and, less commonly, triple products and torsion angles are used in the perception. More precisely, each geometrical parameter is scored with a linear function, and the obtained results are then substituted into the general scoring function. A popular fconv method is based on the maximum-weighted matching search [20]. Here, bond orders are determined relying on statistical occurrence of bond lengths and valence angles for different types of atomic hybridizations. The latter are predicted by maximizing the value of a scoring function that describes bond orders in the molecular system. Another popular I-interpret method [28] uses a more simplistic approach. More precisely, first, for each hybridization state, bounds for some valence angles are precomputed from the training set of data. Then, these are compared with the valence angle values in the given structure and atomic hybridizations are assigned accordingly.

Here, we introduce a new algorithm for the perception of chemical properties of a molecule from its 3D structure using a machine learning approach. More precisely, we use the multiclass support vector machine (SVM) method with a nonlinear kernel function to off-line train the models for the perception of atomic hybridization states and bond orders. The feature space for the SVM model is constituted by almost all available geometrical descriptors that also contain some chemical information. One of the main advantages of using SVM is that it is a convex optimization method with a unique optimal solution. Conversely, the commonly used methods for the minimization of a scoring function often roll the solution to one of the local minima.

We implemented our algorithm in *Knodle*, a KNOwledge-Driven Ligand Extractor – a software library for the perception of bond orders and hybridization states of atoms of small molecules. It has been primarily designed to amend the atom type recognition that can be used in a knowledge-based potential or for virtual screening applications. In the current implementation, *Knodle* converts a ligand molecule from the Protein Data Bank pdb format file to either a mol2 format file, a sdf format file, or a file with an extended atomic parametrization, with types similar to those used in the fconv library.

Method

Overview

The process of bond and atom properties perception is divided into several steps. At the beginning, only information about the coordinates and chemical elements for each atom is available. Figure 1 summarizes the general work-flow of the algorithm. These stages are discussed in more detail in the following sections.

Connectivity and Rings

In the first step of our method, covalent bonds between atoms are determined from their coordinates. For each atom, a list of its neighbors is created, where the neighbors are defined as all the atoms

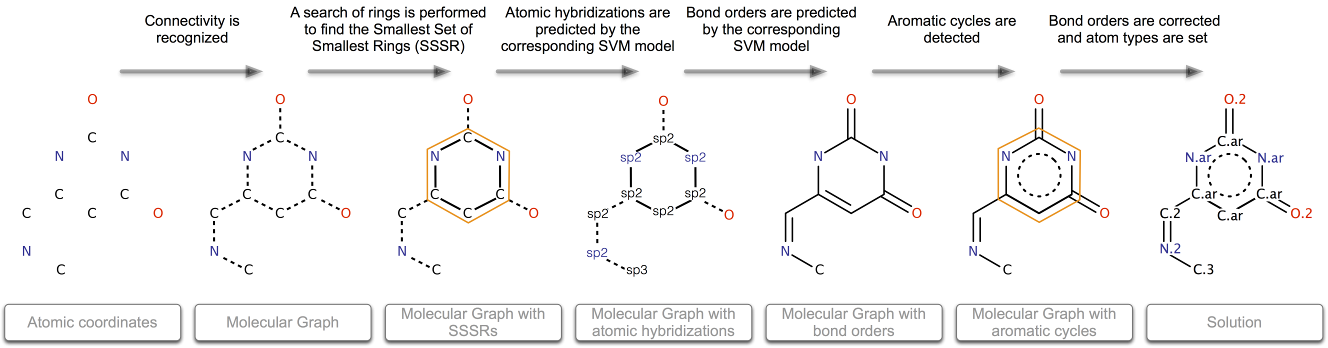

Connectivity is recognized
A search of rings is performed to find the Smallest Set of Smallest Rings (SSSR)
Atomic hybridizations are predicted by the corresponding SVM model
Bond orders are predicted by the corresponding SVM model
Aromatic cycles are detected
Bond orders are corrected and atom types are set
O
C
N
C
O
sp2
sp3
N.ar
C.ar
O.2
C.2
N.2
C.3
Atomic coordinates
Molecular Graph
Molecular Graph with SSSRs
Molecular Graph with atomic hybridizations
Molecular Graph with bond orders
Molecular Graph with aromatic cycles
Solution

Figure 1: Schematic workflow of the *Knodle* algorithm.

within the cutoff distance of $r_c = 2.5$ Å from the given atom. To do so computationally fast, we use the cell linked-list algorithm [1, 2]. More precisely, we split the molecular system into linked cells, each of linear size r_c, and associate each atom with its parent cell where it is contained. To find all the neighbors of an atom, we then traverse its parent cell along with the 26 direct neighbor cells and select those atoms within the cells that are closer to the given atom than r_c. All the found neighbor atoms are stored in a list of neighbors for further analysis. We chose the cutoff distance parameter of $r_c = 2.5$ Å according to the statistics of covalent bonds in the PDBeChem database [10], which does not contain any bonds, excluding the metallic ones, beyond this distance.

A decision as to whether a bond of length l between two atoms exists is taken based on the probability of finding such a distance l in the statistical data for these two atoms. We collected this data from the PDBeChem [10] and PDBbind [18] databases. If we find more bonds with nonzero probability than an atom can form, the bonds with the lowest probability are rejected. Unlike common pairs of atoms, such as C-C or C-N, data for S-Se or P-S pairs contain statistical gaps. We use a linear interpolation where appropriate to fix the gaps. Some ligands in the databases contain unusually long bonds. For example, 1kpm ligand from the PDB-Bind data set (vitamin E) contains very long C-C bonds of $l \simeq 2.019$ Å. This distance is equal to the distance between the opposite carbon atoms in four- and five-membered rings. Therefore, we added a separate verification for such long bonds.

The ring detection task is divided into several steps. First, we build the molecular graph – a graph whose vertices correspond to the atoms and whose edges correspond to the covalent bonds of the molecular structure. Then, we find articulation points of the molecular graph, i.e. the vertices that disconnect the graph if they are removed, and extract a set of disconnected cyclic components from it. To do so, we used Tarjan's depth-first search (DFS) algorithm for the articulation points search [21] as it was implemented in our molecular graph library [3]. Each connected component is then analyzed separately. For this purpose we implemented the algorithm for the smallest set of smallest rings (SSSR) search by Lee et al. [17]. This algorithm detects rings using two path-included distance matrices P_1 and P_2, such that any ring can be represented as a union of two paths taken from P_1 and P_2. More precisely, P_1 contains paths of the minimum length l between given atoms 1 and 2, while P_2 contains paths of length $l + 1$. A ring is then a union of a path from P_1 and a second path taken either from P_1 or P_2. Excessive rings are then removed to create the SSSR. Although most of the ligands form simple ring sets that are easy to detect, there are some complex cases. An example of such molecules is indolocarbazoles – organic compounds with five conjugated aromatic rings, which are considered to be a prospective target for anticancer research. In one of the indolocarbazoles examples, stautosporine (pdb code 1xjd, Fig. 2), fconv failed to recognize aromaticity of the central ring, presumably due to the problems with identifying its sequence of atoms as a ring. *Knodle*, NAOMI and I-interpret methods experienced no difficulties when dealing with such cases.

