Learning a Functional Grammar of Protein Domains using Natural Language Word Embedding Techniques

Daniel WA Buchan, David T Jones[*]

Department of Computer Science, University College London, Gower Street, London, WC1E 6BT

[*]Corresponding Author
email: d.t.jones@ucl.ac.uk
Tel: +44 (0) 20 7679 2000

Abstract

In this paper, using word2vec, a widely-used natural language processing method, we demonstrate that proteins domains may have a learnable implicit semantic "meaning" in the context of their functional contributions to multi-domain proteins in which they are found. Word2vec is a group of models which can be used to produce semantically meaningful embeddings of words or tokens in a fixed-dimension vector space. In this work, we treat multi-domain proteins as "sentences" where domain identifiers are tokens which may be considered as "words". Using all InterPro [1] pfam domain assignments we observe that the embedding could be used to suggest putative GO assignments for Pfam [2] Domains of Unknown Function.

Keywords: Semantic embedding, word2vec, protein domains, machine learning, function prediction

Introduction

Word2vec [3] is a group of models which can be used to learn the embeddings of words in a continuous fixed-dimension vector space, given a corpus of sentences as training data. Often Natural Language Processing (NLP) tasks consider words as sets of unrelated tokens, subjecting them to no-more rigorous analysis than simple frequency counting. While this is mathematically and computationally convenient, it ignores the fact that most words have degrees of similarity, such as verbs with differing tenses, adverbs with differing endings or words which share the same suffixes. Word2vec aims to produce embeddings of words in a vector space where distance in the vector space correctly encodes the degree to which words or terms are similar or can be used in similar semantic context. Although a great degree has been written about these methods it remains unclear exactly why these models are so performant [4]. Nevertheless, they do show good performance in the task of clustering words with related semantic meaning, and interested readers should consult the original paper for further details of the model [3]. Since lexical word embeddings have become popular, they have been adapted and applied directly to protein and gene sequences. prot2vec, gene2vec and seq2vec are examples of such methods [5, 6]. Another prior application of word2vec is the work of A. Viehweger [7], applying protein domain embeddings as a method to encode whole genomes.

Proteins are often composed of discrete domains, and these can either be conceptualised as sub-sequences of independent protein sequences which share homology (and by extension evolutionary origin) [2], or alternatively, domains may be considered structurally, where they are subsections of the proteins which are compact, independently folding and observed to be shared between a variety of proteins [8-10]. An extension of this observation, that proteins can be decomposed into sets of domains, is the hypothesis that domains act as sub-functional units and when composed together, a protein's given combination of domains is what gives rise to the protein's overall specific function [11, 12] In the following study we show that protein domains can be embedded in a "semantically" meaningful vector space and that this embedding space reflects meaningful information about the functional roles (in terms of GO term assignments) of the individual protein domains.

Protein function prediction has received a great deal of attention in the preceding 20 years [13] and a great number of function prediction methods have been developed. Many of these make use of sequence comparison and some manner of nearest neighbour functional assignment [14, 15]. As the field has progressed work has been done to integrate more sophisticated statistical methods and models with many contemporary methods leveraging machine learning with ensemble or meta-prediction methodologies. Current state of the art in protein function is measured by the Critical Assessment in Function Annotation (CAFA) community experiment [16]. In this experiment groups attempt to predict experimentally validated Gene Ontology (GO) terms [17] over a blind set of unannotated protein sequences. The most successful methods in CAFA employ a wide variety of predictive methodologies. Among the most common are methods which integrate data and annotations from a wide variety of sources including blast searches, protein-protein interaction networks, multiple sequence alignment analysis, sequence analysis, expression data and many more [18-21]. A number of other successful methodologies eschew integrating heterogenous data sources and make use of more focussed analyses such as phylogenetic analysis [22], literature analysis [23], MSA analysis [24], domain function analysis [FunFAM, Superfam]. Information about protein domains is typically only included indirectly such as in the methods INGA and PFPDB which make use of PFAM to derive phylogenetic or domain architecture patterns. Less common are methods which directly attempt to annotate domains with function and then leverage this information for function prediction. Both the SIFTER, CATH-Funfam [25]and Superfamily-dcGO [26] methods in CAFA were successful methods which directly leverage such domain function annotations. It is clear that understanding the relationship between protein domains and their function can make a significant contribution to accurate function prediction. Nevertheless, even with the wide range of prediction methodologies, performance and progress in the CAFA experiment indicates that protein function prediction remains a challenging problem in the field of bioinformatics.

In the following work we discuss the use of Word2vec in protein domain embedding. We prepare such a domain embedding and attempt to explore the its properties to discern whether such embeddings encode biological information that may be useful in either a predictive or analytic context. Such embeddings may be a useful adjuncts or input features in protein function prediction as it may give a homology-free way to characterise and functionally cluster protein domains. At the end of the paper we note that such an embedding could be used for the purposes of homology-free GO term inheritance and we show a naïve application of this for PFAM Domains of Unknown Function.

Method

Datasets

InterPro 62 [1] was downloaded along with the associated GO and protein domain assignments. The files were parsed to extract only the Eukaryotic proteins and their GO and Pfam domain assignments. This work looks only at eukaryotic proteins as there are few examples of proteins with multiple domains with independent evolutionary histories in the bacterial and archaeal kingdoms, as such little domain context information would be available for proteins from those kingdoms. Only GO assignments with the following evidence codes were retained: EXP, IBA, IDA, IEP, IGC, IGI, IMP and IPI. These are (respectively); Inferred From Experiment, Inferred from Biological Aspect of ancestor, inferred from Direct Assay, Inferred from Expression pattern, Inferred from Genomic Context, Inferred from Genetic Interaction, Inferred from Mutant Phenotype and Inferred from Physical Interaction. This eliminates all the high throughput and more tenuous computational annotation assignments. The resulting dataset contains 9,030,650 eukaryotic proteins, which have domain assignments over 11,355 of the available Pfam domain families and these proteins are associated with annotations from 2,358 GO Terms.

Not all regions within each protein have been assigned to domains (see table 1). In large part because not all domains are known and assigned but also because many Eukaryotic proteins possess regions of intrinsic disorder [27], regions of low complexity or coiled-coiled sequences. All such unassigned regions were compiled (see below). As Word2vec analyses words based on the semantic context of neighbouring words representing unassigned regions in our corpus could contain important domain context information, and so we wished to preserve this.

These data were then used to derive which Pfam domains are seen to be associated to which GO terms. For every Pfam domain, we associated all GO terms assigned to all the proteins the Pfam domain was observed in. This assigns a varied bag of GO terms to each Pfam domain and this bag of terms can be viewed as representing the spectrum of observed functional diversity for that Pfam domain.

Unassigned sequence region assignments

The sequence database for InterPro 62 was masked for both coiled coil and low complexity regions using pfilt [28]. Disordered regions were derived directly from the existing InterPro disorder annotations. Gap regions which did not contain disorder annotations, coiled-coil or low complexity sequence were assigned given the length of the unassigned regions. These remaining gap regions were binned into size bins based on their lengths (see figure 1). The majority of gap regions are around 100 residues in length, as the typical structural domain size is around 100 residues 5 gap types were created to represent unassigned regions of various sizes which are approximate multiples of the typical domain size, see table 2. All non-domain regions: gaps, disordered, low complexity and coiled-coil regions were then compiled as a set of adjunct domain-like sequence regions to complement the PFAM domain assignments.

Building the word embedding

To build word2vec embeddings we treat protein sequences and their domain assignments as “sentences”. The Pfam IDs and other sequence region assignments are used as tokens/pseudo-words in such a pseudo-sentence. For instance, a typical protein may be converted to a sentence such as “PF00170 PF003534 G200 LowComplexity PF00678”. Which would indicate two leading Pfam domains followed by a gap region up to 200 residues, a region of low complexity sequence finally terminating in a Pfam domain (see figure 2). We compile such sentences for every Eukaryotic protein in InterPro62 and this set of sentences becomes the corpus we use to create the word embedding.

Python library gensim (https://radimrehurek.com/gensim/) was used to create the word2vec model from the corpus. The size parameter was set to 100, representing the dimensionality of the vector space to project the words in to. The minimum word count was set to 0, indicating that all words would be positioned in the vector space. This ensures that all domains, including important infrequent ones are considered, also the embedding uses the skip-gram algorithm and model to build the embedding. The goal of word2vec is to learn the weights in the hidden layer of a simple neural network, this hidden layer is an n by m matrix, where n is the number of input words in the corpus and m is the size parameter (*e.g.* 100). To train these weights the network is given a training task, the skip-gram task, which asks the network to predict, for each word in turn to output the probability that other words from the corpus are near to the input word (i.e. within a given window size, in this instance a window of 5). Once the training is complete the output probabilities are discarded and only the weights of the hidden layer are retained as this matrix is regarded as the word embedding. It is possible to develop alternative training tasks to learn the embedding matrix. A target behaviour of word2vec is that words which fulfil similar semantic roles should be near one another in the embedding and it is believed that the skip-gram task, by having the network learn about which words are local to one another, in turn is encoding this information in the weights of the hidden layer.

The embedding process is illustrated in full in figure 3. For the benchmark below an all-against-all distance matrix of domains was derived.

Benchmark

We are interested in whether word2vec embeds Pfam domains in a manner which is biologically meaningful. This would in turn would indicate that there is some manner of semantic meaning in the positioning or sequence context for protein domains. To investigate the embedding, initially we attempted to project the domain vectors into three dimensions (data not shown) using Multi-Dimensional Scaling. However, the resulting projection did not yield any trivially interpretable result.

