UNIVERSTIY OF HAMBURG

MASTER THESIS

Predicting Protein Crystallization Conditions using Machine Learning

Author:
Michael HÜPPE

Supervisors:
Prof. Dr. Fabian Michael Kern
Prof. Dr. Aymelt Itzen

A thesis submitted in fulfillment of the requirements for the degree of Master of Science

in the

Machine Learning in Bio Informatics
Department of Computer Science

Sunday 23rd November, 2025

Declaration of Authorship

I, Michael HÜPPE, declare that this thesis titled, "Predicting Protein Crystallization Conditions using Machine Learning" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"This one is for the boys with the booming system"

Nicki Minaj

UNIVERSTIY OF HAMBURG

Abstract

Faculty Name
Department of Computer Science

Master of Science

Predicting Protein Crystallization Conditions using Machine Learning

by Michael HÜPPE

The Thesis Abstract is written here (and usually kept to just this page). The page is kept centered vertically so can expand into the blank space above the title too...

Acknowledgements

The acknowledgments and the people to thank go here, don't forget to include your project advisor...

Contents

List of Figures

List of Tables

List of Abbreviations

Physical Constants

Speed of Light $c_0 = 2.997\,924\,58 \times 10^8\,\mathrm{m\,s^{-1}}$ (exact)

List of Symbols

a	distance	m
P	power	W ($\mathrm{J\,s^{-1}}$)
ω	angular frequency	rad

For/Dedicated to/To my…

Chapter 1

Decoding the Crystal Recipe: Predicting Protein Crystallization Conditions via Machine Learning

1.1 Introduction

Protein crystallization is the process of arranging purified protein molecules into a highly ordered, repeating lattice that forms a crystal. This crystalline state is essential for X-ray crystallography, the most widely used method for determining high-resolution protein structures. When X-rays are diffracted by a protein crystal, the resulting patterns allow reconstruction of the electron density and ultimately the atomic structure of the protein.
Crystallization is crucial because structural information provides fundamental insights into protein function and interactions which build the basis of structure-based drug design. Here binding sites derived from the structure enables rational development of small molecules that inhibit or modulate the protein's biological function.
Although AlphaFold predictions often align remarkably well with experimentally determined structures, they are not a substitute for them. Terwilliger *et al*. (2024) argue that AlphaFold accelerates, but cannot replace, experimental structure determination due to its varying local accuracies and occasional failures at the global structural level. This reinforces that, despite the breakthroughs brought by AlphaFold, further methodological innovation in structure determination and crystallography remains essential. Beyond crystallography, protein crystals are also used in neutron diffraction, cryo-electron microscopy benchmarking and biophysical studies of stability and folding.
Because most proteins do not readily form diffraction-quality crystals, the main bottleneck in structural biology is identifying the crystallization conditions under which a given protein will form suitable crystals (Mall *et al*., 2025). Despite advances in structure prediction, identifying suitable crystallization conditions, such as buffer type, pH, salts, and precipitants remains largely empirical and often requires screening hundreds of combinations. This process is both time-consuming and costly, with a high failure rate.

At the same time, extensive data from the Protein Data Bank (PDB) and accurate structural predictions from models like AlphaFold2 have become widely available (Jumper *et al*., 2021). These resources offer a unique opportunity to explore whether machine learning models can predict suitable crystallization conditions directly from protein sequence and/or structure.

1.2 Related Works

1.2.1 Challenge of X-ray Crystallography

Crystallizing a protein is often the bottleneck in X-ray crystallography. Only a small fraction (roughly 2–10%) of proteins produce diffraction-quality crystals, meaning over 90% of crystallization trials fail (Mall *et al.*, 2025). This trial-and-error process is costly, where >70% of the total expense in structure determination is spent on attempts not producing crystals of diffraction quality (Mall *et al.*, 2025). Achieving crystals requires finding the right crystallization conditions (e.g. precipitant chemicals, salts, pH, temperature), but currently these must be determined empirically by screening hundreds or thousands of conditions, which demands a lot of protein (McPherson & Gavira, 2013). This reality motivates computational approaches to predict either whether a protein is likely to crystallize (crystallization propensity) or even which specific conditions might lead to crystals. The former is a widely researched topic with models such as CrystalP2 (Kurgan *et al.*, 2009) or PPCPred (Mizianty & Kurgan, 2011), the latter however has received little to no attention (Jin *et al.*, 2022). While crystallization robots have eased the burden of manual search, discovering optimal conditions still means testing thousands of solutions, wasting protein in the process (Wilson & DeLucas, 2014). Any such predictive model could significantly reduce experimental screening, saving time and cost.

1.2.2 Predicting Protein Crystallization

As mentioned predicting crystallization propensity is a well established research area. Early **classical ML** relied on hand-crafted features such as amino-acid composition, the proteins isoelectric point, hydrophobicity, disorder, predicted structure and classifiers (SVM, RF, LR, GB). Matinyan *et al.* (2024) summarizes multiple tools such as:

- *XtalPred*/XtalPred-RF: feature-distribution scoring.
- *TargetCrys*: two-layer SVM ensemble.
- *Crysalis* (2016): integrated predictions + mutation suggestions.
- *BCrystal* (2020): XGBoost + SHAP-based feature selection.
- *DCFCrystal* (2021): cascaded RF stages (expression → purification → crystallization), with a membrane-protein branch.

These achieved moderate accuracy (60–75%, MCC 0.4–0.6) but demanded expert feature engineering. Deep learning automatically extracts sequence patterns as shown in superior performances presented by:

- *DeepCrystal* (2019): multi-scale CNN on one-hot sequences; ∼ 83% accuracy, MCC 0.66.
- *CLPred* (2020): CNN+BLSTM; accuracy 85%, MCC 0.70.
- *ATTCrys* (2021): adds multi-head attention; MCC 0.72.
- Structure-infused: *SADeepCry* (2022) uses autoencoder+self-attention on sequence+predicted structure; *GCmapCrys* (2023) employs GNNs on AlphaFold2 contact maps.

1.2.3 Predicting Crystallization Conditions from Sequence (and Structure)

While predicting "will it crystallize?" is useful, a more ambitious goal is to predict the actual crystallization conditions that would make a given protein form crystals. This is a multi-output prediction (the combination of reagents, concentrations, pH, etc. that will work) and more challenging. Thousands of successful crystallization recipes are known (recorded in databases like the Protein Data Bank), but each protein is unique and may crystallize in different conditions, often unpredictably with no known patterns to predict crystallization conditions (Zhang *et al.*, 2022). However, this does not stem from a lack of attention, Kirkwood *et al.* (2015) for example studied correlations between a protein's isoelectric point and the pH of its crystallization buffer. However, only weak or inconsistent trends were found. Ultimately, Kirkwood *et al.* (2015) conclude that these trends are not sufficiently robust to guide initial crystallization-pH selection. Interestingly, even proteins with high sequence similarity did **not** necessarily crystallize under similar conditions, highlighting that small sequence/structure differences can lead to different optimal crystallization cocktails indicating a multidimensional problem.

