Development of a Machine Learning Framework for Biomedical Text Mining

Ruben Rodrigues, Hugo Costa and Miguel Rocha

Abstract Biomedical text mining (BTM) aims to create methods for searching and structuring knowledge extracted from biomedical literature. Named entity recognition (NER), a BTM task, seeks to identify mentions to biological entities in texts. Dictionaries, regular expressions, natural language processing and machine learning (ML) algorithms are used in this task. Over the last years, @Note2, an open-source software framework, which includes user-friendly interfaces for important tasks in BTM, has been developed, but it did not include ML-based methods. In this work, the development of a framework, BioTML, including a number of ML-based approaches for NER is proposed, to fill the gap between @Note2 and state-of-the-art ML approaches. BioTML was integrated in @Note2 as a novel plug-in, where *Hidden Markov Models*, *Conditional Random Fields* and *Support Vector Machines* were implemented to address NER tasks, working with a set of over 60 feature types used to train ML models. The implementation was supported in open-source software, such as *MALLET*, *LibSVM*, *ClearNLP* or *OpenNLP*. Several manually annotated corpora were used in the validation of BioTML. The results are promising, while there is room for improvement.

Keywords Biomedical text mining · Named entity recognition · Machine learning

1 Introduction

Nowadays, the life sciences produce large amounts of information spread in scientific literature and databases. The biomedical literature contains non-structured

R. Rodrigues(✉) · M.Rocha
Centre of Biological Engineering, University of Minho, Braga, Portugal
e-mail: pg25227@alunos.uminho.pt

R. Rodrigues · H. Costa
Silicolife, Lda, Braga, Portugal

M.S. Mohamad et al. (eds.), *10th International Conference on PACBB*,
Advances in Intelligent Systems and Computing 477,
DOI: 10.1007/978-3-319-40126-3_5

data [1], written in natural language, making the extraction of high-quality information a difficult challenge. Indeed, biomedical researchers spend large amounts of time extracting useful information from literature. The Biomedical Text Mining (BTM) field is concerned with the extraction of high-quality information from literature from the biological and biomedical domains. BTM emerged to create tools and methodologies that can automate and reduce time-consuming tasks when searching for information lying in biomedical literature [2].

Information Extraction (IE) has the ability to extract high-quality information from text streams. Named Entity Recognition (NER), a task in IE, aims to identify bio-entities in text streams [3]. NER tasks can be performed by distinct approaches, like lexicon-based, rule-based or machine learning-based techniques. NER performance can be tested and validated against gold standard corpora for specific case studies containing curated annotations [4].

Several NER methods have been developed with different advantages and limitations [3]. ML techniques have proven to be reliable, fast, scalable and automated processes. These can be used to perform NER tasks over biomedical literature [5, 7], requiring curated training sets to learn models. Hidden Markov Models (HMM) [8], Conditional Random Fields (CRF) [9] and Support Vector Machines (SVM) [6] are common ML models used in BioTM tasks [5, 10].

@Note2 [11] is a multi-platform BTM workbench written in Java, which encompasses the most important Information Retrieval and Information Extraction tasks. However, ML-based methodologies were not duly exploited in the available versions, a limitation that will be addressed by this work.

Indeed, the main goal of this work is to construct a framework that integrates ML algorithms addressing BTM tasks to fill the gap between @Note2 operations and state-of-the-art ML approaches. To make that possible, this study aims to:

- Create a ML framework with the capacity to train different models from annotated corpora and applying them to raw text for NER purposes;
- Create tools to evaluate and compare the performance of different models for the annotation of bio-entities, enabling the comparison of ML based approaches with other methods already implemented;
- Create a plug-in for @Note2 which allows the connection of ML tools with the remaining @Note2 structures, also defining appropriate user interfaces for the ML operations integrated within @Note2s architecture;
- Validate the overall framework with gold standard corpora.

2 Methods

Machine Learning (ML) is a sub-field of computer science and statistics that has been applied to solve problems in different scientific areas, being deeply related to Artificial Intelligence and Optimization. ML addresses the creation of mathematical models from data [1]. There are several advantages regarding the use of ML methods,

such as the possibility to retrieve accurate results in an automated, fast and scalable way. However, ML methods have some limitations due to the dependency on the input data to train the models and issues as overfitting or underfitting [12]. Metrics as precision, recall and F-scores can be used to evaluate the performance of ML methods. These are calculated regarding a confusion matrix resulting from the application of an ML model to a set of examples, encompassing true and false positives and negatives [12].

