A Comparative Analysis of Chemical Named Entity Recognition Using Support Vector Machines

Samaneh Azari

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Computer Engineering

Eastern Mediterranean University
September 2013
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Computer Engineering.

Assoc. Prof. Dr. Muhammed Salamah
Chair, Department of Computer Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Computer Engineering.

Assoc. Prof. Dr. Ekrem Varoğlu
Supervisor

Examining Committee

1. Prof. Dr. Hakan Altınçay ______________________________

2. Assoc. Prof. Dr. Ekrem Varoğlu ______________________________

3. Asst. Prof. Dr. Nazife Dimililer ______________________________

ABSTRACT

Cheminformatics is the synthesis of computer science and chemistry to collect knowledge about chemicals to provide useful information for drug development. Chemical named entity recognition (CHEM-NER) is the crucial first step to extract useful information from chemical publications and patents. In this dissertation, a classification system based on support vector machine (SVM) which uses wrapper based feature subset selection algorithms is proposed for the CHEM-NER task. The SVM classifier for recognizing chemical named entities needs training and evaluation corpora. Three different standard chemical corpora which contain different number of classes have been used to address the binary-class and multi-class classification problems. Wrapper based feature subset selection algorithms such as Forward Selection, Backward Selection and Simplified Forward Search are used in an attempt to find the most relevant subset of features among several features. The features used include several variations of morphological features, lexical features, orthographic features and spaces. The aim of these experiments is to investigate the classification performance using different subsets of features as well as discovering the most relevant corpus among the available corpora for CHEM-NER task. The results show that in general Forward Search algorithm is more successful in selecting the most suitable subset of features for the CHEM-NER task in terms of F-score measure.

Keywords: Chemical Named Entity Recognition, Feature Extraction, Wrapper Based Feature Subset Selection, Support Vector Machines, Text Mining.

ÖZ

Kemoinformatik, ilaç yapımında kullanılmak üzere kimyasallar hakkında gerekli bilgiyi elde etmek için bilgisayar bilimleri ve kimya anabilim dallarının sentezlenmesi ile ortaya çıkan bir alandır. Kimyasal İsimlendirilmiş Nesne (KİN) tanımı kimya alanında yapılan yayınlardan ve patentlerden bilgi çıkarmanın ilk adımını oluşturur. Bu tezde KİN için Vektör Destek Makineleri (VDM) tabanlı ve sarıcı yöntemlerine dayalı öznitelik alt kümesi seçme algoritmaları kullanılan bir sınıflandırıcı sistemi önerilmiştir. Kimyasal isimlendirilmş nesneleri tanımlamak için kullanılacak VDM sınıflandırıcısını eğitmek ve sistemin başarımını ölçmek için derlemlere ihtiyaç vardır. Bu çalışmada iki-sınıf ve çok-sınıf sınıflandırıcı problemlerini incelemek adına farklı sayıda sınıflar içeren üç farklı kimyasal isimler içeren derlem kullanılmıştır. Eniyi öznitelik alt kümesini elde edbilmek için sargı yöntemine dayalı algoritmalar olarak İleri Seçim, Geri Seçim ve Basitleştirilmiş İleri Seçim algoritmaları kullanılmıştır. Kullanılan öznitelikler çeşitli yapılarda morfolojik, sözlüksel, ortografik ve boşluklardan oluşmaktadır. Bu çalışmada yapılan deneylerin amacı farklı öznitelik alt kümeleri kullanılarak elde edilecek sınıflandırıcı başarılarını incelemenin yanısıra KİN için varolan en uygun derlemi ortaya çıkarmaktır. Sonuçlar İleri Seçim algoritmasının sınıflandırma başarımını en etkin şekilde artıran öznitelik kümesini göstermiştir.

Anahtar Kelimeler: Kimyasal İsimlendirilmiş Nesne Tanımı, Öznitelik Çıkarma, Sarıcı Yöntemlerine Dayalı Öznitelik Alt Kümesi Seçme, Vektör Destek Makineleri, Metin Madenciliği.

To my Mother

For her devotion to her children

ACKNOWLEDGMENT

First of all, I would like to acknowledge my supervisor Assoc. Prof. Dr. Ekrem Varoğlu for his invaluable supervision, his knowledge and continuous encouragement and motivation. His excellent guidance and encouragement made me interested in bioinformatics and cheminformatics.

I would like to extent my gratitude to Prof. Dr. Hakan Altınçay and Asst. Prof. Dr. Nazife Dimililer for their patient and careful review and useful comments on my work and their contributions as members of dissertation defense committee.

Also I would like to thank my family, my brother and my sister in low for providing the motivation and encouragement for pursuing my Master degree. Finally, my deepest appreciation goes to my mother who has supported me financially, emotionally and morally. It would have been impossible for me to accomplish this work without her blessings, encouragement and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF LIST OF SYMBOLS/ABBREVIATIONS

ABBR.	ABBREVIATION Class in the SCAI Corpus
Bio-NER	Biomedical Named Entity Recognition
BS	Backward Selection
CHEBI	Chemical Entities of Biological Interest
CHEM-NER	Chemical Named Entity Recognition
CRF	Conditional Random Fields
CV	Cross-Validation
e_i	i^{th} classifier
FN	False Negative
FP	False Positive
FS	Forward Selection
HMM	Hidden Markov Model
IE	Information Extraction
IR	Information Retrieval
IUPAC	International Union of Pure and Applied Chemistry
SFS	Combination of the K top high classification performance features
KDD	Knowledge Data Discovery

M	Total number of class labels
ME	Maximum Entropy Model
ML	Machine Learning
NE	Named Entity
NER	Named Entity Recognition
NLP	Natural Language Processing
NP	Noun Phrase
POS	Part of Speech
PPI	Protein-protein Interaction
SB	Single Best feature with the highest classification performance among different features
SCAI	Fraunhofer Institute for Algorithms and Scientific Computing
SVM	Support Vector Machine
TM	Text Mining
TN	True Negative
TP	True Positive
Yamcha	Yet Another Multipurpose Chunk Annotator
Φ	Kernel Function

Chapter 1

INTRODUCTION

1.1 Background

Data mining is the process of exploration and analysis of large quantities of data to discover knowledge and find interesting patterns and rules by using automatic or semi-automatic methods [1]. Data mining algorithms have been quite successful on numerical and structured data, but it becomes less successful when it comes to revealing textual information. With great amount of literature and publication available as scientific papers, academic articles, journals and patents, there is a need to use functional tools to exploit the information contained in textual documents. Text Mining has emerged to deal with unstructured natural language documents to extract new, unseen and specific information, such as discovery of patterns, associations and relationships among entities in the text [2]. Typical text mining tasks include text categorization, clustering, information extraction, exploratory data analysis, document summarization, and entity relation identification.

Although text mining is used to handle text and it sounds to be similar to an advanced search engine methodology, it is highly different from the latter. Search engines are information retrieval systems that retrieve information from the vast amount of web pages that already exist. But they are not able to reveal any knowledge from the text. So in such cases text mining is applied to define relationships between different keywords by using methods such as concept clustering, indexing, association, feature extraction, and information visualization. Different applications of text mining include: security applications [3], biomedical applications [4][5], software and applications, online media applications, business and marketing applications [6][7], sentiment analysis [8], academic and research applications.

Natural Language Processing (NLP) is the use of computer science and artificial intelligence techniques applied for discovery of interactions between computers and human languages. NLP aims to extract a comprehensive meaningful representation from free text, so NLP techniques can be used roughly in text mining.

Prior implementations of NLP systems were based on complex sets of hand-written rules and grammar based approach which provided slow and ineffective systems [9]. Introducing statistical and probabilistic models and machine learning algorithms lead to an evolution in natural language processing. Such models are more robust when confronted with real input data that contain error, and more reliable when included as a component of a larger system unfortunately they depend on specifically developed corpora, which have been hand-annotated with the correct values.

Bioinformatics is an interdisciplinary field which is the application of computer science and informational technology applied on molecular biology and medicine. Recently, with the growing amount of publications in biomedical domain especially in the field of genetics and genomics, collecting, retrieving and establishing data to extract meaningful and useful knowledge has become a cumbersome task. So, biomedical text mining (BioNLP) is applied to text and biomedical literature in order to improve the identification of relationships and understanding and management of medical information. The main tasks related to this area are Named Entity Recognition (NER), Inter species Normalization and Relation Extraction [10].

In biological text mining domain, one of the new fields is Cheminformatics [11][12], which is the synthesis of computer science and chemistry to collect knowledge about chemicals to provide useful information for drug development. Recent research has focused on improving chemical named entity recognition to assist researchers to cope with the explosion of chemical publications [13].

In the Chemical NER task several appropriate dictionary resources and NLP techniques have been used depending on the characteristic of the entity classes. Researchers developed systems which cope with each entity class of chemicals using manually set of rules [14][15], dictionary or grammar approach [16][17] and machine learning method [18][19].

Most of the recent studies on Chemical NER focused on developing systems based on supervised machine learning methods [18][20][21] [22]. In this thesis we applied the same classifier algorithm which is the SVM. We also stepped forward and employed feature selection methods to improve our system performance.

1.2 Thesis Contribution

In this study, a classification system which uses wrapper based feature subset selection algorithms is proposed. In particular Forward Search and Backward Search algorithms are considered and their performance is compared to classification systems which combine all features, Simplified Forward Search and the best single feature.

Three different standard chemical corpora which contain various entity classes of chemical names have been used as training and test sets. These three corpora include Fraunhofer Institute for Algorithms and Scientific Computing (SCAI) [23], The International Union of Pure and Applied Chemistry (IUPAC) [24] and Chemical Entities of Biological Interest (ChEBI) [22].

Several features have been extracted from the data sets. Features used include the set of tokens, morphological features, lexical features, orthographic features and spaces. Since these features are extracted from training data, they are considered as “internal resource features“. Moreover, a “dictionary feature” has been used by making use of an external dictionary.

The SVM classifier is trained on different data sets each time using one of the mentioned search algorithms to exploit different subsets of features. The effect of these feature sets is investigated. Best feature subset with high classification performance is selected as the optimal feature set.

1.3 Thesis Outline

The remaining organization of this dissertation will be as follows: In Chapter 2 an overview of text mining in the biomedical domain and chemical domain is given also a literature review on biomedical domain is presented. Chapter 3 presents the architecture of the proposed CHEM-NER system, chemical data, feature extraction and algorithms for feature subset selection used in this study. In Chapter 4 the results of the proposed CHEM-NER system are given and discussed. Finally Chapter 5 provides conclusion on the results and future works related to this field.

Chapter 2

LITERATURE REVIEW

2.1 An Overview of Text Mining in Biomedical Domain

In the recent decades, there has been a tremendous growth in the amount of biomedical data such as biomedical literature and biological databases especially in the field of genomics and proteomics. For instance, PubMed which is a large publicly available scientific biomedical database [25] and online repository contains more than 23 million citations. Therefore, it is essential to employ literature mining tools in order to extract interesting and relevant information for particular biomedical and biological tasks.

Since the literature includes biology, chemistry and medicine, during comprehensive mining all types of information can be extracted. These efforts contain tasks such as predicting possible pathways, discovering relationships between genes and disease, establishing association between genes and biological properties and so forth. A single method which can reveal all kinds of information is usually not possible. Often there is a need to develop an expert system for each individual task.

Abundance of information in the form of unstructured text need automated handling strategy. Text mining is the process of automatically analyzing unstructured text for discovering information and knowledge. In large biomedical documents and databases, text mining includes the following disciplines: Information Retrieval (IR), which includes finding and collecting relevant documents that satisfy the user's specific information need within a large database of documents [26], Information Extraction (IE), a discipline of NLP, is relevant to discover precise information and facts in unstructured text [27]. For instance, identification all entities in the biomedical text that refers to genes (entity recognition) is an IE task.

Another principle is Machine Learning (ML) [28][29], a subfield of Artificial Intelligence centered on building systems that are able to learn from previous experiences in order to find patterns and rules for processing automatically classification, clustering and prediction tasks. Finally, Knowledge Data Discovery (KDD) [30], the process of generation knowledge from structured and unstructured resources by using computational tools to facilitate the process of interpreting and inferencing (Figure 2.1).

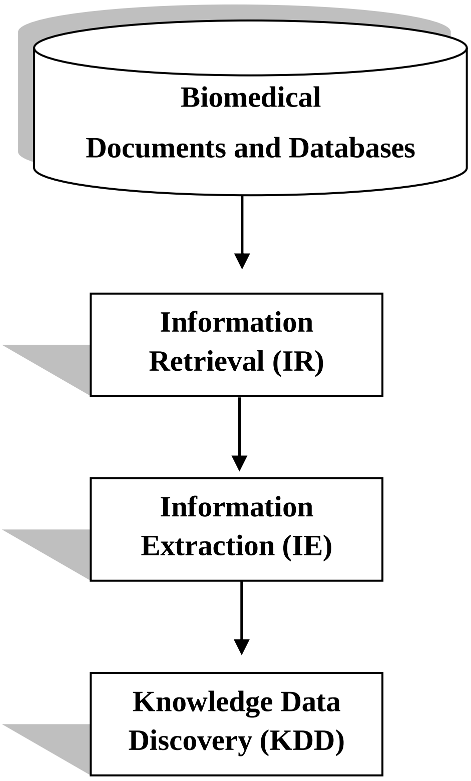

Figure 2.1: An Overview of Text Mining Applied in Biomedical Domain

In the last decades, many scientific competitions and shared tasks related to text mining in the biomedical domain have been organized. Linking Literature, Information and Knowledge for Biology (BioLINK) of 2001-2009 concentrated on the biomedical tools and application in the field of text mining. Knowledge Discovery and Data Mining (KDD) Challenge Cup task in 2002 [31] aimed at determining the priority of articles based on existence of experimental evidence for a gene. In Bio-entity Recognition Challenge of JNLPBA in 2004 [32], a gold standard Genia corpus was provided, which made possible comparison of various NER systems of all participants. Critical Assessment of Information Extraction systems in Biology (BioCreative) I, II II.5 and III in (2004, 2006, 2009 and 2010 respectively) [33] focused on identification of gene/protein as NER task, protein-protein interactions (PPI) as a relationship extraction task and gene mentioned normalization as NE normalization task. Recently, BioCreative IV (2013) has organized a competition challenge on Chemical and drug NER. Text Retrieval Conference (TREC) Genomics track in 2007 [34] focused on information retrieval tasks in this domain. The BioNLP 2009 Shared Task [35] focused on recognizing bio molecular events. The shared task aimed on preparing strong task definitions and gold standard data sets and developing and evaluating biomedical IE systems. BioNLP 2011 concerned on generalizing and extending previous tasks in three principle aspects: text, domain and aimed event types. Recently, BioNLP 2013 follows previous tasks while concentrating on new topics related to Cancer Genetics (CG), Pathway Curation (PC), Gene Regulation Network (GRN), Gene Regulation Ontology (GRO) and Bacteria Biotopes (BB). Also several famous data sets such as Genia, BB have been employed in order to provide realistic evaluation.