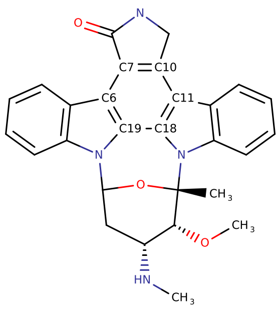


Figure 2: Staurosporine molecule. Ring C6-C7-C10-C11-C18-C19 is not detected by fconv.

Classification

The fundamental goal of this study is to demonstrate that machine-learning methods are suitable for some classification tasks in structural chemistry. More precisely, our practical goal was to predict atomic hybridization states and bond orders in an experimentally determined molecular structure starting from a set of its geometrical descriptors. For this purpose, we used support vector machines (SVM) [24], a convex optimization method, which has found many applications in text, speech and image recognition, bioinformatics, medical data analysis and various other fields due to its robustness and high performance. To solve the optimization problem, we used the LIBSVM library [1] – the multi-platform software for SVM classification [6]. To predict atomic hybridization states and bond orders, we first split all the atoms and all the bonds into several *classification groups* that have corresponding SVM models. These groups are listed in Table 1. An atom is assigned to one of the ten classification groups depending on the number of its heavy neighbors and their chemical elements, its own chemical element, and, specifically for the terminal oxygen atoms, whether it is attached to a ring. One classification group contains very particular oxygen atoms attached to an atom that is a member of a planar ring, while the other classification groups are more general (see Table 1). We assign a bond to one of the four classification groups depending on the chemical symbols and the number of heavy neighbors of the bonding atoms. Table 1 also lists cross-validation accuracies for each classification group, i.e. the ratio of correctly predicted hybridizations and bond orders measured on the training set after a 5-fold cross validation. These numbers demonstrate the *relative* quality of the chosen descriptors for each model. We can see that SVM predict bond orders less accurately compared to the hybridization of the atoms. However, these are corrected in the later steps of our method (see Fig. 1).

The prediction of hybridization states and bond orders then passes through the following steps. First, we detect the classification groups of the atoms, which specify their available descriptors. Table 2 lists all the atomic descriptors that we use to determine atomic hybridization states. For most of the atoms, the optimization cost function depends only on their local geometrical descriptors. We make an exception for the terminal oxygen and nitrogen atoms, which have only a few available descriptors. We determine hybridization states of each atom from these two classification groups after all other atoms have been processed and their hybridizations have become known. It allows us to extend the set of descriptors for such terminal atoms with the hybridization of their only heavy neighbor and thus to use this additional information for more precise chemical perception. We achieved the best results using atomic hybridization states divided into the following four classes: sp1, sp2, sp3 hybridizations and an atom in an aromatic structure. We should note that *hybridization* is a discrete descriptor that takes one of four values in the set {1, 2, 3, 4}, with each value corresponding to one of the aforementioned classes. For the bond orders determination, we use a similar SVM-based procedure. Table 3 lists descriptors that we use to determine bond orders. We analyze all the bonds separately and correct their orders with respect to each other on the further steps of the algorithm. We describe the bonds with five classes: single, double, triple, aromatic and amide bonds.

LIBSVM provides an implementation of a one-vs-one approach for a multiclass classification. In this approach, for each pair of classes, SVM build a model based on a binary classifier. To categorize data points from the training set as members of one of the classes, SVM solve the minimization

[1] LIBSVM-3.20

Atom and bond classification groups	#classes	#features	Training set size	Cross-validation accuracy, %	Approximated/polynomial kernel version
Carbon atoms having one heavy neighbor	3	3	3804	99.63	-
Nitrogen atoms having one heavy neighbor	3	5	8186	99.95	-
sulfur atoms having one heavy neighbor	2	3	188	88.24	-
Oxygen atoms having one heavy neighbor, which is a part of a ring	2	3	4011	94.52	-
All the remaining oxygen atoms bonded to a single heavy atom	2	6	27402	97.48	-
Carbon atoms having two heavy neighbors	4	8	24750	97.62	polynomial
Nitrogen atoms having two heavy neighbors	4	7	9943	91.24	rbf approximation
Carbon atoms having three heavy neighbors	3	13	22928	96.33	polynomial
Nitrogen atoms having three heavy neighbors	4	12	6785	96.18	-
Sulfur atoms having three heavy neighbors	2	9	48	93.75	-
Bonds with terminal atoms (one of the atoms has the only heavy neighbor)	5	5	1915	85.6	-
C-C bonds	4	3	1906	89.72	-
C-N bonds	5	3	1622	78.73	rbf approximation
Other bonds	4	5	14180	84.65	-

Table 1: Classification groups for atoms and bonds that have the corresponding SVM models, number of classes, feature space size, the training set size, the training results, and the SVM kernels used in our model. 'Rbf approximation' stands for the second-order Maclaurin series approximation of the radial basis kernel. 'Polynomial' stands for the third-order polynomial kernel. The 'Cross-validation accuracy' column gives the ratio of correctly predicted hybridizations and bond orders measured on the training set after a 5-fold cross validation.

Descriptor	Atoms of the involved SVM models
electronegativity	all except element-specific classifiers
bonded atom electronegativity · bond length (sorted by the bond length value)	all
bond length (sorted by the bond length value)	all
the minimum value among the angles that the only neighbor of this atom forms with its neighbors	all with con = 1
is in a ring	all with con = 2
is in a planar ring	all with con = 2, 3
valence angles (sorted by the valence angle value)	all with con = 2, C and N atoms with con = 3
the maximum value among torsion angles with the involvement of this atom	all with con = 2
triple product of bond vectors	all with con = 3
is in a 5-membered heteroring	C atoms with con = 3
hybridization state of the neighbor	O atoms with con = 1, if no rings are attached to the connected atom; all N atoms with con = 1
is a possible part of amide	O with con = 1

Table 2: Descriptors for the prediction of hybridization states. Each descriptor is applicable to one or more atom classification groups. An atom is assigned to a classification group according to the number of its heavy neighbors and its chemical element. Here, "con" stands for the "connectivity" parameter of an atom – the number of its heavy neighbors.

