

1 Human-in-the-loop approach to identify functionally 2 important residues of proteins from literature

3 **Melanie Vollmar^{1,*}, Santosh Tirunagari², Deborah Harris¹, David Armstrong¹, Romana
4 Gaborova¹, Deepti Gupta¹, Marcelo Querino Lima Afonso¹, Genevieve Evans¹, and
5 Sameer Velankar¹**

6 ¹Protein Data Bank in Europe, European Molecular Biology Laboratory, European Bioinformatics Institute
7 (EMBL-EBI), Wellcome Genome Campus, Hinxton, Cambridge, CB10 1SD, UK

8 ²Literature Services, European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI),
9 Wellcome Genome Campus, Hinxton, Cambridge, CB10 1SD, UK

10 *corresponding author(s): Melanie Vollmar (melaniev@ebi.ac.uk)

11 ABSTRACT

We present a novel system that leverages curators in the loop to develop a dataset and model for detecting residue-level functional annotations and other protein structure features from standard publication text. Our approach involves the integration of data from multiple resources, including PDBe, EuropePMC, PubMedCentral, and PubMed, combined with annotation guidelines from UniProt, while employing LitSuggest and Huggingface models as tools in the annotation process.

A team of seven annotators manually curated ten articles for named entities, which we utilized to train a starting PubmedBert model from Huggingface. Using a human-in-the-loop annotation system, we developed the best model with commendable performance metrics of 0.90 for precision, 0.92 for recall, and 0.91 for F1-measure.

Our proposed system showcases a successful synergy of machine learning techniques and human expertise in curating a dataset for residue-level functional annotations and protein structure features. The results demonstrate the potential for broader applications in protein research, bridging the gap between advanced machine learning models and the indispensable insights of domain experts.

13 Background & Summary

The three-dimensional (3D) structure of a protein determines its function and provides insights into its mechanisms and processes within a cell. In order to understand biology and its intricate systems, it is essential to determine protein structures, analyze them on a residue level and identify which residues are the key to its function. For more than 50 years the Protein Data Bank (PDB) managed by the wwPDB partners¹ (<https://www.wwpdb.org/>) has been the go-to data resource to access experimentally determined protein structures. The team at Protein Data Bank in Europe (PDBe)² (<https://www.ebi.ac.uk/pdbe/>), as one of the founding partners in the wwPDB, processes and curates a couple of hundred new, experimentally derived structure submissions every week. In a unified process, followed by all the wwPDB data centers, they provide a standard set of annotations³ alongside the atomic coordinates for each structure. However, the structures are deposited before the publication is available, which prevents biocurators from accessing additional knowledge hidden in scientific literature to support and enrich the protein structure data.

To better understand the structure-function relationship for a protein and its relevance in biological context, it would be beneficial to access additional knowledge locked away in unstructured text in scientific publications. For over 20 years, UniProt⁴ (<https://www.uniprot.org/>) has developed processes to manually curate scientific literature and enrich protein sequences, the linear, one-dimensional representation of a protein. However, with a near-exponential increase in publication rate and sequences released, it is impossible to comprehensively extract residue-level functional knowledge from literature at scale and annotate structures or sequences solely through manual curation.

Here, we present a workflow to develop a transformer-based named entity recognition (NER) system to annotate full-text publications as the first step to an automated pipeline to extract residue-level functional information from the literature to provide 3D-structure based protein annotations. The presented algorithm achieved high overall precision, recall and F1-measure scores of 0.90, 0.92 and 0.91, respectively.

Identified annotations can be used to highlight key text spans in publications but can also serve as a starting point for future developments. Entity types and their relationships can be collated across all publications for a particular protein.

36 Applying reasoning and weighing to the identified information may help to discover new knowledge and insight into the
37 intricate systems a protein is involved in within cells and organisms, such as new interaction partners or signaling and
38 metabolic pathways, and how these systems are dependent on a protein's structure-function relationship. Furthermore, these
39 annotations can also serve as a validation source for assessing the biological relevance of predicted protein models. These
40 predicted models are generated by deep learning algorithms such as AlphaFold⁵ and RosettaFold⁶. AlphaFold Protein Structure
41 Database (AlphaFold DB)⁷ (<https://alphafold.ebi.ac.uk/>) contains predicted models provided by DeepMind
42 (<https://deepmind.google/>) while other computationally created models can be deposited to the ModelArchive⁸
43 (<https://modelarchive.org/>). Large-scale structure predictions can also be made accessible by establishing a 3D-
44 Beacon and integrating it into the 3D-Beacons network⁹ (<https://www.ebi.ac.uk/pdbe/pdbe-kb/3dbeacons/>)
45 which is designed to improve FAIRness¹⁰ (Findability, Accessibility, Interoperability and Reusability) of experimental and
46 predicted structure models. Nowadays, the computationally generated models achieve similar quality for chemical and physical
47 descriptors, such as geometry and bond lengths and angles, to those found for experimentally determined structures, at least in
48 areas with high-confidence predictions. Unlike structures in the PDB, these predictions are not supported by experimental
49 evidence like electron density maps or electric potential maps as determined through X-ray crystallography or cryo-electron
50 microscopy, respectively, or chemical shifts from nuclear magnetic resonance. Consequently, analyzing a predicted protein
51 model in isolation is prone to misinterpretation of a residue's location and its interactions with neighbors. However, non-
52 structural publications can provide information on a variety of biochemical and mutational studies which can be used to check
53 if the predicted models are supported by the observations drawn from these studies. If predicted models can explain the
54 observations, the predicted models could be considered functionally validated.

55 Methods

56 The annotation team

57 Project manager

58 The project manager was the lead for the annotation project with more than 15 years of experience in structural biology,
59 more than 500 protein structures in the PDB, and over seven years experience in developing software and machine learning
60 algorithms. As lead, the project manager was responsible for the general management, planning and documentation of the
61 project and was involved in the annotation process.

62 Annotators

63 A team of six PDBe biocurators, involved in the curation of protein structures submitted to the PDB, volunteered in the
64 annotation process. All but one had a PhD, either in biochemistry, bioinformatics or structural biology with a strong background
65 in biochemistry and/or structural biology. Combined, they had 10 years of experience in bioinformatics, 24 years in biochemistry
66 and structural biology, and 31 years in biocuration. While undertaking the annotation process, the team was split over two
67 different sites and time zones, and annotation was carried out in a fully remote setting.

68 Literature selection

69 The general workflow for literature selection is depicted in Figure 1. In the first step we retrieved all the PubMed (<https://pubmed.ncbi.nlm.nih.gov/>) IDs (PMIDs) for publications linked to a protein structure by querying PDBe's ORACLE
70 database on 29th September 2022. On that date, the PDB contained 196,012 PDB entries with 73,019 associated, unique
71 PMIDs.

72 LitSuggest¹¹ (<https://www.ncbi.nlm.nih.gov/research/litsuggest/>), an AI-driven web browser-based
73 trainable system that directly uses PMIDs, was used as a content filtering tool for assessing the abstract and title of our
74 short-listed publications. Using the list of PMIDs generated above, we created seven publication batches of 10,000 IDs each
75 (the positive samples) and batch 8, an exception, had only 3,019 IDs. All batches were matched with an equal number of
76 randomly picked PMIDs from the entire set in PubMed representing the negative samples. It has to be noted, that due to the
77 selection process, there is a small chance that the negative sets may contain some of the PMIDs from the positive batches. For
78 each batch, a model was trained using the corresponding titles and abstracts for the individual PMIDs in the batch. The trained
79 models were used to identify the relevance of newly added IDs to PubMed over several weeks.

80 The same exercise was repeated on the 23rd of January 2023, with 200,612 PDB entries having 74,253 unique PMIDs. The
81 additional PMIDs were added to batch 8 which now contained 4,253 IDs while preserving the original batches used in previous
82 training. Each of the eight trained models were presented with 7 batches, excluding the one that was used to train the model to
83 obtain relevance scores for individual PMIDs. A publication was deemed relevant when the predicted confidence score was
84 ≥ 0.8 across all seven cross-prediction models. This resulted in 63,795 (86%) PMIDs predicted as relevant. The prediction
85 statistics across the eight different models are given in Table 1.

87 To adhere to open data principles and to be able to annotate full-text articles, only open access publications with Pub-
88 MedCentral¹² (<https://www.ncbi.nlm.nih.gov/pmc/>) IDs (PMCs) were identified using the EuropePMC's¹³
89 (<https://europepmc.org/>) article API and included in further works. This further reduced the number of publications
90 included in the study to 14,390, 19% of the initial starting set of 74,253.

91 Lastly, a number of documents were further rejected during the annotation stage due to a primary focus other than a protein
92 structure, often covering drug and fragment screening campaigns or nucleic acid structures.

93 **The annotation tool and schema**

94 For our text annotation project, a number of free and paid-for annotation tools were evaluated regarding the following
95 features: compatibility with PubMed and PubMedCentral, possibility for project management, multi-user co-annotation
96 option, integration of ontologies, open source distribution, web browser based, ease of use, and available documentation. The
97 annotation tool of choice for our project was TeamTat¹⁴ (<https://www.teamtat.org/>). TeamTat is a free tool and
98 its developers focused on biomedical literature and compatibility with PubMed and PubMedCentral, retrieving publication
99 abstract and metadata such as title, authors, journal and publication year for a given PMID. For open access publications with a
100 PMCID, TeamTat retrieved the entire publication from a BioC XML FTP-server¹⁵ (<https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/>). The BioC XML format¹⁶ (<https://bioc.sourceforge.net/>) was introduced by
101 the BioCreative Initiative¹⁷ as a way of making scientific publications interoperable. In the case of open access full-text
102 documents, these came with in-line figures, figure captions, tables and table captions.

