Journal of AI and Data Mining
Vol 7, No 2, 2019, 311-319
DOI: 10.22044/JADM.2018.5929.1700

Improvement of Chemical Named Entity Recognition through Sentence-based Random Under-sampling and Classifier Combination

A. Akkasi[1*] and E. Varoglu[2]

1. Department of Computer Engineering, Bandar Abbas Branch, Islamic Azad University, Bandar Abbas, Iran.
2. Computer Engineering Department, Eastern Mediterranean University, Famagusta, North Cyprus, Via Mersin 10, Turkey.

Received 25 June 2017; Revised 09 April 2018; Accepted 03 June 2018
**Corresponding author: abbas.akkasi@gmail.com(A. Akkasi).*

Abstract
Chemical Named Entity Recognition (NER) is the basic step for the consequent information extraction tasks such as named entity resolution, drug-drug interaction discovery, and extraction of names of the molecules and their properties. Improvement of the performance of such systems may affect the quality of the subsequent tasks. The chemical text from which data for NER is extracted is naturally imbalanced since chemical entities are fewer compared to the other segments of the text. In this work, the class imbalance problem in the context of chemical NER is studied, and an adopted version of random under-sampling for the NER data is leveraged to generate a pool of classifiers. In order to keep the class distribution balanced within each sentence, the well-known random under-sampling method is modified to a sentence-based version, where a random removal of the samples takes place within each sentence instead of considering the dataset as a whole. Furthermore, in order to take the advantages of combination of a set of diverse predictors, an ensemble of classifiers trained with the set of different training data resulted by sentence-based under-sampling is created. The proposed approach is developed and tested using the ChemDNER corpus released by BioCreative IV. The results obtained show that the proposed method improves the classification performance of the baseline classifiers, mainly as a result of an increase in the recall. Furthermore, the combination of high performance classifiers trained using the under-sampled train data surpasses the performance of all single best classifiers and the combination of classifiers using the full data.

Keywords: *Chemical Named Entity Recognition, Class Imbalance Problem, Random Under-Sampling, Classifier Combination.*

1. Introduction

Automatic information extraction from unstructured text has been of interest in many domains, especially in biochemical-related fields [1]. Named Entity Recognition (NER) is an automated method for detecting the citation and classification of named entities in a free text. It is the basic step of many information extraction tasks. The performance of NER systems in the newswire domain is high but the results achieved so far in the chemical domain are not comparable to that degree [2]. The lower degree of success in the latter domain can mainly be attributed to the fact that unlike names of persons, locations, etc. in the news domain, the chemical named entities suffer from varying and complex morphologies as well as belonging to different types of nomenclature that are concurrently used to describe them in related documents [3]. These two factors are the main reasons for the difficulties of developing a single method that can detect all types of chemical/drug mentions with high accuracy. Machine learning techniques can be employed to solve the NER problem in different domains. These strategies have become prevalent in the early 2000s by introducing Maximum Entropy Markov Models (MEMMs) [4] and Conditional Random Fields (CRFs) [5]. The main idea behind this approach is to learn a statistic model using annotated training data and generalizing it to data without annotations (test

data). Lack of enough annotated data for creating a model can sometimes be seen as one of the considerable problems in this strategy, although in the last decade, several attempts have been made to create such corpora in various domains. In this work, a machine learning approach was employed to generate baseline classifiers for the proposed method. Since NER using a machine learning approach is a kind of classification tasks that aim at classifying entity mentions into classes of interest, it is not free from common challenges faced by many tasks defined in this domain. The class imbalance problem [6] is an example of such challenges that deserves attention. In this case, the number of samples that belong to the different classes may vary in big proportions, creating a skewed dataset. This is a natural problem when dealing with NER in a biochemical text since such documents contain less number of chemical named entities compared to the other segments of the text that build the structure of sequences to be labeled. The increase in the number of negative samples inside the train set will result in an increase in false negative predictions, and consequently, leads to a low recall in comparison with the system precision. This leads to an overall reduction in the classification performance of the system in terms of F-score [7]. Our proposed method consists of two steps. Firstly, we use random under-sampling at the sentence level to reduce the data imbalance in the train data. Since it is known that selecting a single best system is not always a trivial and best possible solution [8], for the second step, we use an aggregation of expert classifiers generated using different feature sets and under-sampled training data at different ratios instead of relying on the outcomes of standalone classifiers in order to boost the classification performance. We propose a new sentence-based random under-sampling method that is contrary to the well-known approaches. The adopted version of under-sampling is applied on each individual sentence given in the training data instead of considering all samples within the whole text in order to keep the class distribution balanced inside each sentence. This step helps us to generate classifiers with more balanced precision-recall scores. The ChemDNER [9] dataset, which is one of the most comprehensive chemical/drug corpora, was used for our experiments. The results obtained show an improvement in the performance of the classifiers trained using the under-sampled data compared to the baseline classifiers trained with the original imbalanced data. In addition, combining the classifiers from the latter step, we achieved a further improvement in the recognition performance along with the improvements obtained from baseline classifiers using the under-sampled data.

