# Data Standards & Curation Guide

Megathon 2026: *A Working Document for ML-Ready Biological Dataset Development*

---

# Part 0 — Document Scope

## 0.1 Why This Guide Exists

Dataset management often consumes the majority of machine learning development especially for new or technically difficult problems. The success of these models depends not only on the architecture itself, but on the organization, cleaning, and standardization of the underlying data. This guide, in part, exists as a guideline for how to curate these datasets to be as informative as possible.

Biological datasets are heterogenous and often inconsistently curated or formatted. As a result, they often lack clear structure or metadata. Integrating datasets that should be compatible with machine learning models requires repeated manual intervention. Similarly, when it comes to benchmarking methods, different models are trained on different datasets or data splits, leading to difficult side-by-side comparisons.

Within Rosetta Commons, there is no centralized, searchable, or standardized dataset index. Many valuable datasets exist within our community, but they are stored in formats or locations that make them difficult to find or reuse–pickeled objects, zip archives, nested directories, or partially documented files, to name a few.

## 0.2 Goals of the Megathon Data Track

This even focuses on converting these datasets into a practical infrastructure through a shared index with unified standards, and improved accessibility. We hope that these datasets are easier to find, interpret, and use across the community.

1. Upload ~12 curated datasets to the Rosetta Hugging Face repository.
2. Develop a reusable protocol for ML-ready biological data.
3. Identify friction points in real-time during model training.
4. Establish minimum curation standards.
5. Produce a finalized curation guides and standards document post-event.

# Part I — What Is a Dataset?

## 1.1 Defining Scope

- **What qualifies as a dataset?**
    - Datasets used for machine learning models can vary widely. They differ in terms of content, how they were collected, and how they are labeled. For example, a dataset can be:
        - Raw and unstructured data: data collected as-is with no cleaning or organization (e.g. raw data from an experiment, audio recordings, scraped web pages, filled in form)
        - Curated and annotated: data that has been cleaned, standardized, and enriched with detailed labels, metadata, and/or post-processing.

- **What does not qualify?**
    - Not everything that contains data qualifies as a dataset for machine learning purposes. Often, these are useful to include in metadata if they are informative to specialists, but would not be used for model training. The following generally do not qualify:
        - Single data point: a lone image, document, or record with no accompanying samples to learn from.
        - Unverified or corrupted data: data that is sparse, inaccessible, or too corrupted to be readily used for training or evaluation. This is distinct from data that is mostly intact but contains some noise, missing values, or systematic corruption. Often, these still qualify as datasets with appropriate preprocessing or filtering applied.
        - Derived outputs: generally, model predictions or other outputs that have not been validated or curated. However, there are some widely used model predictions that have been systematically produced, independently validated and/or broadly accepted by the research community as training data (ex. AlphaFold Database). This can be useful to curate, but should be used with caution as it comes with its own set of assumptions.
        - Arbitrary collections of data: a miscellaneous grouping of data with no consistent structure, format, or purpose tied to a specific modeling task.
        - Personal or sensitive data without proper consent: data that cannot be lawfully used for model training due to privacy, legal, or ethical restrictions.

- **Minimal unit of submission**
    - A dataset should correspond to a coherent, self-contained collection that can be meaningfully found, referenced, and reused. The minimal unit depends on the data type. Some examples include:

- Tables or structured files (e.g. csv, paraquet, json): submitted as a single table or clearly defined set of related tables with shared annotations
- Structured archives (e.g. PDB/mmCIF collections): submitted as a curated set with consistent metadata, versioning, and/or post-processing
- Curated splits (train/val/test): must be clearly documented and reproducible, of particular consideration is how the data is split.

# 1.2 Dataset Categories

These categories define some of the major types of datasets relevant to modeling within the Rosetta community and related computational biology workfloats. Each category carries its own set of considerations–what metadata is essential, typical file formats, level of curation, and biological context. Defining these data categories may help in considering what information is important for the curated dataset and to ensure understanding, consistency, and usability.