103 TeamTat also allowed for project management with the project manager being able to upload/retrieve the relevant literature,
104 assign publications to annotators and control the start and end of an annotation round. Entity types, relationship types and
105 ontology referencing were set up and updated by the project manager. TeamTat also supported versioning and after each
106 annotation round, merging statistics were calculated across the corpus as well as inter-annotator agreement and a new version
107 for the publication set was created. Documents could be exported at any point in the annotation process as either BioC XML or
108 BioC JSON. We opted for the BioC XML format, as it enclosed the plain paragraph text and its identified annotations under the
109 same XML tag (<passage>), which allowed for easy retrieval of individual sentences with their respective in-line annotations
110 for downstream transformer training.

111 TeamTat provides access to Medical Subject Headings (MeSH)¹⁸ and the Gene Ontology (GO)^{19, 20} through hard-coded
112 links. Additional ontologies relevant for our project were Sequence Ontology (SO)²¹, Chemical Entities of Biological Interest
113 (ChEBI)²², Gene²³ and PRotein Ontology (PR/PRO)²⁴. For each ontology a short-hand name similar to, e.g. "MESH:" for
114 MeSH, was created and served as a prefix to link an entity type to an ontology. A "DUMMY:" short-hand name was used to
115 collect terms that were not found in any of the other ontologies. Although we linked the different entity types to ontologies,
116 controlled vocabularies and reference databases, we did not apply grounding of terms in the annotation process by linking text
117 spans to unique references.

118 The annotation handbook published by the TeamTat developers (Supplemental Materials of Islamaj et al.²⁵) was adapted
119 to suit our project requirements. The final detailed annotation schema can be found in Supplemental Material. The project
120 manager generated an initial set of annotated publications to define a set of entity types which formed the basis for developing
121 initial guidelines, which were revisited in the subsequent annotation rounds following discussions with the biocurators (see
122 below in [Manual annotation of initial set of publications](#)). The updates to guidelines included adding or removing entity
123 types or clarification on the guidelines. The guidelines continued to be adapted even after switching from fully manual
124 annotation with a team of biocurators to a semi-automated process using a trained model to accommodate the increasingly
125 diverse set of publications. All alterations were done after consultation with the volunteer team of biocurators either in form
126 of open discussion or polling. Focusing on structure and sequence features curated by the UniProt biocurators, we selected
127 entity types that captured details about a particular protein, its structural make-up down to residue level, interaction partners,
128 bound molecules, general properties of the protein, changes to its sequence, organism of origin, experimental methods and
129 evidence to support drawn conclusions. The final list of entity types later used in transformer training was: "bond_interaction",
130 "chemical", "complex_assembly", "evidence", "experimental_method", "gene", "mutant", "oligomeric_state", "protein",
131 "protein_state", "protein_type", "ptm", "residue_name", "residue_name_number", "residue_number", "residue_range", "site",
132 "species", "structure_element", "taxonomy_domain". The "Materials and Methods" and "References" sections were excluded
133 from the annotation process as little to no contextual, residue-level information was expected to be present in these sections.

134 We also developed a detailed user guide (see Supplemental Materials) on how to set up and operate TeamTat from a project
135 manager as well as biocurator perspective. This was used to support the biocurators after initial training when annotating
136 independently.

137 **Manual annotation of initial set of publications**

138 Initially, ten publications (PMC4784909, PMC4786784, PMC4792962, PMC4832331, PMC4833862, PMC4848090, PMC4850273,
139 PMC4850288, PMC4852598, PMC4887326) were chosen randomly from the filtered, open access list described in the [Litera-](#)

ture selection section above. Each biocurator was given two publications to manually annotate, based on the guidance from the example annotations and the handbook. A set of two-hour hackathons were organized weekly to annotate the assigned publications. In case the biocurators were not able to attend the hackathon, web-based access through a personalized web-link for the assigned documents was provided to annotate documents outside of the dedicated sessions. The project manager annotated those publications that could not be annotated by the biocurators in order to achieve double-annotation for each document. We acknowledge and are fully aware that the overrepresentation of the project manager annotated documents increased the likelihood of bias. However, even with the best annotation guidelines shaped by a team of expert annotators, assigning entity types to terms is a highly subjective process. A different team of experienced annotators may introduce a different set of biases, based on their training and understanding. The first round of independent annotation lasted approximately four months, after which the annotations across all ten publications were combined and annotation statistics were calculated within TeamTat.

To increase efficiency and accelerate the annotation process, the decision was made to switch from a fully manual to a semi-automated annotation process. The project manager was made responsible for cleaning and consolidating the annotations for the ten initial publications. Upon completion of this task, the cleaned publications were passed to the lead biocurator, who served as a proof-reader. In this capacity, the lead biocurator flagged annotations and entity types that were still ambiguous. In a number of discussions between the project manager and the lead biocurator those ambiguities were resolved and entity types and annotation guidelines were updated. A graphical illustration of the manual annotation workflow can be found in Figure 2. The project manager then applied a final pass of cleaning and consolidating across the ten initial publications before using the annotated text to train a named entity recognition system. This final, consolidated version was used as ground truth against which the annotation performance of each annotator could be measured.

161 **Annotation evaluation**

162 The quality of manual annotations created by the biocurators was judged using the built-in calculation procedures in TeamTat,
163 which follow a partial agreement model. The following six categories of annotation outcomes were determined by TeamTat:

- 164 • *FA* - Full Agree: same type, concept ID and text span
- 165 • *CA* - Concept Agree: same concept ID and text span, but different types
- 166 • *TA* - Type Agree: same type and text span, but different concept IDs
- 167 • *PA* - Partial Agree: same type and concept ID for overlapping text
- 168 • *DA* - Disagree: different types, different concept IDs for text spans
- 169 • *SN* - Single: text annotated by only one of the annotators

170 The full set of outcomes was only relevant for the initial manual annotation by the biocurators and during the cleaning and
171 remediation steps to create the training data for the initial model. As mentioned in [Annotation tool and schema](#), we did not use
172 concept IDs for grounding terms and only evaluated for prefix matches, which, as they were directly linked to an entity type,
173 always returned a perfect match.

174 In order to investigate whether there was any bias introduced into the annotations by individual biocurators we also applied
175 the SemEval procedure to the manually annotated publications, see [Annotation evaluation using SemEval procedure](#).

176 **Annotation processing for training and evaluation**

177 In order to train a transformer-based annotation algorithm and to be able to calculate annotation statistics to monitor the
178 performance of both the algorithm and the human annotators, the publication text and its in-line annotations needed to be
179 converted from BioC XML into the IOB (Inside Outside Beginning) format²⁶. For each document we iterated over the individual
180 paragraphs, split them into sentences and combined them with their respective annotations using the offset values available
181 in the BioC XML file. For the total list of isolated sentences, we then generated an index. Next, the isolated sentences were
182 converted into tab-separated TSV files. These TSV files were used to calculate various statistics, see in section [Annotation](#)
183 [evaluation using SemEval procedure](#). The index was then randomly split to create three smaller files holding train, test and
184 development sets, containing 70%, 15%, and 15% of sentences, respectively.

185 During the conversion process it was found that a number of open access documents retrieved from NCBI's FTP site
186 had line breaks introduced within a paragraph, often in figure or table captions. These line breaks resulted in shifts of the
187 paragraph offset by "+1", which introduced character position miss-matches for the corresponding annotations. Through
188 personal communication with the maintainers of the FTP site, it was found that the offset shift was likely a result of the
189 conversion process from a number of input file formats provided by publishers to BioC XML. Occasionally, we also found

190 identical sentences and annotations more than once. In this case, only the first occurrence would be included in the data.
191 Although all efforts were made to catch as many errors as possible, on average 21 annotations were lost in each batch, which
192 amounts to 0.2% as each batch had on average 10,037 annotations.

193 **Training a first named entity recognition system**

194 Using the TSV files from above, we trained a first model. The basic principles of our algorithm and training process are given
195 in Algorithm 1. The training routine described also provided the basis for the iterative training to build a semi-automatic
196 annotation system.

Algorithm 1: Iterative Deep Learning Model Training with Curators in Loop

Data: 10 curated articles by 7 curators in Team Tat

Result: Trained Deep Learning model D_n and Ground Truth (GT)

Input : $n = 1$

```
1 while Performance improvement <= 0.5% OR n = 10 do
2   // Step 1
3   Initialise Load 10 curated articles and GT;
4   Train  $D_{n=1}$  with the 10 curated articles and GT;
5   Evaluate curator performance and generate GT using maximum vote;
6   // Iteration n = 1
7   Perform inference using  $D_{n=1}$  on 10 unseen articles;
8   Convert inferences to the Team Tat format;
9   Send articles to a curator for validation and correction;
10  Update GT using curator feedback and resolve discrepancies through maximum vote;
11  Increment n = n + 1;
12  Train  $D_{n+1}$  with updated (n × 10) curated articles and GT;
13 end
```

197 Taking advantage of the rapid developments in natural language processing (NLP), we chose a starting model based
198 on Google's transformer²⁷. For our objective, NER, we looked at BERT (Bidirectional Encoder Representations from
199 Transformers)-based models such as BioBERT²⁸, PubmedBERT²⁹, and BioFormer³⁰. We employed a pre-trained transformer
200 model from Hugging Face (<https://huggingface.co>), namely *microsoft/BioMedNLP-PubMedBERT-base-uncased-
201 abstract-fulltext*. Fine-tuning was conducted for 3 epochs, initially, using the carefully selected hyperparameters listed in Table
202 2. Optimizing the hyperparameters resulted in an improved initial model, v1.2, which was used to annotate a new batch of
203 publications. We also reduced the number of entity types from 23 to 19, as we found during the data preparation step that
204 some entity types had too few samples to allow for a meaningful split into train, test and evaluation set and had a negligible
205 contribution to training.

