Whitepaper:
ProtENN2: Per-residue protein function prediction using neural networks[‡]

Oct 6, 2023

Maxwell L. Bileschi[1,*], David Belanger[1], Irina Ponamareva[2,3], Antonina Andreeva[2], Jaina Mistry[2, †], Bryony Braschi[2], Elspeth Bruford[2,4], Additional collaborators from EMBL-EBI[2, †], Alex Bateman[2], and Lucy J. Colwell[1,3,*]

[1] Google DeepMind
[2] European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI), Wellcome Genome Campus, Hinxton. UK
[3] Dept. of Chemistry, University of Cambridge, Cambridge, UK
[4] Dept. of Haematology, University of Cambridge, Cambridge, UK
† Previous affiliation
[*] Correspondence to mlbileschi@google.com and lcolwell@google.com

‡ A full manuscript is forthcoming.

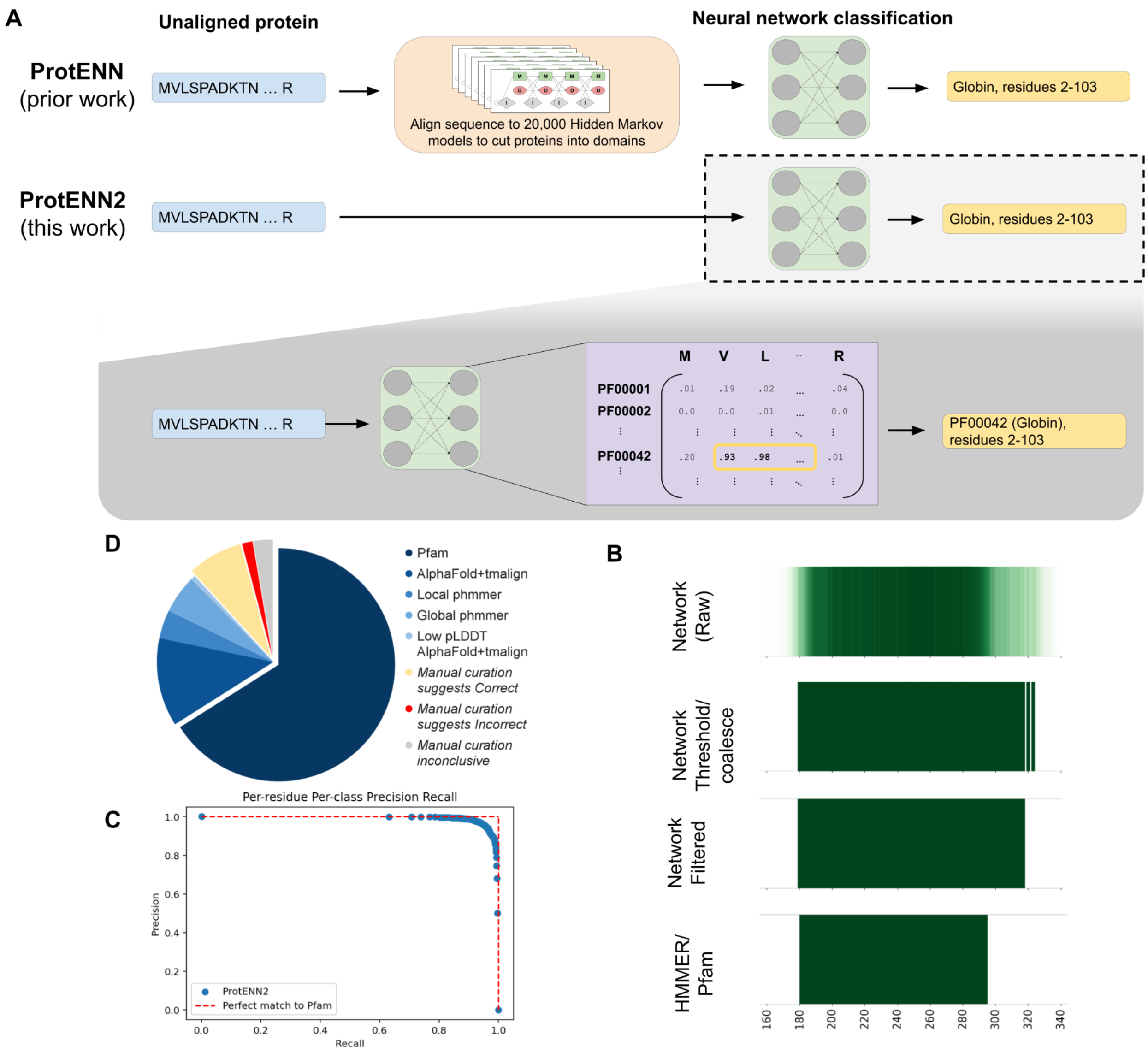

A
Unaligned protein
Neural network classification
ProtENN
(prior work)
MVLSPADKTN ... R
Align sequence to 20,000 Hidden Markov models to cut proteins into domains
Globin, residues 2-103
ProtENN2
(this work)
MVLSPADKTN ... R
Globin, residues 2-103
MVLSPADKTN ... R
M V L ... R
PF00001 .01 .19 .02 ... .04
PF00002 0.0 0.0 .01 ... 0.0
PF00042 .20 .93 .98 ... .01
PF00042 (Globin), residues 2-103
D
Pfam
AlphaFold+tmalign
Local phmmer
Global phmmer
Low pLDDT AlphaFold+tmalign
Manual curation suggests Correct
Manual curation suggests Incorrect
Manual curation inconclusive
B
Network (Raw)
Network Threshold/ coalesce
Network Filtered
HMMER/ Pfam
160
180
200
220
240
260
280
300
320
340
C
Per-residue Per-class Precision Recall
Precision
Recall
0.0
0.2
0.4
0.6
0.8
1.0
ProtENN2
Perfect match to Pfam