2.2 Classification of Text in the Biomedical Domain

Methods for organizing textual documents include classification methods which categorize documents into previously defined categories or clustering methods which group similar documents within the same group.

There are two basic methods for text classification. The first method is knowledge engineering method where the classification is based on a set of manually defined rules [36]. The disadvantage of this method is recognized as knowledge acquisition bottleneck since modification and developing the system needs cumbersome efforts. The second method is machine learning which is based on constructing systems that are able to learn from data. It means that ML-system will be built and trained based on initial data afterward later it can be used to classify new unseen data instances. There are several machine learning algorithms which are categorized based on the type of their input and desired output. The two main machine learning algorithms are supervised learning and unsupervised learning algorithms. Supervised learning will be done through a training set of correctly labeled examples that are tagged with predefined class labels to generate a classifier model for the prediction of new unseen inputs subsequently. Unsupervised learning is done through unlabeled examples to discover a pattern in the data to be able to predict new unseen data. In this study supervised machine learning algorithm is used. Classification model is generated using data whose performance is measured using a validation data set. To evaluate the performance of a classification model, a random portion of the training set is considered as test data set which is omitted from training data set. Then the classification task is done on the test data set. New class labels of test data can be compared with its relevant real class labels to measure the performance of classification task.

Different statistics measures are used to measure the performance of system such as Accuracy, Recall, Precession and F-score [37]. These measures are explained in detail in Appendix A.

2.2.1 Named Entity Recognition (NER)

The aim of designing IE systems is to provide automated text mining systems for extraction of events and the relation between them and retrieving necessary information in the documents automatically. Generally, named entity recognition (NER) systems rely on machine learning methods and algorithms to extract requested information from huge data sets such as biomedical data sets.

In information extraction systems (IE), the first step in classification of unstructured texts is the identification and classification of a known NE. A word or sequence of words in a text which reveal a specific object or a group of objects is defined as a named entity (NE). The aim of NER systems in the newswire domain is to locate and classify mentions of NEs such as persons, organizations and locations in the text as defined by the 6th and 7th Message Understanding Conferences (MUC) [38][39].

2.2.2 Biomedical NER (Bio-NER)

Identification of biomedical entities, such as drugs, proteins, genes, chemicals and diseases is the main aim of NER systems in the biomedical domain. This process is known as biomedical NER. Using the extracted NEs can reveal relationships of entities in biomedical data sets, such as protein-protein interactions (PPI) in biomedical documents [40], discover cancer-associated genes [41], extract physical protein interactions [42], predict gene-disease relationships[43] and drug- drug interaction which are the main research topics in recent studies in this field [44].

In this domain various approaches such as dictionary-based approaches [45][46], rule-based approach [47][48], and machine learning approaches [49][50][51] have been employed for accurate information extraction.

The focus of earlier works on biomedical NER mainly is on dictionary-based and rule-based approaches [52]. The main aim of dictionary based approach is to provide an encyclopedic dictionary that can be used as a reference for searching entities. On the other hand, the goal of rule-based approach is to produce an optimal set of rules covering all NEs by using the training data. Recently, using machine learning based systems has become popular in this area [50][53]. Machine learning models are more robust in facing noisy input data and more reliable when there is a need to develop the system.

Different classification algorithms have been used for data and text mining such as: Support Vector machine model (SVM) [49][53], Conditional Random Fields Model (CRF) [50], Hidden Markov Model (HMM) [54] and Maximum Entropy Model (ME) [55].

2.3 Classification in the Chemical Domain

2.3.1 Chemical NER (CHEM-NER)

Chemical NER refers to identification of entities that corresponds to a chemical target category [18]. Extracting information from chemical properties can provide useful knowledge to categorize drugs and chemical compounds. Using this information is also highly important in biomedical classification applications.

Finding the relation between drugs and disease and classification of disorders according to the effects of chemical compounds are addressable usage of chemical entity recognition. Furthermore, retrieving relevant articles, identifying relationships between chemicals and other entities or determining the chemical structures are other tasks which make use of chemical entity recognition.

Recently, BioCreative IV (2013) has organized a competition challenge on Chemical and drug NER. It includes five tracks such as Interoperability (BioC), Chemical and Drug Named Entity Recognition (CHEMDNER), Comparative Toxic genomics Database (CTD), Gene Ontology (GO) and Interactive Curation (IAT).

2.3.2 Chemical Entities Categories

Chemical names can be categorized into two main groups. These categories are systematic and non-systematic nomenclature.

2.3.2.1 Systematic Nomenclatures

In the case of systematic nomenclatures, chemical entities are based on precise rules which show how these names are formed. These rules are known as grammars, which describe the compound in terms of its structure. These chemical name grammars lead to unambiguous determination of the chemical structure from their systematic names. The International Union of Pure and Applied Chemistry (IUPAC) [24] has been in charge of maintaining the rules of chemical nomenclature since 1892. Based on word morphology, systematic nomenclatures have different forms of characteristic features. This property is extremely useful for the CHEM-NER task. Systematic nomenclatures are composed of chemical segments or terminal symbols which are distinguishable from normal English words. For instance, “benzo” or “mehtyl”, a token which included such elements has a high chance to be a chemical entity.

2.3.2.2 Non-Systematic Nomenclatures

In the case of non-systematic nomenclatures there are no known rules. Instead of rules, common names or abbreviations are frequently used. In this case, recognizing the entities or finding the appropriate relation between them is difficult. For example, the systematic name ‘Aspirin’ in the IUPAC is named as “2-acetyloxybenzoic acid”. Trivial names are catalogued and linked to their structure in resources such as PubChem. Recognition of such entities is normally performed by matching them against a dictionary of names.

Although, there are two main categories, in some cases, a mixture of systematic and non-systematic is used to construct the names. For example “2-hydroxy-toluene” and “2-methyl-phenol” are semi-systematic variants for “1-hydroxy-2-methyl-benzene”. Even if a semi-systematic name shows some regularity similar to systematic names category, it is difficult to assign them to the corresponding structures.

2.3.3 Available Chemical Corpora

Although chemical information is rapidly growing in all sorts of textual data, this domain still suffers from the lack of publicly available chemical corpora which should be manually annotated. Some researchers have generated several annotated corpora which are derived from the MEDLINE abstracts. SCAI corpus which consists of 100 MEDLINE abstracts [23], IUPAC training corpus which contains 463 MEDLINE abstracts [24] and CHEBI corpus a molecular small entity dictionary [22], can be considered as a benchmark to compare other systems that use these freely available corpora. The data typically has been segmented into smaller portions of text which is named tokenized data. Each token has a label. A typical labeling paradigm is the IOB format which make easy to discover the boundaries of chemical entities [56]. (B) indicates that the current token is the beginning of a chemical entity, (I) mentions that the current token is inside a chemical entity and (O) represent the token is not a part of chemical entity anymore. Detailed information of these corpora is presented in section 3.2.

2.3.4 Methods Used in CHEM-NER

Identification of trade names, such as marketed drugs using dictionary matching approach has been a common task in CHEM-NER applications [23]. Less efforts has been spent in the identification of systematic nomenclatures which need more sophisticated approaches. Recent NER research has focused on recognition these systematic names.

CHEM-NER methods are categorized into three groups such as dictionary based, morphology based and context based.

2.3.4.1 Dictionary Based Method

In this approach each word/token in the text will be compared with the entries in a dictionary. This process is called word matching or lookup. Therefore, to get a good result from this method, there is a need for a comprehensive dictionary and an efficient matching algorithm. Dictionaries can be developed manually or automatically from public resources and databases. For example, the Unified Medical Language System (UMLS) [57] is an automatically produced dictionary from chemical databases. Jochem is a dictionary [58], which was automatically generated by different chemical resources such as UMLS, ChEBI, MeSH terms, PubChem and DrugBank [23]. Because of its huge size, it needs some heuristic and statistical methods to maintain and develop.

With high variability in chemical names, instead of exact matching, some other strategy such as using regular expressions or string comparison metrics like Levenshtein Distance [59] can be applied for the matching process.

2.3.4.2 Morphology Based Method

As it was mentioned earlier, nomenclatures which contain chemical terminal symbols (e.g. ‘benzo’ and ‘methyl’) have high probability of being chemical entity. Thus, by tokenizing or segmenting entities, and using a dictionary of chemical name segments to find terminal segments in chemical entities or using some statistical models such as Naïve Bayes model, the chance of detecting chemicals will be increased.

2.3.4.3 Context Aware Systems

Context of mentions is one of the techniques of NER which is based on linguistic analyses of the text such as syntactic analysis. This approach can be used by machine learning models (using statistical methods or NLP techniques) or manual rules (based on language structure). Systems using machine learning model, implement NER as a classification task and try to predict whether the token corresponds to a chemical or not. The main difficulty of this approach is the need for a reasonably large annotated corpus to construct high accurate classification models.

Chapter 3

SYSTEM OVERVIEW

3.1 The Architecture of the Proposed CHEM-NER System

Figure 3.1 shows the architecture used to recognize the chemical entities and evaluate the performance of various classification systems used in this study.

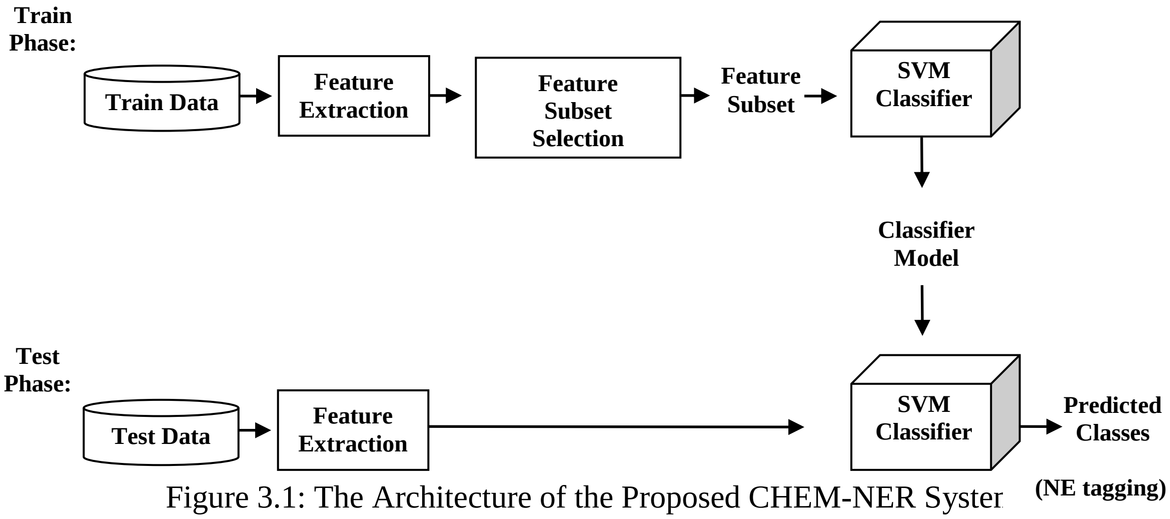

Figure 3.1: The Architecture of the Proposed CHEM-NER Syster

Each SVM classifier is a multi-class SVM which is trained using different subsets of features. The individual features extracted and the algorithms used for selecting feature subsets are explained in detail in sections 3.3 and 3.4, respectivley. Cross-Validation is used to measure the performance of single features as well as combination of features while using a feature subset selection algorithm.

Training is done to build a classification model which is used to predict the test data tags. Since this study aimed to use machine learning approach for NER, we have selected SVM as a machine learning algorithm. The algorithm tries to separate input space into linearly separable feature space by utilizing appropriate kernel function. We used Yamcha which is a SVM-based chunker, to turn the training data set format into acceptable SVM format.

3.1.1 Support Vector Machine

Support vector machine, which is a supervised machine learning algorithm, is intensively appropriate for high dimensional data for the text classification task [28][49][53][60]. The training and test data, which is used in the classification task, consists of data samples. Each sample in the training data set comprises several features and is labeled with a class name. SVM training algorithm constructs a model from the training data set and assigns a target class name to each instance in the test data.

SVM splits the space of possible examples into negative and positive sections by constructing a hyperplane. The subset of training data points, which lie on the boundary of the hyperplane, are called support vectors. A large number of hyperplanes can be constructed to classify the data. But an optimal split will be achieved by the hyperplane that has the largest distance (margin) to the nearest positive and negative examples (Figure 3.2). The largest margin leads to the lowest generalization error of the classifier.