206 **Consecutive rounds of semi-automatic annotation and NER training**

207 To develop a robust algorithm, a diverse corpus, larger than the initial ten publications, was needed. Therefore, a human-in-the
208 loop approach combined with a named entity recognition system (see [Training a first named entity recognition system](#)) was
209 used, to iteratively increase the number of annotated publications in the corpus.

210 In each iteration, a new batch of ten publications randomly selected from the open access list was presented to the current
211 best model to identify text spans and annotate them with their entity types. The returned predictions were in BioC XML format
212 which allowed for visual inspection in the annotation tool TeamTat.

213 At the end of each prediction round, the project manager inspected each of the ten publications in the batch, and fixed any
214 errors in the annotations produced by the NER model. This curation process did not only look at the predicted annotations but
215 rather the pre-annotated spans served as a guide for the annotator, who was still required to read the full text and add missing
216 annotations. Such an approach of post-prediction curation has been implemented as a standard tool in the NLP suite "prodigy"
217 version 3 <https://prodi.gy/> in the function "ner.correct" ("ner.make-gold" in version 2). A similar approach was also
218 used by Gnehm et al.³¹. Any annotations predicted in the "Methods" and "References" sections were removed (see annotation
219 schema in the Supplemental Materials for details). The curated annotations were stored in BioC XML to be later combined
220 with other batches and converted to the IOB format for a new round of model training or being used as ground truth for model
221 performance monitoring. The applied workflow is presented in Figure 3.

222 For entity types that repeatedly produced large numbers of false-positives or false-negatives, i.e. were not correctly identified
223 by the predictor and required manual curation, anonymous biocurator polling was used to improve the annotation process. Here,

examples of ambiguously labeled terms were given to all biocurators and they were asked to assign entity types. A majority vote across the responses determined the entity type and the annotations in the publications were updated accordingly. For example, it was not clear what should be labeled as entity type "*mutant*". After polling, point mutations at specific sequence positions and deletions/insertions of sequence ranges or whole domains and proteins were included.

The decision was made that no new publications would be added for training once the NER model would not improve by more than 0.5% for overall values for F1-measure, precision and recall across all entity types or reaching 100 publications, whichever came first.

As a result of changes to the annotation schema in terms of entity types (described in section [The annotation tool and schema](#)) and adding new publications over time, splitting into train, test and validation sets was carried out anew for every new model training. In order to be able to judge and compare the performance of models v2.1 and v3.1, we therefore employed an additional set of ten publications which had not been used for any training, testing or validation and provided a completely independent test set.

It should also be noted that for inference on a new document, we supplied the publication text split into paragraphs rather than sentences as was used during training. This was aimed at the transformer model's ability to contextualize named entities and as was shown by Luoma and Pyysalo³² and Wang *et al.*³³ this was expected to improve the model's performance.

Annotation evaluation using SemEval procedure

To monitor and evaluate the performance of the trained predictor, we followed the published assessment process for SemEval³⁴. Each predicted annotation was assessed whether it had a matching annotation in the ground truth using the following five categories:

- *Correct* - full agreement between predicted annotation and ground truth annotation in text span and entity type
- *Incorrect* - disagreement between predicted annotation and ground truth annotation in text span and entity type
- *Partial* - text span overlaps in predicted annotation and ground truth annotation but the entity type may differ
- *Missing* - annotation is only found in the ground truth but not in the predicted annotations
- *Spurious* - annotation is only found in the predicted annotations but not in the ground truth

SemEval then evaluated a found match whether it belonged to one of four different classes of matches:

- *Strict* - only evaluate annotations with exact text span boundaries and exact entity types between predicted and ground truth annotations
- *Exact* - allow annotations to have exact text span boundaries with disagreement in entity type between predicted and ground truth annotations
- *Partial* - allow annotations to have partially overlapping text span boundaries with disagreement in entity type between predicted and ground truth annotations
- *Type* - allow annotations that have some form of agreement between predicted and ground truth annotations

For each class of match the precision, recall and F1-measure were determined. The statistics were calculated for annotations in the selected sections title, abstract, introduction, results, discussion, tables, as well as table and figure captions. In order to apply the SemEval procedure to the annotations, the text and in-line annotations had to be converted from BioC XML to the IOB format as described above in [Annotation processing for training and evaluation](#).

Please note that the evaluation was done by comparing the predictions to the ground truth. However, we used all the predictions produced by the different models on the full-text BioC XML rather than predicting only on the sentences included in the ground truth. As a consequence, the models produced predictions that are not found in the ground truth. Those additional sentences were given the O(outside) label and appended to the ground truth. During the evaluation process these annotations were classed as "*spurious*".

For the batches used during consecutive training described above in [Consecutive rounds of semi-automatic annotation and NER training](#), we performed the evaluation across the entire batch. The independent test set, batch 5, for the comparison of autoannotator versions v2.1 and v3.1, additionally underwent a per-document evaluation.

268 Results

269 Manual annotation statistics and annotator performance

270 After merging the annotations from the initial manual annotation round, it became apparent that only seven out of ten
271 publications had been double-annotated, namely PMC4784909, PMC4786784, PMC4792962, PMC4832331, PMC4833862,
272 PMC4850273, and PMC4852598. The remaining three publications had very few double or only single annotations. We
273 therefore calculated inter-annotator agreement using the TeamTat analysis tool for both options, seven documents with double
274 annotations and ten documents with three only having partial annotations. The statistics are provided in Table 3. Excluding
275 the partially annotated publications caused a drop of 1.8% for the full-agreement statistics from 79% for ten documents to
276 77.2% for seven. Although gold-standard double-annotation could not be achieved for all documents, those that were fully
277 annotated had high agreement between the annotators and were expected to provide good quality training data for bootstrapping
278 a transformer model.

279 After further cleaning and term disambiguation discussed between the project manager and the lead biocurator, a consolidated
280 set of annotated documents was created, which served as a ground truth. Table 4 gives the total annotation counts for each
281 publication in the ground truth. We manually annotated 10,451 terms, across all publications ranging from 715 to 1,549 and a
282 mean of 1,045 annotations per article. Those annotations represented 2,988 unique terms across 19 different entity types. Table
283 5 contains the raw counts as well as the number of unique terms found for each entity type. We found that the top-10 entity
284 types for total number of counts were also those with the highest count for unique terms. Those specific entity types sorted
285 by total count in descending order are: "protein", "structure_element", "protein_state", "chemical", "residue_name_number",
286 "protein_type", "evidence", "mutant", "experimental_method" and "site". They represent the most relevant key terms to describe
287 residue-level details in a protein structure and do not only appear with high frequency in the training data, but also provide the
288 algorithm with a diverse set of terms to learn and generalize from.

289 This ground truth was also used to assess each biocurator's performance in the initial annotation round using the SemEval
290 evaluation process. The evaluation was done for all the publications for an annotator, regardless of whether they had been
291 fully or partially annotated. In Table 6, the precision achieved by each annotator is given. Applying a "partial agreement"
292 evaluation strategy, we found that all annotators reach a score of 0.79 and above, which underlines the fact that all biocurators
293 have a strong biochemistry and structural biology background and generally look for the same terms within a publication and
294 find most of the occurrences in the ground truth. The scores for the recall, Table 7, are much lower which indicates that all
295 biocurators are not consistent in annotating the same terms, i.e. within the same document a term may have been assigned
296 different entity types, if a term spans multiple words there may be different span boundaries for the same term or a term may
297 have been missed. The F1-measure for each annotator in Table 8 again supports the finding that generally all annotators share
298 the same understanding for the key terms in the documents but differ in their assignment to a specific entity type and where
299 the span boundaries should be placed. It is worth noting that Annotator0 achieved the highest scores for precision, recall and
300 F1-measure for all four evaluation options. Such dominance from one annotator increases the risk of bias. However, considering
301 that the agreement between different annotators, given in Table 3, is > 75%, i.e. annotators agree at least partially on more than
302 75% of annotations, the majority of annotations will not have been biased by Annotator0's performance.

303 Overall, the annotator statistics underline the high level of expert knowledge of the biocurators and that, although gold
304 standard double annotation was not achieved across all documents and some bias from one annotator may have been introduced,
305 the identified annotations are of high quality.

306 Initial model trained on ten manually annotated publications

307 The initial model (v1.2) was trained on ten manually annotated publications described in [Manual annotation of initial set of](#)
308 [publications](#). Its performance results on the development set are plotted in Figure 4. Throughout the training process, the
309 model's performance consistently improved, as indicated by decreasing losses and increasing precision, recall, F1-measure, and
310 accuracy. However, the increasing gap between training and validation loss indicated overfitting. With the small sample size of
311 10,451 annotations used to develop this first model, overfitting was not surprising, but learning was clearly observed in the loss
312 curves. Consequently, we increased the sample size and explored some hyperparameter settings and iteratively improved the
313 model (see below [Consecutive model training](#)).

314 Consecutive model training

315 In Table 9 we give the overall performance statistics (precision, recall, F1-measure and accuracy) for the respective test sets
316 of the different consecutive models averaged across all entity types. Although there is only minimal change in the overall
317 statistics for the different models, plotting the training and validation loss for each training round and comparing the different
318 models revealed that those trained on the larger corpus are less prone to overfitting (see Figure 5). With each additional batch
319 of annotations added, the training loss for the corresponding model started at a higher point, as would be expected from a
320 larger, more diverse corpus, requiring a model to work harder to learn commonalities in the data. This followed a sharp drop,

321 which lasted for the first three epochs and, finally, all models converged to a similar value after seven epochs. Therefore, all
322 models showed clear signs of learning. To assess how the models performed on data not used for training, we looked at the
323 validation loss determined for each respective validation set. For models v1.2 and v1.4 the validation loss never dropped, but
324 instead showed a continuous increase, which is a clear sign of overfitting (see Figure 5). A model that is prone to overfitting is
325 undesirable, as this would lead to unreliable predictions. After adding additional annotations for models v2.1 and v3.1, we
326 observed a similar drop in the validation loss as we found in the training loss (Figure 5) for the first three epochs followed by a
327 slow increase thereafter. Such behavior indicated that the models trained on a larger corpus exhibited reduced overfitting and
328 that predictions for new, unseen data could be expected to be of similar quality as for the training data.