An alternative means of investigating whether the embedding is biologically meaningful would be to establish if functionally related domains are placed near one another in the embedding. To investigate this, we assigned GO terms to the Pfam domains. This was done by allowing Pfam domains to inherit all GO terms assigned to the proteins each Pfam domain is observed in. Pfam domains inherit an average of 19.6 GO terms, although some domains may have upwards of 100 terms associated, see figure 4. Although this is somewhat imprecise, as GO annotations reflect protein functions rather than domain function, each domain's "bag" of GO terms will reflect the functional diversity for the contexts a domain is observed in. 2,358 GO terms were assigned over the 11,355 Pfam domains observed in the Eukaryotic proteins. These assignments could then be used for a nearest neighbour benchmark test.

Results

Nearest Neighbour Performance

Performance in nearest neighbour functional annotation was calculated to assess whether the vector embedding of domains displayed any meaningful structure. That is, domains with similar functionality were placed near one another in the embedding. Each domain was in turn considered by inheriting the GO terms from its k-nearest neighbours and comparing these predicted terms to the known terms assigned via InterPro annotations. Table 3 gives the precision and Mathew's Correlation Coefficients (MCC) scores for the nearest neighbour benchmark. The MCC value indicates the predicted terms are non-random (greater than 0) which in turn suggests that there is some meaningful structure in the embedding of domains in a vector space. Mean Accuracy is high and this is a consequence of there being a very large number of GO terms where typically only a few (relatively) are used to annotate any given protein or domain. This in turn means any given domain has very large numbers of True Negatives most of which are called correctly. As K is increased recall also increases as the bag of assigned terms gets very large but this comes at the cost of a sharply declining precision.

Word2vec is designed to embed human language words in a vector space such that words which occur in similar semantic contexts are close to one another in the vector space. That our domain embedding is non-random implies that multidomain proteins exhibit some form of semantic structure. That is, certain domains appear in contexts near or adjacent to other domains and it may be possible to learn grammar-like rules which govern this.

It is worth noting that increasing the number of neighbours (increasing K) from which functional roles can be inherited degrades performance in this function-annotation task. Domains are typically involved in a large number of possible different protein functions. By increasing the number of neighbours GO terms can be inherited from the number of false positives is greatly increased and so performance degrades.

Per Ontology Results

MCC values were also calculated for each of the three GO Ontologies (see table 4). Of the 2,358 GO terms used to annotate Eukaryotic sequences in InterPro: 1,018 are from the Molecular Function Ontology, 1,026 are from the Biological Process Ontology and 314 from the Cellular Component Ontology. The MCC values indicate different functional inheritance performance for each ontology with. In the context of the vector embedding this may imply that the simple syntax contained in the domain orderings contains some additional information about where a protein is located within the cell. Given the results of the previous CAFA experiment [16] it may, more simply, be that Cellular Component prediction is an easier task.

In general, we believe the MCC calculated may underestimate the quality of the domain embedding. Given the figures in Table 1 we see that nearly 70% of the proteins are gap regions. This indicates many domain assignments and domain types may be missing. We would expect with better domain coverage we would also have a more robust and biologically meaningful embedding.

Alongside this, using GO assignments to genes to annotate domains is inherently noisy. GO annotations may not be good descriptors of the specific role a domain plays in a given protein. For instance, GO:0051987 (Chaperone Binding), assigned to 92 PFAM domains, might be regarded as property or function of a whole protein rather than just a specific domain. An alternative issue is illustrated by Pfam domain PF00176 which is assigned both GO:0009916 (alternative oxidase activity) and GO:0001733 (galactosylceramide sulfotransferase activity).These assignments come via differing InterPro proteins but as are represent different catalytic reaction chemistries this domain is unlikely to possess both of these. Within the context of a multidomain proteins, domains provide specific sub-functionality such as providing catalytic sites, presenting one or more small molecule binding sites, providing membrane anchoring and so forth. It seems plausible if domains were annotated at a level, that better reflected these more specific sub-functional roles (rather than the protein's role), then the nearest neighbour assignment would return better results. The lack of a computer readable "domain ontology" remains a barrier for large scale studies of domain functionality and evolution.

Comparison to first order Markov representation

As sets of domains are sequences of symbols or states, it is possible to represent the information contained in the corpus of domain strings as a Markov process. We also investigated whether the word2vec domain embedding was a more robust representation of the information contained in the domain corpus than a first order Markov process. Parsing the corpus of proteins, a table of the transition probabilities of all domains against all domains was prepared. A given domain's immediate context can be read from the table as the rows give the probabilities of the following domain and columns indicate the probabilities of preceding domains. It follows that pairs of domains which share both similar row and column vectors are used in the same context in multidomain proteins. A distance matrix of Euclidean distances between all domains' vectors was prepared and the nearest neighbour assignment analysis was described above was performed, the results can be seen in Table 5. These results indicate that the word2vec domain embedding is substantially better at encoding the biological information contained in the corpus of multidomain proteins. The comparison may not be completely equivalent, Markov probabilities take in to account on the preceding symbol (or symbols in higher order chains) whereas the word2vec method considers a window of tokens around each domain, and this feature is likely a better match for modelling protein domain placement. Considering the incoming and outgoing probabilities for each domain could be considered equivalent to considering a window of 3 domains. The default window size for word2vec is 5. This comparison may under report the performance of a Markov process to model this data. However the corpus of multidomain proteins only contains a tiny fraction of the possible 3-mers and 5-mers of domains and with many unassigned regions it getting accurate probabilities may not be possible.

Vector arithmetic on the domain embeddings

One observation of semantic embeddings of natural languages is that arithmetic operations on the vectors frequently have semantic or lexical meanings, one classic example being:

> King – Man + Woman = Queen.

We wished to investigate if simple vector arithmetic or translations for the protein domain embedding might have similar lexical meaning.

In the King to Queen example (see figure 5), subtracting Man from King takes you to a space in the embedding with the meaning of man "removed" such that adding the Woman vector will take you to Queen. We can perform similar vector subtractions for the domain embedding. In this context we would treat a domain's set of GO terms as equivalent to its "meaning", although, as discussed, this is a lossy way to conceptualise the meaning of a domain. Nevertheless, if we subtract two domain vectors we would hope the third vector is in a space where the remaining set of GO terms is the set difference of the two domains.

We took the most common 20 Pfam domains, removing the one that isn't present in eukaryotes and in turn subtracted all possible domain vectors. For the resulting third vector we found the nearest domain and tested the GO term overlaps with the initial two domains. In nearly all cases the resulting domain has minimal GO term overlaps with its parents. It is clear that this operation moves us to a region in the vector space where the domains' "meaning" is profoundly altered, much as removing Man from King might be thought of as moving to a gender-neutral space. What is not clear is what is the functional meaning of this in protein domain terms.

To investigate whether we could find more meaningful movements in the vector space we looked instead for translations in the vector space between mutually exclusive binary annotations. King and Queen are typically used as mutually exclusive labels that straddle some conceptual binary assignment (*i.e.* gender) and much the same is true of many GO terms. For instance, in the Cellular Component Ontology annotation, terms such as Intracellular and Extracellular might be viewed as a similar mutually exclusive binary.

We chose three binary cellular component term pairs; Intracellular (GO:0005622) vs Extracellular (GO:0005615), Nucleus (GO: 0005634) vs Cytoplasm (GO: 0005737) and Cytoplasm (GO: 0005737) vs transmembrane (GO: 0009279). For each pairing we identified proteins with domains annotated exclusively with one term and not the other term. Then for the first term we calculated the vector which moves from the location of the domain with the first term to the closest domain annotated with the second term. As with the prior analysis not having a detailed domain ontology prevents us from knowing if this closest domain is the most appropriate domain to move to. This led to a population of translation vectors which we could test to measure if the translation from a domain with one term to a domain with the other term was always vector oriented in a similar direction. We compared all Intracellular to Extracellular vectors in an all against all fashion and did the same for the other two pairs of terms (see Figure 6). If the translation is preserved in the vector space, we would expect that all the vectors to have a small angle of deflection between them. In the transmembrane case there was no such alignment and no trend in the angles between the vectors. In both the Intracellular to Extracellular and the Nucleus to Cytoplasmic cases, there is a clear distribution which peaks around 1.5 radians, indicating that in general the translation is commonly orthogonal and isn't preserved in the vector space. This stands somewhat at odds with the prior observation that vector arithmetic which encodes semantic translations is a general property of these embeddings. The caveat to make here is that our embedding may not of high enough quality to perform this analysis productively. As noted above there may not be a enough domain coverage to robustly place the domains in the embedding space. Alternatively when choosing the domain pairs, the closest paired domain may not be the correct domain to calculate the angle between either we've selected the wrong extant domain or the correct domain is yet to be added to Pfam.

However, the intracellular to extracellular histogram shows a small leading tail below 1 radian (see figure 7) indicative of a small population of vectors which do approach alignment. And indeed, we are able to find small numbers of genes in InterPro which share Pfam domains and where the difference is a substitution of one or more intracellular annotated domains for extracellular domains. Two examples, such as G3I6X9 (solute carrier family 25 member 46) and A0A0L6WZ71 (glycogen debranching enzyme) or I3L0A0 (Human Transcript TMEM189-UBE2BV1) and G7Y5H3 (Ubiquitin-conjugating enzyme E2 L3), see figure 8. The first pair, G3I6X9 and A0A0L6WZ71, have respectively extracellular and intracellular functions. The second pair; G7Y5H3 has a cytoplasmic function but it is less clear what the role of I3L0A0 might be. The fact that this appears to work in some limited cases may suggest that an embedding based on a dataset with much greater domain coverage might be more accurate.