Regarding BTM, ML methodologies require a training process using annotated data to create a model that can then be used to classify/find the terms in raw (unmarked) text. Commonly used models include Hidden Markov Models (HMM) [8], Conditional Random Fields (CRF) [9] and Support Vector Machines (SVM) [13]. These are typically trained with manually curated corpora. The major limitation of these techniques is the fact that creating curated corpora is a time-consuming process and a trained model may only be applied to a specific problem (i.e. is difficult to generalize to other biological contexts).

ML algorithms can be split in classifiers and transducers. Classifiers are used to classify text tokens as entities of interest (or not) based in the features that characterize the token and its neighborhood. Transducers are also used to annotate the token, but the classification is done not only based in the token features but also in a sequence of previous tokens' features. SVMs are classifiers which calculate, from training data, a hyperplane that separates the examples in distinct classes with a maximal margin separation, building a linear space through the use of a kernel function. On the other hand, HMMs and CRFs are possible transducers. An HMM is a statistical Markov model in which probability distributions are used to model data from time series observations [8]. The training of a HMM consists in model parameters adjustment, from the available features. The hidden states represent the possible NER classes and the observations are the features. CRFs are undirected graphical models used to segment and label sequence data [9]. This model presents advantages over HMMs since the strong independence assumptions made in HMMs can be relaxed in CRFs [14]. The CRF training has similarities with the one from HMMs, but the CRF supports more features in the same model. CRFs can be used for NER tasks, in which the labels for the states are features related to the entity annotation [7, 15].

The ML features can be classified in several groups according to the token characterization that is performed. Examples of those feature groups are orthographic, semantic, morphological, and sentence structure features, among others.

3 Implementation

The main purpose of BioTML is to perform NER tasks using ML approaches. Its main pipelines allow (i) the creation of ML models from annotated corpora and (ii) the capability to predict new annotations in unannotated documents. The conceptual structure of the framework was devised to accomplish these two pipelines (training

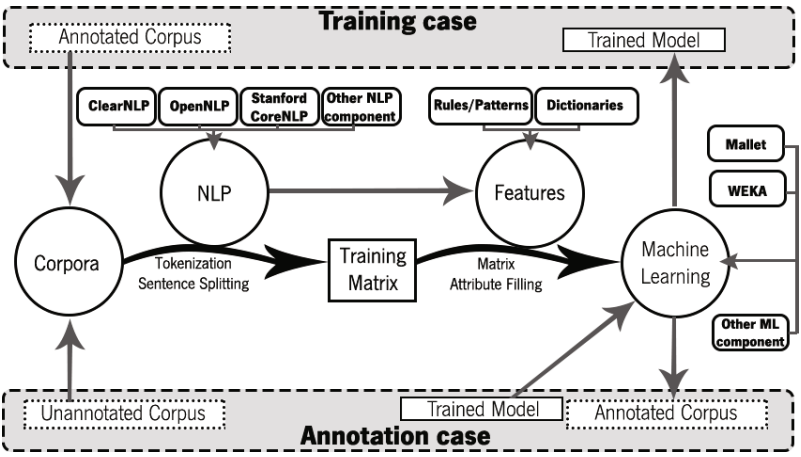

Fig. 1 Conceptual structure of the BioTML framework. The two BioTML main pipelines are the training and annotation processes represented in gray. Regarding the selected pipeline, the BioTML system uses a corpus (annotated or not), processes the tokenization using one of the implemented NLP systems and generates a matrix of features that is used in the ML module to train the model. All modules can be extended with new NLP systems, features or ML systems.

and prediction/ annotation), being divided in 4 main modules: Corpora, NLP, features and machine learning (shown in Figure 1).

The corpora module includes the corpus, document, sentence, token and annotation data structures, being responsible for storing documents and annotations. The NLP module includes components provided by open-source software (like OpenNLP [16], ClearNLP [17] and Stanford CoreNLP [18]). The use of this module is addressed to process tokenization of text streams in the corpora module, to create features and to generate a prediction matrix during annotation in the ML module. The features module includes dictionaries, patterns/rules, and NLP components that allow the creation of new features (e.g. part of speech, lemmas, chunks, dependency parsing, etc.). The features are created for each sentence and token from the corpus in model training. The machine learning module includes algorithms to train models provided by Java-based software packages (like MALLET [19]), evaluation components to test the models and annotators to apply the models over corpora.