329 To further judge the performance of the different models, we also monitored the changes in precision, recall and F1-measure
330 for the individual entity types in the respective test sets, Table 10, Table 11, Table 12, respectively. Generally, even the model
331 trained on only ten publications (v1.2) already did reasonably well across the different entity types with the vast majority of
332 scores for precision, recall and F1-measure being ≥ 0.85 . However, as observed above in the general training and validation
333 loss plots, model v1.2, and to a lesser degree v1.4, was prone to overfitting, which is supported by achieving scores of 1.00
334 for a number of entity types for precision, recall and F1-measure. We also found that not all entity types were predicted with
335 similar confidence by the different models. Such behavior reflects the fact that the training data is highly imbalanced for the
336 different entity types, but also that the annotation schema was updated between models. Therefore, a direct comparison of the
337 different models has to be done with caution.

338 The iterative training approach meant that for every batch that was automatically annotated by one of the models we also
339 had a set of ground truth annotations from curation, which served as training data for the next generation model, but could
340 also be used for evaluation of the predecessor. Following the SemEval protocol, we evaluated the performance of the different
341 models. The statistics for precision, recall and F1-measure for each model and its respective ground truth batch are given in
342 tables Table 13, Table 14 and Table 15. We found that with each training iteration using an increased and more diverse set of
343 annotations, the scores for precision, recall and F1-measure improved, reflecting a model's ability to make predictions closer to
344 the respective ground truth.

345 Finally, we compared models v2.1 and v3.1 on an independent test set of ten publications to serve as ground truth for final
346 model selection. The prediction statistics for the two models on the individual documents in the independent test set are given
347 in Table 16 for precision, Table 17 for recall and F1-measure in Table 18. The statistics were calculated following the SemEval
348 process. We found that both models performed well on the independent test set with some publications proving easier to predict
349 on, i.e. models achieving higher scores. As model v2.1 achieved higher scores for the different evaluation types across all
350 documents, we chose it as the current best model.

351 Conclusion

352 Manually annotating a domain specific corpus with a team of experts is usually a time and cost intensive process. Our presented
353 approach shows that a small corpus of ten publications annotated by a team of highly qualified experts is sufficient to bootstrap
354 an initial model, v1.2, for a human-in-the-loop application. In consecutive rounds, we improved on this initial model to yield
355 our best performing version, v2.1. Our best model is highly specific in annotating biomedical literature concerned with protein
356 structures and should be used to identify key-terms describing the 3-dimensional composition of proteins. We made all our data
357 and models open access and they are available for download or programmatic access from <https://huggingface.co/>.
358 The code provided alongside can be used as a wrapper to run and evaluate the models locally or serve as a start to develop a
359 similar workflow in any other field.

360 Data availability

361 The various data files have been made available at: <https://huggingface.co/PDBEurope>

362 1. Stand-alone curator annotations

363 • CSV

364 • JSON

365 • Inside-outside-beginning (IOB)

366 2. Full-text XML files (without annotations)

367 3. Full-text BioC with annotations in XML format

368 4. Full-text BioC with annotations in JSON format

369 The annotations and documents are made available in a number of formats. We provide the annotations and publications
370 grouped as they were used to train the models v1.2, v1.4, v2.1 and v3.1. The annotations for batch 5, used to compare models
371 v2.1 and v3.1 are provided separately. The plain, full-text XML files of all documents are provided in BioC without annotations.
372 The annotations themselves are provided in-line in BioC for each publication, either as XML or as JavaScript Object Notation
373 (JSON) format. Additionally, the annotations are available as standalone comma-separated values (CSV) and JSON. In these
374 standalone CSV and JSON files, an annotation is described by its unique “*id*”, “*character start*”, “*character end*” to locate the
375 starting and ending character positions within a sentence, “*span*” representing the covered text span and “*entity type*” giving
376 its entity type. To identify from which document an annotation was retrieved, we also use the PMCID of the corresponding
377 publication as “*document id*” in the standalone annotation files. We also provide the annotations and their respective sentences
378 in the IOB format. The IOB files provide sentences with IOB tags and follow the CoNLL NER corpus standards³⁴. The datasets
379 to develop the four different models and the independent test set are available from Huggingface:

380 • https://huggingface.co/datasets/PDBEurope/protein_structure_NER_model_v1.2
381 • https://huggingface.co/datasets/PDBEurope/protein_structure_NER_model_v1.4
382 • https://huggingface.co/datasets/PDBEurope/protein_structure_NER_model_v2.1
383 • https://huggingface.co/datasets/PDBEurope/protein_structure_NER_model_v3.1
384 • https://huggingface.co/datasets/PDBEurope/protein_structure_NER_independent_val_set

386 For example, the data folder for model v1.2 contains the following subfolders and files:

387 • annotated_BioC_JSON: one JSON file for each annotated publication in BioC; < *PMCID* >_ann.json
388 • annotated_BioC_XML: one XML file for each annotated publication in BioC; < *PMCID* >_ann.xml
389 • annotation_CSV: one CSV file for each publication, annotations only; < *PMCID* >.csv
390 • annotation_IOB: all annotated sentences in IOB format and training, testing, development subsets; all.tsv, train.tsv,
391 test.tsv, dev.tsv
392 • annotation_JSON: single JSON file containing all annotations from all documents
393 • raw_BioC_XML: one XML file for each NOT annotated publication in BioC; < *PMCID* >_raw.xml

394 The four trained models, v1.2, v1.4, v2.1 and v3.1 are available from Huggingface:

395 • <https://huggingface.co/PDBEurope/BiomedNLP-PubMedBERT-ProteinStructure-NER-v1.2>
396 • <https://huggingface.co/PDBEurope/BiomedNLP-PubMedBERT-ProteinStructure-NER-v1.4>
397 • <https://huggingface.co/PDBEurope/BiomedNLP-PubMedBERT-ProteinStructure-NER-v2.1>
398 • <https://huggingface.co/PDBEurope/BiomedNLP-PubMedBERT-ProteinStructure-NER-v3.1>

399 **Code availability**

400 The code is available at the repository https://github.com/PDBEurope/ner_for_protein_structures.
401 Detailed documentation and how to install the tools can be found at <https://ner-for-protein-structures.readthedocs.io/en/latest/>. Scripts include those used for cleaning and formatting of the annotations from annotation tool TeamTat and how to generate the datasets in the IOB format for input to deep learning algorithms. Additional scripts
403 allow the calculation of model performance and prediction outcome following the SemEval process.

405 References

- 406 1. wwPDB consortium. Protein data bank: the single global archive for 3d macromolecular structure data. *Nucleic Acids Res.* **47**, D520–D528, <https://doi.org/10.1093/nar/gky949> (2019).
- 407 2. Armstrong, D. e. a. Pdbe: improved findability of macromolecular structure data in the pdb. *Nucleic Acids Res.* **48**, D335–D343, <https://doi.org/10.1093/nar/gkz990> (2020).
- 408 3. Choudhary, P. e. a. Unified access to up-to-date residue-level annotations from uniprotkb and other biological databases for 409 pdb data. *Sci Data* **10**, <https://doi.org/10.1038/s41597-023-02101-6> (2023).
- 410 4. Consortium, T. U. Uniprot: the universal protein knowledgebase in 2023. *Nucleic Acids Res.* **51**, D523–D531, <https://doi.org/10.1093/nar/gkac1052> (2023).
- 411 5. Jumper, J. e. a. Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589, <https://doi.org/10.1038/s41586-021-03819-2> (2021).
- 412 6. Baek, M. e. a. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 413 871–876, <https://doi.org/10.1126/science.abj8754> (2021).
- 414 7. Varadi, M. e. a. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence 415 space with high-accuracy models. *Nucleic Acids Res.* **50**, D439–D444, <https://doi.org/10.1093/nar/gkab1061> (2022).
- 416 8. Schwede, T. e. a. Outcome of a workshop on applications of protein models in biomedical research. *Structure* **17**, 151–159, 417 <https://doi.org/10.1016/j.str.2008.12.014> (2009).
- 418 9. Varadi, M. e. a. 3d-beacons: decreasing the gap between protein sequences and structures through a federated network of 419 protein structure data resources. *GigaScience* **11**, <https://doi.org/10.1093/gigascience/giac118> (2022).
- 420 10. Wilkinson, M. e. a. The fair guiding principles for scientific data management and stewardship. *Sci. Data* **3**, <https://doi.org/10.1038/sdata.2016.18> (2016).
- 421 11. Allot, A. e. a. Litsuggest: a web-based system for literature recommendation and curation using machine learning. *Nucleic 422 Acids Res.* **49**, W352–W358, <https://doi.org/10.1093/nar/gkab326> (2021).
- 423 12. Roberts, R. Pubmed central: The genbank of the published literature. *Proc. Natl Acad. Sci. USA* **98**, 381–382, <https://doi.org/10.1073/pnas.98.2.381> (2001).
- 424 13. Consortium, E. P. Europe pmc: a full-text literature database for the life sciences and platform for innovation. *Nucleic 425 Acids Res.* **43**, D1042–D1048, <https://doi.org/10.1093/nar/gku1061> (2015).
- 426 14. Islamaj, R. e. a. Teamtat: a collaborative text annotation tool. *Nucleic Acids Res.* **48**, W5–W11, <https://doi.org/10.1093/nar/gkaa333> (2020).
- 427 15. Comeau, D. e. a. Pmc text mining subset in bioc: about 3 million full text articles and growing. *Bioinformatics* **35**, 428 3533–3535, <https://doi.org/10.1093/bioinformatics/btz070> (2019).
- 429 16. Comeau, D. e. a. Bioc: a minimalist approach to interoperability for biomedical text processing. *Database (Oxford)* **2013**, bat064, <https://doi.org/10.1093/database/bat064> (2013).
- 430 17. Hirschman, L. e. a. Overview of biocreative: critical assessment of information extraction for biology. *BMC Bioinforma.* **6**, 431 S1, <https://doi.org/10.1186/1471-2105-6-S1-S1> (2005).
- 432 18. Rogers, F. Medical subject headings. *Bull. Med. Libr. Assoc.* **51**, 114–6 (1963). <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC197951/pdf/mlab00186-0145.pdf>.
- 433 19. Ashburner, M. e. a. Gene ontology: tool for the unification of biology. *Nat. Genet.* **25**, 25–9, <https://doi.org/10.1038/75556> (2000).
- 434 20. et al., T. G. O. C. The gene ontology knowledgebase in 2023. *Genetics* **224**, iyad031, <https://doi.org/10.1093/genetics/iyad031> (2023).
- 435 21. Eilbeck, K. e. a. The sequence ontology: A tool for the unification of genome annotations. *Genome Biol.* **6**, R44, 436 <https://doi.org/10.1186/gb-2005-6-5-r44> (2005).
- 437 22. Hastings, J. e. a. Chebi in 2016: Improved services and an expanding collection of metabolites. *Nucleic Acids Res.* **44**, 438 D1214–D1219, <https://doi.org/10.1093/nar/gkv1031> (2016).
- 439 23. Maglott, D. e. a. Entrez gene: gene-centered information at ncbi. *Nucleic Acids Res.* **39**, D52–D57, <https://doi.org/10.1093/nar/gkq1237> (2011).