Figure 1: A) A model is trained to predict both the set of Pfam domains that are found in a full protein sequence and also the corresponding sequence region(s). A raw, unaligned amino acid sequence is input to a neural network to produce a sequence-length by number-of-Pfam-classes matrix M that contains the predicted confidence that each residue belongs to each Pfam domain. A straightforward algorithm transforms this large, mostly-empty matrix into a list of domain calls for a protein. B) Illustration of straightforward domain-calling procedure, and comparison to Pfam HMMER calls; no calls fall outside this range. C) Per-residue precision-recall of model compared to Pfam 32.0. D) Accuracy of domain calls by ProtENN2 on proteins far from the training set, assessed by multiple tools, in addition to Pfam.

Summary

We trained and publicly released an ensemble of 31 convolutional neural networks to annotate whole protein sequences with both (i) the predicted domains found in each sequence, and (ii) the residue coordinates of these predictions, as depicted in Figure 1A. To generate these predictions, we first predict membership across all 19,632 Pfam families from Pfam v35.0 for each residue, and then convert these predictions into contiguous regions or predicted domains (Figure 1B, Methods). We further predict Pfam

clan membership–defined as matching some HMM that belongs to a clan–for each residue, for each of 655 Pfam clans. We introduce a benchmark task for both domain calling and localization, and use this task to evaluate the performance of a model trained on a randomly chosen 80% of the 45.6 million full-length UniProt reference proteome(UniProt Consortium 2019) sequences that are annotated in Pfam v32.0 (Algorithm M1).

Compared to Pfam, ProtENN2 learns to predict family membership per-residue accurately, but not perfectly (Figure 1C), as is desired–our goal is to correctly predict domains where Pfam does not. To verify the domain calls made by this approach without restricting our analysis to those calls already in Pfam, we use structural corroboration via AlphaFold(Jumper et al. 2021; Zhang and Skolnick 2005) and tmalign(Jumper et al. 2021; Zhang and Skolnick 2005) where possible (44% of cases) and use phmmer(Eddy 2011) otherwise (Figure M5). By combining these approaches with additional expert human curation we estimate the precision of ProtENN2 domain calls is between 99.2 and 99.8% (see Methods for details).

It is important to ensure the network isn't just "memorizing" the training set–that is, just performing well because the test set is so close to the training set("Assignment of Homology to Genome Sequences Using a Library of Hidden Markov Models That Represent All Proteins of Known Structure" 2001; Petti and Eddy 2022). To address this issue, we evaluate the ability of the model to detect remote homologs using two sets of sequences: those with low sequence identity to any other sequence (including the training data), and those where Pfam cannot detect homology at any reasonable statistical threshold.

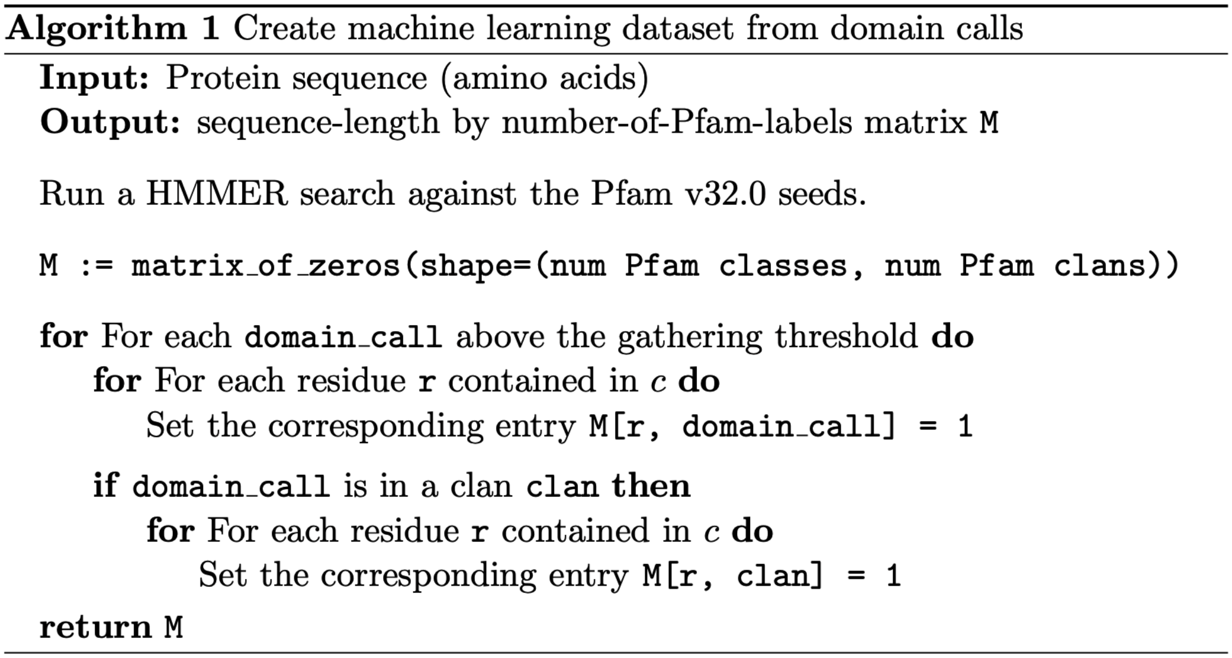
The above-described computational validation pipeline corroborates 17,091 (87.8%) of the calls on a set of held out test sequences that are each the only sequence from their UniRef50 cluster found across the combined train, dev and test sets (Figure 1D, Methods). A further remote homology test set is defined using the 5.9% of proteins (7,941 domains) in our held-out test set have a ProtENN2-called domain where Pfam has no call at very low thresholds–5 bits–indicating these calls are very remote from a currently-accepted definition of these families (Methods). Further, 14% of these 7,941 remote putative domain sequences lack structural predictions in the AlphaFold DB.