## A. Protein Structures

Includes datasets where the primary data unit is a 3D protein structure or coordinate set. Important distinctions include:

- Specialized structural classes: these are structures that have distinct geometry, annotation needs, and labeling conventions. Some examples include *antibodies*, *motifs* (catalytic, binding, etc), or *peptides*.

- Predicted vs. modeled vs experimental: whether the structure originates from experimental techniques (X-ray, Cryo-EM, NMR) or computational models (derived from prediction or modeling methods). Often the residue annotations can differ.

Some common sources for protein structure datasets include but are not limited to the Protein Data Bank (PDB), Cambridge Structural Database (CSD), AlphaFold predicted structures (ex. AFDB), or molecular dynamics datasets (ATLAS or mdCATH).

Handling of deposited protein structure should be done with care. Consider recording the structure origin, resolution/quality metrics, and file format (PDB/mmCIF). Consider pre-processing to handle consistent residue labeling schemes, post-processing for unresolved residues or missing densities were handled, and whether any post-processing was applied (e.g. relaxation, fixing rotamers, trimming termini). Many structures may vary in terms of composition, and it is often valuable to record chain identifiers, biologically relevant assemblies vs. asymmetric units, co-factors, and ligands.

## B. Structural Biology Data

Includes datasets where the primary data unit is an experimental observable related to the protein structure or derived from a structural determination method, rather than a final

coordinate model. These datasets are often used to build, validate, or restrain structural models; they may be linked to one or more protein structures; they may reveal biophysical characteristics of the protein. Examples of these data collection methods include:

- X-ray crystallography: these data often support the generation and interpretation of high-resolution atomic models. Datasets from X-ray crystallography can include different data types (e.g. diffraction images, reflection files, electron density maps) and quality metrics (resolution cutoff, B-factors, R/Rfree, clashscores)

- Small angle X-ray scattering (SAXS): these provide low-resolution ensemble-averaged information about macromolecular shape and size in solution. Datasets from SAXS can include raw or processed profiles and derived parameters (e.g. Rg).

- Nuclear magnetic resonance (NMR); NMR experiments provide high-dimensional restraints and local structural information; they are often used for structure determination, ensemble modeling, and dynamics analysis. Datasets from NMR can include chemical shifts, NOE-derived distance restraints, and relaxation data.

- Cryo-electron microscopy (Cryo-EM): Cryo-EM provides 3D density maps from vitrified samples, ranging in resolution. Different dataset types can include maps, particle stacks, angular distribution plots, map quality metrics, and observed heterogeneity.

Experimental data from structural biology techniques can depend greatly on the macromolecules' preparation. For all the listed methods, other dataforms to consider include the experimental sample conditions (temperature, pH, buffer composition), identifiers connecting these models to databases, file-formats, quality metrics, and even instrument origin. For many public datasets, this information is not or may not be readily available, but is often recorded in the raw datafiles.

## C. Sequences

Includes datasets where the primary data unit is a linear sequence representation of a macromolecule, often nucleotide (DNA/RNA) or amino-acid (protein) sequence. These are often accompanied by metadata, annotations, alignments, or structures.

- Genomes and proteomes: complete or partial sets of genes/proteins for an organism, species, or metagenomic sample.
    - *Sources*: RefSeq, Uniprot, metagenomic assemblies, custom annotations, etc.
    - *Completeness and quality*: presence of partial ORFs, sequencing errors, unassembled contigs, frameshifts
    - *Annotation layers*: isoforms, regulatory elements, signaling peptides, transmembrane regions, domain architecture

- Protein families: curated groupings of related sequences that share homology, structural motifs, function, or domain composition. Reporting of sequences varies widely between sources and should be handled with care.
    - *Source*: Uniprot, SwissProt, UniRef50/90, Pfam, CATH, or SCOP. Often these have some family definition such as clustering thresholds, structural families, or functional curation.
    - *Functional annotation*: active-site residues, catalytic motifs, co-factor dependences, subscellular localization

- Multiple sequence alignments (MSAs): aligned sequences, conventionally used for analyzing conservation, coevolution, mutational effects, or used for training generative models or structure prediction algorithms. Alignments differ widely based on data source, method, and post-processing.
    - *Alignment method*: HHbits, HMMER, MMSeqs, MAFFT, structural alignments
    - *Post-processing:* can vary widely and are often repeated per project. These include handling of gaps, data coverage, depth, residue numbering, offset handling, noncanonical residues,

## D. Functional Data

Includes datasets where the primary information is a measurable property associated with a sequence or structure rather than the sequence or structure itself. These are often biochemical, biophysical, or phenotypic properties reflecting functions like activity, binding, or stability under specific experimental conditions, and therefore are highly sensitive to experimental context.