Descriptor	Applicable to
electronegativity$_1$; electronegativity$_2$	all bonds
hybridization$_1$; hybridization$_2$	all bonds
bond length	all bonds
the maximum value of torsion angles formed with two atoms involved in the bond	if both atoms are bonded with more than 1 heavy atom

Table 3: Descriptors for the prediction of bond orders. Each descriptor is applicable to one or more bond classification groups. Indices 1 and 2 correspond to the first and the second atoms involved in the bond.

problem, which consists of minimizing the empirical risk with a regularization penalty as

$$\min_{\mathbf{w},b} \quad \frac{\lambda}{2}||\mathbf{w}||^2 + \sum_i \xi_i \qquad (1)$$

$$s.t. \quad y_i(\mathbf{w}\cdot\mathbf{x_i} + b) \geq 1 - \xi_i \qquad (2)$$

$$\xi_i \geq 0, \qquad (3)$$

where $\mathbf{w}$ is a weight matrix stacking the weight vectors corresponding to each one-vs-one subproblem, b is the offset parameter, $X = \{(\mathbf{x}_i, y_i), i = 1, ..., N\}$ is the training set of labeled geometrical features, $\sum_i \xi_i$ is the hinge-loss empirical risk function, and $||\mathbf{w}||^2$ is the L_2-norm regularization penalty. The regularization parameter λ determines the importance of the regularization term with respect to the empirical risk. We have also defined the inner product in the feature space through the radial basis kernel function as

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma||\mathbf{x}_i - \mathbf{x}_j||^2),\ \gamma > 0, \quad (4)$$

with parameter γ describing the support of the kernel. We determined parameters λ and γ using a 5-fold cross-validation procedure. We also tested several other kernel functions. However, we achieved the best classification results with the radial basis kernel function. We should note that the calculation of the radial basis kernel functions for all support vectors of a given SVM model is computationally expensive. Therefore, we speeded up the classification procedure in the following way. First, SVM models with small γ parameter ($\gamma \leq 0.01$) were approximated with the second-order Maclaurin series in the same way as was recently described by Claesen et al. [7]. Unfortunately, most of the SVM models required $\gamma > 1$ for proper accuracy, making such approximation impossible. For two of these models, however, a replacement of the radial basis kernel function with a third-order polynomial resulted in only a slight decrease in accuracy alongside drastic improvement in time performance. More precisely, in SVM models that describe carbon atoms connected to 2 or 3 heavy atoms, we use the following third-order polynomial kernel function,

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma\langle\mathbf{x}_i\mathbf{x}_j\rangle + c_0)^3,\ \gamma > 0\ , \quad (5)$$

where we determined the parameters γ and c_0 trough a five-fold cross-validation procedure. In the released version of our method, we let the user choose between the default fast approximate SVM kernels and the optional slow exact ones.

After the 5-fold cross-validation procedure, the training results vary from ≈85% for the bond order prediction to ≈95 – 100% for the prediction of the hybridization state. Although the results achieved by SVM are further analyzed to correct bond orders and assign atomic types in accordance with functional groups (see Fig. 1), atomic hybridization states are determined accurately enough at this stage. We should note that our model of hybridization states and bond order determination is very simplistic and requires correct molecular geometry as the input of the algorithm.

We did not take into consideration hydrogen atoms, as they are rarely determined experimentally and are often not included in protein-ligand complex structures. Most of the hydrogen atoms in the training and the test sets were added after the structure determination, thus information about their geometry is not representative for the SVM model training.

We should emphasize that the machine-learning step is performed off-line. That is, *Knodle* classifies a feature vector $\mathbf{x}$ by computing the corresponding set of scores for each pair of classes kl as

$$score_{kl}(\mathbf{x}) = \sum_{i\ \in\ \text{support vectors}\ (kl)} c_{i,kl}K(\mathbf{x}, \mathbf{x}_{i,kl}) - \rho_{kl}, \quad (6)$$

with support vectors $\mathbf{x}_{i,kl}$, kernel weights $c_{i,kl}$, margins ρ_{kl}, and the kernel parameters determined offline upon the training stage. Having C classes, this operation is repeated $C \times (C-1)/2$ times and then the class with the largest number of positive scores is assigned to the feature vector $\mathbf{x}$. We trained all the SVM models using the PDBBindCN general data set (v2014)[18] consisting of 10,605 ligands, which was *randomly* split into the training set (7,605 ligands) and the test set (3,000 ligands). We also extended the training set with all sulfur-containing ligands from a newer release of PDBBindCN general data set (v2015) that did not intersect with the test set. These contain 236 additional ligands.

Identification of Aromaticity

After the determination of the hybridization states and bond orders, we proceed to the determination of aromaticity (see Fig. 1). For this, we start with the processing of all the planar ring candidates with fewer than four bonded heavy atoms to get information about their hybridization, the number of aromatic bonds this atom is involved in, whether an atom can be a lone pair carrier and whether it has a bonded exocyclic oxygen or sulfur sp2 atoms. This information is analyzed and either the number of π-electrons in the ring is calculated, or the ring is

