International Journal on Advanced Science Engineering Information Technology

Vol.8 (2018) No. 5
ISSN: 2088-5334

Comparative Analysis of Different Data Representations for The Task of Chemical Compound Extraction

Basel Alshaikhdeeb[#], Kamsuriah Ahmad[#]

[#] Center for Software Technology and Management, Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia 43600 Bangi, Selangor Darul Ehsan, Malaysia
Email: shaikhdeeb@gmail.com, kamsuriah@ukm.edu.my

Abstract—**Chemical Compound Extraction refers to the task of recognizing chemical instances such as oxygen nitrogen and others. The majority of studies that addressed the task of chemical compound extraction used machine-learning techniques. The key challenge behind using machine-learning techniques lies in employing a robust set of features. The literature shows that there are numerous types of features used in the task of chemical compound extraction. Such dimensionality of features can be determined via data representation. Some researchers have used N-gram representation for biomedical named entity recognition, where the most significant terms are represented as features. Meanwhile, others have used detailed-attribute representation in which the features are generalized. As a result, identifying the best combination of features to yield high-accuracy classification becomes challenging. This paper aims to apply the Wrapper Subset Selection approach using two data representations—N-gram and detailed-attributes. Since each data representation would suit a specific classification algorithm, two classifiers were utilized—Naïve Bayes (for detailed-attributes) and Support Vector Machine (for N-gram). The results show that the application of feature selection using detailed-attributes outperformed that of N-gram representation by achieving a 0.722 f-measure. Despite the higher classification accuracy, the selected features using detailed-attribute representation have more meaning and can be applied for further datasets.**

Keywords—**chemical compounds extraction; data representation; N-gram; detailed-attributes; Naïve Bayes; support vector machine; attribute selection**

I. INTRODUCTION

The dramatic growth of information over the web nowadays has posed several challenging issues [1]. One such issue is the recognition of biomedical entities from text. Chemical compound extraction refers to the task of recognizing chemical entities such as oxygen and nitrogen and others [2]. The majority of research studies addressing the task of chemical compound extraction utilize machine-learning techniques. This is because the technique enables the discrimination of the occurrence of such entities [3]. However, machine-learning techniques are highly impacted by the utilized features, where a set of robust features would significantly improve the accuracy of extraction [4]. Some studies have investigated the taxonomy of features used in chemical compound extraction tasks [5]. In general, there are three main categories for these features. The first consists of morphological features, which are related to the spelling system of the chemical instances. The second involves the syntactic features, which relate to the grammatical aspect behind the chemical entities (e.g., noun, adjective, etc.). The third is dictionary-based features, which are related to predefined instances such as abbreviations or molecular formula.

Several researchers have examined the categories mentioned above. For example, Rocktaschel et al. [6] used the Conditional Random Fields (CRF) classification method with dictionary-based features for biomedical entity recognition based on an SCAI dataset. Similarly, Lamurias et al. [7] used dictionary-based features for biomedical entity recognition in an ontology. The ontology was employed to address the semantic similarity between entities. CRF was also employed as a classification method to classify the entities.

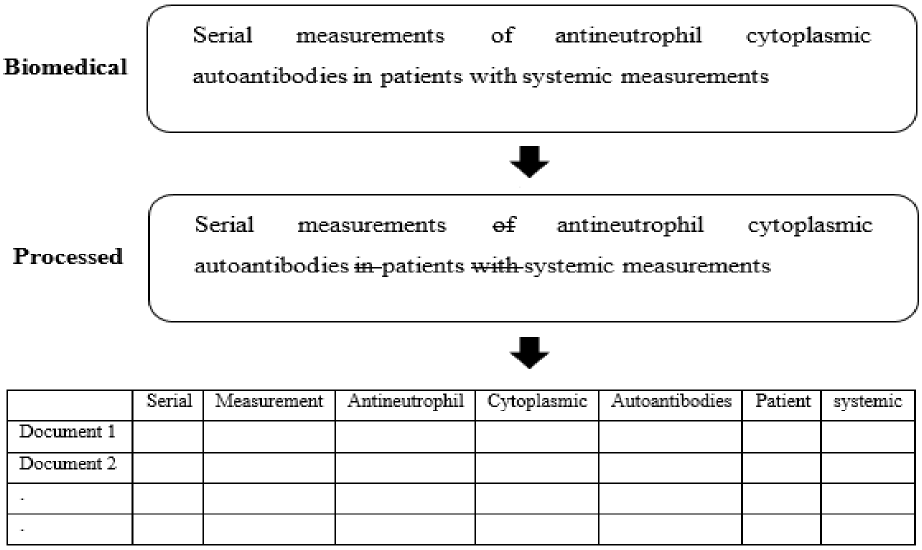
On the other hand, Basista-Navarro et al. [8] used a combination of different features including morphological and dictionary-based features. Within the morphological features, the authors utilized Greek letters, punctuations, and digits. Similar to the latter studies, the authors also used CRF as a classification method.