Liao and Sun (2025) emphasize the importance of crystal packing and intermolecular packing interfaces in determining crystallization conditions. In their work, they present Molecular Assembly Simulation in Crystal Lattice (MASCL), a framework for simulating crystal packing using AlphaFold combined with symmetrical docking. Crystallization conditions are predicted using a patch-based method that quantifies molecular interface similarity between proteins. For a given target protein, proteins with the most similar physicochemical interface descriptors are used as reference points from which test crystallization conditions are chosen. Meaning no de-novo crystallization conditions are constructed.

This pipeline of constructing a "crystal fingerprint" for a given protein and comparing it to previously crystallized proteins was evaluated on lysozyme, a common model protein in crystallization studies. In this test case, the proposed AAI-PatchBag approach successfully identified conditions yielding crystals with the desired packing characteristics. However,the use of lyzosome as a model system has been repeatedly criticized (Chayen & Saridakis, 2001). Particularly in the context of assessing prediction accuracy lysozome benchmarks should be assed with caution, because it is well known and used for its unusually high crystallizability (Ghosh, 2023). It crystallizes across a broad range of pH values without loss of crystal quality (Iwai *et al.*, 2008), and also tolerates wide variations in temperature and salt concentration (Ataka & Asai, 1988).

Nevertheless, for lysozyme specifically, Liao and Sun (2025) show that similarity in crystal packing information has a stronger influence on predicting successful crystallization conditions than sequence or structural homology alone.

In contrast Lee *et al.* (2019) introduced a proof-of-concept deep learning model to map protein sequence to de-novo crystallization conditions. They parsed crystallization records from PDB entries and framed the task as a multi-label classification: for a given protein sequence, predict which "crystallization terms" (e.g. specific buffers, salts, precipitants like PEG, etc.) appeared in successful recipes for that protein. Essentially, the model learns associations between sequence features and the types of reagents or techniques that tend to be used. Remarkably, this sequence-to-condition model did show predictive power. A simple 1-layer CNN could achieve a weighted F1-score around 0.46 on held-out proteins, substantially better than random guessing in this high-dimensional space. The CNN outperformed a fully-connected network, suggesting that local sequence motifs (captured by convolutional filters) were

informative for certain crystallization agents. For example, the authors noted that hydrophilic or charged residues in the sequence strongly influenced buffer and salt predictions (likely because they affect the protein's isoelectric point and solubility). This indicates real biochemical signal: proteins rich in acidic/basic residues might require certain pH buffers or salt conditions, etc., whereas hydrophobic patches might correlate with needing precipitating agents like PEGs or additives.

It's important to emphasize that this research is still early-stage. An F1 of 0.45 means the model is far from perfectly pinpointing the exact crystallization recipe, but it is better than trial-and-error alone and demonstrates that sequence patterns can inform what conditions are likely to work. As a future direction, the authors suggested incorporating more variables (e.g. predicting optimal pH and temperature as continuous values, not just class labels) and using the approach to focus screening on a smaller set of candidate conditions. Nonetheless, this is a promising frontier: even a modest predictor that suggests, say, the top 10 most likely crystallization cocktails for a new protein (instead of blindly testing 1000) would be hugely valuable.

Chapter 2

Theory

2.1 Proteins

2.1.1 Protein Crystallization

2.2 Trees

Chapter 3

Data

3.1 The Protein database

The Protein Data Bank (PDB) is the central repository for experimentally determined three-dimensional structures of proteins, nucleic acids, and complexes. As of 2025, it contains over 220,000 entries, the majority of which are proteins solved via X-ray crystallography. Each entry includes not only the atomic coordinates of the protein structure but also extensive metadata about the experimental conditions used during crystallization. The following outlines the data acquisition, format, and structure to enable data preprocessing and analysis.

3.1.1 Acquisition

Using the RCSB Search API, all entries solved by X-ray diffraction (about 80%) were queried. Entries that did not contain any information about the crystallization at all or the entry was otherwise incomplete were not downloaded. A Python script then converted these identifiers into download links for the mmCIF files (.cif.gz), wrote them to text files, and split the list into several parts to enable parallel downloads via wget. The complete set of structures was downloaded in compressed mmCIF format and stored locally (after decompression to .cif where needed). Due to the large number of entries, the download process took approximately two days. However, the total download time naturally depends on the available internet bandwidth.

3.1.2 Format

The PDB provides structural data in two formats: PDB and CIF. The newer CIF format was chosen because, unlike the fixed-width PDB format, it does not impose size limitations on large structures. In addition, it can represent complex features such as branched carbohydrates and offers greater detail and flexibility than its predecessor. The data items are in the format of '_' + category name + '.' + attribute name. Data categories can be either saved in either key-value or in tabular format. These can be easily parsed into a column vector as is in CSV files. For example the crystallization conditions are saved in key-value format and look like the following for 3P4V:

```
_exptl_crystal_grow.crystal_id      1
_exptl_crystal_grow.method          'VAPOR DIFFUSION, SITTING DROP'
_exptl_crystal_grow.apparatus       None
_exptl_crystal_grow.atmosphere      None
_exptl_crystal_grow.pH              9.5
_exptl_crystal_grow.temp            298.0
_exptl_crystal_grow.pdbx_details    '3.2M (NH4)2SO4, 0.1M Glycine, ...'
_exptl_crystal_grow.time            None
```

A category is stored in tabular format when a token defines multiple values. In this case, loop_ is followed by rows of data-item names, with data values separated by whitespace. Notably, the protein's structural information i.e., the atom_sites category is represented in this format, as shown in the following example:

```
loop_
_atom_site.group_PDB
_atom_site.id
_atom_site.type_symbol
_atom_site.label_atom_id
_atom_site.label_alt_id
_atom_site.label_comp_id
_atom_site.label_asym_id
_atom_site.label_entity_id
_atom_site.label_seq_id
_atom_site.pdbx_PDB_ins_code
_atom_site.Cartn_x
_atom_site.Cartn_y
_atom_site.Cartn_z
ATOM   1    N  N   . VAL A 1 1   ? 6.204   16.869  4.854   1.00 49.05 ...
ATOM   2    C  CA  . VAL A 1 1   ? 6.913   17.759  4.607   1.00 43.14 ...
ATOM   3    C  C   . VAL A 1 1   ? 8.504   17.378  4.797   1.00 24.80 ...
ATOM   4    O  O   . VAL A 1 1   ? 8.805   17.011  5.943   1.00 37.68 ...
```

3.1.3 Structure

In structural biology, one rarely needs to work with all proteins deposited in the PDB. Consequently, the PDB is designed as an entry-oriented resource: users typically retrieve and parse the complete dataset for a single protein of interest. This design is well aligned with the standard workflow in structural biology, where researchers focus on a small number of proteins but require all available structural, experimental, and metadata associated with those entries.

In this project, however, the goal is fundamentally different: to identify patterns across many proteins, using all deposited entries in the database. Parsing a single CIF file takes ~100 ms using the fastest available library, Gemmi which amounts to approximately 9 hours for the entire PDB. Repeating such operations would be not feasible for data analysis where data typically has to be read multiple times across sessions.

