Knowledge-Based Systems 85 (2015) 37–51

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Joint model for feature selection and parameter optimization coupled with classifier ensemble in chemical mention recognition

Asif Ekbal *, Sriparna Saha

Department of Computer Science and Engineering, Indian Institute of Technology Patna, India

ARTICLE INFO

Article history:
Received 15 June 2014
Received in revised form 18 April 2015
Accepted 18 April 2015
Available online 25 April 2015

Keywords:
Feature selection
Parameter optimization
Multiobjective optimization (MOO)
Single objective optimization (SOO)
Conditional random field (CRF)
Support vector machine (SVM)

ABSTRACT

Mention recognition in chemical texts plays an important role in a wide-spread range of application areas. Feature selection and parameter optimization are the two important issues in machine learning. While the former improves the quality of a classifier by removing the redundant and irrelevant features, the later concerns finding the most suitable parameter values, which have significant impact on the overall classification performance. In this paper we formulate a joint model that performs feature selection and parameter optimization simultaneously, and propose two approaches based on the concepts of single and multiobjective optimization techniques. Classifier ensemble techniques are also employed to improve the performance further. We identify and implement variety of features that are mostly domain-independent. Experiments are performed with various configurations on the benchmark patent and Medline datasets. Evaluation shows encouraging performance in all the settings.

1. Introduction

Information extraction addresses the issues of finding relevant information from a huge collection of documents. Significant amount of information available in the web are unstructured. Information added to it daily are enormous in size, and therefore organizing, and finding relevant information poses an important challenge in our day-to-day life. In life science publications and patents, chemical compounds like small signal molecules or other biological active chemical substances are the important entity classes. Chemical names have different nomenclatures, and are represented in several forms. Some of the well-known representations include SMILES, InChI and IUPAC. While SMILES and InChI allow a direct structure search, IUPAC like names appear more frequent in biochemical texts. Dictionary based approach may be sufficient to identify the trivial chemical names, and to map these into their corresponding chemical structures. However enumerating all the IUPAC like names are not easier. Because of the varieties of interests, there has been a need to develop text mining methods for automatic identification of chemical compounds. It helps in meaningful search in the sense that it returns the documents that contain elements of the class, to which the entity belongs to. Ultimately this could be beneficial to find relations e.g. to adverse reactions or diseases.

Use of huge text corpora databases like Medline[1] [23,1] will be greatly facilitated if all the documents are well classified and ranked according to some similarity measurement. It provides an easier and faster access to the useful information relevant to the entities. Retrieval, extraction and gathering of information about the particular entities of our interest (for example, IUPAC-like names) at a high recall rate can be implemented. These can, thereafter, be mapped to the corresponding entries in the database. In order to automate the entire process there is a great demand to develop the efficient methods for text mining.

Developing a complete dictionary is the main constraint in traditional systems such as **dictionary based system**. Thus **rule based systems** and **machine learning (ML) based systems** are more popularly being used as these do not require any comprehensive dictionary. Regular expression based patterns are, in general, used to develop the rule based system [22]. Machine learning approaches, especially supervised algorithms need sufficiently large amount of annotated text for training. The learning algorithm makes use of this dataset to extract statistical information in order to capture the inherent dependencies in the data. The information is used to create a knowledge base that can classify, label or tag the unseen instances.

* Corresponding author.
E-mail addresses: asif@iitp.ac.in (A. Ekbal), sriparna@iitp.ac.in (S. Saha).
URL: http://www.iitp.ac.in/index.php/schools-and-centers/engineering/computer-science-a-engineering/people/faculty/dr-sriparna-saha.html (S. Saha).

[1] Online. Available at http://www.nlm.nih.gov/bsd/licensee/medpmmenu.html.

http://dx.doi.org/10.1016/j.knosys.2015.04.015
0950-7051/

The performance of any classification technique heavily depends on the features used for training/testing and the parameters used. Feature selection [21,20] is a technique that determines the most relevant set of features to build the robust machine learning models. This is also known as the variable selection, attribute selection or variable subset selection. By removing the most irrelevant and redundant features from the data, feature selection helps to improve the performance of a classifier. In general a classifier has several parameters whose values heavily influence its performance for any problem. Thus determining appropriate values of parameters of a classifier is another crucial issue. Here we formulate the problem of **appropriate feature and parameter selection** of a supervised machine learning algorithm as an optimization issue, and develop some evolutionary optimization based techniques to solve these.

In recent years few works have been reported that address the issues of feature and parameter selection in any supervised classifier. One such method is proposed in [7] that was evaluated for named entity recognition (NER) involving Indian languages. The work [7] describes a multiobjective optimization (MOO) based technique that automatically selects the most relevant set of features and parameters for two classification techniques, namely Conditional Random Field (CRF) [18] and Support Vector Machine (SVM) [25]. In contrast to [7], our current work focuses on to develop some automated techniques in order to perform detection and classification of chemical names in texts. The scenarios and challenges in Indian language and chemical domains are not similar. An information extraction system in the respective domain poses different challenges, and hence the processing steps and the attributes (or, features) are quite dissimilar. Extraction of relevant entities from the chemical text is more challenging compared to the Indian language text because of the appearance of very long and complex wordforms, presence of many symbols and common words inside the chemical names, etc. Because of the complexity involved in the current domain, more extensive feature set had to be implemented for it. Moreover in the current work we also employ a two-stage approach. In the first step two algorithms for automatic feature selection and parameter optimization are developed. In the second step, solutions obtained from the output of the first step are combined using the classifier ensemble techniques. It is also to be mentioned that we perform more extensive parameter optimization in our current work.

In [19], a cat swam optimization based feature and parameter optimization technique has been developed to automatically determine the relevant set of features and the value of kernel parameter for SVM. Here different data sets from UCI machine learning repository were utilized to evaluate the classification accuracy of the proposed technique. In [14], a particle swarm optimization (PSO) based approach is used to automatically determine the most relevant set of features and the kernel parameter for SVM. In [13], a feature and parameter selection technique is developed to improve the accuracy of solar power prediction. Authors made use of various machine learning techniques, namely Least Median Square (LMS), Multilayer Perceptron (MLP) and SVM. In [15] a genetic algorithm (GA) based feature and parameter selection technique has been developed for the SVM based classifier. Another GA based feature selection technique is developed in [11], where only the problem of feature selection was addressed under the single objective optimization (SOO) framework. In the current paper we deal with the problems of feature selection as well as parameter optimization. We formulate this in a joint model, and then solve using SOO as well as MOO based techniques. Thus the problems and the techniques used in [11] and in the current paper are completely different.

As we mentioned earlier, in our present work we formulate a joint model that is able to tackle the problems of feature selection and parameter optimization. The algorithm is comprising of two steps; first step of which deals with the feature selection and parameter optimization methods, and the second step makes use of an ensemble algorithm to combine the solutions of the first step. For SOO, GA [8] is used as the underlying optimization technique. For MOO, the most popular GA based technique, namely non-dominated sorting GA-II (NSGA-II) [5] is used as the underlying optimization technique. The proposed approaches are evaluated for the extraction of chemical mentions in the forms of IUPAC and IUPAC-like names from the text. We use CRF and SVM as the base learning algorithms. We implement variety of features, most of these are automatically extracted from the given training/test datasets. It is also to be noted that we did not make use of any heavy domain-specific resources and/or tools *except* the PubChem database.[2] Note that PubChem is a database that contains the chemical molecules and their activities against biological assays. The system is maintained by the National Center for Biotechnology Information (NCBI), a component of the National Library of Medicine, which is part of the United States National Institutes of Health (NIH). PubChem can be accessed for free through a web user interface.

The single objective GA based feature selection and parameter optimization method yields a set of solutions on the best population, whereas multiobjective NSGA-II yields a set of solutions on the final Pareto optimal front. The feature selection is performed for both the classifiers, CRF and SVM. The set of solutions obtained in the final population of these two after application of GA based technique are then merged together using a GA based classifier ensemble technique [6]. Similarly after the application of MOO based technique we obtain a set of solutions on the final Pareto optimal front for CRF and SVM each. We combine the outputs of these solutions using a MOO based classifier ensemble technique proposed in [24]. The methods are evaluated on the benchmark setup of patent and Medline datasets. For the SOO based approach we obtain the overall *F*-measure values of 74.05%, 74.00% and 88.49% for the patent test data sets of 2008, 2009 and Medline test data set, respectively. The multiobjective based approach yields the overall *F*-measure values of 75.55%, 76.07% and 89.77%, respectively, for the patent test data sets of 2008, 2009 and Medline test data set. Comparisons show that the proposed technique (i.e. automatic feature selection and parameter optimization) is more effective than the systems that make use of all the available features and default parameter values (i.e. baselines 1–4). We also show that a two-stage approach, where feature selection and parameter optimization are performed first followed by classifier ensemble, could be more effective compared to the approach that only performs feature selection and parameter optimization. It is evident that our proposed approach achieves the state-of-the-art performance.

2. Methods

In this section we describe the proposed approach for chemical name identification and classification. As already mentioned, we propose a technique to address two crucial issues of any machine learning algorithm, *viz.* feature selection and parameter optimization. We model these two problems jointly, and solve using SOO and MOO based techniques.

2.1. Problem formulation

The problem of feature selection and parameter optimization is formulated under the SOO as follows:

[2] ftp://ftp.ncbi.nlm.nih.gov/pubchem/Compound/CURRENT-Full/XML/.

Given a set of features Ω, appropriate parameters $\mathcal{P}$ and a classification quality measure D, determine the feature subset F^* and parameter subset P^* such that:

$$D(F^*, P^*) = \max_{F \in \Omega, P \in \mathcal{P}} D(F, P)$$

Chemical mention extraction (i.e. detection and classification)[3] is an important task in many text processing activities including information extraction [3]. The key task can be thought of as a two-step process that involves identification of every word/term and classifying them into some predetermined categories. The categories could vary depending upon the application domain. For example, in the current work the overall task is cast as the process of identification of chemical entities from the texts and classification of them into the classes that represent the IUPAC and IUPAC related entity types.