452 24. Natale, D. e. a. Protein ontology (pro): enhancing and scaling up the representation of protein entities. *Nucleic Acids Res.*
453 45, D339–D346, <https://doi.org/10.1093/nar/gkw1075> (2017).

454 25. Islamaj, R. e. a. Nlm-chem-bc7: manually annotated full-text resources for chemical entity annotation and indexing in
455 biomedical articles. *Database* 2022, <https://doi.org/10.1093/database/baac102> (2022).

456 26. Ramshaw, M., L.A. & Marcus. Text chunking using transformation-based learning. In *Third Workshop on Very Large
457 Corpora* (1995). <https://aclanthology.org/W95-0107.pdf>.

458 27. Vaswani, A. e. a. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information
459 Processing Systems*, NIPS'17, 6000–6010 (Curran Associates Inc., Red Hook, NY, USA, 2017). [https://proceedings.
neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf](https://proceedings.
460 neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fb053c1c4a845aa-Paper.pdf).

461 28. Lee, J. e. a. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*
462 36, 1234–1240, <https://doi.org/10.1093/bioinformatics/btz682> (2019).

463 29. Gu, Y. e. a. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput.
464 Healthc.* 3, 1–23, <https://doi.org/10.1145/3458754> (2021).

465 30. Fang, L. e. a. Bioformer: an efficient transformer language model for biomedical text mining, [https://doi.org/10.48550/
466 arXiv.2302.01588](https://doi.org/10.48550/arXiv.2302.01588) (2023).

467 31. Gnehm, A.-S. e. a. Evaluation of transfer learning and domain adaptation for analyzing german-speaking job advertisements.
468 In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 3892–3901 (European Language
469 Resources Association, Marseille, France, 2022). <https://aclanthology.org/2022.lrec-1.414.pdf>.

470 32. Luoma, S., J. & Pyysalo. Exploring cross-sentence contexts for named entity recognition with BERT. In *Proceedings of the
471 28th International Conference on Computational Linguistics*, 904–914, <https://doi.org/10.18653/v1/2020.coling-main.78>
472 (International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020).

473 33. Wang, X. e. a. Improving named entity recognition by external context retrieving and cooperative learning. In *Proceedings
474 of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference
475 on Natural Language Processing (Volume 1: Long Papers)*, 1800–1812, <https://doi.org/10.18653/v1/2021.acl-long.142>
476 (Association for Computational Linguistics, 2021).

477 34. Segura-Bedmar, I. e. a. Semeval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (ddiextraction
478 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh
479 International Workshop on Semantic Evaluation (SemEval 2013)*, 341–350 (Association for Computational Linguistics,
480 2013). <https://aclanthology.org/S13-2056.pdf>.

481 Acknowledgements

482 We would like to thank Rezarta Islamaj Doğan (National Center for Biotechnology Information, National Library of Medicine,
483 Bethesda, MD 20894, USA) for her patience to explain aspects of the annotation tool TeamTat to M.V. and give some highly
484 valued tips towards the design of the annotation schema. Funding was provided to M.V. as ARISE Fellowship from the
485 European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No
486 945405.

487 Author contributions statement

488 M.V. identified relevant literature, designed and developed annotation guidelines and handbooks, conceived the experiment(s),
489 developed all the scripts for data preparation and formatting, trained the NER models, analysed the results, generated all the
490 figures and wrote the manuscript. S.T. provided guidance, starting code base and advice for training the NER models and
491 conducting SemEval analysis and contributed to writing the manuscript. D.H. took part in the annotation of the ten initial
492 documents, helped shape the annotation guidelines. D.A., R.G., D.G., M.Q.L.A. and G.E. took part in the annotation of the ten
493 initial documents and reviewed the manuscript. S.V. and D.H. reviewed the manuscript.

494 Competing interests

495 The authors have declared no competing interests.

496 **Figures & Tables**

batch	number of positive PubMed IDs	percent of total
1-10,000	8,682	87%
10,001-20,000	8,626	87%
20,001-30,000	8,612	86%
30,001-40,000	8,543	85%
40,001-50,000	8,601	86%
50,001-60,000	8,552	86%
60,001-70,000	8,543	85%
70,001-74,253	3,580	84%

Table 1. Proportion of publications selected from batches of 10,000 using seven independent LitSuggest models and a confidence score ≥ 0.8

Hyperparameter	Value	Description
Evaluation Strategy	Epoch-based evaluation	The approach used to evaluate the model during training.
Learning Rate	2e-5	The step size at which the model's weights are updated during training.
Per-Device Training Batch Size	2 (v1.2, v1.4); 5 (v2.1); 10 (v3.1)	The number of samples used for each gradient update during training on each device.
Per-Device Evaluation Batch Size	2 (v1.2, v1.4); 5 (v2.1); 10 (v3.1)	The number of samples used for evaluation on each device.
Number of Training Epochs	10	The number of times the entire training dataset is passed through the model during training.
Weight Decay	0.01	A regularization term used to prevent overfitting by penalizing large weights in the model.

Table 2. Hyperparameters for the Named Entity Recognition Model - all models

# docs	Total	FA ¹	CA ²	TA ³	PA ⁴	DA ⁵	SN ⁶
10	11,388	8,997 (79.0 %)	191 (1.68 %)	334 (2.93 %)	283 (2.49 %)	1,388 (12.19 %)	195 (1.71 %)
7	7,965	6,146 (77.2 %)	191 (2.4 %)	46 (0.58 %)	271 (3.4 %)	1,311 (16.46 %)	0 (0.0 %)

Table 3. Inter-annotator agreement between the biocurators for the initial ten publications after the first independent annotation round

¹FA - Full Agree: same type, concept ID and text span

²CA - Concept Agree: same concept ID and text span, but different types

³TA - Type Agree: same type and text span, but different concept IDs

⁴PA - Partial Agree: same type and concept ID for overlapping text

⁵DA - Disagree: different types, concept IDs or text spans

⁶SN - Single: text annotated by only some of annotators

Document ID	annotation count
PMC4784909	865
PMC4786784	1,549
PMC4792962	1,268
PMC4832331	739
PMC4833862	1,044
PMC4848090	987
PMC4850273	1,121
PMC4850288	716
PMC4852598	1,229
PMC4887326	933
Sum	10,451

Table 4. Total number of annotation count for each of the initial ten manually annotated publications, after curation and consolidation

Entity type	Total count	Unique mentions
Chemical	1,030	260
Complex_assembly	289	81
Evidence	645	155
Experimental_method	570	302
Gene	154	43
Mutant	614	188
Oligomeric_state	151	23
Protein	1,457	122
Protein_state	1,093	303
Protein_type	756	267
PTM	134	40
Residue_name	139	49
Residue_name_number	795	281
Residue_number	48	20
Residue_range	90	85
Site	519	209
Species	205	60
Structure_element	1,448	453
Taxonomy_domain	314	47
Sum	10,451	2,988

Table 5. Total and unique annotation count for the different entity types in the initial ten manually annotated publications, after curation and consolidation

annotator	strict	exact	partial	type	# documents
annotator0	0.82	0.88	0.92	0.87	8
annotator1	0.70	0.80	0.87	0.74	2
annotator2	0.52	0.80	0.88	0.59	2
annotator3	0.55	0.65	0.79	0.69	2
annotator4	0.49	0.71	0.85	0.72	2
annotator5	0.49	0.86	0.92	0.53	2
annotator6	0.78	0.90	0.94	0.82	2

