scientific data

OPEN

DATA DESCRIPTOR

Labeled dataset of X-ray protein ligand images in 3D point cloud and validated deep learning models

Cristina F. Bazzano[1,2], Luiz F. G. Alves[2], Guilherme P. Telles[1] ✉ & Daniela B. B. Trivella[2] ✉

***LigPCDS (Ligand Point Cloud Data Set)* is the first dataset of chemically labeled 3D point clouds of protein ligands. 3D images and structures of ligands were derived from X-ray protein crystallography experimental datasets deposited at the Protein Data Bank. The 3D point cloud format allowed for a computer-comprehensive representation of the ligand's experimental data, enabling the interpretation of the ligand's chemical structure using a building block-like labeling approach. For constructing *LigPCDS*, the images of the ligands were interpolated from their difference electron density map into a 3D grid-like structure, filtered around their atomic spheres, and stored in point clouds. The density value was used as a single feature. Chemical vocabularies, based on atoms and their cyclic structural arrangements, were designed and used to pointwise label these 3D representations of the ligands. The proposed imaging and labeling approaches were validated by training semantic segmentation deep learning models on a stratified dataset from *LigPCDS*, which could recover the protein ligand's chemical structure with good performance. *LigPCDS* can be used to achieve solutions for building known and yet unknown protein ligands (small organic molecules) from experimental X-ray protein crystallography, in silico ligand screening, drug design, and to understand protein function in basic biology.**

Background & Summary

Ligands are small molecules that bind to proteins, generally modifying their function. These molecules represent the active principles of known medicines (active pharmaceutical ingredient, API), or drug prototypes (*e.g*. natural products, fragments and other synthetic small molecules) in drug discovery pipelines. Currently, the 3D structure of protein-ligand complexes is mostly obtained by X-ray protein crystallography[1]. Experimental X-ray protein crystallographic data are freely available at the data centers of the global Protein Data Bank (PDB, https://www.wwpdb.org/)[2,3], a worldwide archive of macromolecular structure data.

The electron density map is the primary result of an X-ray protein crystallography experiment[4]. It is a continuous function ρ(x,y,z) of intensity values in the real space, being measured in electrons per cubic angstrom ($eÅ^{-3}$). It represents the electron cloud around each atom of the protein and of its ligands in the 3D space, allowing for deciphering the protein-ligand atomic 3D structure[4,5]. The presence of ligands in X-ray protein structures can be detected in the calculated difference electron density map, the Fo-Fc map, which highlights the presence of additional molecules binding to the protein, such as the ligands[4,6–9]. The 3D image of a ligand is observed in high intensity regions of the Fo-Fc map, being named *blob* or density cluster. A ligand *blob* is displayed by applying a contour to the Fo-Fc, usually using the sigma (σ) scale, which filters the points above a cutoff value (*e.g*. 3σ), highlighting the ligand structural features[4,6–9]. The interpretation of the chemical structure of a ligand in the Fo-Fc map is a central task for understanding the functionality of the protein and guiding structure-based drug design (SBDD) in modern drug discovery pipelines.

Existing solutions for known ligand building are based on up to 200 known and common molecules from PDB. These solutions use mathematical and topological descriptors of Fo-Fc maps and suggest a list of molecules that best explain and fit into a *blob*[10–14]. While identifying known ligands with such approaches, the accuracies range from 32% to 72.5% for the best prediction[12]. This indicates that the ligand building problem still lacks accurate solutions, even for known ligand building, and that there is potential for improvement. A very recent development was reported by Karolczak and coworkers[15] using deep learning and point clouds, with average

[1]Institute of Computing, University of Campinas (UNICAMP), Campinas, 13083-852, SP, Brazil. [2]Brazilian Biosciences National Laboratory (LNBio), Brazilian Center for Research in Energy and Materials (CNPEM), Campinas, 13083-100, SP, Brazil. ✉e-mail: gpt@ic.unicamp.br; daniela.trivella@lnbio.cnpem.br

accuracies reaching 67.2% and 93.6% in the top-10 known ligand suggestions. However, their approach still relies on the whole known protein ligand structures as the training and searching sets. When the ligand is unknown, as the case of novel natural products[6,9,16] or other molecules with yet unknown chemical structures in X-ray protein databases, the current automated solutions are not accurate and cannot provide a reliable support for the crystallographer's interpretation of the ligand *blob* and of the chemical structure of this new ligand.

PDB[2,3] is a leading global resource for experimental data which is growing tremendously fast, with around 10 thousand deposits per year[17]. However, mining PDB data is difficult, mainly due to human errors in ligand interpretation and local low-quality *blobs*[5,11,18,19]. In addition, retrieving and manipulating ligand data in 3D grid-like format requests specific knowledge and crystallographic packages capable of reading crystallographic data. This may explain the lack of deep learning (DL)[17,20] approaches for ligand prediction from their *blobs*. DL with 3D point cloud have been showing remarkable results in other fields[21–24], and has started to be used for ligand interpretation in X-ray protein crystallography[15]. However, no chemical labeling of the ligand *blobs* is available nor has been validated (*i.e.*, is capable of being learned by a supervised machine learning – ML – model) to reconstitute novel protein ligand chemical structures.

To fill these gaps, we have created and validated the first chemically labeled dataset of experimental 3D images of protein ligands in 3D point clouds representations, named *LigPCDS*, with 244,226 ligand entries from PDB. The workflow for obtaining *LigPCDS* and its validation through successfully trained DL models is presented in Fig. 1.

For *LigPCDS* construction, a list of valid ligands from the Research Collaboratory for Structural Bioinformatics Protein Data Bank[3] (RCSB PDB, the US data center at https://www.rcsb.org/) was filtered and downloaded with experimental data. The entries were refined with Dimple v2.6.1 (https://ccp4.github.io/dimple/)[25] in a standardized procedure, without any added ligand (no heteroatoms), intended to normalize data quality and evidence the ligand *blob* in the Fo-Fc maps. The 3D image of the ligands were derived from their Fo-Fc maps with Gemmi[26] v0.5.8, based on the atomic positions of the ligand entries. Gemmi v0.5.8 was further used to create their representations in 3D point clouds, with an adequate scale, background removal, mask and contours. Finally, ligand 3D point clouds were labeled pointwise using an atomic sphere modelling, and designed chemical vocabularies. Different labeling approaches were proposed as vocabularies based on the atoms themselves and their cyclic structural arrangements, representing building blocks to construct the entire ligand chemical structure (Fig. 1a).

For validation of the labeling approach, a stratified training dataset (n = 78,902) from *LigPCDS* was used to train DL models for the semantic segmentation of the ligand's 3D representation (Fig. 1b). Four vocabularies led to good performance DL models (Fig. 1c): (i) the "Vocabulary of the Ligand Region", composed by generic atoms of any type; (ii) the "Vocabulary of Generic Atoms and Cycles", composed by generic atoms outside cyclic arrangements and generic atoms into cyclic structures (called here cycles); (iii) the "Vocabulary of Generic Atoms and Cycles C347CA56", composed by generic atoms outside cyclic arrangements, generic atoms in non-aromatic cyclic structures of size 3 to 7 and in aromatic cyclic structures of sizes 5 and 6; and (iv) the "Vocabulary of Atom Symbols with Groups", composed by the atom symbols with groupings. All vocabularies also contain the background class (regions in the images with no ligand atom), which is an important category to separate the background noise from the ligand itself. The mean accuracy of the validated models in their cross-validation, ranged from 49.7% (SEM = 0.4, CI = [−19.4, 20.2]) to 77.4% (SEM = 0.2, CI = [−11.7, 12.1]) in terms of the Intersection over Union (mIoU) metric[27]; and from 62.4% (SEM = 0.4, CI = [−18.8, 19.7]) to 87.0% (SEM = 0.2, CI = [−8.4, 8.8]) in F1-score (mF1)[28]. The accuracy of the validated models reinforces the reliability of the methods used to construct *LigPCDS* and suggests its future use by other machine learning tasks.

The robustness, size and labeling approaches of *LigPCDS*, together with the validated DL models, expands the possibility of interpreting unknown protein ligands, and further opens avenues for other DL applications based on protein ligands (*e.g.* in basic biology, natural product and drug discovery). As a first application using the validated DL models from *LigPCDS*, we have developed the NP^3 Blob Label (https://github.com/danielatrivella/np3_ligand/tree/master/np3_blob_label), an open source application designed to assist unknown ligand building in high performance drug discovery pipelines, including those focused on novel natural products (to be published). *LigPCDS* may also be used to address the problem of known ligand building, by using the ligands codes (unique structures) as labels for training DL classification tasks.

Methods

The *LigPCDS* dataset creation followed six major steps (Fig. 1), which are summarized below and explained in detail in the next subsections.

1. Creation of a list of valid ligands from RCSB PDB.
2. Creation of the representations of the ligand 3D image in 3D point clouds.
3. Creation of chemical vocabularies and ligand structure labeling.
4. Labeling ligand 3D point clouds.
5. Creation of a stratified training dataset from *LigPCDS*.
6. Training, optimization and validation of DL models.

The validation steps (steps 5 and 6 in Fig. 1b,c) of *LigPCDS* methodology are presented in the Technical Validation section.

Hardware. The hardware used to execute the *LigPCDS* creation and the DL models training is a computer with the following configuration: AMD Ryzen 9 3950X CPU, 16 cores and 32 threads, 128 Gb RAM and 2x GeForce RTX 2080 SUPER GPUs with 8 Gb of dedicated RAM each (hardware A). Exceptions were for specific

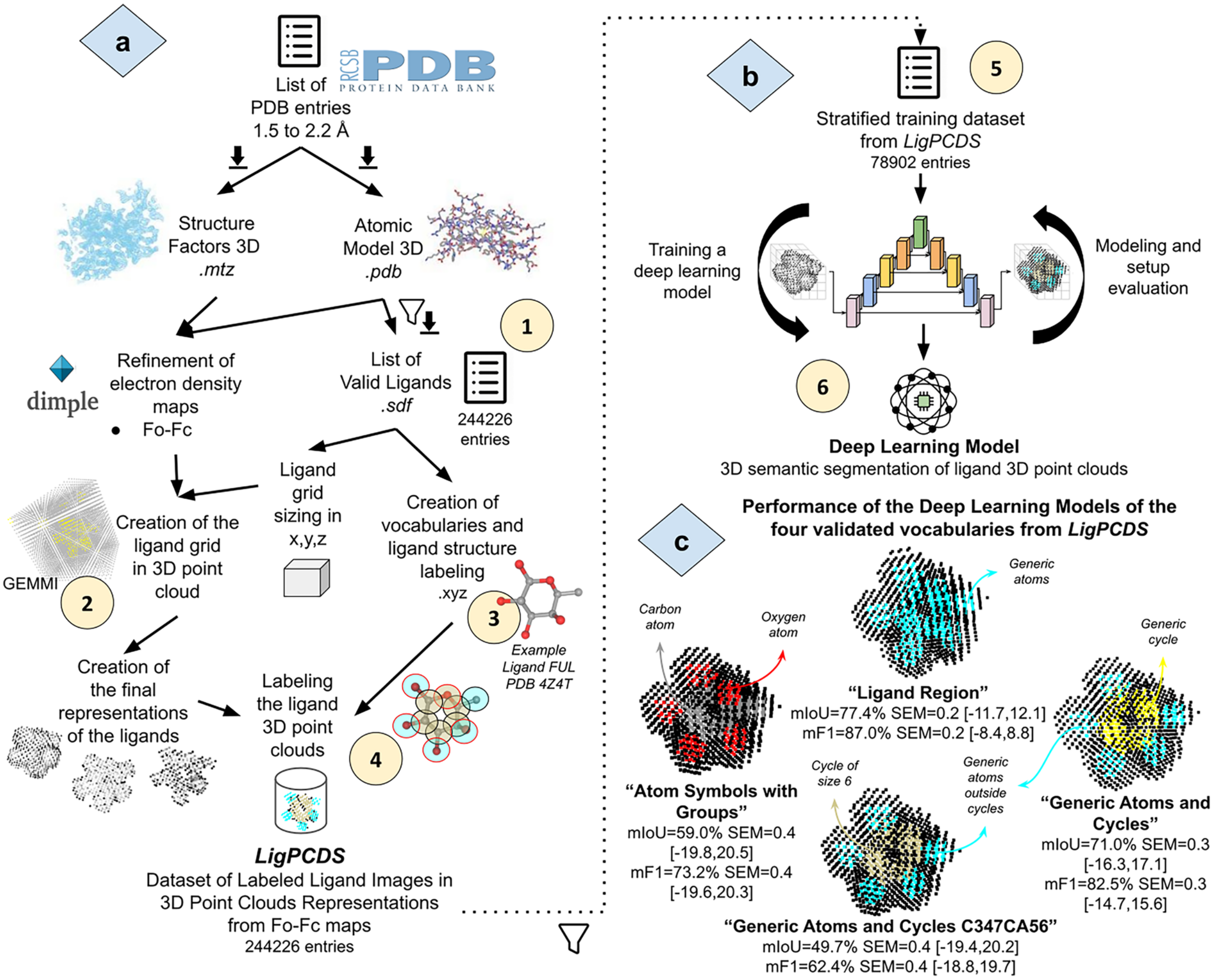

a
List of PDB entries 1.5 to 2.2 Å
RCSB PDB PROTEIN DATA BANK
Structure Factors 3D .mtz
Atomic Model 3D .pdb
1
dimple
Refinement of electron density maps
• Fo-Fc
List of Valid Ligands .sdf
244226 entries
Ligand grid sizing in x,y,z
Creation of the ligand grid in 3D point cloud
GEMMI
2
Creation of vocabularies and ligand structure labeling .xyz
3
Example Ligand FUL PDB 4Z4T
Creation of the final representations of the ligands
Labeling the ligand 3D point clouds
4
LigPCDS
Dataset of Labeled Ligand Images in 3D Point Clouds Representations from Fo-Fc maps
244226 entries
b
5
Stratified training dataset from LigPCDS
78902 entries
Training a deep learning model
Modeling and setup evaluation
6
Deep Learning Model
3D semantic segmentation of ligand 3D point clouds
c
Performance of the Deep Learning Models of the four validated vocabularies from LigPCDS
Generic atoms
"Ligand Region"
mIoU=77.4% SEM=0.2 [-11.7,12.1]
mF1=87.0% SEM=0.2 [-8.4,8.8]
Carbon atom
Oxygen atom
"Atom Symbols with Groups"
mIoU=59.0% SEM=0.4 [-19.8,20.5]
mF1=73.2% SEM=0.4 [-19.6,20.3]
Generic cycle
"Generic Atoms and Cycles"
mIoU=71.0% SEM=0.3 [-16.3,17.1]
mF1=82.5% SEM=0.3 [-14.7,15.6]
Cycle of size 6
Generic atoms outside cycles
"Generic Atoms and Cycles C347CA56"
mIoU=49.7% SEM=0.4 [-19.4,20.2]
mF1=62.4% SEM=0.4 [-18.8,19.7]