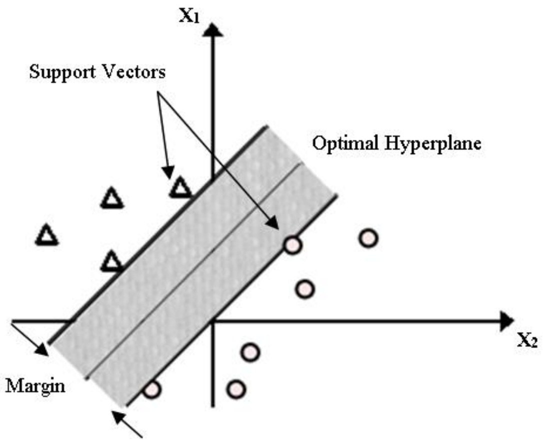

Figure 3.2: Linearly Separable Binary Classification Problem

Moreover, SVM uses a kernel function that transforms the non-linearly separable input space into a linearly separable higher dimensional feature space (Figure 3.3). Kernel function represents the similarity between data points measured in the higher dimensional space in order to define the class of possible patterns. There are several kernel functions such as linear, polynomial, radial basis and sigmoid function.

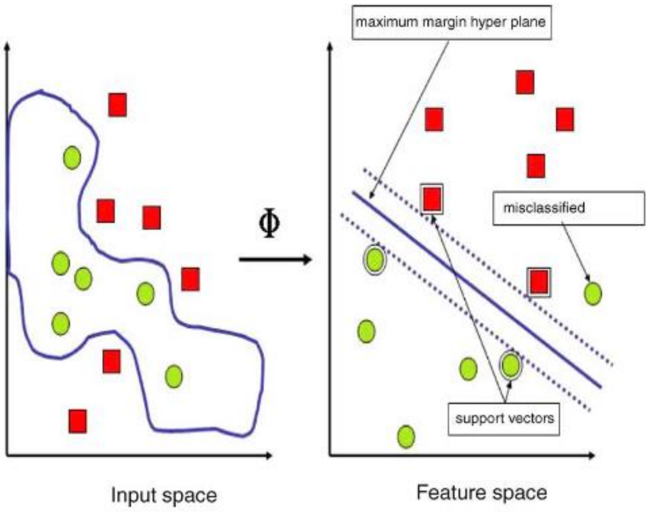

Figure 3.3: Transformation the Non-linearly Separable Input Space into a Linearly Separable Higher Dimensional Feature Space by Using of Kernel Function Φ.

The basic SVM is used for two-class data sets, which makes a linear classification. But it can be enhanced to M-class data set. There two approaches to solve multi-class problems such as one-versus-rest and pair-wise combination [61].

In the one-versus-rest approach, for M classes included in the training data, there are M binary SVM classifiers. The training set of *i*th SVM composed of all samples of *i*th class which are labeled as positive samples and with all samples from other classes which are labeled as negative samples. Each binary SVM classifier will predict the label of the new input. The SVM classifier with the highest output determines the class of input data.

In the pair-wise method, a multi-class model which is based on majority voting on the combined binary classifiers will be used. In total M(M-1)/2 individual binary SVM classifiers are required [62][63] one for each pair. Since, each classifier has one vote; the class with the highest number of votes will be selected.

3.1.1.1 Machine Learning Using Yamcha

In this study, Yet Another Multipurpose Chunk Annotator (Yamcha) which is a SVM-based chunker is used for training the classifiers [64]. Yamcha as an open source text chunker is applicable in several NLP tasks such as POS tagging, NER and test chunking [65]. It uses SVM as its learning algorithm. Yamcha takes the input data in the appropriate format and transforms it to feature vectors which are usable for open source TinySVM software [66]. Figure 3.4 shows an example of the input data file. Each line corresponds to a word or a token. A collection of lines which are separated by a blank line forms a sentence. A token consists of several columns. The number of columns should be fixed for all tokens. Next to the each token there are several features which are separated by a white space. The last column of each line indicates the true tag which should be trained by SVM. Yamcha utilizes the context window so that it may use the preceding and following tokens with their respective features as static window and the predicted classes of preceding tokens as dynamic window. The content of mentioned windows will be used as features set to predict the tag of the current token. For example in Figure 3.4 for the current token in position 0 which is highlighted as well, the size of static window is [-2..2] and the dynamic window size is [-2..-1].

	Token	Morphological Feature (2 gram suffix)	Lexical Feature (POS)	Orthographic Feature (uppercase)	Space (Left &Right space)	**Tag**
Position:-4	trimethylsilyl	yl	NN	no	yes	B-IUPAC
Position:-3	iodide	de	NN	no	yes	I-IUPAC
Position:-2	in	In	IN	no	yes	O
Position:-1	acetonitrile	le	NN	no	yes	B-TRIVIAL
Position:0	(	(	(	no	no	O
Position:+1	Me3SiI	il	LS	yes	no	B-SUM
Position:+2	/	/	SYM	no	no	O
Position:+3	CH3CN	CN	NN	yes	no	B-SUM
Position:+4	)	)	)	no	no	O

Figure 3.4: An Example of the Input data to Yamcha.

Yamcha computes the number of all features used in the data set, and gives a unique positive integer corresponds to each feature. An example of feature vector representation by Yamcha is shown in Figure 3.5.

```
I-IUPAC        99:1 5166:1 5168:1 5178:1 5211:1 5228:1 5978:1 5981:1
I-MODIFIER     4438:1 5166:1 5168:1 5191:1 5211:1 5917:1 5978:1 5980:1
I-MODIFIER     7:1 5166:1 5168:1 5171:1 5211:1 5219:1 5978:1 5980:1
I-MODIFIER     8:1 5166:1 5168:1 5172:1 5211:1 5220:1 5978:1 5980:1
I-PARTIUPAC    10:1 5166:1 5168:1 5173:1 5211:1 5222:1 5978:1 5980:1
```

Figure 3.5: An Example of Feature Vector Representation by Yamcha

Each line corresponds to a vector. The correct class of each sample is given in the leftmost column. On the left side of each colon, the positive integer denotes the feature number. A “1” indicates that the vector contains the feature presented by its corresponding number. In addition to the context window which is tunable, Yamcha has other redefinable parameters such as parsing-direction, degree of polynomial kernel and algorithms for solving multi-class problems. In this study several experiments have been done to obtain the best tune for mentioned parameters. Therefore, the default value for context window which is [-2 +2] and second degree of polynomial kernel is used. There are two approaches for parsing-direction such as forward parsing direction (left to right) and backward parsing direction (right to left). The result of experiments showed that backward direction is more successful because it is more effective in boundary detection. Most of the chemical entities are long and descriptive which are tokenized into several tokens so that in the IOB format the first token is labeled as ‘B’ and the others are labeled as ‘I’. Therefore, backward parsing by improving the boundary detection helps the classifier in recognizing ‘I’ tokens and ‘B’ tokens. The method for addressing to the multi-class problem is considered as pair wise method.

3.2 Data

Annotated corpora are essential for training and performance assessment of NER systems. In this study three different corpora named as SCAI, IUPAC and CHEBI which contain MEDLINE abstracts are used for training and testing of the CHEM-NER system. These data sets are tokenized in the IOB format. The areas which these data sets focus on are given in Table 3.1.

Table 3.1: Annotated Corpora used for CHEM-NER Task

Corpus	Focus
SCAI	General chemicals
IUPAC	IUPAC Entities
CHEBI	Molecular entities

Chemical names can be classified into different groups according their properties. In our study, seven different classes available in the SCAI corpus are considered. The name and description of these classes are given in Table 3.2

Table 3.2: Chemical Classes Defined for CHEM-NER Task

CLASS	Description	Example
IUPAC	Systematic and semi systematic names, IUPAC and IUPAC like names	2-Acetoxybenzoic acid
PARTIUPAC	Partial IUPAC names	17beta-
MODIFIER	Part of the drugs and chemicals group	Derivative, group, moiety
FAMILY	Chemical family names	Iodopyridazines, terpenoids
SUM	Molecular formula	CH(OH)CHI2
TRIVIAL	Brand (trade), generic names of compounds	Aspirin, Panadol
ABBREVIATION	Abbreviations and acronyms of chemicals and drugs	GABA, DHT

3.2.1 SCAI

SCAI corpus has been developed by the Fraunhofer Institute for Algorithms and Scientific Computing and is freely available as an annotated corpus [23]. It contains seven different classes of chemical entities and is considered as a gold-standard corpus. Since it is widely used in cheminformatics classification studies [20][23] and contains a large number of classes of entities, this data set is considered as the primary data set in this thesis.

SCAI corpus contains 100 MEDLINE abstracts. Table 3.3 presents the statistical information including the number of chemical compounds for each class and also the total number of chemical compounds, sentences and tokens for SCAI data set. Remaining tokens are named as ‘OUT’ tokens.

Table 3.3: Statistics of SCAI Corpus

CLASS	No. of entities
IUPAC	391
PARTIUPAC	92
MODIFIER	104
FAMILY	99
SUM	49
TRIVIAL	414
ABBREVIATION	161
No. of chemical entities	1310
No. of sentences	914
No. of tokens	30,734

3.2.2 IUPAC

IUPAC training corpus contains 463 MEDLINE abstracts out of 10,000 sampled MEDLINE abstracts [21][23]. Table 3.4 shows relevant statistical information including the number of chemical compounds for each class and also the total number of chemical compounds, sentences and tokens for this corpus.

Table 3.4: Statistics of IUPAC training Corpus

CLASS	No. of entities
IUPAC **PARTIUPAC** **MODIFIER**	3,712 322 1,040
No. of chemical entities	5,074
No. of sentences	3,744
No. of tokens	161,591

As can be seen from Table 3.4 IUPAC training corpus contains only the three main classes of entities present in the SCAI the remaining tokens are labeled as 'OUT' tokens.

IUPAC test corpus, which is originally provided separate from the IUPA training corpus, contains 1,000 MEDLINE records [21]. Table 3.5 shows relevant statistical information including the number of chemical compounds for each class and also the total number of chemical compounds, sentences and tokens for this corpus.

Table 3.5: Statistics of IUPAC test Corpus

CLASS	No. of entities
IUPAC **PARTIUPAC** **MODIFIER**	151 0 14
No. of chemical entities	165
No. of sentences	4,878
No. of tokens	124,122

As can be seen from Table 3.5 IUPAC test corpus contains only the two main classes of entities present in the IUPAC training corpus the remaining tokens are labeled as ‘OUT’ tokens.

3.2.3 CHEBI

Chemical Entities of Biological Interest (ChEBI) is a molecular small entity dictionary. It is not a comprehensive molecular dictionary, but is curated manually which provides an extremely high quality. The entities are organized by their chemical properties. ChEBI includes chemical classes such as biological and pharmacological compounds, trivial names, IUPAC, and sum formula. It is generally composed of chemical compounds SMILES and InChI [22]. CHEBI is freely available corpus which was published in 2009 with the purpose of being as a gold standard to be used in text mining researches.

CHEBI is published in the form of XML files. So to prepare the data in an appropriate format, tokenization as an initial step in NER problems, has been done. Tokens in CHEBI corpus are labeled as chemical or nonchemical names in IOB format. Table 3.6 shows the statistics of this corpus. In other words in CHEBI tokens are only marked as chemical entities or non-chemical entities. So, any classification problem using CHEBI is in essence a 2-class classification task.

Table 3.6: Statistics of CHEBI Corpus

No. of chemical entities	18,061
No. of sentences	4,985
No. of tokens	336,393

The summary statistics of all corpora used in this study is shown in Table 3.7.

Table 3.7: Summary Statistics of all corpora

CLASS	SCAI	IUPAC training	IUPAC test	CHEBI
IUPAC	391	3,712	151	Undefined
PARTIUPAC	92	322	0	Undefined
MODIFIER	104	1,040	14	Undefined
FAMILY	99	0	0	Undefined
SUM	49	0	0	Undefined
TRIVIAL	414	0	0	Undefined
ABBRIVIATION	161	0	0	Undefined
No. of Chemical Entities	1310	5,074	165	18,061
No. of Sentences	914	3,744	4,878	4,985
No. of Tokens	30,734	161,591	124,122	336,393

3.3 Feature Extraction

Feature extraction is the process of converting the high dimensional input data into a set of features in order to reduce the size of feature space and remove the redundant and irrelevant data [67]. Reducing dimensionality improves the speed of process of learning algorithms. This is an important concept in many topics such as pattern recognition, data mining, image processing and machine learning. By carefully choosing the features extracted there is a high chance to increase the accuracy and performance of system in desired task.

Regarding chemical structure properties, chemicals generally include morphological and orthographic rules, so extracting appropriate features based on their formation can increase the performance of NER task. In this study, Features similar to the work introduced by other researchers [21][65][68] have been used. These features are presented next.

3.3.1 Tokens

Token corresponds to each word in the sentences. Considering that these corpora are composed of sentences correlated to abstracts, and sentences contain some words which are known as tokens or single unit of text, so for each token the preceding and the following tokens in the training data can be considered as a feature so that it has a positive effect on the NER performance [63].

3.3.2 Preceding Class(es)

For each token the corresponding dynamic content of the context window, which are generated dynamically during the tagging process, have been used as features to predict the preceding tokens.

3.3.3 Morphological Features

Morphological features or affixes are the first *n* beginning/ending letters of the token. In this study, bi-, tri- and tetra-grams have been considered as affixes of the token. These features have significant contribution in recognizing systematic nomenclatures which are based on chemical segmentation [13].

3.3.4 Lexical Features

Functional lexical features are grammatical form of the words. In this study part-of-speech (POS) and noun phrase (NP) have been used as lexical features which are described below. Genia tagger, a tagger specifically trained using MEDLINE abstracts [69], has been used to extract these features.