To address this, the first step after downloading the database was to restructure it into a feature-based data model, in which information is grouped by category rather than by entry. Each category is stored as a separate Parquet file (also convertible to CSV), where each attribute becomes a column spanning all proteins. This enables highly targeted access: if an analysis requires only a single attribute, the corresponding column can be read directly without loading irrelevant data.

Although the initial restructuring (reading, grouping, and writing) takes ~11 hours, it drastically improves downstream performance. For example, extracting all crystallization-condition information from the entire database now takes ~10s, compared to the 9 hours required to parse every CIF individually.

Storage efficiency is also greatly improved. The full PDB occupies ~46 GB and allocates ~110 GB, making it impossible to load into memory. In contrast, the derived

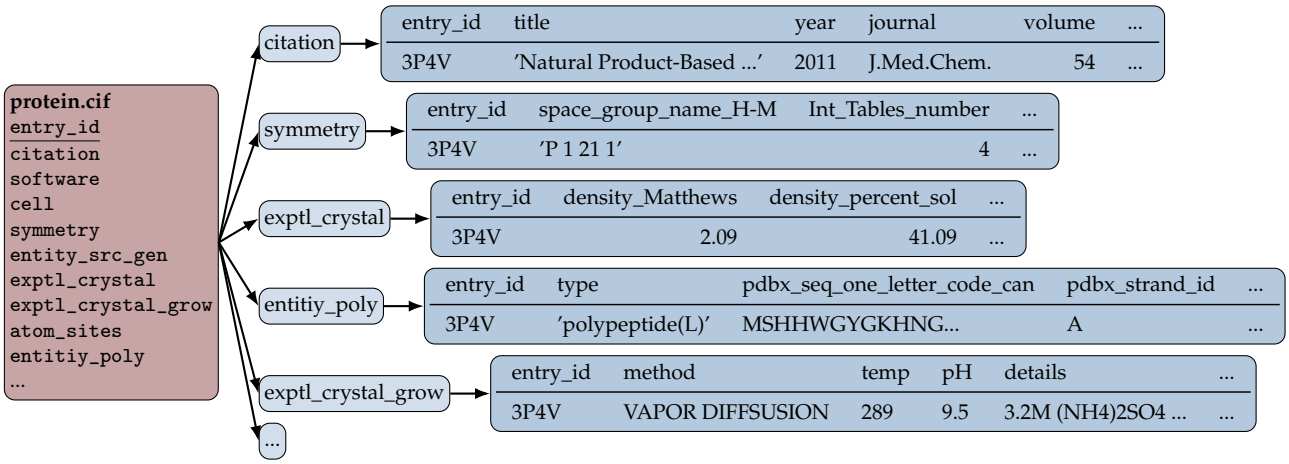


FIGURE 3.1: Visualization of the restructured database format derived from the `protein.cif` file. The `protein.cif` serves as the point of origin and contains all available information for a given protein. In the restructured schema, each CIF category is stored as a separate table whose columns correspond to the category attributes, while each row represents a protein entry. To enable relational operations, all category tables share a common `entry_id` column.

dataset containing only crystallization conditions is about 20 MB, small enough to be memory-resident and repeatedly queried without overhead.

This dramatic reduction in both time and storage requirements arises because most information in CIF files is irrelevant for crystallization-condition prediction. Metadata such as software provenance or publication information is not needed for this project and can be safely excluded from the feature-based representation.

However, to enable analyses that combine multiple categories, the restructured database must be joinable in a way that allows the original information to be reconstructed. This was accomplished using a star schema, in which each category table is indexed by the unique `entry_id` of the protein. This design ensures that information from different categories (such as crystallization conditions and sequence data) can be linked by performing joins on the shared entry identifier. The restructured database format is depicted in Figure 3.1.

3.1.4 Description

After filtering for majorly incomplete entries the total number of deposited proteins was 196743 as of 01.09.2025. Each of these entries has around 700 attributes across 68 categories. However, as mentioned before the minority of these attributes/categories is of importance when predicting crystallization conditions. More specifically, the important categories and their attributes are the following:

Input Categories

The `entity_poly` category of the mmCIF format describes each polymeric entity in a macromolecular structure. A polymer represents a sequence of linked monomers (e.g., amino acids or nucleotides), and this category provides essential metadata about its type, sequence, and representation in the structural model. `entity_id` Unique identifier linking the polymer to the corresponding entry in the `entity` category.

`type` Specifies the polymer type, such as polypeptide(L), DNA, RNA, or polysaccharide.
`nstd_linkage` Indicates whether the polymer contains non-standard chemical linkages (e.g., cross-links or modified connectivity).
`nstd_monomer` Flags the presence of non-standard or modified monomers within the sequence.
`pdbx_seq_one_letter_code` The polymer sequence in one-letter code, including symbols for modified residues.
`pdbx_seq_one_letter_code_can` Canonicalized one-letter sequence where modified residues are mapped to their closest standard equivalents.
`pdbx_strand_id` Lists the chain identifiers (e.g., A, B) representing this polymer in the structure.
`pdbx_target_identifier` Optional external identifier used mainly in structural genomics pipelines.

The `atom_site` category contains the atomic coordinates and related information that define the three-dimensional structure of the macromolecule. Each row corresponds to a single atom and records properties such as atom name, element, residue identifier, chain, Cartesian coordinates, occupancy, and atomic displacement parameters.

Label Categories

The category `exptl_crystal_grow` defines the crystallization conditions. It contains 17 attributes that describe the experimental setup and methodology used to grow the protein crystal. However, as seen in Table 3.1 for 13 of the attributes the majority of entries do not contain any information.

Attribute	Missing percentage
pdbx_details	0.04
method	10.78
temp	10.79
pH	22.30
pdbx_pH_range	75.24
temp_details	98.56
details, seeding	99.83
apparatus, atmosphere, method_ref, pressure, pressure_esd, seeding_ref, temp_esd, time	99.84

TABLE 3.1: The percentage of missing values for each attribute in the crystallization condition category. Only 5 attributes have sufficient data.

Thus, the attributes of relevance are the following:
`method`: Describes the crystallization technique employed, such as vapor diffusion, batch crystallization, or microbatch methods (in free text).
`pH`: Specifies the pH of the crystallization solution, which strongly influences protein stability and crystal formation (numerical).
`temp`: Records the temperature at which the crystallization experiment was performed, typically given in Kelvin or Celsius (in free text).

`pdbx_details`: Provides free-text experimental details, such as buffer components, precipitants, additives, or other conditions important for reproducing the crystallization setup (in free text).

Moreover, it might be of importance how well the protein has crystallized. Crystal quality measures are primarily provided in the `exptl_crystal` category, which describes the physical properties of the crystal, and in the `diffrn` and `reflns` categories, which contain diffraction statistics such as resolution, completeness, and R-factors that reflect the overall quality of the crystal.