The kinds of tasks such as mention extraction can be evaluated using the standard measures such as recall, precision and *F*-measure. While recall tries to increase the number of correctly retrieved entities as much as possible, precision tries to reduce the number of incorrectly tagged entities. These two capture two different classification qualities. In case of single objective formulation of feature and parameter selection problem, we only optimize a function, namely the *F*-measure. The problem is then stated as follows:

Given a set of features Ω, appropriate parameters $\mathcal{P}$ and a classification quality measure, *F*-measure, determine the feature subset F^* and parameter subset P^* such that *maximize* [*F*-measure] where $F^* \subseteq \Omega$ and $P^* \subseteq \mathcal{P}$.

The problem of feature selection and parameter optimization within the framework of MOO is formulated as follows: Given a set of features Ω, appropriate parameters $\mathcal{P}$ and two classification quality measures, recall and precision, determine the feature subset F^* and parameter subset P^* such that *maximize* [recall, precision] where $F^* \subseteq \Omega$ and $P^* \subseteq \mathcal{P}$.

Here we optimize two different parameters of CRF. These are (a) the hyper-parameter of CRF. CRF tends to overfit to the training data with the larger values of this parameter. This parameter makes a balance between overfitting and underfitting. The results obtained in CRF can significantly be influenced with the tuning of this parameter. In general optimal value for this parameter can be determined by tuning on the development data or using the cross-validation techniques; (b) the second parameter determines the cut-off threshold for the features. CRF utilizes the features that appears no less than NUM times in the given training data. The default value is 1. In case of large data set, the number of unique features would amount to several millions. In such cases, setting the value of this parameter to a reasonably higher range may lead to the decrease in the number of features. This, in turn, reduces the overall complexity of the problem.

In case of SVM we use LibSVM[4] toolkit available with Weka Machine Learning Suite [10]. Kernel function plays a very vital role in SVM learning. In our work we automatically determine the values of the following parameters of SVM.

- **Type of kernel function**: it can be polynomial kernel, radial basis kernel, sigmoid kernel or linear kernel. Default is the polynomial kernel.
- **Degree of the kernel function**: Default value for this parameter is 3.
- **Coefficient of the kernel function**: Default value for this parameter is 0.
- **Value of gamma in the kernel function**: Default value for this parameter is $\frac{1}{number\ of\ features}$.
- **Epsilon width of tube for regression**: Default value for this parameter is 0.1.

In general the search space for this type of problems is huge. Thus, exhaustive search strategies cannot be applied in this case.

2.2. Genetic algorithms

Genetic algorithm(GA) [8] is a popularly used randomized search and optimization technique. The working principles are guided by evolution and natural genetics. Parameters of the search space are encoded in the forms of strings called chromosomes. The set of such strings is known as a population. At the beginning initial population is created randomly. Each of the chromosomes represents a point in the search space, and it is associated with an objective or fitness function. The fitness function denotes the degree of goodness of the corresponding chromosome. Based on the principle of survival of the fittest few chromosomes are selected that go into the mating pool. New generation of chromosomes is obtained by applying the biologically inspired operators like crossover and mutation. Crossover is an operation that is applied to generate an offspring chromosome by exchanging information of two parent chromosomes. The mutation operation is used to produce a new chromosome by changing parts of the parent chromosome according to a certain probability. The operations of selection, crossover and mutation are repeated for a fixed number of generations or till a termination condition is satisfied.

2.2.1. Nondominated sorting genetic algorithm-II (NSGA-II)

Genetic algorithms are known to be more effective for solving MOO than classical methods such as weighted metrics or goal programming, because of their population-based nature [4]. A particularly popular GA of this type is NSGA-II [5].

In NSGA-II, a random parent population P_0 is initially created and sorted based on the partial order defined by the non-domination relation. The result is a (sorted) sequence of non-dominated fronts $F_1, F_2, F_3, \ldots, F_n$. Each solution in the population is assigned a fitness value which is equal to its non-domination level in the partial order. A child population Q_0 of size N is created from the parent population P_0 by using recombination and mutation operators. Then iteration begins. At the tth iteration, a combined population $R_t = P_t + Q_t$ is formed. The sub-population P_t of size N contains the best solutions found so far, according to the partial order imposed by the non-domination relation. The population R_t is sorted, obtaining a sequence of non-dominated fronts. The algorithm keeps adding entire fronts to P_{t+1}, until the total size of P_{t+1} reaches N. To choose exactly N solutions, the solutions of the last included front are sorted using the crowded comparison operator [5] and the best among them (i.e., those with lower crowding distance) are selected to fill in the available slots in P_{t+1}. The crowded comparison operator measures the density of a particular solution: solutions which are in a less crowded regions are given higher priority to be selected. The new population P_{t+1} is then used for crossover and mutation to create a population Q_{t+1} of size N. The basic steps of the algorithm is shown in Fig. 1.

2.3. Single and multiobjective GA for joint model of feature selection and parameter optimization

Single objective GA and a multiobjective GA, along the lines of NSGA-II [5], are used as the underlying optimization techniques in order to solve our problems. Note that these algorithms are very general. In place of NSGA-II we could have used any other MOGA (multiobjective genetic algorithm) technique and in place of GA

[3] Also referred to as the task of named entity recognition.
[4] http://weka.wikispaces.com/LibSVM.

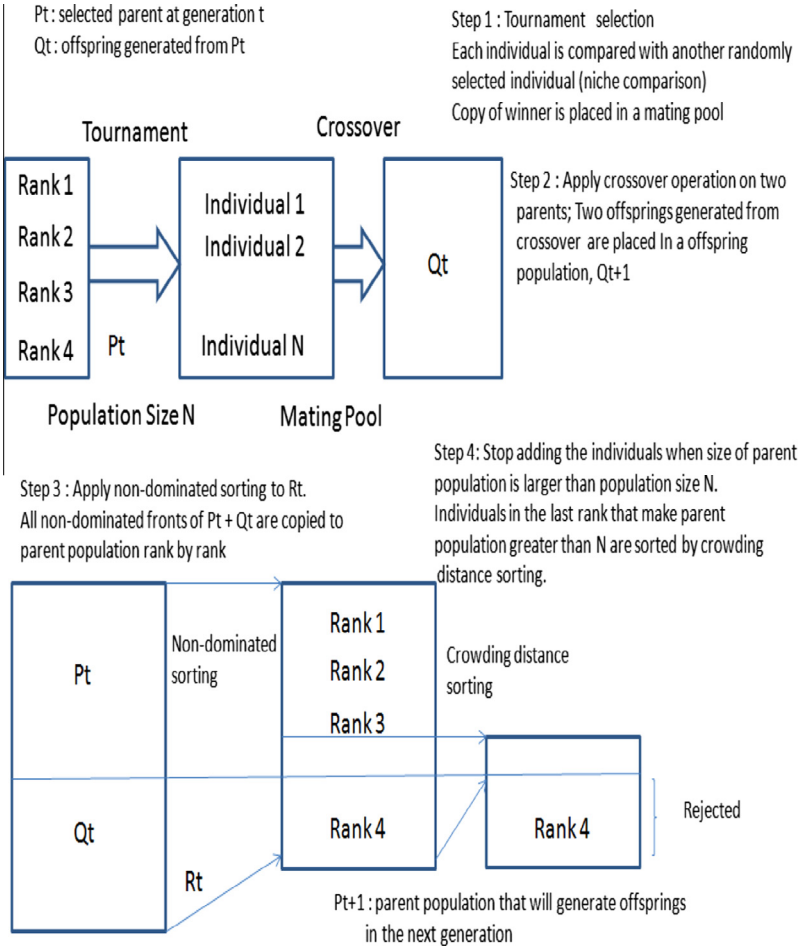

Fig. 1. NSGA-II procedure.

we could have used any other optimization technique like simulated annealing or differential evolution.

The basic steps of the proposed joint model for feature selection and parameter optimization are shown in Fig. 2. The steps are enumerated below:

- GA or NSGA-II is utilized to automatically determine the appropriate feature and parameter combination for a CRF-based classifier using the procedure mentioned below. After the complete execution, final population will contain a set of solutions; some may be good with respect to one classification quality measure, recall and some may be good with respect to the other classification quality measure, precision.
- Similarly GA or NSGA-II is also utilized to automatically determine the appropriate feature and parameter combination for a SVM based classifier. It yields a set of solutions at the end.
- The different solutions of CRF and SVM based approaches are combined using a classifier ensemble technique.

Different steps of the proposed MOO based technique are provided in Fig. 3.

2.3.1. Chromosome representation and population initialization

Let us assume that the number of features is F and the number of possible parameters is P. Thus, the length of the chromosome is $F + P$. In case of CRF, we determine the optimal values of "hyper-parameter for CRF" (denoted by c) and "cut off threshold for the number of features" (denoted by f). Thus the parameter values encoded in the chromosomes represent the values for these two parameters. As an example, the encoding of a particular chromosome is represented in Fig. 4. Here, $F = 8$ (i.e., total 8 different features are available) and $P = 2$. The chromosome represents the use of 4 features, i.e., first, third, sixth and eighth for constructing the particular CRF based classifier with the parameter values $c = 2.5$ and $f = 4$. The values of the features of each chromosome are randomly initialized to either 0 or 1. For feature representation, the value of 1 at the ith position represents that this feature is used for constructing the classifier, and the value of 0 represents that the feature is not used. The parameter values are initialized with some real values such as the following: The parameter i is randomly initialized to a real value (r) between P_i^{min} to P_i^{max}; $r = \frac{rand()}{RAND\_MAX+1} * (P_i^{max} - P_i^{min} + 1) + P_i^{min}$ where P_i^{min} is the minimum value of this parameter and P_i^{max} is the maximum possible value of this parameter. In case of CRF, we varied the values of c in the range of 0.5–2.5 and f in the range of 1–10. In case of SVM we optimize five different parameters: kernel function, degree of the kernel function, gamma value of the kernel function, coefficient value of the kernel function and epsilon parameter. The parameter for kernel function can take four values: 1, 2, 3 and 4 that denote the linear, polynomial, radial basis function and sigmoid kernel function, respectively. The parameter 'degree of kernel' is an integer entity which is varied between 2 and 6. The parameter 'gamma' value is a real entity and is varied in the range of 1/45–1/10. The value of parameter 'coefficient' is varied in the range of 0–4. Finally, the parameter 'epsilon' is varied in the range of 0.1–0.5. If the population size is *POP* then all the *POP* number of chromosomes of this population are initialized in the above way. By population, here we denote a set of chromosomes.