Domains of Unknown Function

As the word2vec embedding has some meaningful structure with regards GO term inheritance we can also use a nearest neighbour approach to suggest putative sets of GO terms that each eukaryotic Pfam Domain of Unknown Function (Pfam DUFs) may take part in. This allows a homology-free way to estimate GO assignments. Our corpus of eukaryotic genes contained annotations from 3,918 DUFs. Using a single nearest neighbour inheritance method, 1,292 of these domains could be assigned new GO terms (i.e. their nearest neighbour in the embedding was annotated and was not a gap or other sequence region). On average each DUF gets 11 novel GO terms assigned. Surveying the GO assignments, we note that the mean ontology depth for each assigned term (*i.e.* the shortest number of steps from an assigned term to the root of the ontology) is a depth on the graph 4.9 steps from the root of the ontology. The distribution of assigned term depths is also somewhat positively skewed (data not shown). The BP, MF and CC ontologies have maximum depths of 16, 16 and 11 respectively. This indicates that the typical term assignments are somewhat general, closer to generic terms such as ‘protein binding’ rather than terms which indicate explicit functional roles, such as catalytic mechanisms. In figure 9 the distribution of terms indicates that the majority of DUFs receive only a handful of putative GO assignments. We suggest that such assignments could be used as general starting points for Pfam domain annotations and with relatively fewer terms to confirm in most these shouldn’t make such annotation tasks more onerous or obfuscated. We make these annotations available (see http://bioinf.cs.ucl.ac.uk/downloads/word2vec) and note they could make a starting point for future annotation of these domains in Pfam.

Discussion

Applying word2vec to protein domains, making the assumption that multi-domain proteins are sentence-like, reveals that domains display some manner of semantic or lexical structure. Given this, it should be possible in future to elucidate statistical or semantic rules for domain placement in multi-domain proteins using grammatical inference methods. This would have applications in protein design and modelling.

The word2vec algorithm was designed to work over very large corpuses of human language, and whilst the nine million Eukaryotic InterPro sequences used in this study is a relatively large corpus, the corpus of “sentences” currently has too sparse a level of GO annotation to allow us to develop a high quality embedding of word-tokens which maps well to GO term defined function. A further limitation lies in the amount of domain coverage. Nearly 70% of the proteins remain unassigned to domains and without greater domain coverage a truly robust domain embedding may not be possible. Additionally, multi-domain proteins typically have fewer than six domains, and often just two or three, whereas human sentences comprise longer sequences. This may mean sequential sets of domains are unlikely to provide sufficient contextual information to produce an informative vector embedding. All these issues might be addressed by retuning the word2vec model to make it more appropriate for domain data. Word2vec offers several trainable parameters which may allow the method to be adapted for better performance with protein domains, however it may be the case that an entirely different architecture will be needed.

Using GO annotations to annotate domains is necessarily noisy. It is not clear that they are the best way to encode the lexical “meaning” of an isolated domain in its multi-domain context. In future, a finer grained annotation of domains’ sub-functional roles will be necessary to correctly interpret the lexical meaning of arithmetic transformations of vectors in the embedding space. Nevertheless, this work does open up the tantalising possibility that protein domains have contextual lexical meaning that could be learned and in turn could be used to derive rules for multidomain protein evolution. However, even in light of these limitations the vector embedding allows us to suggest preliminary function roles for many, as yet, unannotated Pfam domains, and combined with other sources of functional information, this could help improve our overall ability to assign functions to proteins and the genes which encode them.

Code & Data

All code is available on github and the domain assignments, genism model, token distance matrix and DUF assignments are available via our webserver
https://github.com/psipred/domain_word2vec_scripts
https://bioinfadmin.cs.ucl.ac.uk/downloads/word2vec/

References

1. Finn, R.D., et al., *InterPro in 2017-beyond protein family and domain annotations.* Nucleic Acids Res, 2017. **45**(D1): p. D190-d199.
2. Finn, R.D., et al., *The Pfam protein families database: towards a more sustainable future.* Nucleic Acids Res, 2016. **44**(D1): p. D279-85.
3. Mikolov, T.C.K.C., G, Dean, J, *Efficient Estimation of Word Respresentations in Vector Space.* arXiv, 2013.
4. Goldberg, Y.O., L, *word2vec Explained: Deriving Mikolov et al's Negative-Sampling Word-Embedding Method.* arXiv, 2014.
5. Asgari, E. and M.R. Mofrad, *Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics.* PLoS One, 2015. **10**(11): p. e0141287.
6. Yang, K.K., et al., *Learned protein embeddings for machine learning.* Bioinformatics, 2018. **34**(15): p. 2642-2648.
7. A. Viehweger, S.K., D. H. Parks, B. König, M. Marz, *An encoding of genome content for machine learning.* biorxiv, 2019.
8. Andreeva, A., et al., *SCOP2 prototype: a new approach to protein structure mining.* Nucleic Acids Res, 2014. **42**(Database issue): p. D310-4.
9. Cheng, H., et al., *ECOD: an evolutionary classification of protein domains.* PLoS Comput Biol, 2014. **10**(12): p. e1003926.
10. Dawson, N.L., et al., *CATH: an expanded resource to predict protein function through structure and sequence.* Nucleic Acids Res, 2017. **45**(D1): p. D289-d295.
11. Das, S. and C.A. Orengo, *Protein function annotation using protein domain family resources.* Methods, 2015.
12. Nepomnyachiy, S., N. Ben-Tal, and R. Kolodny, *Complex evolutionary footprints revealed in an analysis of reused protein segments of diverse lengths.* Proc Natl Acad Sci U S A, 2017. **114**(44): p. 11703-11708.
13. Friedberg, I., *Automated protein function prediction--the genomic challenge.* Brief Bioinform, 2006. **7**(3): p. 225-42.
14. Watson, J.D., R.A. Laskowski, and J.M. Thornton, *Predicting protein function from sequence and structural data.* Curr Opin Struct Biol, 2005. **15**(3): p. 275-84.
15. Loewenstein, Y., et al., *Protein function annotation by homology-based inference.* Genome Biol, 2009. **10**(2): p. 207.
16. Radivojac, P., et al., *A large-scale evaluation of computational protein function prediction.* Nat Methods, 2013. **10**(3): p. 221-7.
17. Consortium, G.O., *Expansion of the Gene Ontology knowledgebase and resources.* Nucleic Acids Res, 2017. **45**(D1): p. D331-d338.
18. Cozzetto, D., et al., *Protein function prediction by massive integration of evolutionary analyses and multiple data sources.* BMC Bioinformatics, 2013. **14 Suppl 3**: p. S1.
19. Lan, L., et al., *MS-kNN: protein function prediction by integrating multiple data sources.* BMC Bioinformatics, 2013. **14 Suppl 3**: p. S8.

20. Goldberg, T., et al., *LocTree3 prediction of localization.* Nucleic Acids Res, 2014. **42**(Web Server issue): p. W350-5.
21. Khan, I.K., et al., *The PFP and ESG protein function prediction methods in 2014: effect of database updates and ensemble approaches.* Gigascience, 2015. **4**: p. 43.
22. Almeida-e-Silva, D.C. and R.Z. Vencio, *SIFTER-T: a scalable and optimized framework for the SIFTER phylogenomic method of probabilistic protein domain annotation.* Biotechniques, 2015. **58**(3): p. 140-2.
23. Van Landeghem, S., et al., *Exploring Biomolecular Literature with EVEX: Connecting Genes through Events, Homology, and Indirect Associations.* Adv Bioinformatics, 2012. **2012**: p. 582765.
24. Falda, M., et al., *Argot2: a large scale function prediction tool relying on semantic similarity of weighted Gene Ontology terms.* BMC Bioinformatics, 2012. **13 Suppl 4**: p. S14.
25. Das, S., et al., *Functional classification of CATH superfamilies: a domain-based approach for protein function annotation.* Bioinformatics, 2016. **32**(18): p. 2889.
26. Fang, H. and J. Gough, *A domain-centric solution to functional genomics via dcGO Predictor.* BMC Bioinformatics, 2013. **14 Suppl 3**: p. S9.
27. Walsh, I., et al., *Comprehensive large-scale assessment of intrinsic protein disorder.* Bioinformatics, 2015. **31**(2): p. 201-8.
28. Jones, D.T., *Protein secondary structure prediction based on position-specific scoring matrices.* J Mol Biol, 1999. **292**(2): p. 195-202.

Figure Legends

Figure 1: Distribution of gap regions (regions without Pfam domain assignments) in InterPro Eukaryotic sequences.

Figure 2: The example of the domain and sequence region assignment. Pfam domains and disorder regions are derived from InterPro annotations. Low Complexity and Coiled Coil regions are calculated by Pfilt and gaps are assigned given their size.

Figure 3: Compiling protein “sentences”. InterPro compiles assignments of domains on Uniprot protein sequences. We take only the Pfam domain assignments the InterPro database stores and complement those with the assignments of Disorder and our own Low Complexity (LC) and coiled-coil (CC) region assignments. These are then tokenised to create a corpus of “sentences”. The corpus can then be used as input to word2vec. The output is a vector space which places each token at a point within that space, here stylised in 2D. Tokens which appear in similar syntactic contexts in the corpus should be placed near one another in the vector space.