Fig. 1 Workflow used to obtain *LigPCDS*, the deep learning models training and the validated labeling approaches. (**a**) *LigPCDS* creation schema. In step 1, a list of PDB entries, with resolutions ranging from 1.5 to 2.2 Å, was retrieved from RCSB (.pdb and.mtz) and their free and organic ligands were downloaded, filtered and validated (.sdf). It resulted in the list of valid ligands with 244,226 entries. In step 2, Dimple v2.6.1 was used to refine the PDB entries and calculate their Fo-Fc maps. Next, for each ligand, a grid sizing was defined to cover its entire *blob*. Each ligand's grid was interpolated from its Fo-Fc map to a 3D point cloud and processed to create the final 3D representations of the ligands. In step 3, vocabularies of chemical classes were created and used for labeling the structure of the valid ligands atom-wise. They were based on the chemical atoms themselves and on cyclic substructures of the ligands. Finally, in step 4 the labels of the structure of the ligands were extrapolated pointwise, using an atomic sphere model, for labeling the final 3D representations of the ligands, resulting in *LigPCDS*. (**b**) General schema used to train and obtain the validated DL models. A stratified training dataset was created from *LigPCDS* with n = 78,902 ligand entries, separated in k = 13 similar groups (step 5). The *LigPCDS* entries of this dataset were used to train DL models in semantic segmentation tasks using the Minkowski Engine[47] architecture and networks based on the 3D U-Net[52]. Cycles of training, evaluation and changes continued until good performance DL models were obtained and validated (step 6). (**c**) Four of the proposed labeling approaches were validated and are illustrated with ligand FUL from PDB (entry 4Z4T). The average performance in the cross-validation of the best DL model trained with each vocabulary is presented by the mIoU and the mF1 metrics, with corresponding SEM and confidence interval (CI). k = 1 was used in the tests except for the model trained with the vocabulary of "Generic Atoms and Cycles C347CA56", which used the average k-fold value and k = 13. Image "Machine Learning" is by Srinivas Agra and image "intelligence" is by Gacem Tachfin from the Noun Project (CCBY3.0).

DL models analyses that are point out in the text and used hardware B, a cluster with the following configuration: AMD EPYC 7742 CPU with 64 cores and 80 threads, 384 Gb RAM and 4 GPUs NVIDIA HGX A100 with 40 Gb of dedicated RAM each.

List of valid ligands. To obtain a list of ligands (step 1, Fig. 1), the advanced search tool of the RCSB PDB (https://www.rcsb.org/) was initially used to retrieve all entries with resolution between 1.5 Å and 2.2 Å, in December 2019. The chosen resolution range aligns with the most frequent resolution values found in the

PDB (Supplementary Figure 1) and those typically obtained in structural biology and drug discovery projects. Additional selections to the retrieved RCSB PDB files were: the presence of free ligands (non-covalent), availability of experimental data (entries with electron density maps also deposited), data originated from X-ray experiments with proteins, and deposited at PDB after January 2008 (more stringent validation metrics in PDB). For the free ligands, we have selected organic molecules formed by atoms of carbon, oxygen, nitrogen, phosphor, sulfur, iodine, fluorine, chlorine, bromine or selenium; hydrogen atoms were omitted here due to their poor detection by X-ray crystallography at the chosen resolution range. At this stage, this resolution range would reduce data variations caused by large differences in resolution for *LigPCDS* construction, while keeping ligand information that is still difficult to predict. Other ranges were not tested so far, and may be used in the future.

A total of 39,353 PDB entries were selected using the above criteria, containing 13,189 unique ligand codes (unique ligand structure). The.pdb and.mtz files of these RCSB PDB entries were downloaded automatically. The coordinate lines representing the ligands present in the protein chains of these PDB entries were isolated from the retrieved files and saved into individual.pdb files. This procedure resulted in a total of 293,822 available ligand entries from 39,169 PDB entries, containing 13,074 unique ligand codes.

The Structure Data Format (SDF) file of each ligand entry was also downloaded from RCSB PDB. An SDF file is a chemical file format for molecular data based on the MOL-file format - which can store single or multiple molecules, describing all their atoms in 3D coordinates. Each ligand's SDF file was used to build and validate the ligand representative molecular graph (chemical structures). The free ligand entries with validated SDF files were used to propose chemical vocabularies for labeling the structure of protein ligands using a building block-like approach. This structure validation resulted in a total of 259,606 ligand entries from 39,052 PDB entries, containing 12,972 unique ligand codes.

To validate the experimental data of each PDB entry, a standardized procedure was proposed to refine the datasets downloaded from RCSB PDB (.mtz and.pdb files), without the ligand atomic entries, aiming to improve the *blob* imaging and to remove any failed PDB entry (described in the next subsection). In addition, the ligand entries with validated SDF files were also used to extract the ligand's 3D representations from their correctly refined Fo-Fc maps (described in the next subsections). The ligand entries that raised an error in any step were removed from the **list of valid ligands**.

The final **list of valid ligands** contains 244,226 entries of ligands from 36,202 PDB deposits. These ligands represent non-covalent protein ligands composed by C, O, N, P, S, Se, F, Cl, Br and/or I atoms, where 12,239 are unique ligand codes (unique structures) with frequencies ranging from 1 to 33,063 occurrences (20 ± 526). Single atoms or ions (e.g. Cl-) correspond to 8.6% of the ligand entries (n = 21,003), while the other 91.4% are valid molecular structures (n = 223,223). The median size of valid ligands is 6 atoms and the mean size is 11 non-hydrogen atoms, with sizes ranging from 1 to 140 non-hydrogen atoms. These statistics indicate a great imbalance problem in the list of valid ligands, which is related to the diversity of non-covalent ligands deposited in PDB. They also highlight the diversity of potential protein ligands with importance in biology and drug discovery. Many of such ligands are still to be discovered and will have to be interpreted in the future, as novel X-ray protein structures in complex with ligands are obtained.

The RCSB PDB downloads were automated with Python v3.8 scripts, and the ligand entries validation used the functionalities of the RDKit package v2019.09.3 (https://www.rdkit.org). 16.9% of the ligand entries and 8% of the PDB entries were excluded during validation, 11.6% of the ligand entries due to invalid SDF files (minor download errors are also included), 4.0% due to refinement errors and 1.3% due to errors in the creation and labeling of the ligand's 3D representation. This indicates poor quality of part of the ligand entries, further highlighting the difficulties for directly applying data mining techniques on PDB data[19].

Ligand 3D representation in point cloud.

Next in *LigPCDS* creation, the 3D representations of the ligands present in the list of valid ligands were designed and created. Considering the variability and flexibility in the size and conformation of ligands, the ease and speed of manipulating point clouds[29], and the availability of many good performance deep learning architectures for 3D point clouds[30], we have chosen point clouds as the format to represent the 3D images of ligands in *LigPCDS*.

The point clouds were initially extracted from the Fo-Fc maps using a ligand grid. For this, a 3D grid box was drawn around the ligand and the electron density intensity values in each x,y,z coordinate of the grid was computed and stored in the color channels of the point cloud. Then, contours and scales were applied to extract the 3D representations of the ligand images, without background and noise. Nine types of 3D representations (at different contours and scales) were generated to each ligand and are available at *LigPCDS*. The representation type to be used in a given application will depend on the desired application of the user, in a case-by-case basis. For our deep learning model of ligand chemical structure prediction, the qRankMask_5 representation showed the best results.

The detailed schema used in *LigPCDS* for creating the 3D representations of ligands in 3D point cloud format (step 2, Fig. 1) is shown in Fig. 2. A step-by-step explanation of this process is given below.

Refinement of the Fo-Fc maps (experimental data preparation).

Before extracting the 3D representations of the ligand's *blob* in 3D point clouds, each PDB entry in the list of valid ligands were first refined using the Dimple software v2.6.1 (https://ccp4.github.io/dimple/), a macromolecular crystallographic pipeline for refinement incorporated into the CCP4 program suite[25]. A standardized Dimple refinement was performed for each PDB entry using their respective downloaded.mtz and.pdb files, with the option of removing heteroatoms (it removes all ligands from the.pdb file) and with two refinement cycles (longer refinement). The other parameters of Dimple received their default values. Dimple refinement was carried out with two primary objectives: first, to highlight the presence of any ligand *blob* in the crystal structure. With the "remove heteroatom" parameter active, the unmodeled electron density related to the ligands (high values in the Fo-Fc maps) could be revealed,

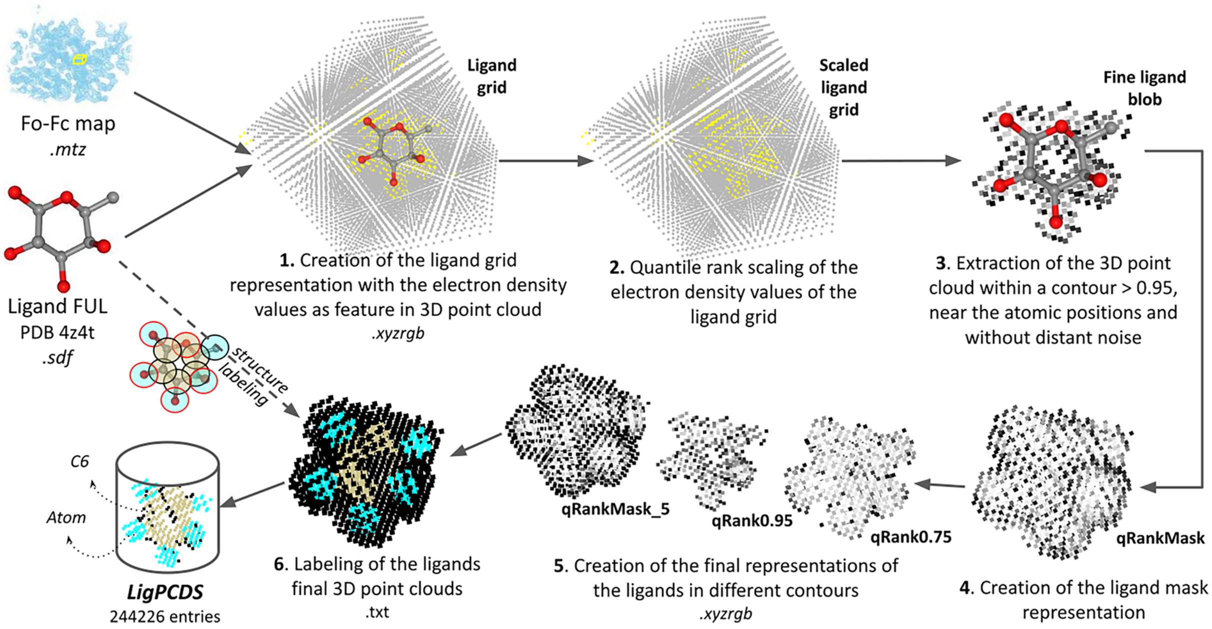

Fo-Fc map
.mtz
Ligand FUL
PDB 4z4t
.sdf
Ligand grid
1. Creation of the ligand grid representation with the electron density values as feature in 3D point cloud
.xyzrgb
Scaled ligand grid
2. Quantile rank scaling of the electron density values of the ligand grid
Fine ligand blob
3. Extraction of the 3D point cloud within a contour > 0.95, near the atomic positions and without distant noise
structure labeling
qRankMask
4. Creation of the ligand mask representation
qRankMask_5
qRank0.95
qRank0.75
5. Creation of the final representations of the ligands in different contours
.xyzrgb
6. Labeling of the ligands final 3D point clouds
.txt
C6
Atom
LigPCDS
244226 entries

Fig. 2 Schema for creating the labeled representations of ligands in 3D point cloud format for *LigPCDS*. The ligand FUL of PDB (entry 4Z4T) was used to exemplify the creation of the ligand's 3D point cloud starting from the grid up to the final 3D representations. (1) The ligand's grid representation is sized and interpolated from its Fo-Fc map in all its x,y,z positions, using the Gemmi package[26]. The ligand's grid is stored in point cloud format (.xyzrgb) with the density value of each point saved in its RGB channels (feature as colors). (2) The density values of the ligand's grid 3D point cloud are transformed and normalized using the quantile rank scale[33]. (3) The points of the ligand's grid within a contour of 0.95 (value > 0.95) are selected and only the points near the ligand's atomic positions and closely connected (with a distance between points smaller than grid space * 1.42 + 0.15) are retained, the rest is removed as noise. This creates the fine ligand *blob* representation. (4) The ligand's mask point cloud is created from this fine ligand *blob* by applying a 1.1 Å radius expansion from its borders and is named "qRankMask". (5) The final representations of the ligands are created by applying different contours in the ligand's mask representation and extracting the selected 3D point cloud. The final representations are named as "qRank" followed by the contour value, *e.g.* "qRank0.95". Additionally, a representation equal to the ligand's mask and with all values below 0.5 set to zero is created and named "qRankMask_5". This schema corresponds to the procedures used to complete step 2 in the *LigPCDS* creation workflow (Fig. 1a). (6) Finally, the labels of the ligand's structure are used for pointwise labeling the final 3D representations of the ligands, which corresponds to step 4 of the *LigPCDS* workflow (Fig. 1a).

and any bias related to incorrect ligand structure modelling on the PDB deposit would be removed. Second, to improve the overall Fo-Fc map and the local quality of the ligand *blob*, further normalizing the model refinement standards for the different crystal structures present in the list of valid ligands. The PDB entries that presented errors in the refinement were excluded. The **list of valid ligands** at this point contained 36,325 PDB entries successfully refined, with 247,878 ligand entries listed, from which 12,250 were unique ligands.

Extraction of the ligand grid representation in 3D point cloud (procedure 1, Fig. 2). A ligand grid was then created to extract the 3D image of each ligand *blob* (found in the refined Fo-Fc map) into the 3D point cloud format. The ligand grid is a bounding box defined on the boundary of the ligand's atomic positions, plus a gap, designed to cover the complete shape of the ligand *blob*. This procedure used the original SDF coordinates of the ligand to locate the center of its molecular structure in the refined Fo-Fc map, and to retrieve the ligand's atomic 3D coordinates, thus computing the bounding box on the boundary of its atomic positions. Through experimental inspection, this box was expanded with an additional gap equal to 4.2 Å in its boundaries (equal to the diameter of the largest theoretical radius[31] - Supplementary Table 1), and then, a second 120% expansion of its size was performed. The obtained dimensions defined the size of the ligand grid in the Fo-Fc map, centered on the ligand boundary box.

