Machine learning models for automatic Gene Ontology annotation of biological texts*

Jayati H. Jui[0000−0002−9718−6387] and Milos Hauskrecht[0000−0002−7818−0633]

University of Pittsburgh, Pittsburgh, PA 15260, USA
{jaj146,milos}@pitt.edu

Abstract. Gene ontology (GO) is a major source of biological knowledge that describes the functions of genes and gene products using a comprehensive set of controlled vocabularies or terms organized in a hierarchical structure. Automatic annotation of biological texts using gene ontology (GO) terms gained attention of the scientific community as it helps to quickly identify relevant documents or parts of text related to specific biological function or process. In this paper, we propose and investigate a new GO-term annotation strategy that uses a non-parametric k-nearest neighbor model that relies on various vector-based representations of training documents and GO-terms linked to these documents. Our vector representations are based on machine learning and natural language processing (NLP) models that include singular value decomposition, word2vec and topics-based scoring. We evaluate the performance of our model on a large benchmark corpus using a variety of standard and hierarchical evaluation metrics.

Keywords: Gene Ontology (GO) · GO-term text annotation.

1 Introduction

Gene Ontology (GO) is the largest and most diverse open-source repository of structured and standardized vocabulary that describes complex biological functions of genes and gene products across different organisms. The GO knowledge base is developed and maintained by the Gene Ontology (GO) Consortium. It defines vocabulary and its structure using functional attributes known as GO terms, and links these to different genes and gene products. GO ontology can be used for a variety of purposes. One important problem is the annotation of documents or text with GO terms which can help researchers to identify articles based on important biological relations mentioned in the articles.

The early GO annotation efforts of text were based on manual annotations. Unfortunately, such annotations were time-consuming and required well-established guidelines to avoid inconsistencies and errors [5,3]. The focus of recent

*Supported by the Defense Advanced Research Projects Agency (DARPA) through Cooperative Agreement D20AC00002 awarded by the U.S. Department of the Interior, Interior Business Center. The content of the article does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

GO-annotation effort has been gradually shifting towards automatic methods based on Natural Language Processing (NLP) and machine learning (ML) solutions. BioCreAtIvE text mining competitions were among the first attempts to design solutions to facilitate automatic annotations of genes and their products [2,8]. Different methods have been devised. These span pattern-based approaches that fit text to predefined patterns with specific keywords matches driving the annotation to more advanced machine learning models relying on gene concepts and language based features [4,6,7].

In this work, we develop and explore GO-term annotation solutions of biological text that rely on the state-of-the-art NLP and ML techniques. Briefly, the annotation problem can be seen as a supervised multi-label classification problem with GO terms defining the class labels. To predict the labels we rely on non-parametric methods where documents are featurized and represented using various NLP vector-based models: Singular value decomposition (SVD), Word2Vec, and topic-based models in which classes are modeled as collections of class attributes or topics. To evaluate our solution we used a benchmark dataset with article abstracts and their GO annotations.

2 Methods

Corpus: We have created a benchmark corpus with the latest GO annotations for our model training and assessment. Using the Gene Ontology Annotation (GOA) database from Uniprot[1] (Uniprot-GOA), we retrieved all human GO annotations with references to PubMed articles. After updating the old Uniprot-GOA annotations with the latest GO functional attributes, our final corpus had ∼42k articles with 14707 unique GO annotations. We randomly split the dataset into the disjoint train and test sets with a 90:10 ratio resulting in ∼38k train and ∼4k test documents. The distribution of the three GO categories in the dataset are available in a supplementary document[2]. The dataset used in this study is available on GitHub[3].

Text Processing and Vectorization: We performed text processing of all documents in the corpus using the **scyspaCy**[4] python package built for biomedical, clinical, and scientific text analysis. We utilized scispaCy's "en_core_sci_md" model to conduct Named Entity Recognition (NER). We removed all words from the documents that were not recognized as NER entities in order to shorten the documents and computed Term frequency-inverse document frequency (TF-iDF) of each document. Using document TF-iDFs, we computed two vector representations of each document: SVD and Word2Vec. SVD is a popular dimensionality reduction technique for data with a large number of features. We computed 100-dimensional SVDs of the sparse document TF-iDFs such that maximum variation is captured within the first 100 components. Pre-trained Word2Vec word embeddings were extracted directly from scispaCy's "en_core_sci_md" model.