3.2 Details Parsing

entry_id	pdbx_details
3P4H	Crystallization was carried out in sitting-drop vapor-diffusion setups with 1:1 mixtures of protein solution containing 0.7 mM Cko and 1.8 mM MnCl2 and reservoir solution containing 20% PEG 3350 and 0.2 M Na2HPO4 , pH 9.5, VAPOR DIFFUSION, SITTING DROP, temperature 298K
3P4V	3.2M (NH4)2SO4, 0.1M Glycine, pH 9.5, VAPOR DIFFUSION, HANGING DROP, temperature 289K
3P62	100mM sodium cacodylate, 100 mM sodium acetate, 16-18% isopropanol, pH 6.2, VAPOR DIFFUSION, SITTING DROP, temperature 293K
3PCA	pH 8.4
6LJR	PEG
3PF1	15-17% PEG 4K 0.2M KCl, protein dialysed in 10mM NaOAc 50mM NaCl, 10% glycerol 0.4%C8E4, pH 5.5, VAPOR DIFFUSION, HANGING DROP, temperature 295K

TABLE 3.2: Examples of free text descriptions of crystallization conditions given in the PDB. The examples show the high variety in possible expressions of crystallization conditions.

The examples in Table 3.2 illustrate the complexity involved in parsing protein crystallization conditions from free text into a uniform, machine-readable chemical cocktail representation. The first entry (3P4H) demonstrates that crystallization conditions are often embedded in long, narrative-style descriptions that mix experimental setup, protein concentration, co-factors, and reservoir composition in a single sentence. Extracting a structured cocktail from such text requires an algorithm that can reliably identify and segment chemically relevant entities (salts, buffers, precipitants, additives) and distinguish them from procedural information.

The second entry (3P4V) highlights the lack of uniformity in how units and concentrations are reported. Here, components are given in varying molar units, whereas other entries in the table use percentages. A robust parsing pipeline must therefore handle multiple concentration formats, convert them into a common representation. The third entry (3P62) further emphasizes semantic heterogeneity in naming: chemicals may be referred to by systematic names (e.g. “sodium cacodylate”, “sodium acetate”) rather than by their chemical formulas, in contrast to entries that use formula-based names (e.g. $MnCl_2$, Na_2HPO_4). Peat *et al.* (2005) illustrated

the difference in naming conventions by showing the 30 different ways ammonium sulfate is spelled in the dataset. Using string matching the updated dataset presented 141 different spelling attributed to ammonium sulfate. This necessitates normalization against a chemical dictionary or ontology to map synonymic names and formulas to a shared canonical identifier.

Entry 6LJR and 3PCA illustrates that some `pdbx_details` fields contain only minimal useful information. Such cases require the parser to handle incomplete cocktails and to distinguish between genuinely missing data and conditions that were simply not reported. The variability in reporting detail is substantial: while the average description length is 93 characters (SD = 72), the lengths range from as little as a single character to as much as 1758 characters. This large spread underscores the heterogeneity and inconsistency in the level of detail provided across entries.

Finally, the entry 3PF1 illustrates more subtle challenges, including the presence of concentration ranges, ambiguous abbreviations (PEG 4K instead of PEG 4000), and incomplete specifications (e.g. "%" without an explicit indication of whether it is w/v or v/v). Additionally, the same text string can interleave multiple components without clear delimiters, making tokenization and assignment of units to the correct solute non-trivial. These examples underscore that converting free-text crystallization descriptions into uniform chemical cocktails is not a simple extraction task, but a complex natural language processing and normalization problem that must account for heterogeneous syntax, inconsistent units, synonymic naming, incomplete information, and domain-specific ambiguities.

3.2.1 Pipeline

Lynch *et al*. (2020) constructed a multi-stage extraction pipeline designed to transform the unstructured crystallization metadata contained in the PDB into a consistent format. The goal of their proposed workflow is to impose a controlled vocabulary on the free text description and thereby enable large-scale analyses of crystallization conditions. The complete procedure consists of the four major steps 1. data acquisition, 2. details parsing and text normalization, 3. curating a compound dictionary to create a controlled vocabulary, and finally 4. the construction of the crystallization details dataset. In the following, the focus is on the actual parsing as their dataset was not used and the controlled vocabulary was heavily expanded upon using string matching algorithms instead of manual specification.

The address the variation in wording, punctuation, spelling and chemical names Lynch *et al*. (2020) designed a custom parsing function that performs multiple passes over the raw text. This function extracts: chemical component names, their reported concentrations (if present) and incubation temperatures as well as pH level. The parser explicitly handles inconsistent spacing and irregular phrasing that commonly appear in the deposited records. However, it does not handle punctuation or typographical errors. These were added to the pipeline to improve the parsing. To identify misspellings and variant forms of chemical names in the dataset, a fuzzy string-matching approach was employed. First, all chemical names occurring more than 200 times were treated as reliable entries and were collected into a reference set of *correct chemicals*. All remaining chemical names, which appeared less frequently, were considered potential misspellings or variants.

For each infrequent chemical name c_{var}, the algorithm computed its similarity to all names in the reference set using a normalized edit-distance metric. Specifically,

the similarity score was based on the Levenshtein ratio, which measures the minimum number of character insertions, deletions, or substitutions required to transform one string into another, normalized by the maximum string length. This score ranges from 0 (no similarity) to 100 (identical strings).

For every variant c_{var}, the algorithm selected the most similar reference chemical c_{ref} and evaluated whether their similarity exceeded a threshold of 90 %. If this condition was met, the two names were considered variants of the same underlying chemical, and a mapping $c_{\text{var}} \mapsto c_{\text{ref}}$ was created. This procedure ensured that common spelling errors, hyphenation differences, and minor typographical variations were systematically corrected and unified across the dataset.The string similarity between chemical names can be misleadingly high, which required manual inspection of the resulting mappings. For example, ammonia and ammonium share a similarity of approximately 80 %, yet they are chemically distinct and should not be merged. Likewise, even with a threshold of 90 %, names such as disodium malonate and sodium malonate or hexanediol and heptanediol were incorrectly matched, despite referring to different compounds. To avoid such false positives, mappings in which the difference between source and target consisted solely of common chemical modifiers (e.g., di-, tri-, mono-, hydrogen, etc.) or while chemical names ('sodium ammonium tartrate' -> 'sodium potassium tartrate') were removed, as these often indicate genuinely different forms rather than typographical variants. This resulted in around 25 % of the false chemicals being successfully mapped to a more common name.

However, some examples were too unique to be added to the parsing pipeline. An example is 8C9L for which the entered description is "'0.1MBis Tris Propane pH 6.50.02 MSodium potassium phosphate pH 7.520 % w/vPEG 335010% v/vEthylene glycol'". Here the problem is that after each numerical value a space is missing not being clear if a pH of 7.5 is meant. Implementing to many anomalies interfered with the parsing of the more typical descriptions.