Finally, Usie et al. [9] also used a combination of features including morphological and dictionary-based features. In particular, the morphological features used in their study were built using a regular expression.

As noted from the literature, the feature space can be seen as highly dimensional. Therefore, it is imperative to carry out a feature reduction task in order to identify the best combinations for these features.

There are two main categories of feature selection including filter-based (ranking) methods and wrapper-based methods [10]. Filter-based methods aim at utilizing a ranking mechanism over the attributes. This ranking mechanism aims to examine the usefulness of every attribute regarding classification. Contrary to this, wrapper-based methods utilize a classifier to identify the best attributes. In this manner, the classifier would act as an evaluation metric for each attribute. According to Inza et al. [11], wrapper-based methods yield better performance compared to filter-based ones. Also, the filter-based method suits specific cases, where the aim is to find the best N features or attributes. Since our study mainly focuses on identifying the most appropriate set of features for the process of BNER, the wrapper-based method is considered the best method and therefore adopted for this study.

Despite the usefulness of feature selection methods, there are different possible representations for data. One of the standard data representations used for text classification is the N-gram. This representation aims to turn the most significant tokens or grams as features [12]. This can be performed by bringing all the terms in a particular dataset and then eliminating unnecessary ones such as redundant terms, stopwords, numbers, and punctuation. The remaining terms will be employed as features in which the representation of data would be described as present and absent. Figure 1 depicts this type of representation.



Biomedical

Serial measurements of antineutrophil cytoplasmic autoantibodies in patients with systemic measurements

Processed

Serial measurements ~~of~~ antineutrophil cytoplasmic autoantibodies ~~in~~ patients ~~with~~ systemic measurements

	Serial	Measurement	Antineutrophil	Cytoplasmic	Autoantibodies	Patient	systemic
Document 1							
Document 2							
.							
.							

Fig. 1. N-gram representation

Also, the data can be represented using detailed features in which the features can be generalized such as capitalization, suffixes, prefixes, and others. Table 1 shows a sample of this type of representation.

TABLE I
A SAMPLE OF DETAILED FEATURE REPRESENTATION

Token	**Detailed Feature 1 (e.g., Length)**	**Detailed Feature 2 (e.g. Prefix)**	**Detailed Feature 2 (e.g., Suffix)**	**……**	**Detailed Feature *n* (e.g., Frequency)**
Antineutrophil	14	Anti	phil	……	0.347
Cytoplasmic	11	Cyto	mic	……	0.234
Hydroxyalkyl	12	Hydro	kyl	……	0.624
Lipoxygenase	12	Lip	nase	……	0.261

The N-gram representation is usually used with classifiers such as the Support Vector Machine, where the features are numerous and represented in a vector space. Meanwhile, the detailed feature representation is usually utilized with a Naïve Bayes classifier in which the features are nominal and limited to a specific range. This study aims to compare both representations with the two classifiers using the wrapper-based feature selection, to determine the most appropriate set of features.

The paper is structurally organized as follows: Section II highlights the proposed method and its phases; Section III shows the experimental results and includes a discussion with an analysis of the results; finally, Section IV provides the conclusion to this study.

II. Material and Method

As shown in Figure 2, the proposed method in this study consists of four phases. The first phase is related to the data that will be used in the experiments, which contains biomedical instances, specifically, chemical compounds. The second phase is feature extraction, where the features are extracted and represented using two paradigms including the detailed-attributes and N-gram representations. The third phase is the classification process, where two classifiers—NB and SVM—are utilized for the detailed-attribute representation and N-gram representation, respectively. The fourth phase concentrates on feature selection, which will be conducted using the wrapper-based approach. The next section illustrates these phases in further detail.