As above, using our computational validation pipeline for corroboration, together with additional expert manual curation for those calls that are inaccessible to Pfam/HMMER, we find support–either computational or from a curator–for ~82% of calls. Curators find that around 5% of ProtENN2 model calls for this highly remote set of sequences are probably incorrect (supplemental CSV). Finally, we assessed ProtENN2's accuracy using a metagenomic dataset(Richardson et al. 2022). We applied ProtENN2 to 252 known, expressed proteins from the human gut microbiome, and found that 246/252 ProtENN2 predictions were in concordance with AlphaFold, phmmer, or expert curator opinion (Methods).

Data

We use pfamseq from Pfam v32.0 (as in (Bileschi et al. 2022)). We split this set randomly into 80% training data, 10% dev data, and 10% test data. A randomly-selected subset of 100,000 sequences from the test subset is investigated in more detail. This helps us assess the quality of both (i) the specific residues called by the network, and (ii) the resulting sequence regions (domains). We include two remote homology evaluations, including proteins far from the training data, and domains that are extremely statistically unlikely per current Pfam HMMs.

We randomly select a set of 200 million residue-class pairs class-residue pairs for Figure 1C from the set of 100,000 held out test sequences.



```
Algorithm 1 Create machine learning dataset from domain calls
  Input: Protein sequence (amino acids)
  Output: sequence-length by number-of-Pfam-labels matrix M

  Run a HMMER search against the Pfam v32.0 seeds.

  M := matrix_of_zeros(shape=(num Pfam classes, num Pfam clans))

  for For each domain_call above the gathering threshold do
      for For each residue r contained in $c$ do
          Set the corresponding entry M[r, domain_call] = 1

      if domain_call is in a clan clan then
          for For each residue r contained in $c$ do
              Set the corresponding entry M[r, clan] = 1
  return M
```

Algorithm M1. Create a machine learning dataset from domain calls.