The Gemmi package[26] v0.5.8 was then used to interpolate the values of the Fo-Fc map for all x,y,z positions of the ligand grid. The obtained 3D grid was stored in a point cloud format, named the **ligand grid representation**. The difference electron density value of each point was chosen as the feature for the ligand 3D representation. The interpolated density value of each point (feature) was stored in the color channels of the 3D point clouds of the ligand grid representation. A spacing equal to 0.5 Å for the points of the ligand grid was tested and chosen. This value is smaller than the distance of a chemical bond (a sigma C-C bond measures around 1.54 Å) and allows to retain more details in the final 3D representations.

The Gemmi v0.5.8 Python package[26] for structural biology provides a framework of functions to manipulate electron density maps in indexable 3D grids, behaving like standard numerical vectors. Gemmi v0.5.8 allows extracting 3D grids from specific regions of an electron density map with different spacing between the points. It uses an implementation of the trilinear interpolation of the 8 closest points[32] of a given position of a map to compute its electron density value.

Transformation and scale of the ligand grid representation (procedure 2, Fig. 2). The quantile rank scale was then used to transform and scale the ligand grid to allow for their correct comparison. This is an equivalent approach to histogram equalization[33,34] in image processing. This scale normalizes the values in the range from 0 to 1. The quantile rank scale is used in other crystallography applications[33], and replaces the density value ρ(x,y,z) of each point by its position in the quantile distribution of the points for the region being considered. This scale does not change the shape of the electron density, all points that have the same ρ density values have the same value in this function. Furthermore, unlike the sigma scale, which must be applied globally across the entire electron density map, the quantile rank scale can be applied locally within a box to compare the same region. The sigma and quantile rank scales are comparable, with 1σ, 2σ or 3σ contours corresponding to quantile positions that vary approximately between 0.85, 0.95 and 0.98[33]. The use of the quantile rank scale allows to speed up calculations for data extraction, improves comparison, and excludes noise from the electron density map of distant regions, since the resolution of X-ray protein crystallographic data varies locally[35].

A fast implementation of the quantile rank scale function was created for this project: first it sorts the density values inside the ligand grid representation and then replaces the value of each point by its position in the ranked quantile distribution of the 3D-grid. Ties receive the first occurring position to the left. The **scaled ligand grid representation** for 247,424 ligand entries, 12,245 being unique ligands, were successfully created at this step.

Extraction of the fine ligand blob 3D representation (procedure 3, Fig. 2). The next step consisted in removing noise from the scaled ligand grid. For this, the scaled ligand grid representation was filtered to retrieve only the points within a contour of 0.95 (value > 0.95). Then, only the points near the ligand atomic positions and closely connected (with a distance between points smaller than the grid space × 1.42 + 0.15) were retained. By applying a neighborhood searching approach it was possible to remove the noisy points filtered from the ligand grid representation at 0.95 contour; in other words, the points that were not closely connected to the ligand atomic positions were removed here. This created the **fine ligand *blob* 3D representation** with a strong signal level and without noise. Python's Open3D package[29] v0.12 functionality was used to create the 3D point cloud of the ligand grid, mask and final representations (described in the next section). This package has an implementation of KDTrees using the FLANN library[36] for quick access of the closest neighborhood of the point clouds. This allowed searching with good performance.

Creation of the ligand mask representation (procedure 4, Fig. 2). The fine ligand *blob* 3D representation at 0.95 contour was then used as a reference for the *blob* location and shape. This 3D representation was expanded from its boundary points with a radius equal to 1.1 Å in the scaled ligand grid. The resulting 3D point cloud was stored as the final ligand mask representation and was named **qRankMask**. By doing this expansion on the "fine ligand *blob* 3D representation", instead directly on the scaled ligand grid representation at 0.95 contour (no filters), we could prevent distant noisy points from being included in the qRankMask and further in the final representations of the ligands.

Creation of the final representations of the ligands in 3D point cloud (procedure 5, Fig. 2). Finally, the 3D representations of the list of valid ligands in 3D point cloud were created. Nine types of 3D representations were generated per ligand entry by exploring different contour levels. All of them compose *LigPCDS*. The representation types were named: qRank0.5, qRank0.7, qRank0.75, qRank0.8, qRank0.85, qRank0.9, qRank0.95, qRankMask, and qRankMask_5. These fine sliced 3D point clouds were obtained by applying, to the ligand mask representation (qRankMask), contours at 0.5, 0.7, 0.75, 0.8, 0.85, 0.9 and 0.95 on the quantile rank scale. The different contours used are related to the representation name suffix. These point clouds have as a single feature the scaled density value of the qRankMask normalized again from 0 to 1, where each contour value is the new 0 in the final representation. For qRankMask_5 a different approach was used, aiming to join types qRank0.5 and qRankMask which gave better results in the models training: values below 0.5 were set to 0 in the qRankMask, and all the normalized values of contour 0.5 were directly used as feature. In other words, week points (below 0.5) were clipped.

The ligand mask representations (qRankMask and qRankMask_5) and the representations with a quantile rank contour ≤ 0.8 (qRank0.5, qRank0.7, qRank0.75, qRank0.8) gave better results when training the validated deep learning models, with a very small difference between their accuracies. The representation qRankMask_5 was chosen as the best result for the validated segmentation models; it maintains the ligand mask shape with good accuracy. Depending on the usage goals of this dataset, different representation types may give the best results.

A total of 244,283 ligand entries, 12,239 being unique ligands, had their **final 3D representations** successfully created. The first and fourth columns of Fig. 3 show the final 3D point clouds of two different ligands in four different representation types and the ligand mask. This figure illustrates the impact of the contour value on the final 3D point cloud of the ligands.

The mean time to create the ligand grid representation in 3D point cloud was 0.33 seconds per ligand. The mean time to create all representation types was 0.39 seconds per ligand (mean time for a spacing of the points equal to 0.5 Å). Other ways to create the 3D representations of ligands in 3D point clouds may also be tested in the future. This work provides one of the possible frameworks of functions to create 3D representations of protein ligands in 3D point clouds (imaging approach), which were successfully tested to be used in ML approaches.

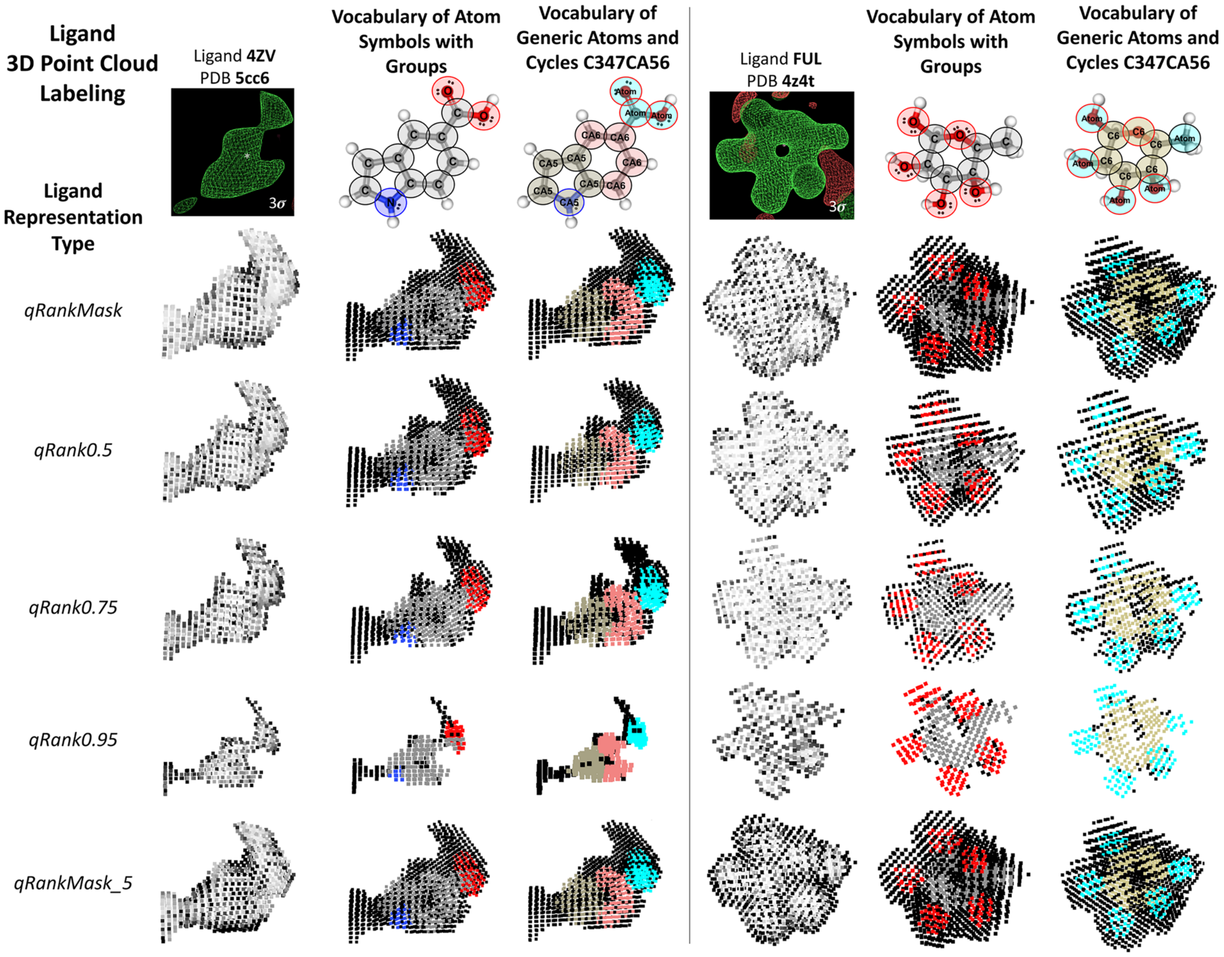


Fig. 3 Example of a ligand's 3D point cloud labeling for five different representation types. Two ligands are used for illustration: 4ZV (PDB entry 5cc6, resolution 2.1 Å) and FUL (PDB entry 4z4t, resolution 1.8 Å). Their *blobs* from their Fo-Fc maps are shown in the top of the panel with a contour of 3σ (image created with Coot). The *LigPCDS* visualization script was used to draw the ligands' 3D point clouds. For ligand FUL, it is possible to see the pattern of a ring in the qRank0.95 representation; it results from the cyclic substructure of size six, present in its structure. In ligand 4ZV this pattern is not clear, possibly due to the mobility of this molecule - which is indicated by the presence of noise around its image (*blob*) and its representations (bottom left and top right of the ligand region – black points labeled as background). Furthermore, the qRank0.95 representation of ligand 4ZV is partially fragmented, with missing points, while for ligand FUL all points with labels are completely covered. There is more visual correspondence between the ligand's image in the 3σ Fo-Fc maps and the qRank0.95 point cloud.

Chemical vocabularies and ligand structure labeling.

Chemical vocabularies were designed (step 3, Fig. 1) to compose the building blocks to label the created 3D representations of ligands in 3D point clouds from *LigPCDS*. The set of uniquely used labels is referred to as vocabulary and the unique labels are referred to as classes.

Data labeling can be very difficult depending on the amount of data and on the availability of validated references[37]. The labeling in *LigPCDS* was designed to first label the ligand's structure atom-wise with building blocks (classes) and then to extrapolate it to the ligand 3D representations (the ligand chemical structure – next subsection). The implemented structure labeling approach was inspired by ML solutions that model chemical structures of small molecules for drug design[38].

Four simplified chemical vocabularies were designed and validated (please see Technical Validation section) for labeling the ligand's structure (Table 1). They are based on the atom's symbol (the atom itself), which represent the individual scattering contribution of each atom to the electron density map; and on cyclic structures information, which adds a layer of 3D spatial distribution and geometrical restrains for the ligand region, and consequently to the *blob* region. All vocabularies also contain the background class, which represents non-atom regions of the ligands, and is only used in the labeling of the ligand 3D point cloud.

The four valid vocabularies designed are simplifications of two major labeling approaches: i) the AtomSymbol-based, with the chemical symbol of organic atoms (e.g. C, O, N, P, S, Se, Br, Cl, F, I); and ii) the SP-based, with the SP hybridization attributed to each atom (e.g., sp, sp2, sp3, sp3d1, sp3d2, sp3d3), which is defined by the atom steric number. The cyclic structure arrangement information is also included in both Atom

Vocabulary	Labeling Approach	d_{max}	Classes	Number of Classes
Ligand Region	SP	1	Background, Atom	2
Generic Atoms and Cycles	SP	2.1	Background, Atom, C (Cycle – generic cyclic structure)	3
Generic Atoms and Cycles C347CA56	SP	1,535.2	Background, Atom, C5 (Cycle of size 5), CA5 (Aromatic Cycle of size 5), C6, CA6, C3, C4, C7	9
Atom Symbols with Groups	AtomSymbol	41.4	Background, C, O, N, PSe, Halo	6

Table 1. Description of the four valid vocabularies. All valid vocabularies are presented with their maximum imbalance ratio (d_{max}) in the valid ligands list, their classes names and size.

Symbol and SP hybridization labeling. Please refer to Supplementary Note 1 for more information about the process in designing the chemical vocabularies. A brief explanation of the four valid vocabularies, which are directly mapped from the major labeling approaches, is given below and is summarized in Table 1:

I) "Vocabulary of the Ligand Region" (SP-based, 2 classes): labels all atoms with the generic atom class;
II) "Vocabulary of Generic Atoms and Cycles" (SP-based, 3 classes): labels the atoms as generic atoms outside cyclic structures and atoms in generic cyclic structures (of any size and type);
III) "Vocabulary of Generic Atoms and Cycles C347CA56" (SP-based, 9 classes): labels the atoms as generic atoms outside cyclic structures and atoms in cyclic structures with sizes (ranging from 3 to 7), where cyclic structures with sizes 5 and 6 are further labeled according to their aromaticity (aromatic or not). Aromatic cyclic structures of sizes 4 and 7 are not distinguished from non-aromatic ones due to their low abundance. Cyclic structures with more than 7 atoms are not distinguished from atoms outside cyclic structures as large cyclic arrangements are more flexible and may not have a shape pattern in the Fo-Fc map;
IV) "Vocabulary of Atom Symbols with Groups" (AtomSymbol-based, 6 classes): labels the ligand atoms with their chemical symbol, if it is one of the most common atom symbols in organic molecules (C, O, N); or with the following groupings: the "halo" group, if it is a halogen atom (atom symbols F, Cl, Br and I), and the "PSe" group, if it is one of the remaining atoms with lower abundance in the dataset (atom symbols P, S and Se).

