RESEARCH ARTICLE

Gene extraction from Biomedical Literature

Kevin E. Meza[1,†*], José K. González[1,†], Victor E. Nieto[1,†], Diego A. Camacho[1,†] and Carlos F. Méndez[2,^]

*Correspondence:
kevinml@lcg.unam.mx
[1]Center for Genomic Sciences, UNAM, University Avenue, Cuernavaca, Morelos, Mexico
[†]0
Full list of author information is available at the end of the article
[†]All the authors contributed equally for this project ^Principal Investigator

Abstract

The extraction of knowledge in biomedical literature has become an important task in the latest years, because it settles a milestone for new experimental research projects. Here we present a bioinformatic approach for the extraction of human gene names within biomedical abstracts, making use of a supervised Machine Learning model: Conditional Random Fields (CRFs). The model we describe has an F1-score average of 0.8638 (precision: 0.942 , recall: 0.797) using a training set of .70 and a test set of .30 (out of 455 tagged abstracts). During the development, we could see that the performance of the CRF dramatically can improve as the data set is increased. We also determined the cross validation and iteration numbers, we tuned some hyperparameters and changed the features that the CRF will take into account for tagging the wanted data in order to get the highest possible score.

Keywords: Machine Learning; supervised learning; BioNLP; CRF

Background

New original biomedical research is constantly being published in journals at rates that exceed human capability of reading and analysing it. Various computational models have been used throughout the years to retrieve precise and specific information from this enormous text compilations. These models can perform a wide variety of tasks including topic classification, abstract analysis, and accurate knowledge extraction, such is the case of Machine learning, which is a discipline in the field of Artificial Intelligence that is extensively used for this purposes that can be supervised or unsupervised. In unsupervised machine learning, the classes or groups inside the dataset given to the algorithm are unknown, the aim of this approach is to train a model that finds itself patterns in the characteristics of the information given and proceeds to cluster it. Often useful when not knowing too much information about your data.

In a supervised approach, the words in the training dataset are associated with a class and the resulting model is able to predict the class of the new words given in the testing dataset. A classificator can be multiclass or binari, depending on the amount of categories in which the model has to classificate the words, in this work we present a binary classifier that distinguishes between gene or non gene word types. For the article purpose, we used a conditional model (CRF) that calculates the probabilities of a potential label sequence conditioned to a sequence of observations. The observations poses arbitrary characteristics that are essential for the probability calculation. In these model the information of other observations can be included apart from the ones of the present word, that is, the characteristics of the next or previous observations is considered as well, of course this idea can be extended to as many observations as desired and taking account of specific characteristics.

Methods

The statistical machine learning aspect of our approach for named entity recognition involves learning a statistical model using the CRF framework, which are probabilistics models to segment and label sequence data. The model represents a relationship between features (words in this case) and name tags associated with entities (GENE), basically the process consists in identifying the name tags associated with the given features based on the model, and also based in the features of near words to find a sequence of tags that maximizes the conditional distribution. The features are observations related to the entities and the parameters of the model represents the conditional distribution of the tags given the features (He, Y. et al., 2008). All files used throughout this project are available in our GitHub: "Gene-discovery-from- biomedical-literature" https://github.com/kevinLCG/Gene-discovery-from-biomedical-literature.

Training the CRF

The files that were employed for training the model are located in the "datasets/" directory within the GitHub. First, the original dataset: "text-annotated-abstracts.txt", located in the same directory and already processed with CoreNLP), was splitted in two new separated datasets, corresponding to .70 and .30 of the available data, one for training the model and the other one for testing it, respectively. Both files were tagged examples, preprocessed with CoreNLP which provided tokenization, and lemmatization. Then the training dataset was splitted in 7 different groups, each containing more training data than the foreone. This was made to evaluate how the performance of the CRF changes as the training dataset is increased.

Testing the CRF

The testing file represented .30 of the available data . It contained preprocessed abstracts as in the training. Tags were not provided and the classification of the model was compared with the tagged version of the file. Statistics and values of the trained model are reported in the "report/" directory.

Results

Influence of the iteration number, cv, and hyperparameters in the model's F1-score

All results in Table 1 were obtained with the .70 training set. A poor performance was observed when the "max iter" was reduced to one quarter of the original configuration (table 1 RUN 2), recall was the most affected value, precision did not varied considerably (data not shown). Large "max iter" parameter did not improve the f1-score, actually, f1-score dropped .05 from the original configuration (table 1 RUN1). We found that "cv" tends to affect the f1-score more than "n iter" parameter when they were alternatively elevated (table 1 RUN 4 and RUN 5) Our best model, and the one we used to make the following analysis was RUN 8 (table 1).

The training dataset size has an impact in the CRF performance

Precision "P" (figure 1a) is a metric that shows the fraction of words correctly tagged as genes from all the words tagged as genes by the model (TP/TP+FP);

while recall "R" (figure 1b), is the fraction of retrieved genes by the CRF from all the genes in the test set; and F-score (figure 1c) is their harmonic mean taking into consideration the two aforementioned parameters: 2PR/ (P+R). The obtained results can be seen in figure 1. We observe a clear increase in the three scores as the training set grows from .10 to .70.