The remainder of this paper is organized as what follows. In the next section, a literature review on chemical NER is presented. The class imbalance problem is discussed in Section 3. Section 4 gives an overview of the proposed method. The experimental setup and the results obtained are given in Section 5. A conclusion is then provided, and future work is considered in the last section.

2. Literature review on chemical NER

A large number of applications have been implemented to deal with NER in the newswire, biomedicine, and other domains [10, 11] but lack of the large publicly available annotated text corpora for chemical domain is the main reason for the scarcity of chemical NER systems [9]. However, the ChemDNER task has been organized under the BioCreative IV event [12] as part of a community challenge to promote the development of chemical entities in a text. Leaman et al. [13] have developed tmChem to recognize the chemical entities by combining two independent machine learning models in an ensemble. Lowe et al. [14] have introduced LeadMine, a novel hybrid system that combines the rule based approach and the dictionary method together. Usie et al. [15] have implemented a tool named CheNER, where they combined the machine learning approach with dictionary and rule-based methods. Khabsa et al. [16] have created multiple extractors using CRFs [5], where they extracted some new features to improve the performance. Akhondi et al. [17] have implemented a hybrid system combining the common existing domain dictionaries and regular expressions. Dai et al. [18] have used a new representative tag scheme, IOBSE, where they highlighted the importance of tag set selection and the use of fine grained tokenization. Lu et al. [19] have used a semi-supervised learning approach based on mixed CRFs with word clustering. Campos et al. [20] have proposed a document processing pipeline for the annotation of chemical entities based on combination of two CRF models with some new features and a post-processing phase. Munkhdalai et al. [2] have applied domain knowledge in their own work to extend BANNER [21] as one of the state of art NER systems in biological domains. All the aforementioned studies have been carried out using the ChemDNER corpus, even though there are several other studies on NER in other domains such as newswire and gene mention detection. However, to the best of our knowledge, none of the studies

in the field has considered the highly imbalanced characteristics of the data.

3. A Brief review of class imbalance problem

The class imbalance problem refers to the phenomenon where the data used for classification contains more samples in some classes compared to some others, i.e. skewed class distribution [22]. When classifying samples in datasets suffering from imbalanced class distribution, most classifiers are biased towards the major classes and will have a poor performance on minority classes [23]. In the case of binary classification, the positive or target class makes up the minority class, whereas samples from the negative class constitute the majority class. Since the negative class has more samples, classifiers tend to classify the test samples as negative, thus producing many false negatives. This typically results in high precision-low recall classifiers, which, in turn, degrades the overall classifier performance often measured in terms of F-score, the harmonic mean of both [24]. The chemical named entity recognition task, where the recognition of chemical compounds or drug names from the free text is the main objective, suffers considerably from this problem. Naturally, the number of entities that make up the target (positive) class is much fewer compared to the other segments of the text that are not of interest (negative class). There are two main approaches mostly used as a solution to this problem. Sampling approaches try to make balance between the number of samples belonging to the major and minor classes by increasing the number of positive samples (over-sampling) or downsizing the number of samples from the negative class (under-sampling) [25]. Alternatively, algorithmic level solutions try to make changes in the learning algorithms in order to increase the performance of the classifiers [26]. The sampling methods are more commonly used in many application-oriented tasks. However, both sampling methods have their own drawbacks. Over-sampling increases the time required for training the classifiers, and furthermore, may cause over-fitting because of using duplicated samples in the training data. On the other hand, under-sampling potentially ignores some useful majority class instances [27, 28]. Various kinds of solutions based on sampling strategies or algorithmic approaches have been proposed by researchers; they can be found in [26] in more detail. In this work, we chose to use the random under-sampling method that tries to reduce the number of negative samples by removing them randomly but at the same time keeping the number of positive samples unchanged. A detailed description of the method applied during our experiments is presented in Section 5.

4. Proposed method

The drawbacks of using imbalanced datasets for training motivated us to make use of under-sampled data in this work. Additionally, in order to improve the recognition performance of chemical NER system, an ensemble of CRF classifiers was created. The pool of classifiers used for our ensemble was constructed using a diverse set of classifiers trained with different features extracted from various sets of under-sampled data.

The first step of the proposed method involves data preparation and pre-processing. Detection of sentence boundaries, removing nested named entities, tokenization, and converting class tags to the IOB2 format are performed in this step. Next, various features are extracted. Features extracted are made up of commonly used features for the NER task as well as some domain-relevant features. All features are extracted and their detailed information is presented in Section 6. Following feature extraction, CRF classifiers using full data are trained using a combination of features. All CRFs are trained using the Mallet toolkit [29] in the experiments. Then a set of different training data is sampled from the original imbalanced training data using various under-sampling ratios R_s with the same features extracted previously. The next step involves the manual selection of classifiers for the ensemble. Since one of the criteria is to make use of relatively strong classifiers in a pool, we select the baseline classifiers whose performance on development dataset is relatively high. In the final step, the majority voting method [30] is used to find the decision of final ensemble decision on the test data.