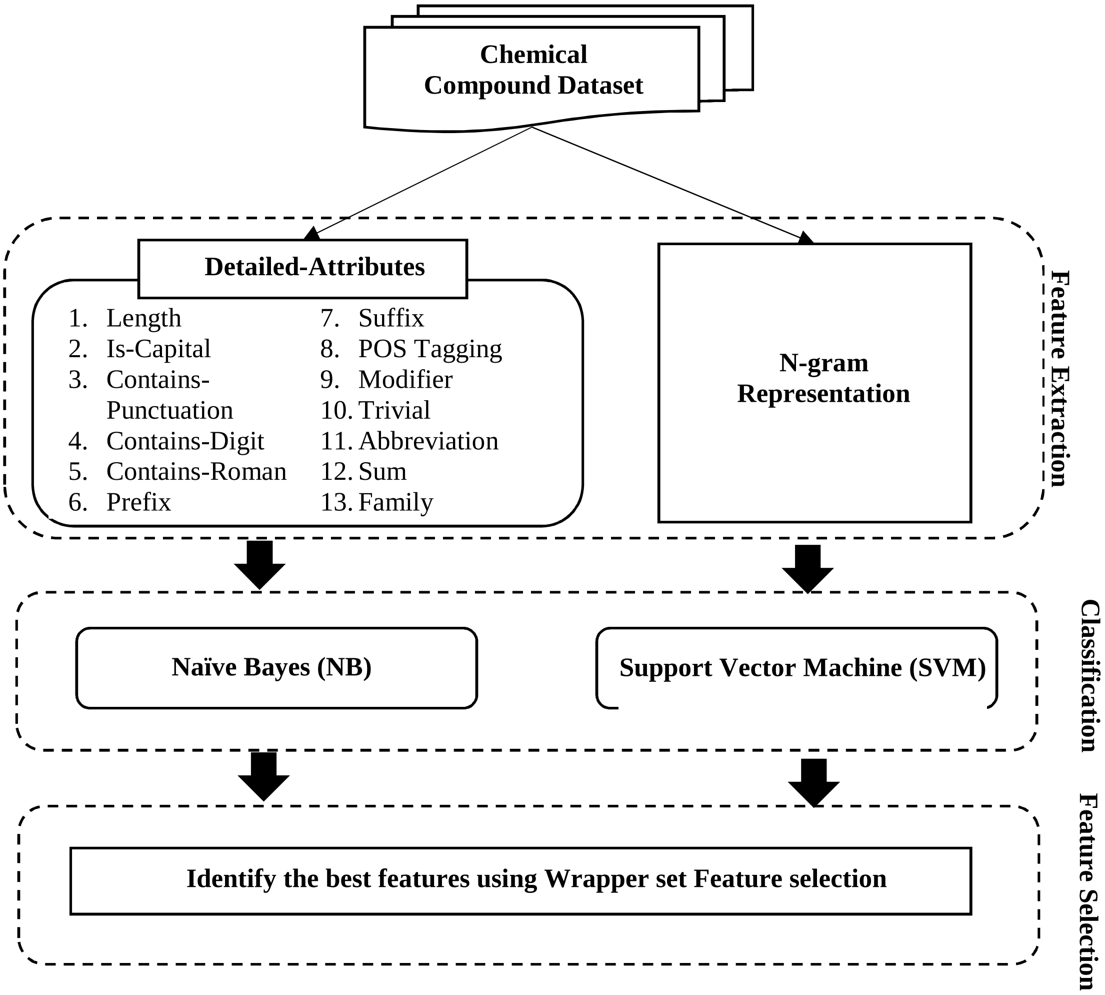


Fig. 2. The components of the proposed method

A. Chemical Compound Dataset

This study uses a SCAI dataset, which was introduced by Kolarik et al. [13]. The dataset contains vast amounts of labeled chemical compounds. Table 2 shows a snippet of this dataset.

TABLE II
SCAI SNIPPET

Token	Class
The	|O
synthesis	|O
of	|O
a	|O
range	|O
of	|O
hydroxy	|IUPAC

From Table 2, it can be seen that the data contains a series of tokens, in which every token is represented in each line. Every token has a class label, either '|O', which refers to the regular tokens, or '|IUPAC', which refers to chemical entities.

B. Feature Extraction

This phase aims to extract the features and represent them in particular data representation. Two paradigms are used, which are the detailed-attributes and N-gram representations. The two paradigms are discussed in the following sub-sections.