This conceptual structure can be summarized in the two main pipelines: one that takes an annotated and curated corpus to create a model for NER, and the other that takes a model and an unannotated corpus to perform an NER task and returns an annotated corpus. Those pipelines were integrated on @Note2 in the form of a novel plug-in that allows the connection of both platforms. In the training operation, an annotated corpus is retrieved from @Note2 platform and converted into a BioTML corpus. This data structure is used as input in BioTML to train a model. In the annotation operation, the @Note2 API is used to retrieve a @Note2 unannotated corpus that is converted into a BioTML corpus. BioTML receives this unannotated corpus as input, the model linkage, settings and configurations to create an annotated

corpus. The results can then be viewed using the functionalities provided by @Note2. Figure 2 shows some screenshots showing the main operations of this plug-in. The plug-in is available by installing the latest @Note2 version on http://www.anote-project.org/, the installation steps and plug-in documentation are described in http://darwin.di.uminho.pt/anote2/wiki/index.php/Machine_Learning_biotml.

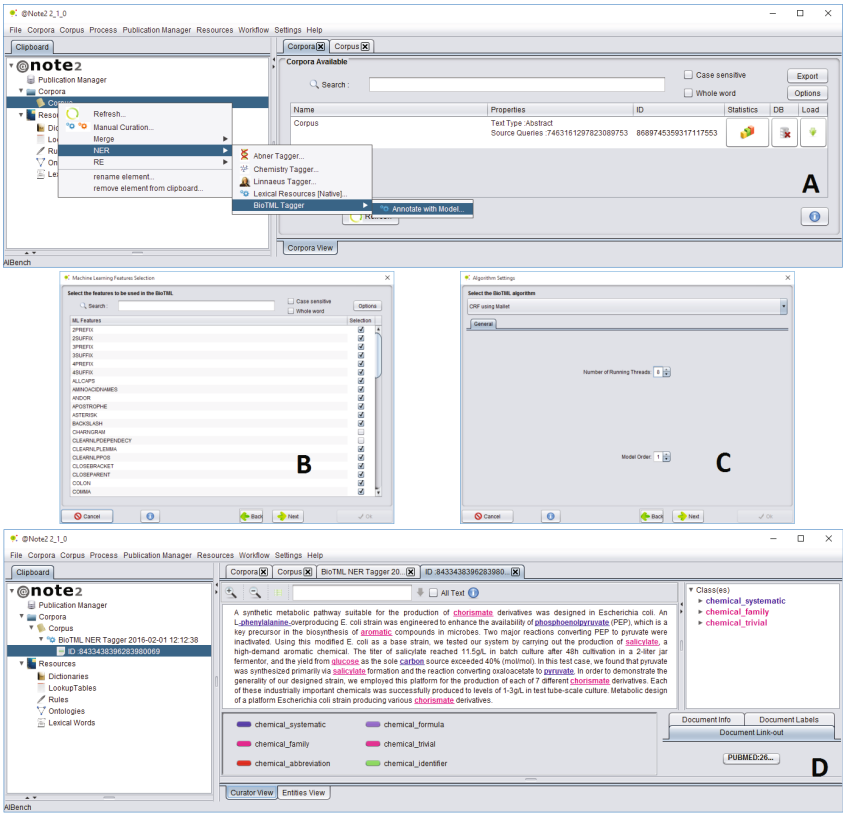

Fig. 2 Screenshots of BioTML plug-in. A - The plug-in GUI is accessed using a corpus instance from @Note2 clipboard. B - The feature types used for model training can be selected. C - The algorithm used for model creation is defined (HMM, CRF or SVM). D - A document annotated using a previously trained BioTML model.

4 Case Studies

4.1 BioTML Validation Using JNLPBA Corpus

BioTML was firstly validated using the JNLPBA corpus of the BioNLP/ NLPBA 2004 Shared Task [20]. The prediction capabilities of BioTML were tested using the training set to create the NER model and the test set to predict the annotations, using

the provided evaluation tool. The full corpus contains 5 types of NER annotations: protein, DNA, RNA, cell line and cell type. For each class, an NER model was trained using all features. The achieved prediction scores are given in Figure 3 and a comparison of BioTML against other systems is provided, namely Gimli [15] and the best system in the challenge [21].