4.1. Proposed sentence-based random under-sampling for NER

As mentioned in Section 3, all the known sampling techniques are applied on the train dataset as a whole. However, since the training data used in NER applications is composed of a set of individual tokenized sentences, considering all tokens within the text altogether in the sampling process without considering sentence boundaries may result in keeping many sentences unchanged in terms of the sampling ratio. This is due to the fact that negative samples are randomly removed from any part of the text, and at the high sampling ratios, very few or even no samples may

be removed from some sentences. Therefore, we propose to apply random under-sampling on each sentence individually. In this case, under-sampling is applied more uniformly on every sentence that is potentially imbalanced. Figure 1 depicts the sentence-based random under-sampling algorithm. As it can be seen in this figure, firstly, the sentence boundaries are determined before tokenization. Using the information in the gold standard training data, all entity mentions and their starting and ending indices are determined in the next step. Since the most commonly used tag scheme is the IOB2 format [31], this scheme is also used throughout the experiments here. According to this tag scheme, a "B-Tag" represents a token that is the first token of an entity, an "I-Tag" represents a token that is part of an entity, and "O-Tag"s represent a token that is not part of any entity. Hence "B-Tag"s and "I-Tag"s can be considered as positive samples, whereas "O-Tag"s can be considered as negatives. The sampling ratio R can then be calculated for every sentence using Equation (1).

$$R = \frac{\sum B-Tagtokens + I-Tagokens}{\sum O-Tagtokens} \qquad (1)$$

Random under-sampling is carried out on each sentence at the input under-sampling ratio R_s using the algorithm. The algorithm is repeated for all sentences in the train data. In order to find the input sampling ratio R_s, which maximizes the classification performance on the whole train data, experiments are carried out for incremented values of R_s until the best value is found as the sampling ratio, R_{best}.. A proper value for R_{best} can be found using the validation data or using n-fold cross-validation [8] in the cases that a validation dataset does not exist. Random under-sampling is applied only on the training data, and the test data is kept unchanged since there is no prior information about the positive and negative samples in the unlabeled test data.

```
/* Random Undersampling algorithm applied for each sentence S */

    N: Number of token in S
    Np: Total number of tokens with B- or I- tag in S
    Nn: Total number of tokens with O tags in S
    Ns: Number of selected tokens with O tag in S
    K: Number of entities in S
    Rs: Input sampling ratio
    R: Acti=ual sampling ratio od sentence S
    startk: Location of the first token of sentence k
    endk: location of the last token of sentence k

    for j=1 .. N
        Mark all positive tokens in S as selected
        R = Nn / Np
        If Rs <=R     /*no need for sampling*/
            Mark all O tgged tokens in S as selected abd return
S
        Else
            for i = startj .. endj
                Mark a token of O tags randomly
                Ns = Ns1
                If Rs <= Ns /Np return S
            end for
            return S
```

Figure 1: Random under-sampling algorithm used on each sentence.

5. Experiments and results

The ChemDNER corpus [32] released by BioCreative IV [12] was used to evaluate the effectiveness of the proposed method. This corpus includes three datasets: training, validation, and test data, which wereannotated by domain experts.

Organizers of the ChemDNER task used 10000 abstracts from PubMed to create their corpus. Initially, all datasets were converted to a proper format acceptable for the classification algorithm. As the first step, the sentences were separated using sentence detector module of Apache Open NLP toolkit [33]. Then the tokenization algorithm proposed in [34] was applied. Next, we converted classes of entity mentions into the IOB2 format. Table 1 presents statistics about the dataset used. The total number of tokens (samples) as well as the number of positive and negative samples in each dataset is given. It can clearly be seen in table 1 that there is a high imbalance in the distribution of positive samples and negative samples in the corpus, and the dataset is heavily skewed in favor of negative samples. More precisely, 93% of tokens in each one of the datasets belong to the negative class, and only 7% of all tokens belong to the positive class.

Table 1. Statistics about ChemDNER Corpus.