1) Detailed-Attributes: In this paradigm, the feature space is articulated generally, in which several features can be generalized in one category; for example, the occurrence of underscore, dash and period can be generalized as a punctuation feature. The features that will be used under this paradigm can be stated as follows:

2) Length: This feature examines the number of characters within the token. This can be an indicator since the majority of the chemical compounds tend to be longer [14]. In this study, the length was implemented via the setting of two thresholds as 5 and ten characters. In this manner, the tokens that have lengths of less than five would be considered short, the tokens that have a length between 5 and ten characters would be considered average, and tokens with lengths more than ten characters would be considered long.

3) Is-Capital: This feature examines the case of the token—whether Capital (e.g., Oxadiazole), Upper (e.g., CHLORO), or lower (e.g., furanyl). To implement this feature, the regular expression was used.

4) Contains-Digit: This feature examines the occurrence of digits, whether Digit-only (e.g., 612), No-digit (e.g., allopurinol), or hybrid (e.g., 4-chloro). To implement this feature, the regular expression was used.

5) Contains-Punctuation: This feature examines the occurrence of special characters such as brackets or separators, where the cases could be Punctuation-only (e.g. "-"), No-punctuation (e.g., pyrimidine), or hybrid (e.g., 8-benzylidene). To implement this feature, the regular expression was used.

6) Contains-Roman: This feature examines the occurrence of Roman characters such as 'XI,' 'II,' and 'III,' where the examination of these features would be emphasized as true or false. To implement this feature, the regular expression was used.

7) Prefixes: This feature examines the occurrence of letters in the beginning that mutually appear with the chemical compounds such as 'iso' in "isopropylidene," "isoprenoids," and "isonipecotic." To implement this feature, a predefined list of prefixes was used.

8) Suffixes: this feature examines the occurrence of the ending letters that mutually appear in the chemical compounds such as 'ane' in "diaminoheptane," "dioxane," and "aminopropane." To implement this feature, a predefined list of suffixes was used.

9) Part-Of-Speech Tagging: This feature examines the grammatical aspect of the token, whether it is a verb, noun, or adjective. To implement this feature, the Stanford POS tagging was used [15].

10) Modifier: This feature examines the occurrence of certain keywords that mutually appear before or after the chemical compound. For example, the phrase 'tryptophan derivatives' contain keywords that are 'derivatives'; this keyword frequently occurs after chemical instances. To implement this feature, a predefined list of modifiers was used.

11) Abbreviation: This feature examines the occurrence of abbreviated chemical compounds such as 'dme,' which denotes 'dimethoxyethane' [1]. To implement this feature, a predefined list of abbreviations was used.

12) Trivial: This feature examines the occurrence of specific chemical compounds by their trivial name or company-code. For example, Ethyl is the trivial name for Ethanol. To implement this feature, a predefined list of trivial instances was used.

13) Sum: This feature examines the occurrence of the molecular formula of a particular chemical compound such as CO_2 for Carbon dioxide. To implement this feature, a predefined list of molecular formulae was used.

14) Family: This feature examines the occurrence of a broader class of multiple chemical compounds such as alcohol, which is considered to be a broader class of different compounds like methanol. To implement this feature, a predefined list of family instances was used.

15) N-gram: In this paradigm, the most significant terms within the dataset's tokens are represented as attributes or features. In order to do so, the data undergoes multiple processing tasks. Firstly, the data is processed to remove special characters, digits, and stopwords. Secondly, the remaining tokens are stemmed using a Porter stemmer [16]. Stemming removes derivations such as 'ing', 'es', 'ed', and others. Finally, the duplicated tokens resulting from the stemming task (e.g. modifies → modify and modification → modify) are removed. The result of this paradigm is that only the most important terms will appear in the dataset. In this study, the final terms amounted to 465 terms. Table 3 shows a sample of these terms.

TABLE III
SAMPLE OF N-GRAM RESULTS

Significant Terms
Trypto
Dihydro
Hetero
Cetate
Hydroxy
Dipep

C. Classification

In this phase, two classification algorithms are used, which are Naïve Bayes and Support Vector Machine. The NB classifier will be applied to the detailed-attribute representation. Meanwhile, the SVM classifier will be applied to the N-gram representation. This is due to each of these classifiers requiring specific data representations that suit their capability. Both classifications have been adjusted to train on 80% of the data and tested on the remaining 20% of the data. The following sub-sections illustrate the mechanism of each classifier.