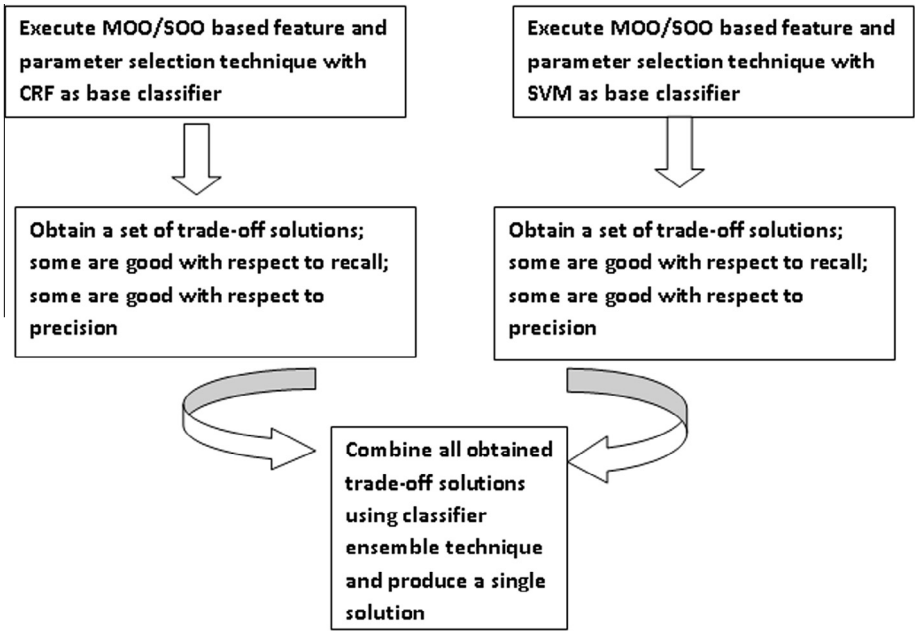

Fig. 2. Joint model for feature and parameter selection.

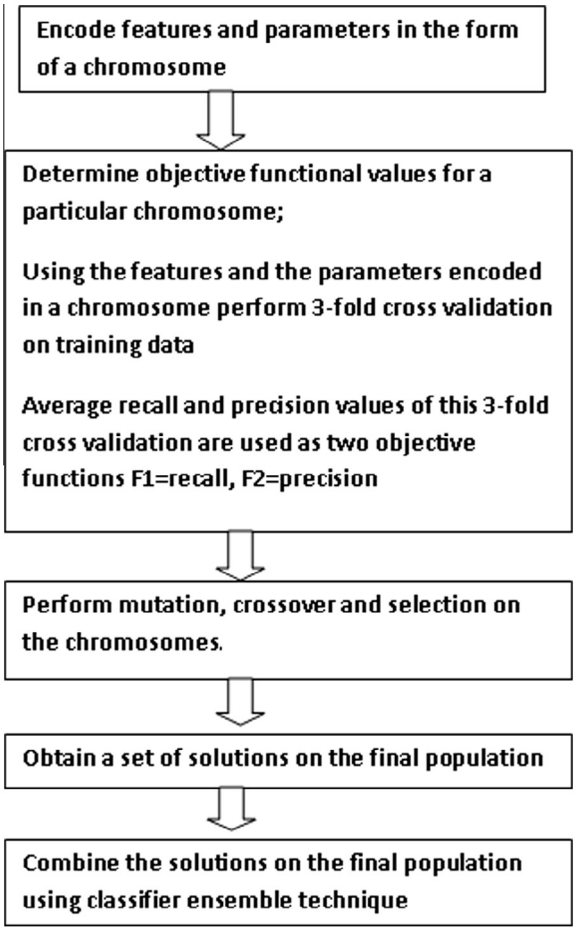

Fig. 3. Different steps of the proposed MOO based technique.

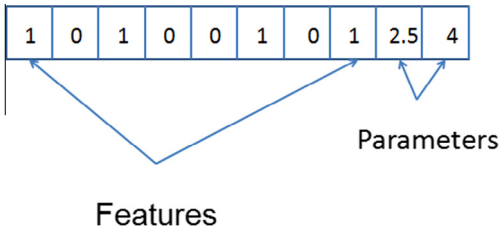

Fig. 4. Chromosome representation for SOO and MOO based feature and parameter selection.

2.3.2. Fitness computation

In order to compute the fitness we execute the following sequence of steps.

(1) Let us assume that there are N number of 1's in the chromosome. It represents that N features are present. Suppose P be the parameter combination for this chromosome.
(2) Construct the classifier with only these N features and P parameter combination encoded in the chromosome.
(3) The entire training set is divided into 3 parts. Train the classifier with 2/3 portions of training data with the set of features and parameters encoded in the corresponding chromosome. Evaluate classifier with the rest 1/3 portion.
(4) Compute the overall recall, precision and F-measure values of the classifier for the test data.
(5) Perform 3-fold cross validation by repeating the steps 2–4. Compute the overall average recall, precision and F-measure values of the classifier from this cross validation experiment.

In case of SOO, the objective function corresponding to a particular chromosome is $f = F\text{-measure}_{avg}$. The function is maximized using the search capability of GA.

In case of MOO, the objective functions corresponding to a particular chromosome are:

$$f_2 = \text{recall}_{avg} \text{ and } f_2 = \text{precision}_{avg}$$

These two objective functions are maximized using the search capability of NSGA-II.

2.3.2.1. Motivation of using recall and precision as two objective functions. The performance of any MOO algorithm greatly depends on the choice of the objective functions. The functions should be as much contradictory as possible. Here we choose recall and precision as the two objective functions. These two metrics are defined as below:

$$\text{recall} = \frac{\text{Number of entities correctly identified by the system}}{\text{Number of entities in the gold standard test data}} \quad (1)$$

$$\text{precision} = \frac{\text{Number of entities correctly identified by the system}}{\text{Number of entities identified by the system}} \quad (2)$$

The above definitions show that recall tries to increase the number of correctly retrieved entities as much as possible, but the goal of precision is to decrease the number of misclassified entities. Ideally these two metrics capture two different classification qualities, and often their relationships are inverse in the sense that one's high value may be obtained at the cost of others' low value. For example, in an information retrieval system, recall can be improved by retrieving more and more documents. This, however, may include many irrelevant documents, and hence precision may be affected at the greater extent.

Let us consider the following example:

Suppose there are 100 relevant entities; system identifies 600 entities and out of this only 60 entities are correct. As per the definition, recall of the system is 0.6 (i.e. 60/100) whereas precision is 0.1 (i.e. 60/600). In this case, however recall is acceptable, precision is very low. In another scenario, suppose the system identifies in total 20 entities, out of which 10 are correct. The recall of the system is 0.1 (10/100), whereas precision is 0.5 (10/20). Here precision seems to be acceptable but the recall is very low.

This is the underlying motivation of simultaneously optimizing these two objectives. The objective functions corresponding to a particular chromosome are $f_1 = \text{recall}_{avg}$ and $f_2 = \text{precision}_{avg}$. The objective is to: $max[f_1, f_2]$. These two objective functions are simultaneously optimized using the search capability of NSGA-II.

F-measure is a metric that combines both the recall and precision. The *F*-measure can be interpreted as a weighted average of precision and recall, where *F*-measure reaches its best value at 1 and worst score at 0. The *F*-measure is computed as below:

$$F\text{-measure} = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} \quad (3)$$

The harmonic mean of two numbers tends to be closer to the smaller of the two. For the *F*-measure value to be high, both *precision* and *recall* should be high.

It has been discussed very thoroughly in [4] that it is not possible to identify all the non-dominated solutions using a weighted sum approach. The solutions located on the convex part of the Pareto front can be found. The other important motivation of using MOO is that it provides a set of solutions, some of them may be good with respect to recall and some may be good with respect to precision. Finally we have combined the outputs of these classifiers using a classifier ensemble technique. Hence, MOO can indeed be a good candidate to solve these types of problems. Here, no weight is needed to combine the objectives, and hence no prior information about the problem is needed *apriori*. Moreover optimization of *F*-measure does not guarantee optimization of both recall and precision. Thus MOO is indeed needed to optimize recall and precision simultaneously.

Note that by optimizing *F*-measure using SOO based technique we can generate solutions on the best population which are good with respect to *F*-measure. But in order to get solutions which are good with respect to precision and in order to generate solutions which are good with respect to recall, we have to optimize recall and precision separately as two different objective functions. As already explained, depending upon the application we may require the solutions which are better with respect to either of these two objectives. Therefore, MOO and SOO both are necessary.

2.3.3. Genetic operators

For single objective GA, normal single point crossover [12] is used. For binary encoding each bit of the chromosome is randomly replaced by either 0 or 1. In contrast for real encoding each position of the chromosome is mutated with probability μ_m as follows. The bit position is updated with a random variable that is drawn from a *Laplacian distribution*, $p(\epsilon)e^{-\frac{|\epsilon-\mu|}{\delta}}$, where the scaling factor δ sets the magnitude of perturbation. Here, μ is the corresponding value at the position that is to be perturbed. A value equal to 0.1 is chosen to be assigned to the scaling factor δ. The newly generated value is used to replace the old value at the position. As a result of the generation of a random variable using *Laplacian distribution*, there will always be a non-zero probability of generating any valid position from the other valid positions. At the same time the probability of generating a new value near the old value is high. Proportional selection strategy is used to implement the Roulette wheel selection [8].