	Training set	Validation set	Test set	Entire corpus
No. of abstracts	3500	3500	3000	10000
Total No. of samples	899343	893180	772847	2565370
No. of negative samples	834395	829038	718186	2381619
No. of positives samples	64948	64142	54661	183751

In the next step, the common features used for NER as well as some binary features associated with the chemical domain are extracted as new features. The domain-related features mostly show the presence or absence of a token in a specific list of chemical elements, amino acids, and common chemical prefixes and suffixes [35]. The output of OSCAR [36], one of state of the art NER systems in chemical/drug domain, is used as a feature. Additional features such as space in conjunction with Bag of Words, as suggested by the developers of ChemSpot [37], and another state of the art systems used in chemical NER are also used. The Brown's clustering algorithm [38] is employed in order to extract the clustering features. The N-gram features are extracted at the character level for each token including N-gram prefixes and N-gram suffixes for N = 1-4. The orthographic features extracted are commonly used orthographic features in other NER tasks [10]. Word shape feature represents the number of various types of existing characters in a token with a representative character for each type. The context features refer to the previous and next tokens that surround the current token.

Table 2. Feature sets used.

Feature set #	Feature set	Feature set #	Feature set	Feature set #	Feature set
1	In-domain features (Chemical names, etc.)	7	Space + tf	13	N-gram + Space + POS
2	Word Clusters	8	Space + tfidf	14	N-gram + POS + space + word shape
3	N-gram	9	Word shape	15	OSCAR's output
4	Orthographic	10	N-gram + Orthographic	16	1,3,4,5,6,9 + Context tokens
5	POS	11	N-gram + Word shape	17	16 + OSCAR's Output
6	Space	12	N-gram + Space	18	All features

ese features are used in isolation or in combination as the feature sets. The feature sets used are given in table 2. Although a numerous number of feature sets can be generated using different combinations of features, only those combinations that generate good results when tested with the validation data are used. The combination of all feature sets is also considered (Feature set #18) for reference. Table 3 shows the performance of 18 baseline classifiers constructed using each one of the corresponding feature sets in table 2. Performances are represented in terms of recall, precision, and F-score.

Table 3. Performance of Baseline Classifiers (E_i denotes the performance of classifiers using feature set i).

	Validation			Test				Validation			Test		
	R	**P**	**F**	**R**	**P**	**F**		**R**	**P**	**F**	**R**	**P**	**F**
$\mathbf{E_1}$	51.38	75.50	61.15	51.27	77.54	61.73	$\mathbf{E_{10}}$	67.10	77.40	71.88	67.03	79.79	72.86
$\mathbf{E_2}$	54.36	75.26	63.12	54.21	77.21	63.70	$\mathbf{E_{11}}$	68.02	77.53	72.46	67.66	79.76	73.21
$\mathbf{E_3}$	66.61	77.41	71.60	66.71	7976	76.65	$\mathbf{E_{12}}$	70.74	81.04	75.54	70.18	83.39	76.22
$\mathbf{E_4}$	50.71	74.89	60.47	50.23	76.61	60.68	$\mathbf{E_{13}}$	71.19	80.66	75.63	70.60	82.65	76.22
$\mathbf{E_5}$	48.66	72.43	58.21	48.14	74.22	58.40	$\mathbf{E_{14}}$	70.37	78.99	72.44	67.18	74.24	70.53
$\mathbf{E_6}$	56.66	79.56	66.19	56.34	81.33	66.57	$\mathbf{E_{15}}$	65.47	80.66	72.26	63.76	78.16	70.23
$\mathbf{E_7}$	51.30	79.01	62.21	50.33	80.52	61.94	$\mathbf{E_{16}}$	75.09	84.39	79.47	74.46	85.94	79.47
$\mathbf{E_8}$	50.18	76.18	60.51	30.48	79.95	44.13	$\mathbf{E_{17}}$	**76.95**	**85.14**	**80.81**	**76.21**	**85.93**	**80.77**
$\mathbf{E_9}$	53.89	75.28	62.81	53.68	77.43	63.40	$\mathbf{E_{18}}$	77.11	84.59	80.68	70.05	85.10	80.32

In the next step, random under-sampling is applied to the training data used for the classifiers listed in Table 3 using different input sampling ratios R_s in the range of 3-25. Based on the results for all classifiers with different feature sets tested on validation data, the upper bound is selected as 25 since the maximum performance is seen when the sampling ratio is around R_s = 23. Then those classifiers whose baseline performance was better than 70% in F-score and whose performance could be improved through under-sampling for some R_s were used to form the final ensemble. The five selected classifiers (E_{14}, E_{15}, E_{16}, E_{17}, and E_{18}) and their performances for each input sampling ratio R_s is given in table 4.

Table 4. Performance of five selected classifiers with different sampling ratios R_s on validation data.