3.3.4.1 POS Tag

Grammatically, a Part Of Speech (POS) is a linguistic class of words which refers to the syntactic rule of the lexical element. Part of speech has the eight standard types such as nouns, adjective, adverbs, verbs, conjunctions, determiners, prepositions, and pronouns. Also interjections and punctuation marks are included as POS tags. The first four of eight common POS are called content words and the former four POS are called as function words [70].

The positive effect of using POS features has been reported by other research works [68][71] specially in word boundary detection. Since most of the words in dictionary are in the content words category [70] significance the effect of this feature in the proposed system will be considered.

3.3.4.2 Noun Phrase Tag

A phrase whose head word is noun is counted as a noun phrase (NP). Noun phrases occur frequently therefore recognizing them may lead to improvement in the boundary detection in mentioned identification tasks.

3.3.5 Orthographic Features

Orthographical features are based on word formation patterns and rules of spelling. They may include hyphenation, capitalization, punctuation and etc. The presence of an orthographic feature is marked as ‘1’ and its absence as ‘0’ in the feature vector. Table 3.8 shows the orthographic features used in this study with their relevant regular expression and examples.

Table 3.8 : Orthographic Features used in the SVM

Ortho. Features	**Reg. Ex.**	**Example**	**Ortho. Features**	**Reg. Ex.**	**Example**
All Caps	/^[A-Z]+$/	NPS	Pattern	/thy|xy|CH|NH|acid/	hydroxy
Is Real	[-0-9]+[.,]+[0-9.,]+	9	Any Slash	/[\Q \/ \ \E]/	MeSiI/CH3CN
Is Dash	^[- – — −]$	-	Uppercase	/[A-Z]+/	BuS
Is quote	^[„ “ ” ” ‘ ’]+$	“	2 Upper	/.*[A-Z]+.*[A-Z]+/	AacCmES
Is Slash	/^[\Q \/ \ \E]$/	/	Alpha & Other	/(.+[a-zA-Z]+.*)|(.*[a-zA-Z]+.+)/	derivatives
Initial Upper	/^[A-Z]+/	Br2	Hyphen	/-+/	C-14
Any Punctuation	/[\Q (){}[]=+% !|_<>*@#& ?\E]/	[(3)H] kainic acid	Upper or Digit	/([A-Z]+)|([0-9]+)/	4-AHCP
2Upper & Digit	[A-Z]+[0-9]*[A-Z]+	CH3CN	Any Digit	/([0-9]+)/	3a

3.3.6 Spaces

It has been reported in [21] that detecting spaces preceding and following the tokens has a positive effect in boundary detection during CHEM-NER. Here left space, right space and both features left and right spaces have been used. Again the presence of a space is marked as ‘1’ and absence as ‘0’ in the feature vector. Table 3.9 shows the list of features and with their respective type used in this study. Each feature is given a unique feature number to make reference to specific features easier during the classification to follow.

Table 3.9: List of Features used

Feature Number	Feature Name	Type
f_1	2 gram Prefix	Morphological Features
f_2	2 gram Suffix	
f_3	3 gram Prefix	
f_4	3 gram Suffix	
f_5	4 gram Prefix	
f_6	4 gram Suffix	
f_7	All Caps	Orthographical Features
f_8	Is Real	
f_9	Is Dash	
f_{10}	Is quote	
f_{11}	Is Slash	
f_{12}	Initial Upper	
f_{13}	Any Punctuation	
f_{14}	2Upper & Digit	
f_{15}	Pattern: thy|xy|CH|NH|acid	
f_{16}	Any Slash	
f_{17}	Uppercase	
f_{18}	2 Upper	
f_{19}	Alpha & Other	
f_{20}	Hyphen	
f_{21}	Upper Or Digit	
f_{22}	Any Digit	
f_{23}	Left Space	Spaces
f_{24}	Right Space	
f_{25}	Left & Right	
f_{26}	POS	Lexical Features
f_{27}	NP	

3.4 Feature Combination

Although it may seem reasonable to use all available features, in practice feature combination or feature subset selection is used in order to ignore redundant features, reduce the dimensionality and use the combination of most useful features during a classification task [1]. This is a systematic approach which usually improves the performance of the classification system. The simplest paradigm is to test all possible subsets of features to find the best subset which gives the best result, but it is an exhaustive search of space which for *n* features 2^n subsets should be tried so it is a time consuming method.

A simple feature selection process has four steps: 1) a scale that evaluates the performance of a subset 2) a search strategy for producing subsets 3) a criterion for stopping searching 4) a subset validation function [12].

Three types of standard feature selection methods are: embedded, filter, and wrapper approach. Embedded approach performs feature selection during the operation of model construction and decides whether a feature is accepted or rejected. Decision tree classifiers utilize this method [72].

Filter approach is completely free of the classification task and will be done before the task starts. A proxy measure like pair wise correlation will be used to select the set of features. In fact the filter approach tries to evaluate the merit of features from the data. Since it selects the features in a preprocessing step which is done before the classification task starts, the effect of selected features on the performance of the inducting algorithm will be ignored. This is a low computational method and is fast to perform.

Wrapper approach is somewhat similar to the exhaustive method but with lower complexity. It uses a predictive model to evaluate the fitness of a feature set. For each subset, it trains a model. Although, it is computationally expensive and maybe object to overfitting usually satisfactory results are obtained.

In the current study the wrapper approach is addressed. The evaluation measure used is the Micro-averaged F-score (see Appendix A for details of Micro-average F-score). Three kinds of search algorithms such as Simplified Forward Search, Forward Search and Backward Search have been used. Results obtained using feature sets of each search algorithm will be compared with each other as well as results of full set of features and the single best feature.

3.4.1 Wrapper Based Search Algorithms

As it was mentioned, three types of greedy search strategies which attempt to establish an optimal feature set by adding or removing features have been applied.

3.4.1.1 Simplified Forward Search (SFS)

This heuristic approach starts with getting the results of all single features and sorting them in descending order, the first best single feature will be combined with the second best single feature and the subset will be evaluated. If the result improves, this new feature set will be kept and the next top single feature will be added and evaluated, otherwise the process will be stopped and the SFS feature set will be obtained.

3.4.1.2 Forward Selection (FS)

This greedy search strategy attempts to establish an optimal feature set by adding randomly one more single feature at each iteration. It starts with the Single Best feature and will be expanded until combining new single features no longer improves the results [73].

3.4.1.3 Backward Selection (BS)

BS is a greedy search algorithm. However, unlike FS it starts with the set of all features. At each iteration randomly one single feature will be omitted, and the performance of the new feature set will be evaluated. If the result improves then elimination is acceptable otherwise omitted single feature will be kept in the feature set. BS will be terminated if there is no longer improvement in the elimination of features [73].

3.4.2 Single Best (SB)

Computing the results of all features separately then, choosing the one that has the highest result, is a simple heuristic approach that is named as Single Best. There is no feature combination in this method and can be considered as a baseline reference to make comparison with the results of other approaches.

3.4.3 Combination of All Features

Another approach in feature subset selection is using the combination of all features which can be considered as a base system in order to compare with other approaches. Some researchers reported that the combination of all features often has the highest performance and there is no need to feature selection for SVM in biomedical domain [28][74]. In this study we will investigate how feature subset selection can be effective on chemical names entities.

3.4.4 Cross-Validation of the Models

Cross-Validation is a common method to evaluate the performance of a classification task [73]. In general 10-fold Cross-Validation is selected which statistically has been proved is good enough to evaluate the classification results.

In this study, Cross-Validation has been done on both SCAI and IUPAC training data sets in order to choose the feature subsets. We have considered 10-fold Cross-

Validation for SCAI and 3-fold Cross-Validation for IUPAC training corpus since it was a big corpus and performing 10-fold Cross-Validation takes lots of time. In 10-fold Cross-Validation data set is divided into 10 roughly equal folds. The classifier will be trained on 9 folds and will be tested on the remaining fold. Since there are 10-folds the generating of test and train data set takes 10 repetitions. Finally, the performance of classifiers using these data sets is calculated as average of all repetitions in terms of Micro-average.

Chapter 4

RESULTS and DISCUSSION

In this study, wrapper based feature selection is used to test if feature subset selection methods can improve SVM classifier performance for the CHEM-NER task. Three different chemical corpora which contain various classes of chemical names as described in section 3.2 have been used as training and test sets.

4.1 Classification Performance using Single Features

In order to obtain a high performance in CHEM-NER elucidating the patterns hidden in chemical data is essential and leads to have a good understanding of their functional structure. So analyzing the effect of single features will give a general understanding on the structure of different chemical classes used in a corpus. Furthermore, the performance of single features is required for the implementation of the SFS feature selection algorithm.

4.1.1 Classification Performance using Single Features in the SCAI Corpus

The SCAI corpus is used as the main corpus since it is the most comprehensive data set as it contains 7 different classes of chemical entities. Since the SCAI corpus does not contain a separate train and test set, 70% of the data is reserved as train data and the remaining 30% as test data.

Table 4.1 shows the classification performance using single features sorted according to the Micro-average F-score for all entities using default tuning parameters of the system. The recognition performance for each individual class is also given. All the experiments are done by 10-fold Cross-Validation using 70% of SCAI data as the training data set. In addition, last row of this table shows the average F-score for each chemical class obtained using individual F-score for each feature.

Table 4.1: Classification Performance Using Single Features (SCAI corpus)

			CLASS						
Feature No.	**Feature Name**	**Micro- Average F-score**	**ABBR.**	**MODIFIER**	**PARTIUPAC**	**TRIVIAL**	**IUPAC**	**SUM**	**FAMILY**
			F-score						
f_2	2 gram Suffix	**0.4377**	0.0432	**0.4333**	0.2424	**0.4511**	**0.6246**	0.0000	0.0460
f_4	3 gram Suffix	0.3460	0.0000	0.2963	0.2128	0.3133	0.5523	0.0000	0.0455
f_{26}	POS	0.2879	0.0429	0.4000	0.2574	0.1044	0.5009	0.0000	0.0440
f_{22}	Any Digit	0.2831	0.0290	0.3471	**0.2800**	0.0921	0.4899	0.0000	0.1075
f_{23}	Left Space	0.2799	0.0145	0.3894	0.2128	0.0726	0.5144	0.0000	0.0870
f_{21}	Upper Or Digit	0.2747	0.0000	0.3697	0.1616	0.1033	0.4793	0.0000	0.1064
f_{19}	Alpha & Other	0.2725	0.0141	0.3529	**0.2800**	0.0773	0.4795	0.0000	0.1042
f_{15}	Pattern	0.2561	0.0292	0.3577	0.1474	0.0965	0.4464	0.0000	0.1064
f_{10}	Is quote	0.2545	0.0147	0.3697	0.1064	0.0822	0.4649	0.0000	0.1064
f_{17}	Uppercase	0.2525	0.0541	0.3833	0.2041	0.0829	0.4359	0.0000	**0.1087**
f_{24}	Right Space	0.2485	0.0145	0.3590	0.2292	0.0557	0.4590	0.0000	0.0851
f_{12}	Initial Upper	0.2469	0.0544	0.3833	0.2222	0.0720	0.4249	0.0000	0.1087
f_8	Is Real	0.2465	0.0288	0.3361	0.1875	0.0924	0.4229	0.0000	0.1075
f_6	4 gram Suffix	0.2341	0.0000	0.2909	0.1522	0.0833	0.4492	0.0000	0.0460
f_{27}	NP	0.2332	0.0145	0.3529	0.1935	0.0388	0.4412	0.0000	0.0449
f_{18}	2 Upper	0.2328	0.0699	0.3697	0.1667	0.0822	0.3977	0.0000	0.1087
f_{14}	2Upper & Digit	0.2311	0.0292	0.3740	0.1087	0.0916	0.4070	0.0000	0.1087
f_9	Is Dash	0.2302	0.0147	0.3559	0.1474	0.0919	0.4046	0.0000	0.1075
f_{11}	Is Slash	0.2301	0.0286	0.3833	0.1075	0.0924	0.4047	0.0000	0.1075
f_{16}	Any Slash	0.2301	0.0286	0.3833	0.1075	0.0924	0.4047	0.0000	0.1075
f_{13}	Any Punctuation	0.2283	0.0146	0.3621	0.1505	0.0773	0.4093	0.0000	0.1064
f_{20}	Hyphen	0.2274	0.0147	0.3559	0.1099	0.0919	0.4023	0.0000	0.1075
f_7	All Caps	0.2270	**0.0833**	0.3729	0.1875	0.0829	0.3760	0.0000	0.1075
f_{25}	Left & Right	0.2217	0.0000	0.3717	0.1538	0.0670	0.4015	0.0000	0.0860
f_1	2 gram Prefix	0.2121	0.0000	0.3119	0.2105	0.0287	0.4109	0.0000	0.0460
f_5	4 gram Prefix	0.2085	0.0000	0.3243	0.1895	0.0342	0.4032	0.0000	0.0455
f_3	3 gram Prefix	0.1982	0.0000	0.3091	0.1739	0.0228	0.3898	0.0000	0.0455
Average F-score			0.0240	**0.3600**	0.1800	0.100	**0.4400**	0.000	0.0900

The results given in Table 4.1 show that the system trained using the 2-gram suffix feature (f_2) achieves the highest performance. It has the highest performance in recognizing classes such as MODIFIER, TRIVIAL and IUPAC. This result can be attributed to the fact that these classes make up for totally 69.3% of all chemical entities included in the SCAI corpus. Moreover, using all caps feature (f_7) has the highest result in recognizing the Abbreviation class since most of the entities that belong to this class are capitalized.