- Enzyme kinetics: Quantitative measurements describing catalytic activity and efficiency, often used in enzyme engineering/design or mutational analysis to probe function.
    - Typical readouts include:
        - kcat, Km, kcat/Km (catalytic efficiency)
        - kcat/kuncat (rate enhancement)
        - substrate specificity
        - pH and ionic strength profiles, temperature dependence, ion dependence
        - Single turnover vs. steady-state kinetics
    - Consider the different methods of data fitting (Michaelis-menten model, Hil model, inhibition models), substrate identity, and detection chemistry (absorbance, fluorogenic, colorimetric)
- Binding data: Measurements describing molecular interactions between proteins, ligands, nucleic acids, peptides, and/or co-factors.
    - Typical readouts include:
        - Kd, IC50, kon, koff
        - Binding stoichiometry and thermodynamic signatures
    - Measurements regarding binding are often collected via SPR, BLI, ITC, competition assays, or high-throughput screens.

    - Consider the contextual details of the experiment (e.g. protein concentration, immobilization artefacts, ligand solubility, non-specific binding) that may affect data quality.
- Stability data: these measurements often describe the thermodynamic properties of a protein and reflect their folding/unfolding, aggregation, or expression fitness.
    - Typical readouts include:
        - Tm (melting temperature)
        - dG of folding/unfolding (assumes either two-state or multi-state model)
        - Expression level or solubility
    - These datasets are sensitive to buffer composition, denaturants, mutations, and assay linearity. Classically, these parameters are derived from CD and DSF.
- Activity screens and high-throughput measurements: developed in an effort of generating thousands to millions of datapoints
    - Methods for high-throughput measurements include deep mutational scanning (DMS) scores, FACS or growth enrichment rations, fluorescence screens, reporter-based screens, or phenotypic readouts.
    - These datasets typically require normalization (e.g. log enrichment), noise modeling, and may be limited to lower-resolution values. Consider selection bias in these datasets.

Functional data must be unambiguously linked to an underlying biomolecule. Important metadata here include the protein identifier, variant, sequence, and/or construct.

Most functions are often context-dependent, and so experimental conditions are typically vital. Metadata that may be useful here include pH, buffer, ionic strength, temperature, cofactors, substrate/ligand identity, and additives.

A clear definition of the assay will enable reproducibility and data/model interpretability. Consider the assay modality (e.g. fluorescence, spectrophotometric), detection chemistry (e.g. fluorophore, wavelength(s), quenchers), datatype (e.g. endpoint measurement, initial rate), linear vs. calibration curves, and uncertainty/model fitting/experimental error.

For most functional readouts, make sure to have consistent units, normalization, and/or transformations. Often it is useful to have both the raw data, and the transformed data for future reprocessing. It may also be useful to note any inconsistencies between datapoints.

## E. Macromolecular Interactions (e.g. PPI, PLI)

Includes datasets that describe physical, structural, reported, or functional interactions between macromolecules. These interactions may involve two or more proteins (PPIs), proteins with nucleic acids, or proteins with small molecules/ligands (PLIs). Such datasets are typically used in developing methods for docking, binder design, affinity prediction, hot spot identification, and molecular recognition.

- Interfaces from structure: residue- or atom-level interaction networks derived from experimentally solved or computationally modeled datasets.
    - Typical features derived from here include biophysical interactions (e.g. hydrogen bonds, salt-bridges, pi-stacking, cation-pi interactions), metrics such as buried surface area, interface hydrophobicity, or shape complementarity
    - These datasets are highly varied, and may or may not capture aspects involved in interactions such as conformational changes, dynamic interactions, or multi-domain assemblies.
    - Representations of these interactions include structures, graphs, or residue annotations.