	E14			E15			E16			E17			E18		
	R	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
Base R_s	70.37	78.99	72.44	65.47	80.66	72.26	75.09	84.39	79.47	76.95	85.14	80.81	77.11	84.59	80.68
3	75.80	60.10	67.05	81.12	56.84	66.84	82.88	67.00	74.10	83.63	63.40	72.12	84.53	68.45	75.64
4	77.47	55.28	64.52	79.05	59.67	68.01	82.02	70.34	75.73	85.04	71.59	77.74	84.14	71.50	77.31
5	78.11	64.29	70.53	78.44	59.01	67.35	82.51	72.34	77.09	83.49	73.73	78.31	84.59	71.91	77.74
6	77.65	67.28	72.09	78.40	63.89	70.41	81.77	74.69	78.07	83.72	74.10	78.61	83.28	74.41	78.60
7	78.68	66.10	71.84	77.29	60.80	68.06	81.25	75.25	78.14	83.00	75.43	79.03	82.88	76.40	79.50
8	79.52	65.16	71.63	77.85	63.92	70.20	81.26	76.02	78.55	82.26	76.90	79.49	81.87	76.93	79.32
9	78.86	66.28	72.03	78.14	68.52	73.02	81.04	76.94	78.94	81.95	78.10	79.98	82.12	77.93	79.97
10	77.81	67.62	72.36	78.75	63.08	70.05	80.56	77.74	79.13	82.68	78.68	80.63	82.55	78.62	80.54
11	77.77	69.98	73.67	76.06	67.98	71.79	80.54	78.48	79.50	83.07	74.59	78.60	81.93	78.46	80.16
12	78.94	69.10	73.69	77.77	68.93	73.09	80.14	78.29	79.21	82.21	79.11	80.63	82.68	78.90	80.75
13	76.98	70.75	73.74	76.72	71.19	73.85	80.14	78.42	79.27	81.64	80.05	80.84	81.67	79.17	80.40
14	77.83	67.88	72.52	77.17	70.22	73.53	80.43	78.86	79.64	81.84	80.15	80.99	81.60	79.85	80.72
15	79.06	69.91	74.20	76.32	70.56	73.33	80.64	78.91	79.77	81.94	79.67	80.79	81.59	80.23	80.90
16	78.53	67.48	72.59	73.61	67.24	70.28	80.16	79.60	79.88	82.18	79.50	80.82	81.01	79.90	80.45
17	78.36	66.14	71.74	73.58	68.87	71.15	80.17	78.63	79.39	81.52	80.59	81.05	80.00	75.90	77.90
18	77.70	70.03	73.66	76.04	72.55	74.25	80.01	79.97	**79.99**	80.92	80.08	80.50	81.21	81.15	81.18
19	78.54	71.18	74.68	73.64	57.69	64.69	79.45	80.30	79.87	81.51	81.45	**81.48**	81.80	80.03	80.91
20	78.76	71.10	74.73	76.25	72.14	74.14	79.91	79.82	79.86	81.36	80.90	81.13	81.13	81.20	81.17
21	73.29	71.20	72.23	74.08	71.82	72.93	79.59	80.27	79.93	81.35	81.18	81.26	81.94	80.75	81.34
22	77.50	71.78	74.53	75.66	73.21	**74.41**	79.30	80.52	79.91	81.29	81.36	81.33	80.69	81.10	80.89
23	77.36	72.56	**74.89**	75.93	71.00	73.38	79.55	80.13	79.84	81.32	81.16	81.24	81.47	82.04	**81.75**
24	77.85	72.02	74.82	76.13	68.46	72.09	79.68	79.52	79.60	82.13	78.64	80.34	81.28	81.57	81.42
25	77.84	70.36	73.91	76.09	70.59	73.24	79.30	80.61	79.95	81.06	81.17	81.12	81.17	81.49	81.33

The performance improvement can mainly be attributed to the increase in recall values with a slight decrease in precision resulting in relatively more balanced classifiers in terms of precision-recall compared to the baseline classifiers that are mainly higher in precision. Table 4 shows that the best performance for each classifier is achieved using a different R_{best} value in the range of 18-23, which are within close range to the upper bound R_s value used. The effect of classifier combination on different ensembles is investigated next. For each case, the majority voting method is used to decide for the final vote of the ensemble. Five different ensembles are formed using different strategies. Table 5 shows the 5 different ensembles and their performances on validation as well as the test data. C_1 is the ensemble of 18 classifiers in their baseline

Table 5. Performance of various classifier ensembles.

	Validation			Test		
	R	**P**	**F**	**R**	**P**	**F**
C_1	66.18	81.27	72.95	65.33	80.11	71.97
C_2	75.34	83.46	79.19	73.67	84.38	78.66
C_3	77.16	84.80	80.80	77.6	84.90	81.08
C_4	82.21	82.78	**82.49**	81.01	83.91	**82.43**
C_5	83.26	79.69	81.43	81.53	81.36	81.44

C1- Combination of all 18 baseline classifiers, C2- combination of 18 classifiers trained using under-sampled data at Rbest, C3- Combination of 5 strong selected classifiers in their base line form, C4- Combination of 5 strong selected classifiers trained