Although using the uppercase feature (f_{17}) contributes the most to the recognition of the Family class the F-score 0.1087 is still very low. A simple analysis shows that some entities in the Family class are similar to entities in the IUPAC class. For example 'Pyrimidine' is a common entity in both classes. Since IUPAC class is a major class, the classifier mostly recognizes these entities wrongly as either IUPAC class or OUT class which leads to the low F-score value. In such cases, using a dictionary may be effective in improving the recognition of such classes [23].

Also it can be seen that the systems trained using either the feature Alpha & Other (f_{19}) or Any digit (f_{22}) achieve the highest performance in recognizing entities in the PARTIUPAC class. Since most of the entities in this class contain a digit, using this feature seems relevant. It should be mentioned that in fact entities which belong to the PARTIUPAC class are partial IUPAC names which means that the PARTIUPAC class can be considered as a subset of the IUPAC class. Therefore, due to problems in boundary detection, many PARTIUPAC entities are often misclassified as IUPAC entities, or vice versa.

It is interesting to note that the recognition performance of entities in the SUM class is zero. In other words, the classifier completely fails in recognizing entities in this class. One reason may be the fact that from the totally 49 SUM entities included in the SCAI corpus only 50% of them exists in 70% of the SCAI data which is considered as the train data. Therefore, due to the under-representation to this class the model is not well trained for the SUM class and has a high generalization error. In section 4.3.3 it is shown that when 100% of the SCAI data is used as train data, the model is more successful in recognizing unseen SUM examples of the CHEBI corpus which is selected as test data.

Figure 4.1 shows the recognition performance of each feature used for each class in the SCAI corpus. According to the results of average F-score of each class (the last row of Table 4.1) and Figure 4.1 it can be seen that the entities that belong to classes MODIFIER and IUPAC can be recognized more successfully compared to others. Moreover, since the top five best performing features in Table 4.1 are members of affixes, spaces and lexical features so it can be expected that the combination of these features can be effective in the NER task and are likely to be included in the subset of features that will be extracted by wrapper based search algorithms.

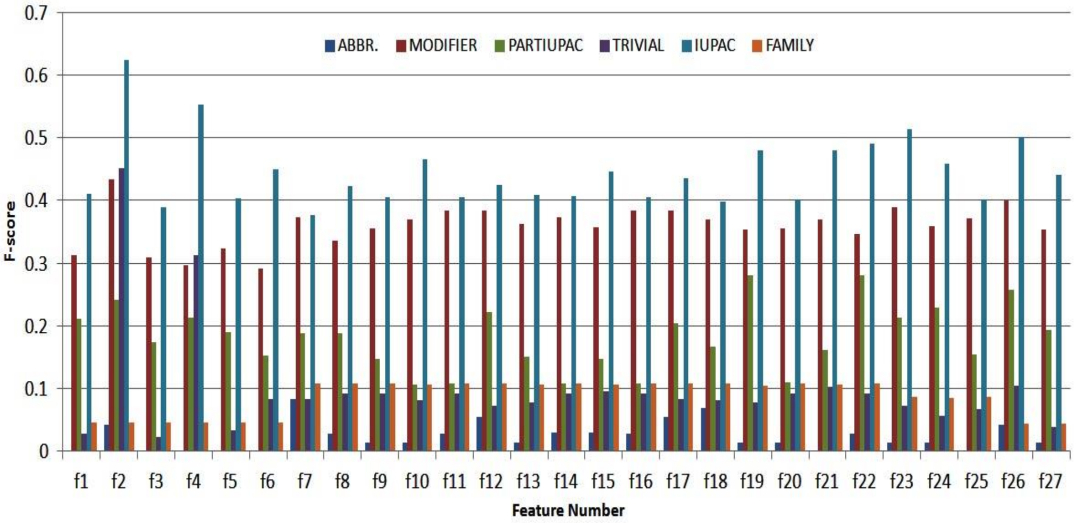

Figure 4.1: The Recognition Performance of each Class (SCAI Corpus)

4.1.2 Classification Performance using Single Features in the IUPAC training Corpus

Table 4.2 shows the results using single features on IUPAC training corpus, sorted according to the Micro-average F-score. All the experiments are performed using 3-fold Cross-Validation.

Table 4.2: Classification Performance Using Single Features (IUPAC training corpus)

			CLASS		
Feature No.	**Feature Name**	**Micro Average F-score**	**MODIFIER**	**PARTIUPAC**	**IUPAC**
			F-score		
f_2	2 gram Suffix	**0.7229**	**0.6433**	**0.4201**	**0.7621**
f_4	3 gram Suffix	0.7058	0.6269	0.4070	0.7446
f_{26}	POS	0.6829	0.6409	0.3906	0.7122
f_6	4 gram Suffix	0.6820	0.6100	0.4027	0.7181
f_{21}	Upper or Digit	0.6777	0.6299	0.3884	0.7092
f_3	3 gram Prefix	0.6735	0.6156	0.4088	0.7049
f_1	2 gram Prefix	0.6665	0.6061	0.4148	0.6976
f_5	4 gram Prefix	0.6664	0.6124	0.4079	0.6967
f_{27}	NP	0.6606	0.6240	0.3710	0.6888
f_{17}	Uppercase	0.6600	0.6111	0.3761	0.6910
f_{22}	Any Digit	0.6544	0.6150	0.3930	0.6807
f_{12}	Initial Upper	0.6529	0.6111	0.3683	0.6820
f_{23}	Left Space	0.6486	0.6260	0.3826	0.6717
f_{18}	2 Upper	0.6482	0.6222	0.3731	0.6726
f_7	All Caps	0.6428	0.6073	0.3812	0.6690
f_{19}	Alpha & Other	0.6419	0.6070	0.3532	0.6695
f_{10}	Is quote	0.6416	0.6095	0.3870	0.6661
f_{24}	Right Space	0.6359	0.6116	0.3911	0.6580
f_{13}	Any Punctuation	0.6349	0.6121	0.3656	0.6581
f_8	Is Real	0.6340	0.6018	0.3777	0.6588
f_{20}	Hyphen	0.6318	0.6117	0.3672	0.6543
f_{25}	Left & Right	0.6317	0.6009	0.3700	0.6566
f_{14}	2Upper & Digit	0.6315	0.6143	0.3731	0.6527
f_{11}	Is Slash	0.6293	0.6083	0.3644	0.6519
f_{16}	Any Slash	0.6293	0.6090	0.3644	0.6518
f_{15}	Pattern:	0.6289	0.6090	0.3731	0.6507
f_9	Is Dash	0.6288	0.6083	0.3680	0.6511
Average F-score			**0.6200**	0.3800	**0.6800**

3-fold Cross-Validation is preferred instead of the 10-fold case used for the evaluation of the SCAI corpus, since the IUPAC training corpus is a very large corpus. In addition, last row of this table shows the average F-score for each chemical class obtained using individual F-score for each feature. Similar to the previous case, it can be seen that the highest performance is achieved using the 2-gram suffix feature (f_2). Furthermore, the top three best performing single features of the IUPAC training corpus are the same as for the SCAI corpus: 2-gram suffix (f_2), 3 gram Suffix (f_4) and POS (f_{26}) features. This shows that suffixes have good effect in classification of chemicals. Also the POS feature mostly has successful effect in distincting common English words from chemical entities. In fact 5 features are common within the top 10 best contributing features for the 2 data sets, SCAI and IUPAC training.

Figure 4.2 shows the recognition performance achieved using each feature for different classes in the IUPAC training corpus. According to the results of average F-score of each class (the last row of Table 4.2) and Figure 4.2 it can be seen that again the entities that belong to classes MODIFIER and IUPAC can be recognized more successfully compared to entities in the PARTIUPAC class. Furthermore, the top ten best performing features in Table 4.2 are members of affixes and lexical features. But unlike the obtained results for the SCAI corpus, in the IUPAC training corpus space features do not have high contribution. It can be expected that the combination of affixes and lexical features can be effective in the NER task and are likely to exist in the subset of features that will be extracted by wrapper based search algorithms in the IUPAC training corpus.

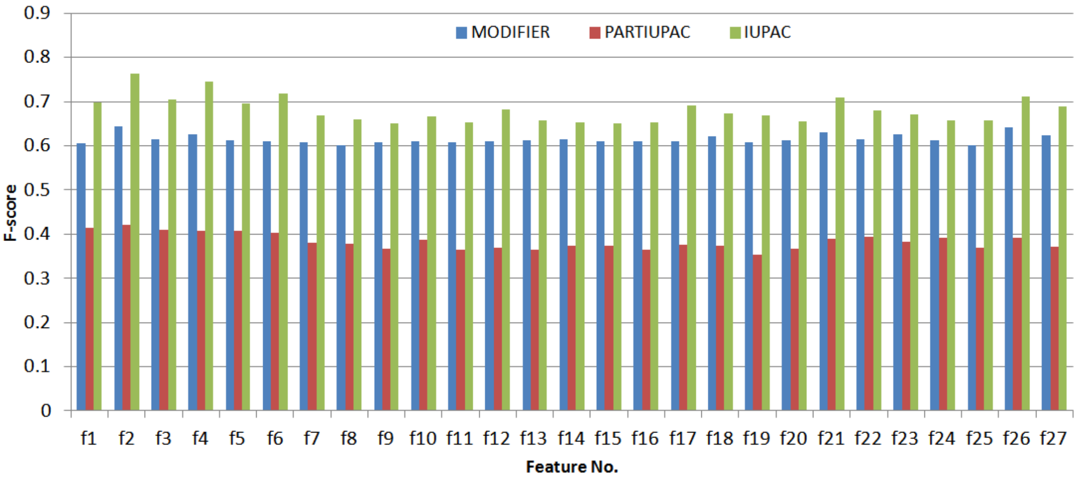

Figure 4.2: The Recognition Performance for each Class (IUPAC training Corpus)

4.1.3 Focus on the 2-gram Suffix Feature

IUPAC like names which include systematic and semi systematic chemical names, is one of the important and most common chemical entity class. Therefore recently researches have focused on improving the recognition of this class [20][21][23]. Considering both Table 4.1 and Table 4.2 the best performance obtained in predicting the names in IUPAC class is achieved by models which used the 2 gram suffix (f_2) feature. High F-scores of 0.6246 and 0.7621 are obtained on SCAI corpus and IUPAC training corpus respectively from CV experiments. An evaluation has been done to see the performance of the model using the feature f_2 in recognizing unseen IUPAC examples. Table 4.3 shows the corresponding results. Each time one corpus is selected as the training data and the other one as the test data. Then the recognition performance of names which belong to IUPAC class is calculated. When IUPAC training corpus is chosen as the training data set and evaluation is done on the SCAI corpus, the F-score for the IUPAC class is 0.6861.

Table 4.3 : F-score for IUPAC Class using 2-gram Suffix Feature

		Train Data	
		IUPAC training	**SCAI**
Test Data	**IUPAC training**	0.7621 (CV)	0.5784
	SCAI	**0.6861**	0.6246 (CV)

One reason of this result may be the success in recognizing long IUPAC names (e.g. methyl 3a-hydroxy-1,2,3,3 a,8,8 a- hexahydropyrrolo[2,3-b] indole -2-carboxylate). But the model still suffers from recognizing short IUPAC names which leads to misclassifying IUPAC names into classes such as MODIFIER and PARTIUPAC. In most cases they are also recognized as non-chemical entities which are labeled as OUT tags.

When the SCAI data is selected as train data and IUPAC training corpus as the test data the recognition performance is 0.5784. Further analysis of the lower performance reveals the difficulty in boundary recognition such as wrong classification of single letters or isolated numbers. Also, it causes a misclassification of long IUPAC names specially those which do not contain brackets or parentheses (e.g. 1- O-methyl-2-deoxy-3,5-di-O-p-toluoyl-D-ribofuranose). In this case, these long IUPAC names are mislabeled as OUT tags. Also 50% of mislabeled IUPAC names are recognized as PARTIUPAC names. This result may be due to the lesser number of IUPAC names contained in the SCAI corpus compared to the IUPAC training corpus. Therefore, the model which is trained by SCAI corpus is not as well trained and has higher generalization error.

In conclusion, although the 2-gram suffix (f_2) has good contribution to the recognition of IUPAC names in general, there may be small differences in the performance achieved using different training data. If there are not enough chemical entities in the training data in this class underfitting may take place which may lead to have high error. Nevertheless, the success of this feature in recognition of entities in the IUPAC class is not corpus dependent.

4.2 Investigating the Performance of Wrapper Based Feature Selection Algorithms

Generally, when all features are combined to train a system, it is likely that the subset may contain an overlap of some features. In such a case the performance of the classification system may not improve as desired. Therefore, feature selection is usually applied to find an optimal subset of useful and "well mixing" features.

In this section, feature subset selection has been considered using two training corpora SCAI and IUPAC training in order to investigate which subset of features is more successful in prediction of chemical named entities.

Three selection methods, Forward Search (FS), Backward Search (BS) and Simplified Forward Search (SFS) have been considered and their performance is compared to that of single best feature and the system which uses combination of all features.

4.2.1 Wrapper Based Feature Selection Using SCAI Corpus

Table 4.4 shows the subset of features obtained using wrapper based search algorithms through CV experiments on the SCAI corpus. For each category, the proportion of features selected by each method is shown.