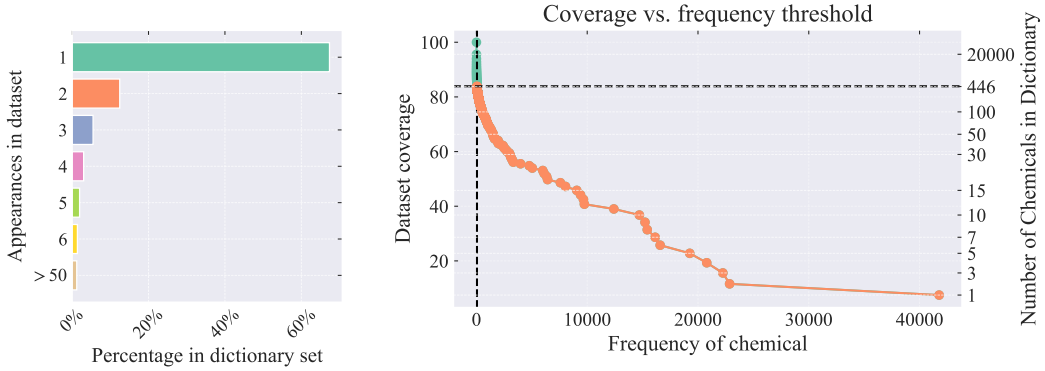
3.2.2 Parsing Quality

However, the parsing was not perfect in that it identified words as chemicals even though they were not. Originally, the parsing resulting in the majority of chemicals being parsed being present only once. This resulted in a total unique chemical count of around 30 000. This can be seen in Figure 3.2a.

Similar to the string matching approach described in subsection 3.2.1 the first step to reduce the number of unique chemicals was to check for common string patterns. This included for example the token "crystal tracking id <crystal id>" with part of the crystal id typically being picked up as the concentration. This removed around 350 of the faulty chemical names. Similarly, sometimes verbs were also identified as proteins. Using Spacy all chemicals that only occured once were screened. Verbs were removed after confirming that the model can successfully distinguish between them and chemical names. This got rid of around 2400 false chemicals.

After applying the string-matching approach, all chemicals occurring fewer than 50 times were removed from the parsed set. This threshold was chosen based on the coverage analysis shown in Figure 3.2b.The plot shows that the top 446 most occurring chemicals cover around 84 % of the dataset. When taking out every name that only occurred once and is thus very likely due to a parsing error the data coverage is above 98%. This increased the confidence that the chemical cocktails described in the free text were accurately captured by the parsed dictionaries.

PH and temperature values were sometimes provided in the free-text field but not in the corresponding numerical fields. After extracting and parsing pH and temperature from the free text, the proportion of missing values decreased to 9.2% for pH and 8.3% for temperature.



(A) Occurrence counts of unique chemical names extracted from free text. Although 25,000 unique names were parsed, 67% occur only once—likely due to typographical or parsing errors.

(B) Coverage of the dataset as a function of chemical name frequency. Applying a frequency threshold of 50—i.e., requiring a chemical to be mentioned at least 50 times before inclusion in the chemical dictionary—removes 99% of unique names, leaving 446 entries while still covering 84% of the dataset. From the dictionary set only 1.2% of unique chemical names occur more than 50 times in the dataset.

FIGURE 3.2: As shown in Figure 3.2a, most chemicals appear only a few times, and all parsed names represent only a small fraction of chemically meaningful compositions. This is also seen by the density of points and the almost logarithmic scaling of right y axis in Figure 3.2b.

The parsing process aimed not only to extract chemical compounds from the free text but also to recover pH and temperature values. A straightforward way to assess this is to compare the parsed numerical values with those entered in the designated PDB fields, as shown in Figure 3.3. The parsed and entered values correlate strongly approximately 96% for pH and nearly perfectly for temperature as seen in Figure 3.3b. Inspection of the remaining pH (Figure 3.3a) mismatches shows that they primarily come from crystallographers reporting multiple pH values for different solutions, rather than from parsing errors. The few temperature mismatches likewise stem from genuine discrepancies between the entered and reported values. In such cases, the value provided in the numerical field was preferred.

3.3 Data Analysis

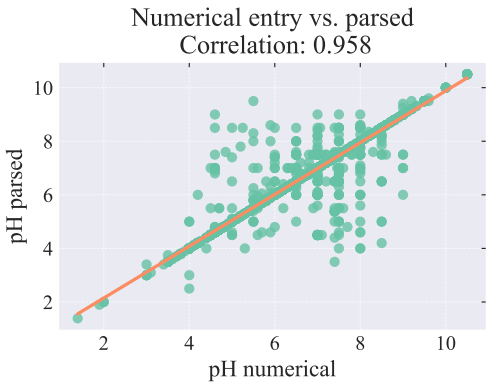
3.3.1 Univariate Analysis

Histograms, distributions, summary statistics

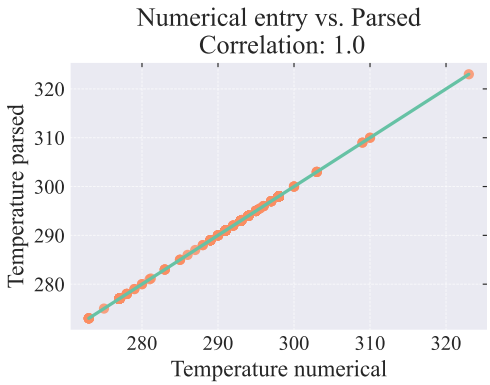
3.3.1.1 Input

3.3.1.2 Label

After parsing it can be seen that the descriptions that do not contain any information about the chemical cocktail is ~3.5 %. They are very similar to entry 3PCA in

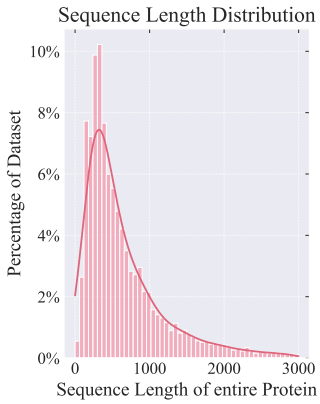


(A) Comparison of pH values parsed from free text with the single pH value recorded in the numerical field.

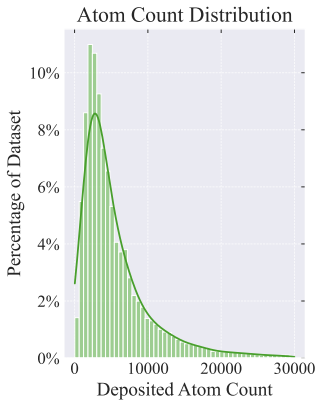


(B) Comparison temperature values parsed from free text with the numerical field entered temperature value.

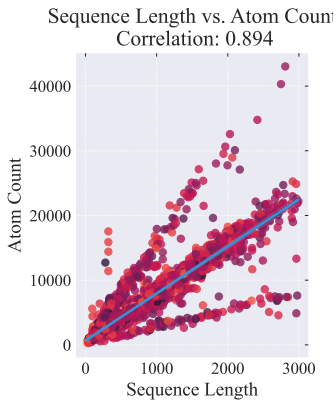
FIGURE 3.3: Multiple pH values reported in free text for different solutions leads to mismatches. In contrast the temperature field for crystallization produced almost no mismatches since only one value was given in free text.



(A) Sequence Length Distribution



(B) Atom count Distribution



(C) Atom count vs. Sequence Length

FIGURE 3.4: Visualization of the distribution of sequence lengths (Figure 3.4a) and deposited atom counts (Figure 3.4b). Most proteins have sequences between 50 and 400 residues, with models containing roughly 1,000–8,000 atoms. The relationship is largely linear as seen in Figure 3.4c, though missing or extra atoms in some structures cause deviations in slope.