- Docking benchmarks: often a curated set of a specific type of molecular interactions.
    - Examples include: antibody-antigen interactions, receptor-ligand structures, and protein-small molecule benchmarks.
    - Less curated by still important are negative poses and annotations of interaction type (rigid body, flexible, induced-fit, large rearrangement)

- Mutational scanning datasets: high-resolution interaction landscapes may capture energetic contributions of residues or ligands.
    - Datasets include alanine scanning, deep mutational scanning, hotspot mapping, and epitope mapping datasets.

Data of these interactions are similarly affected by their biological and experimental contexts. Consider reporting the experimental curation method and its sensitivity.

Different types of interaction exist and the curation of their computational representation depends on a scientist's interpretation of the biological problem at hand. Many macromolecules interact with one another (ranging from proteins and metals to proteins and metabolites). The best representations for these is an outstanding question of research. As such, including standard representations (such as sequence string, PDB codes, or SMILES string) is strongly encouraged.

## F. Membrane Proteins

Includes datasets focused on membrane-embedded or membrane-associated proteins. Often these require additional context because membrane environments strongly influence their structure, stability, and function. As such, consider these more detailed metadata:

- Transmembrane annotation: information about membrane spanning segments, orientation of domains, domain boundaries.

- Solubility and expression: detergent solubility and stability profiles, functional assays under different solubilization conditions, expression level, folding efficiency

- Environment composition: the structural and functional properties of a membrane protein heavily depends on its environment; consider lipid composition, membrane mimic used, and purification conditions

## G. Small molecules

Includes ligands, cofactors, substrates, fragments, metabolites, and other chemically defined small molecules. These datasets are essential for modeling tasks such as docking and scoring, and can be leveraged for protein design or understanding structure-activity relationships. Consider how these small molecules are represented:

- Chemical representations: provide standardized, unambiguous identifiers and/or structural formats.
    - Examples include SMILES, SDF, MOL2, PDBQT
    - Databases include PubChem CID, ChEBI ID, and DrugBank ID
    - Note the tautomer, stereochemistry, and charge state

- Protonation and tautomer states: chemical state can dramatically alter docking, scoring, and conformer of a small molecule
    - Consider including information on pH and protonation and whether tautomers were enumerated, filtered, or fixed.

---

# Part II — The Curation Guide

The following is a stepwise framework for curating datasets. This guide is intended to support consistent and transparent curation across diverse data types.

---

## Step 1: Does the Dataset Meet Minimum Criteria?

Before starting on dataset curation, consider the minimum criteria for generating a dataset. These checks are meant as a guide to ensure maximum interpretability, reproducibility, and suitability for downstream use.

**Minimum criteria:**

1. **Defined biological question or purpose**: the dataset should have a clear objective, task, or biological context.
2. **Consistent labeling and structure**: within a dataset, there should be some coherent scheme such as residue numbering, sequence formatting, measurement units, or annotation conventions to name a few.
3. **Known origin**: the dataset origin must be documented (source database, experiment, publication doi, lab, website).
4. **Sufficient metadata or outlining to reproduce preprocessing**: preprocessing steps must be either explicitly documented or referenced. These processing steps include filtering, alignments, and normalization.

---

## Step 2: What Type of Data Is It?

Determine its primary data type. Each category has specific expectations to ensure the dataset is maximally usable for model training and consistent with community standards. Refer to 1.2 Dataset categories for more in depth information regarding specific data types. For some datasets, you may be labeling and annotating data across categories. Not all data types will be pertinent for the specific biological question.

- **Protein structure data**: focus is on coordinates, geometry, and structural annotations
    - *Standardized formatting*: consistent file format (PDB/mmCIF), standardized chain IDs, residue numbering, chain handling, insertion codes, purification tags
    - *Atom representation*: completeness of heavy atoms, hydrogens, alternate conformers, ligands, metals
    - *Coordinate integrity*: missing density handling, occupancy values, clashes, resolution, quality metrics
    - *Experimental vs. predicted*: label the source of the data
    - *Post-processing*: relaxation, rotamer fixing, termini trimming, treatment of unresolved regions

- **Structural biology data**: focus on experimental observables from structural biology techniques, rather than final coordinate models
    - *Experimental conditions*: temperature, pH, buffer composition, instrument identifiers, etc.
    - *Format consistency*: raw vs. processed data
    - *Origin of data*: accession IDs (PDB, EMDB, BMRB) and mapping between data and models, if applicable
    - *Pre- or post-processing*: normalization, filtering, etc.