As stated earlier, since these classifiers are of low recall-high precision type, the performance of ensemble C_1 does not even reach those of strong single classifiers in their baseline form. It can clearly be concluded that combination of low recall-high precision classifiers is not helpful. This phenomenon is an innate property for classification tasks. Where the negative-positive data imbalance exists, the need for under-sampling is evident. In order to test this

phenomenon, we formed a second ensemble, C_2, which consisted of the same 18 classifiers but this time, in their under-sampled form, each one trained at its own best under-sampling ratio, R_{best}, obtained from experiments conducted for all values of R_s in the range of 3-25, as explained earlier. It can be seen that a combination of under-sampled classifiers result in over 9% improvement in the recall on validation data (8% on test data) and over 2% increase in precision on validation data (4% on test data), resulting in an improvement of 6.24% in F-score on validation data and 6.69% on test data. Clearly, under-sampling has a very big effect on improving recall and a lesser impact on precision. It can be argued that the improvement over F-score can mainly be attributed to the higher recall values of these classifiers compared to their baseline counterparts. Although the performance of all 18 classifiers for all under-sampling ratios R_s is not presented, the increase in recall values can be seen from the subset of classifiers given in Table 4. We can deduce that the combination of under-sampled classifiers clearly outperform the ensemble of classifiers in C_1, mainly due to higher recall characteristics. Ensemble C_3 is formed using 5 classifiers (E_{14}, E_{15}, E_{16}, E_{17}, and $E_{18)}$ selected using the criteria explained earlier in their baseline form. Since these classifiers are relatively strong ones (very strong in precision and relatively stronger in recall), compared to the other classifiers in the pool of 18, their combination results in an F-score of 80.80% on validation data but still lags behind that of the single best baseline classifier E_{17} (F-score 80.81%). However, it is still worth noting that a combination of 5 strong baseline classifiers, C_3,

outperforms the combination of all baseline classifiers, C_1, by 7.85% in F-score on validation data and 9.11% on test data. This result suggests that there is clearly a need for classifier selection of some sort in order to possibly obtain an improvement after combination of classifiers. C_4 is the ensemble formed using the 5 classifiers (E_{14}, E_{15}, E_{16}, E_{17}, and E_{18}) as in C_3 but this time each classifier is trained using sampled data at its own best performing under-sampling ratio, R_{best}. In other words, this ensemble is made up of relatively strong classifiers, which have room for improvement by under-sampling and also sampled at the best ratio R_{best} (sampling ratio is set to 23, 22, 18, 19, and 23 for E_{14}, E_{15}, E_{16}, E_{17}, and E_{18}, respectively). A careful examination of the performance of these classifiers from table 4 shows that they all have the characteristic of being balanced in terms of precision-recall. The performance of this carefully selected ensemble exceeds the performance of other combinations. This result shows the significance of careful selection of classifiers to be included in the ensemble, and the effect of the good choice of a sampling ratio during under-sampling. It also reveals once again the combination of a diverse set of classifiers with balanced precision-recall behavior results in an effective ensemble in terms of entity recognition performance. A final ensemble, C_5, is formed using all 5 classifiers from C_4 but the ensemble contains all classifiers in table 4 without their baseline forms. In other words, the 5 strong classifiers are trained for every value of R_s in the range of 3-25, and a total of (5 * 23 = 115) are included in the classifier pool. The ensemble contains a diverse set not only in terms of the type of features used but also in terms of the precision-recall characteristics. Here, the advantage clearly is the fact that one would not have to go through the trouble of finding a R_{best} for each classifier before combination (such as the case in C_4). However, although this ensemble presents a good recognition performance and ranks second after ensemble C_4, the existence of relatively weak classifiers, especially for low values of R, degrades the performance of the ensemble slightly compared to that of C_5.

6. Conclusion and future work

In this work, the effect of random under-sampling on chemical named entity recognition, a classification task that severely suffers from imbalanced data has been investigated. A new sentence-based random under-sampling approach that aims to make the ratio of positive and negative samples for each sentence independently balance is proposed. ChemDNER corpus was used as the main source of corpora throughout the work. By applying the proposed under-sampling method, the performance of individual classifiers improved mainly due to an increase in recall. We achieved a further improvement by forming an ensemble of classifiers. The selection of classifiers for the ensemble was done using heuristics upon the observation of recognition performance of base classifiers on validation data. The final prediction was obtained using the majority voting scheme. Future work includes the automation of the classifier selection process through the use of evolutionary algorithms and extending the study on other biomedical corpora for further validation of the results.

References

[1] Khabsa M, Giles CL. (2015). Chemical entity extraction using CRF and an ensemble of extractors. J Cheminform; 7:1-9.

[2] Munkhdalai, T., Li, M., Batsuren, K., Park, H., Choi, N. and Ryu, K.H. (2015). Incorporating domain knowledge in chemical and biomedical named entity recognition with word representations. J Cheminform; 7: p. S9.

