ISSN:0975-9646

A Survey on Machine Learning Techniques to Extract Chemical Names from Text Documents

Ms. Snehal P. Umare[1], Dr. Neeta A. Deshpande[2]

[1]P.G. Student, Computer Engg. Department, MCOERC, Nashik
Savitribai Phule Pune University, India
[2]Professor, Computer Engg. Department, MCOERC, Nashik
Savitribai Phule Pune University, India

Abstract**—**The chemical name extraction has a great importance in the biomedical field. Named Entity Recognition is the subtask of information extraction that is used to identify named entities in the given data. There are various dictionary-based, rule-based and machine learning approaches available for Named Entity Recognition. Rule based techniques include hand written rules. In this paper an extensive survey of machine learning models such as Hidden Markov Model (HMM), Support Vector Machine (SVM), Conditional Random Fields (CRFs) etc. that are used to develop NER systems is carried out.

Keywords— **Chemical name extraction, Named Entity, Hidden markov model, Conditional random fields, Feature functions, Support vector machine.**

I. Introduction

Named Entity Problem refers to the extraction of named entity from the documents [1]. Named entity problem includes extraction of names of person, organization, e-mail addresses, products etc [1]. Named Entity Recognition is an important field in the natural language processing. There are different approaches to solve Named entity problems [1]:
1) Dictionary based NER systems [9]
2) Rule-based NER systems [10]
3) Machine learning (ML)-based NER systems

1) Dictionary based approach [2]:
A dictionary is a set of words for a specific domain. Examples of dictionaries in the chemistry and biomedicine domains are the Jochem dictionary [3] and DrugBank dictionary. Jochem dictionary is used to identify small molecules and drugs in the text, and the DrugBank dictionary is used for drugs.
The dictionary-based systems contain the list of terms used to identify the occurrences of chemical names in the text. The system performs string matching. It specifies whether a word or a group of words selected from the text matches a term from the dictionary. The string matching algorithms are divided into two types [2]:
1. Exact matching: It searches for the chemical terms in the dictionary that exactly matches the text.
2. Flexible or approximate matching: This process performs an approximate matching for the given chemical terms to the text. It allows insertion, deletion or substitution for some characters. Fuzzy matching is performed.
The disadvantage of dictionary based systems is that it generally offers high precision but poor recall in case of spelling errors in the text. No dictionary is complete to cover all chemical names and variations. Also maintaining dictionaries is costly and time-consuming.

2) Rule-based NER systems [2]:
Rule-based systems use a set of hand-made rules to extract the chemical names from the free text. These rules include grammar based and syntactic (e.g., word precedence) rules. Two types of rules are used in the rule-based systems [2]:
1. Pattern-based rules: These rules are related to orthographic or morphological patterns of the words.
2. Context-based rules: These rules are related to the context of the words in the text.

3) Machine learning (ML)-based NER systems [2]:
Machine learning NER systems use statistical models for recognising specific entity names by utilising a feature-based representation of the observed data that depends on the annotated documents. Two basic steps are required to develop the ML-based systems [2]:
1. Training: The machine-learning model is trained to use the annotations in the annotated documents.
2. Annotating: The annotation is performed on the documents to produce the chemical names based on the past experience learned from the annotated documents.

Chemical names are found in various formats such as CAS/ registry number, Common/trivial name, Systematic/IUPAC name, or formula [1]. One can list a large number of handcrafted rules to differentiate chemical names from background text. Rule-based solutions require broad domain knowledge about chemical nomenclatures, which obstruct the acceptance of such solutions [1]. Dictionary-based extraction solutions with string matching or regular expressions are normally used to extract non-systematic chemical names, such as names in the form of registry numbers (e.g., CAS numbers), trademarks, and trivial names [1]. Trivial names are commonly used because their equivalent systematic names are considered too difficult [1].

Over the years, many learning-based NE techniques have been developed, such as the hidden Markov model (HMM), support vector machine (SVM) and conditional random fields (CRFs).

II. Machine learning techniques

A. Hidden Markov Model (HMM) [4] [5]:

HMM stands for Hidden Markov Model. HMM is a generative model. In HMM there is an observed sequence O= $\{o_1, o_2, o_3....o_n\}$ and F = $\{f_1, f_2 ...f_n\}$ is the feature set associated with word i. The goal is to calculate the tag sequence M= $\{m_1, m_{2..}...m_n\}$ for which the conditional probability of tag sequence given the observation sequence is maximized. The model assigns the joint probability to paired observation and label sequence [4]. Then the parameters are trained to maximize the joint likelihood of training sets [4]. HMM model uses forward-backward algorithm, Viterbi Algorithm and Estimation-Modification method for modelling. Basic theory of HMM is easy to understand and implement. To define joint probability over observation and label sequence HMM needs to number all probable observation sequence. Hence it makes various assumptions about data like Markovian assumption i.e. current label depends only on the previous label. It is not practical to represent multiple overlapping features and long term dependencies. Number of parameters to be evaluated is huge. So HMM needs a large data set for training.