- **Sequence data**: focus is on linear representations (sequences) of macromolecules
    - *Sequence quality*: completeness, frameshifts, removal of fragments, insertions, experimental artifacts
    - *Source metadata*: taxonomy, database accession, versioning
    - *For MSAs:* alignment method, gap handling, residue numbering, coverage, depth

- **Functional data:** focus is on measurements of biochemical, biophysical, or phenotypic properties
    - *Units*: consistent units for measured or fit parameters
    - *Assay conditions*: pH, buffer, ionic strength, temperature, cofactors, substrate/ligand identity, etc.
    - *Measurement model*: Michaelis-Menten, Hill fit, binding model, etc.
    - *Data quality*: replicates, outlier fitting, noise modeling
    - *Readout type*: luminescence, fluorescence, absorbance, calorimetric, endpoint vs. initial rate. Consider also recording the wavelengths
    - *Linking to biomolecules*: sequence, structure, and/or variant identifiers must be unambiguous.

- **Macromolecular interactions:** focus is on two or more macromolecules interacting
    - *Interface definition*: distance cutoff, atom vs. residue counts, buried surface area, interaction types, etc.
    - *Representation*: structural models, graph, atom/residue annotations, etc.
    - *Mutational landscape data*: alanine scanning, DMS, hotspot mapping
    - *Experimental modality metadata*: SPR, BLI, ITC, FACs, competition assay
    - *Assay conditions*: pH, buffer, ionic strength, temperature, cofactors, substrate/ligand identity etc.
    - *Context*: stoichiometry, conformational heterogeneity.

- **Membrane proteins:** focus is on proteins that are membrane-bound or -associated
    - *Transmembrane annotation*: helix boundaries, orientations, etc.
    - *Environment composition*: lipid type, detergent, nanodiscs, micelles

- **Small molecules*:*** focus is on ligands, substrates, co-factors, metabolites, and other small molecules

- *Chemical representation*: SMILES, SDF, MOL2, PDBQT, stereochemistry, canonicalization
- *Protonation and tautomer states*: pH assumptions, charge assignment
- *Conformer generation*: method, number of conformers, minimization procedure
- *Physiochemical properties*: computed descriptors or experimental measurements
- *Biochemical context*: binding affinity, docking scores, or experimental conditions

---

# Step 3: Raw → Processed Workflow

Once the dataset has been identified, the next step is to document and proceed with curating the data. Depending on the purpose, downstream task, or curation standards required, this process may range from minimal cleaning to extensive transformation. It is highly encouraged to retain the raw data in your curation to ensure reproducibility and allow for easy revisiting of the curation.

## 3.1 Raw Data

Raw data refers to the original, unmodified files obtained from experiments, public databases, or computational pipelines. This is essential for preserving reproducibility.

- **Original format**: specify the file type and structure exactly as it was obtained.
- **Storage location**: indicate where the raw data is stored (local repository, shared server, archival storage, and version identifiers or timestamps where applicable.
- **Frozen snapshot**: for databases that update continuously (e.g. PDB), record the exact version of the data of retrieval
- **Read-only archive**: any reprocessing should produce new files, rather than overwrite the original files. Preserve the raw data as a read-only archive

## 3.2 Cleaning and Standardization

This step aims to transform the raw data into a clean, structured, and consistent format while still preserving biological meaning.

- **Cleaning**: removal or identification of formatting errors, malformed entries, duplicates, or unreadable files.
- **Standardization:** ensure consistent naming conventions, identifier formats, residue numbering, masking
- **Unit normalization**: convert all measurements to standardized units
- **Removing corrupt or invalid entries**: filter or flag samples with missing fields, unphysical values, duplicates, or measurement failures.
- **Masking**: alternatively, generate explicit masks for corrupted or invalid entries that can be tracked, excluded, or handled differently during downstream modeling without permanently discarding them.