In case of multiobjective version the crossover operation is similar to that of NSGA-II. For mutation of the feature values, normal binary mutation operation of NSGA-II is used. For mutation of the parameter values the real mutation of NSGA-II is used. The method of feature and parameter optimization uses crowded binary tournament selection as in NSGA-II. The most distinguishing feature of NSGA-II is its elitism operation, where non-dominated sorting [4] operation is applied among the parent and child populations. The set of new solutions is propagated to the next generation.

2.3.4. Combining solutions of the final population

Both the approaches are executed for a fixed number of generations. In case of SOO, we obtain a set of solutions on the best population. The final solution is the one with the best *F*-measure value. Some of the solutions on the best population may have high recall values whereas some could have high precision values. Thus instead of selecting a single solution we use a SOO based classifier ensemble technique [6] to combine the solutions, obtained in the best population.

In case of MOO, two objective functions were optimized. The near-Pareto-optimal chromosomes of the last generation denote different solutions to the problems of feature selection and parameter optimization. Multiobjective optimization algorithm produces a number of non-dominated solutions [4] on the final Pareto optimal front. The solutions represent different feature and parameter subsets for the classifier. As per the algorithmic point of view each of these solutions is important, and none of these is better compared to the other. Thus it is very difficult to select a unique solution from the best population. Hence rather than selecting a solution we combine all the outputs of the classifiers using a MOO based classifier ensemble technique [24].

2.3.5. Selection of a single solution from Pareto optimal front

In another experiment we also select one solution from the final Pareto optimal front obtained at the end of feature selection and parameter optimization technique. In order to achieve this we execute the following steps. For all solutions on the final Pareto front we compute the *F*-measure values. The solution with the highest *F*-measure is chosen as the best one. We report this solution (from the first stage of our algorithm) for comparing with the baselines, especially the first four ones (refer to Section 4.3). This is not the unique process of selecting the best solution, and depending upon

the requirement there can be many others. For example, depending on the need one could also choose the solutions with the higher recall or precision values. Please note that this is not the final output of the system. This is an intermediate result. In our second setting we combine all the solutions of the final Pareto optimal front using a MOO based classifier ensemble technique [24]. This is done to further improve the performance as well as to compare with the last two baselines (i.e. baselines 5–6).

2.4. Evolutionary optimization based classifier ensemble

In this section, we present a solution to the problem of classifier ensemble under the SOO and MOO frameworks. The SOO and MOO based techniques are based on GA and NSGA-II, respectively. For SOO and MOO based ensemble learning we employ the algorithms proposed in [6,24], respectively.

2.4.1. String representation and population initialization

If the total number of available classifiers is M and total number of output tags (i.e., number of classes) is O, then the length of the chromosome is $M \times O$. Each chromosome encodes the weights of votes for possible O classes in each classifier. The encoding of a chromosome is represented in Fig. 5. We consider that total 9 votes are possible, i.e. $M = 3$ and $O = 3$. Combinations of three output classes in three classifiers are as follows:

Classifier-1: 0.59, 0.12 and 0.56;
Classifier-2: 0.09, 0.91 and 0.02;
Classifier-3: 0.76, 0.5 and 0.21.

We use real encoding, i.e. each bit of the chromosome is randomly initialized to a real value (r) that ranges between 0 and 1. Here, $r = \frac{rand()}{RAND\_MAX+1}$. If the population size is P then all the P number of chromosomes of this population are initialized in the above way.

2.4.2. Fitness computation

At first all the individual classifiers are trained on the training set and evaluated on the test set. Thereafter we execute the following steps in sequence.

(1) Suppose, there are total M number of classifiers. Let us assume that F-measure values of these M classifiers on the development set be $F_i, i = 1, \ldots, M$, respectively. It is to be noted that a part of the training set is used as the development set.
(2) The outputs of all the individual classifiers are combined to create an ensemble. For the ensemble the output label of each instance is determined based on the weighted voting of these M classifiers' outputs. Weight of the output class predicted by the ith classifier is equal to $I(m,i)$, where $I(m,i)$ denotes the particular entry in the chromosome that corresponds to mth classifier and ith class. The final weight assigned to a particular class for a particular word w is:

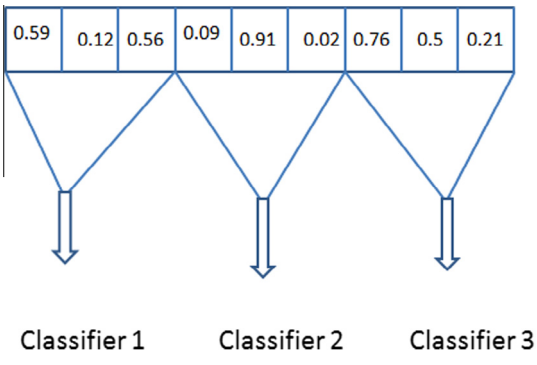

Fig. 5. Chromosome representation for real voting.

$$f(c_i) = \sum I(m,i) \times F_m,$$
$$\forall m = 1 : M \ \& \ op(w,m) = c_i$$

Here, $op(w,m)$ denotes the output class predicted by the mth classifier for the word w. The final output is assigned based on the highest weighted vote.

(3) For the development set, the overall recall, precision and F-measure values of the ensemble classifier are computed. In case of SOO, the objective function is the final F-measure. In case of MOO based approach, the objective functions corresponding to a particular chromosome are $f_1 =$ recall and $f_2 =$ precision. The main goal is to maximize these two objective functions using the search capability of NSGA-II.

2.4.3. Genetic operators used for MOO based approach

For GA based approach single point crossover and Roulette wheel selection [8] are used. For mutation we have used the Laplacian distribution based mutation operation as done in case of parameters of the previous stage (described in Section 2.3.3).

As implemented in NSGA-II we use the crowded binary tournament selection. This step is followed by conventional crossover and mutation operations. The mutation operation is exactly the same as used in SOO. The most distinguishing characteristic of NSGA-II is the elitism operation, where we select the non-dominated solutions [4] from the set of parent and child populations, and propagate these to the subsequent generation. The solutions obtained in the last generation represent different solutions for the ensemble construction. These are actually the solutions that lie on the Pareto front or its near. Final solution is selected by the procedure described in Section 2.3.5.

2.5. Time complexity

Here we analyse the time complexity of the proposed algorithm. Note that in the current work we have used NSGA-II as the underlying optimization strategy for MOO based technique. The complexity of NSGA-II is $O(MN^2)$ where M: number of objectives and N: population size. Different steps of the proposed algorithm are having the following complexities:

- Initialization takes $O(N)$ time.
- For fitness computation, for each chromosome we need to perform training and testing for a particular classifier. Let the training time be $time_{train}$ and the test time be $time_{test}$. Hence the total time for fitness computation for a particular chromosome is $O(time_{train} + time_{test})$. Thus for a population of size N, total complexity is $O((time_{train} + time_{test}) \times N)$.
- Other operators also take $O(N)$ time.
- Thus the overall time complexity for MOO based feature selection and parameter optimization approach is $O((time_{train} + time_{test}) \times N + MN^2)$.

Finally the solutions obtained on the final Pareto optimal front are combined using a classifier ensemble technique. The classifier ensemble technique is having the following complexities:

- Initialization takes $O(N)$ time where N: population size.
- Fitness computation depends on the size of the data set. If the data set size is d then fitness computation takes $O(d)$ time for each chromosome. Thus for N number of chromosomes, the complexity is $O(d \times N)$.
- Other operators take $O(N)$ time.
- Thus the overall time for MOO based classifier ensemble approach is $O(d \times N + MN^2)$.

Finally, the total complexity of the joint model for chemical name detection and classification is $O((time_{train} + time_{test}) \times N + MN^2 + d \times N)$.

3. Features for chemical name identification

In this section we describe the features that we extracted for chemical name identification and classification. We formulate the overall task within the framework of supervised machine learning algorithm. Our focus is on recognizing IUPAC and IUPAC-like mentions of chemical names. In general, different nomenclatures are used for representing chemical entities, and in many cases these representations are combined by the chemists. Rather than narrowing our attention only to the correct IUPAC terms, the definition is broadened to include chemical substances represented in a IUPAC-like manner. Additionally it also includes IUPAC names in which a part is abbreviated or fragmented, or denotes a group name. The task is cast as a sequential labeling problem, and we use two popular machine learning techniques, namely CRF and SVM. The problem is to determine the most optimal class (or, tag) sequence $T = t_1, t_2, t_3, \ldots, t_i, \ldots, t_j$ for a sequence of words $W = w_1, w_2, w_3, \ldots, w_i, \ldots, w_n$. Probabilistically, this can be modeled as equivalent to finding out $argmax_T P(T|W)$.

Performance of any classification technique depends upon the feature sets. Our feature set is mixed in nature that exploits morphology, syntax and contextual information along with the various statistics that are computed from the given datasets. In order to preserve the domain-independence property we did not make use of heavy domain-specific resources and/or tools. We extract few features from the external resource like PubChem database. For example, the prefix and suffix character sequences extracted from this database serve as the domain dependent features. The features are described as below.