Figure 4 distribution of GO term assignments

Figure 5: Example demonstrating semantically meaningful vector algebra. In A) four terms are placed in the vector space. If we subtract the **Man** vector from **King** (graph B), we move to an undefined point in the vector space. Adding the **Woman** vector (C) moves to the **Queen** vector.

Figure 6: Comparing translation vector from one binary GO property to another. A) Putative vector embedding of Intracellular (blue dots) and Extracellular (orange crosses) labelled domains. B) Vectors which translate each intracellular domain to its closest Extracellular labelled domain. C) Vectors are extracted and pooled D) Angle between each vector is compared to find vectors that point in the same direction.

Figure 7: Histogram of transformation vector angles. For intracellular to extracellular.

Figure 8: Diagram of intra/extra-cellular domain swaps. Both proteins share PFAM domain PF00179. In protein I3L0A0 domain PF10520 has been assigned the GO extracellular GO term (GO:0005615). In protein G7Y5H3 the substituted domains, PF014699 and PF14701, are both labelled with the intracellular GO term (GO:0005622)

Figure 9: Frequency of the number of GO terms assigned to DUFs

Table Legends

Table 1: Table of the total residue counts across the Eukaryotic Interpro protein set and the number of residues assigned to each class of domain or region

Table 2: Names and sizes of gap pseudo-domains and the number of interpro proteins where we observe at least one of these regions.

Table 3: Mean precision and accuracy and Mathew’s Correlation Coefficients given nearest neighbour inheritance of GO terms.

Table 4: MCC values for nearest neighbour inheritance of GO terms, calculated for each separate GO ontology.

Table 5: Comparison of MCC performance between 1st order Markov encoding and the word2vec embedding of the domain corpus

Class	Residue Count	Percentage
Total	5,001,517,961	-
Domains	1,256,832,058	25.1
Gaps	3,405,089,896	68.1
Disordered	167,103,753	3.3
Coiled Coil	3,309,167	0.06
Low Complexity	2,079,334	0.04

Table 1: Table of the total residue counts across the Eukaryotic Interpro protein set and the number of residues assigned to each class of domain or region

Gap Region ID	**Size (residues)**	**Protein Count**
G100	20-100	4,234,931
G200	101-200	2,635,225
G300	201-300	1,168,553
G400	301-400	575,517
G500	401- >500	926,673

Table 2: Names and sizes of gap pseudo-domains and the number of interpro proteins where we observe at least one of these regions.

k-Nearest Neighbours	**Mean Precision**	**Mean Recall**	**Mean Accuracy**	**Mean MCC**
1	0.33	0.30	0.99	0.28
3	0.23	0.42	0.98	0.28
5	0.18	0.49	0.98	0.26
10	0.12	0.57	0.96	0.23

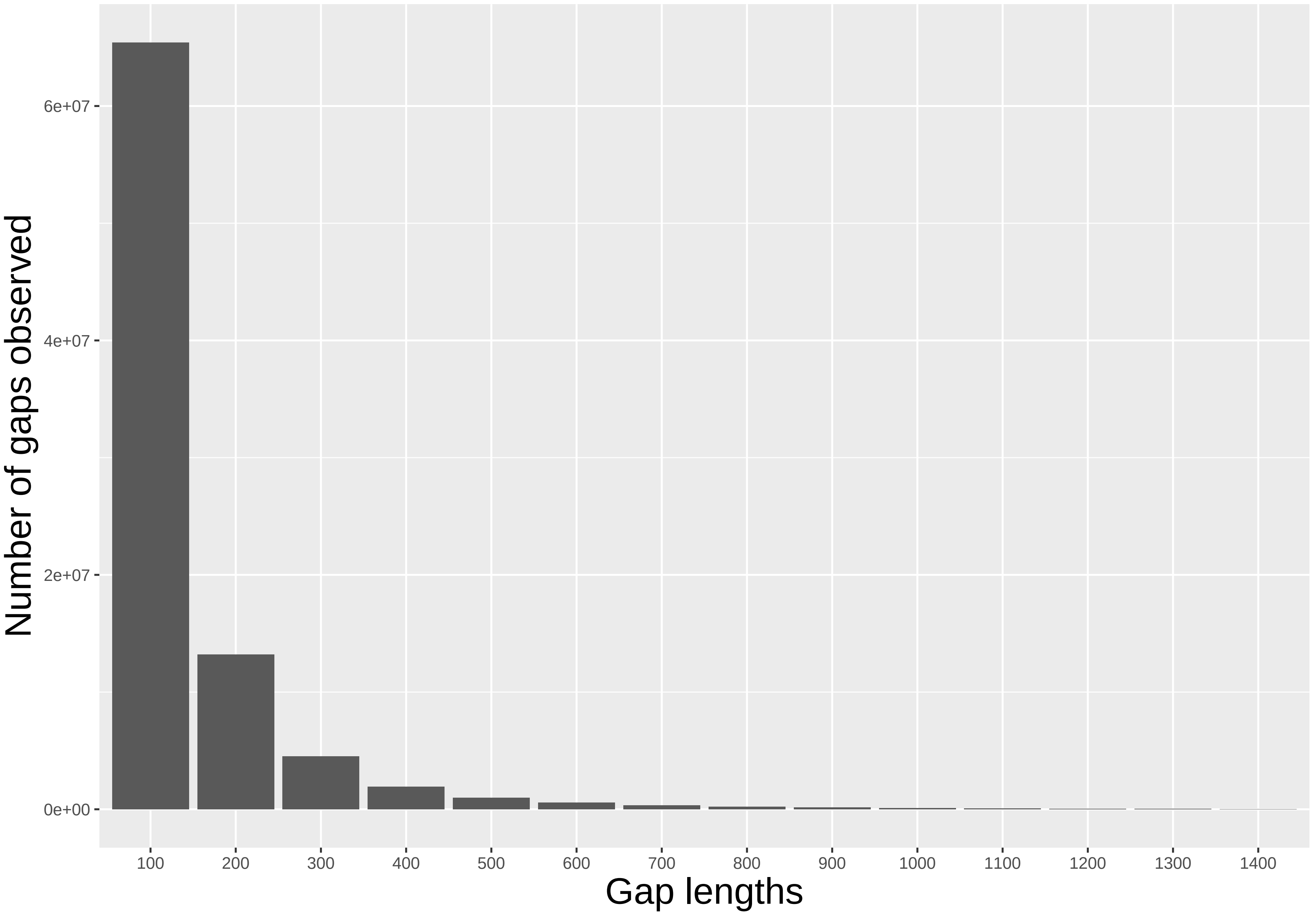
Table 3: Mean precision and accuracy and Mathew’s Correlation Coefficients given nearest neighbour inheritance of GO terms.

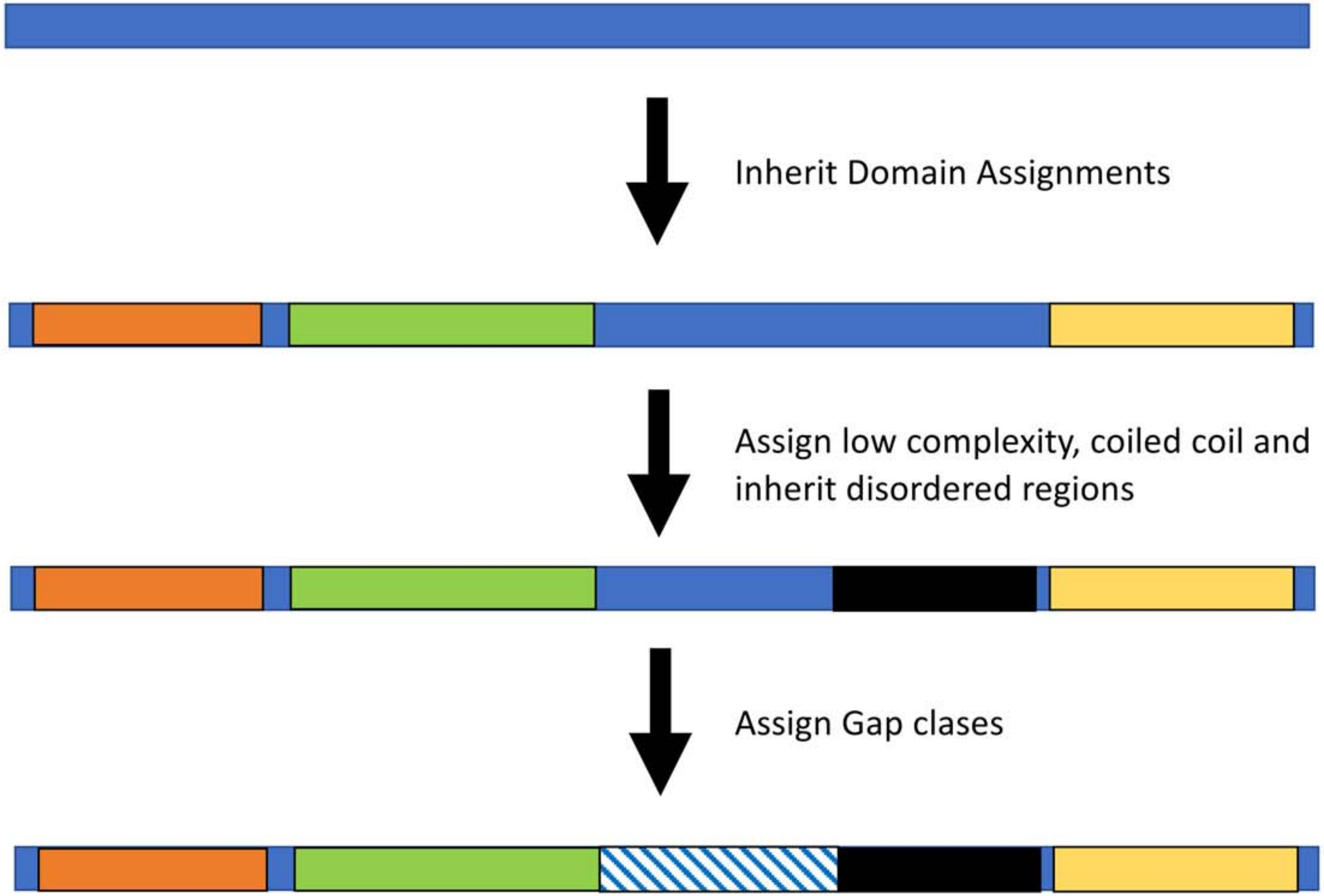
	k			
Ontology	1	3	5	10
Biological Process	0.27	0.20	0.19	0.17
Molecular Function	0.30	0.23	0.22	0.19
Cellular Component	0.33	0.22	0.22	0.20