Network

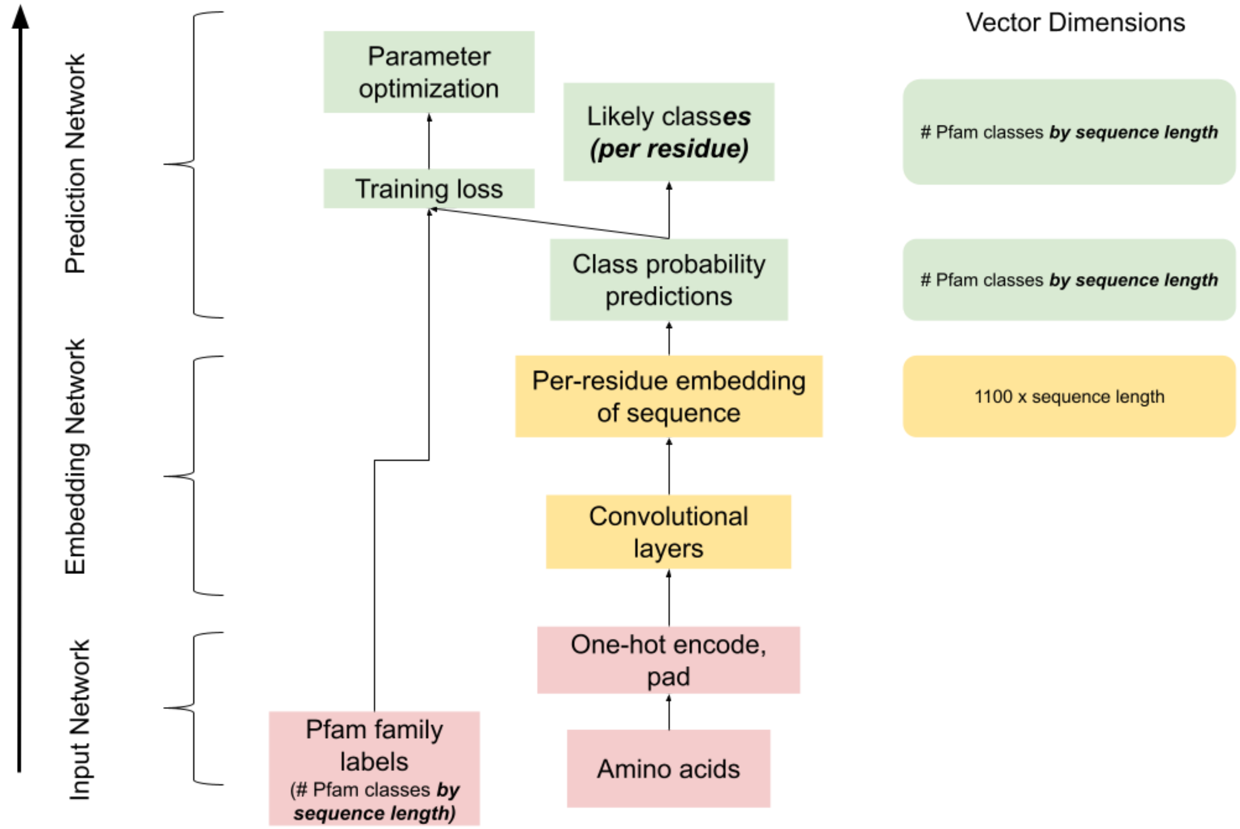

Vector Dimensions
Prediction Network
Embedding Network
Input Network
Parameter optimization
Training loss
Likely classes (per residue)
Class probability predictions
Per-residue embedding of sequence
Convolutional layers
One-hot encode, pad
Amino acids
Pfam family labels (# Pfam classes by sequence length)
# Pfam classes by sequence length
# Pfam classes by sequence length
1100 x sequence length

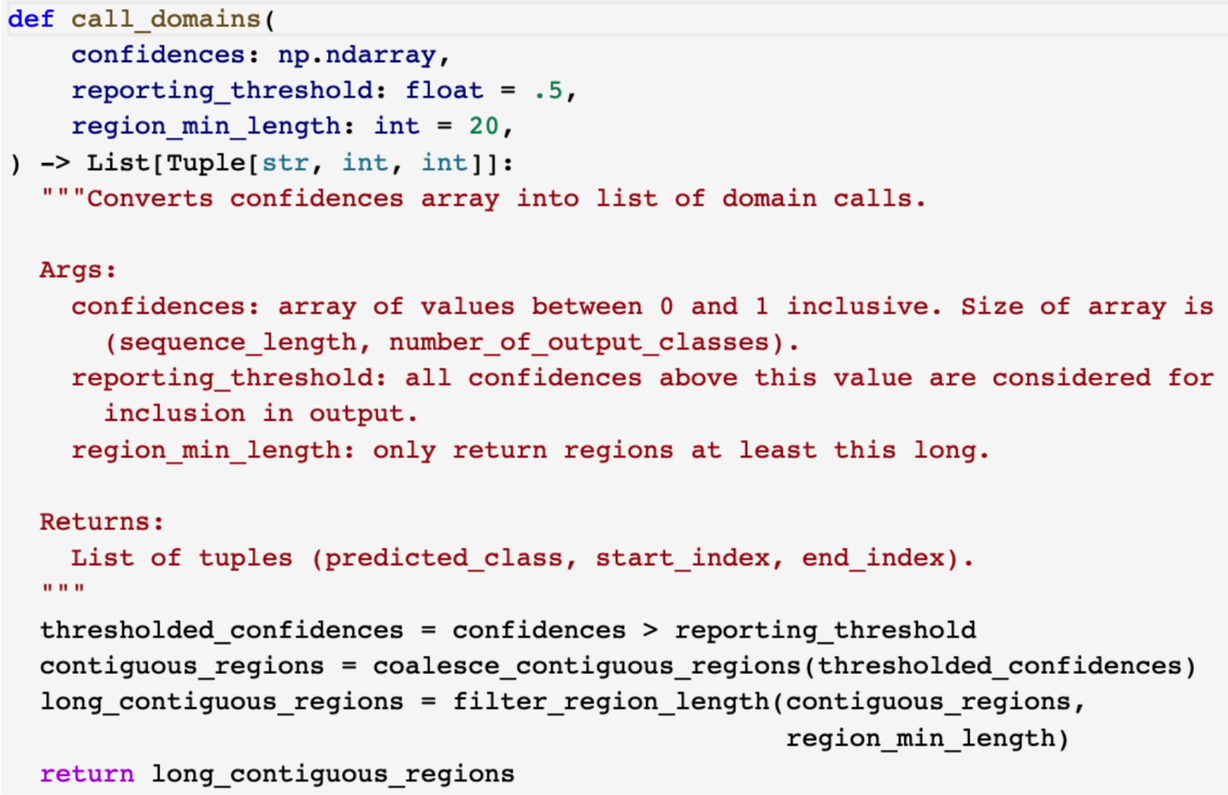
Figure M1: Network architecture. The network architecture is almost identical to Bileschi et al(Bileschi et al. 2022), with the differences being highlighted in bold and italics in the above figure. These are that the labels, instead of being a one-hot vector of length # Pfam classes, are now a multi-hot 2d vector of length # Pfam classes ***x sequence length***. Further, the predictions outputted by the model, and thus used for computing the training loss, are similarly the same shape vectors. The sequence embeddings are computed as in (Bileschi et al. 2022). The ensembling of individual networks (ProtCNN2) into an ensemble (ProtENN2) is done by taking the computed probabilities of every residue, for every class, and averaging. We note that there is a prediction for each Pfam family *and clan*, so that the probability of being within a domain belonging to one of few hundred Pfam clans also gets predicted for each residue.