[3] Vazquez, M., Krallinger, M., Leitner, F. and Valencia, A. (2011). Text mining for drugs and chemical compounds: methods, tools and applications. Mol Inform; 30:506-519.

[4] McCallum, A., Freitag, D., & Pereira, F. C. (2000). Maximum Entropy Markov Models for Information Extraction and Segmentation. In: ICML June, pp. 591-598.

[5] Lafferty, J., McCallum, A. and Pereira, F.C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

[6] Guo, X., Yin, Y., Dong, C., Yang, G. and Zhou, G. (2008). On the class imbalance problem. In: ICNC'08. Fourth International Conference on Natural Computation, vol. 4. pp. 192-201.

[7] Magdy, W. and Jones, G.J. (2010). PRES: a score metric for evaluating recall-oriented information retrieval applications. In: 33rd international ACM SIGIR conference on Research and development in information retrieval. ACM, pp. 611-618.

[8] Devijver, P.A. and Kittler, J. (1982). Pattern recognition: A statistical approach (Vol. 761). London: Prentice-Hall.

[9] Usié, A., Alves, R., Solsona, F., Vázquez, M. and Valencia, A. (2014). CheNER: chemical named entity recognizer. Bioinformatics, 30:1039-1040.

[10] Krallinger, M., Leitner, F., Rabal, O., Vazquez, M., Oyarzabal, J. and Valencia, A. (2015). CHEMDNER: The drugs and chemical names extraction challenge. J Cheminform, 7: p. S1.

[11] Krallinger, M., Valencia, A. and Hirschman, L. (2008). Linking genes to literature: text mining, information extraction, and retrieval applications for biology. GENOME BIOL, vol. 9, pp. 1-14.

[12] Arighi, C.N., Wu, C.H., Cohen, K.B., Hirschman, L., Krallinger, M., Valencia, A., Lu, Z., Wilbur, J.W. and Wiegers, T.C. (2014). BioCreative-IV virtual issue. Database, p.bau039.

[13] Leaman, R., Wei, C.H. and Lu, Z. (2015). tmChem: a high performance approach for chemical named entity recognition and normalization. J Cheminform; 7: S3.

[14] Lowe, D.M. and Sayle, R.A. (2015). LeadMine: a grammar and dictionary driven approach to entity recognition. J Cheminform; 7:S5.

[15] Usié, A., Cruz, J., Comas, J., Solsona, F. and Alves, R (2015). CheNER: a tool for the identification of chemical entities and their classes in biomedical literature. J Cheminform 7:S15.

[16] Khabsa, M. and Giles, C.L. (2015) Chemical entity extraction using CRF and an ensemble of extractors. J Cheminform, vol. 7, pp. 1-9.

[17] Akhondi, S.A., Hettne, K.M., van der Horst, E., van Mulligen, E.M. and Kors, J.A. (2015). Recognition of chemical entities: combining dictionary-based and grammar-based approaches. J Cheminform; 7:S10.

[18] Dai, H.J., Lai, P.T., Chang, Y.C. and Tsai, R.T.H. (2015). Enhancing of chemical compound and drug name recognition using representative tag scheme and fine-grained tokenization. J Cheminform, vol. 7, pp. 1-10.

[19] Lu, Y., Ji, D., Yao, X., Wei, X. and Liang, X. (2015). CHEMDNER system with mixed conditional random fields and multi-scale word clustering. J Cheminform, vol. 7, pp. 1-5.

[20] Campos, D., Matos, S. and Oliveira, J.L. (2015). A document processing pipeline for annotating chemical entities in scientific documents. J Cheminform; 7:S7.

[21] Leaman, R. and Gonzalez, G. (2008). BANNER: an executable survey of advances in biomedical named entity recognition. In: Pacific Symposium on Biocomputing, pp. 652-663.

[22] Folorunso, S.O. (2012) Theoretical Comparison of Undersampling Techniques Against Their Underlying Data Reduction Techniques, covenant University, Nigeria.

[23] Longadge, R. and Dongre, S., (2013). Class Imbalance Problem in Data Mining Review. arXiv preprint arXiv:1305.1707.

[24] Powers, D.M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.Technical Report.

[25] Kubat, M. and Matwin, S. (1997) Addressing the curse of imbalanced training sets: one-sided selection. In ICML, pp. 179-186.

[26] He, H. and Ma, Y. eds. (2013). Imbalanced learning: foundations, algorithms, and applications. John Wiley & Sons.

[27] Drummond, C. and Holte, R.C. (2003). C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In Workshop on learning from imbalanced datasets II.

[28] Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P. (2002). SMOTE: synthetic minority over-sampling technique. J ARTIF INTELL RES, pp. 321-357.

[29] McCallum, A.K. (2002). Mallet: A machine learning for language toolkit.