Table 6. Precision for manual annotation compared to ground truth for each annotator using SemEval evaluation

annotator	strict	exact	partial	type	# documents
annotator0	0.64	0.69	0.72	0.68	8
annotator1	0.39	0.44	0.49	0.41	2
annotator2	0.37	0.58	0.63	0.42	2
annotator3	0.43	0.50	0.62	0.61	2
annotator4	0.05	0.08	0.09	0.08	2
annotator5	0.14	0.24	0.25	0.14	2
annotator6	0.20	0.23	0.24	0.21	2

Table 7. Recall for manual annotation compared to ground truth for each annotator using SemEval evaluation

annotator	strict	exact	partial	type	# documents
annotator0	0.72	0.77	0.81	0.76	8
annotator1	0.50	0.57	0.62	0.53	2
annotator2	0.43	0.67	0.74	0.42	2
annotator3	0.48	0.57	0.69	0.61	2
annotator4	0.10	0.14	0.17	0.15	2
annotator5	0.21	0.37	0.40	0.23	2
annotator6	0.32	0.36	0.38	0.33	2

Table 8. F1-measure for manual annotation compared to ground truth for each annotator using SemEval evaluation

model	Overall precision	Overall recall	Overall F1-measure	Overall accuracy
v1.2	0.87	0.89	0.88	0.95
v1.4	0.90	0.92	0.91	0.96
v2.1	0.90	0.92	0.91	0.96
v3.1	0.91	0.92	0.91	0.96

Table 9. Overall training statistics for consecutive models

Entity	v1.2	v1.4	v2.1	v3.1
Bond Interaction	-	-	0.93	0.82
Chemical	0.84	0.90	0.89	0.92
Complex Assembly	0.85	0.88	0.91	0.89
Evidence	0.74	0.86	0.84	0.89
Experimental Method	0.77	0.73	0.85	0.80
Gene	0.86	0.89	0.79	0.79
Mutant	0.83	0.93	0.91	0.92
Oligomeric State	0.94	0.88	0.93	0.96
Protein	0.91	0.97	0.94	0.96
Protein State	0.80	0.78	0.83	0.86
Protein Type	0.85	0.84	0.85	0.85
PTM	0.88	0.64	0.70	0.85
Residue Name	0.86	0.97	0.92	0.74
Residue Name Number	0.99	0.98	0.95	0.96
Residue Number	1.00	1.00	0.80	0.70
Residue Range	1.00	0.86	0.81	0.89
Site	0.83	0.83	0.85	0.88
Species	0.96	0.97	0.94	0.95
Structure Element	0.88	0.91	0.91	0.91
Taxonomy Domain	0.95	0.97	0.99	0.98

Table 10. Precision for the different models for the different entity types on the test set

Entity	v1.2	v1.4	v2.1	v3.1
Bond Interaction	-	-	0.88	0.91
Chemical	0.90	0.93	0.91	0.91
Complex Assembly	0.76	0.91	0.93	0.90
Evidence	0.76	0.89	0.88	0.88
Experimental Method	0.75	0.76	0.85	0.82
Gene	0.92	0.86	0.86	0.65
Mutant	0.92	0.95	0.97	0.94
Oligomeric State	1.00	1.00	0.99	1.00
Protein	0.93	0.97	0.97	0.96
Protein State	0.83	0.85	0.88	0.88
Protein Type	0.84	0.90	0.85	0.88
PTM	0.76	0.81	0.70	0.79
Residue Name	0.95	0.92	0.97	0.96
Residue Name Number	0.99	0.99	0.96	0.98
Residue Number	1.00	0.93	0.97	0.73
Residue Range	0.80	0.91	0.70	0.86
Site	0.82	0.86	0.87	0.90
Species	0.98	1.00	0.96	0.95
Structure Element	0.86	0.92	0.92	0.92
Taxonomy Domain	0.97	0.96	0.98	0.98

Table 11. Recall for the different models for the different entity types on the test set

Entity	v1.2	v1.4	v2.1	v3.1
Bond Interaction	-	-	0.90	0.86
Chemical	0.87	0.92	0.90	0.92
Complex Assembly	0.80	0.89	0.92	0.90
Evidence	0.75	0.88	0.86	0.89
Experimental Method	0.76	0.75	0.85	0.81
Gene	0.89	0.88	0.82	0.71
Mutant	0.88	0.94	0.94	0.93
Oligomeric State	0.97	0.93	0.96	0.98
Protein	0.92	0.97	0.95	0.96
Protein State	0.81	0.81	0.85	0.87
Protein Type	0.84	0.87	0.85	0.87
PTM	0.81	0.71	0.70	0.82
Residue Name	0.91	0.94	0.94	0.84
Residue Name Number	0.99	0.99	0.96	0.97
Residue Number	1.00	0.96	0.88	0.71
Residue Range	0.89	0.89	0.75	0.87
Site	0.82	0.85	0.86	0.89
Species	0.97	0.98	0.95	0.95
Structure Element	0.87	0.91	0.92	0.91
Taxonomy Domain	0.96	0.97	0.98	0.98

Table 12. F1-measure for the different models for the different entity types on the test set

model	data batch	strict	exact	partial	type
v1.2	batch 2	0.66	0.75	0.83	0.75
v1.4	batch 3	0.69	0.79	0.85	0.75
v2.1	batch 4	0.75	0.83	0.88	0.82
v2.1	batch 5	0.78	0.85	0.91	0.85
v3.1	batch 5	0.77	0.84	0.90	0.86

Table 13. Precision for models and their respective publication batches compared to ground truth for each batch using SemEval evaluation

model	data batch	strict	exact	partial	type
v1.2	batch 2	0.67	0.77	0.85	0.76
v1.4	batch 3	0.71	0.80	0.87	0.77
v2.1	batch 4	0.75	0.83	0.89	0.83
v2.1	batch 5	0.75	0.82	0.87	0.81
v3.1	batch 5	0.75	0.79	0.86	0.81

Table 14. Recall for models and their respective publication batches compared to ground truth for each batch using SemEval evaluation

model	data batch	strict	exact	partial	type
v1.2	batch 2	0.67	0.76	0.84	0.76
v1.4	batch 3	0.70	0.79	0.86	0.76
v2.1	batch 4	0.75	0.83	0.89	0.82
v2.1	batch 5	0.76	0.83	0.89	0.83
v3.1	batch 5	0.75	0.81	0.88	0.83

Table 15. F1-measure for models and their respective publication batches compared to ground truth for each batch using SemEval evaluation

Document ID	v2.1				v3.1			
	strict	exact	partial	type	strict	exact	partial	type
PMC4806292	0.78	0.84	0.91	0.87	0.78	0.84	0.91	0.86
PMC4817029	0.86	0.90	0.94	0.91	0.86	0.89	0.94	0.92
PMC4980666	0.74	0.76	0.83	0.82	0.72	0.75	0.83	0.82
PMC4981400	0.74	0.81	0.89	0.87	0.76	0.83	0.89	0.86
PMC4993997	0.78	0.84	0.91	0.85	0.78	0.81	0.90	0.85
PMC5012862	0.85	0.92	0.95	0.89	0.83	0.90	0.94	0.89
PMC5014086	0.74	0.87	0.93	0.83	0.72	0.84	0.92	0.83
PMC5063996	0.77	0.88	0.92	0.83	0.77	0.87	0.92	0.85
PMC5173035	0.69	0.81	0.89	0.75	0.68	0.77	0.87	0.80
PMC5603727	0.74	0.80	0.89	0.85	0.74	0.79	0.88	0.86

Table 16. Precision for models v2.1 and v3.1 on individual documents of batch 5 using SemEval evaluation