### 3.3 ML-Oriented Transformation

This step adapts the cleaned dataset into representation suitable for machine learning workflows. Here, the transformations impose modeling assumptions that should be explicitly documented and reproducible. It is best practice and most helpful to keep a script that can reproduce this transformation for further iterations and improvement on curation.

Some examples of transformations include:

- **Feature extraction:** transforming raw scientific data into numerical, bounded, or categorical features. It is best practice to ensure that this processing has biophysical grounding and can be easily automated/reproduced. Try to avoid manual curation, but if you do, make sure to use consistent reasoning applied across the dataset as a whole.
- **Representation choice:** selecting how the data is represented for modeling (e.g. graph, voxel, sequence, point clouds, tokenization).
- **Encoding decisions:** one-hot encoding, learned embeddings, physiochemical parameterization, topological encodings.
- **Handling missing data:** masking strategies, data removal, explicit missing-value tokens
- **Label transformation:** log-transforming affinities, binning continuous variables, normalizing scores, etc.

At this stage, the transformation is highly dependent on the intended algorithm and its required input. It is often most useful to save the cleaned, standardized dataset in addition to the ML-oriented transformation to facilitate reversal or testing of other types of data representations.

---

## Step 4: Data Structure Decision

Once data processing is complete, choose the appropriate data structure(s) and file formats. The goal here is to ensure efficient storage, fast loading, and clear organization. Different datasets might benefit from different storage formats or a combination of storage formats.

In general, storage complexity should reflect the dataset scale and its intended use.

### 4.1 Data Structure Formats

**Tabular Formats (tables / dataframes):** use a table when each sample corresponds to a row with a fixed set of fields; this is ideal when the datasets fits a relational scheme with clearly defined columns, including functional data, annotations, metadata, and descriptors.

**Structured arrays or matrices**: use a matrix or dense array when samples share a uniform shape or size. This includes MSAs converted to numerical or one-hot encodings, contact maps, distance matrices, or embeddings. This is optimized for fast vectorization.

**Object storage**: use object storage for heterogenous, variable-sized, or complex data objects. This is optimal for avoiding forcing data into rigid shapes or sizes. Typically, each sample is a file saved as an object. This includes structural formats (PDB/mmCIF or SDF/MOL2), raw data outputs (maps, raw data), or nested representations (for instance, large json objects).

### 4.2 Storage Location

Document where the dataset is stored and ensure it is accessible, persistent, and versioned.

- *Local or on-disk storage* is ideal for simple and appropriate for small datasets
- *Cluster or HPC storage* is good for large archives or raw datasets
- *Cloud platforms (e.g. HuggingFace, S3)* is suitable for public datasets, processed files
- *External pointers* require reproducibility safeguarding but are good for very large, raw datasets and include version information, accession IDs, and download scripts.

---

## Step 5: Dataset Validation and Quality Assessment

After processing, the dataset should be assessed for both data integrity and biological plausibility. This is helpful to ensure scientific accuracy, internal consistency, and suitability for downstream training.

### 5.1 Integrity Checks

An overlooked part of data curation is ensuring that the dataset is well-formatted and internally coherent. A short checklist might include:

- Entries follow the expected fields, data types, formats
- Measurements and identifiers are reliably and consistently linked to each other
- Missing values match documented masking and/or removal strategy
- Detection of duplicate entries, redundant records, or inconsistent labels.
- Ensure files can all be parsed correctly and are properly formatted

### 5.2 Biological Plausibility Checks

Evaluate whether the dataset reflects reasonable biological constraints. In particular, if you make data processing decisions, ensure they are either (a) common best practice standards and/or (b) statistically rigorous. A short checklist might include:

- Verify canonical residues, expected length, annotations, reasonable masking of sequences.
- Check for chain continuity, occupancy issues, steric clashes
- Confirm functional values fall within expected physical ranges
- Ensure any cutoffs obey physical thresholds and/or match metadata

### 5.3 Statistical Quality Checks

Assess your data distribution to detect anomalies, outliers, and bias introduced by or resulting from the data curation. Visual assessment can be quite powerful here.