[30] Kittler, J., Hatef, M., Duin, R. P., & Matas, J. (1998). On combining classifiers. IEEE transactions on pattern analysis and machine intelligence, vol. 20, no. 3, pp. 226-239.

[31] Shen, H. and Sarkar, A. (2005). Voting between multiple data representations for text chunking. Springer Berlin Heidelberg, pp, 389-400.

[32] Krallinger, M., Rabal, O., Leitner, F., Vazquez, M., Salgado, D., Lu, Z., Leaman, R., Lu, Y., Ji, D., Lowe, D.M. and Sayle, R.A. (2015). The CHEMDNER corpus of chemicals and drugs and its annotation principles. J Cheminform, vol. 7, pp. 1-17.

[33] https://opennlp.apache.org/.

[34] Akkasi, A., Varoğlu, E., & Dimililer, N. (2016). ChemTok: A New Rule Based Tokenizer for Chemical Named Entity Recognition. Biomed Research International.

[35] Chemical Affixes. In Affixes: The building block of English. Retrieved from http://www.affixes.org/themes/index.html.

[36] Jessop, D.M., Adams, S.E., Willighagen, E.L., Hawizy, L. and Murray-Rust, P. (2011) OSCAR4: a flexible architecture for chemical text-mining. J Cheminform, 3:41.

[37] Huber, T., Rocktäschel, T., Weidlich, M., Thomas, P. and Leser, U. (2013). Extended feature set for chemical named entity recognition and indexing. In: BioCreative Challenge Evaluation Workshop, October p. 88.

[38] Brown, P.F., Desouza, P.V., Mercer, R.L., Pietra, V.J.D. and Lai, J.C. (1992). Class-based n-gram models of natural language. COMPUT LINGUIST, vol. 18, pp. 467-479.

نشریه هوش مصنوعی و داده کاوی

الگوی بهبود تشخیص موجودیت های اسمی شمیایی از طریق ساده سازی کاهشی مبتنی بر جمله و ترکیب طبقه بندها

عباس عکاسی[1] و اکرم وارغلو[2,*]

[1] گروه کامپیوتر، دانشگاه آزاد اسلامی واحد بندر عباس، بندر عباس، ایران.

[2] گروه مهندسی کامپیوتر، دانشگاه مدیترانه ی شرقی، فاماگوستا، قبرس شمالی.

ارسال ۲۰۱۷/۰۶/۲۵؛ بازنگری ۲۰۱۸/۰۴/۰۹؛ پذیرش ۲۰۱۸/۰۶/۰۳

چکیده:

تشخیص موجودیتهای اسمی شیمیایی پایه‌ای ترین مرحله برای عملیات مختلف مرتبط با استخراج اطلاعات مانند کشف روابط بین دارویی، استخراج اسامی مولکولها و خواص آنها و بسیاری از وظایف دیگر میباشد. بنابراین بهبود کارایی این مرحله میتواند تاثیر بسزایی در کارایی عملیات آتی مرتبط داشته باشد. متون شیمیایی از نظر تعداد کلمات مرتبط و غیر مرتبط با شیمی معمولا دارای عدم توازن هستند به این دلیل که کلمات تشکیل دهنده ی متون که مربوط به شیمی هستند از نظر تعداد بسیار محدود تر از کلمات غیر مرتبط میباشند. در این مقاله مشکل عدم توازن کلاس در کار تشخیص موجودیت‌های اسمی شیمیایی مورد مطالعه قرار گرفته و یک نسخه‌ی تغییر یافته از ساده سازی کاهشی برای استفاده در موضوع تشخیص موجودیت‌های اسمی جهت ساختن مجموعه‌ای از طبقه بندها ارائه شده است. به منظور حفظ توزیع کلاسها در داخل هر جمله، روش ساده سازی کاهشی استاندارد، به نسخه‌ی مبتنی بر جمله تغییر داده شده است. دراین رویکرد، حذف نمونه داده‌های نامربوط به طور تصادفی از داخل هر جمله مستقلا حذف میشوند در حالیکه در متد اصلی همه‌ی نمونه‌ها از تمامی جملات باهم در نظر گرفته میشوند و حذف نمونه‌های نامرتبط از مجموعه‌ی تجمیع شده اتفاق میافتد. علاوه براین به منظور بهره وری از مزایای پیش بینی‌های متفاوت؛ از یادگیری تجمیعی جهت ترکیب طبقه بندهای مختلف ساخته شده با داده‌های ساده شده بر طبق روش پیشنهادی استفاده شده است. برای انجام آزمایشات از مجموعه داده‌ی ارائه شده توسط BioCreative IV برای رقابت محققین در حوزه ی تشخیص موجودیت‌های اسمی شیمیایی استفاده شده است.

کلمات کلیدی: تشخیص موجودیتهای اسمس شیمیایی، مشکل عدم توازن کلاس، ترکیب طبقه بندها، نمونه برداری تصادفی.