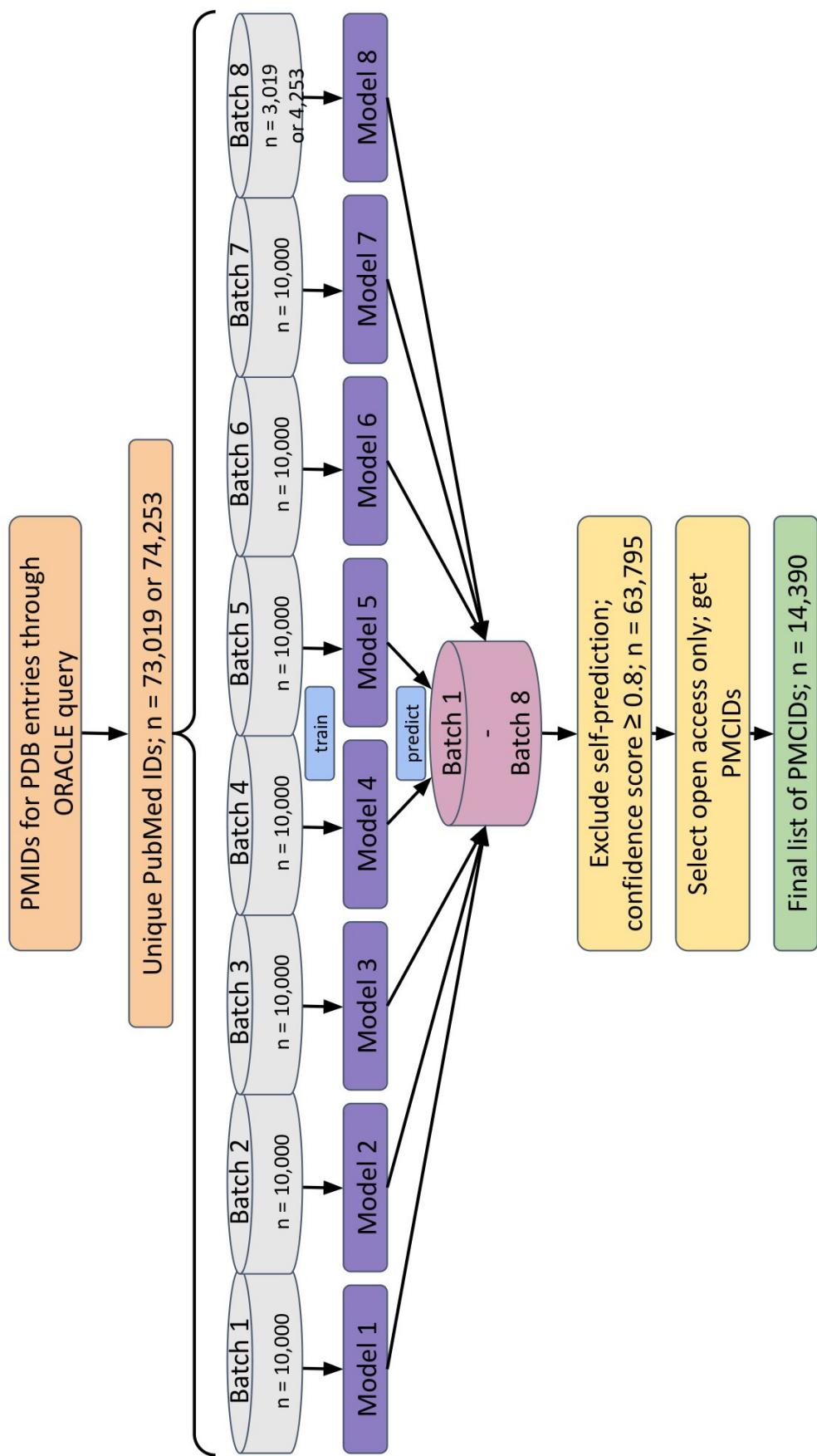


Figure 1. Schema of literature selection workflow

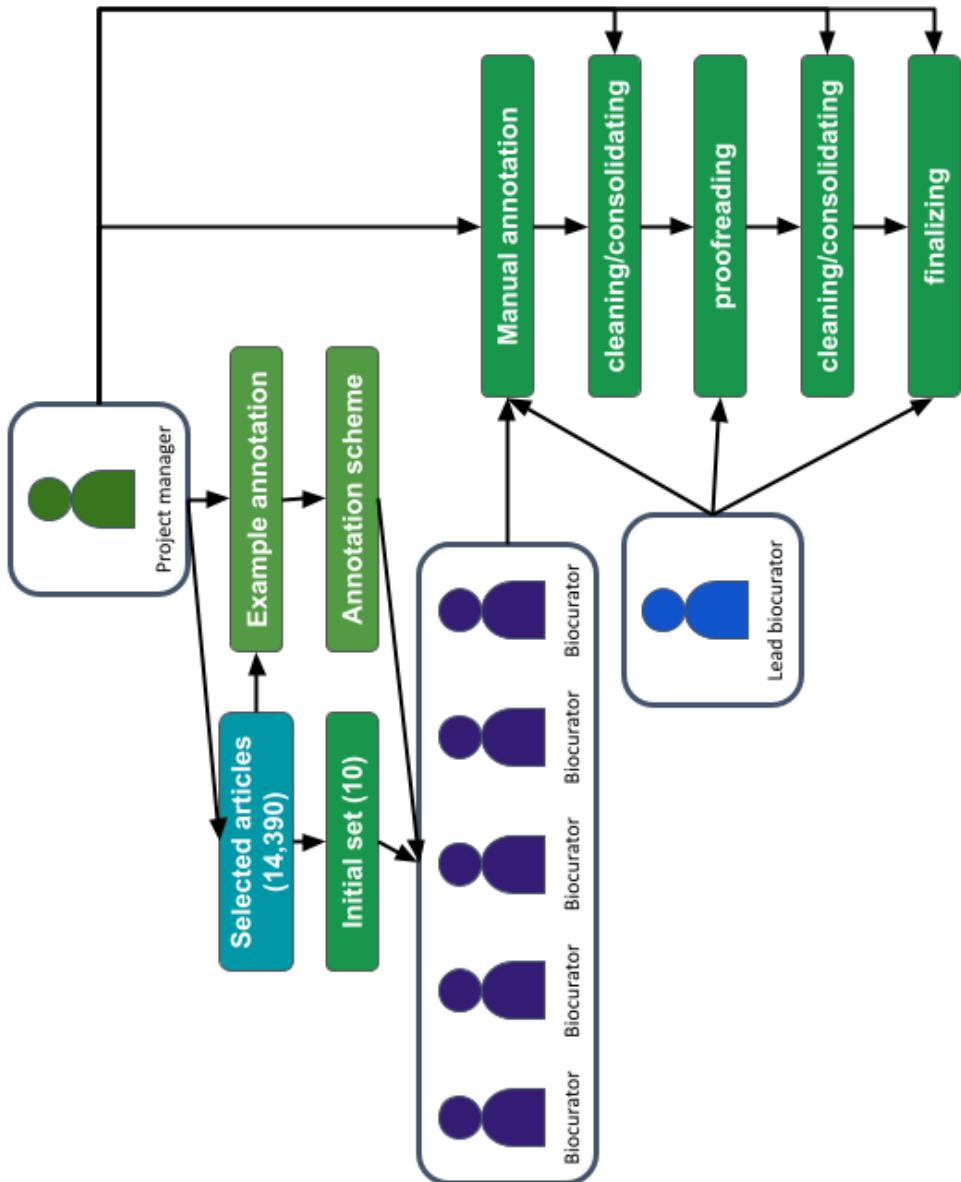


Figure 2. Schema of manual annotation workflow

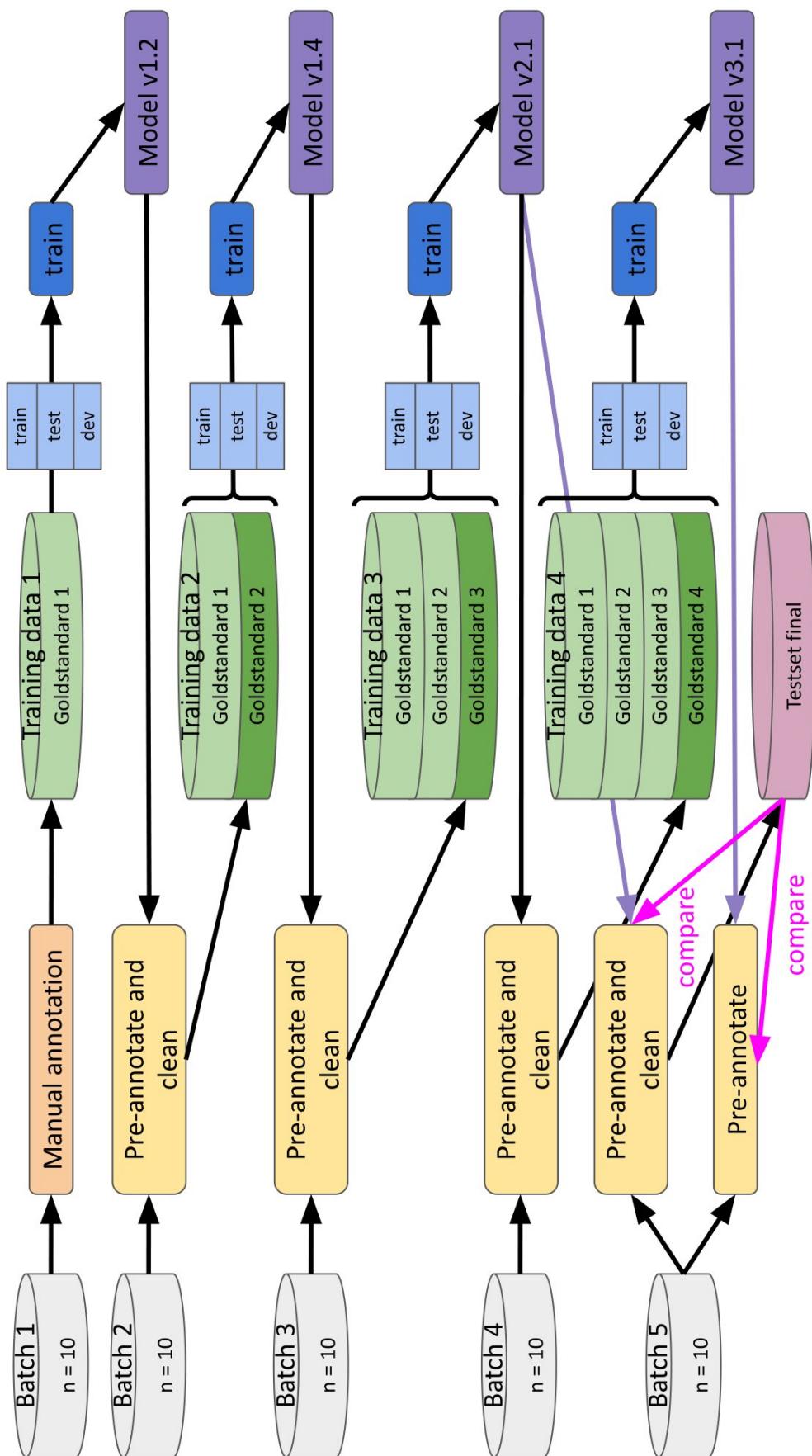


Figure 3. Iterative, human-in-the-loop build up of training data

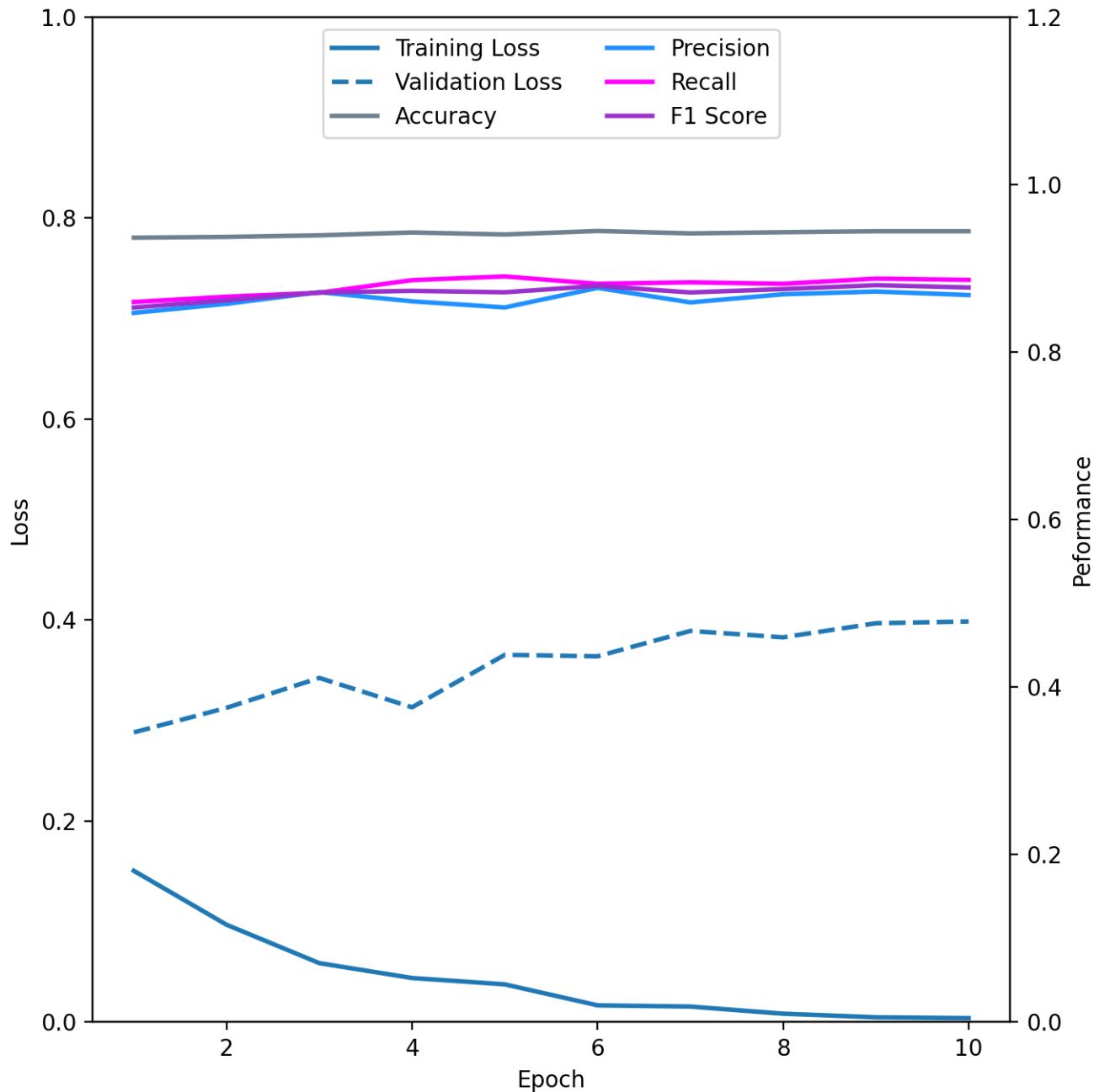


Figure 4. Training and validation loss for the first model v1.2 as well as accuracy, precision, recall and F1-measure

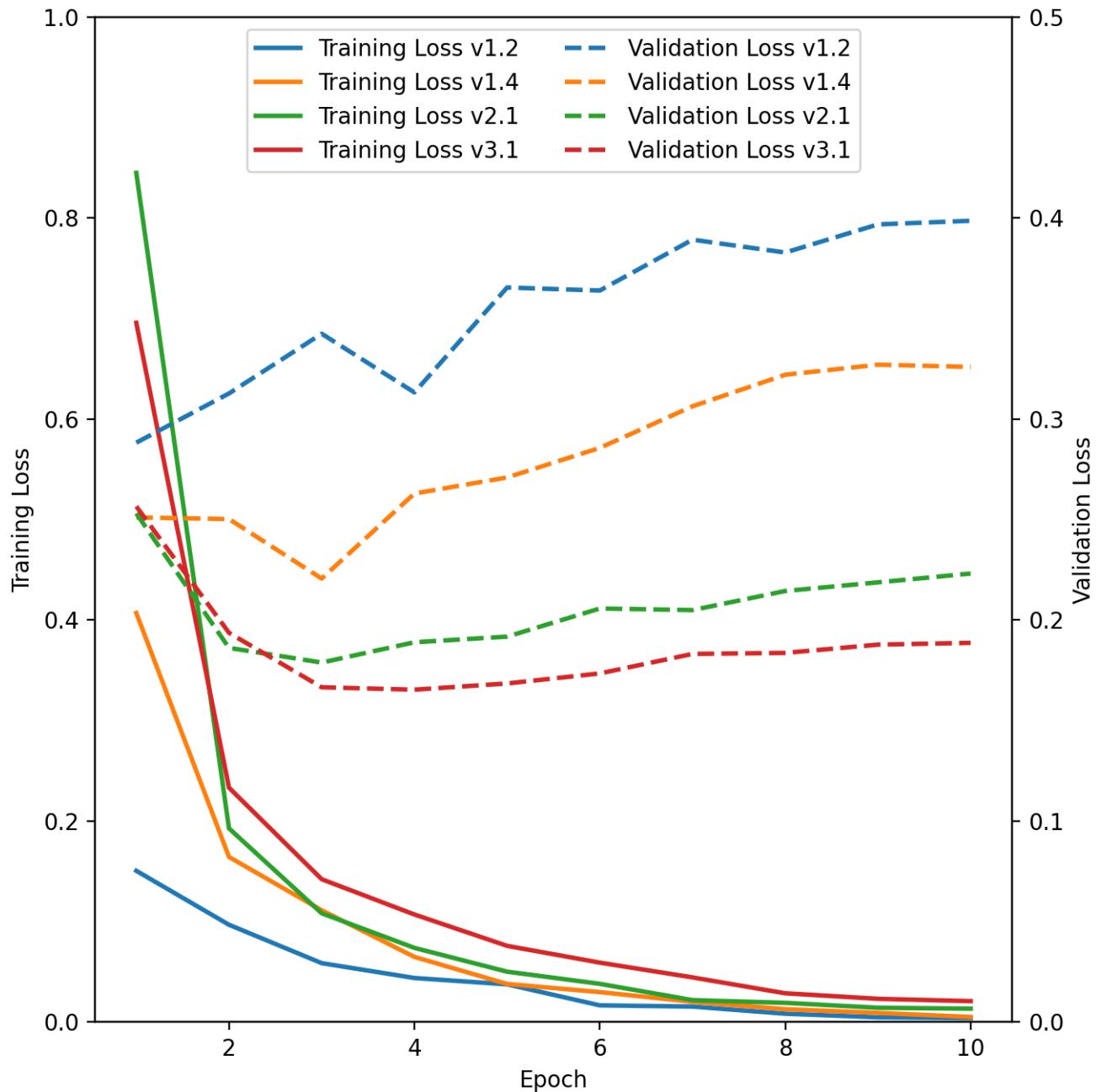


Figure 5. Training and validation loss for the different consecutive models v1.2, v1.4, v2.1 and v3.1

Document ID	v2.1				v3.1			
	strict	exact	partial	type	strict	exact	partial	type
PMC4806292	0.73	0.78	0.85	0.81	0.73	0.79	0.85	0.81
PMC4817029	0.84	0.89	0.92	0.90	0.83	0.86	0.90	0.88
PMC4980666	0.74	0.77	0.84	0.82	0.71	0.74	0.82	0.81
PMC4981400	0.75	0.82	0.90	0.88	0.73	0.80	0.86	0.83
PMC4993997	0.74	0.80	0.87	0.81	0.75	0.78	0.86	0.81
PMC5012862	0.78	0.84	0.87	0.82	0.75	0.81	0.84	0.80