- *Assessment of data distribution:* inspect histograms, feature ranges, outliers, long tails, artefacts from normalizations
- *Redundancy and diversity*: quantify redundancy or diversity by assessing data coverage and data coverage. This may change based on the splitting strategy. See Step 6.
- *Replicate consistency*: compute statistics of sample (e.g. variance, standard deviation, count), flag inconsistencies

---

# Step 6: Data Splitting Strategy

Splitting a curated dataset into training/validation/test sets should enforce meaningful separation between samples and avoid information leakage. In biological dataset, similarity comes in several forms–sequence, structure, functional, and chemistry—so splits should be based on measures that reflect the underlying biology and the model's intended generalization task.

If a split already exists, preserve and clearly label any published or benchmark splits. Make sure to document the source and any known limitation (e.g. potential leakage) for the intended use.

Regardless of the data modality it is recommended that you:

1. Cluster first, then split
2. Assign entire clusters to the same train/val/test split
3. Choose thresholds consistent with the intended generalization
4. Write a reproducible script that can regenerate the data splitting.

Below are some guidelines for how to perform data clustering:

## 6.1. Sequence-based clustering

Use **global sequence identity** to prevent homologous or near identical sequences from appearing in both training and test sets.

Best practices here include clustering sequences at a chosen identity threshold (with MMSeqs or CD-HIT), assigning clusters to the same split (train/val/test), and documenting parameters and clustering statistics.

Recommended thresholds depend on the desired generalization.

- ≥ 50% identity reflects moderate separation and may test for local generalization
- ≤ 30% identity reflects strong separation and tests family-level or fold-level generalization.

## 6.2. Structure-based clustering

Use **structural similarity metrics** for datasets defined by 3D coordinates. This is often in addition to sequence identity splits.

There are two key metrics here: RMSD and TM-score.

- **RMSD**: < 2.0 Å is effectively identical backbone geometry while 2-4 Å is similar topology
- **TM-Score**: ≥ 0.5 is essentially the same fold while < 0.5 are different folds, topologies

## 6.3. Interaction-based clustering (PPIs, PLIs, Interfaces)

For interaction datasets, separation must account for both components of the interaction. Interaction prediction models can memorize interface geometries or ligand chemistries

For protein-protein interactions, split by sequence identity of *both* partners. For protein-ligand interactions, enforce both sequence and structural separation of proteins and chemical separation of ligands (by molecular similarity).

Best practice is to ensure *no complex* in the test set shares 30-40% identity with a training partner, near identical ligand chemistry, and structurally similar interfaces.

## 6.4 Chemical and small molecule clustering

Small molecule datasets require careful handling; it is recommended to use **Tanimoto similarity** to cluster based on standard molecular fingerprints.

Recommended thresholds include:

- Tanimoto ≥ 0.7-0.8 are highly similar analogs and should remain in the same split
- Tanimoto ≥ 0.5-0.6 are moderately similar

# Part IV — Hugging Face Integration

## 4.1 Why Hugging Face?

Hugging Face provides a standardized, accessible platform for hosting, versioning, and distributing curated datasets. These can be easily browsed, downloaded, and loaded through a consistent API. Hugging Face is powered by Git and therefore supports version control, documentation, and community access.

## 4.2 Dataset Card Requirements

All datasets uploaded to Hugging Face must include a dataset card that documents the essential information for users. At minimum, dataset cards should describe the dataset's purpose, source, curation steps, processing files, data fields, and licensing information.

A template and detailed instructions are available at:
https://huggingface.co/spaces/RosettaCommons/MolecularDatasetCurationGuide

# Part V — Complications + Pain Points

A running list of pain points and complications encountered when curating a dataset are:

- Horrendous pickle files
- Buried zip archives
- Missing metadata
- Inconsistent units
- No dataset index
- Chemical representation complexity (metallo enzymes)
- RDKit limitations
- AtomWorks backend considerations
- Lack of centralized dataset list