B. Conditional Random Fields (CRF) [6][7]:

CRFs are a group of probabilistic, undirected graphical models. Let O be an input sequence $O = (o_1,...,o_n)$. S be the state sequence $S=\{s_1,s_2....s_n\}$. CRF calculates probability $P(S \mid O)$ of a possible label/state sequence $S = (s_1,...,s_n)$, given an input sequence *O*. In the context of chemical named entity recognition this observation sequence O refers to the tokenized text. This is the sequence of tokens which are defined by a process called tokenization i.e. splitting the text at white space, punctuation marks and parentheses.

A CRF in general is an undirected probabilistic graphical model

$$P_\lambda(s|o) = \frac{1}{Z_0} exp\left[\sum_{m=1}^{M}\sum_{n=1}^{N} \lambda_n \times f_n(s_{m-1}, s_m, O, m)\right]$$

Where, $f_n(m_{t-1}, m_t, o, m)$ is a feature function whose weight λ_n, is to be learned via training. The values of the feature functions may range between −1 . . . +1, but typically they are binary. To make all conditional probabilities sum up to 1, we must calculate the normalization factor Z_0 [6].

$$Z_0 = \sum_{s} exp\left[\sum_{m=1}^{M}\sum_{n=1}^{N} \lambda_n \times f_n(s_{m-1}, s_m, O, m)\right]$$

Z_0 is defined as the sum of exponential number of sequences, hence is difficult to compute. Z_0 depends on O and the parameters λ[7].

The parameters λ and f() can take arbitrary real values, and the whole exp function will be non-negative [4]. The scalar λ_n is the weight for feature $f_n()$. The λ_n's are the parameters of the CRF model, and must be learned, similar to θ = {π, Ø, A} in HMMs [7].

1) Procedure to find Start probability (π): Start probability is the probability that the sentence start with particular tag [6].

So start probabilities (π) = (Number of sentences that start with particular tag / Total number of sentences in the corpus) [4].

2) Procedure to find Transition probability (Ø): If there is two pair of tags called T_i and T_j then transition probability is the probability of occurring of tag T_j after T_i [6]. So Transition Probability (A) = (Total number of sequences from T_i to T_j / Total number of T_i) [6].

3) Procedure to find emission probability (A): Emission probability is the probability of assigning particular tag to the word in the corpus or document [6]. So emission probability (A) = (Total number of occurrence of word as a tag/Total occurrence of that tag [6].

Feature Functions [7]: The feature functions are the key components of CRF [4]. In linear-chain CRF, the general form of a feature function is $f_n(S_{m-1}, S_m, O_, t)$, where S_{m-1}, S_m are the adjacent states. O is the input sequence. And we are in sequence m. These are arbitrary functions that produce a real value. Typically when using linear-chain CRF in sequence tagging, we consider simple binary-valued features. Each feature has a corresponding weight λ_n to specify if the feature is favoured. If $\lambda_n > 0$, then when feature f_n is active, it increases the probability of the tag sequence S [7]. So the CRF model prefers to have the feature active and tends to tag accordingly [7]. If, on the other hand, $\lambda_n < 0$, the CRF model prefers to avoid having the feature active and also tags accordingly [7].

C. Support Vector Machine (SVM) [8]:

Support Vector Machine is supervised learning model with associated learning algorithms that analyse data and recognize patterns, used for classification and regression analysis. In a classification task using SVM, the task usually involves training and testing data which consist of some data instances. The goal is to predict the class of the data instances. It is one of the famous binary classifier giving best results for fewer data sets and can be applied to multi-class problems by extending the algorithm. The SVM classifier used in the training set for making the classifier model and classify the testing data based on this model with the use of features.

D. Context aware CRF [1]:

Compared to NE methods like hidden markov model, support vector machine CRFs are more expressive and are able to model overlapping, non independent features of the output [1]. We have seen above how CRF can be used for sequence tagging. Let us now discuss how CRF with a variation Context aware CRF is more effective for sequence tagging in large documents.

For tagging a document, it is first tokenized. Then features are generated for each token. When the document is large this process becomes time consuming. Hence to reduce computation without hurting tagging accuracy, CaCRF technique is used. The CaCRF segments a full document and generates a list of text fragments based on simple rules [1]. Each text fragment contains at least one candidate chemical name along with its context [1]. Features are then generated for the list of fragments only [1]. This method can significantly reduce the amount of data to be processed, and increase the online runtime performance of CRF [1].