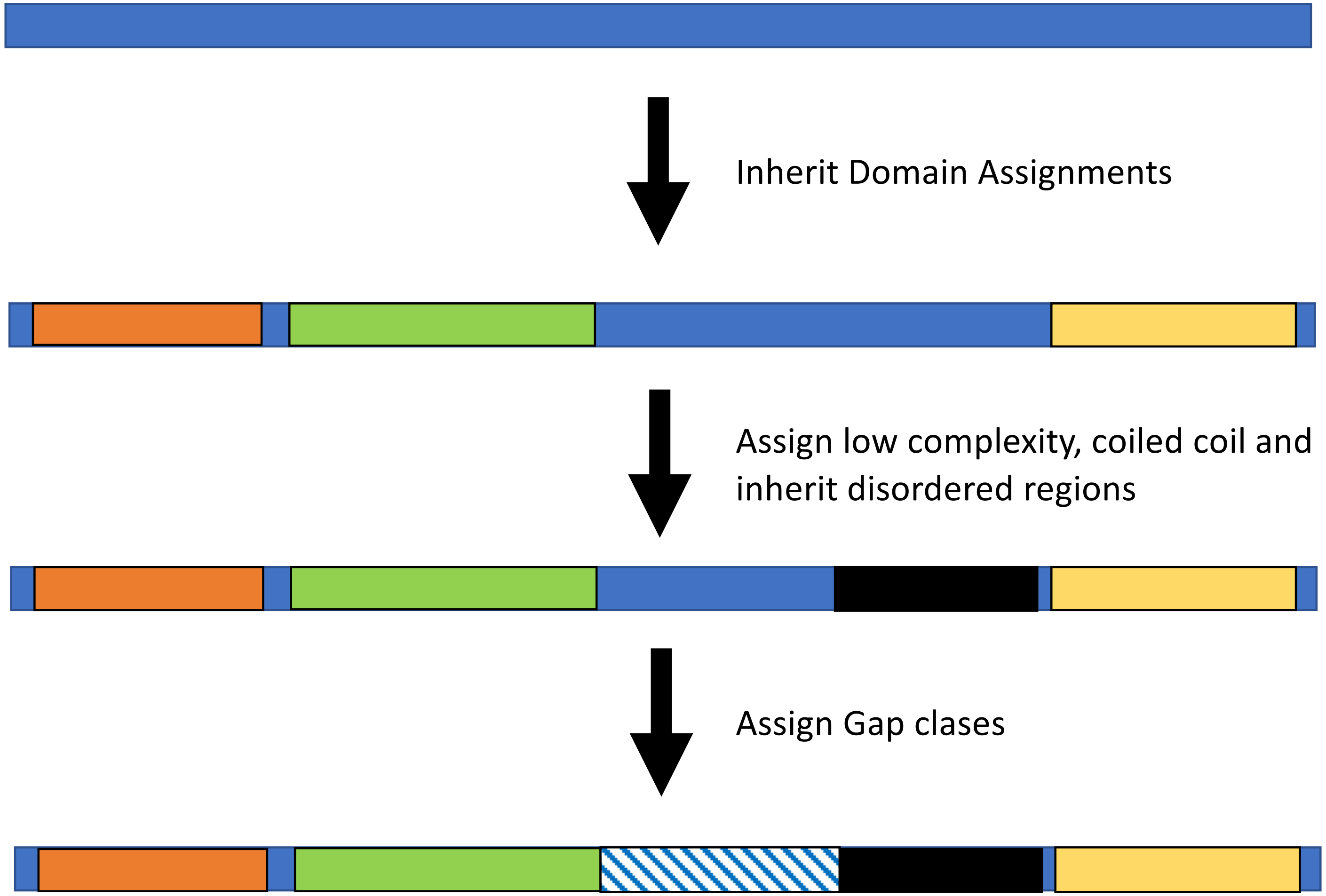
Table 4: MCC values for nearest neighbour inheritance of GO terms, calculated for each separate GO ontology.

	Mean MCC	
k-Nearest Neighbours	**Word2vec**	**Markov**
1	0.28	0.13
5	0.28	0.14
5	0.26	0.14
10	0.23	0.11

Table 5: Comparison of MCC performance between 1st order Markov encoding and the word2vec embedding of the domain corpus


Number of gaps observed
6e+07
4e+07
2e+07
0e+00
100
200
300
400
500
600
700
800
900
1000
1100
1200
1300
1400
Gap lengths


Inherit Domain Assignments
Assign low complexity, coiled coil and inherit disordered regions
Assign Gap clases

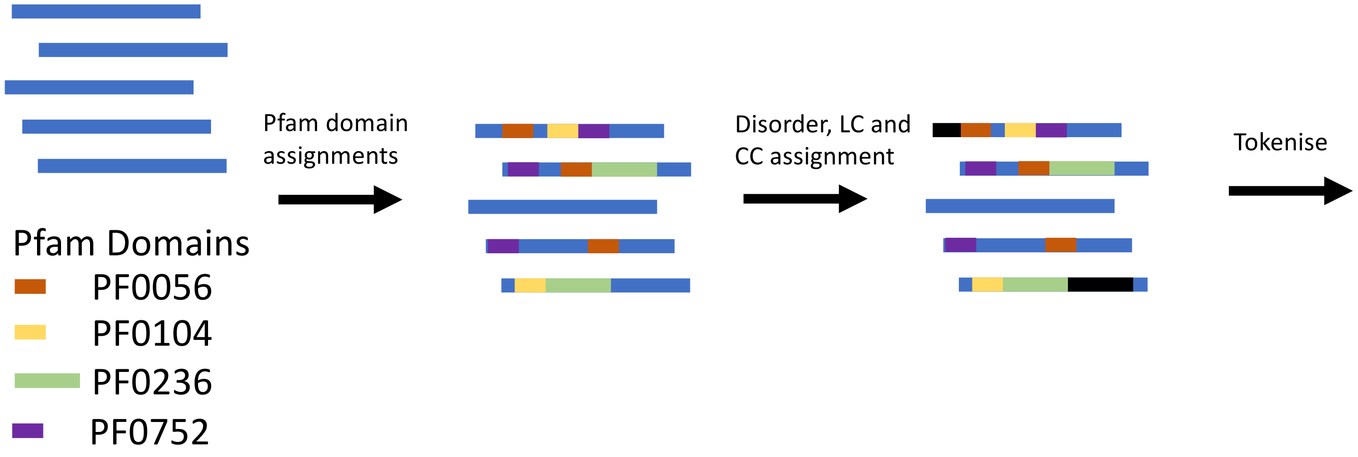


Inherit Domain Assignments

Assign low complexity, coiled coil and
inherit disordered regions

Assign Gap clases

Uniprot Eukaryotic proteins



Pfam domain assignments

Disorder, LC and CC assignment

Tokenise

Pfam Domains

- 🟧 PF0056
- 🟨 PF0104
- 🟩 PF0236
- 🟪 PF0752

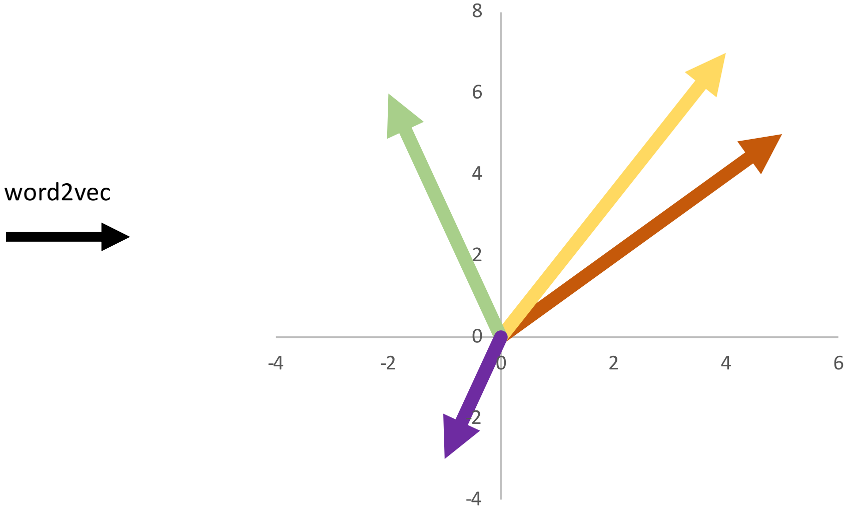
Corpus

- DISORDER PF0056 GAP100 PF0104 PF0752 GAP300
- PF0752 GAP100 PF0056 PF0236 GAP200
- GAP500
- PF0752 GAP300 PF0056 GAP200
- PF0104 PF0236 DISORDER