(1) **Context words**: We use the local contextual information surrounding the current token as the features. We use the contexts within the preceding three and succeeding three tokens.
(2) **Word prefix and suffix**: These are the fixed length character strings stripped either from the beginning (for prefix) and end positions (for suffix) of the words. We experiment with $n = 3$ (i.e., 6 features) and 4 (i.e., 8 features) both.
(3) **Infrequent word**: The words that appear frequently in the training set have a tendency of not being chemical name. Here we define a threshold value equal to 10 to decide whether the target word is a possible candidate of chemical name or not.
(4) **Unknown token feature**: This is defined in such a way that sets a feature value to 1 if the word appears in the training set, and 0 otherwise. For the training set the value of this feature is assigned randomly.
(5) **Word normalization**: Word normalization feature is defined to capture how a target word is orthographically constructed. The mappings are defined as follows: Capitalized character to 'A', small character to 'a', and all the consecutive digits to '0'. As examples, 'IL' is mapped to 'AA'; 'IL-2' is mapped to 'AA-0'; and 'IL-88' is mapped to 'AA-0'. The names having the similar structures are mapped to the same group that denotes a particular chemical class.
(6) **Orthographic features**: We define several orthographic features depending upon the contents of the wordforms. The features check whether *initial letter of the word is capitalized*, *all the letters of word are capitalized*, *word contains a capital letter inside*, *initial letter is capital; then a mixture of small and capital letters*, *word consists of digits only*, *word contains digit with special character*, *word starts with digit and then a sequence of alphabetic characters*, *word contains digit inside*, etc. Often the chemical names contain special characters like (',','-','.',')', '(', etc. As an example, '-' (hyphen) often appears inside the chemical names. Features that check the presence of ATGC sequence and stop words are also defined. We have in total 24 orthographic features.
(7) **Informative words**: The frequently occurring words that precede and follow the chemical names provide useful evidence for identifying the chemical names. We extract such frequently occurring words that appear within the contexts of previous two and next two tokens of the chemical names in the training set. After removing the stop words we create two different lists, each entity of which is considered to be informative. Two features are defined in such a way that they fire whenever the current token appears in any of these two lists.
(8) **Chemical prefix and suffix**: We extract the most frequently occurring prefixes and suffixes of length two from the IUPAC entities present in the training data. Thereafter two binary valued features are defined that fire if only if the current token contains any of these prefixes and suffixes.
(9) **PubChem prefix and suffix**: We extract most frequently occurring prefixes and suffixes of length two from the IUPAC chemical names of the PubChem database.[5] A binary valued feature is then defined that fires if and only if any of these inflections matches with the character sequences stripped either from the beginning or from the end positions of words.
(10) **Dynamic NE information**: This is the output label(s) of the previous token(s). The value of this feature is determined dynamically at run time. This feature is used for SVM.

Descriptions of the set of features are shown in Table 1.

4. Datasets, experiments and analysis

In this section we present the details of datasets, report the results of the different experiments with necessary analysis, and provide the comparisons with the existing systems.

4.1. Data sets

Chemical names in text can appear in various forms. One standardized nomenclature comes from the International Union of Pure and Applied Chemistry (IUPAC) and forms a systematic way of naming organic chemical compounds that can be mapped to their structures.

We use the datasets available at this web.[6] Brief statistics of the datasets are reported in Table 2. The training set was collected from the Medline abstracts. We experiment with two different test sets, one from the Medline abstracts and the other is from a collection of patent documents. The dataset for patent was labeled with seven classes as follows:

TRIVIAL: It denotes the single word terms like aspirin, estragon, testosterone, Acetylsalicylate, etc.
IUPAC: It denotes the multiword systematic names. For example, 1-hexoxy-4-methyl-hexane, 1,4-dihydronaphthoquinones, etc.
PART: It denotes the partial chemical names. Some of the examples are 8-(methylthio)-and . . . , 17beta-, etc.
MODIFIER: It denotes the words that modify the chemical names.

[5] http://pubchem.ncbi.nlm.nih.gov/.
[6] http://www.scai.fraunhofer.de/chem-corpora.html.

Table 1
Description of features.

Name of the feature	Explanation
allCapital	All characters are in capital letters
initialCapital	Initial character is Capital or not
capitalInner	Inner characters are capital or not
initialCapitalThenMix	First character is capital and next characters are mixed type (allowed characters)
allDigit	All characters are digits or not
realNumber	Word is real number or not
digitWithSpecialCharacter	Word contains special characters along with digit or not
initialDigitThenAlpha	Word with first digit character followed by alphabets or not
digitInner	Inner characters of a word are digit or not
specialChar	Word contains the special characters or not
twoBegConsecutiveWordMatch	Matching two consecutive words with the beginning two tokens of a multiword name
twoEndConsecutiveWordMatch	Matching two consecutive words with the last two tokens of a multiword name
stopWordMatch	Matching word with the stopword list
wordMatchFirst	Matching word to the first token of a chemical entity
wordMatchLast	Matching word to the last token of chemical entity
wordMatchVerb	Matching word with the possible list of verbs
wordNormalization	Normalizing surface form of words
romanNumber	Word is a representation of a Roman number
GreekNumber	Word is a Greek number representation
digitCommaDigit	Digit, digit is a substring of the word
singleCapital	Word contains only one capital letter
digitAlphaDigit	Initial letter is digit, intermediate characters are alphabets and the last character is again a digit
alphaDigitAlpha	Word starts and ends with alphabet and intermediate characters are all digits
wordPreviouslyOccured	Word previously occurred in the training data or not
initialSmallThenMix	Word starting with small letter and then followed by mixed (capital or small) letters
initialCapitalThenSmall	Word starting with capital letter and followed by small letters
initialAlphaThenDigit	Word starting with alphabet followed by digits
initialCapitalsThenDigit	Word with a sequence of capital letters followed by digits
IsDash	Word is a dash e.g.: [- – —]
IsSlash	Word is a slash e.g.: [/]
IsQuote	Word is a quote e.g.: [‘,]
Autom.Prefixes	Matching a prefix of a token against the list of prefixes extracted from chemical names
Autom.Suffixes	Matching a suffix of a token against the list of suffixes extracted from chemical names
Spaces_left	Feature indicating if a white space is preceding the token
Spaces_right	Feature indicating if a white space is following the token
Prefix_list	Matching a prefix (length 2) of a token against the list of prefixes of intermediate or first token of IUPAC names from pubChem
Suffix_list	Matching a suffix (length 2) of a token against the list of suffixes of intermediate or last token of IUPAC names from PubChem
RelImp_prefix_list	Matching a word against a list of words which are the beginning tokens of chemical names
RelImp_suffix_list	Matching a word against a list of words which are the ending tokens of chemical names
Context features	We have considered various contexts within the window size of $[-3, +3]$
Bigram feature	Bigram feature template for CRF
Pre_i	Prefixes of length up to i characters
Suf_i	Suffixes of length up to i characters
Word length feature	Length of the word is considered
Infrequent word	Frequency of occurrences of the word in the training data is considered
Informative word	Denotes the set of words that appear more frequently in the surrounding context of chemical name
sequenceATGC	Feature that checks the presence of ATGC sequence

Table 2
Statistics of the datasets.

Set	#abstracts	#sentences	#tokens	#IUPAC names
Training	463	3700	161,591	3712
Test set (Medline)	1000	5305	124,122	151
Test set (Patent, 2008)	26	152	4309	411
Test set (Patent, 2009)	27	160	4417	471

ABBREVIATION: It denotes the abbreviation but does not include those appearing as part of IUPAC names such as TPA and AMPA.
SUM: It denotes the sum formulas like CH_3SNa, KOH, etc.
FAMILY: This class represents the families of chemical names such as disaccharide, pyrimidine and hydrazides. But this does not include the pharmacological/functional families such as anti-inflammatory drug and chelator.

However the training set was labeled with all these classes, test set does not have instances of all such classes. The test set was annotated with only three classes, namely IUPAC, PART and MODIFIER. The test set of Medline contains only the instances of IUPAC and MODIFIER classes. In order to properly denote the boundaries of multiword chemical names, all the classes are further divided using the BIO notation, where ‘B-XXX’ refers to the beginning of a multi-word/single-word name of type ‘XXX’, ‘I-XXX’ refers to the intermediate parts of the name and ‘O’ refers to the entities outside the name. Here for each token all the features mentioned in Section 3 are extracted. Features are added as tab-separated in a single file. The last column contains the output class. Examples of data sets are given below:

(0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 _ 0 0 0 0 0 0 (ND ND ND (ND ND ND (0 0 0
0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 B-IUPAC
propargyloxy 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 aaaaaaaaaaaa 0 0 0 0 0 0 p pr pro
prop y xy oxy loxy a 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 I-IUPAC
) 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 _ 0 0 0 0 0 0) ND ND ND) ND ND ND) 0 0 1
0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 I-IUPAC
methyl 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 aaaaaa 0 0 0 0 0 0 m me met meth l yl
hyl thyl a 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 I-IUPAC
acyclonucleoside 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 aaaaaaaaaaaaaaa 0 0 0 0 0 0 a
ac acy acyc e de ide side a 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 I-IUPAC

4.2. Experimental setup

For the experiments we use the C++ based CRF++ package,[7] a simple, customizable, and open source implementation of CRF for segmenting or labeling sequential data. For SVM experiments we use YamCha[8] toolkit along with TinySVM-0.07.[9] Here, the *pairwise multi-class decision* method and the *polynomial kernel function* are used. In YamCha, only polynomial kernel function is available. But in order to automatically optimize the other parameters of SVM like kernel type, degree of kernel function, coefficient of kernel function, gamma in kernel function, epsilon parameter we use LibSVM implementation as available in the Weka tool box.[10] We set the following parameter values for GA and NSGA-II: population size = 100, number of generations = 50, probability of mutation = 0.2 and probability of crossover = 0.9.

In this work we use all the features described in Section 3, and optimize two parameters of CRF and five parameters of SVM.

Initially, we construct the following six *baseline* models based on CRF and SVM. Here, $C[-i,+j]$ denotes the context spanning from the previous ith word to the next jth word with the current token at position 0; Pre_i and Suf_i denote the prefixes and suffixes of character sequences up to i of the current word, respectively.