An important modification of the network ProtENN(Bileschi et al. 2022) is required to produce per-residue Pfam family confidence predictions: after applying convolutional layers, we are left with a 1100-dimensional vector (embedding) for each residue. Instead of pooling this into a single 1100-dimensional vector, we perform a 20,000-way classification problem (into every Pfam class and clan) *for each residue* (Algorithm 1). We compute domain calls from pre-residue calls (Figure 1A) by coalescing confidently-called residues, then filtering out short calls and calls that conflict with each other (Figure 1B, Methods).

Domain calling procedure

We describe Figure 1B in more detail. The first row shows the raw activations from the neural network ensemble for family PF05175 for protein UniProtKB:A0A010NNR2. The second line shows how the thresholded activations at .5 confidence are coalesced into contiguous regions. The third line shows that activations with (probably spuriously) short matches < length 20 are removed. The final line shows the Pfam/HMMER computed label. Interestingly, though it appears that the neural networks' call is too long in this example, G3DSA has a longer call than Pfam for a methyltransferase-containing superfamily, supporting the extra residues called by the neural network. Moreover, the Pfam/HMMER match is a partial HMM match, which is further evidence that the longer match from the neural network may be justified.

We note that omitting calls < 20 residues causes some true matches to be omitted from the output of the domain calling procedure, like very short repeats, but find that overall this improves the precision/recall tradeoff.



```python
def call_domains(
    confidences: np.ndarray,
    reporting_threshold: float = .5,
    region_min_length: int = 20,
) -> List[Tuple[str, int, int]]:
  """Converts confidences array into list of domain calls.

  Args:
    confidences: array of values between 0 and 1 inclusive. Size of array is
      (sequence_length, number_of_output_classes).
    reporting_threshold: all confidences above this value are considered for
      inclusion in output.
    region_min_length: only return regions at least this long.

  Returns:
    List of tuples (predicted_class, start_index, end_index).
  """
  thresholded_confidences = confidences > reporting_threshold
  contiguous_regions = coalesce_contiguous_regions(thresholded_confidences)
  long_contiguous_regions = filter_region_length(contiguous_regions,
                                                 region_min_length)
  return long_contiguous_regions
```

Figure M2: Domain calling algorithm. A procedure to convert the neural network output into a number of domain calls. First, a confidence reporting threshold between 0 and 1 is chosen. All residues with at least this confidence are then coalesced into contiguous regions as candidate domain calls. Finally, because we have a strong prior that domains are conserved units of some considerable length (say, greater than 20 residues), we filter out spurious matches.

Pfam uses the idea of "clan competition" to pick the best domain for a region when multiple calls are present(Mistry et al. 2021). They seek to only have overlapping calls for HMMs belonging to one clan; overlapping calls from different clans are assumed to be nonhomologous, and thus errors (except in the case of nested domains). We use this technique as well, except we further use *inter*-clan competition, and to do so we simply choose the longest domain called, even from different clans (unless the two overlapping calls are known to be nested domains). We find that inter-clan competition removes many short false-positive calls, improving precision. Further, we find that although Pfam/HMMER use e-value statistics to do clan competition, choosing the longest call for ProtENN2 gives the same call as

Pfam/HMMER *within* a clan very often, only differing on <5% of calls, with approximately half of this movement within clans being between two families without substantial differences.

Whenever the network predicts a family for a residue at a higher probability than the predicted probability for the corresponding clan, we set the residue's *clan* probability to that of the family as a post-processing step.

Per-class, per-residue calling

Using Pfam as ground truth, we see a precision of 0.964, and a recall of 0.943. (Figure 1C). It is not our goal to get perfect precision (there truly are domains that are missing from Pfam). However, getting a recall of 100% is more desirable–assuming Pfam is *always* correct–because we'd like to call every true domain that Pfam does. See "Calls missing from ProtENN" below.

Computing precision and recall for domains with Pfam

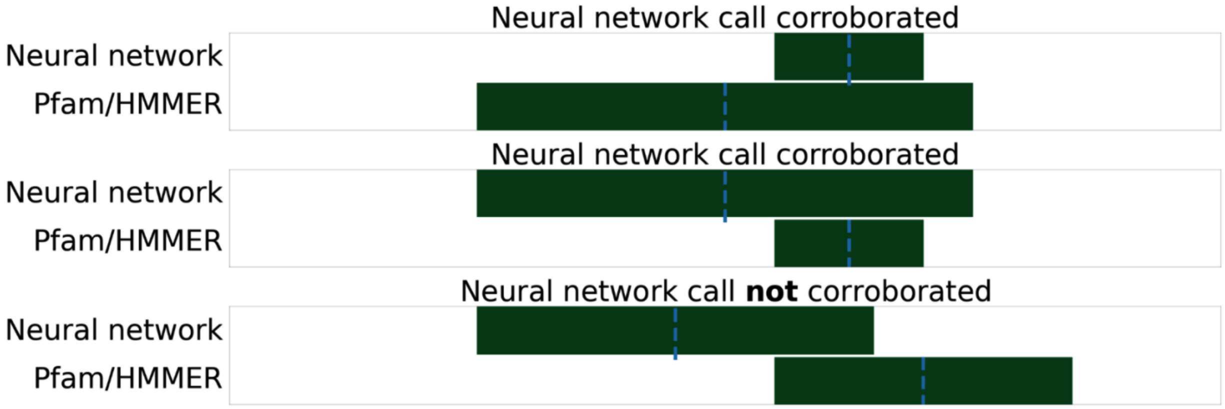

Neural network call corroborated
Neural network
Pfam/HMMER
Neural network call corroborated
Neural network
Pfam/HMMER
Neural network call **not** corroborated
Neural network
Pfam/HMMER

Figure M3: Midpoint overlap. When our neural network call's midpoint (dotted blue line) lies within Pfam/HMMER, we consider our call to be covered by Pfam/HMMER's call.