[1] https://www.ebi.ac.uk/GOA/index
[2] https://github.com/juijayati/GOA-AIME2023.git
[3] https://github.com/juijayati/GOA-AIME2023.git
[4] https://allenai.github.io/scispacy/

Word2Vec vectors were weighted using the Tf-iDF weights of the words to generate document embeddings.

Prediction model: To label the text, we employ a non-parametric method that models the relationships between documents and GO terms with the help of documents' vector representations. More specifically, vector representations of documents in the training data and their associated GO-labels are used to make prediction on the test documents using the k-nn approach applied to their vector representations. Our method can be summarized as follows :

For a test article Q

- Compute vector representations of Q and assign topics to Q
- Extract k most similar documents from the training set using the k Nearest Neighbor (k-NN) strategy.
- Build a set of GO terms $\mathcal{G}$ by combining all GO annotations from the top-k articles extracted from the training set.
- Calculate document-based similarity score $\phi_D(Q,t)$ and topic-based similarity score $\phi_T(Q,t)$ for each GO-term $t \in \mathcal{G}$
- For each term t in $\mathcal{G}$, calculate annotation likelihood of term t given Q as:

$$l(t|Q) = \phi_D(Q,t) * \phi_T(Q,t) \tag{1}$$

- Annotate Q with top n terms in $\mathcal{G}$ based on the highest likelihoods.

Topic Assignment: For assigning topics to a query article, we used scispaCy's "Gene Ontology" linker that links NER entities to a set of UMLS[5] concepts related to GO functional attributes. The topics of the test articles were determined by direct mapping of the UMLS concept names to GO terms.

Document-based score: Document-based similarity score for a GO term $t \in \mathcal{G}$ given a query article Q is calculated using the documents in the list K_Q of top-k most similar documents that annotates t.

$$\phi_D(Q,t) = \left(1 + \sum_{d \in K_Q : t \text{ annotates } d} sim(Q,d)\right)^2 \tag{2}$$

Topic-based score: Topic-based similarity score for a GO term $t \in \mathcal{G}$ given a query article Q is calculated based on the maximum semantic similarity between the term t and any topics of Q that is in the same path of the GO hierarchy as the term t. The semantic similarity between two go terms is defined as:

$$sim(t_1,t_2) = \begin{cases} 1 & \text{if } t_1 = t_2 \\ \frac{1}{dist(t_1,t_2)} & \text{if } t_2 \in ancestors(t_1) \cup children(t_1) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $dist(t_1,t_2)$ denotes the semantic distance between the terms t_1 and t_2 and is defined as the shortest distance between t_1 and t_2 in GO hierarchy. The

[5] https://www.nlm.nih.gov/research/umls/index.html

Table 1. Model performances on the benchmark test corpus

Vectorization	Vector Dimension	GO Scoring	R_{10}	TREC	BioCreAtIve		
				MMR_{10}	hP_{10}	hR_{10}	hF_{10}
SVD	100	Doc	0.41	0.44	0.19	0.65	0.25
Word2Vec	200	Doc	0.43	0.45	0.19	0.67	0.26
SVD	100	Doc + Topic	0.43	0.45	**0.21**	0.66	0.27
Word2Vec	200	Doc + Topic	**0.45**	**0.46**	**0.21**	**0.69**	**0.28**

topic-based score is then calculated to reflect the maximum semantic similarity between a candidate GO term t and the topic set T_Q of a query Q.

$$\phi_T(Q,t) = \left(1 + \max_{t_Q \in\ T_Q} sim(t, t_Q)\right)^2 \tag{4}$$

3 Results and Discussion

We evaluated our models on the test corpus consisting of 4034 articles, and 4191 unique GO annotations among which 343 annotations were not present in the training corpus. For evaluation, we used evaluation metrics developed for hierarchical biological ontologies. In particular, we considered Mean Reciprocal Rank (MRR_n) used in TREC question answering track and hierarchical measures of precision (hP_n), recall (hR_n) and F-scores (hF_n) introduced at BioCreAtIve IV competition [9,1]. We also considered Recall at rank n (R_n) that measures the exact recall achieved by the model's top n predictions. The detailed explanations of the evaluation metrics are available in a supplementary document[3].