Table 4.4: Feature Subsets used in SB, SFS, FS and BS methods (SCAI Corpus)

Method	Feature Set	Number Of Features	Morphological Features [1..6]	Orthographical Features [7..22]	Spaces [23..25]	Lexical Features [26,27]
SB	f_{2}	1	16.6%	-	-	-
SFS	$f_{2}.f_{4}.f_{19}.f_{21}.f_{22}.f_{23}.f_{26}$	7	33.3%	18.75%	33.3%	50%
FS	$f_{2}.f_{8}.f_{11}.f_{13}.f_{14}.f_{15}.f_{16}.f_{18}.f_{21}.f_{22}.f_{23}.f_{24}.f_{25}.f_{26}$	14	16.6%	56.25%	100%	50%
BS	$f_{1}.f_{2}.f_{5}.f_{7}.f_{8}.f_{9}.f_{11}.f_{13}.f_{14}.f_{15}.f_{16}.f_{17}.f_{18}.f_{19}.f_{20}.f_{21}.f_{22}.f_{23}.f_{26}.f_{27}$	20	50%	87.5%	33.3%	100%

Table 4.5 shows the common features between the feature subsets of wrapper based search algorithms in the SCAI corpus. Each time the common features between two methods are determined. Finally the common features between all three methods are achieved.

Table 4.5: Common Features among Feature Subsets used in SFS, FS and BS methods (SCAI Corpus)

Methods	Common features
(SFS, FS)	$f_{2}.f_{21}.f_{22}.f_{23}.f_{26}$
(FS, BS)	$f_{2}.f_{8}.f_{11}.f_{13}.f_{14}.f_{15}.f_{16}.f_{18}.f_{21}.f_{22}.f_{23}.f_{26}$
(BS, SFS)	$f_{2}.f_{19}.f_{21}.f_{22}.f_{23}.f_{26}$
(SFS, FS, BS)	**$f_{2}.f_{21}.f_{22}.f_{23}.f_{26}$**

It is seen that the common features between feature subsets of three selection methods are member of affixes, spaces and lexical features which have high contribution in recognizing chemical classes in the SCAI corpus. Furthermore, 12 features are common using FS and BS algorithms. Therefore it is expected that the recognition performance using both algorithms will be similar.

The results of Table 4.1 shows that space features have high contribution to the classification of entities in the SCAI corpus. Since space features are effective in boundary detection and FS method chooses all the space features, it is expected that the FS method will have a higher performance compared with other methods. Table 4.6 shows the results obtained by 10-fold Cross-Validation using SCAI corpus for all methods.

Table 4.6: Classification Performance using SB, SFS, FS, BS and All features (SCAI Corpus)

Method	Micro av. F-score	CLASS						
		ABBR.	MODIFIER	PARTIUPAC	TRIVIAL	IUPAC	SUM	FAMILY
		F-score						
FS	**0.548**	**0.2169**	0.5985	**0.5172**	0.5551	0.7141	0.0000	0.0851
BS	0.5425	0.1707	**0.6015**	0.4404	**0.5675**	0.7127	0.0000	0.0659
SB	0.4377	0.0432	0.4333	0.2424	0.4629	0.6246	0.0000	0.0460
SFS	0.5208	0.1169	0.5606	0.4673	0.5060	**0.7205**	0.0000	0.0444
All Features	0.5229	0.1274	0.5802	0.3846	0.5285	0.7097	0.0000	**0.087**

It can be seen that the performance of the methods can be sorted as FS, BS, All features, SFS and SB in terms of recognition performance. The best performance is achieved using the FS method, and the worst using the single best feature. Results show that on SCAI corpus, feature subset selection has improved the results by 11% compared to that of using of the single best feature. Also feature subset selection has improved the results by 2.5% compared to that using of the all features. The fact that SFS method has been less successful may be due to the fact that combination of best single features may lead to a subset of features which include many single features which are similar in nature. It is seen once again that the combination of best features may not always improve the classification performance as desired. The same may be true for combination of all features. As expected the performance of the FS and BS methods are similar due to high number of common features. As it was mentioned in section 4.1.1 IUPAC names in the SCAI corpus can be successfully classified when the classification model uses affixes, spaces and lexical features. This point can be strengthened by considering Table 4.6 which shows that SFS method which contains the mentioned features has the highest performance in recognition of IUPAC names.

It can be seen that each selection algorithm performs differently in predicting names which belong to different classes. Therefore a system which uses an ensemble of classifiers in addition to a combination of features may be suggested for future work. In such a case each classifier may be useful for the recognition of a specific chemical entity class.

4.2.2 Analysis of SCAI Data in terms of Class Distribution

It is well known that when standard classification algorithms are applied to unbalanced data, the algorithms lead to favor major classes where classification performance of minor classes may be poor [75]. In order to further investigate the reasons for different classification performance for different types of entities we have analyzed the SCAI data set in terms of how the data is distributed among different classes. The second column in Table 4.7 shows number of entities for each class in 70% of the SCAI corpus used as train data out of total number of entities in 100% of the same corpus.

Table 4.7: Entity Distribution of the SCAI Corpus

Class	No. of entities of each class in train data	FS Performance (CV) F-score
IUPAC	335 / 391	**0.7141**
MODIFIER	86 / 104	**0.5985**
TRIVIAL	338 / 414	**0.5551**
PARTIUPAC	77 / 92	**0.5172**
ABBREVIATION	103 / 161	0.2169
FAMILY	72 / 99	0.0851
SUM	28 / 49	0.0000

As it can be seen, the classification performance for entities in the major classes such as IUPAC, MODIFIER, TRIVIAL and PARTIUPAC is in general higher compared to minor classes.

Although the number of entities in classes such as ABBREVIATION and FAMILY is comparable to the numbers that belong to MODIFIER and PARTIUPAC classes, still their success of recognition is low. This may due to the similarity of these entities in structure with entities in other classes. One way to possibly improve recognition in these classes may be to use dictionaries.

4.2.3 Wrapper Based Feature Selection Using IUPAC training Corpus

Table 4.8 shows the results of the same set of experiments repeated using IUPAC training corpus. All the results given are obtained using 3-fold Cross-Validation.

Table 4.8: Feature Subsets used in SB, SFS, FS and BS methods (IUPAC training Corpus)

Method	Feature Set	Number Of Features	Morphological Features [1..6]	Orthographical Features [7..22]	Spaces [23..25]	Lexical Features [26,27]
SB	f_2	1	16.6%	-	-	-
SFS	$f_2.f_3.f_4.f_6.f_{21}.f_{26}$	6	66.6%	6.25%	0%	50%
FS	$f_1.f_2.f_4.f_6.f_{17}.f_{21}.f_{22}.f_{23}.f_{26}.f_{27}$	10	66.6%	18.75%	33.3%	100%
BS	$f_1.f_2.f_3.f_6.f_7.f_8.f_{10}.f_{12}.f_{13}.f_{16}.f_{17}.f_{18}.f_{19}.f_{20}.f_{22}.f_{23}.f_{24}.f_{25}.f_{26}.f_{27}$	20	66.6%	68.75	100%	100%

Table 4.9 shows the common features between the feature subsets of wrapper based search algorithms in the IUPAC training corpus.

Table 4.9: Common Features among Feature Subsets used in SFS, FS and BS methods (IUPAC training Corpus)

Methods	Common features
(SFS, FS)	$f_2.f_4.f_6.f_{21}.f_{26}$
(FS, BS)	$f_1.f_2.f_6.f_{17}.f_{22}.f_{23}.f_{26}.f_{27}$
(BS, SFS)	$f_2.f_3.f_6.f_{26}$
(SFS, FS, BS)	$\mathbf{f_2.f_6.f_{26}}$

It is seen that the common features between feature subsets of three selection methods are member of affixes and lexical features which have high contribution in recognizing chemical classes in the IUPAC training corpus. Again the most number of features are shared between the FS and BS methods. Table 4.10 shows the results obtained by 3-fold Cross-Validation using IUPAC training corpus for all methods.

Table 4.10: Classification Performance using SB, SFS, BS and All features (IUPAC training Corpus)

Method	Micro Average F-score	CLASS		
		MODIFIER	PARTIUPAC	IUPAC
		F-score		
FS	**0.7581**	**0.6978**	**0.4664**	**0.7937**
BS	0.7495	0.6891	0.4471	0.7861
SB	0.7229	0.6433	0.4201	0.7621
SFS	0.7517	0.6770	0.4593	0.7900
All Features	0.7394	0.6751	0.4420	0.7768

It can be seen that the performance of the methods can be sorted as FS, SFS, BS, All features and SB in terms of recognition performance.

The best performance is achieved using the FS method, and the worst using the single best feature. Results show that feature selection leads to a 3.5% and 1.8% improvement in comparison to using SB and all features, respectively. These results are in agreement with the ones obtained using the SCAI corpus. Furthermore, it can be seen that the FS method is most successful in classifying all entity types compared to all the methods.

The result related to the recognition of IUPAC class entities is the same as the one concluded in section 4.1.2. Since entities in this class can be successfully recognized using affixes and lexical features, both FS and SFS methods which choose features from these families are proven to be effective in recognition of entities in this class.

We have further examined the performance of the algorithms considered using the IUPAC test corpus. Results obtained using the test corpus for different selection algorithms are given in Table 4.11.

Table 4.11: Classification Performance of different selection algorithms (IUPAC test Corpus)

Method	Micro Average F-score
FS	0.3796
BS	0.4161
SB	**0.5011**
SFS	0.3387
All Features	0.4142

Since the IUPAC class is a major class in both IUPAC training and IUPAC test corpora, and while the Single Best feature is very effective in recognizing entities that belong to this class, SB classification system which uses feature (f_2) achieves the highest results in comparison to other methods.

In conclusion, considering the results using both SCAI and IUPAC training corpora, although SVM is usually know to be 'immune' for feature selection, experiments show that there is still room for improvement using wrapper based selection algorithms. High F-score values are obtained using FS and BS methods compared to the case where all the features are used or single best feature is used.

4.2.4 Comparison of Classifier Performance with Different Number of Classes using SCAI Corpus

So far the results presented belong to the classification of chemical entities with different specific classes. For the case of SCAI corpus the classification problem was assigning a class label out of 8 possible classes (7 different entities and OUT class) which is a multi-class classification problem. Often, it is only required to label a token as a "chemical entity" or a "non-chemical entity". In this case the classification problem is reduced to the binary case. Table 4.12 shows the results obtained both for the CV case and the case where 70% data is used as train data and the other 30% as test data.

Table 4.12: Evaluation of SCAI Corpus with Different No. of Classes

<table>
<tr><th><b>No. of Classes used for Training & Testing</b></th><th colspan="2"><b>7</b></th><th colspan="2"><b>2</b></th></tr>
<tr><td rowspan="2"><b>Method</b></td><td>CV</td><td>Using
Train| Test Data</td><td>CV</td><td>Using
Train| Test Data</td></tr>
<tr><td colspan="4"><b>Micro Average F-score</b></td></tr>
<tr><td>FS</td><td>0.5480</td><td>0.5136</td><td><b>0.5961</b></td><td><b>0.5853</b></td></tr>
<tr><td>BS</td><td>0.5425</td><td>0.5045</td><td><b>0.5949</b></td><td><b>0.5815</b></td></tr>
<tr><td>SB</td><td>0.4377</td><td>0.4173</td><td><b>0.4794</b></td><td><b>0.4718</b></td></tr>
<tr><td>SFS</td><td>0.5208</td><td>0.4959</td><td><b>0.5842</b></td><td><b>0.5785</b></td></tr>
<tr><td>All Features</td><td>0.5229</td><td>0.4972</td><td><b>0.5894</b></td><td><b>0.5803</b></td></tr>
</table>

As expected due to the fact that the classification problem has become an easier task, the results have improved in all cases.

4.3 Cross Corpus Annotation

In this section we performed experiments where different classification systems are trained and tested on different available corpora in an attempt to discover which of the available corpora is more suitable for a general CHEM-NER task.

Since the two corpora contain different number of classes (SCAI: 7, IUPAC: 3) three different experiments are carried out for each feature selection algorithm. In the first experiment the classifier is trained using the original train data, with all available classes in the data set. In the second experiment the number of classes on the train data is set as 3 keeping only the classes available in the IUPAC training data set (which contains the subset of classes in the SCAI set). The other tokens which are originally labeled as chemical entities in the SCAI data are annotated as the 'OUT' class. In the third case, all chemical entities are marked simply as "chemical entities”, regardless of their specific classes, and all remaining tokens as “non-chemical entities”, a two-class problem.

4.3.1 Train on SCAI and Test on IUPAC training corpus using Different Number of Classes

The results obtained for different classification systems using the SCAI corpus as train data and the IUPAC training corpus as test data are given in Table 4.13. It is seen that the classification models trained using SCAI data set are generally not very successful in predicting the entities in the IUPAC training corpus successfully.

The result of experiment 1 is low since the system is trained using samples from 7 entity classes but tested with only 3 classes. The performance for experiment 2 is higher since the system is trained using samples that belong to entity classes that are only available in the test set. The reason why the results achieved for the binary classification problem are in general lower compared to the multi-class experiments (except for the case of SB algorithm) needs further investigation. However, one reason may be due to the fact that, in the binary case there is more number of entities that need to be correctly classified. Some of these entities are known to be very different to predict due to their syntax and structure. Since IUPAC training data set mostly contains entities in the IUPAC class which can be successfully classified using the 2 gram suffix feature due to its structure the Single Best classifier which uses this feature only achieves the best performance in all three experiments among other selection algorithms.

Table 4.13: Comparison of Classification Performance using Different number of classes for IUPAC training Corpus

	Train on SCAI Corpus		
No. of Classes used for Training	7	3	2
	Test on IUPAC training Corpus		
No. of Classes used for Testing	3	3	2
Method	**Micro Average F-score**		
FS	0.2250	0.2621	0.1957
BS	0.3585	0.4051	0.3214
SB	**0.4831**	**0.5266**	**0.5636**
SFS	0.3399	0.3897	0.3171
All Features	0.3261	0.3633	0.2952

4.3.2 Train on IUPAC training corpus and Test on SCAI using Different Number of Classes

The IUPAC training corpus only contains three classes, IUPAC, PARTIUPAC and MODIFIER. So a classification model constructed using IUPAC training data as the training set is only capable of predicting these tokens into three classes which are common classes between this corpus and the SCAI corpus used for testing.