PMC5014086	0.70	0.82	0.88	0.78	0.68	0.80	0.87	0.78
PMC5063996	0.74	0.86	0.88	0.80	0.72	0.82	0.87	0.80
PMC5173035	0.67	0.79	0.87	0.73	0.67	0.75	0.85	0.78
PMC5603727	0.72	0.77	0.86	0.82	0.71	0.75	0.86	0.83

Table 17. Recall for models v2.1 and v3.1 on individual documents of batch 5 using SemEval evaluation

Document ID	v2.1				v3.1			
	strict	exact	partial	type	strict	exact	partial	type
PMC4806292	0.75	0.81	0.88	0.84	0.76	0.82	0.88	0.83
PMC4817029	0.85	0.90	0.93	0.91	0.85	0.88	0.92	0.90
PMC4980666	0.74	0.77	0.84	0.82	0.72	0.74	0.82	0.81
PMC4981400	0.75	0.82	0.89	0.87	0.74	0.82	0.88	0.84
PMC4993997	0.76	0.82	0.89	0.83	0.76	0.79	0.88	0.83
PMC5012862	0.81	0.88	0.91	0.85	0.79	0.86	0.89	0.84
PMC5014086	0.72	0.85	0.91	0.80	0.70	0.82	0.89	0.80
PMC5063996	0.75	0.86	0.90	0.82	0.75	0.85	0.90	0.83
PMC5173035	0.68	0.80	0.88	0.74	0.67	0.76	0.86	0.79
PMC5603727	0.73	0.79	0.87	0.83	0.72	0.77	0.86	0.85

Table 18. F1-measure for models v2.1 and v3.1 on individual documents of batch 5 using SemEval evaluation