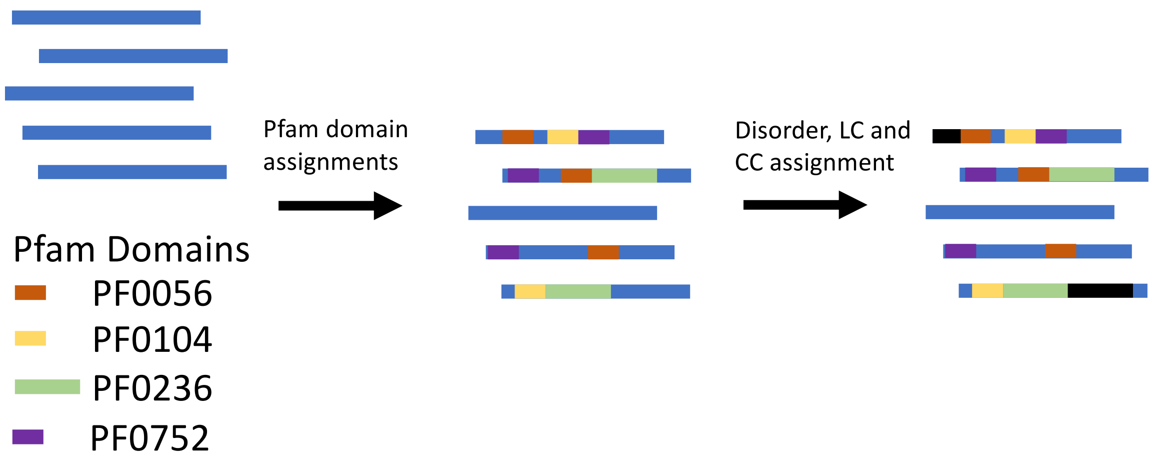
Corpus

- DISORDER PF0056 GAP100 PF0104 PF0752 GAP300
- PF0752 GAP100 PF0056 PF0236 GAP200
- GAP500
- PF0752 GAP300 PF0056 GAP200
- PF0104 PF0236 DISORDER

word2vec



Uniprot Eukaryotic proteins


Pfam domain assignments
Disorder, LC and CC assignment
Tokenise

Pfam Domains
- 🟧 PF0056
- 🟨 PF0104
- 🟩 PF0236
- 🟪 PF0752

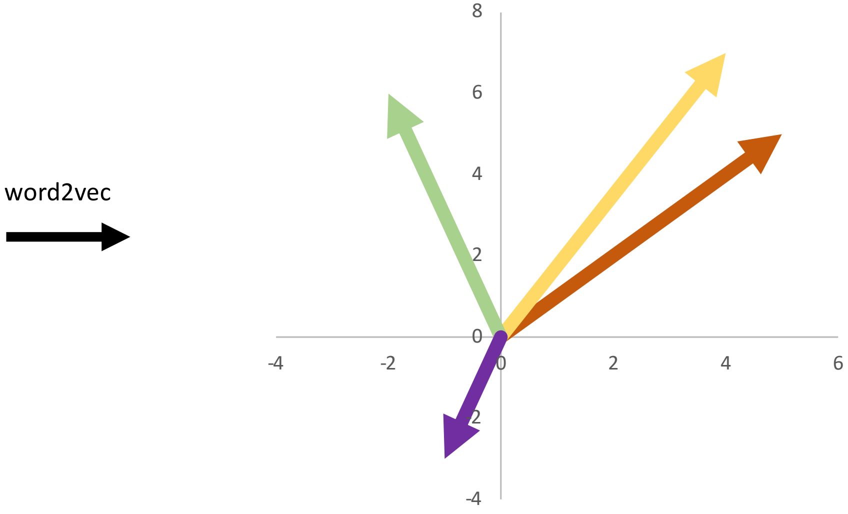
Corpus

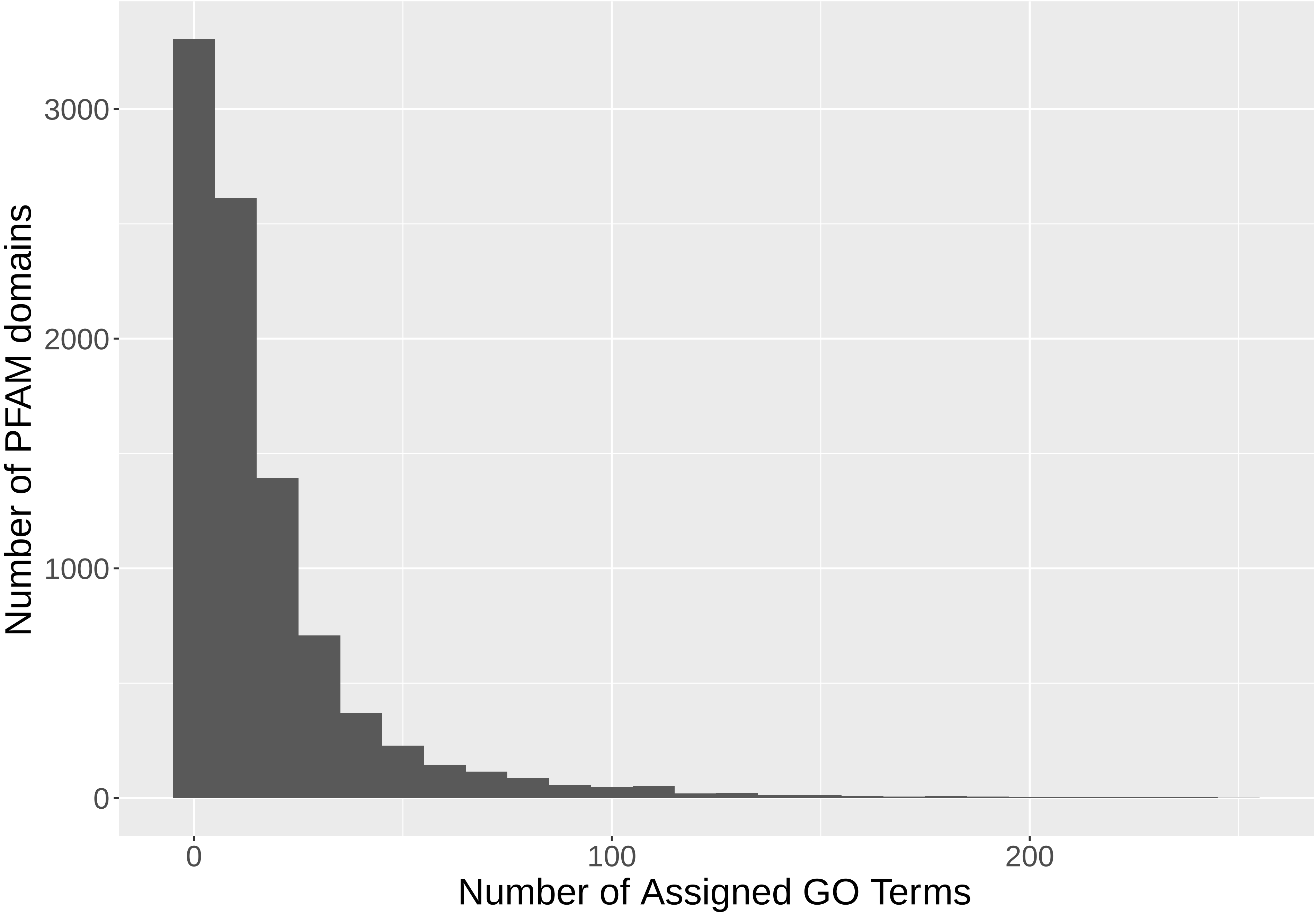
- DISORDER PF0056 GAP100 PF0104 PF0752 GAP300
- PF0752 GAP100 PF0056 PF0236 GAP200
- GAP500
- PF0752 GAP300 PF0056 GAP200
- PF0104 PF0236 DISORDER