(1) **Baseline 1**: CRF classifier trained using the default parameter values with the feature combinations: $C[-3,+3]$, Pre_4, Suf_4, and all the features described in Section 3.
(2) **Baseline 2**: CRF classifier trained using the default parameter values with the feature combinations: $C[-2,+2], Pre_4, Suf_4$, and all other features described in Section 3.
(3) **Baseline 3**: SVM classifier trained using the default parameter values with the feature combinations: $C[-3,+3], Pre_4$, Suf_4, and all other features described in Section 3.
(4) **Baseline 4**: SVM classifier trained using default parameter values with the feature combinations: $C[-2,+2], Pre_4, Suf_4$, and all other features described in Section 3.
(5) **Baseline 5**: All the classifiers of the final Pareto optimal front generated after application of the proposed feature and parameter selection technique on CRF and SVM are combined using majority voting. Suppose, there are total M number of classifiers. Now, for the ensemble classifier the output label for each token is determined using the majority voting of these M classifiers' outputs. The combined score of a particular class(c_i) for a particular word w is $f(c_i) = n$ where n is the number of classifiers that assign the output class c_i for the word w. The class receiving the maximum weight is selected as the final decision.
(6) **Baseline 6**: All the CRF and SVM based classifiers of the final Pareto optimal front generated after application of the proposed feature and parameter selection technique are combined using weighted voting. Suppose, there are M classifiers. Let, the overall F-measure values of these M classifiers for the development set be $F_i, i = 1,\ldots,M$, respectively. For the ensemble classifier the output label for each token is determined using the weighted voting of these M classifiers' outputs. The combined score of a particular class for a particular word w is:

$$f(c_i) = \sum F_m,$$
$$\forall m = 1 : M \;\&\; op(w,m) = c_i$$

Here, $op(w,m)$ denotes the output class provided by the mth classifier for the word w. The final class label is decided based on the highest weight.

4.3. Results and analysis

At first we apply SOO and MOO based feature selection and parameter optimization techniques (Section 2.3) to solve the problem of chemical mention detection and classification with respect to two different classifiers, CRF and SVM. In the first step a CRF is trained using the training dataset of Medline and evaluation is done on three independent test sets. The recall, precision and F-measure values obtained by the proposed SOO and MOO based techniques are shown in Table 3 for the chemical test datasets prepared in 2008, 2009; and for the Medline. After application of the proposed SOO based approach on a chemical test dataset prepared in 2008 we obtain a F-measure of 72.69% (recall = 62.12% and precision = 87.59%) and after applying the same technique to a annotated chemical test dataset prepared in 2009, the F-measure obtained is 72.65% (recall = 62.11% and precision = 87.50%). Whereas with the test dataset of Medline our system is able to achieve the F-measure of 87.10% (recall = 88.12% and precision = 86.10%). The MOO based approach of feature and parameter selection produces the F-measures of 73.78% (recall = 63.00% and precision = 88.99%) and 74.05% (recall = 63.51% and precision = 88.78%) for the datasets of 2008 and 2009, respectively. The same MOO based system shows the recall, precision and F-measure values of 89.90%, 87.76% and 88.82%, respectively, for the Medline data set. The significant drop in performance for the patent datasets is because of its inherent characteristics. Instead of sampling from a set of sentences and text snippets, this particular dataset was created by selecting the tokens which are hard to identify.

The analysis of different datasets shows two main problems: Only IUPAC, PARTIUPAC and MODIFIER names are included in the training dataset, but in both the test datasets fragments (of chemical names) occurred frequently. Another problem found with the training dataset is that it does not have any instance of some of the categories, which are present in the test data. All such

[7] http://crfpp.sourceforge.net.
[8] http://chasen-org/taku/software/yamcha/.
[9] http://cl.aist-nara.ac.jp/taku-ku/software/TinySVM.
[10] http://weka.wikispaces.com/LibSVMdeal.

Table 3
Overall results of the proposed system on three different test datasets.

Model	Test Corpus 2008			Test Corpus of 2009			Medline		
	Recall	Precision	*F*-measure	Recall	Precision	*F*-measure	Recall	Precision	*F*-measure
SOO + CRF based approach	62.12	87.59	72.69	62.11	87.50	72.65	88.12	86.10	87.10
SOO + SVM based approach (optimizing degree of kernel only)	61.99	87.91	72.71	61.61	86.41	71.93	89.01	82.15	85.44
SOO + SVM (optimizing five parameters)	62.22	87.94	72.87	61.81	86.56	72.12	89.71	82.61	86.01
SOO based ensemble [CRF + SVM]	63.45	88.90	74.05	63.52	88.61	74.00	89.91	87.11	88.49
MOO + CRF based approach	63.00	88.99	73.78	63.51	88.78	74.05	89.90	87.76	88.82
MOO + SVM based approach (optimizing degree of kernel only)	62.97	88.99	73.75	62.43	87.50	72.87	90.62	83.08	86.69
MOO + SVM (optimizing five parameters)	62.99	89.01	73.77	62.61	87.75	73.08	90.81	83.28	86.88
MOO based ensemble[CRF + SVM]	65.10	90.01	75.55	65.71	90.32	76.07	90.90	88.66	89.77
Baseline-1	61.41	86.87	71.95	62.01	86.18	72.12	88.26	85.77	87.00
Baseline-2	61.55	86.96	72.08	62.25	86.41	72.30	88.02	85.38	86.68
Baseline-3	61.01	86.17	71.44	61.87	88.26	72.74	88.19	81.63	84.78
Baseline-4	61.25	86.60	71.75	61.95	88.90	73.02	88.22	81.09	84.50
Baseline-5	63.90	89.56	74.58	64.90	89.79	75.34	90.03	87.94	88.97
Baseline-6	63.99	89.78	74.72	64.95	89.97	75.44	90.13	87.99	89.05

examples that appear in the test dataset are not correctly predicted because the training data does not contain any such class instances. In fact it uses only the classes learned from the training set and as a result performance of the MOO based system reaches to 74.05% for the chemical test dataset. But the Medline test data has only two classes, namely IUPAC and MODIFIER, and these are also present in the training data. Because of this, the proposed system attains reasonably high accuracy on this particular dataset of Medline.

Similarly the proposed SOO and MOO based approaches are applied for solving the problem with respect to SVM. The best solution is selected based on the *F*-measure value. The corresponding recall, precision and *F*-measure are reported in Table 3. Results show that, for all the three test datasets, CRF performs superior compared to SVM.

The SOO based feature and parameter selection technique, when applied on CRF and SVM, produces a set of promising solutions including the best one on the final population. Based on the features and parameters represented by the chromosomes corresponding to these solutions we generate a set of CRF and SVM based models. It is to be noted that we have two different populations representing the solutions obtained from CRF and SVM. Each of the chromosomes in a population represents a particular feature and parameter combination for a classifier (either CRF or SVM). The solutions that exist on a particular population do not conform to the uniform characteristics. In order to further improve the performance we combine all the CRF and SVM based models using a SOO based classifier ensemble technique [6]. Overall evaluation results along with the *baseline* models are reported in Table 3 for all the three datasets. The SOO based feature and parameter selection technique alone achieves the encouraging performance for all the three test corpora. Final SOO based ensemble yields the overall recall, precision and *F*-measure values of 63.45%, 88.90% and 74.05%, respectively for the test corpus of 2008; 63.52%, 88.61% and 74.00%, respectively for the test corpus of 2009; and 89.91%, 87.11% and 88.49%, respectively for the Medline test corpus.

After application of the MOO based feature selection and parameter optimization technique for the CRF based classifier we obtain a set of Pareto optimal solutions. Based on the features and parameters of these solutions we generate a set of CRF models. Some of them are good with respect to recall and some are good with respect to precision. Similarly after executing the MOO based approach for the SVM based classifier we obtain another set of solutions on the final Pareto front. These solutions represent different feature and parameter combinations. By using these features and parameters, we can again generate several SVM models. We take the union of these CRF and SVM based models, and combine their outputs using a MOO based classifier ensemble technique [24]. Overall evaluation results along with the *baseline* models are reported in Table 3 for the chemical test corpus prepared in 2008, 2009, and Medline. Evaluation shows that we can achieve reasonable performance for all the three test data sets. The final output of our proposed approach, which employs MOO based ensemble, yields the overall recall, precision and *F*-measure values of 65.10%, 90.01% and 75.55%, respectively for the test corpus of 2008; 65.71%, 90.32% and 76.07%, respectively for the test corpus of 2009; and 90.90%, 88.66% and 89.77%, respectively for the Medline test corpus.

Results also show that the proposed MOO based technique is more effective than the systems developed using all the features and default parameter settings. For the test corpus of 2009, MOO along with CRF attains 1.93% and 1.75% performance improvements over the first two baselines, respectively. Again MOO with SVM based approach attains the performance improvements of 0.24% and 0.06% *F*-measure points over the third and fourth baselines, respectively. These baselines use all the features and default parameter values to generate SVM based models. Experiments of these different settings prove the efficiency of the MOO based automatic feature selection and parameter optimization technique over the manual feature and parameter selection technique. It is also worthy to combine the outputs of the classifiers on the final Pareto front (obtained in the first stage) using a MOO based ensemble technique. It is evident from the final evaluation figures reported in Table 3 that MOO based ensemble achieves the performance improvement of 2.02% *F*-measure over the best individual classifier (here it is CRF based feature and parameter selection approach) for the 2009 test corpus. This MOO based ensemble also performs better than the two existing classical ensemble techniques, namely the fifth and sixth baselines. We achieve the increments of 1.41% and 1.31% *F*-measure values, respectively, over these two models. Evaluation also suggests that MOO suits more compared to SOO. For 2009 patent test data it achieves 2.07% *F*-measure improvement over the SOO based approach.

For Medline test data we also observe quite similar patterns in the evaluation results. Here CRF provides better performance than SVM. The MOO based approach along with CRF performs better than the baselines, which use all the features and default parameter values. Table 3 shows that the proposed approach attains the performance increments of 1.82% and 2.14% *F*-measure values, respectively over the first two baselines. Similarly MOO along with SVM based approach attains the performance increments of 2.10% and 2.38% *F*-measure values over the third and fourth baselines,

respectively. Highest accuracies are obtained when MOO based ensemble is employed at the second stage of our algorithm. It achieves the increments of 0.95%, 0.80% and 0.72% *F*-measure values over the CRF based feature and parameter selection method, fifth baseline and sixth baseline, respectively. Similar kind of conclusions can also be drawn for the 2008 test dataset.

For illustration, we have also plotted the set of solutions (in terms of recall, precision and *F*-measure values) obtained by the proposed MOO based feature and parameter selection technique for three different data sets when executed with the CRF based classifier. These are shown in Fig. 6(a)–(c), respectively. These plots show that for all the cases we obtain multiple solutions; some of these are better with respect to recall whereas some are better with respect to precision. Thus the proposed MOO based technique indeed provides a variety of trade-off solutions. Depending on the user preference or application domain any single solution is finally selected as the optimal one. Similarly we also present the boxplots of the *F*-measure values obtained by the solutions on the best population obtained after the application of SOO based technique when executed on three data sets with CRF as the base classifier. These plots are shown in Fig. 7(a)–(c), respectively.