2) Naïve Bayes: Naïve Bayes is a machine learning classification method that mainly depends on a statistical model. The idea behind this classifier lies in examining each feature independently by the classes [17]. Hence, the Naïve Bayes classifier attempts to predict a class using Equation (1) [18]:

$$PredictedClass = Max\ P(Ci) \qquad (1)$$

Where Max P(Ci) is the most probable class label. In order to calculate the probability of the classes, Equation (2) is applied:

$$P(Ci) = P(Fj \mid Ci) \qquad (2)$$

Where P(Fj | Ci) is the probability of every feature along with each class label.

3) Support Vector Machine: SVM is one of the classifiers that utilizes the vector space technique in which the features are represented in 2-D via X and Y axes [19]. The values of the features that will be used for representation in the 2-D space are considered to be the occurrence of each term in accordance to the dataset. Once the features are depicted in the vector space, a hyperplane, which is a margin that separates the data into two classes, will be implemented. Accurate acquisition of the hyperplane will lead to accurate classification results. The hyperplane can be calculated based on Equation (3):

$$f(\vec{x}) = \begin{cases} +1: & (\vec{x} \times \vec{w}) + b > 0 \\ -1: & Otherwise \end{cases} \qquad (3)$$

The SVM model adjusts to the most accurate hyperplane that has the greatest margin. One example of this is the chemical and non-chemical data instances that are divided by a hyperplane in which the shortest path is between the nearest chemical instance and nearest non-chemical instance [20].

D. Feature Selection

This phase applies the feature selection, whereby the most appropriate features will be identified. Hence, the Wrapper Subset Selection (WSS) approach was adopted. This approach is based on a wrapping mechanism in which a search will be performed to find the most robust subset within the featured space [21]. WSS employs a classification method to assess the effectiveness of each feature. Therefore, this study will integrate both SVM and NB with WSS in order to measure the accuracy of each combination of features.

To describe the problem of dimensionality in the chemical compound extraction task, a chemical data D is considered, which consists of sequences $D = \{t_1, t_2, t_3, \dots, t_m\}$, where every token denotes a term within the data. The term is either a regular term or a chemical compound. Evidently, for every token there are different features that correlate with it $f = \{f_1, f_2, f_3, \dots, f_n\}$. In this manner, every feature should be assessed separately to obtain the best combination. However, evaluating each feature separately may lead to numerous possibilities. The single evaluation required to specify the number of combination of features are bi-combination (e.g. the combination of f_1 and f_2 or the combination of f_1 andf_3), tri-combination (e.g. the combination of f_1, f_2 andf_3) or even any number of possible combinations ranging from 1 to n, where n represents the number of features. In this manner, the problem can be formulated based on Equation (4):

$$\sum_{n=13} \frac{n!}{(n-r)! \times r!} \qquad (4)$$

Where *n* is the number of features and *r* is the number of combinations. The number of utilized features in the detailed representation is 13, which seems to be small. However, examining every possibility of each possible combination would be tedious. Table 4 shows the number of possibilities for each combination.

TABLE IV
NUMBER OF POSSIBILITIES FOR EACH COMBINATION

Number of combinations	Number of possibilities
r = 1	13
r = 2	78
r = 3	286
r = 4	715
r = 5	1287
r = 6	1716
r = 7	1716
r = 8	1287
r = 9	715
r = 10	286
r = 11	78
r = 12	13
r = 13	1
Total	8191