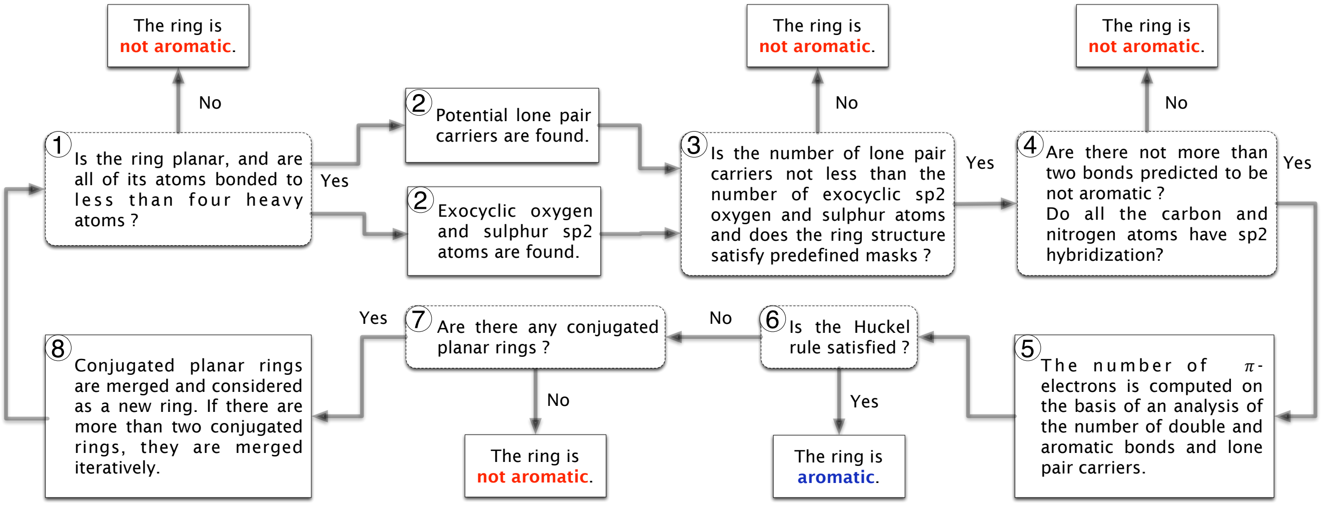

The ring is not aromatic.
No
1 Is the ring planar, and are all of its atoms bonded to less than four heavy atoms ?
Yes
2 Potential lone pair carriers are found.
2 Exocyclic oxygen and sulphur sp2 atoms are found.
The ring is not aromatic.
No
3 Is the number of lone pair carriers not less than the number of exocyclic sp2 oxygen and sulphur atoms and does the ring structure satisfy predefined masks ?
Yes
The ring is not aromatic.
No
4 Are there not more than two bonds predicted to be not aromatic ? Do all the carbon and nitrogen atoms have sp2 hybridization?
Yes
8 Conjugated planar rings are merged and considered as a new ring. If there are more than two conjugated rings, they are merged iteratively.
Yes
7 Are there any conjugated planar rings ?
No
The ring is not aromatic.
No
6 Is the Huckel rule satisfied ?
Yes
The ring is aromatic.
5 The number of π-electrons is computed on the basis of an analysis of the number of double and aromatic bonds and lone pair carriers.

Figure 3: Aromaticity detection workflow.

rejected as a nonaromatic one. Finally, we assign aromaticity according to the Huckel rule [14]. As all the rings are elements of the SSSR, a special procedure is required to deal with some conjugated rings. More precisely, if a ring violates the Huckel rule but has a conjugated planar ring, the two rings are merged and considered as a new ring. Fig. 3 presents a schematic workflow of the aromaticity detection.

Assignment of Atom Types and Bond Orders

In the last steps of the ligand analysis, we set types corresponding to the functional groups. During this stage, we remove excessive double bonds and correct hybridization states when needed. As *Knodle* is primarily designed for the small molecules' type recognition, we aimed at a very accurate and exhaustive system of atomic types that would represent the corresponding chemical properties. Thus, we adapted the internal types system of the fconv library, which consists of 160 types that can fully describe chemical properties of all atoms in non-metallic ligands. The assignment of atom types is based on common knowledge about properties of the functional groups, similar to the ones used in the fconv method and the Pluto [8] program for Cambridge Structural Database (CSD) structures analysis.

More specifically, we start a loop over all the atoms to analyze their neighbors and assign functional groups in accordance with their atomic hybridizations derived by SVM. The list of functional groups includes carboxylic, phosphate, amino, guanidinium, amidinium groups and other, which can be deduced from the list of the extended atom types. We make the decision about the functional group representing the current atom based on its hybridization and its bonds predicted by SVM, as well as its heavy neighbors. We use the following information about the heavy neighbors: their element symbols, hybridizations, atomic types (if they are already set), and the number of their neighbors. We also use the same information about the neighbors of neighbors if the current atom is a candidate to belong to a functional group involving more than two atoms.

There are functional groups of a higher priority, such as amide and carboxylic groups. For example, even if there are geometrical distortions and SVM predict a carbon atom of an amide group to be sp3, it becomes sp2. For guanidinium, amidinium, imino and amino groups we make an additional loop after all the atoms have been parsed to make sure that there are no conflicts in functional groups assignment. In the end, we assign bond orders on the basis of the SVM predictions and the information about the functional groups. In case of any uncertainties, for example when the sum of the bond orders exceeds the atom valence, we make the final decision about the bond types based on very rough tests, such as the verification of the length of the shortest bond between the atoms. Our extended types can then be converted into the popular Tripos Mol2 format [8].

Computational Details

The presented method is implemented using the C++ programming language and compiled with the gcc-4.8 compiler on Linux and the clang compiler on Mac OS systems with -O3 optimization level. The test benchmarks were run on the following machines: a desktop with Linux and 3.10 GHz Intel(R) Core(TM) i5-4440 CPU processor and 16 GB 1600 MHz DDR3 RAM, and a MacBook Pro late 2013 laptop with a 2.6 GHz Intel Core i7 processor and 16 GB 1600 MHz DDR3 RAM. *Knodle* was tested with two versions of its SVM kernels, the slow exact one, and the fast approximate one. The user has the choice of these two options.

Results and Discussion

We compared *Knodle's* performance with three popular methods for the molecular types recognition and format conversion, NAOMI [22], fconv[20] and I-interpret[28]. To do so, we ran several series of tests. First, we assessed the accuracy of the methods on the Labute's benchmark set of 179 protein-ligand complexes [15]. Second, we assessed the accuracy of the methods on a part of the PDBBindCN general data set (v2014) consisting of 3,000 protein-ligand complexes that were randomly excluded from the SVM model training. Third, we performed a more general evaluation of our method using 332,974 ligands extracted from the Ligand Expo database [11]. These 332,974 ligands do not intersect with those ligands that were used for training. Finally, we compared the running times of the methods using 10,605 protein-ligand complexes from the PDBBindCN general data set (v2014), and 10,605 isolated ligands from the same data set. All the programs were run with the default parameters using a pdb file of a protein-ligand complex or the ligand alone as input. The output of each program run was a mol2 ligand file. *Knodle* was tested with the default fast approximate SVM kernels as well as with the optional slow exact ones. To measure the timings, we ran each program three times and chose the best running time.

Labute's Test Set

Although the Labute's benchmark set [15] is nowadays far from being exhaustive or representative, we decided to include it for historical reasons, since most of the previously developed perception methods were validated on this set. For this set, we ran the four programs and manually compared the results with the reference structures, whose chemical schemes can be found in the RCSB PDB database [5]. We must mention here that 59 of 179 complexes of the Labute's test set appeared to be in the training set, which can make this assessment somewhat biased. Therefore, we further assessed Knodle's accuracy on other two sets. Table 4 lists the results obtained on this benchmark.

PDB ID(s) (ligand name)	*Knodle*	fconv	NAOMI	I-interpret
1aaq	🟥	🟥	🟩	🟥
1aqb	🟥	🟥	🟩	🟩
3fx2	🟥	🟥	🟥	🟥
8xia	🟥	🟥	🟥	🟥
1nnb (DAN)	🟥	🟩	🟩	🟩
1tlp, 2tmn	🟩	🟥	🟥	🟥
2r04	🟩	🟥	🟥	🟥
1htg	🟩	🟥	🟩	🟩
4gr1 (RGS)	🟩	🟩	🟩	🟥
4fab (FDS)	🟩	🟥	🟥	🟩
1cps	🟩	🟩	🟩	🟥
1mnc	🟩	🟩	🟩	🟥
5tln	🟥	🟩	🟩	🟩
1bzm	🟩	🟩	🟥	🟩
	HEM ligands:			
7cat	🟩	🟥	🟩	🟩
1phf,1phe (HEM)	🟩	🟥	🟩	🟩
1mbi	🟩	🟥	🟩	🟩

Table 4: Structure perception errors obtained on the Labute's benchmark set by the *Knodle*, fconv, NAOMI, and I-interpret algorithms. The green and red colors stand for the correct and erroneous structure perception cases. A more detailed illustrated description of errors can be found in Table S1 and Figure S1 from Supporting Information.