For each domain call, we consider our networks' call and Pfam's call to be corroborated if the midpoint of one of these calls lies within the region of the other call, a technique we refer to as "single midpoint overlap" (Figure M3). This condition is somewhat permissive, but we also measure precision and recall at the residue level to provide a different view of concordance between Pfam/HMMER and the neural networks.

Further, we consider the call correct if it is to the same "lifted clan"; as in (Bileschi et al. 2022), two families are in the same lifted clan if they are in the same clan, or if they are the same family (and that family belongs to no clan).

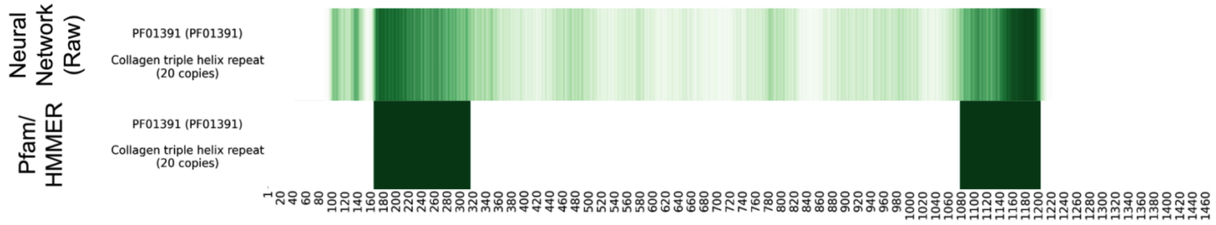


Figure M4: Raw neural network probabilities. Neural networks show collagen repeats where Pfam/HMMER does not when examining the neural network output before the domain calling procedure for a collagen repeat for protein UniProtKB:P02461. We find that the characteristic collagen GPP motif exists throughout, as predicted by the neural network. Notably, domain calls for PF01391 in the middle region of the protein are not predicted by Pfam/HMMER even at a very low reporting threshold of 5 bits. Per-residue functional localization provided by our neural network allows for this insight, and builds confidence in the neural network model's accuracy. We note that this middle region *is covered* by the envelopes that HMMER calculates as part of the domain calling procedure.

We note that "double midpoint overlap" is also a reasonable choice for measuring the performance of our neural networks instead of "single midpoint overlap" (Figure M3), but we chose not to use this for two reasons: 1) as our experiments show (Figure 1D), the currently-called domains from Pfam/HMMER aren't as quite reliable as we needed - sometimes they make surprising choices about domain boundaries (Figure M4), and 2) the main concern of this work isn't necessarily annotating domains that are already part of the training data; instead, it's generalizing beyond Pfam.

Computing recall for domains with AlphaFold and TMalign

We consider a ProtENN2 domain prediction for a Pfam family correct if there is a structurally homologous region that has the same Pfam family already called. This is extremely similar to TMalign and AlphaFold's use in (Gane et al. 2022). In more detail, we consider two regions to be structurally homologous if

- There are AlphaFold v3 structures for both regions (so that structures for which AlphaFold performed particularly poorly are not included)
- These structures align with TMalign
- The computed alignment has a tmalign score >= .5 (a value mentioned in (Zhang and Skolnick 2005) for "roughly in the same fold")
- The computed alignment's residues cover 3 distinct secondary structure regions in both the subject and query sequences, including a "loop" as a distinct region (so that, for example, two arbitrary alpha helices are not considered "homologous")
- The mean pLDDT over the aligned residues is >= 70 (so that, for example, we don't try to align disordered regions).

These rules apply to both ProtENN2 calls to Pfam families *as well as to clans*. That is, when ProtENN2 produces a clan call with no corresponding family, we can apply the same procedure.

We also report the number of calls that meet most of these criteria, but have low pLDDT scores, as we sometimes manually see that AlphaFold has been a bit under-confident.

Computing recall for domains with phmmer

Much like with AlphaFold and TMalign, we consider a ProENN2 domain prediction for a Pfam family (or clan) correct if there is currently-labeled Pfam domain call

- With the same clan label (or the same family if the family has no clan)
- The call has a phmmer alignment *to ProtENN2's called region*. We call this a "local" phmmer search because we only consider both Pfam's called region and ProtENN2's called region, not the entire protein sequences. We require a bit score >= 25.0. This value was chosen as follows:
    - Independently curating a number of entries, qualitatively assessing each call's support.
    - By noting that most Pfam HMM hmmsearch calls use a threshold of around 25 bits.

- By noting that the default incT bit score value chosen for phmmer's web server is 25.0(Finn et al. 2015).

By analogy, we consider a "global" phmmer search as one where two entire proteins are compared, not just the two called regions.