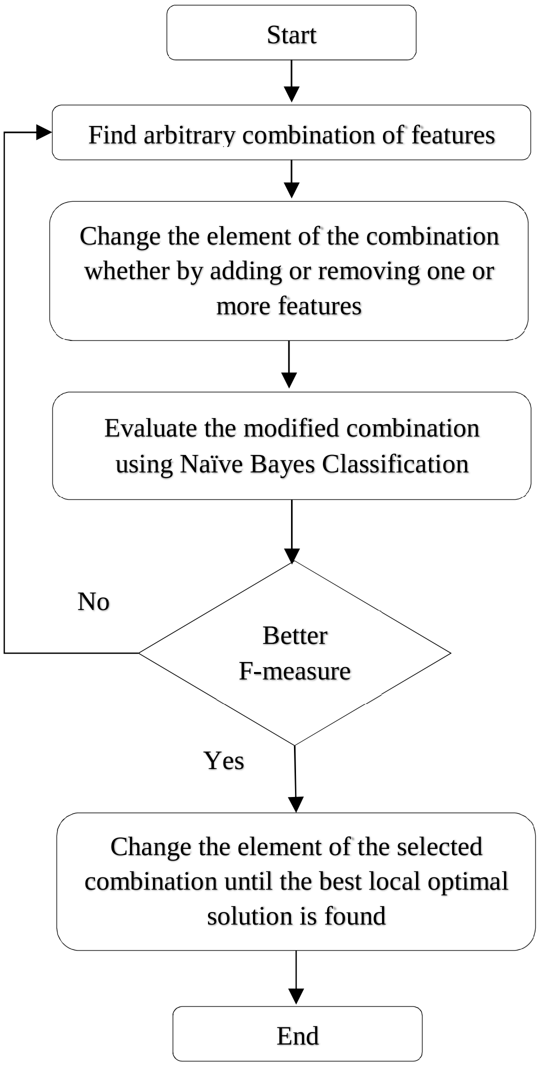

Start
Find arbitrary combination of features
Change the element of the combination whether by adding or removing one or more features
Evaluate the modified combination using Naïve Bayes Classification
No
Better F-measure
Yes
Change the element of the selected combination until the best local optimal solution is found
End

Fig. 3. HC algorithm flowchart

As shown in Table 4, the total number of possibilities for individual combinations is 8191. Examining each possibility separately would prove tedious, especially when the computation for an individual run is time-consuming. In the same manner, examining the possibilities for the N-gram, which contains 465 features, would increase the problem of dimensionality. Therefore, it is necessary to apply feature reduction.

It is important to note that the search algorithm used in our study is Hill Climbing. Hill Climbing (HC) is a heuristic search algorithm that seeks to find nearly optimized solutions [22]. HC is a local search algorithm that has been used on hard optimization problems. A key characteristic of the local search algorithm is that it can be applied on problems that require finding a solution with the maximized criterion among a number of candidate solutions [23]. Local search algorithms work by moving from one solution to another in the search space through making some local changes until the optimal solution is found.

Similarly, HC begins with an arbitrary solution, then tries to figure out a better solution by incrementally changing the elements of the solution [24]. The flowchart for the HC algorithm is depicted in Figure 3.

III. Results and Discussion

As with any machine-learning task, the evaluation will be conducted using precision, recall, and f-measure. Also, the evaluation will be based on two paradigms; the detailed-attribute using NB and the N-gram using SVM. The following sub-sections show the results obtained in this study.

1) Results of Detailed-attribute using NB: As mentioned earlier, this section shows the results of applying the NB classifier with the detailed-attribute paradigm. The features are evaluated separately and with the total combination of features. Table 5 shows the results.

TABLE V
Results of NB with detailed-attribute

Feature	Precision	Recall	F-measure
Length	0.4755	0.5	0.48747
IsCapital	0.4755	0.5	0.48747
ContainsDigit	0.4755	0.5	0.48747
ContainsPunctuation	0.4755	0.5	0.48747
ContainsRoman	0.4755	0.5	0.48747
Prefixes	0.7422	0.6186	**0.6561**
Suffixes	0.5694	0.5949	**0.5792**
POS tagging	0.6563	0.6202	**0.6354**
Modifier	0.6377	0.5188	0.5241
Abbreviation	0.7256	0.5020	0.4916
Trivial	0.9756	0.5021	0.4917
Sum	0.4755	0.5	0.48747
Family	0.7411	0.5181	0.5229
Total	0.6488	0.6606	0.6544

Table 5 shows that prefix, POS and suffix obtained the greatest f-measure values. This denotes the importance of these features in extracting chemical compounds.

On the other hand, even though morphological features (i.e., F1 to F5) and dictionary features (i.e., F9 to F13) have yielded lower performance, different studies have suggested that these features be combined with other features to yield reasonable performance [5]. The total combination of all features has shown similar performance to that of the independent use of the prefix (i.e., around 0.65).

2) Results of N-gram using SVM: Also, the results of applying SVM with the N-gram are depicted in Table 6.

TABLE VI
Results of SVM with N-gram

Features	Precision	Recall	F-measure
465 terms	0.716	0.694	0.704

As shown in Table 6, the results of applying SVM are 0.716 for precision, 0.694 for recall, and 0.704 for the f-measure. It is evident that the results of applying the SVM with N-gram have outperformed the results of applying NB with detailed-attributes. This is proven by the 0.704 f-measure achieved by SVM and 0.654 achieved by NB with all features. This outperformance can be justified from the numerous features used in the SVM paradigm (i.e., 465 features) compared to the 13 features used by NB.