As we can see from the table, most of the *Knodle*'s errors are caused by ambiguous geometry of the ligands (1AAQ, 3FX2, 8XIA, 1NNB), and one error was made in a long retinol chain (1AQB). All errors except the 1NNB one are shared with at least one of the other packages. As we have mentioned above, for the sake of computational performance, we replaced several radial SVM kernels with the faster ones (see the last column in Table 1). This replacement, generally, does not affect the accuracy of the method on the training set. However, in the Labute's set, one of the new kernels causes one more error (5TLN). The choice between the default fast and optional slow versions of the kernels is at the user's discretion. Overall, on the Labute's benchmark set, *Knodle* made five to six perception errors depending on the chosen SVM kernels, NAOMI made seven errors, fconv made thirteen errors, and I-interpret made nine errors. A more detailed analysis of the errors can be found in Table S1 from Supporting Information. The corresponding chemical schemes are plotted in Figure S1 from Supporting Information.

PDBBindCN Test Set

To further validate the results, we performed a series of tests on a part of PDBBindCN general data set (v2014) [18] that was not used for the training (3000 ligands). Here, we assessed the quality of *Knodle* (with fast kernels), fconv, NAOMI and I-interpret. This set is more statistically significant than the previous one, but still allows a visual inspection of errors along with a comparison of mol2 types predicted by the four methods. We first obtained the results using an automatic parser of the mol2 files. The parser was designed to compare mol2 atom types and, consequently, hybridizations. Differences in the perception of tautomeric forms were not considered as errors. Some typical examples of the substructures where tautomerism occurs are guanidinium and amidinium groups, pyrimidines with amino substituents, pyrimidones and keto-enol groups. Nitrogens in most of the nitrogen-containing groups from this list are usually planar trigonal and have N.pl3 SYBYL types, even if the bond is single, thus making such tautomeric states easily distinguishable at the parsing stage. We compared the results of the four tools with the mol2 ligand files provided by PDBBindCN. Afterward, we manually examined all the hybridization inconsistencies by visual inspection of ligand structures and their SMILES notations from the RCSB PDB[5] and PDBeChem [10] databases. Some of the inconsistencies were not interpreted as errors because of different tautomeric states of the corresponding groups. We excluded several structures from the comparison as they contained a metal or metalloid atom that affected prediction of the correct bond orders and hybridizations in its vicinity. An example of such a structure containing a ferrocenyl group is shown in Fig. 4. In this example, it is incorrect to compare 5-membered nonaromatic carbon rings with certain double bond locations with the ferrocenyl group consisting of two rings with a delocalized electron density.

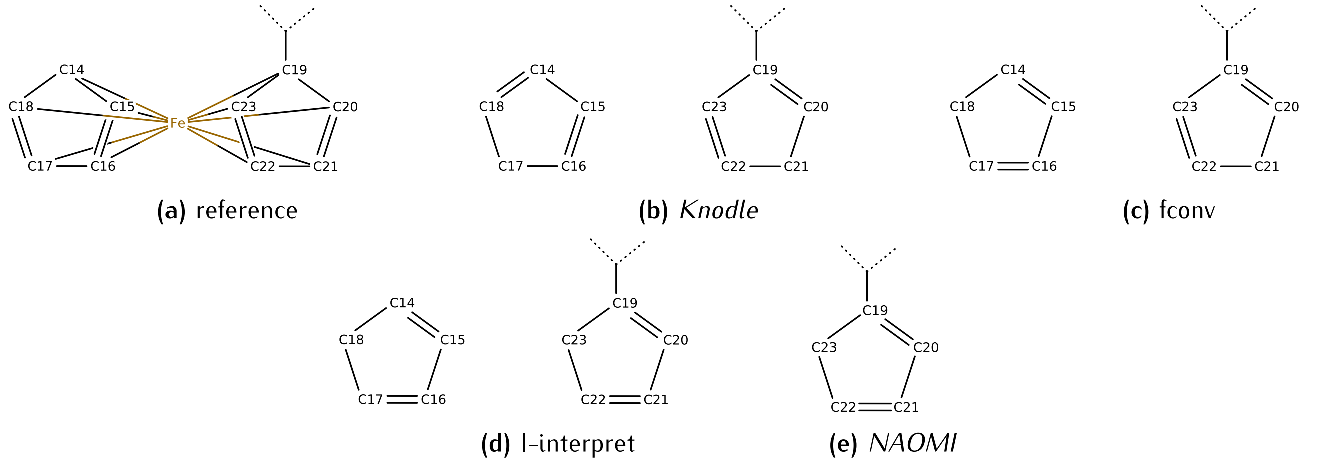
Overall, as can be seen in Table 5, out of the four tested methods, fconv demonstrated the worst performance with approximately 13% of errors. Knodle performs slightly better than NAOMI having 3.9% versus 4.8% of errors, correspondingly, while I-interpret made 6.2% of errors. A complete list of inconsistencies along with their descriptions can be found in Table S2 from Supporting Information.

Test set size	3000
Number of inconsistencies	544
Number of ligands incorrectly perceived by at least one of the methods	497
Number of errors in metallic structures	7

Tool	*Knodle*	fconv	NAOMI	I-interpret
#errors	116	383	140	181
Relative error	3.9%	12.8%	4.7%	6.0%

Table 5: Description of the PDBBindCN test set and the perception results of the four methods on this set.

As our prediction method relies on geometrical descriptors, *Knodle* is sensitive to the structure quality and the quantity of available descriptors. Consequently, most common errors either occur due to disruptions in molecular structures or appear in terminal atoms that are connected to a single heavy neighbor. For example, *Knodle* predicts wrong terminal oxygen hybridizations more often than NAOMI and I-interpret, especially in peptides. Besides common disruptions such as enormous bond lengths or valence angles, e.g. C-C bonds longer than 1.5 Å in the 3r4p structure, *Knodle* generally does not recognize aromaticity in



(a) reference **(b)** *Knodle* **(c)** fconv

(d) I-interpret **(e)** *NAOMI*

Figure 4: Perception of the ferrocenyl-containing 3p3h structure, excluded from the assessment of the methods. Here, NAOMI reported only the largest disconnected component.

those cycles where one of the atoms bulges out of the plane, for example, in the 3qgy structure (Fig. 5), where the value of the C3-C1-C2-C14 dihedral angle is about 23°. All the four programs make errors in molecules with a long chain of sp2-hybridized carbons like those in the EAH ligand in the 3nhi structure. Plenty of errors in the NAOMI, fconv and I-interpret results appeared in aromatic heterocycles with two exocyclic sp2-hybridized oxygens, as is shown in Fig. 6, where one of the cycle's carbons was predicted to have sp3 hybridization instead of the sp2 one. *Knodle* avoids these errors successfully. On the other hand, *Knodle* produces errors in molecules with adjacent cycles separated by a bond, where one of the cycles is not aromatic and thus the second cannot be aromatic either, as is shown in Fig. 7. Here, our method first incorrectly detects aromaticity of the 5-membered rings, then it changes the previously detected double bond between C13 and C14 atoms into a single bond, and thus the planar C14 atom perceives the wrong sp3 hybridization.