Statistical breakdown for recall on randomly selected 100k proteins

After choosing a sensible default confidence value (trading off precision and recall), ~83% of our calls are supported by a homologous, single-midpoint-overlapping Pfam call, another ~7% of our calls are supported by a localized, structural alignment between our called domain and a high-pLDDT AlphaFold structure having the same Pfam call. Local and global phmmer detected strong homology to a member of the same Pfam lifted clan for another ~5.5%. Low-confidence AlphaFold structures provide weaker evidence that .2% of our additional calls are correct. We are thus left with 3.6% of our calls as uncorroborated by automated tooling. Through a process of manual curation done on 20 randomly selected proteins–by the authors–we estimate that of these 3.6% of calls: 80% are correct, 5% are incorrect, and we were unable to make a determination for 15%. This gives us an estimated precision of between 99.2% and 99.8%, depending on whether the undetermined calls are correct (Supplement). We note that some ProtENN2 calls may be *localized* incorrectly within a sequence; these are considered corroborated by the global phmmer search.

UniRef50 benchmark set

We measure performance on sequences that are comparatively distant from our training set by measuring performance on 20,749 (20.7%) sequences from our test set found in size-1 UniRef50 clusters(Suzek et al. 2015) that appear in our test set, on which ProtENN2 makes 19,456 domain calls. We selected a random set of 50 proteins for our manual curation set.

Calls that are correct for a protein, but are localized incorrectly

We can repeat the above phmmer procedure by trying to align entire proteins–without regard to location within either the query or subject–to look for homology. This is what we call a "global" phmmer search: a search that tries to align two protein sequences instead of two protein domains. This procedure is useful for identifying homology for particularly short domain calls, like some beta propeller repeats, or zinc fingers, where the called region might be truly homologous, but might not be able to reach the statistical significance of 25.0 bits simply because of the number of residues. This approach *does* allow some false positive calls, where the ProtENN2-called domain is present in a protein, but is not localized appropriately. We find this happens most often when a call is "overextended", but not connected. For example, protein B7ANX8 has two ProtENN2-derived calls to PF18283 in Pfam-N, one from 525 to 550, and the other from 564 to 734. Based on the AlphaFold structure and based on an hmmsearch call to PF18283, the second call is correct. However, the first call is not correct, and results in a poor alignment to other PF18283 members. Further, when using a global phmmer search to corroborate ProtENN2 calls, this sort of mistake is unfortunately considered correct. In cases such as this, ProtENN2 was "trying" to extend the PF18283 call further toward the N terminal (which is arguably correct, given how some PF18283 calls include this disordered region), but the probability mass dipped below the reporting threshold in the middle, causing the domain calling algorithm to (incorrectly) identify this as two separate domains.

Performance on a random test-train split of UniProt Reference Proteomes

As an analogous chart to Figure 1D, we present the pie chart when the randomly held-out test set of 100,000 proteins are used.

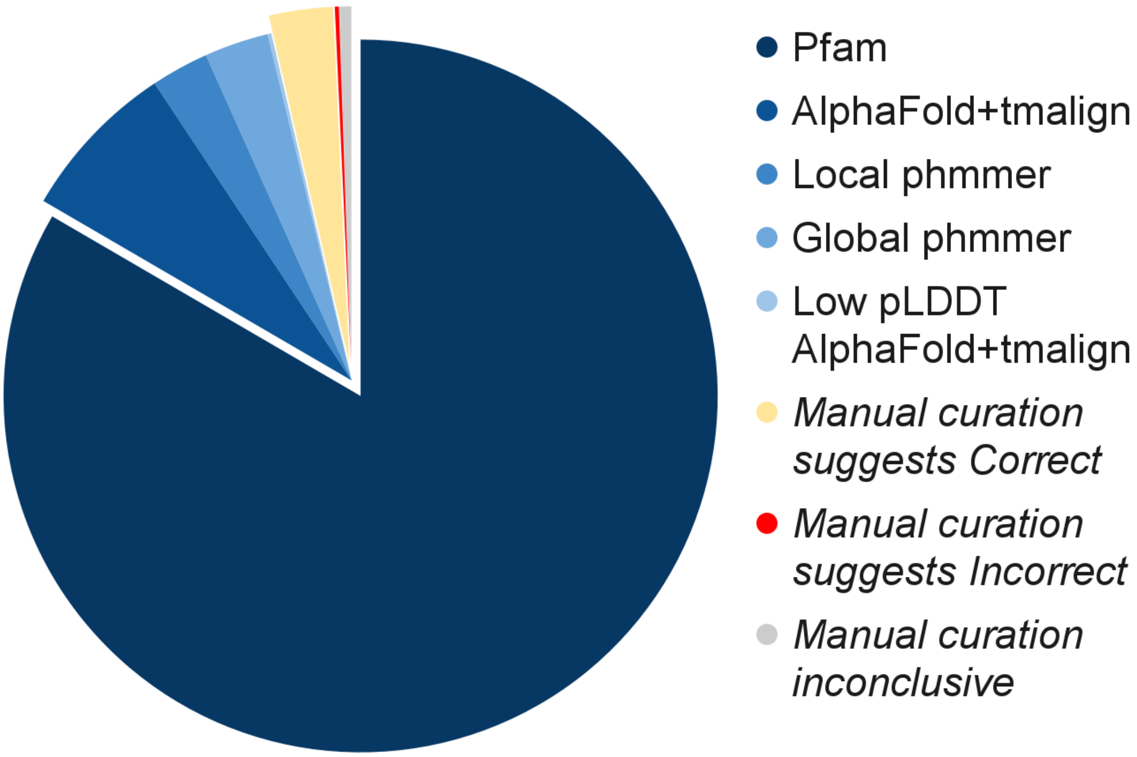


Figure M5: estimated precision of calls on randomly-split test set.