The ligand structure labeling procedure was automated in a Python script with the RDKit package v2019.09.3 and was used to implement both the AtomSymbol-based and SP-based approaches. It works as follows. For each ligand: (i) all cyclic structures in the ligand structure are retrieved; (ii) for each atom of the ligand, its label is set to its SP hybridization (one of sp, sp2, sp3, sp3d, sp3d2, sp3d3), or its atom symbol (one of C, O, N, P, S, I, F, Se, Cl and Br), depending on the parameters. This label is concatenated with the smaller cyclic structure in size and aromatic cyclic arrangement type in which this atom appears (one of C3, CA4, C4, CA5, C5, CA6, C6, CA7 or C7 in this order), if any. Finally, (iii) the labels of all atoms are returned. The labels are mapped to the atoms using their unique coordinates in the 3D space.

These two major approaches (AtomSymbol-based and SP-based) were used to label the structures of the ligands in the list of valid ligands, resulting in 244,226 ligand structures successfully labeled. The ligands structural labeling results were saved to tables in .xyz files (CSV format), with one atom per row and their information and label by column. These results were stored in the xyz directory of the data record of each major approach: SP-based and AtomSymbol-based labeling (detailed in the Data Records section). The mapping from the two major approaches to the four validated vocabularies was performed by matching their labels with the provided mapping tables presented in Supplementary Tables 2, 3 (see Usage Note for more details). Examples of structure labeling with these two major approaches and their four mapped and validated vocabularies are illustrated for the molecules beta-L-fucose and 1*H*-indole-5-carboxylic acid, which have the following ligand codes in PDB: FUL and 4ZV, respectively (Fig. 4).

The four valid vocabularies are further described with the distribution of occurrences of their classes by atom in the final list of valid ligands (Figs. 5 and 6). These distributions help visualize the class imbalance problem[39,40] present in *LigPCDS*, a crucial information to understand its limits for semantic segmentation tasks. Also, the maximum imbalance ratio[40] (d_{max}, Eq. 1) was computed to indicate, for each vocabulary, the maximum level imbalance across classes and to help the comparison of the viability of the different vocabularies.[39,40]

$$d_{\max} = \frac{\max\limits_{i}\{C_i\}}{\min\limits_{i}\{C_i\}} \tag{1}$$

where C_i is the number of atoms labeled as class i, and $\max\limits_{i}\{C_i\}$ and $\min\limits_{i}\{C_i\}$ are the maximum and minimum number of labeled atoms among classes, respectively.

Figure 5 displays the distribution of class occurrences using the SP-based vocabularies for the atoms of the ligands in the list of valid ligands. Figure 6 displays this distribution for the AtomSymbol-based vocabularies.

The two vocabularies that kept more chemical information and had good accuracy in relevant classes of the validated models (please see Technical Validation) were selected as the **best labeling approaches**: the "Vocabulary of Generic Atoms and Cycles C347CA56" and the "Vocabulary of Atom Symbols with Groups".

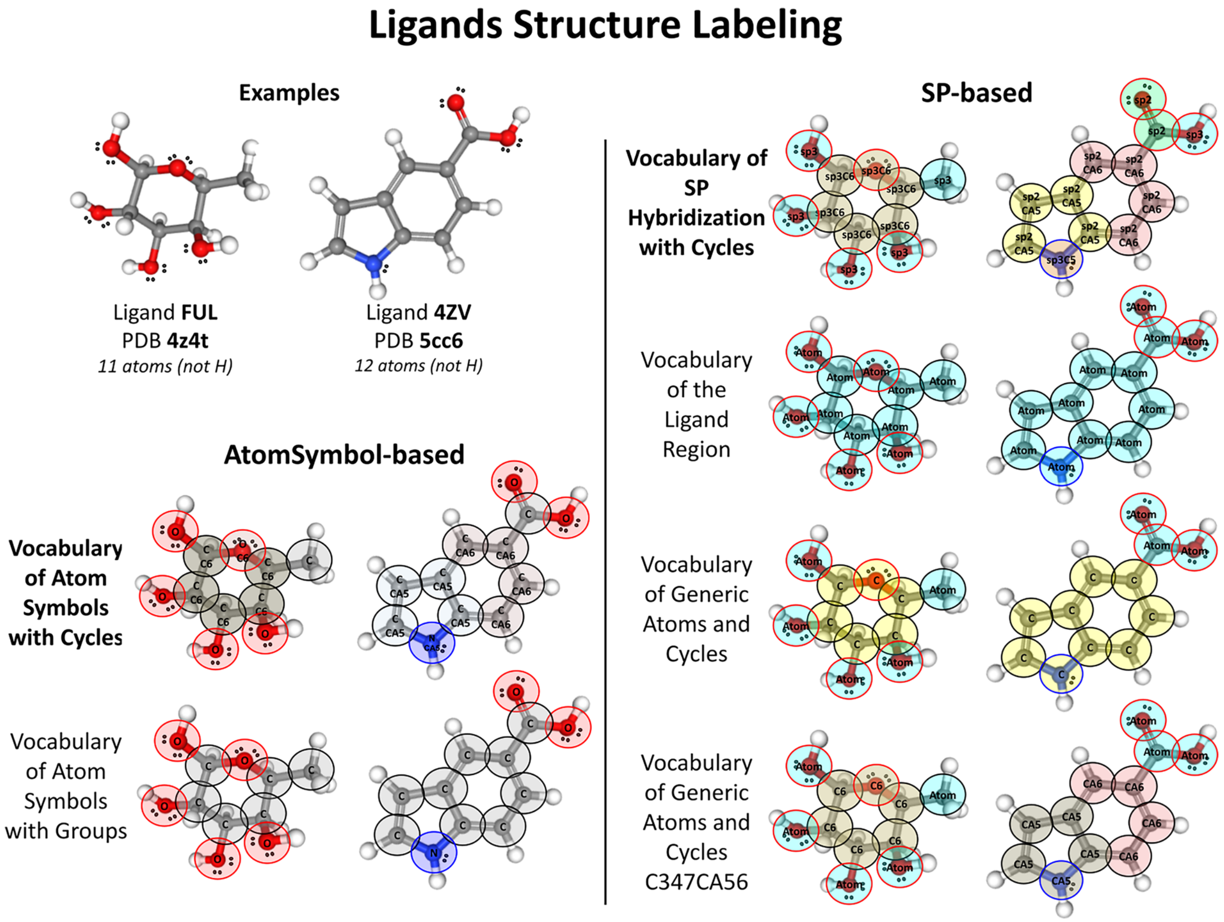

Ligands Structure Labeling
Examples
Ligand FUL
PDB 4z4t
11 atoms (not H)
Ligand 4ZV
PDB 5cc6
12 atoms (not H)
AtomSymbol-based
Vocabulary of Atom Symbols with Cycles
Vocabulary of Atom Symbols with Groups
SP-based
Vocabulary of SP Hybridization with Cycles
Vocabulary of the Ligand Region
Vocabulary of Generic Atoms and Cycles
Vocabulary of Generic Atoms and Cycles C347CA56

Fig. 4 Examples of a ligand's structure labeling. Ligands 4ZV and FUL from PDB (entries 5CC6 and 4Z4T, respectively) are shown on the top left panel and were used as example to illustrate all the proposed vocabularies: the "Vocabulary of SP hybridization with Cycles" and its three mappings (SP-based approach), which are shown on the right panel; and the "Vocabulary of Atom Symbols with Cycles" and its mappings (AtomSymbol-based approach), which are shown on the bottom left panel. The label of each atom is written inside its atomic sphere (represented by a circle), which is colored according to its label in the filling and the border color received the atom color in the 2D structure.

Labeling of the final representations of the ligands in 3D point clouds. The last step to obtain *LigPCDS* (step 4, Fig. 1; procedure 6, Fig. 2) is the pointwise labeling of the final representations of the list of valid ligands in 3D point clouds. This was done with the atom-wise extrapolation of the labels of the ligands' structures (previous section) to their final representations in 3D point clouds.

A widely used model to calculate the atomic volume of molecules is to treat atoms as rigid spheres[41]. These spheres have a radius equal to the van der Waals theoretical atomic radius for each atom type, and serve as a model to represent the electron density volume that would be occupied by each atom of the molecule. The electron density is theoretically distributed as a Gaussian centered on each atom[41], with high intensity values at the center. When a contour is applied to the electron density (*e.g.* in the sigma or quantile rank scale), only the central peak of each Gaussian is visible[42]. Batsanov's work[31] summarizes the available data on the van der Waals theoretical atomic radius for molecules and crystals. The work that describes XGen[42], for fitting ligands in the real space of electron density maps, provided information for the typical experimental X-ray radius for organic elements at different experimental electron density resolutions.

It was thus decided to use the modeling of an atomic sphere to extrapolate the labeling from the atoms of the ligands structure to their final 3D point clouds, using as radius 65% of the experimental radius provided by XGen for each atom type. This percentage was chosen to recover the central region of the density peak of each atom, while keeping the contour of the ligand's structure. The resolution of the PDB entries was used to select the sets of radii for each ligand entry, rounding the resolution to the first decimal place (values tabled for resolutions 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1 and 2.2 Å in Supplementary Table 1). For Selenium (Se) atom, which does not appear in the XGen table, it was assigned the radii of the Bromine (Br) atom. The points in the representation of the ligands that are not covered by the atomic spheres with 65% of the experimental radius of XGen received the labeling of background noise ("background" class – regions in the Fo-Fc map without a ligand atom). Points in the intersection region of two or more atomic spheres received the label of the nearest atom center. Other

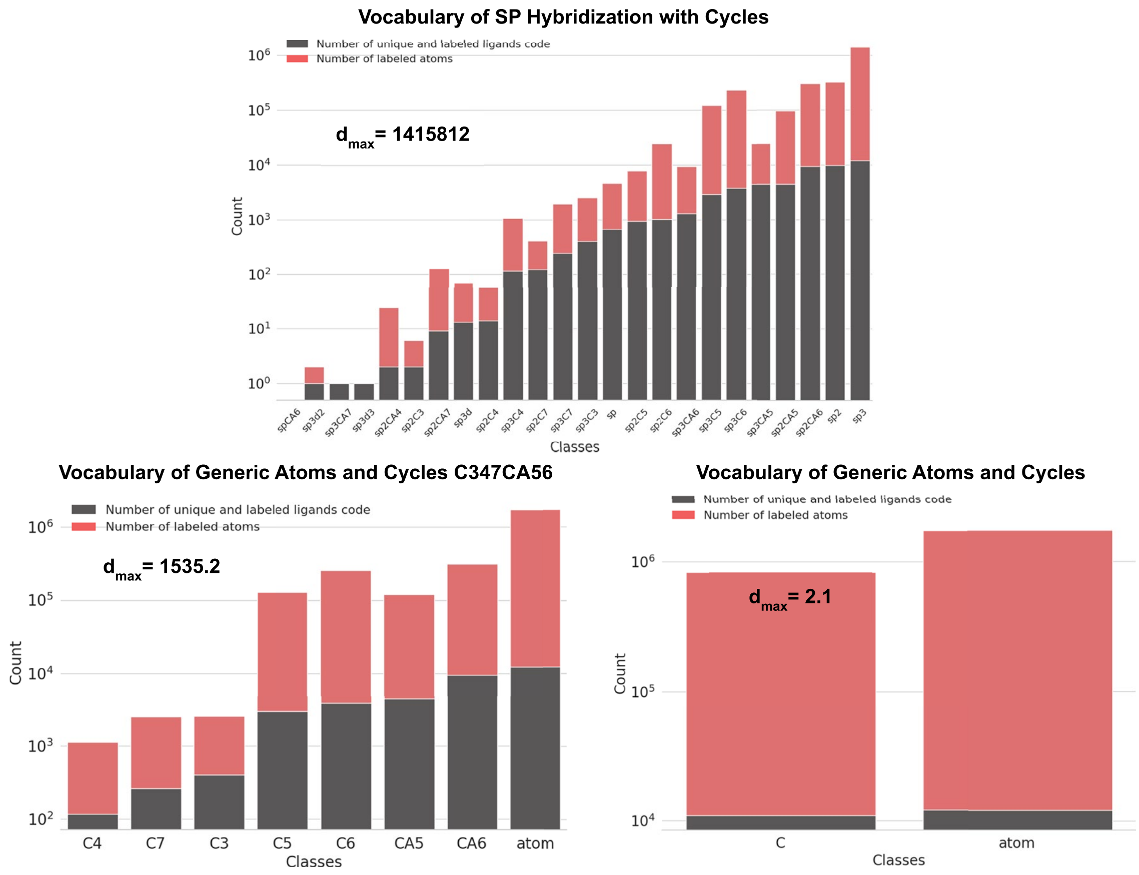


Fig. 5 Class distribution of SP-based vocabularies. Distribution of the class occurrence in the SP-based vocabularies by labeled atom of all entries of the final list of valid ligands. Their corresponding imbalance ratio (d_{max}) is also presented. The distribution for the "Vocabulary of the Ligand Region" is omitted because all 2,566,614 atoms were labeled with the same generic class of atoms. The background class is not used in these distributions.

percentages of atomic radii were not tested. This procedure was implemented with the functionality of the Open3D v0.12 library for quick access of the neighborhood of each point.

The ligand's structure labeling was extrapolated to the final 3D representations of the ligands present in the list of valid ligands using the two major labeling approaches (SP-based and AtomSymbol-based). A dataset of labeled 3D representations of the difference electron density of ligands in point cloud was obtained for each major vocabulary. The ligand final 3D point clouds that were correctly labeled and tested constitute 244,226 entries in the final list of valid ligands. The point cloud labeling testing is detailed in the Technical Validation section.

The labeled records of ligand images in 3D point cloud representations were called "LigPCDS-SP" and "LigPCDS-AtomSymbol", which correspond to the SP-based and AtomSymbol-based labeling approaches, respectively, and compose *LigPCDS*. This dataset covers entries of free protein ligands of organic molecules (non-covalent protein ligands composed by C, O, N, P, S, Se, F, Cl, Br or I atoms), obtained from X-ray protein crystallography, with experimental resolutions ranging from 1.5 to 2.2 Å. These records (SP-based and AtomSymbol-based) are organized by PDB entry and contain all the final 3D point clouds of the list of valid ligands that appear in the respective entry. The organization of this dataset is detailed in the Data Records section. Two examples of final labeled 3D point clouds of ligands with the "Vocabulary of Generic Atoms and Cycles C347CA56" and the "Vocabulary of Atoms Symbols with Groups" are presented in Fig. 3 for different representation types.