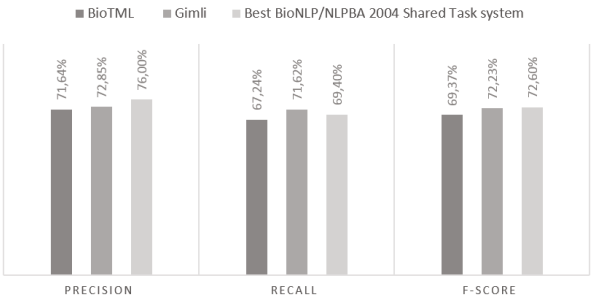

Fig. 3 Overall evaluation scores compared against other annotation systems.

As observed, the system has F-scores approximately 3% lower than both contenders. Although BioTML was not capable to perform at the level of the best systems, the results are promising since the difference is not large and the systems used in the comparison are among the best available. Indeed, being the first results of a novel platform, there is much room for improvements, since the features can be optimized for each class and no post-processing was performed.

4.2 BioCreative V: BioTML Used for Annotation of Chemical Entities in Patents

BioCreative proposes community challenges to evaluate BTM systems. Our framework was submitted to BioCreative V CHEMDNER-patents task [22], being used to perform the detection of chemical mentions in patents. To evaluate BioTML, we applied an approach in which CRFs trained with a set of specific features obtained by a feature optimization process (defined for each chemical class) were used to train ML models. We only used CRFs for this case study because the training of SVM algorithms was slower compared to the CRFs training and the HMM algorithms performed lower scores in the internal prediction results.

Each participant could submit 5 test set annotations. For each run, our framework predicted annotations using models trained over different sets of curated annotations. The predictions produced by each model were submitted and evaluated. The results including the systems' scores, BioTML test set scores, BioTML best prediction scores in the development set (trained using only the training set) and baseline scores

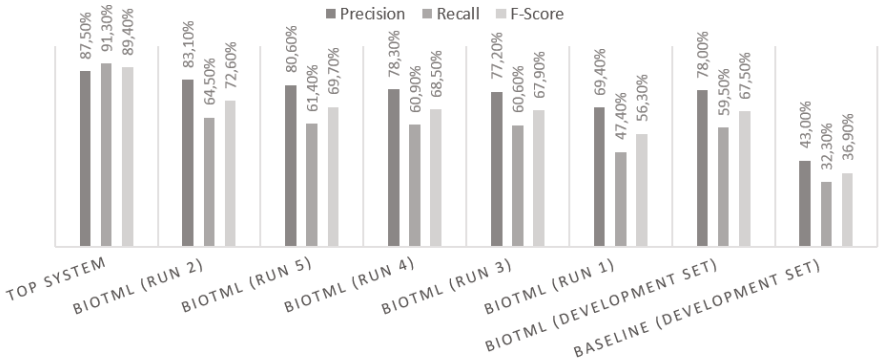

Fig. 4 BioCreative V task CHEMDNER-patents results and development results comparison against baseline results.

are given in Figure 4. The baseline scores were created by a case sensitive lookup list of all chemicals present in the training data. With these scores, we could verify if the training and development data share the same entity names and if our system could perform prediction of new entities present in the development set and not in the training set.

The results show that BioTML performed with high precision but low recall. A possible reason can be overfitting and a solution could be the use of other features addressed to identify a larger range of chemicals (e.g. word clustering). Another problem was the absence of a post-processing step, since a few annotation errors have a large impact in the precision, and could be corrected by post-processing steps as odd parenthesis verification in systematic chemical entities, re-annotation of chemical names using the predicted chemical names and dictionaries, addition of rules to only allow annotations that are between spaces or a selected punctuation. Overall, the system is not only capable to perform chemical annotation predictions with high precision, but also performs predictions with better recall than systems that only use dictionaries/lookup lists.

5 Conclusions and Further Work

ML approaches have been developed and implemented for different areas of BTM, including NER, using HMMs, CRFs and SVMs, among other algorithms. In this work, a ML-based framework dedicated to NER tasks, named as BioTML, has been developed and validated. The main innovation of this work is the creation of a modular framework, integrating several NLP and ML systems that can be enhanced with further systems using the provided API. Also, it is capable to predict NER annotations

with high performance. The BioTML integration into @Note2 allows using ML approaches for NER in an user-friendly environment.