3) Results of applying the WSS feature selection: This section highlights the results of applying the WSS feature selection for both paradigms—SVM with N-gram and NB with the detailed-attribute. Table 7 depicts the results.

As shown in Table 7, the results of applying the feature selection on SVM with N-gram has led to the selection of 100 features with an f-measure of 0.718. In contrast, the results of applying the feature selection on NB with detailed-attributes have led to 4 features with an f-measure of 0.722. It is clear that the detailed-attribute representation has outperformed the N-gram representation regarding classification accuracy. Also, the selected features of the N-gram representation can be depicted as meaningless terms. Comparatively, the selected features of the detailed representation tend to be more generalized. This can facilitate the process of applying the selected features to new datasets to achieve higher accuracy.

As shown in Table 7, the results of applying the feature selection on SVM with N-gram has led to the selection of 100 features with an f-measure of 0.718. In contrast, the results of applying the feature selection on NB with detailed-attribute has led to 4 features with an f-measure of 0.722. It is clear that the detailed-attribute representation has outperformed the N-gram representation regarding classification accuracy. Also, the selected features of the N-gram representation can be depicted as meaningless terms. Comparatively, the selected features of the detailed representation tend to be more generalized. This can facilitate the process of applying the selected features to new datasets to achieve higher accuracy. To compare the acquired results with the related works, it is evident that NB with detailed-attribute showed superior performance by acquiring a 72.2% f-measure compared to the work of Rocktaschel et al. [6], which used the SCAI dataset, acquiring a 63% f-measure, and Usie et al. [9], which used the same dataset, and acquired a 68% f-measure.

TABLE VII
RESULTS OF APPLYING WSS FEATURE SELECTION

Paradigm	Selected Features	No. of features	Precision	Recall	F-measure
SVM with N-gram	hyd, py, carb, dime, nitr, trypto, but, meth, acy, sulf, dipep, dihydro, benz, mono, trii, iso, palm, iod, naph, etho, niso, testo, tyr, threo, cyclo, chol, prop, deox, uri, flu, adria, alka, glu, trig, ethy, nucl, xyl, phth, oxo, pip, brom, thio, acid, aden, dini, hetero, tamox, lact, cefo, tazo, allop, augus, yl, xy, ones, one, in, cin, mino, cetate, lic, yla, ic, phene, ium, sium, ine, chlor, ene, ide, ate, pril, lix, cid, rile, am, MD, VBZ, JJR, RP, CD, NNPS, PRP, WDT, NNS, JJ, qutation, EX, CC, VBG, POS, :, -RRB-, VBN, VB, NNP, DT, JJS, fullstop, QotationItalic	100	0.7545	0.698	0.718
NB with detailed-attributes	Contains-Digit, Prefix, POS tagging, Trivial	4	0.703	0.745	0.722

IV. CONCLUSION

This paper conducted a comparative study between two data representations—N-gram and detailed-attribute. N-gram was used with a SVM classifier, while the detailed-attribute was used with a NB classifier. Both data representations underwent a feature selection using the WSS approach. The results show that the detailed-attribute with NB yielded superior performance by achieving a 72.2% f-measure. For future researches, it is highly recommended that new data representations such as word embedding be applied and the results examined.

ACKNOWLEDGMENTS

Universiti Kebangsaan Malaysia supported this research.

REFERENCES

[1] Basel Alshaikhdeeb and Kamsuriah Ahmad, "Integrating correlation clustering and agglomerative hierarchical clustering For holistic schema matching," Journal of Computer Science, vol. 11, p. 484, 2015.

[2] B. Alshaikhdeeb and K. Ahmad, "Feature selection for chemical compound extraction using wrapper approach with Naive Bayes classifier," in 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI), 2017, pp. 1-6.doi:10.1109/ICEEI.2017.8312421.

[3] Yaoyun Zhang, Jun Xu, Hui Chen, Jingqi Wang, Yonghui Wu, Manu Prakasam, and Hua Xu, "Chemical named entity recognition in patents by domain knowledge and unsupervised feature learning," Database, vol. 2016, p. baw049, 2016.

[4] Baydaa Hashim and Nazlia Omar, "A Back Propagation Neural Network for Identifying Multi-Word Biomedical Named Entities," 2016, vol. 11, 2016.doi:682-690 http://www.praiseworthyprize.org/jsm/index.php?journal=irecos&amp;page=article&amp;op=view&amp;path%5B%5D=19206.