Data Records

The *LigPCDS* dataset v1.0.1 is stored in the Zenodo repository (https://doi.org/10.5281/zenodo.15174758)[43] from CERN, the European Organization for Nuclear Research. The LigPCDS-SP and LigPCDS-AtomSymbol records were deposited in separated files named LigPCDS-SP_record and LigPCDS-AtomSymbol_record, respectively, in.zip format, each one containing:

- A zipped file with the dataset of labeled ligand representations in 3D point cloud, named *LigPCDS-SP* or *LigPCDS-AtomSymbol*, concatenated with the string "_reso-1.5-2.2_gridspace-0.5.zip" (resolution range and

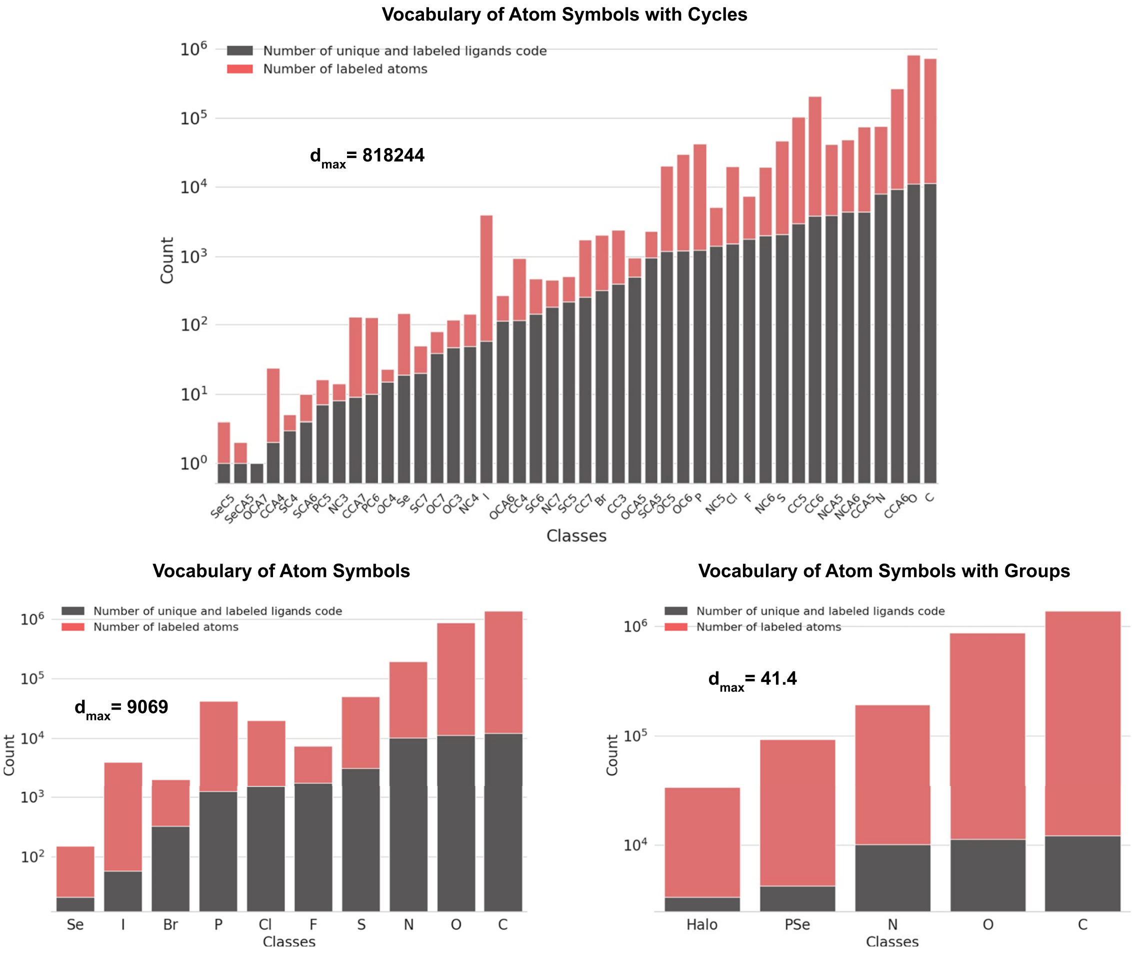


Fig. 6 Class distribution of AtomSymbol-based vocabularies. Distribution of class occurrence in the AtomSymbol-based vocabularies by labeled atom of all entries of the final list of valid ligands. Their corresponding imbalance ratio (d_{max}) is also presented.

grid spacing used). This folder is organized with a subfolder for each PDB entry, named with the respective PDB entry ID, which contains:

- The 3D point cloud of all the valid ligands that appear in the respective PDB entry for all the nine final representation types (qRank0.5, qRank0.7, qRank0.75, qRank0.8, qRank0.85, qRank0.9, qRank0.95, qRankMask, and qRankMask_5) and their corresponding labels. The format of these files are as follows:
    - The ligand 3D representations in point cloud are stored in .xyzrgb files in CSV format and are named with a unique ligand ID equal to the PDB entry ID concatenated with the ligand code, the chain code and the residue number in which it appears in the respective protein structure. This filename is followed by the string "lig_point_cloud_fofc" plus the representation type. These .xyzrgb files contain:
        - The x,y,z position of each point of the 3D point cloud in the first three columns and their feature in the following three columns (color channels). These feature columns have the same values.
    - The labels of the ligand 3D point clouds are stored in.txt files with the index of the vocabulary class of each point by row, in the same order of the points of the representation. The vocabulary index ranges from 0 to the number of classes of the respective vocabulary minus one, or "−1" to indicate the "background" label. These files are named with a unique ligand ID equal to the PDB entry ID, the ligand code, the chain code and the residue number in which it appears in the respective protein structure. The string "lig_pc_labels" plus the representation type completes the file name. The representation type qRankMask_5 uses the same label file named with the qRankMask type.

- One folder with the proposed vocabularies and mappings named as "vocabulary_" followed by the respective labeling approach, SP or AtomSymbol, containing:The major vocabulary used to label each dataset in a.txt file containing one class by row. The order of the classes by row defines the classes order and index, which starts with 0 in the first row and ends with the size (number of classes) of the vocabulary minus one. Only the "background" class is not present in the vocabulary file and will always receive an index equal to −1 in the ligand labels files. The vocabulary files are named with the prefix "vocabulary_valid_ligands_PDB_1.5_2.2_" followed by a suffix equal to the labeling approach "SP-based" or "AtomSymbol-based".

    - The mapping tables for the valid vocabulary are in CSV files and are named with the prefix equal to "mapping_", followed by a suffix equal to the vocabulary name. These tables contain one column named "source", with the index of the source class in the respective major vocabulary based (SP or AtomSymbol) and another column named "target", with the index of the target class in the new mapped vocabulary. In these mapping tables the "background" class receives a source index equal to the size of the respective vocabulary instead of −1, to facilitate the mapping (explained in the Usage Notes). Additionally, there are two columns named "classes" and "mapping", that contain the classes names of the source and target vocabularies, respectively.

- Another folder named "ligands_lists" containing three tables with a list of ligands:

    - One table is a CSV file with the final list of valid ligands and their classes count in the respective major labeling approach. This table is named with the prefix "valid_ligands_list" followed by the filters used to select this set of entries and a suffix equal to the base vocabulary used (SP or AtomSymbol). This table contains one ligand by row and their information by column, such as: ligand code and ID, PDB entry ID, PDB entry resolution, global B factor and the vocabulary classes count by labeled atom (these columns are detailed in the table listed below).
    - Another table in CSV file describing the columns in the table of the final list of valid ligands named "valid_ligands_list_columns_description.csv". It contains two columns: one named "column_name" which contains the names of columns of the valid ligands list table, and another column named "column_description" with the description of the named columns.
    - A table with the stratified training dataset named as "training_dataset_valid_ligands_undersampling_maxLigCode_1000_kfolds_13_gridspace_0.5_" followed by the base labeling used (SP or AtomSymbol).

- A zipped file containing the xyz directory with the ligands structure labeling result for the list of valid ligands. This file is named with the prefix "xyz_" followed by the filters used to create the list of valid ligands and a suffix equal to the base labeling approach (SP or AtomSymbol). The xyz directories contain one .xyz file for each labeled ligand entry which is named with the ligand ID followed by "_class.xyz".
- A zipped file with the validated DL models of each labeling approach, named with a prefix equal to "DL_models_" followed by SP or AtomSymbol. The DL models are stored in checkpoint files (.ckpt) named with the model name followed by the tag "ligs" and the number of ligands used in the training dataset, the tag "img" and the representation type used, the tag "gridspace" and the grid spacing used and the tag "k" followed by the subset k used for test and validation.

    - There is also a metadata table for each model describing some of the training setup (e.g. number of epochs) of each DL model.

The ligands grid representation of the list of valid ligands is also available as a different zipped file named "LigPCDS-Grids_reso-1.5-2.2_gridspace-0.5.zip". It contains one subfolder for each PDB entry containing their ligand grid representations in .xyzrgb files.

- The ligand grid files are named with the ligand ID followed by the string "grid_point_cloud_fofc.xyzrgb".

Technical Validation

Structure labeling automatic test.

An automatic test based on reverse engineering was implemented in the algorithm that labels the ligand structure (step 3, Fig. 1). This test increased the structure labeling quality in the final list of valid ligands by using the SMILES (Simplified Molecular Input Line Entry System) of the ligands present in their retrieved SDF files, as a ground truth for their chemical structure. SMILES is commonly used in chemistry to write the chemical structure of molecules in a simplified and short ASCII string. The structure labeling automatic test uses the SMILES of the ligand entries to retrieve the expected labeling for their structure and then verifies if it matches the structure labeling from their deposited chemical structure (3D atomic positions and chemical bonds) defined in their SDF files. The script validates the set of returned classes and, if any conflict is found, it marks the ligand entry with an error tag. Ligands with mismatching labels between their SMILES and deposited chemical structure in the SDF files were not included in the final list of valid ligands. This automatic test of the structure labeling procedure removed 588 ligand entries (0.2% of the initial list with 293,822 ligands). Most errors were due to wrongly defined SDF files (e.g. a chemical bond defined between wrong atoms) or missing atoms in the deposited structure that affected the labeling and prevented the match.

Structure labeling manual test. Automated labeling is a very error-prone process, as variations in the data that deviate from the expected structure can generate undesirable behavior in the algorithms and create labeling noise in the dataset. To increase the quality of the algorithms used in the ligand's structure labeling, automated case tests were performed with 8 manually labeled structures of ligands. The ligand's structures were labeled with the SP-based and the AtomSymbol-based approaches. The chosen ligands that compose the list of test cases have the following codes in PDB: 0YB, 1EJ, 58 T, DJ4, I3C, MB5, MTE and Q0S. The choice for these ligands sought to cover a wide range of classes from the proposed vocabularies in different chemical arrangements. This test automatically compares the automatic structure labeling result against the manually labeled structures, defined as the truth table of each test case.

More ligands may be manually labeled and added to the list of structure labeling test cases. The user must follow the format expected by the testing script to correctly evaluate the new structures. All the ligands present in the list of structure labeling test cases are automatically tested against the automatic labeling function. This ensures the correctness of the algorithms in the manually labeled structures.

Point cloud labeling test. A test to verify the creation and labeling of the final 3D point clouds of ligands was also implemented. It checks if all the points of the final representations of the ligands, that are covered by 1/4 of an atomic sphere of its structure, have the same label as the respective atom label. The ligand entries that had their final 3D point cloud created and that raised a mismatch in this checking were removed and not included in the final list of valid ligands. This represented only 57 entries (0.02% of the final point clouds created). The remaining 244,226 entries compose the final list of valid ligands.

Stratified training dataset. An undersampling technique[39] was applied to the final valid ligands list (step 5, Fig. 1) to create a stratified training dataset that deals with the imbalance problem present in the classes distribution (Figs. 5, 6) and in the chemical diversity of the ligands structure (many repetitions of few common structures, more than one class by entry and with both very frequent and rare classes).

The undersampling in the list of valid ligands was implemented to remove noise from non-relevant structures (few atoms) or tiny point clouds (having a small size), and to prevent losing entries with rare classes or including many repetitions of the same structure. It works as follows: (1) it removes all entries with less than the minimum number of atoms (set equal to 4); (2) it removes all entries with "qRank0.95" point cloud size smaller than 150 points; and, (3) it removes the entries of frequent ligand structures (maximum occurrence by ligand code was set to 1,000). By using an anti-clustering method[44], this maintained the diversity among entries related to the following selected characteristics: B factor, minimum occupancy, resolution and size of the ligand mask point cloud. At the end of this process, a stratified training dataset with 78,902 entries of ligands from 26,976 PDB entries and 11,925 unique structures was obtained.

Subsequently, this training dataset was partitioned in train, test and validation datasets using a cross-validation (CV) technique[45] to avoid overfitting[46]. The CV implemented to partition the training dataset was the k-fold cross-validation. The average performance of the model trained on each subset is the performance of the CV[46].

To ensure diversity in the k-fold subsets, a stratified separation was performed in the training dataset using the anti-clustering algorithm[44]. It partitioned the ligand entries into k similar groups, maintaining, within each group, a diversity of entries in relation to the same selected characteristics. Furthermore, each group was further separated into two subgroups referring to the test and validation set using the anti-clustering algorithm with the same selected characteristics. Due to time constraints for training a deep learning model, only a subset k was selected for testing and validation in most training jobs, and the applied CV method was the "hold-out"[45].

The 78,902 entries of the stratified training dataset of ligands were partitioned in $k=13$ similar groups. Each group was also partitioned into two other subgroups corresponding to the entries selected for validation and test of the respective group. Thus, for each k, 72,833 ligand entries are used for training, 3,034 for validation and 3,035 for testing. These values may vary by at most 2 units depending on the group.

The four mapped vocabularies had $d_{max} < 2000$ (Table 1) and were selected as viable for training: "Ligand Region", "Generic Atoms and Cycles", "Generic Atoms and Cycles C347CA56" and "Atom Symbols with Groups". The occurrence distribution of the classes in these four viable vocabularies were recomputed using only the labeled atoms present in the stratified training dataset. These distributions are presented in Fig. 7 together with their new maximum imbalance ratio (d_{max}) in the stratified training dataset. The distribution for the "Vocabulary of the Ligand Region" is omitted again because all 1,671,853 atoms were labeled with the same generic class of atoms. The background class is not used in these distributions.

These four vocabularies continued to show a viability for training a DL model ($d_{max} < 1000$) using the obtained stratified training dataset. Only the "Vocabulary of Atom Symbols with Groups", which had a $d_{max} < 50$, had an increase in its imbalance ratio. This may be due to rare atoms that repeatedly appear in the same ligand structure (same ligand code) and these frequent structures were limited by the undersampling procedure (i.e. chlorine ions were removed), but this did not affect the viability of the respective vocabulary.