ML limitations like the necessity of representative and large datasets, high variety of feature types, optimization of selected features and fine-tuning of the settings are important and were taken into account in BioTML design. Although the achieved results are promising, the system could be improved with the implementation of more features, post-processing steps (e.g. with dictionary matching and/or regular expression rules) and other NER approaches which could bring more options to create models fitted to the given datasets.

Additionally, the framework can be improved in several points, which will be addressed in the future, including the implementation of further ML algorithms, of a command line interface or of automatic feature selection algorithms. Also, relation extraction tasks can be handled by BioTML without major changes.

Acknowledgments This work is co-funded by the North Portugal Regional Operational Programme, under the "Portugal 2020", through the European Regional Development Fund (ERDF), within project SISBI- Ref NORTE-01-0247-FEDER-003381.

References

1. Feldman, R., Sanger, J.: The Text Mining Hand Book - Advanced Approaches in Analysing Unstructured Data (2007)
2. Shatkay, H., Craven, M.: Mining the biomedical literature (2012)
3. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **1–20**, 2007 (1991)
4. Kim, J.D., Ohta, T., Tateisi, Y., Tsujii, J.: GENIA corpus - A semantically annotated corpus for bio-textmining. Bioinformatics **19** (2003)
5. Eom, J., Zhang, B.: PubMiner : Machine Learning-based Text Mining for Biomedical Information Analysis. Genomics **2**, 99–106 (2004)
6. Takeuchi, K., Collier, N.: Bio-medical entity extraction using support vector machines. Artificial Intelligence in Medicine **33**, 125–137 (2005)
7. Bundschus, M., Dejori, M., Stetter, M., Tresp, V., Kriegel, H.P.: Extraction of semantic biomedical relations from text using conditional random fields. BMC Bioinformatics **9**, 207 (2008)
8. Ramage, D.: Hidden Markov models fundamentals. Standford CS229 Section Notes, pp. 1–13 (2007)
9. Sutton, C.: An Introduction to Conditional Random Fields. Foundations and Trends in Machine Learning **4**(4), 267–373 (2012)
10. Torii, M., Wagholikar, K., Liu, H.: Detecting concept mentions in biomedical text using hidden Markov model: multiple concept types at once or one at a time? Journal of Biomedical Semantics **5**, 3 (2014)
11. Lourenço, A., Carreira, R., Carneiro, S., Maia, P., Glez-Peña, D., Fdez-Riverola, F., Ferreira, E.C., Rocha, I., Rocha, M.: @Note: A workbench for Biomedical Text Mining. Journal of Biomedical Informatics **42**(4), 710–720 (2009)
12. Batanlar, Y., Özuysal, M.: Introduction to machine learning. Methods in Molecular Biology **1107**, 105–128 (2014)
13. Quan, C., Wang, M., Ren, F.: An unsupervised text mining method for relation extraction from biomedical literature. PLoS ONE **9**(7), 1–8 (2014)
14. Pereira, F., Lafferty, J., Mccallum, A.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of 18th International Conference on Machine Learning, (ICML), pp. 282–289 (2001)

15. Campos, D., Matos, S., Oliveira, J.L.: Gimli: open source and high-performance biomedical name recognition. BMC Bioinformatics **14**, 54 (2013)
16. Morton, T., Kottmann, J., Baldridge, J.: OpenNLP: A Java-based NLP Toolkit (2005)
17. Choi, J.D.: Optimization of Natural Language Processing Components for Robustness and Scalability. PhD thesis, University of Colorado at Boulder (2012)
18. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D: The stanford coreNLP natural language processing toolkit. In: Proceedings of 52nd Annual Meet. Assoc. Comput. Linguistics: System Demonstrations, pp. 55–60 (2014)
19. McCallum, A.K.: MALLET: A Machine Learning for Language Toolkit (2002)
20. Kim, J.D., Ohta, T., Tsuruoka, Y., Tateisi, Y., Collier, N.: Introduction to the bio-entity recognition task at JNLPBA. In: Proceedings of Intern. Joint Workshop Natural Language Processing in Biomedicine and Its Applications, pp. 70–75 (2004)
21. Zhou, G., Su, J.: Exploring deep knowledge resources in biomedical name recognition. In: Workshop on Natural Language Processing in Biomedicine and Its Applications at COLING, pp. 96–99 (2004)
22. Krallinger, M., et al.: Overview of the CHEMDNER patents task. In: Proceedings of the Fifth BioCreative Challenge Evaluation Workshop, pp. 63–75 (2015)