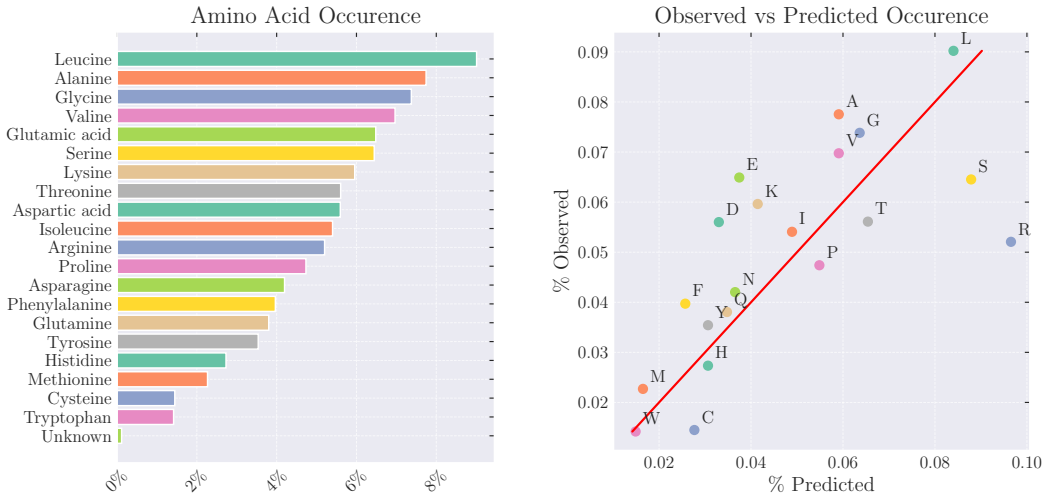
that they only define the pH or temperature. Additionally, concentrations were attributed to chemicals in ~88% of the times.

3.3.2 Multivariate Analysis

Feature-feature relationships, correlation matrices, scatter plots

3.3.3 Outlier and Anomaly Detection

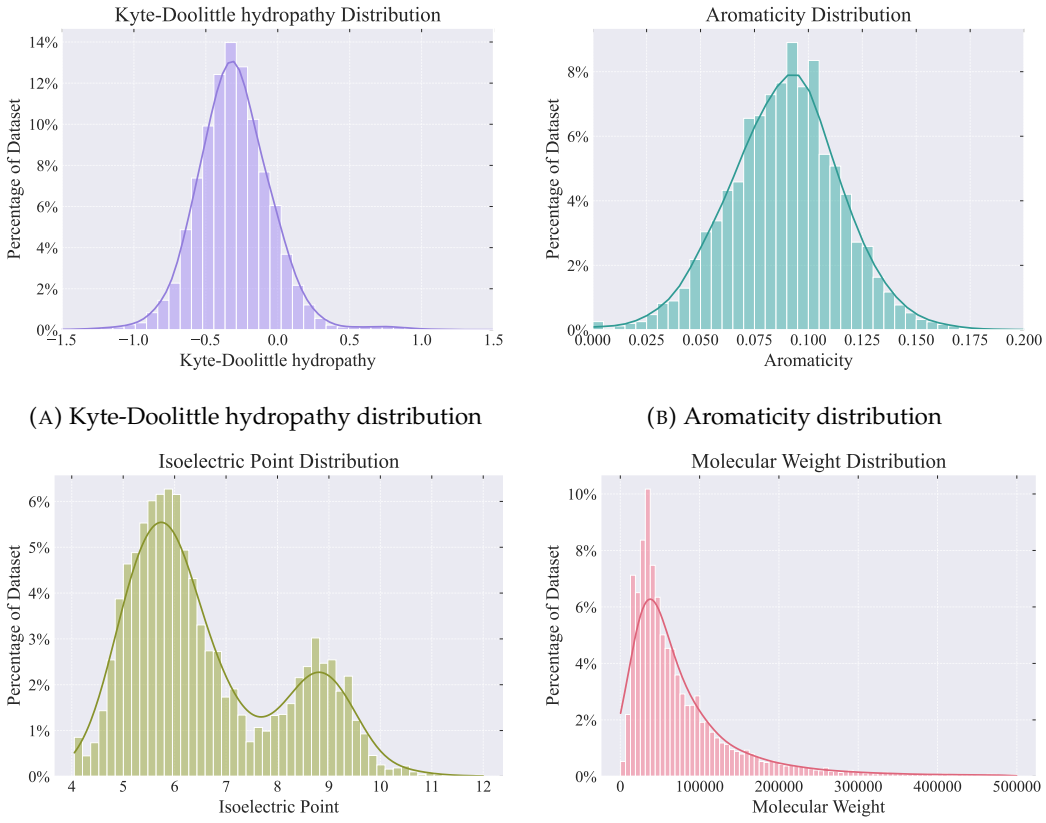
Identifying unusual values or patterns



(A) Distribution of amino acids in the sequences. Residue frequencies vary substantially—Leucine is over four times more common than Tryptophan.

(B) Observed vs. predicted residue frequencies, illustrating the relationship between amino-acid occurrence and the codon patterns that encode them.

FIGURE 3.5: Figure 3.5a shows the distribution of amino acids across all sequences, while Figure 3.5b compares observed residue frequencies with those predicted from codon usage.



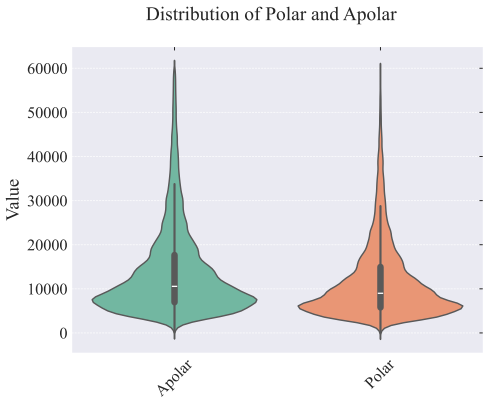
(A) Kyte-Doolittle hydropathy distribution

(B) Aromaticity distribution

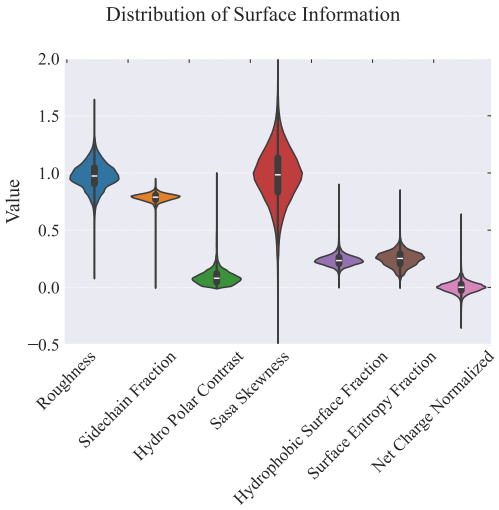
(C) Isoelectric point distribution

(D) Molecular weight distribution

FIGURE 3.6: Distributions of sequence-derived features across the dataset. Kyte–Doolittle hydropathy scores are roughly normally distributed with a slightly negative mean, while aromaticity values form a mildly positive normal distribution. The isoelectric point shows a bimodal pattern with peaks near pH 6 and pH 9. Molecular weights decrease sharply beyond 50,000 Da, mirroring trends in sequence length and atom count seen in Figure 3.4.