Deep learning architecture, training pipeline and evaluation metrics. The Minkowski Engine (ME)[47], an open-source deep learning architecture for sparse tensor pointwise convolution in 3D point clouds, was used for training semantic segmentation tasks with *LigPCDS* (step 6, Fig. 1). A hybrid dilated (or atrous) 3D convolution[48–50], called "MinkUNet34C_CONVATROUS_HYBRID", was implemented by modifying the provided "MinkUNet34C" network, which was used in semantic segmentation tasks with good results[51]. These networks are based on the 3D U-Net network[52], they accept inputs of different sizes and are illustrated in Supplementary Figure 2.

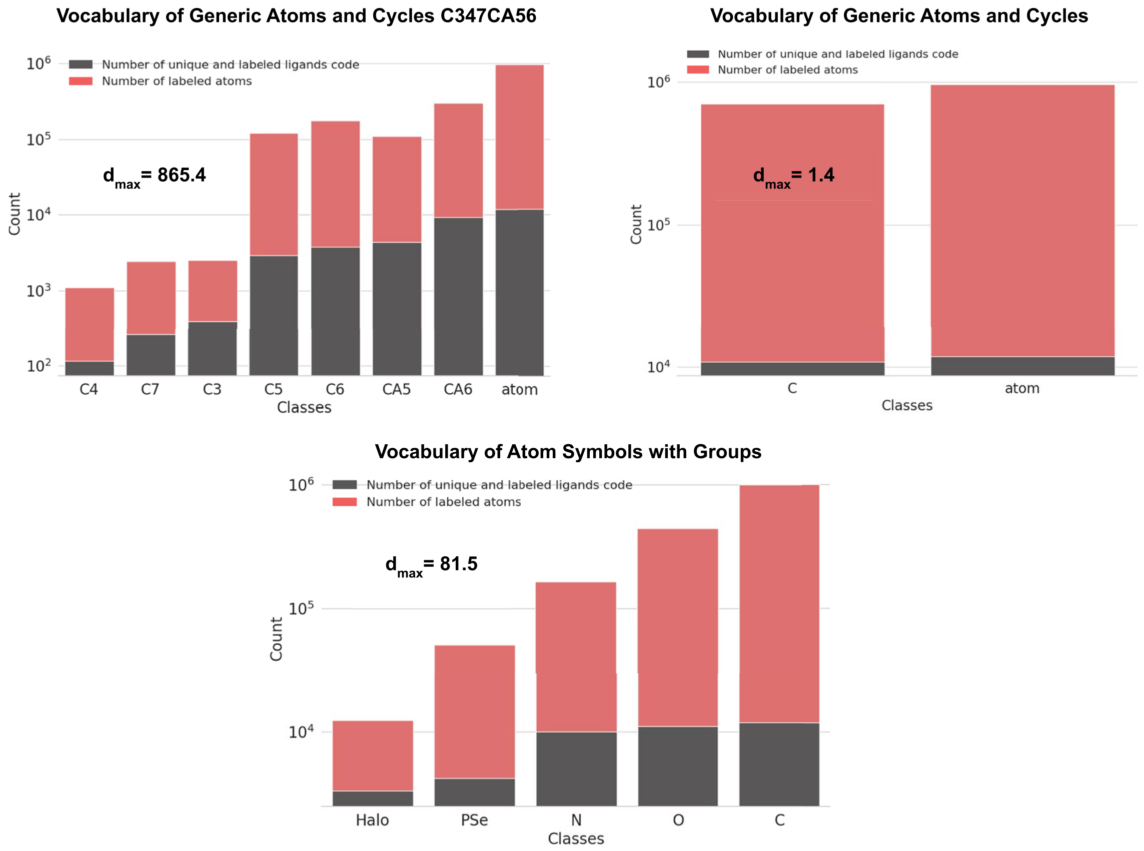


Fig. 7 Class distribution of the viable and validated vocabularies in the stratified training dataset from *LigPCDS*. Data shown refers to the distribution of class occurrence in the viable vocabularies by labeled atom on the entries of the stratified training dataset. Their corresponding imbalance ratio is also displayed.

LigPCDS training pipeline was implemented with the pytorch-lightning Python library[47,53]. The user selects the value of k to be used in the training and the entries in the k-th group are used for the validation and test sets. The remaining entries are used for the train set.

The hyperparameters implemented and evaluated in the training pipeline are (with the value that gave the best results between brackets): the total batch size (16 entries), the loss function (wSL - weighted Symmetric cross entropy Learning[54]), the loss weight (yes), the random rotation rate (50%), the deep neural network (MinkUNet34C_CONVATROUS_HYBRID) and the optimizer function (SGD - Stochastic Gradient Descent[55]). The training parameters related to the ligand 3D point clouds that were evaluated are (best value between brackets): the representation type (qRankMask_5) and the representation spacing (0.5 Å). The best setup is summarized in Supplementary Table 4 and detailed in Supplementary Note 2 and Supplementary Note 4. The main evaluation metric used in this project is the Intersection over Union (IoU)[27]. The mean IoU (mIoU) of all classes is used as a global training metric[51,56]. Additionally, the F1 score (or Dice Coefficient)[28] and the Recall and Precision rates by class are also computed. These metrics are computed using the accumulated hit and missing points of all entries in the respective set. The 95% bootstrap confidence interval (CI)[57,58] of the evaluated metrics in the test set was also computed for individual predictions, together with the standard error of the mean (SEM), using the evaluation by entry. More details on these metrics are presented in Supplementary Note 3.

Validation of DL models and best results.

The four viable vocabularies were validated by training good performance DL models for the semantic segmentation of the stratified dataset from *LigPCDS* (Table 2) with the best setup. The four validated DL models trained with these labeling approaches were named, respectively (and following Fig. 1c in clockwise order, starting from the top vocabulary): "LigandRegion", "AtomCycles", "AtomC347CA56" and "AtomSymbolGroups". The CV method used in the evaluation of these four models was the hold-out with $k = 1$, except for the model AtomC347CA56 which used the k-fold cross validation method (evaluated using hardware B). The value $k = 13$ gave the best result in the k-fold CV of model AtomC347CA56 (Supplementary Table 5). The average time to train the DL models in the best setup and using two GPUs of hardware A was of about 4.5 hours for each 10 epochs.

The confusion matrices of the test of these four validated DL models are presented in Fig. 8. This matrix summarizes the hits and errors of the models by class using the IoU metric. This data helps visualize the impact of the class imbalance problem on the model's performance. It allows checking the confusion between classes,

DL Model	d_{max}	Epochs	Test mIoU	F1 score	Precision	Recall
Ligand Region k = 1	1	120	77.38 0.22 [−11.7,12.1]	86.96 0.16 [−8.4,8.8]	86.53 0.16 [−8.7,9.1]	87.42 0.14 [−7.8,8.2]
Atom Cycles k = 1	1.4	120	70.95 0.30 [−16.3,17.1]	82.49 0.27 [−14.7,15.6]	80.46 0.26 [−13.7,14.5]	84.86 0.22 [−11.7,12.6]
Atom C347CA56 k-fold*	865	200	49.66 0.36 [−19.4,20.2]	62.41 0.35 [−18.8,19.7]	58.2 0.29 [−15.7,16.6]	74.09 0.28 [−14.9,15.8]
Atom Symbol Groups k = 1	81.5	160	59.03 0.37 [−19.8,20.5]	73.16 0.36 [−19.6,20.3]	68.67 0.35 [−18.8,19.5]	79.61 0.29 [−15.4,16.2]

Table 2. Test evaluation of the four viable and validated models. Results from the hold-out CV against 3,035 ligands from the k = 1 test subset of the stratified training dataset, except for model "AtomC347CA56" which results from the k-fold cross validation method. The imbalance ratio (d_{max}) was recomputed for the stratified training dataset. The standard error of the mean (SEM) is provided below the overall for each metric and their confidence interval is provided between square brackets, both computed for individual predictions. *The SEM and confidence intervals for model AtomC347CA56 were computed using k = 13.

helps to understand the errors that are occurring and the impact of minor or difficult to converge classes[59]. Additionally, the distribution of the overall accuracy of the validated models is presented in Fig. 9, showing a higher concentration of values above 80%.

The learning curves of the different classes are shown for all models together in Supplementary Figure 3. The loss weights used in each model training to deal with the imbalance between classes are presented in Supplementary Note 4. The reasoning for the groupings proposed for the vocabulary "Atom Symbols with Groups" is also presented in Supplementary Note 4 using a not validated model named AtomSymbol (from an unviable labeling used for comparison).

The four validated models ("LigandRegion", "AtomCycles", "AtomC347CA56" and "AtomSymbolGroups") had more than 49% of accuracy in mIoU and more than 62% in F1-score. Although there is no reference for an acceptable accuracy cutoff, the visual inspection of the results showed that values above 50% can still present correct predictions or very close to the expected. One reason for this low cutoff is that differences between the expected and predicted ligand conformation and atomic radii sizes may decrease the model's overall accuracy and increase the confusion with the background class (an error in the border of the ligand region), which is noted in Fig. 8 with the high values of the first column of all matrices. In other words, the correct predictions may be dislocated in space and result in lower accuracy. The other confusions presented in Fig. 8 are between very similar classes, and thus, are consistent with the proposed labeling approaches. The imbalance between the classes can greatly affect the average accuracy of the models, as is seen in the relation between their average accuracy and d_{max} (Table 2). This highlights the difficulty of dealing with rare classes and/or difficult to converge classes.

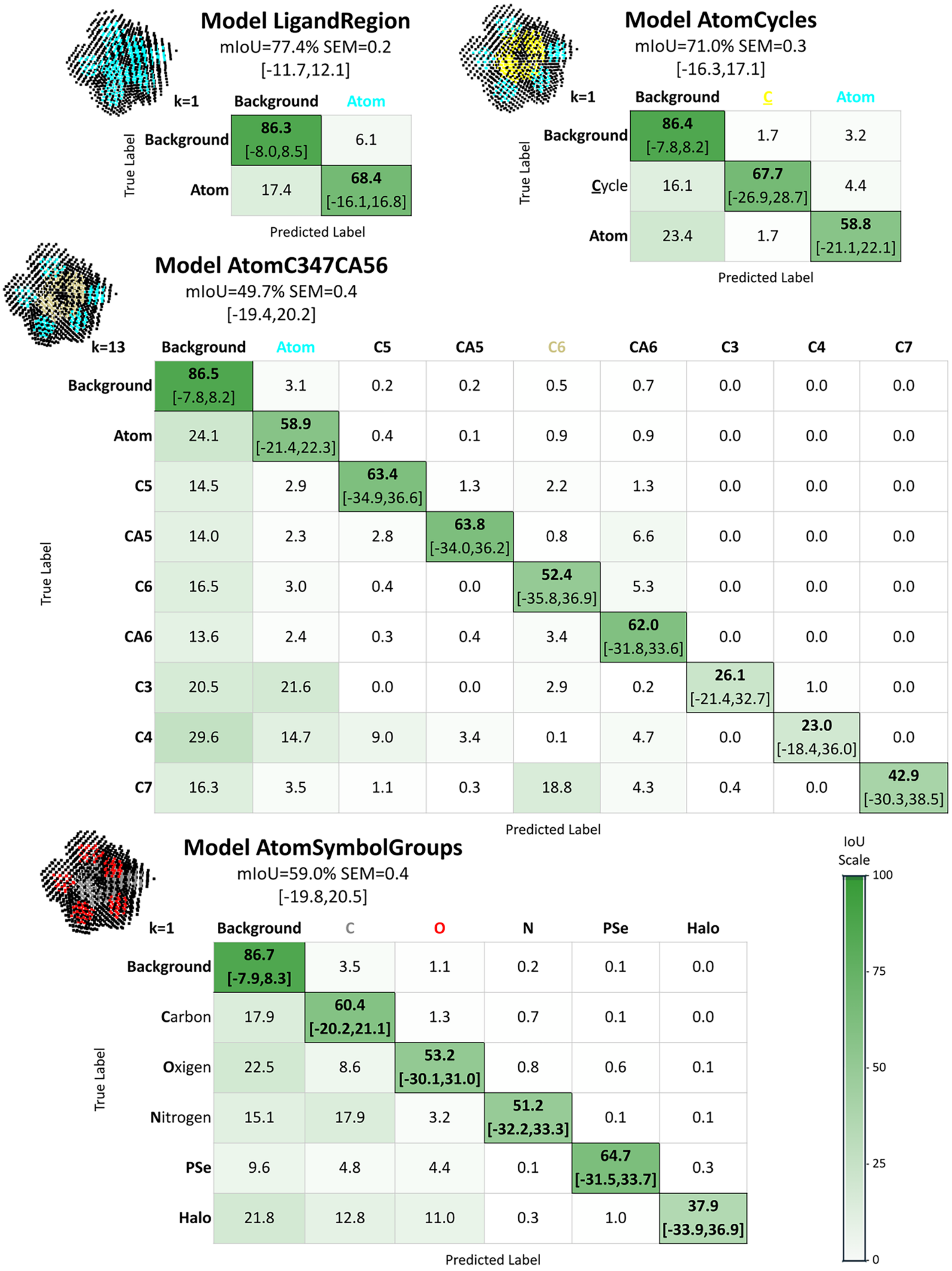
The two validated models that kept more chemical information in the vocabularies and had good accuracy in relevant classes were selected as the **best results**: the model "AtomC347CA56" and the model "AtomSymbolGroups". The first model can bring macro information about the arrangement of the ligand structure in cyclic structures, while the second model can bring micro information about its atom types. These results validated the imaging and labeling approaches of *LigPCDS*, and indicate its use by machine learning solutions. Other vocabulary mappings from the proposed labeling approach may be tested. The user may decide which data best suits their needs depending on their goals.

The two best models were also used to plot their mIoU by ligand entry in the k = 1 test subset against the resolution, size of the qRank0.95 3D point cloud and B-factor characteristics of the entries from the stratified dataset of *LigPCDS* (Fig. 10 below and Supplementary Figure 5). Lower resolutions (higher numbers meaning lower global quality), in the test range 1.5 to 2.2 Å, slowly decreased the performance of the DL models with high variance. High average B factors of the ligand entries, which is directly related to the local quality of the ligand *blob*, was the characteristics evaluated that most explained the decreasing performance of the models, still with high variance. This corroborates with the fact that the noise in experimental crystallographic data varies locally, and therefore, entries with poor global resolution may still have good predictions, if the local ligand image (*blob*) is well defined and has low noise (but this is not always the case). More details on this analysis are given in Supplementary Note 5.