Statistical analysis of variance (ANOVA), is performed in order to examine whether the MOO based feature and parameter selection technique really outperforms the best individual classifier, six *baseline* ensembles and the corresponding single objective GA based approaches. Here, all the classifiers and the proposed ensemble techniques are executed 10 times. Thereafter, ANOVA analysis is carried out on these outputs. Evaluation results of the ANOVA analyses are shown in Table 4 for the MOO based feature selection and parameter optimization approach for the Medline data set. ANOVA tests show that the differences in mean recall, precision and *F*-measure are statistically significant as p value is less than 0.05 in each case. Results also reveal that MOO based techniques truly perform better than the corresponding single objective GA based techniques.

We present the features and parameters selected by the proposed MOO based approach for the CRF classifier in Table 6. This shows that only a small set of features is actually relevant. For example, out of 45 features only 17 are selected for the test dataset of Patent 2003. In the baselines 1–4 we utilize all the 45 features. This is the another important aspect of our proposed approach which proves the necessity of feature selection in the chemical domain. Similarly the feature and parameter combinations selected by the MOO based approach for the SVM based classifier are shown in Table 7. It is again evident, that with only a limited number of features, we can achieve improved performance.

In another experiment, we removed the unknown classes (categories which are present in the test data, but not available in training, for example: SUM, FAMILY, ABBREVIATION, TRIVIAL, etc.) from the training data. With this setting, the MOO based

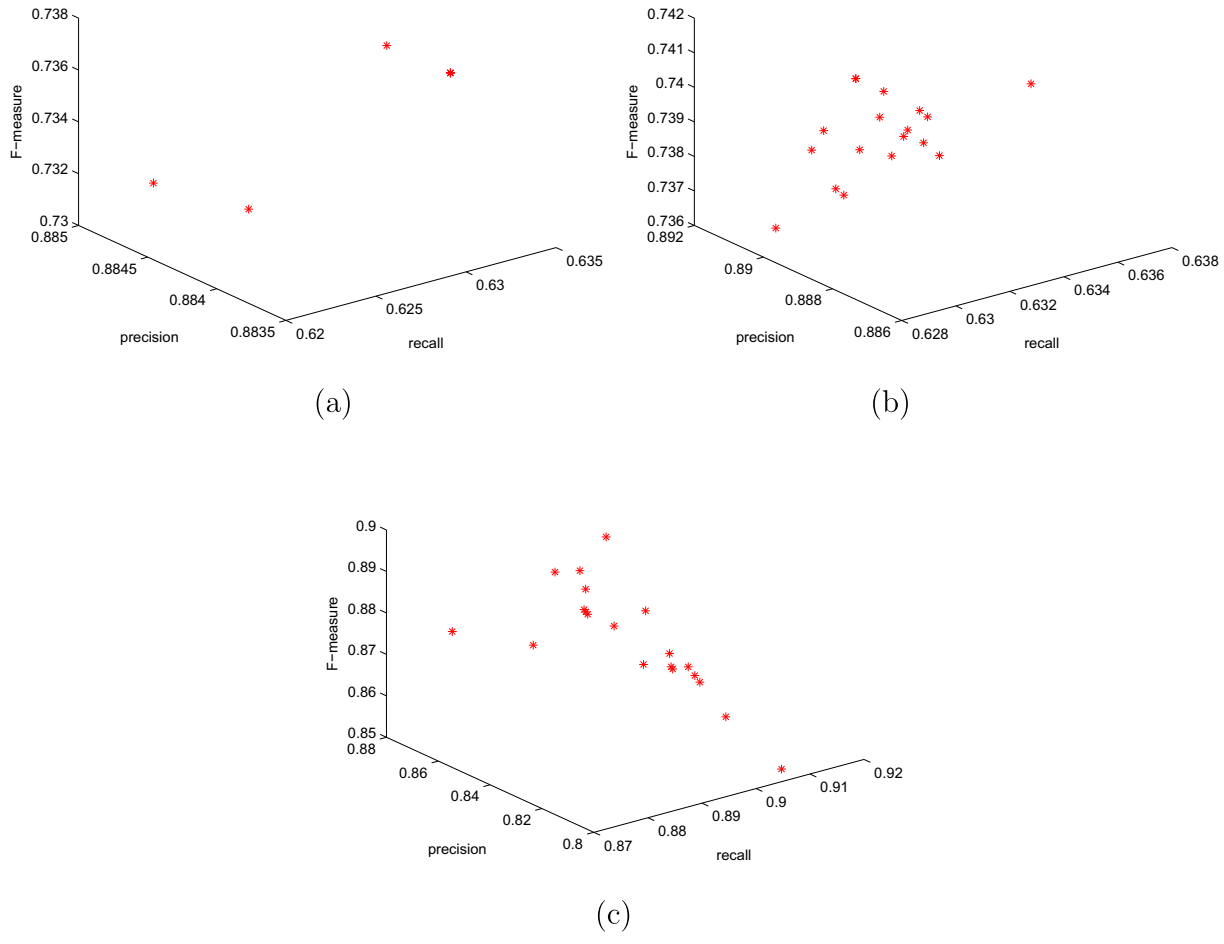

Fig. 6. Set of solutions obtained by the proposed MOO based feature and parameter selection technique with CRF as the base classifier for (a) Test Corpus 2008, (b) Test Corpus 2009 and (c) Medline test corpus.

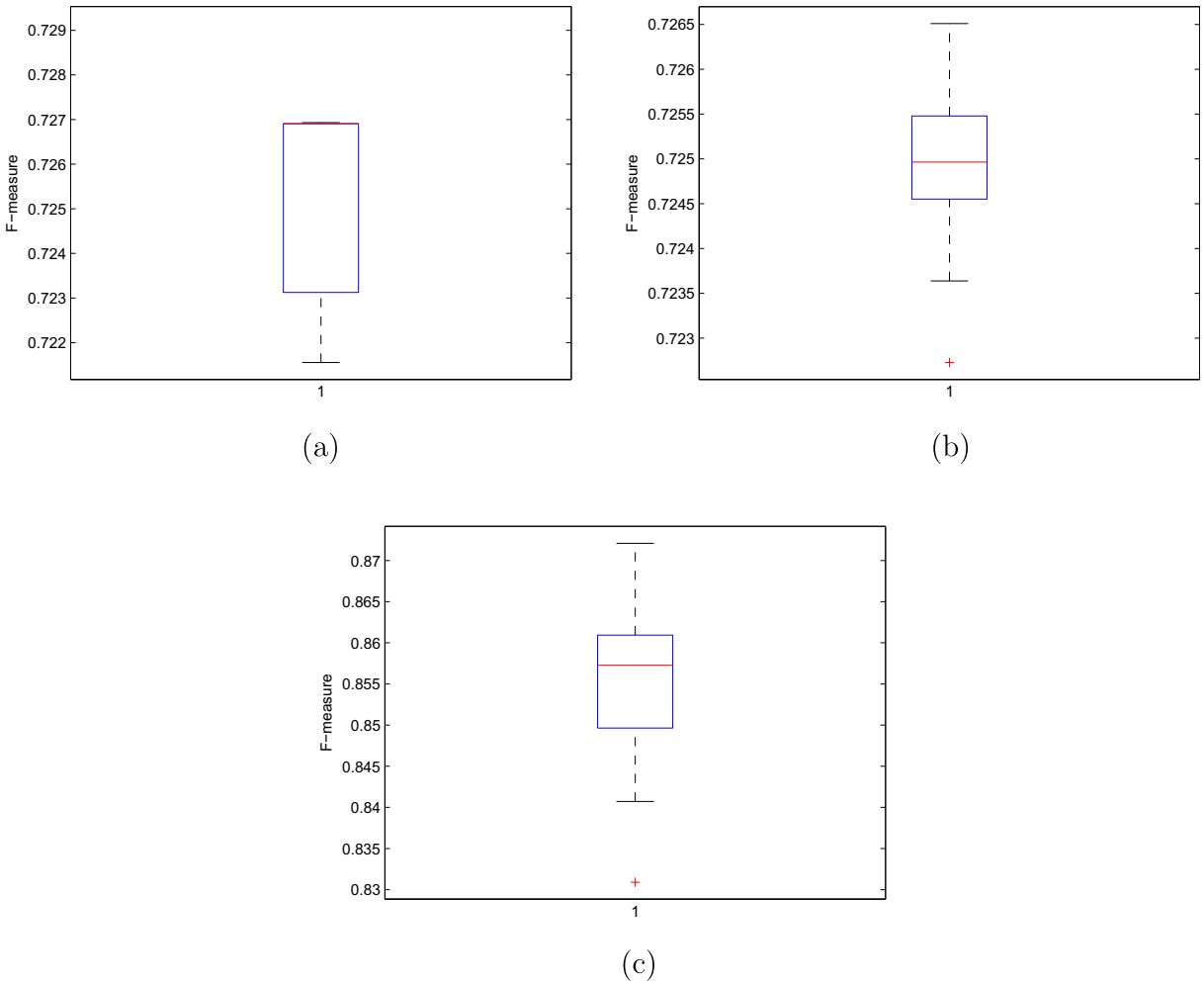

Fig. 7. Boxplots of the *F*-measure values of the solutions on the best population of the proposed SOO based feature and parameter selection technique for (a) Test Corpus 2008, (b) Test Corpus 2009 and (c) Medline test corpus.

Table 4
Estimated marginal means and pairwise comparison between the proposed MOO based feature and parameter selection technique and several other ensembles for Medline test data.