Ligand Expo Test Set

For a more general evaluation of our method, we ran *Knodle* on ligands in the pdb format from the Ligand Expo database [11]. To construct the pruned test set for the assessment, we first removed all empty entries, monatomic ligands, and entries with disconnected components. We also ignored truncated ligands with the number of heavy atoms smaller than in the reference chemical formulas provided by Ligand Expo. Finally, we excluded from the comparison metallic ligands, as it is difficult to distinguish whether a perception error is caused by the vicinity of the metal or not, and intersections with the training set. Table 6 describes the structure of the pruned Ligand Expo test set, which contains 332,974 ligands.

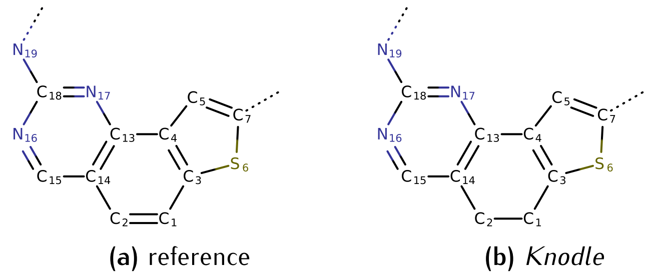
Ligand Expo dataset set size	739,847
# Skipped entries, in total	406,873
# Empty entries	9,009
# Monatomic ligands	147,958
# Metallic ligands[a]	40,032
# Truncated ligands[a]	169,933
# Disconnected components[a]	28,343
# Intersections with the training set[a]	11,598
# Compared structures	**332,974**

Table 6: Ligand Expo test set statistics.

[a] If these entries have not been skipped previously.

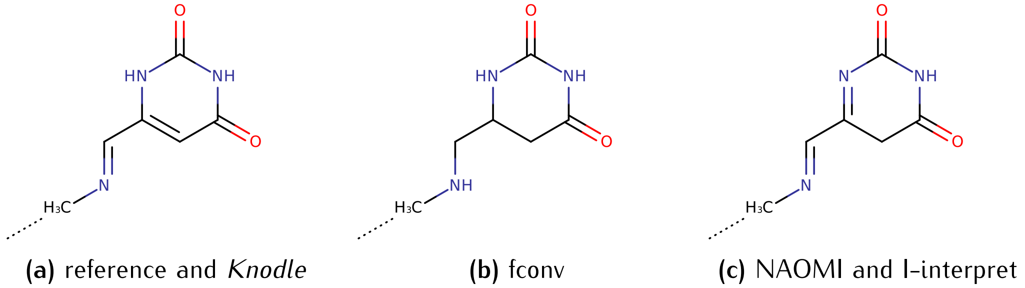
To proceed with the evaluation, we ran *Knodle* to convert 332,974 ligands from the pruned Ligand Expo test set into sdf files. Then, we neutralized the predictions and converted them again to the canonical SMILES format. Afterward, we compared the results with the SMILES strings of the reference structures.

The comparison identified 34,676 converted structures that had bond orders different from the reference. For these structures, we compared their



(a) reference **(b)** *Knodle*

Figure 5: Perception of the 3qgy structure. Here, *Knodle* does not recognize aromaticity of the nonplanar C1-C2-C14-C13-C4-C3 ring, where the value of the C3-C1-C2-C14 dihedral angle is of about 23°.



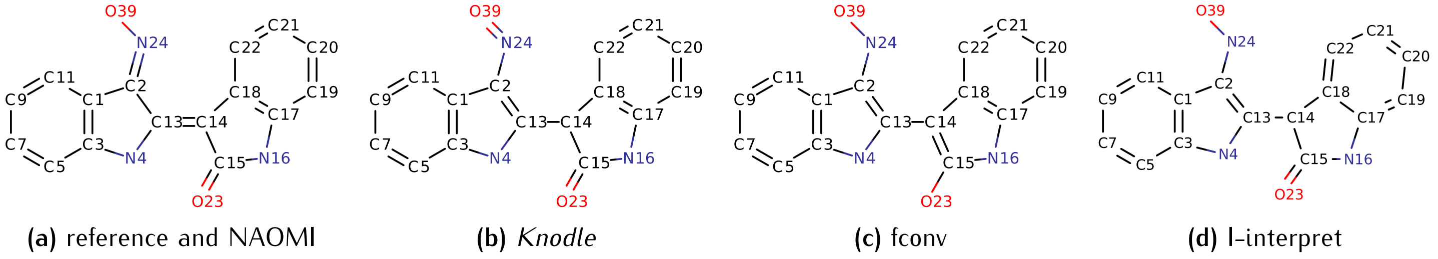
(a) reference and *Knodle* **(b)** fconv **(c)** NAOMI and I-interpret

Figure 6: Perception of the 3fcf structure, an example of an aromatic heterocycle with two exocyclic sp2-hybridized oxygens. NAOMI, I-interpret, and fconv incorrectly predict hybridization of one of the cycle's carbons.

InCHI codes to detect different tautomeric states and any other inconsistencies in the SMILES strings caused by the conversion algorithm. We considered 17,730 structures to be correct at this stage. Along with the tautomeric states, 1,918 structures with incorrect bond orders were generated because of ambiguous oxidation states in such molecules as NADP/NADPH, which cannot be truly distinguished by relying only on the available geometrical information. Overall, 15,028 components remained incorrect, accounting for 4.5 % of the number of the compared structures. Errors produced by our method can be well described, with the 25 most frequently incorrectly perceived ligands given in Table 7.

Most of the errors occur in GOL, or glycerol, a small alcohol compound with three terminal oxygen atoms, one or more of which are erroneously predicted to have sp2 hybridization. The fact that *Knodle* often produces errors in the terminal bonds with oxygen has already been mentioned above in the PDBBindCN test section. However, the wrong perception of GOL oxygens is not a typical error. Indeed, most frequently, it is the sp2 atoms that are perceived by *Knodle* as sp3, which can be illustrated with the number of errors in the ACE, PLP, ACO, FAD, BME and EDO structures.