The results of the three experiments described in section 4.3 using IUPAC training data as the train corpus and SCAI data as the test corpus are shown in Table 4.14. As expected the classification performance is lowest in the first experiment since the test data contains entities belonging to the classes not available in the train data set.

Classification performance improves in the second experiment dramatically when the test set only contains entities available in the train set. Also the classification performance for the binary case is higher. The reason why the performance in this experiment is lower than the second experiment needs further investigation, since the

2-class problem is assumed to be an easier classification. One reason may be the fact that in the third experiment there are far more entities that need to be predicted, some of which are known to be difficult to predict. On a separate note the performance of the FS method is better than those of other algorithms for all three experiments, in agreement with the results of Table 4.6 and Table 4.10.

Table 4.14: Comparison of Classification Performance using Different number of classes for SCAI Corpus

	Train on IUPAC training		
No. of Classes used for Training	3	3	2
	Test on SCAI		
No. of Classes used for Testing	7	3	2
Method	**Micro Average F-score**		
FS	**0.3791**	**0.6429**	**0.4645**
BS	0.3788	0.6401	0.4548
SB	0.3759	0.6390	0.4540
SFS	0.3758	0.6388	0.4509
All Features	0.3721	0.6385	0.4494

4.3.3 Extending CHEM-NER Annotation to the CHEBI Corpus

In this section, SVM classifiers will be trained on SCAI and IUPAC training corpora separately for tagging entities in the CHEBI corpus, which is a corpus where each token is simply marked as a “chemical entity” or “non-chemical entity”. Different experiments using classification models making use of features extracted based on the previously mentioned wrapper based feature selection algorithms will be carried out. The purpose of these experiments is to investigate which of the two corpora is more appropriate for correctly labeling entities in a previously unseen corpus, CHEBI.

As it was mentioned in section 3.2.3, CHEBI is a molecular dictionary which mostly contains trivial and trade names. So in order to achieve a high performance in the NER task on CHEBI corpus, the classification model should be able to recognize trivial names with high success. In these experiments SCAI and IUPAC training corpus are used as training data sets. Since both of these corpora contain multiple classes whereas in CHEBI tokens are marked as "chemical entities" or "non-chemical entities" the SCAI and IUPAC training corpora are firstly converted to this structure. Table 4.15 shows a comparison between classification performance on CHEBI data set by using SCAI and IUPAC training corpora as training data sets. Here, unlike previous experiments all (100%) of SCAI data is used as train data.

Table 4.15: Classification Performance of Different Selection Algorithms for Different Training data sets using CHEBI Corpus

Training Corpus / **Method**	**SCAI**	**IUPAC training**
	Micro Average F-score	
FS	**0.4031**	0.0651
BS	0.3891	0.0533
SB	0.3154	0.0597
SFS	0.3995	0.0652
All Features	0.3893	0.0531

Results show that more successful results can be obtained using SCAI as the train data set in comparison to using IUPAC training corpus. This may be due to the fact that the entities in the CHEBI set, although not explicitly annotated as such, may in fact belong to classes in the SCAI set as stated in section 3.2.1. We have analyzed the predicted CHEBI which is tagged by the best model of Table 4.15 using SCAI train set classes. Table 4.16 shows the number of CHEBI entities which are classified to different classes.

Table 4.16: Number of Entities Tagged for each Class in the CHEBI Corpus

CLASS	No. of True Positives
IUPAC	2,118
PARTIUPAC	389
MODIFIER	126
FAMILY	55
SUM	52
TRIVIAL	2,581
ABBREVIATION	0
No. of Predicted Chemical Entities	5,321

As it was mentioned in the section 3.2.3 CHEBI mainly is composed of chemical entity classes such as TRIVIAL, IUPAC and sum formula. The result in Table 4.16 supports this claim. Although it is predicted that CHEBI corpus includes IUPAC names, a class frequently available in the IUPAC training corpus, the classifiers trained IUPAC training data are unsuccessful. Considering IUPAC entities in the CHEBI corpus it is seen that most of these entities are short IUPAC names, which are very hard to predict, compared to long ones. On the other hand, we know from Table 4.2 that the IUPAC training corpus does not contain entities in the FAMILY, SUM and TRIVIAL classes. Therefore models which are trained on this corpus could not predict entities in these classes as chemical entities in the CHEBI corpus. This is one reason for obtaining low result compared to using SCAI corpus as training data set. Nevertheless, the extremely low performance obtained using classifiers trained on IUPAC training data set need further investigation.

4.3.3.1 Investigating the Effect of Dictionary Feature on the Recognition Performance of the SVM

In this section, a dictionary feature is used to test if this feature can improve classification performance for CHEM-NER task. The dictionary feature used in this study is established in the following way: Each entity in the text will be compared with the entries in a dictionary. The presence of a dictionary feature is marked as '1' and its absence as '0' in the feature vector. To achieve a good result from this method, there is a need for a comprehensive and up-to-date dictionary and an efficient matching algorithm. In this study a text corpus with biomedical entities generated by CALBC (Collaborative Annotation of a Large Biomedical Corpus) [76] is used as data resource and exact matching is considered as the matching algorithm. The result of Table 4.15 shows that the best system for correctly labeling entities in the CHEBI corpus is a model which is trained on SCAI corpus using the FS algorithm. The dictionary feature is added to the features selected by the FS algorithm. Then a SVM classifier is trained on SCAI corpus for tagging entities in the CHEBI corpus. Table 4.17 shows the results of the classification with and without using dictionary feature in terms of F-score.

Table 4.17: Effect of using Dictionary Feature on the CHEBI Corpus

Without Dictionary Feature	With Dictionary Feature
Micro Average F-score	
0.4031	**0.4333**

It can be seen that using dictionary feature has improved the classification performance about 3%. Statistics presented in Table 4.18 show that using the dictionary feature mostly increase the number of entities labeled as IUPAC, TRIVIAL and FAMILY classes. On the other hand, the number of entities labeled as MODIFIER class has been decreased. Overall 6,116 entities are labeled as chemical entities when using the dictionary feature in comparison to 5,321 entities when dictionary is not used. This has resulted in the improvement in the result of system which has improved the overall F-score. It is also well known that the use of dictionaries generally improves recall.

Table 4.18: Comparison of the Number of Entities Predicted in each Class with and without Dictionary Feature on the CHEBI Corpus

CLASS	**Without Dictionary Feature**	**With Dictionary Feature**
	No. of True Positives	No. of True Positives
IUPAC	2,118	2,278
PARTIUPAC	389	405
MODIFIER	126	101
FAMILY	55	123
SUM	52	53
TRIVIAL	2,581	3,156
ABBREVIATION	0	0
No. of Predicted Chemical Entities	5,321	6,116

Further analysis reveals that using the dictionary feature was very efficient in boundary detection. For example the entity “ethyl esters” in the annotated CHEBI was tagged as ethyl/”B-Chemical” esters/”I-Chemical”. Before using the dictionary feature the tags were as following: ethyl/”B-Chemical” esters/’OUT’. But using dictionary gave the correct tags. Although using dictionary feature improves the result only by 3% the result are promising. The reason for this low improvement may be due to the type of resource which has been used as the dictionary. The used resource is not a chemical text corpus; it mostly includes biomedical entities which are very different than chemical entities. The other reason may refer to the fact that the resource is outdated compared to the chemical corpora. This issue is very important for CHEM-NER task for high exploiting new chemical entities. In conclusion, the use of a more updated and comprehensive dictionary may improve the results to a great extent.

4.3.4 Scoring Feature Selection Algorithms in terms of Classification Performance

Table 4.19 summarizes the set of classification experiments done in this study. Each set includes the results of 5 feature subset selection methods. For a given train-test set the last column shows the score given to the method according to its rank. The best performance receives a score 5 and the worst receives 1. The total score received by each method is shown in Table 4.20.

Table 4.19: Scoring of Feature Subset Selection Methods in terms of Classification Performance

No.	Train set	Test Set	Method	No. of Features	No. of Classes of Train data set	Micro-average F-score	Score
1	70 %SCAI	30%SCAI	FS	14	8	0.5136	5
2	70 %SCAI	30%SCAI	BS	20	8	0.5045	4
3	70 %SCAI	30%SCAI	SB	1	8	0.4173	1
4	70 %SCAI	30%SCAI	SFS	7	8	0.4959	2
5	70 %SCAI	30%SCAI	All Feat.	27	8	0.4972	3
6	IUPAC training	IUPAC test	FS	10	4	0.3796	2
7	IUPAC training	IUPAC test	BS	20	4	0.4161	4
8	IUPAC training	IUPAC test	SB	1	4	0.5011	5
9	IUPAC training	IUPAC test	SFS	6	4	0.3387	1
10	IUPAC training	IUPAC test	All Feat.	27	4	0.4142	3
11	100%SCAI	IUPAC training	FS	14	8	0.2250	1
12	100%SCAI	IUPAC training	BS	20	8	0.3585	4
13	100%SCAI	IUPAC training	SB	1	8	0.4831	5
14	100%SCAI	IUPAC training	SFS	7	8	0.3399	3
15	100%SCAI	IUPAC training	All Feat.	27	8	0.3261	2
16	IUPAC training	100%SCAI	FS	10	4	0.3791	5
17	IUPAC training	100%SCAI	BS	20	4	0.3788	4
18	IUPAC training	100%SCAI	SB	1	4	0.3759	3
19	IUPAC training	100%SCAI	SFS	6	4	0.3758	2
20	IUPAC training	100%SCAI	All Feat.	27	4	0.3721	1
21	100%SCAI	CHEBI	FS	14	2	0.4031	5
22	100%SCAI	CHEBI	BS	20	2	0.3891	2
23	100%SCAI	CHEBI	SB	1	2	0.3154	1
24	100%SCAI	CHEBI	SFS	7	2	0.3995	4
25	100%SCAI	CHEBI	All Feat.	27	2	0.3893	3
26	IUPAC training	CHEBI	FS	10	2	0.0651	4
27	IUPAC training	CHEBI	BS	20	2	0.0533	2
28	IUPAC training	CHEBI	SB	1	2	0.0597	3
29	IUPAC training	CHEBI	SFS	6	2	0.0652	5
30	IUPAC training	CHEBI	All Feat.	27	2	0.0531	1

It can be seen that the FS algorithm receives the highest score, which means it is the most successful search algorithm among other methods used in this study. It can further be seen that the methods can be sorted as FS, BS, SB, SFS and all features in terms of recognition performance.

Table 4.20: Summary of Scores Received by each Method

	Feature Subset Selection Method				
	FS	BS	SB	SFS	All Features
Scores	5	4	1	2	3
	2	4	5	1	3
	1	4	5	3	2
	5	4	3	2	1
	5	2	1	4	3
	4	2	3	5	1
Total Score	**22**	20	18	17	13

Chapter 5

CONCLUSION

In this thesis, the chemical named entity recognition problem is investigated using various corpora and different wrapper based feature selection algorithms. SVM is considered as the supervised machine learning algorithm which is intensively appropriate for high dimensional data in NER tasks.

SCAI, IUPAC training, IUPAC test and CHEBI corpora are used for either training or test data sets. Based on the morphology of chemical entities in the feature extraction step, several features have been extracted from the data sets. Wrapper based feature subset selection is used to obtain optimal subset of features for the classification task. The results of our experiments show that feature selection enhances the performance of the SVM classifier in comparison to the cases when all features extracted are used by the SVM. Furthermore, the result indicates that the Forward Search algorithm has achieved the best performance in the CHEM-NER task in terms of F-score.

Our machine learning system mostly concentrates on recognizing successfully systematic nomenclatures such as IUPAC and IUPAC-LIKE names; however we have shown that the performance of the system can be improved further by using a dictionary feature which increases the recognition performance of non systematic chemical names such as TRIVIAL and FAMILY names similar to [23].

We have also analyzed the classification performance using all available corpora, in search of a “best corpus”. Results show that although the SCAI data set is more comprehensive as it contains chemical entities that belong to more classes, its classification performance on unseen data may be limited due to its comparatively small size. On the other hand, the IUPAC training data set is fairly larger and seems more suitable for training a classification system. However, it has two main drawbacks: it consists of only 3 classes and is very unbalanced in terms of the number of samples in each class. Nevertheless, despite its size, the SCAI data set seems to be more suitable for the binary classification task where the third set, CHEBI data set, is annotated.

Future works may include the following:

1- The exploration of the usefulness of the algorithms and features discussed in this thesis to the newly released gold standard CHEM-NER data set, for the BioCreative IV challenge.

2- The exploration of the usefulness of the feature selection algorithms using a different machine learning algorithm such as Conditional Random Fields.

3- The construction of a NER system which uses an ensemble of classifiers where each base classifier is designed for specific class.

4- The exploration of the usefulness of using a comprehensive and up-to-date chemical dictionary such as Jochem.

5- Using FS algorithm to select the best subset for each class and employ an ensemble of classifiers for the final classification task.

REFERENCES

[1] Tan, P. N. (2007). Introduction to data mining. Pearson Education India.

[2] Grobelnik, M., Mladenic, D., & Milic-Frayling, N. (2000). Text mining as integration of several related research areas: report on KDD's workshop on text mining 2000. ACM SIGKDD Explorations Newsletter, 2(2), 99-102.

[3] Zanasi, A. (2009, January). Virtual weapons for real wars: text mining for national security. In Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems CISIS’08 (pp. 53-60). Springer Berlin Heidelberg.

[4] Cohen, K. B., & Hunter, L. (2008). Getting started in text mining. PLoS computational biology, 4(1), e20.