[5] Basel Alshaikhdeeb and Kamsuriah Ahmad, "Biomedical Named Entity Recognition: A Review," International Journal on Advanced Science, Engineering and Information Technology, vol. 6, 2016.

[6] Tim Rocktäschel, Michael Weidlich, and Ulf Leser, "ChemSpot: a hybrid system for chemical named entity recognition," Bioinformatics, vol. 28, pp. 1633-1640, 2012.

[7] Andre Lamurias, Tiago Grego, and Francisco M Couto, "Chemical compound and drug name recognition using CRFs and semantic similarity based on ChEBI," in BioCreative Challenge Evaluation Workshop, 2013, p. 75.doi.

[8] Riza Batista-Navarro, Rafal Rak, and Sophia Ananiadou, "Optimising chemical named entity recognition with pre-processing analytics, knowledge-rich features, and heuristics," J Chem Inf, vol. 7, p. S6, 2015.

[9] Anabel Usié, Joaquim Cruz, Jorge Comas, F Solson, and Rui Alves, "CheNER: a tool for the identification of chemical entities and their classes in biomedical literature," J Cheminform, vol. 7, p. S15, 2015.

[10] Haider Banka and Suresh Dara, "A Hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification, and validation," Pattern Recognition Letters, vol. 52, pp. 94-100, 2015/01/15/ 2015.doi:https://doi.org/10.1016/j.patrec.2014.10.007 http://www.sciencedirect.com/science/article/pii/S0167865514003146

[11] Iñaki Inza, Pedro Larrañaga, Rosa Blanco, and Antonio J Cerrolaza, "Filter versus wrapper gene selection approaches in DNA microarray domains," Artificial intelligence in medicine, vol. 31, pp. 91-103, 2004.

[12] Robert Leaman, "Advancing biomedical named entity recognition with multivariate feature selection and semantically motivated features," Arizona State University, 2013Retrieved from.

[13] Corinna Kolárik, Roman Klinger, Christoph M Friedrich, Martin Hofmann-Apitius, and Juliane Fluck, "Chemical names: terminological resources and corpora annotation," in Workshop on Building and evaluating resources for biomedical text mining (6th edition of the Language Resources and Evaluation Conference), 2008.

[14] E Alharbi and S Tiun, "A Hybrid Method of Linguistic Features and Clustering Approach for Identifying Biomedical Named Entities," Asian Journal of Applied Sciences, vol. 8, pp. 210-216, 2015.

[15] Stanford, "Part-of-Speech Tagger," ed, 2014.

[16] Peter Willett, "The Porter stemming algorithm: then and now," Program, vol. 40, pp. 219-223, 2006.

[17] Bo Tang, Steven Kay, and Haibo He, "Toward optimal feature selection in naive Bayes for text categorization," 2016.

[18] Songbo Tan, Xueqi Cheng, Yuefen Wang, and Hongbo Xu, "Adapting naive bayes to domain adaptation for sentiment analysis," in Advances in Information Retrieval, ed: Springer, 2009, pp. 337-349.

[19] Ahmed Almusawi and Haleh Amintoosi, "DNS Tunneling Detection Method Based on Multilabel Support Vector Machine," Security and Communication Networks, vol. 2018, 2018.

[20] Samaneh Moghaddam and Martin Ester, "AQA: aspect-based opinion question answering," in Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on, 2011, pp. 89-96.doi.

[21] Suzanne Little, Ovidio Salvetti, and Petra Perner, "Evaluation of feature subset selection, feature weighting, and prototype selection for biomedical applications," Advances in Case-Based Reasoning, pp. 312-324, 2008.

[22] Yuri Bykov and Sanja Petrovic, "A Step Counting Hill Climbing Algorithm applied to University Examination Timetabling," Journal of Scheduling, vol. 19, pp. 479-492, 2016.

[23] Ruizhi Li, Shuli Hu, Yiyuan Wang, and Minghao Yin, "A local search algorithm with tabu strategy and perturbation mechanism for generalized vertex cover problem," Neural Computing and Applications, vol. 28, pp. 1775-1785, 2017.

[24] Ivan Piza-Davila, Guillermo Sanchez-Diaz, Manuel S Lazo-Cortes, and Luis Rizo-Dominguez, "A CUDA-based Hill-climbing Algorithm to Find Irreducible Testors from a Training Matrix," Pattern Recognition Lett*ers*, 2017.