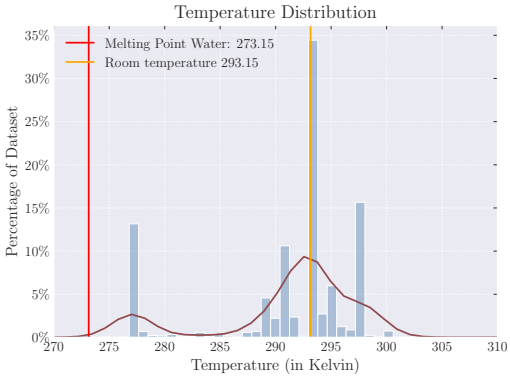


(A) Distribution of apolar vs. polar surface atoms.

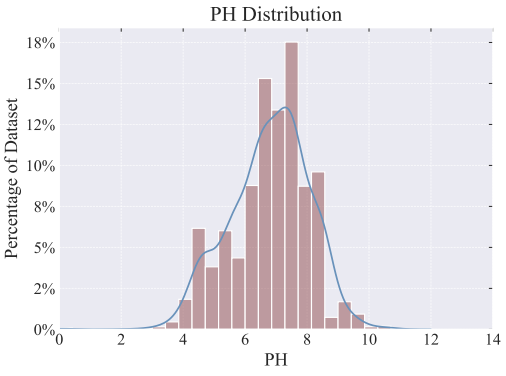


(B) Comparison of different surface features.

FIGURE 3.7: Distributions of derived surface features. Apolar and polar surface atoms show approximately normal distributions with comparable overall exposure. Surface roughness, sidechain fraction, and SASA skewness are centered around 1, while the remaining surface descriptors cluster near zero or show slight positive shifts.



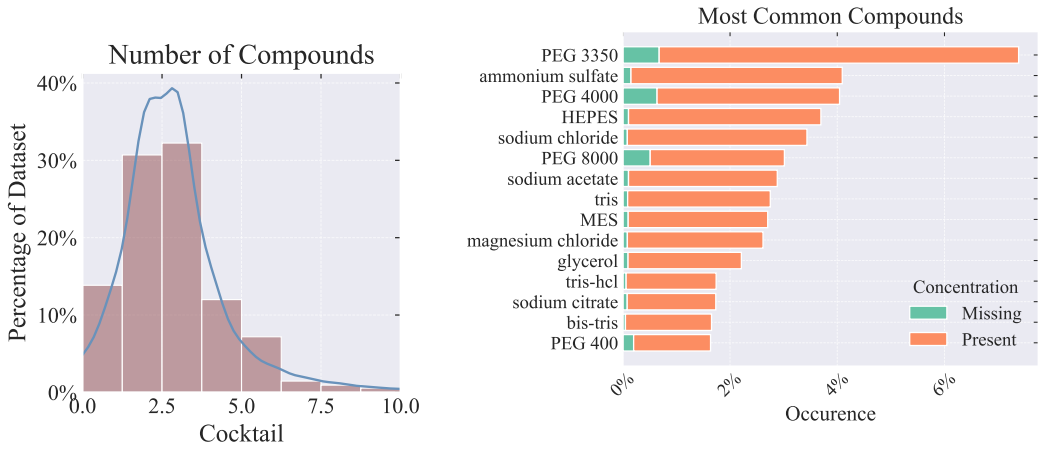
(A) Temperature Distribution



(B) pH Distribution

PH Distribution

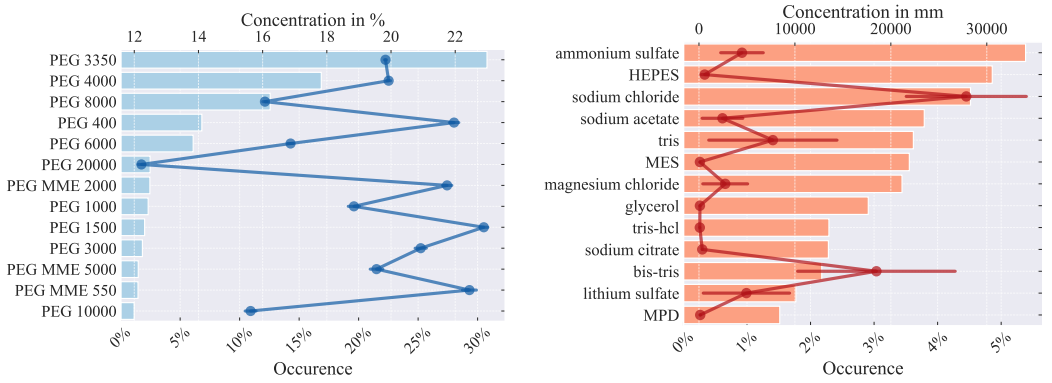
FIGURE 3.8: Distribution of numerical fields in the condition category. Only a few distinct temperatures are reported, with 35% of samples specifying room temperature and most remaining values clustered around it; a secondary peak appears at 4 °C. In contrast, pH values follow an approximately normal distribution centered near pH~7.



(A) Number of compounds in a cocktail is on average 3 but sometimes specifies as much as 10 chemicals.

(B) The most common compounds across all cocktails feature multiple PEGs, buffers and salts. For the majority of the compounds a concentration value is given.

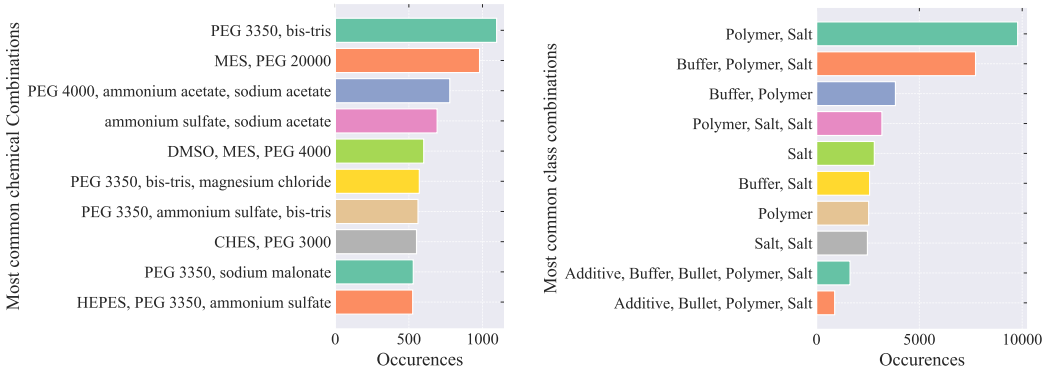
FIGURE 3.9: General Cocktail composition



(A) Most common PEGs and the average concentration value at which they occur

(B) Most common chemicals and the average concentration value at which they occur

FIGURE 3.10: General Cocktail composition



(A) Most frequent specific chemical combinations found in the crystallization conditions, showing the dominant pairs and triplets of reagents across the dataset.