Annotating MGnify

It is computationally very expensive to run HMMER across all of these sequences (and further, the Pfam gathering thresholds are not well-calibrated due to its size), but we ran all ProtENN2 ensemble elements (31 of them) for all 2.5 billion proteins, where HMMER was run on cluster representatives, chosen by mmseqs2(Steinegger and Söding 2017).

Difference between MGnify and Pfam domain calling

Professional curators at MGnify identified a reporting threshold of .05 as a suitable tradeoff between precision and recall for their use case, whereas the Pfam team determined that a reporting threshold of .025 was more apt for their use case. Further, the team at MGnify didn’t want to compete calls (to get only one call per clan) or to use competition to filter out potentially overlapping nonhomologous calls–they examined some examples of this and determined that the calls actually were homologous, but clan relationships were missing from Pfam. However, for Pfam, inter-clan competition is used.

The authors of this paper recommend the settings used for the Pfam database release as defaults.

Funding

EMBL core funding, BBSRC/NSF ECOD funding. The work of the HGNC is supported by National Human Genome Research Institute (NHGRI) grant U24HG003345. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

Acknowledgements

The authors thank the HUGO Gene Nomenclature Committee for its review of novel predictions on the human genome; Jamie Smith for countless conversations and guidance throughout this project. L.J.C. gratefully acknowledges support from the Simons Foundation.

Author contributions

Maxwell L Bileschi: conception, implementation, curation, writing
David Belanger: machine learning methodology, code review
Irina Ponamareva: curation
Antonina Andreeva: curation
Jaina Mistry: software implementation and curation
Bryony Braschi: curation
Elspeth Bruford: curation
Other EMBL-EBI Collaborators: curation
Alex Bateman: curation, bioinformatics methodology
Lucy J Colwell: conception, bioinformatics methodology, curation, writing

Competing interests

M.L.B., D.B. and L.J.C. performed research as part of their employment at Google LLC. Google is a technology company that sells machine learning services as part of its business. Portions of this work may be covered by US patent WO2020210591A1, filed by Google.

References

"Assignment of Homology to Genome Sequences Using a Library of Hidden Markov Models That Represent All Proteins of Known Structure." 2001. *Journal of Molecular Biology* 313 (4): 903–19.

Bileschi, Maxwell L., David Belanger, Drew H. Bryant, Theo Sanderson, Brandon Carter, D. Sculley, Alex Bateman, Mark A. DePristo, and Lucy J. Colwell. 2022. "Using Deep Learning to Annotate the Protein Universe." *Nature Biotechnology* 40 (6): 932–37.

Eddy, Sean R. 2011. "Accelerated Profile HMM Searches." *PLoS Computational Biology* 7 (10): e1002195.

Finn, Robert D., Jody Clements, William Arndt, Benjamin L. Miller, Travis J. Wheeler, Fabian Schreiber, Alex Bateman, and Sean R. Eddy. 2015. "HMMER Web Server: 2015 Update." *Nucleic Acids Research* 43 (W1): W30–38.

Gane, A., M. L. Bileschi, D. Dohan, E. Speretta, A. Héliou, L. Meng-Papaxanthos, H. Zellner, et al. 2022. “ProtNLM: Model-Based Natural Language Protein Annotation.” *Preprint*.

Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, et al. 2021. “Highly Accurate Protein Structure Prediction with AlphaFold.” *Nature* 596 (7873): 583–89.

Mistry, Jaina, Sara Chuguransky, Lowri Williams, Matloob Qureshi, Gustavo A. Salazar, Erik L. L. Sonnhammer, Silvio C. E. Tosatto, et al. 2021. “Pfam: The Protein Families Database in 2021.” *Nucleic Acids Research* 49 (D1): D412–19.

Petti, Samantha, and Sean R. Eddy. 2022. “Constructing Benchmark Test Sets for Biological Sequence Analysis Using Independent Set Algorithms.” *PLoS Computational Biology* 18 (3): e1009492.

Richardson, Lorna, Ben Allen, Germana Baldi, Martin Beracochea, Maxwell L. Bileschi, Tony Burdett, Josephine Burgin, et al. 2022. “MGnify: The Microbiome Sequence Data Analysis Resource in 2023.” *Nucleic Acids Research* 51 (D1): D753–59.

Steinegger, Martin, and Johannes Söding. 2017. “MMseqs2 Enables Sensitive Protein Sequence Searching for the Analysis of Massive Data Sets.” *Nature Biotechnology* 35 (11): 1026–28.

Suzek, Baris E., Yuqi Wang, Hongzhan Huang, Peter B. McGarvey, Cathy H. Wu, and UniProt Consortium. 2015. “UniRef Clusters: A Comprehensive and Scalable Alternative for Improving Sequence Similarity Searches.” *Bioinformatics* 31 (6): 926–32.

UniProt Consortium. 2019. “UniProt: A Worldwide Hub of Protein Knowledge.” *Nucleic Acids Research* 47 (D1): D506–15.

Zhang, Yang, and Jeffrey Skolnick. 2005. “TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score.” *Nucleic Acids Research* 33 (7): 2302–9.

Supplement

Code availability

Code to download and use the Pfam-N 35.0 models (ProtENN2) is provided in GitHub at https://github.com/google-research/google-research/protenn, licensed under Apache License 2.0.

Manual curation of proteins in random UniProtKB Reference Proteomes split
Manual curation of proteins in UniRef50 split
Manual curation of proteins in no-sensitive-Pfam-calls split