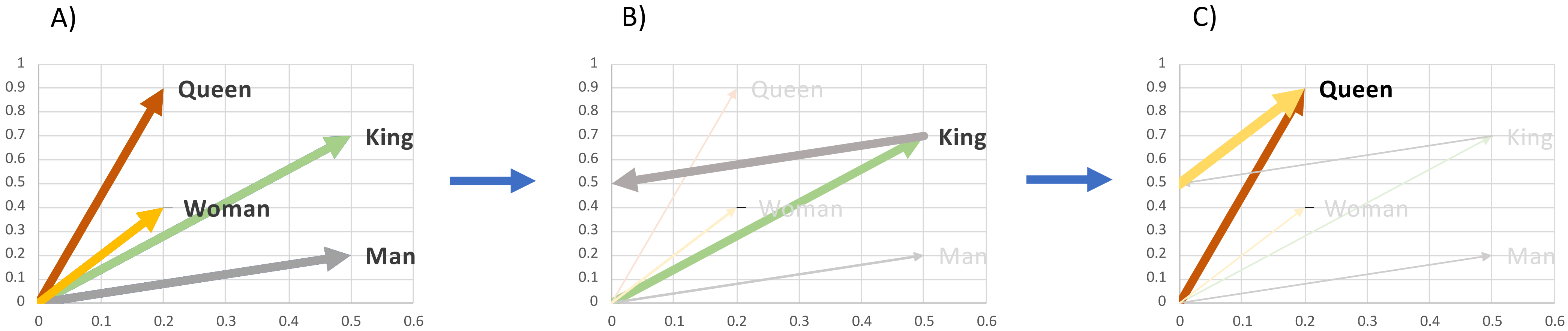
Corpus

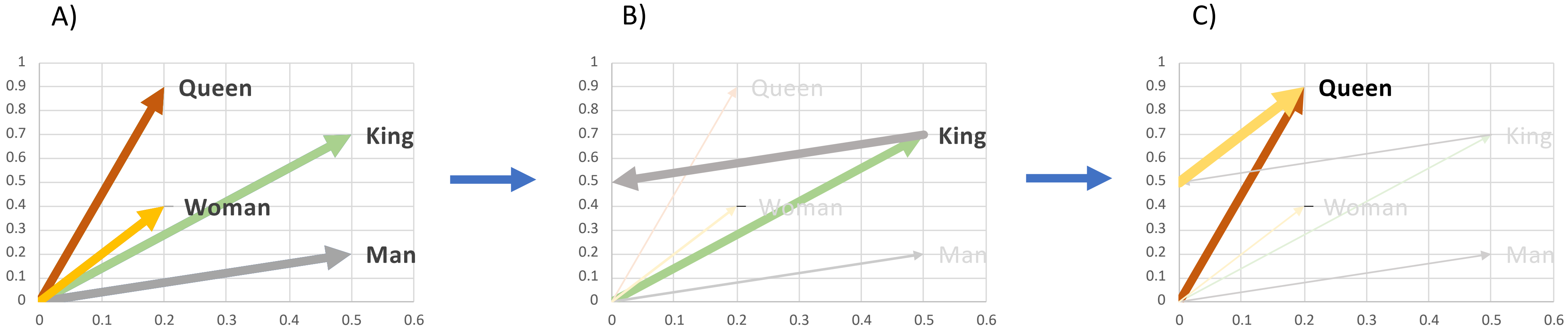
- DISORDER PF0056 GAP100 PF0104 PF0752 GAP300
- PF0752 GAP100 PF0056 PF0236 GAP200
- GAP500
- PF0752 GAP300 PF0056 GAP200
- PF0104 PF0236 DISORDER

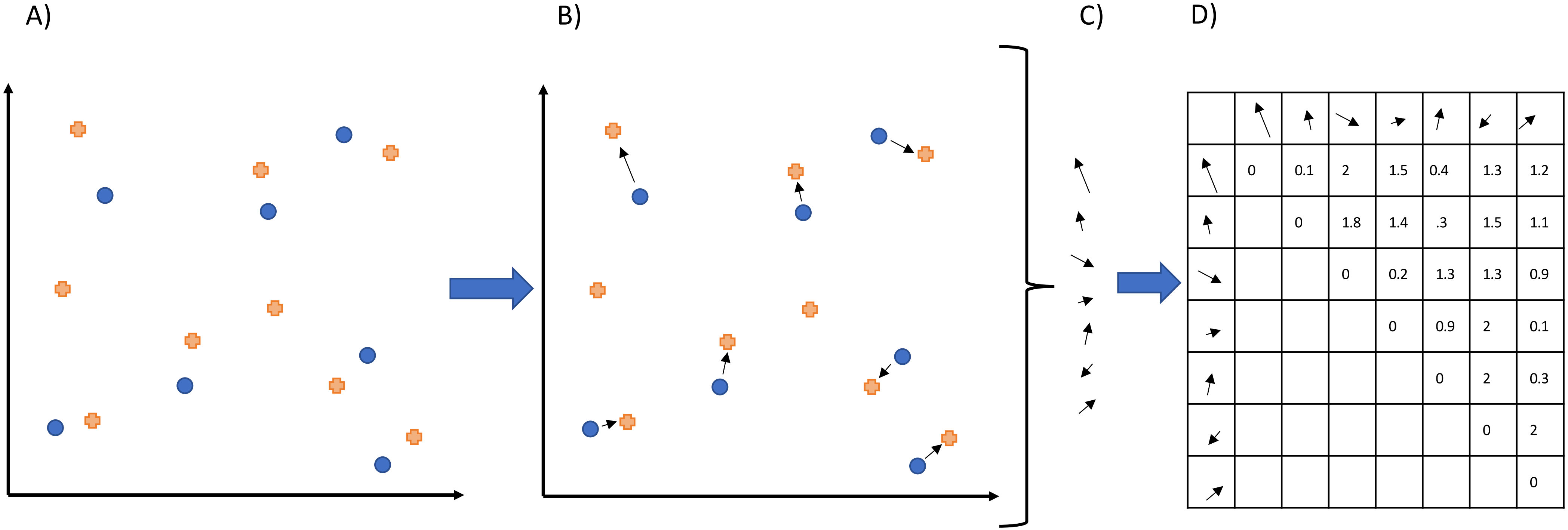
word2vec


8
6
4
2
0
2
-4
-4
-2
0
2
4
6

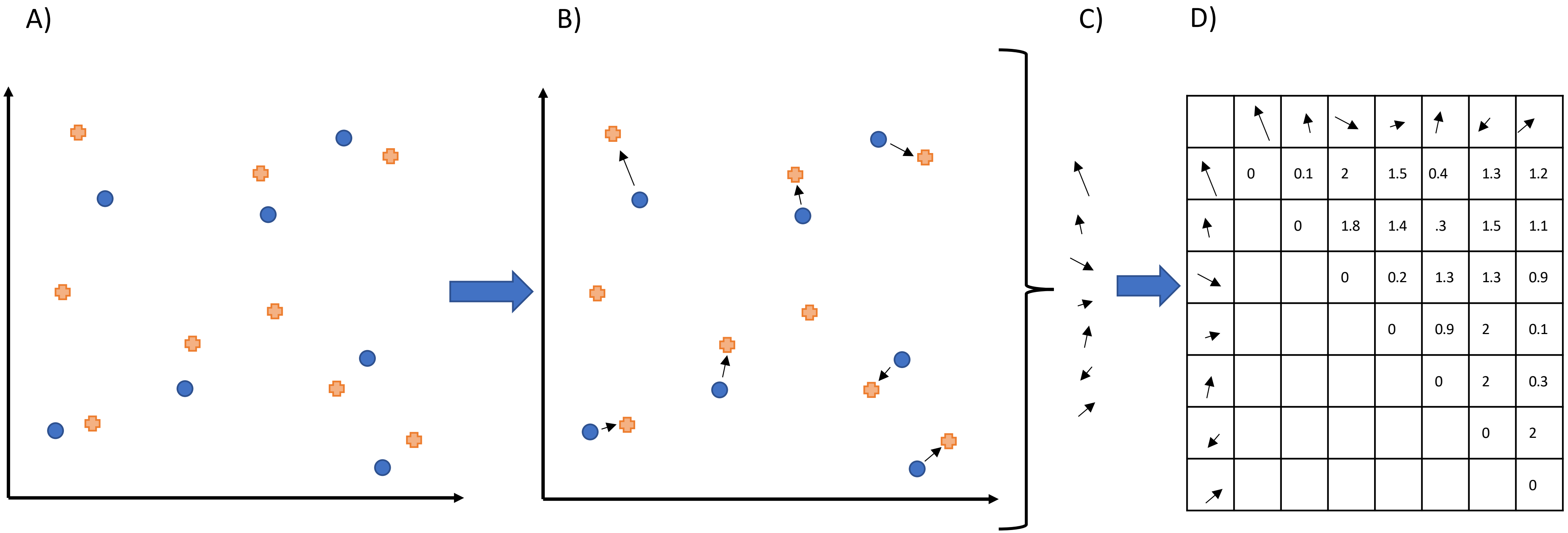

Number of PFAM domains
3000
2000
1000
0
0
100
200
Number of Assigned GO Terms


A)
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.1
0.2
0.3
0.4
0.5
0.6
Queen
King
Woman
Man
B)
Queen
King
Woman
Man
C)
Queen
King
Woman
Man