(B) Most frequent combinations of chemical classes occurring together in crystallization setups, highlighting the typical pairing patterns between polymers, buffers, and salts.

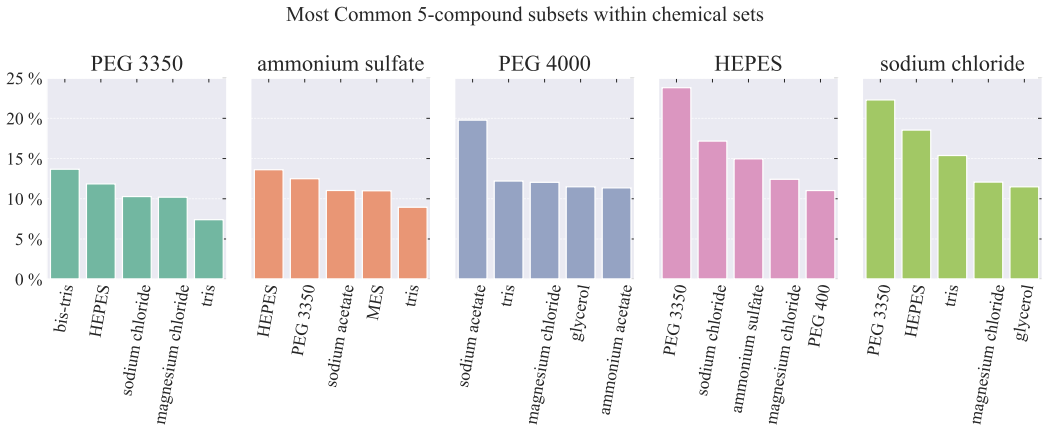


FIGURE 3.12: Most common 5-compound subsets within chemical sets. For several highly frequent chemicals—including PEG 3350, ammonium sulfate, PEG 4000, HEPES, and sodium chloride—the plot shows the compounds most often co-occurring with them in crystallization conditions, highlighting typical pairing patterns among buffers, salts, polymers, and additives.

Chapter 4

Results

Chapter 5

Discussion

Appendix A

Data Appendix

Bibliography

Ataka, M., & Asai, M. (1988). Systematic studies on the crystallization of lysozyme: Determination and use of phase diagrams. *Journal of Crystal Growth*, *90*(1), 86–93. https://doi.org/10.1016/0022-0248(88)90302-8

Chayen, N. E., & Saridakis, E. (2001). Is lysozyme really the ideal model protein? *Journal of Crystal Growth*, *232*(1), 262–264. https://doi.org/10.1016/S0022-0248(01)01203-9

Ghosh, R. (2023). Membrane-Based Micro-Volume Dialysis Method for Rapid and High-Throughput Protein Crystallization. *Processes*, *11*(7), 2148. https://doi.org/10.3390/pr11072148

Iwai, W., Yagi, D., Ishikawa, T., Ohnishi, Y., Tanaka, I., & Niimura, N. (2008). Crystallization and evaluation of hen egg-white lysozyme crystals for protein pH titration in the crystalline state. *Journal of Synchrotron Radiation*, *15*(Pt 3), 312–315. https://doi.org/10.1107/S0909049507059559

Jin, C., Shi, Z., Kang, C., Lin, K., & Zhang, H. (2022). TLCrys: Transfer Learning Based Method for Protein Crystallization Prediction. *International Journal of Molecular Sciences*, *23*(2), 972. https://doi.org/10.3390/ijms23020972

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., … Hassabis, D. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, *596*(7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2

Kirkwood, J., Hargreaves, D., O'Keefe, S., & Wilson, J. (2015). Using isoelectric point to determine the pH for initial protein crystallization trials. *Bioinformatics*, *31*(9), 1444–1451. https://doi.org/10.1093/bioinformatics/btv011

Kurgan, L., Razib, A. A., Aghakhani, S., Dick, S., Mizianty, M., & Jahandideh, S. (2009). CRYSTALP2: Sequence-based protein crystallization propensity prediction. *BMC Structural Biology*, *9*(1), 50. https://doi.org/10.1186/1472-6807-9-50

Lee, H., Wu, Z. H., Corbi-Verge, C., Mok, M., Kang, S., Liao, S., Zhang, Z., & Garton, M. (2019). De novo crystallization condition prediction with deep learning [Accessed: 2025-07-31]. *Advances in Neural Information Processing Systems*, *32*. https://mlcb.github.io/mlcb2019_proceedings/papers/paper_3.pdf

Liao, K.-J., & Sun, Y.-J. (2025). Using AlphaFold and Symmetrical Docking to Predict Protein–Protein Interactions for Exploring Potential Crystallization Conditions. *Proteins*, *93*(10), 1747–1766. https://doi.org/10.1002/prot.26844

Lynch, M. L., Dudek, M. F., & Bowman, S. E. J. (2020). A Searchable Database of Crystallization Cocktails in the PDB: Analyzing the Chemical Condition Space. *Patterns*, *1*(4), 100024. https://doi.org/10.1016/j.patter.2020.100024

Mall, R., Kaushik, R., Martinez, Z. A., Thomson, M. W., & Castiglione, F. (2025). Benchmarking protein language models for protein crystallization. *Scientific Reports*, *15*(1), 2381. https://doi.org/10.1038/s41598-025-86519-5

Matinyan, S., Filipcik, P., & Abrahams, J. P. (2024). Deep learning applications in protein crystallography. *Acta Crystallographica. Section A, Foundations and Advances*, *80*(Pt 1), 1–17. https://doi.org/10.1107/S2053273323009300

McPherson, A., & Gavira, J. A. (2013). Introduction to protein crystallization. *Acta Crystallographica. Section F, Structural Biology Communications*, *70*(Pt 1), 2–20. https://doi.org/10.1107/S2053230X13033141

Mizianty, M. J., & Kurgan, L. (2011). Sequence-based prediction of protein crystallization, purification and production propensity. *Bioinformatics*, *27*(13), i24–i33.

Peat, T. S., Christopher, J. A., & Newman, J. (2005). Tapping the Protein Data Bank for crystallization information. *Acta Crystallographica. Section D, Biological Crystallography*, *61*(Pt 12), 1662–1669. https://doi.org/10.1107/S0907444905033202

Terwilliger, T. C., Liebschner, D., Croll, T. I., Williams, C. J., McCoy, A. J., Poon, B. K., Afonine, P. V., Oeffner, R. D., Richardson, J. S., Read, R. J., & Adams, P. D. (2024). AlphaFold predictions are valuable hypotheses and accelerate but do not replace experimental structure determination. *Nature Methods*, *21*(1), 110–116. https://doi.org/10.1038/s41592-023-02087-4

Wilson, W. W., & DeLucas, L. J. (2014). Applications of the second virial coefficient: Protein crystallization and solubility. *Acta Crystallographica Section F: Structural Biology Communications*, *70*(5), 543–554. https://doi.org/10.1107/S2053230X1400867X

Zhang, X., Xu, Z., Zhou, J., Xing, X., & Li, L. (2022). Enhancement of Protein Crystallization Using Nano-Sized Metal–Organic Framework. *Crystals*, *12*(5), 578. https://doi.org/10.3390/cryst12050578