A lot of errors occur in molecules with just a few atoms. Sometimes this happens due to the bond length and angle value distortions, as in the NO2 (nitrite ion NO_2^-) and AZI (azide ion N_3^-) ligands: the sp2 bonds of both of them have lengths of



(a) reference and NAOMI **(b)** *Knodle* **(c)** fconv **(d)** I-interpret

Figure 7: Perception of the 1unh structure, an example of a molecule with adjacent cycles separated by a bond.

name	#errors	name	#errors	name	#errors	name	#errors	name	#errors
GOL	1050	ACE	967	CMO	674	OXY	551	CYC	493
BCR	483	FAD	295	CYN	221	NO2	141	BLA	130
PGV	120	PEV	102	ACO	92	FMN	84	TGL	84
AZI	83	PSO	83	PEK	81	PEB	77	NAD	71
PLP	66	BME	66	H4B	57	EDO	53	M7G	49

Table 7: The 25 ligands from the LigandExpo dataset in which *Knodle* conversion errors most frequently occurred. Figure S2 from Supporting Information presents chemical schemes of these molecular structures.

values greater than 1.4 Å. The only angle of the AZI molecule often takes values smaller than 160°, while our method expects it to be straight. Diatomic compounds structure prediction causes some problems too, presumably because of the lack of training information. For example, CMO (carbon monoxide CO) could not be predicted correctly, as the training set does not contain compounds with sp1 oxygens. Moreover, the SYBYL mol2 types do not include any type representing an sp1-hybridized oxygen. There were also no statistics representing a double bond between the two oxygens for the OXY ligand (O_2 molecule). *Knodle* often fails to predict the triple bond in the CYN ligand (cyanide ion CN^-), in which the only data available are the element symbols and the bond length. We should note that the small amount of descriptors in this particular case is not a problem in itself, as a triple bond should be easily distinguishable thanks to its short length. Instead, these errors arise for the following reason. First, we do not have diatomic compounds in the training set. Second, the model that we use for the terminal atoms has a descriptor for the angle between the atom, its neighbor and the neighbor of the neighbor. For diatomic compounds this angle does not exist, the descriptor takes a default value and the prediction result of the model is generally undefined. Consequently, a bond in cyanide is often erroneously assigned even for structures without geometrical distortions. As a remedy, we plan to add diatomic compounds into the training set for a future release of our library.

For CYC, BLA and PEB ligands, errors occur in exocyclic bonds. BCR, PGV, PEV, PEK and TGL are molecules containing long aliphatic chains, in which the bond lengths and angles sometimes do not correspond to the commonly observed geometry. Several BCR structures have atoms with very short bonds of about 1.2 Å length that are common for sp1 hybridization, while these atoms should be sp2-hybridized. Conversely, in some structures such as PGV, atoms that should be sp2 have the geometry of sp3 atoms. Most of the errors in FMN, NAD and PSO ligands occur due to the distortions in the aromatic rings, i.e., bond lengths greater than 1.5 Å, small valence angles and enormous dihedral angles of 40-50°. In M7G and H4B ligands, aromatics is incorrectly perceived in a ring that is not aromatic. This ring contains four sp2 atoms, and the decision of whether it is aromatic or not is taken based on the hybridizations of only one (M7G) or two (H4B) carbon atoms that should be sp3. These, however, have short bond lengths and straight torsion angles, and thus are predicted as sp2.

Overall, the performance of our method is very similar to that reported by the authors of NAOMI [22], i.e. 6 out of 20[2] most frequently incorrectly predicted ligands are shared between the two methods, and many of the errors are caused by disruptions in molecular geometry or occur at similar locations.

Running Time

Generally, for the structure perception algorithms, running time is not of the highest priority compared to the accuracy of the perception. Nonetheless, to complete the assessment of different methods, we performed a comparison of running times of *Knodle*, fconv, NAOMI and I-interpret methods using the PDBBindCN general data set (v2014) consisting of 10,605 full protein-ligand complexes and the corresponding ligands, converted to the pdb file for-

[2] The list of the most frequently incorrectly perceived structures presented by the authors of NAOMI contained 20 ligands.

mat. Table 8 lists the timings of conversion of isolated ligands measured on two platforms. Overall, we can see that when converting small files, all the four tested methods demonstrate a very similar performance, with fconv being the fastest one, which is however negated by its accuracy. Table S3 from the Supporting Information also lists the timings of the conversion of full protein-ligand complexes. In this case, a vast time can be spent in reading the input files, and *Knodle* is significantly faster than other methods. We should add that we recommend using the default approximate SVM kernels in our method, as they are more computationally efficient with the same accuracy for the prediction results.

Package	Total time of extraction from the PDB format (10,605 isolated ligands), s	
	Linux	Mac OS
Knodle (exact slow kernels)	80	97.8
Knodle (approximate fast kernels)	59	79.5
Fconv	30	65.7
NAOMI	74	–[a]
I-interpret	78	–[a]

Table 8: Running times of the *Knodle*, fconv, NAOMI, and I-interpret methods for the PDBBind 2014 dataset. The timings are given for the set 10,605 isolated ligands. Running times for *Knodle* are given for the two versions of its SVM kernels, the exact one and the approximate one. [a]NAOMI and I-interpret executables for Mac OS are not available.

Conclusion

Here we presented *Knodle*, a software library for the recognition of atomic types for small molecules. We demonstrated that this task can be accomplished very efficiently using nonlinear support vector machines. To accelerate the computation of the nonlinear radial kernel functions, we approximated the most time-consuming of them using Maclaurin series expansions and third-order polynomials. The accuracy of the results and running time comparable with other popular methods show that SVM is a good alternative to existing approaches for the perception of atomic structures, being a more general and extendable tool. More precisely, on the popular Labute's benchmark set consisting of 179 protein-ligand complexes, *Knodle* made five to six perception errors, depending on the kernels chosen, NAOMI made seven errors, I-interpret made nine errors, while fconv made thirteen errors. On a larger set of 3,000 protein-ligand structures collected from the PDBBindCN general data set (v2014), *Knodle* and NAOMI had a comparable accuracy of 3.9% and 4.7% of errors, correspondingly, while I-interpret made 6.0% of errors, and fconv produced about 12.8 % of errors. On a more general set of 332,974 entries from the Ligand Expo database, *Knodle* made 4.5 % of perception errors. Overall, SVM demonstrated robustness, power, and ease in structure perception tasks studied here. In the future, when more structural data becomes available, it will be straightforward to retrain our SVM models. *Knodle* is available at `https://team.inria.fr/nano-d/software/Knodle`. A graphical user interface for *Knodle* will be made available at `http://www.samson-connect.net`.

Acknowledgement

The authors thank Dr. Damien Jouvenot from the University of Grenoble Alpes for providing his expertise on electronic structure of small molecules. The authors also thank Julie Bourget from Inria Grenoble for her invaluable assistance with the project. This study was partially supported by the Russian Federal Target Program 'Research and Development' (contract 14.587.21.0011, RFMEFI58715X0011) and the 5top100 program of the Ministry of Education and Science of the Russian Federation.