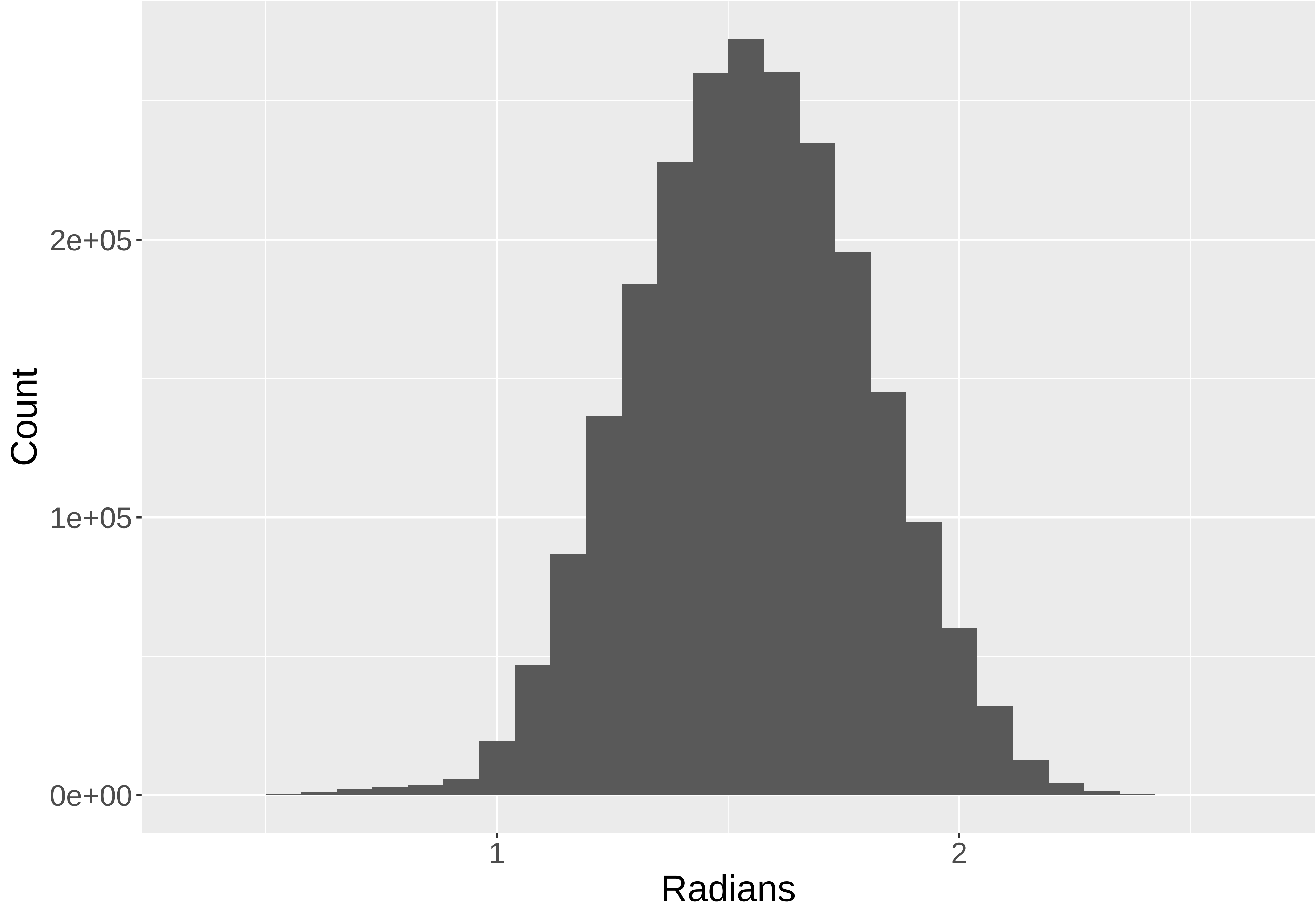

A)
B)
C)
Queen
King
Woman
Man
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

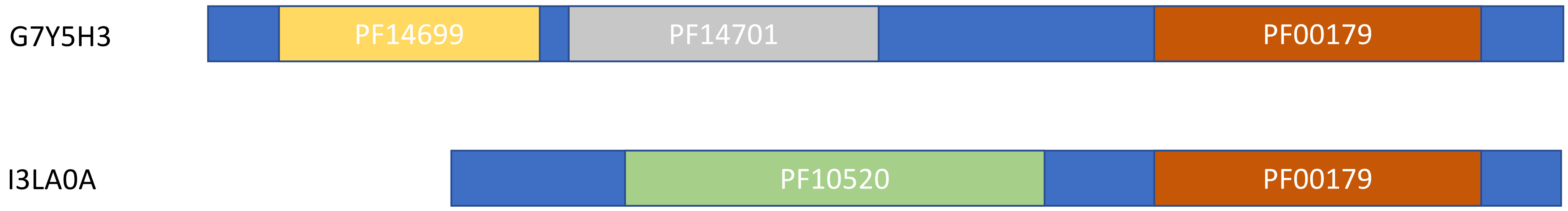


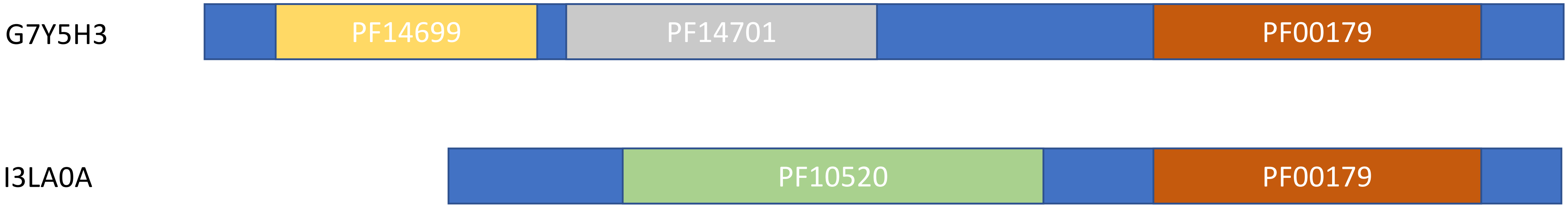
	↖	↑	↘	→	↑	↙	↗
↖	0	0.1	2	1.5	0.4	1.3	1.2
↑		0	1.8	1.4	.3	1.5	1.1
↘			0	0.2	1.3	1.3	0.9
→				0	0.9	2	0.1
↑					0	2	0.3
↙						0	2
↗							0

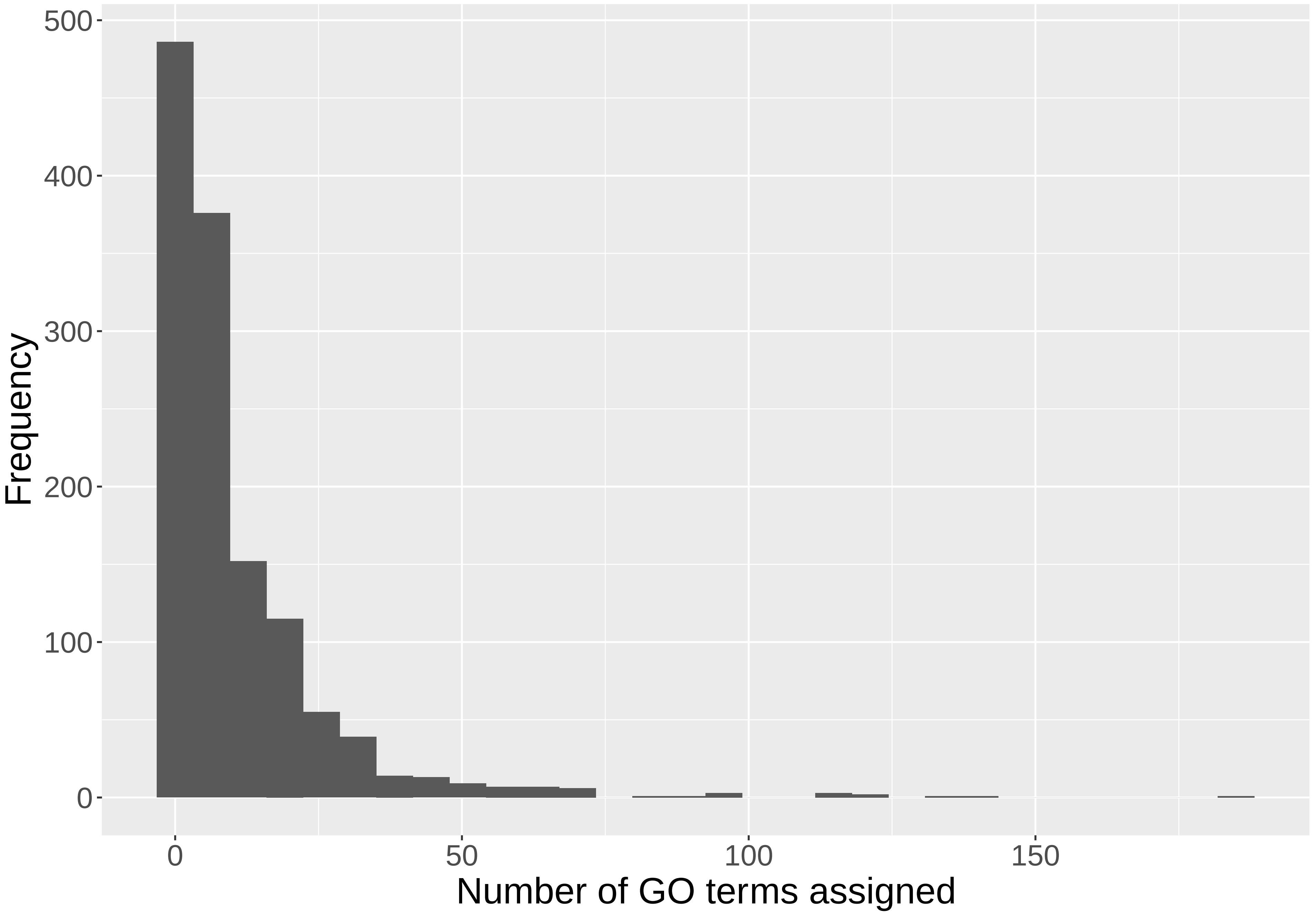


	↖	↑	↘	→	↑	↙	↗
↖	0	0.1	2	1.5	0.4	1.3	1.2
↑		0	1.8	1.4	.3	1.5	1.1
↘			0	0.2	1.3	1.3	0.9
→				0	0.9	2	0.1
↑					0	2	0.3
↙						0	2
↗							0


Count
2e+05
1e+05
0e+00
1
2
Radians


G7Y5H3
PF14699
PF14701
PF00179
I3LA0A
PF10520
PF00179


G7Y5H3
PF14699
PF14701
PF00179
I3LA0A
PF10520
PF00179


Frequency
500
400
300
200
100
0
0
50
100
150
Number of GO terms assigned