The classification performance of the proposed machine learning models are summarized in Table 1. All statistics were based on the top 10 GO terms predicted by the models. As can be seen from the results, the Word2Vec model combined with document and topic-based GO terms scoring achieved best performance across all five evaluation metrics. It is interesting to see that applying a dimensionality reduction technique like SVD on the TF-iDFs was able to achieve comparable performance to Word2Vec models. In contrast to TF-iDF or SVD, Word2Vec captures the context of words and the semantic relationship between words. We note that scispaCy's word vectors were trained on biomedical and clinical corpora and offers vector representations of key biological words and concepts. Since the corpus introduced in this study is built using biological texts, the Word2Vec models provide a better representation of the articles than term frequencies. The Word2Vec models also strictly outperform SVD models except for hierarchical precision metric. Furthermore, it can be seen that incorporating topic-based similarity scores in addition to document-based similarities enabled improved scoring of the gold standard GO terms. It shows that a rough set of terms related to the actual protein functions can be identified via direct word mentions or textual cues from the NER entity tokens.

Both SVD and Word2Vec models achieved high hierarchical recall (hR_{10}) on the training data. This indicates that the ancestor sets of the predicted terms and

the true annotations have a high overlap. Higher hR_{10} were achieved by topic-based models because topic-based similarity scores prioritize semantic similarity between two terms with ancestor-descendent relationship. However, the hierarchical precision of the models remained very low. Hierarchical precision favors predictions of more general GO terms with fewer ancestors. This is contradictory to providing the most specific terms for annotation it a poor metric for such ontologies. According MRR_{10}, the first prediction of a true annotation occurs within the top three predicted terms. Finally, 45% of the true annotations were typically included in the top-10 predictions, as indicated by the R_{10} statistics. Additional results regarding the performance of the top Word2Vec model across three GO categories are available in a supplementary document[3].

4 Conclusions

We have proposed and investigated an automated approach for the annotation of biomedical articles with GO terms that represent molecular functions, underlying biological processes, and cellular components mentioned in the text. The annotation of a test article uses k-nearest neighbor matching of training articles using their vector representation. In the future, we plan to investigate additional modern text vectorization methods offered, for example, by BERT or ELMO architectures for biological domains, as well as featurization based on gene or gene products mentioned in the articles.

References

1. Arighi, C., Cohen, K., Hirschman, L., Lu, Z., Tudor, C., Wiegers, T., Wilbur, W., Wu, C.: Proceedings of the fourth biocreative challenge evaluation workshop (2013)
2. Blaschke, C., Leon, E.A., Krallinger, M., Valencia, A.: Evaluation of biocreative assessment of task 2. BMC bioinformatics **6**, 1–13 (2005)
3. Camon, E.B., Barrell, D.G., Dimmer, E.C., Lee, V., Magrane, M., Maslen, J., Binns, D., Apweiler, R.: An evaluation of go annotation retrieval for biocreative and goa. BMC bioinformatics **6**, 1–11 (2005)
4. Chen, Y.D., Yang, C.J., Li, W.G., Huang, C.Y., Chiang, J.H., et al.: Gene ontology evidence sentence extraction and concept extraction: Two rule-based approaches (2013)
5. Faria, D., Schlicker, A., Pesquita, C., Bastos, H., Ferreira, A.E., Albrecht, M., Falcão, A.O.: Mining go annotations for improving annotation consistency. PloS one **7**(7), e40519 (2012)
6. Gobeill, J., Pasche, E., Vishnyakova, D., Ruch, P.: Closing the loop: from paper to protein annotation using supervised gene ontology classification. Database **2014** (2014)
7. Lena, P.D., Domeniconi, G., Margara, L., Moro, G.: Gota: Go term annotation of biomedical literature. BMC bioinformatics **16**, 1–13 (2015)
8. Lu, Z., Hirschman, L.: Biocuration workflows and text mining: overview of the biocreative 2012 workshop track ii. Database **2012** (2012)
9. Voorhees, E.M., Buckland, L.: Overview of the trec 2003 question answering track. In: TREC. vol. 2003, pp. 54–68 (2003)