Supporting Information Available : Descriptions of errors in the Labute and PDBBindCN tests, structures of the most frequent incorrectly predicted molecules in the LigandExpo test and a more detailed comparison of running times. This material is available free of charge via the Internet at `http://pubs.acs.org/`.

Corresponding Author : Sergei Grudinin, NANO-D, INRIA Rhone-Alpes Research Center Minatec Campus 17 rue des Martyrs 38054 Grenoble France. Phone: +33 4 38 78 16 91. E-mail: sergei.grudinin@inria.fr.

References

[1] MP Allen and DJ Tildesley. *Computer Simulation of Liquids*. Oxford University Press, 1989.

[2] Svetlana Artemova, Sergei Grudinin, and Stephane Redon. A comparison of neighbor search algorithms for large rigid molecules. *J. Comput. Chem.*, 32(13):2865–2877, 2011.

[3] Svetlana Artemova, Sergei Grudinin, and Stephane Redon. Fast construction of assembly trees for molecular graphs. *J. Comput. Chem.*, 32(8):1589–1598, 2011.

[4] Jon C Baber and Edward E Hodgkin. Automatic assignment of chemical connectivity to organic m the cambridge structural database. *J. Chem. Inf. Comput. Sci.*, 32(5):401–406, 1992.

[5] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, TN Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic Acids Res.*, 28:235–242, 2000. `www.rcsb.org` (accessed Jan 20, 2016).

[6] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm` (accessed Jan 20, 2016).

[7] Marc Claesen, Frank De Smet, Johan AK Suykens, and Bart De Moor. Fast prediction with svm models containing rbf kernels. *arXiv preprint arXiv:1403.0736*, 2014.

[8] Matthew Clark, Richard D Cramer, and Nicole Van Opdenbosch. Validation of the general purpose tripos 5.2 force field. *J. Comput. Chem.*, 10(8):982–1012, 1989.

[9] Anna Katharina Dehof, Alexander Rurainski, Quang Bao Anh Bui, Sebastian Böcker, Hans-Peter Lenhof, and Andreas Hildebrandt. Automated bond order assignment as an optimization problem. *Bioinformatics*, 27(5):619–625, 2011.

[10] D. Dimitropoulos, J. Ionides, and K Henrick. Unit 14.3: Using pdbechem to search the pdb ligand dictionary. In A.D. Baxevanis, R.D.M. Page, G.A. Petsko, L.D. Stein, and G.D. Stormo, editors, *Current Protocols in Bioinformatics*, volume 27, pages 14.3.1–14.3.3. John Wiley & Sons, Hoboken, N. J., 2006.

[11] Zukang Feng, Li Chen, Himabindu Maddula, Ozgur Akcan, Rose Oughtred, Helen M Berman, and John Westbrook. Ligand depot: a data warehouse for ligands bound to macromolecules. *Bioinformatics*, 20(13):2153–5, Sep 2004.

[12] Matheus Froeyen and Piet Herdewijn. Correct bond order assignment in a molecular framework using integer linear programming with application to molecules where only non-hydrogen atom coordinates are available. *J. Chem. Inf. Model.*, 45(5):1267–1274, 2005.

[13] Manfred Hendlich, Friedrich Rippmann, and Gerhard Barnickel. Bali: Automatic assignment of bond and atom types for protein ligands in the brookhaven protein databank. *J. Chem. Inf. Comput. Sci.*, 37(4):774–778, 1997.

[14] Erich Hückel. Quantentheoretische beiträge zum benzolproblem. *Zeitschrift für Physik A Hadrons and Nuclei*, 70(3):204–286, 1931.

[15] Paul Labute. On the perception of molecules from 3d atomic coordinates. *J. Chem. Inf. Model.*, 45(2):215–221, 2005.

[16] Elke Lang, Claus-Wilhelm von der Lieth, and Thomas Förster. Automatic assignment of bond orders based on the analysis of the internal coordinates of molecular structures. *Anal. Chim. Acta*, 265(2):283–289, 1992.

[17] Chang Joon Lee, Young-Mook Kang, Kwang-Hwi Cho, and Kyoung Tai No. A robust method for searching the smallest set of smallest rings with a path-included distance matrix. *Proc. Natl. Acad. Sci. U. S. A.*, 106(41):17355–17358, 2009.

[18] Zhihai Liu, Yan Li, Li Han, Jie Li, Jie Liu, Zhixiong Zhao, Wei Nie, Yuchen Liu, and Renxiao

Wang. Pdb-wide collection of binding data: Current status of the pdbbind database. *Bioinformatics*, 1(31):405–412, 2015.

[19] Gerd Neudert and Gerhard Klebe. Dsx: a knowledge-based scoring function for the assessment of protein–ligand complexes. *J. Chem. Inf. Model.*, 51(10):2731–2745, 2011.

[20] Gerd Neudert and Gerhard Klebe. Fconv: Format conversion, manipulation and feature computation of molecular data. *Bioinformatics*, 27(7):1021–1022, 2011.

[21] Robert Tarjan. Depth-first search and linear graph algorithms. *SIAM J. Computing*, 1(2):146–160, 1972.

[22] Sascha Urbaczek, Adrian Kolodzik, Inken Groth, Stefan Heuser, and Matthias Rarey. Reading pdb: Perception of molecules from 3d atomic coordinates. *J. Chem. Inf. Model.*, 53(1):76–87, 2012.

[23] Daan MF van Aalten, R Bywater, John BC Findlay, Manfred Hendlich, Rob WW Hooft, and Gert Vriend. Prodrg, a program for generating molecular topologies and unique molecular descriptors from coordinates of small molecules. *J. Comput.-Aided Mol. Des.*, 10(3):255–262, 1996.

[24] Vladimir Vapnik, Steven E Golowich, and Alex Smola. Support vector method for function approximation, regression estimation, and signal processing. *Advances in Neural Information Processing Systems*, pages 281–287, 1997.

[25] Junmei Wang, Wei Wang, Peter A Kollman, and David A Case. Automatic atom type and bond type perception in molecular mechanical calculations. *J. Mol. Graphics Modell.*, 25(2):247–260, 2006.

[26] Bohdan Waszkowycz, David E Clark, and Emanuela Gancia. Outstanding challenges in protein–ligand docking and structure-based virtual screening. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.*, 1(2):229–259, 2011.

[27] Qian Zhang, Wei Zhang, Youyong Li, Junmei Wang, Liling Zhang, and Tingjun Hou. A rule-based algorithm for automatic bond type perception. *J. Cheminf.*, 4(1):1–10, 2012.

[28] Yuan Zhao, Tiejun Cheng, and Renxiao Wang. Automatic perception of organic molecules based on essential structural information. *J. Chem. Inf. Model.*, 47(4):1379–1385, 2007.