[5] Doms, A., & Schroeder, M. (2005). GoPubMed: exploring PubMed with the gene ontology. Nucleic acids research, 33(suppl 2), W783-W786.

[6] Coussement, K., & Van den Poel, D. (2008). Integrating the voice of customers through call center emails into a decision support system for churn prediction. Information & Management, 45(3), 164-174.

[7] Coussement, K., & Van den Poel, D. (2008). Improving customer complaint management by automatic email classification using linguistic style features as predictors. Decision Support Systems, 44(4), 870-882.

[8] Cambria, E., Speer, R., Havasi, C., & Hussain, A. (2010). Senticnet: A publicly available semantic resource for opinion mining. Artificial Intelligence, 14-18.

[9] Dridan, R., Kordoni, V., & Nicholson, J. (2008). Enhancing Performance of Lexicalised Grammars. In ACL (pp. 613-621).

[10] Kim, J. D., Ohta, T., Pyysalo, S., Kano, Y., & Tsujii, J. I. (2009, June). Overview of BioNLP'09 shared task on event extraction. In Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task (pp. 1-9). Association for Computational Linguistics.

[11] Brown, F. K. (1998). Chemoinformatics: what is it and how does it impact drug discovery. Annual reports in medicinal chemistry, 33, 375-384.

[12] Brown, F. (2005). Editorial opinion: chemoinformatics-a ten year update. Current opinion in drug discovery & development, 8(3), 298.

[13] Vazquez, M., Krallinger, M., Leitner, F., & Valencia, A. (2011). Text mining for drugs and chemical compounds: methods, tools and applications. Molecular Informatics, 30(67), 506-519.

[14] Narayanaswamy, M., Ravikumar, K. E., Vijay-Shanker, K., & Ay-shanker, K. V. (2003). A biological named entity recognizer. In Pac Symp Biocomput (p. 427).

[15] Kemp, N., & Lynch, M. (1998). Extraction of information from the text of chemical patents. 1. identification of specific chemical names. Journal of chemical information and computer sciences, 38(4), 544-551.

[16] Kolářik, C., Hofmann-Apitius, M., Zimmermann, M., & Fluck, J. (2007). Identification of new drug classification terms in textual resources. Bioinformatics, 23(13), i264-i272.

[17] Rebholz-Schuhmann, D., Kirsch, H., Arregui, M., Gaudan, S., Riethoven, M., & Stoehr, P. (2007). EBIMed—text crunching to gather facts for proteins from Medline. Bioinformatics, 23(2), e237-e244.

[18] Corbett, P., Batchelor, C., & Teufel, S. (2007, June). Annotation of chemical named entities. In Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing (pp. 57-64). Association for Computational Linguistics.

[19] Sun, B., Tan, Q., Mitra, P., & Giles, C. L. (2007, May). Extraction and search of chemical formulae in text documents on the web. In Proceedings of the 16th international conference on World Wide Web (pp. 251-260). ACM.

[20] Rocktäschel, T., Weidlich, M., & Leser, U. (2012). ChemSpot: a hybrid system for chemical named entity recognition. Bioinformatics, 28(12), 1633-1640.

[21] Klinger, R., Kolářik, C., Fluck, J., Hofmann-Apitius, M., & Friedrich, C. M. (2008). Detection of IUPAC and IUPAC-like chemical names. Bioinformatics, 24(13), i268-i276.

[22] Tiago, G., Catia, P., & Bastos Hugo, P. (2012). Chemical entity recognition and resolution to ChEBI. ISRN Bioinformatics, 2012.

[23] Kolárik, C., Klinger, R., Friedrich, C. M., Hofmann-Apitius, M., & Fluck, J. (2008). 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) (Vol. 36).

[24] McNaught, A. D., & Wilkinson, A. (1997). Compendium of chemical terminology (Vol. 1669). Oxford: Blackwell Science.

[25] http://www.ncbi.nlm.nih.gov/pubmed

[26] MOFFAT, A. A., & Bell, T. C. (1999). Managing gigabytes: compressing and indexing documents and images. Morgan Kaufmann.

[27] Cowie, J., & Lehnert, W. (1996). Information extraction. Communications of the ACM, 39(1), 80-91.

[28] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features (pp. 137-142). Springer Berlin Heidelberg.

[29] Freitag, D. (2000). Machine learning for information extraction in informal domains. Machine learning, 39(2-3), 169-202.

[30] Trybula, W. J. (1997). Data Mining and Knowledge Discovery. Annual Review of Information Science and Technology (ARIST), 32, 197-229.

[31] http://www.sigkdd.org/kddcup/index.php?section=2002&method=info

[32] Kim, J. D., Ohta, T., Tsuruoka, Y., Tateisi, Y., & Collier, N. (2004, August). Introduction to the bio-entity recognition task at JNLPBA. In Proceedings of the international joint workshop on natural language processing in biomedicine and its applications (pp. 70-75). Association for Computational Linguistics.

[33] http://www.biocreative.org

[34] http://trec.nist.gov/data/genomics.html

[35] http://www.bionlp-st.org/home

[36] Hayes, P. J., & Weinstein, S. P. (1990, May). CONSTRUE/TIS: A System for Content-Based Indexing of a Database of News Stories. In IAAI (Vol. 90, pp. 49-64).

[37] Makhoul, J., Kubala, F., Schwartz, R., & Weischedel, R. (1999, February). Performance measures for information extraction. In Proceedings of DARPA Broadcast News Workshop (pp. 249-252).

[38] Grishman, R., & Sundheim, B. (1996, August). Message Understanding Conference-6: A Brief History. In COLING (Vol. 96, pp. 466-471).

[39] Chinchor, N., & Marsh, E. (1998, July). Muc-7 information extraction task definition. In Proceeding of the Seventh Message Understanding Conference (MUC-7), Appendices (pp. 359-367).

[40] Donaldson, I., Martin, J., de Bruijn, B., Wolting, C., Lay, V., Tuekam, B., ... & Hogue, C. W. (2003). PreBIND and Textomy–mining the biomedical literature for protein-protein interactions using a support vector machine. BMC bioinformatics, 4(1), 11.

[41] Scanlan, M. J., Gordon, C. M., Williamson, B., Lee, S. Y., Chen, Y. T., Stockert, E., ... & Old, L. J. (2002). Identification of cancer/testis genes by database mining and mRNA expression analysis. International journal of cancer, 98(4), 485-492.

[42] Krallinger, M., Leitner, F., Rodriguez-Penagos, C., & Valencia, A. (2008). Overview of the protein-protein interaction annotation extraction task of BioCreative II. Genome Biology, 9(Suppl 2), S4.

[43] Gonzalez, G., Uribe, J. C., Tari, L., Brophy, C., & Baral, C. (2006). Mining gene-disease relationships from biomedical literature: weighting protein-protein interactions and connectivity measures. In Pac Symp Biocomput (pp. 28-39).

[44] Tari, L., Anwar, S., Liang, S., Cai, J., & Baral, C. (2010). Discovering drug–drug interactions: a text-mining and reasoning approach based on properties of drug metabolism. Bioinformatics, 26(18), i547-i553.

[45] Krauthammer, M., Rzhetsky, A., Morozov, P., & Friedman, C. (2000). Using BLAST for identifying gene and protein names in journal articles. Gene, 259(1), 245-252.

[46] Tuason, O., Chen, L., Liu, H., Blake, J. A., & Friedman, C. (2003). Biological nomenclatures: a source of lexical knowledge and ambiguity. In Proceedings of the Pacific Symposium of Biocomputing (No. 9, p. 238).

[47] Fukuda, K. I., Tsunoda, T., Tamura, A., & Takagi, T. (1998, January). Toward information extraction: identifying protein names from biological papers. In Pac Symp Biocomput (Vol. 707, No. 18, pp. 707-718).

[48] Proux, D., Rechenmann, F., Julliard, L., Pillet, V., & Jacq, B. (1998). Detecting gene symbols and names in biological texts: a first step toward pertinent information extraction. GENOME INFORMATICS SERIES, 72-80.

[49] Takeuchi, K., & Collier, N. (2005). Bio-medical entity extraction using support vector machines. Artificial Intelligence in Medicine, 33(2), 125-137.

[50] Settles, B. (2004, August). Biomedical named entity recognition using conditional random fields and rich feature sets. In Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (pp. 104-107). Association for Computational Linguistics.

[51] Finkel, J., Dingare, S., Nguyen, H., Nissim, M., Manning, C., & Sinclair, G. (2004, August). Exploiting context for biomedical entity recognition: From syntax to the web. In Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (pp. 88-91). Association for Computational Linguistics.

[52] Park, J. C., & Kim, J. J. (2006). Named entity recognition. Text mining for biology and biomedicine, 121-142.

[53] Kazama, J. I., Makino, T., Ohta, Y., & Tsujii, J. I. (2002, July). Tuning support vector machines for biomedical named entity recognition. In Proceedings of the ACL-02 workshop on Natural language processing in the biomedical domain-Volume 3 (pp. 1-8). Association for Computational Linguistics.

[54] Shen, D., Zhang, J., Zhou, G., Su, J., & Tan, C. L. (2003, July). Effective adaptation of a hidden markov model-based named entity recognizer for biomedical domain. In Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine-Volume 13 (pp. 49-56). Association for Computational Linguistics.

[55] Lin, Y. F., Tsai, T. H., Chou, W. C., Wu, K. P., Sung, T. Y., & Hsu, W. L. (2004). A Maximum Entropy Approach to Biomedical Named Entity Recognition. In BIOKDD (pp. 56-61).

[56] Ramshaw, L. A., & Marcus, M. P. (1999). Text chunking using transformation-based learning. In Natural language processing using very large corpora (pp. 157-176). Springer Netherlands.

[57] Bodenreider, O. (2004). The unified medical language system (UMLS): integrating biomedical terminology. Nucleic acids research, 32(suppl 1), D267-D270.

[58] http://www.biosemantics.org/index.php?page=Jochem

[59] Ristad, E. S., & Yianilos, P. N. (1998). Learning string-edit distance. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 20(5), 522-532.

[60] Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, 2(2), 121-167.

[61] Weston, J., & Watkins, C. (1998). Multi-class support vector machines. Technical Report CSD-TR-98-04, Department of Computer Science, Royal Holloway, University of London, May.

[62] Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, 2(2), 121-167.

[63] Takeuchi, K., & Collier, N. (2005). Bio-medical entity extraction using support vector machines. Artificial Intelligence in Medicine, 33(2), 125-137.

[64] http://chasen.org/~taku/software/yamcha/

[65] Mitsumori, T., Fation, S., Murata, M., Doi, K., & Doi, H. (2005). Gene/protein name recognition based on support vector machine using dictionary as features. BMC bioinformatics, 6(Suppl 1), S8.

[66] http://chasen.org/~taku/software/TinySVM/

[67] Liu, L. H., & Motoda, H. (Eds.). (1998). Feature extraction, construction and selection: A data mining perspective. Springer.

[68] Zhou, G., Zhang, J., Su, J., Shen, D., & Tan, C. (2004). Recognizing names in biomedical texts: a machine learning approach. Bioinformatics, 20(7), 1178-1190.

[69] Tsuruoka, Y., Tateishi, Y., Kim, J. D., Ohta, T., McNaught, J., Ananiadou, S., & Tsujii, J. I. (2005). Developing a robust part-of-speech tagger for biomedical text. In Advances in informatics (pp. 382-392). Springer Berlin Heidelberg.

[70] Konchady, M. (2006). Text Mining Application Programming (Programming Series). Charles River Media, Inc.

[71] Collier, N., & Takeuchi, K. (2004). Comparison of character-level and part of speech features for name recognition in biomedical texts. Journal of Biomedical Informatics, 37(6), 423-435.

[72] Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. Systems, Man and Cybernetics, IEEE Transactions on, 21(3), 660-674.

[73] Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. Artificial intelligence, 97(1), 273-324.

[74] Taira, H., & Haruno, M. (1999, July). Feature selection in SVM text categorization. In AAAI/IAAI (pp. 480-486).

[75] He, H., & Garcia, E. A. (2009). Learning from imbalanced data. Knowledge and Data Engineering, IEEE Transactions on, 21(9), 1263-1284.

[76] http://www.ebi.ac.uk/Rebholz-srv/CALBC/

APPENDIX

Appendix A: Statistics Measures

Different statistics measures such as Accuracy, Recall, Precession and F-score are used to measure the performance of system.

Confusion matrix is composed of 4 terms such as TP, FP, TN, and FN. True positive (TP) refers to number of positive samples which are classified correctly. True negative (TN) is number of negative examples which are identified correctly. False positive (FP) denotes number of negative examples which are classified incorrectly as positive examples and finally false negative (FN) indicates number of positive examples which are identified incorrectly as negative examples.

		Test tag	
		Positive	Negative
Train tag	Positive	TP	FN
	Negative	FP	TN

(1) Accuracy is the ratio of correctly classified examples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad Eq. 2.1$$

(2) Recall or sensitivity is the proportional of correctly classified positive examples.

$$Recall = \frac{TP}{TP + FN} \qquad Eq. 2.2$$

(3) Precision or positive predictive value (PPV) is the proportion of examples classified to be positive that were correct.

$$Precision = \frac{TP}{TP + FP} \qquad Eq. 2.3$$

(4) F-score is the harmonic average of recall and precision is used to measure the overall performance of classification task.

$$F - score = \frac{2 * precision * recall}{precision + recall} \qquad Eq. 2.4$$

For multi-class NER tasks to have overall F-score among all classes there is a need to compute the number of TP, TN, FP, FN for each class. There are two ways to determine the overall F-score: 1) by computing the average of the individual F-scores which is named as Macro-average F-score 2) by counting the total TP, FP, FN and TN for all NEs in the data set which is named as Micro-average F-score. In this study Micro-average F-score is used.