TABLE I. A Survey on various machine learning techniques to extract chemical names

No.	*NER Approach*	*Model used*	*Algorithms used*	*Used in*	*Reference Paper*	*Data Set used*	*Precision*	*Recall*	*F-measure*
1.	Dictionary based approach		1) Exact Matching algorithm 2) Approx. matching algorithm	To find non-systematic chemical names e.g. Trivial/Common names.	[9]	1000 Medline abstracts	98%	88%	92.73%
2.	Rule based approach		1) Pattern based algorithm 2) Context based algorithm	To find CAS/registry nos.	[10]	50 Medline abstract	76%	84%	80%
3.	Machine learning based approach	i. HMM	1) Forward-backward algorithm, 2) Viterbi algorithm, 3)Estimation Modification Algorithm	To find systematic chemical names	[13]	GENIA corpus-670 MEDLINE abstracts.	63.8%	61.3%	62.5%
		ii.CRF		To find systematic chemical names	[11]	1000 MEDLINE Abstracts	86.5%	84.8%	85.6%
		iii.SVM		To find systematic chemical names	[12]	All gene names in the training and devtest corpora	71.4%	72.8%	72.1%
		iv.CaCRF		To find systematic chemical names	[14]	100 patents	90.91%	82.19%	86.33 %

CONCLUSION

In this survey, we have studied different techniques employed for chemical name extraction like dictionary lookup, rule based and machine learning techniques like HMM, SVM, CRF and CaCRF. Dictionary based approach is suitable for extracting trivial chemical names as well as chemical formulae. Rule based approach is suitable for extracting CAS/Registry numbers. Machine learning techniques can best extract systematic chemical names. Among these techniques CRF is more expressive. It can represent multiple features of a word and can handle long term dependency problem faced by HMM. It has generally increased recall and greater precision as compared to other machine learning methods.

REFERENCES

[1] Su Yan, W.Scott Spangler, and Ying Chen, “Chemical Name Extraction Based on Automatic Training Data Generation and Rich Feature Set”, Bioinformatics, vol. 10, NO. 5, SEPTEMBER/OCTOBER 2013.

[2] Safaa Eltyeb and Naomie Salim, “Chemical named entities recognition: a review on approaches and applications .”

[3] Hettne KM, Stierum RH, Schuemie MJ, Hendriksen PJ, Schijvenaars BJ, Mulligen EM, Kleinjans J, Kors JA: A dictionary to identify small molecules and drugs in free text. Bioinformatics 2009.

[4] Sudha Morwal 1 , Nusrat Jahan 2 and Deepti Chopra 3, “Named Entity Recognition using Hidden Markov Model (HMM)”, International Journal on Natural Language Computing (IJNLC) Vol. 1, No.4, December 2012.

[5] Gitimoni Talukdar1, Pranjal Protim Borah2, Arup Baruah3, “A SURVEY OF NAMED ENTITY RECOGNITION IN ASSAMESE AND OTHER INDIAN LANGUAGES”. Department of Computer

Science and Engineering, Assam Don Bosco University, Guwahati, India.

[6] Asif Ekbal, Sivaji Bandyopadhyay, "A Conditional Random Field Approach for Named Entity Recognition in Bengali and Hindi", CSLI Publications, Linguistic Issues in Language Technology – LiLT, Volume 2, Issue 1 November, 2009.

[7] Xiaojin Zhu, CS838-1 Advanced NLP: Conditional Random Fields, 2007.

[8] Nusrat Jahan, Sudha Morwal, INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, "Named Entity Recognition in Indian Languages: A Survey", Department of computer science, Banasthali University, Jaipur-302001, Rajasthan, India.

[9] Sergei Egorov, PhD, Anton Yuryev, PhD, and Nikolai Daraselia, PhD.
"A Simple and Practical Dictionary-based Approach for Identification of Proteins in Medline Abstracts".

[10] R.Porkodi, B.LShivakumar, "Rule based approach for constructing Gene/Protein names Dictionary from Medline abstract", Department of Computer Science, Bharathair University, Coimbatore, Tamilnadu, India.

[11] R. Klinger, C. Kolarik, J. Fluck, M. Hofmann-Apitius, and C.M. Friedrich, "Detection of IUPAC and IUPAC-Like Chemical Names," Bioinformatics, vol. 24, 2008.

[12] Steffen Bickel, Ulf Brefeld, Lukas Faulstich, Jorg Hakenberg, Ulf Leser, Conrad Plake, Tobias Schefer, "A Support Vector Machine Classifier for Gene Name Recognition". Humboldt-University at zu Berlin, Department of Computer Science Unter den Linden 6, Berlin Germany.

[13] Dan Shen, Jie Zhang, Guodong Zhou, Jian Su, Chew-Lim Tan, "Effective Adaptation of a Hidden Markov Model-based Named Entity
Recognizer for Biomedical Domain".

[14] S. Yan, W.S. Spangler, and Y. Chen, "Cross Media Entity Extraction and Linkage for Chemical Documents," Proc. 25th AAAI Conf. Artificial Intelligence (AAAI '11), 2011.