Usage Notes

The ligand 3D point clouds files are in .xyzrgb format. This is a well-known and used format to store 3D point clouds, which has customized reading functions in libraries aimed to manipulate point cloud representations. In addition, the Open3D Python package may be used to read and manipulate the ligand representations.

The visualization of the 3D point clouds in *LigPCDS-SP* and *LigPCDS*-AtomSymbol can be assessed with the Python script named "visualize_*LigPCDS*", provided in the "src" folder of the "np3_LigPCDS" repository (open access described in the Code Availability section). This script will render, for each ligand ID present in the dataset (user provided), the 3D point clouds of its representations, further colored by the feature value of its points and for each representation type selected by the user. The representation types are separated by columns in the x-axis, with a distance equal to 2 times the x-axis size of the ligand's 3D point cloud. The representations may also be rendered in another row colored by the labeled class of each point from the ligand's label files (user provided). The rows are translated in the z-axis by 3 times the z-axis size of the ligand's 3D point cloud. The script opens a new window that contains a 3D display in the xyz space of the Open3d Python package v0.12. This display allows zoom, translation and rotation of the point



Model LigandRegion

mIoU=77.4% SEM=0.2
[-11.7,12.1]

k=1	Background	Atom
Background	86.3 [-8.0,8.5]	6.1
Atom	17.4	68.4 [-16.1,16.8]

Model AtomCycles

mIoU=71.0% SEM=0.3
[-16.3,17.1]

k=1	Background	C	Atom
Background	86.4 [-7.8,8.2]	1.7	3.2
Cycle	16.1	67.7 [-26.9,28.7]	4.4
Atom	23.4	1.7	58.8 [-21.1,22.1]

Model AtomC347CA56

mIoU=49.7% SEM=0.4
[-19.4,20.2]

k=13	Background	Atom	C5	CA5	C6	CA6	C3	C4	C7
Background	86.5 [-7.8,8.2]	3.1	0.2	0.2	0.5	0.7	0.0	0.0	0.0
Atom	24.1	58.9 [-21.4,22.3]	0.4	0.1	0.9	0.9	0.0	0.0	0.0
C5	14.5	2.9	63.4 [-34.9,36.6]	1.3	2.2	1.3	0.0	0.0	0.0
CA5	14.0	2.3	2.8	63.8 [-34.0,36.2]	0.8	6.6	0.0	0.0	0.0
C6	16.5	3.0	0.4	0.0	52.4 [-35.8,36.9]	5.3	0.0	0.0	0.0
CA6	13.6	2.4	0.3	0.4	3.4	62.0 [-31.8,33.6]	0.0	0.0	0.0
C3	20.5	21.6	0.0	0.0	2.9	0.2	26.1 [-21.4,32.7]	1.0	0.0
C4	29.6	14.7	9.0	3.4	0.1	4.7	0.0	23.0 [-18.4,36.0]	0.0
C7	16.3	3.5	1.1	0.3	18.8	4.3	0.4	0.0	42.9 [-30.3,38.5]

Model AtomSymbolGroups

mIoU=59.0% SEM=0.4
[-19.8,20.5]

k=1	Background	C	O	N	PSe	Halo
Background	86.7 [-7.9,8.3]	3.5	1.1	0.2	0.1	0.0
Carbon	17.9	60.4 [-20.2,21.1]	1.3	0.7	0.1	0.0
Oxigen	22.5	8.6	53.2 [-30.1,31.0]	0.8	0.6	0.1
Nitrogen	15.1	17.9	3.2	51.2 [-32.2,33.3]	0.1	0.1
PSe	9.6	4.8	4.4	0.1	64.7 [-31.5,33.7]	0.3
Halo	21.8	12.8	11.0	0.3	1.0	37.9 [-33.9,36.9]

Fig. 8 Confusion matrices with the test IoU evaluation of the four validated models: "LigandRegion", "AtomCycles", "AtomC347CA56" and "AtomSymbolGroups". Results from the hold-out CV against 3,035 ligands from the k = 1 and k = 13 test subsets of the stratified training dataset. It contains the expected classes (true label) by row and the predicted classes (predicted label) by column. It was normalized by row. The SEM for individual predictions is provided below the overall by model, and the CI is provided between square brackets and below the overall by class (main diagonal). For model "AtomCycles", the Cycle class is abbreviated in the deposited vocabulary as "C". Similarly for model "AtomC347CA56", the "C" in the classes names is an abbreviation for "Cycle". The model's classes are illustrated with ligand FUL from PDB (entry 4Z4T), the classes that appear in this structure have their column names colored accordingly, and the background class always receives the black color.

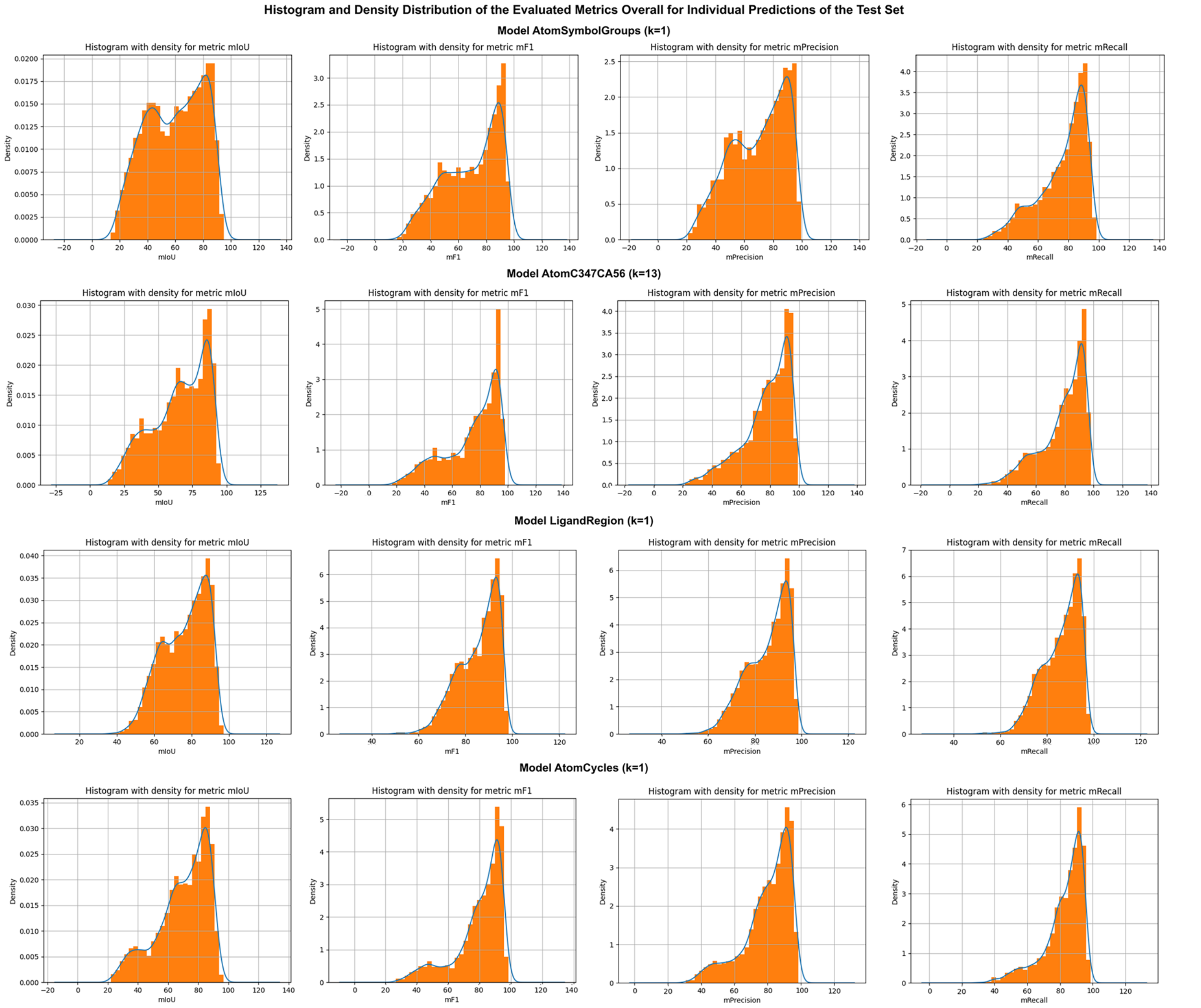


Fig. 9 Evaluation of the validated DL Models in the test set used to plot the histogram and density distribution of the 3,035 individual overall predictions for metrics IoU, F1-score, Precision and Recall. The test subsets k = 1 and k = 13 were used and are indicated after the model's name.

clouds and point size scaling. The user can register the best poses by taking pictures of the display (more information on this display capabilities in the Open3D http://www.open3d.org/docs/latest/tutorial/Basic/visualization.html).

In the vocabulary mapping tables, it is expected that the "background" class is assigned with a source index equal to the size of the respective vocabulary (last index). This format allows an easy indexing of the mapping for fast replacement: the new target index can be ordered by the source index and used to replace the old source index values by the new target index values in a given set of labels. To apply a mapping in a ligand label file the user may first order the mapping table by the "source" column in increasing order; then, get the values of the "target" column as a NumPy[60] v1.17 array. Next, the user should read the set of labels from the ligand label file, convert the labels to a Numpy array and replace the values equal to −1 with the size of the vocabulary ("background" label index adjustment). Finally, the Numpy array with the target index column ordered by source may be indexed with the labels array to return the new set of mapped labels. The classes of the mapped vocabulary are stored in the "mapping" column and may be ordered ascending by the "target" column. This is how the mapping is implemented for this work and for the training pipeline.

The visualizations of the predictions can be assessed with the Python script named "visualize_predictions", which is in the "src" folder of the np3_DL_repository (see Code Availability). This repository also contains one script named "plot_learning_curves.py" with auxiliary functions to plot the learning curves of a training process (data retrieved with TensorBoard[61] v2.2).

Code availability

The code used to create *LigPCDS* and to train the DL models is freely available in the "np3_ligand" repository of Github in v1.0.1: https://github.com/danielatrivella/np3_ligand. This repository also contains installation instructions with the full list of used packages and their versions. These two tasks are separated in two

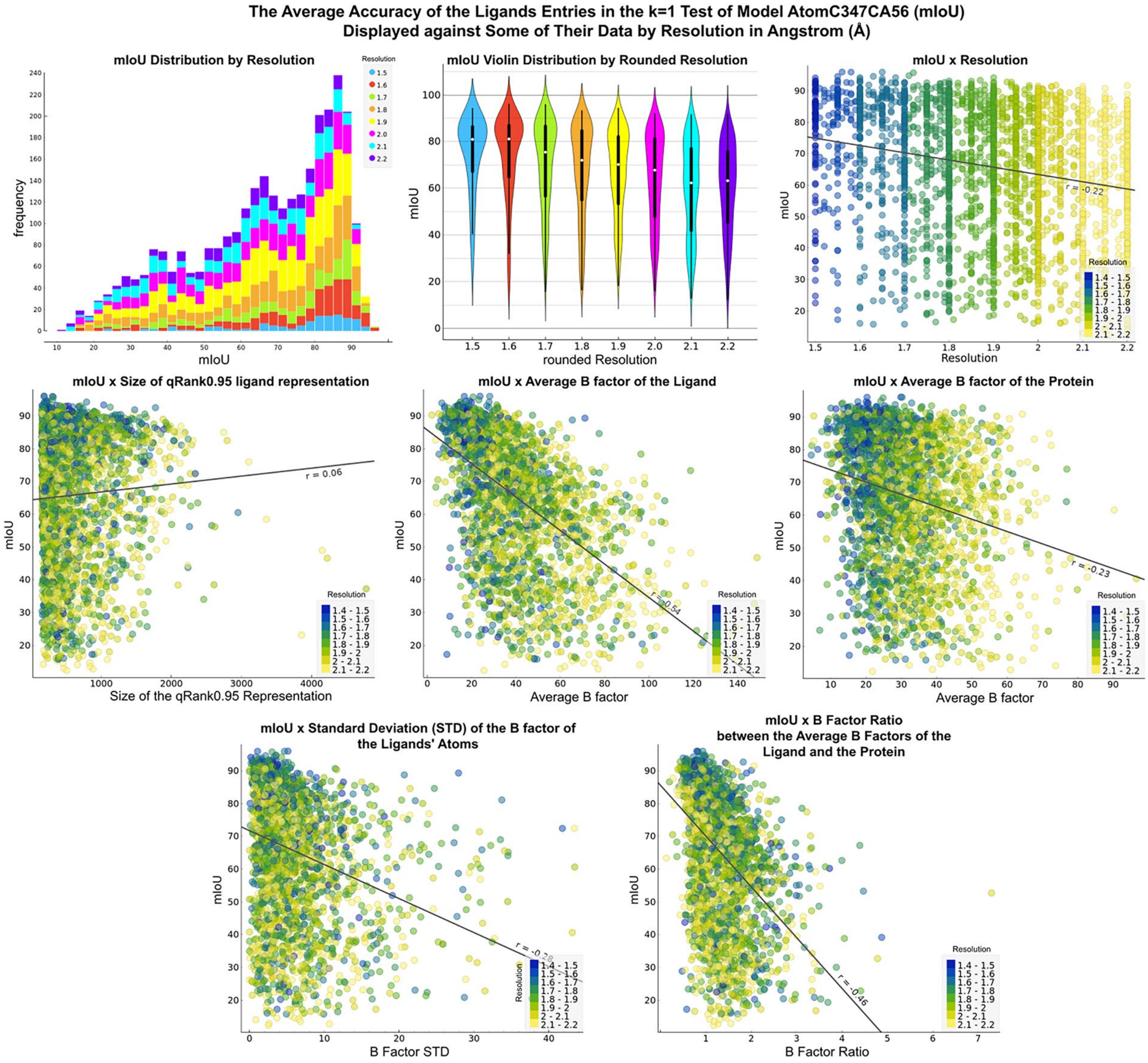


Fig. 10 Model AtomC347CA56 accuracy in mIoU against characteristics of the k = 1 test ligands. Resolution, size of the qRank0.95 point cloud, and B-factors values for the ligand and protein were investigated. The plots were constructed with Orange[62]. All plots are colored by the resolution of the entries, grouped in intervals of 0.1 Å. A simple linear regression was performed on all plots and the Pearson's correlation coefficient, variable r, is shown along with the regression-fitted line. The variable r squared (r^2) is the coefficient of determination.

directories named "np3_LigPCDS" and "np3_DL_segmentation", respectively. There is also a directory in this repository for the NP^3 Blob Label application (to be published). The "np3_ligand" repository also contains a manual of use for each task and installation instructions. With this open-source project, all steps of the workflow - presented in Fig. 1 - may be reproducible using the pipeline of available scripts (detailed in the repository documentation). The scripts of the first step of this workflow need special attention. This step depends on the APIs of RCSB PDB to download the entries data and it was designed to work with the RCSB PDB version available in December 2019. Any new updates to access RCSB PDB must be updated in the code for new downloads to work.