One aspect to take into consideration is that our method has an improvement in the the f1-score (f1= 0.864) when cross validation and number of iterations parameters were modified compared with the model Mendez C.,2019 http://pakal.ccg.unam.mx/cmendezc/conditional-random-fields presents, which has a f1-score of 0.845.

Table 1 Obtained scores when varying training data-set size (default values: cv=10 , n iter= 20 , max iter=100).It can be seen that RUN 8 has the best F1-score

	F1 SCORE	CV	N ITER	MAX ITER	EXTRA
RUN 1	0.84	10	20	300/700	-
RUN 2	0.79	10	20	25	-
RUN 3	0.83	10	20	50	-
RUN 4	0.859	50	50	100	-
RUN 5	0.861	50	10	100	-
RUN 6	0.856	10	20	100	-
RUN 7	0.84	10	20	100	$\epsilon = 0.001$
RUN 8	0.864	10	20	100	$\epsilon = 0.01$
RUN 9	0.836	50	10	100	-
RUN 10	0.8495	50	10	200	$\epsilon = 0.001$
RUN 11	0.8508	50	10	200	$\epsilon = 0.001$

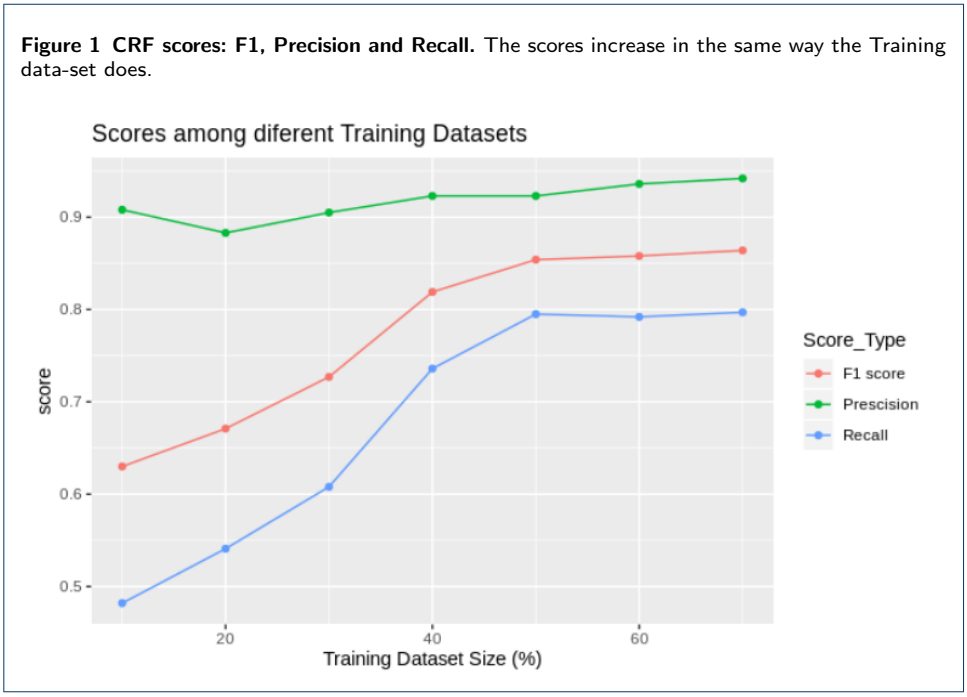

Figure 1 CRF scores: F1, Precision and Recall. The scores increase in the same way the Training data-set does.

Discussion

In the report obtained using the .70 training report (located in the "reports/" directory), we can appreciate what the model learnt from the given data. Inside

this rules, it can be seen that the "lemma" of the analyzed word and the one of the adjacent words is a characteristic that is highly taken into account, as the model "learns" some words that are very likely to be within gene names such as *transforming* and *fibronectin*, probably because such words are very ubiquitous in gene names. It also considers the "word" characteristic of the next two words, such as *factor* and *RII*, and if the model finds these words, it will very likely classify the word two positions before, as a gene name. We could also notice some rules that indicate that the word that is analyzed is not a gene, such is the case when the word two positions after is *factor*.

The weights of the top positive rules are much more larger than the top negative ones. This could mean that the model focuses more in positive correlations and does not give much importance to negative correlation rules.

Conclusions

Entity recognition in Biology is particularly a difficult task, because of the diversity of names, the rapid increase of biological knowledge and the lack of standards. In this study, we present a machine learning approach, specifically Conditional Random Fields, to identify human gene names from biomedical literature. Despite that the performance of the model might still be little improvable, we obtained results high enough to be considered valuable. Therefore, we propose that this approach can be extensively useful to extract relevant information from literature in our final project, in which the taggs to be pursuited will be bacteria and medium names that will be obtained from recent articles related to bioremediation.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Center for Genomic Sciences, UNAM, University Avenue, Cuernavaca, Morelos, Mexico. [2]Center for Genomic Sciences, UNAM, University Avenue, Cuernavaca, Morelos, Mexico.

References

1. He, Y., Kayaalp, M. (2008). Biological entity recognition with conditional random fields. AMIA ... Annual Symposium proceedings. AMIA Symposium, 2008, 293–297.
2. Manning, Christopher D., Surdeanu, Mihai, Bauer, John, Finkel, Jenny, Bethard, Steven J., and McClosky, David. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.