Data availability

The *LigPCDS* dataset v1.0.1 is available at https://doi.org/10.5281/zenodo.15174758[43]. Code, additional files and scrips are available at https://github.com/danielatrivella/np3_ligand, as mentioned above.

Received: 22 January 2025; Accepted: 15 September 2025;
Published online: 31 October 2025

References

1. Papageorgiou, A. C., Poudel, N. & Mattsson, J. Protein Structure Analysis and Validation with X-Ray Crystallography. in 377–404. https://doi.org/10.1007/978-1-0716-0775-6_25 (2021).
2. Berman, H., Henrick, K. & Nakamura, H. Announcing the worldwide Protein Data Bank. *Nat. Struct. Mol. Biol.* **10**, 980–980, https://doi.org/10.1038/nsb1203-980 (2003).
3. Berman, H. M. The Protein Data Bank. *Nucleic Acids Res.* **28**, 235–242, https://doi.org/10.1093/nar/28.1.235 (2000).
4. Wlodawer, A., Minor, W., Dauter, Z. & Jaskolski, M. Protein crystallography for non-crystallographers, or how to get the best (but not more) from published macromolecular structures. *FEBS J.* **275**, 1–21, https://doi.org/10.1111/j.1742-4658.2007.06178.x (2008).
5. Kleywegt, G. J. & Alwyn Jones, T. Model building and refinement practice. in *Methods Enzymol.* 208–230, https://doi.org/10.1016/S0076-6879(97)77013-7 (1997).
6. Aguda, A. H. *et al.* Affinity Crystallography: A New Approach to Extracting High-Affinity Enzyme Inhibitors from Natural Extracts. *J. Nat. Prod.* **79**, 1962–1970, https://doi.org/10.1021/acs.jnatprod.6b00215 (2016).
7. Mooij, W. T. M. *et al.* Automated Protein–Ligand Crystallography for Structure-Based Drug Design. *ChemMedChem* **1**, 827–838, https://doi.org/10.1002/cmdc.200600074 (2006).
8. Shumilin, I. A. *et al.* Identification of Unknown Protein Function Using Metabolite Cocktail Screening. *Structure* **20**, 1715–1725, https://doi.org/10.1016/j.str.2012.07.016 (2012).
9. Meneghello, R. *et al.* High-throughput protein crystallography to empower natural product-based drug discovery. *Acta Crystallogr. Sect. F Struct. Biol. Commun.* **81**, https://doi.org/10.1107/S2053230X25001542 (2025).
10. Carolan, C. G. & Lamzin, V. S. Automated identification of crystallographic ligands using sparse-density representations. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **70**, 1844–1853, https://doi.org/10.1107/S1399004714008578 (2014).
11. Beshnova, D. A., Pereira, J. & Lamzin, V. S. Estimation of the protein–ligand interaction energy for model building and validation. *Acta Crystallogr. Sect. D Struct. Biol.* **73**, 195–202, https://doi.org/10.1107/S2059798317003400 (2017).
12. Kowiel, M. *et al.* Automatic recognition of ligands in electron density by machine learning. *Bioinformatics* **35**, 452–461, https://doi.org/10.1093/bioinformatics/bty626 (2019).
13. Terwilliger, T. C., Adams, P. D., Moriarty, N. W. & Cohn, J. D. Ligand identification using electron-density map correlations. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **63**, 101–107, https://doi.org/10.1107/S0907444906046233 (2007).
14. Zwart, P. H., Langer, G. G. & Lamzin, V. S. Modelling bound ligands in protein crystal structures. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **60**, 2230–2239, https://doi.org/10.1107/S0907444904012995 (2004).
15. Karolczak, J. *et al.* Ligand Identification in CryoEM and X-ray Maps Using Deep Learning. *bioRxiv* https://doi.org/10.1101/2024.08.27.610022 (2024).
16. Bazzano, C. F. *et al.* NP 3 MS Workflow: An Open-Source Software System to Empower Natural Product-Based Drug Discovery Using Untargeted Metabolomics. *Anal. Chem.* **96**, 7460–7469, https://doi.org/10.1021/acs.analchem.3c05829 (2024).
17. Vollmar, M. & Evans, G. Machine learning applications in macromolecular X-ray crystallography. *Crystallogr. Rev.* **27**, 54–101, https://doi.org/10.1080/0889311X.2021.1982914 (2021).
18. Pozharski, E., Weichenberger, C. X. & Rupp, B. Techniques, tools and best practices for ligand electron-density analysis and results from their application to deposited crystal structures. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **69**, 150–167, https://doi.org/10.1107/S0907444912044423 (2013).
19. Dauter, Z., Wlodawer, A., Minor, W., Jaskolski, M. & Rupp, B. Avoidable errors in deposited macromolecular structures: an impediment to efficient data mining. *IUCrJ* **1**, 179–193, https://doi.org/10.1107/S2052252514005442 (2014).
20. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444, https://doi.org/10.1038/nature14539 (2015).
21. Choy, C., Gwak, J. & Savarese, S. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 3070–3079, https://doi.org/10.1109/CVPR.2019.00319 (IEEE, 2019).
22. Guo, Y. *et al.* Deep Learning for 3D Point Clouds: A Survey. *arXiv* https://doi.org/10.48550/arXiv.1912.12033 (2019).
23. Singh, R. D., Mittal, A. & Bhatia, R. K. 3D convolutional neural network for object recognition: a review. *Multimed. Tools Appl.* **78**, 15951–15995, https://doi.org/10.1007/s11042-018-6912-6 (2019).
24. Ahmed, E. *et al.* A survey on Deep Learning Advances on Different 3D Data Representations. *arXiv* https://doi.org/10.48550/arXiv.1808.01462 (2018).
25. Winn, M. D. *et al.* Overview of the CCP4 suite and current developments. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **67**, 235–242, https://doi.org/10.1107/S0907444910045749 (2011).
26. Wojdyr, M. GEMMI: A library for structural biology. *J. Open Source Softw.* **7**, 4200, https://doi.org/10.21105/joss.04200 (2022).
27. Murphy, A. H. The Finley Affair: A Signal Event in the History of Forecast Verification. *Weather Forecast*. **11**, 3–20, 10.1175/1520-0434(1996)011<0003:TFAASE>2.0.CO;2 (1996).
28. Dice, L. R. Measures of the Amount of Ecologic Association Between Species. *Ecology* **26**, 297–302, https://doi.org/10.2307/1932409 (1945).
29. Zhou, Q.-Y., Park, J. & Koltun, V. Open3D: A Modern Library for 3D Data Processing. *arXiv* https://doi.org/10.48550/arXiv.1801.09847 (2018).
30. Guo, Y. *et al.* Deep Learning for 3D Point Clouds: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**, 4338–4364, https://doi.org/10.1109/TPAMI.2020.3005434 (2021).
31. Batsanov, S. S. Van der Waals Radii of Elements. *Inorg. Mater.* **37**, 871–885, https://doi.org/10.1023/A:1011625728803 (2001).
32. Afonine, P. V. *et al.* Real-space refinement in PHENIX for cryo-EM and crystallography. *Acta Crystallogr. Sect. D Struct. Biol.* **74**, 531–544, https://doi.org/10.1107/S2059798318006551 (2018).
33. Urzhumtsev, A., Afonine, P. V., Lunin, V. Y., Terwilliger, T. C. & Adams, P. D. Metrics for comparison of crystallographic maps. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **70**, 2593–2606, https://doi.org/10.1107/S1399004714016289 (2014).
34. Hawkes, P. W. Digital image processing. *Nature* **285**, 174–175, https://doi.org/10.1038/285174b0 (1980).
35. Lamb, A. L., Kappock, T. J. & Silvaggi, N. R. You are lost without a map: Navigating the sea of protein structures. *Biochim. Biophys. Acta - Proteins Proteomics* **1854**, 258–268, https://doi.org/10.1016/j.bbapap.2014.12.021 (2015).
36. Muja, M. & Lowe, D. G. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. in *International Conference on Computer Vision Theory and Applications ISBN 978-989-8111-69-2* 331–340 (INSTICC Press 2009, 2009).
37. Fredriksson, T., Mattos, D. I., Bosch, J. & Olsson, H. H. Data Labeling: An Empirical Investigation into Industrial Challenges and Mitigation Strategies. in Morisio, M., Torchiano, M., Jedlitschka, A. (eds) *Product-Focused Software Process Improvement. PROFES 2020. Lecture Notes in Computer Science(), vol 12562. Springer, Cham.* 202–216, https://doi.org/10.1007/978-3-030-64148-1_13 (2020).
38. Jin, W., Barzilay, R. & Jaakkola, T. Junction Tree Variational Autoencoder for Molecular Graph Generation. *arXiv*. https://doi.org/10.48550/arXiv.1802.04364 (2018).
39. Johnson, J. M. & Khoshgoftaar, T. M. Survey on deep learning with class imbalance. *J. Big Data* **6**, 27, https://doi.org/10.1186/s40537-019-0192-5 (2019).
40. Buda, M., Maki, A. & Mazurowski, M. A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* **106**, 249–259, https://doi.org/10.1016/j.neunet.2018.07.011 (2018).
41. Grant, J. A. & Pickup, B. T. A Gaussian Description of Molecular Shape. *J. Phys. Chem.* **99**, 3503–3510, https://doi.org/10.1021/j100011a016 (1995).
42. Jain, A. N. *et al.* XGen: Real-Space Fitting of Complex Ligand Conformational Ensembles to X-ray Electron Density Maps. *J. Med. Chem.* **63**, 10509–10528, https://doi.org/10.1021/acs.jmedchem.0c01373 (2020).

43. Bazzano, C. F., Alves, L. F. G., Telles, G. P. & Trivella, D. B. B. LigPCDS: Labeled Dataset of X-ray Protein Ligand 3D Images in Point Clouds and Validated Deep Learning Models (1.0.0) [Data set]. *Zenodo* https://doi.org/10.5281/zenodo.15174758 (2023).
44. Papenberg, M. & Klau, G. W. Using anticlustering to partition data sets into equivalent parts. *Psychol. Methods* **26**, 161–174, https://doi.org/10.1037/met0000301 (2021).
45. Little, M. A. *et al*. Using and understanding cross-validation strategies. Perspectives on Saeb *et al*. *Gigascience* **6**, https://doi.org/10.1093/gigascience/gix020 (2017).
46. Shoba Ranganathan, Kenta Nakai, C. S. *Encyclopedia of Bioinformatics and Computational Biology - 1st Edition*. (Elsivier, 2018).
47. Choy, C., Gwak, J. & Savarese, S. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. *arXiv* https://doi.org/10.48550/arXiv.1904.08755 (2019).
48. Yu, F. & Koltun, V. *Multi-Scale Context Aggregation by Dilated Convolutions*. https://doi.org/10.48550/arXiv.1511.07122 (2015).
49. Wang, P. *et al*. Understanding Convolution for Semantic Segmentation. in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* 1451–1460, https://doi.org/10.1109/WACV.2018.00163 (IEEE, 2018).
50. Dumoulin, V. & Visin, F. A guide to convolution arithmetic for deep learning. https://doi.org/10.48550/arXiv.1603.07285 (2016).
51. Dai, A. *et al*. *ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes*. https://doi.org/10.48550/arXiv.1702.04405 (2017).
52. Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T. & Ronneberger, O. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation (2016).
53. Falcon, W. *et al*. PyTorchLightning/pytorch-lightning. at https://doi.org/10.5281/zenodo.3828935 (2020).
54. Wang, Y. *et al*. *Symmetric Cross Entropy for Robust Learning with Noisy Labels*. https://doi.org/10.48550/arXiv.1908.06112 (2019).
55. Ruder, S. *An overview of gradient descent optimization algorithms*. https://doi.org/10.48550/arXiv.1609.04747 (2016).
56. Lin, T.-Y. *et al*. Microsoft COCO: Common Objects in Context. in Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. *(eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8693. Springer, Cham*. 740–755, https://doi.org/10.1007/978-3-319-10602-1_48 (2014).
57. Efron, B. & Tibshiran, R. J. *An Introduction to the Bootstrap*. (Chapman & Hall/CRC, 1993).
58. Jurdi, R. E & Gaël Varoquaux, O. Confidence intervals for performance estimates in 3D medical image segmentation. (2023).
59. Khan, S. H., Hayat, M., Bennamoun, M., Sohel, F. & Togneri, R. Cost Sensitive Learning of Deep Feature Representations from Imbalanced Data. https://doi.org/10.48550/arXiv.1508.03422 (2015).
60. Harris *et al*. Array programming with NumPy. *Nature* **585**, 357–362, https://doi.org/10.1038/s41586-020-2649-2 (2020).
61. Abadi, M. *et al*. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. https://doi.org/10.48550/arXiv.1603.04467 (2016).
62. Demsar, J. *et al*. Orange: data mining toolbox in python. *J. Mach. Learn. Res.* **14**, 2349–2353 (2013).

Acknowledgements

Authors acknowledge Dr Rafael de Felicio, Dr Helder Veras Ribeiro Filho, Dr João Victor da Silva Guerra, Dr José Geraldo de Carvalho Pereira, Dr Leandro Oliveira Bortot, Dr Paulo Sergio Lopes de Oliveira, Dr Eduardo Xavier Silva Miqueles, Dr. Thiago Spina, Dr. Marjorie Bruder and Dr. Joane Rustiguel from CNPEM for discussions on this project. Authors also acknowledge the Scientific Computing Department of LNBio-CNPEM, and the Tepui Facility of the Synchrotron Light Source SIRIUS-LNLS-CNPEM for the computational infrastructure support, respectively, on the main hardware used and on hardware B. This work was supported by the Serrapilheira Institute grant number Serra-1709-19681.

Author contributions

C.F.B.: Defined the computing strategies, designed, implemented and executed the scripts and tests. Wrote the manuscript and compiled the data for deposition. L.F.G.A.: Discussion based on statistics and executed some of the tests. G.P.T.: Conceptualization, discussions based on machine learning, co-coordination of the project. D.B.B.T.: Conceptualization, discussions based on chemistry and protein crystallography, funding and co-coordination of the project. All authors contributed to the text and agreed with the final version of the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41597-025-06002-8.

Correspondence and requests for materials should be addressed to G.P.T. or D.B.B.T.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025