Evaluation criterion	Technique (I)	Comp.	Mean Diff. (I–J)	Significance value
F-measure	*MOO* based ensemble	MOO + CRF	0.95 ± 0.013	$1.1623e-009$
F-measure	*MOO* based ensemble	MOO + SVM	3.08 ± 0.011	$2.7623e-008$
F-measure	*MOO* based ensemble	GA based ensemble	1.28 ± 0.012	$2.2356e-010$
F-measure	*MOO* based ensemble	*Baseline 1*	2.77 ± 0.014	$3.3990e-009$
F-measure	*MOO* based ensemble	*Baseline 2*	3.09 ± 0.011	$8.6386e-010$
F-measure	*MOO* based ensemble	*Baseline 3*	4.99 ± 0.014	$5.5376e-010$
F-measure	*MOO* based ensemble	*Baseline 4*	5.27 ± 0.013	$2.7126e-010$
F-measure	*MOO* based ensemble	*Baseline 5*	0.80 ± 0.009	$4.5176e-010$
F-measure	*MOO* based ensemble	*Baseline 6*	0.72 ± 0.008	$3.8326e-010$

technique attains the recall, precision and *F*-measure values of 80.76%, 90.97% and 85.55%, respectively for the patent 2008 dataset with CRF. For 2009 data set with the same configuration we attain the recall, precision and *F*-measure values of 81.22%, 90.92% and 85.79%, respectively. On the other hand the same technique with SVM achieves the recall, precision and *F*-measure values of 79.95%, 86.76% and 83.22%, respectively for the 2008 patent test data set. For 2008 dataset the method shows the recall, precision and *F*-measure values of 79.35%, 85.52% and 82.32%, respectively. Finally the outputs of all the CRF and SVM classifiers present on the final Pareto optimal front are combined by the MOO based ensemble technique. This yields the recall, precision and *F*-measure values of 82.76%, 91.34% and 86.84%, respectively.

Table 5
Comparison with the existing approaches (we report percentages).

System	Used approach	Data set	*F*-measure (%)
Klinger et al. [17]	CRF	Medline	85.6
Grego et al. [9]	CRF	2008 patent data	50.73
		2009 patent data	50.63
		Medline	63.20
OSCAR3 [2]		2008 patent data	35.34
		2009 patent data	35.93
		Medline	36.02
Proposed approach	CRF, SVM	2008 patent data	75.55
		2009 patent data	76.07
		Medline	89.77

Table 6
Features and parameters identified by the proposed MOO based approach for CRF classifier. Here $C[i,j]$: content spacing from the ith position to the jth position with 0 as the current one.

Language	Features	Parameters
Test corpus of 2008	$C[-2,+2], Suf_2, Pre_4$, initialAlphaThenDigit, wordPreviouslyOccured, singleCapital, digitCommaDigit, wordNormalization, wordMatchVerb, wordMatchFirst, stopWordMatch, twoEndConsecutiveWordMatch, initialCapitalThenMix, allCapital, initialCapital, isSlash, RelImp_prefix_list, Bigram feature	$c = 2.184$, $f = 9$
Test corpus of 2009	$C[-2,+2], Pre_4, Suf_1$, initialCapitalsThenDigit, initialAlphaThenDigit, wordNormalization, digitAlphaDigit, GreekNumber, romanNumber, wordMatchLast, stopWordMatch, twoBegConsecutiveWordMatch, twoEndConsecutiveWordMatch, digitInner, capitalInner, allCapital, initialCapital, isDash, isQuote, isSlash, RelImp_prefix_list, RelImp_suffix_list, Informative word, Bigram feature	$c = 2.22$, $f = 3$
Medline Test Corpus	$C[-2,+2], Pre_1, Suf_4$, initialCapitalsThenDigit, initialAlphaThenDigit, initialCapitalThenSmall, wordPreviouslyOccured, alphaDigitAlpha, digitAlphaDigit, digitCommaDigit, GreekNumber, romanNumber, wordMatchVerb, wordMatchLast, stopWordMatch, digitInner, digitWithSpecialChar, realNumber, allDigit, initialCapitalThenMix, allCapital, initialCapital, isDash, isQuote, Autom.Prefixes, RelImp_prefix_list, Informative word, Bigram feature	$c = 1.186$, $f = 1$

Table 7
Features and parameters identified by the proposed MOO based approach for SVM classifier. Here $C[i,j]$: content spacing from the ith position to the jth position with 0 as the current one.

Language	Features	Parameters
Test corpus of 2008	$C[-2,+2], Suf_2, Pre_4$, initialAlphaThenDigit, wordPreviouslyOccured, singleCapital, digitCommaDigit, romanNumber, wordNormalization, wordMatchVerb, wordatchFirst, stopWordMatch, twoEndConsecutiveWordMatch, twoBegConsecutiveWordMatch, digitInner, initialCapitalThenMix, capitalInner, Autom.Prefixes, Autom.Suffixes RelImp_prefix_list, Dynamic NE feature,	Polynomial kernel, deg = 2, epsilon = 0.1, gamma = 1/21, coeff = 0
Test corpus of 2009	$C[-2,+2], Pre_3, Suf_1$, initialCapitalsThenDigit, initialAlphaThenDigit, initialCapitalThenSmall, initialSmallThenMix, alphaDigitAlpha, digitAlphaDigit, singleCapital, digitCommaDigit, GreekNumber, romanNumber, wordMatchVerb, wordMatchLast, stopWordMatch, twoBegConsecutiveWordMatch, digitInner, initialDigitThenAlpha, digitWithSpecialChar, sequenceATGC, Autom.Suffixes, Informative word, Dynamic NE feature	Polynomial kernel, deg = 3, gamma = 1/24, coeff = 0, epsilon = 0.2
Medline Test Corpus	$C[-2,+2], Pre_1, Suf_4$, initialCapitalsThenDigit, initialSmallThenMix, initialCapitalThenSmall, wordPreviouslyOccured, alphaDigitAlpha, digitAlphaDigit, digitCommaDigit, GreekNumber, romanNumber, WordMatchVerb, stopWordMatch, twoEndConsecutiveWordMatch, twoBegConsecutiveWordMatch, allDigit, initialCapitalThenMix, capitalInner, singleCapital, initialCapital, isDash, Autom.Prefixes, digitInner, Autom.Suffixes, RelImp_prefix_list, Informative word, Dynamic NE feature	Polynomial kernel, deg = 2, epsilon = 0.1, gamma = 1/28, coeffi = 0

4.4. Comparison with the existing systems

We compare the performance of our proposed approach with the other existing techniques. We present the comparative evaluation results in Table 5.

The state-of-the-art system proposed in [17] is developed for entity extraction in chemical domain. The evaluation with different orders and offset conjunctions of CRF demonstrates the importance of these parameters. The classifier was trained with the following set of features. It made use of various binary valued features that check whether all the characters are capitalized; token is a real number, a dash, a quote, a slash; whether there are spaces to the left or right. It also incorporates the features based on the prefix and/or suffix strings, bag-of-word feature, etc. Features were also extracted from the prefix and suffix lists that were generated from the IUPAC names mentioned in the data available from PubChem. The lists of prefixes and suffixes of four length character sequences were used that contain 714 and 661 entries, respectively. A third list which includes 300 suffixes extracted from the last tokens of IUPAC names was also used to improve the detection of IUPAC names. The motivation of using these three lists is to provide the system with a possibility to generalize in excess of the training data. The parameters of the CRF based classifier are selected by applying 30-fold bootstrapping on the training set. System proposed in [17] achieved recall, precision and F-measure values of 86.5%, 84.8% and 85.6%, respectively on the Medline test dataset.

Our system (two-stage MOO based approach) achieves an increment of 4.17% F-measure for the Medline test dataset. This improvement is due to the use of a rich feature set, use of a systematic approach of feature selection and parameter optimization of classifiers using MOO, and finally for employing the ensemble. Note that our baseline results are better than the results obtained in [17]. This may be due to the fact that the baselines 1–4 use all the available features, and these were proved to be helpful for the identification and classification of chemical entities. In baselines 5 and 6 we utilize the MOO based feature and parameter selection technique to generate a set of solutions. Finally, instead of using ensemble technique proposed in [24], we used some traditional ensemble techniques. As these two baseline models utilize some of our proposed resources and/or techniques, the performance gains do not look very convincing. However the ensemble output seems to be satisfactory if we compare the evaluation figures with the other baselines.

We also compare our approach with the existing techniques proposed in [9,2,16]. Note that in [9] a CRF based machine learning system is developed for the chemical name identification. This paper addressed only the issue of identification, and no classification was performed. In contrast in our work we perform identification as well as classification. In the identification phase we detect the chemical names from the text, and in classification phase we classify the entities into some predefined categories of interest. In our work we perform these two tasks simultaneously, and it is

more complex than the mere identification task. The CRF based system proposed in [9] utilizes only a set of few features like stem/root word, prefixes, suffixes and digit features. We executed this CRF based approach with only these features on the three test data sets used in our experiments. For 2008 patent data set the system proposed in [9] attains the recall, precision and *F*-measure values of 47.06%, 55.01% and 50.73%, respectively. Similarly for 2009 patent dataset the same system attains the recall, precision and *F*-measure values of 46.94%, 54.95% and 50.63%, respectively. For Medline test data, the same system shows the recall, precision and *F*-measure values of 66.38%, 60.32% and 63.20%, respectively.

We also executed the OSCAR3 system developed in [2], the version of which is downloaded from the site.[11] This is a chemical entity recognition system which identifies chemical names from text and classifies them into the following categories: CM (Chemical): A chemical compound/class of compounds; RN (Reaction): A chemical verb or nominalisation thereof, e.g. demethylation; CJ (Chemical adjective): A chemical in adjective form e.g. benzylic; CPR (Chemical prefix): e.g. 1,2- in 1,2-transposition; ASE (Single word enzyme names-from-chemicals): e.g. peroxidase; PRW: Potential reaction word; ONT: A term that does not fit into any other category.

These chemical entities are not same as in our case. So for fair comparison we execute the OSCAR3 system on our test datasets. We obtain the accuracies of 35.34%, 35.93% and 36.02%, respectively, for the patent test data set 2008, patent test data set 2009 and Medline test data set.

5. Conclusion

In this paper we have proposed a joint model of feature selection and parameter optimization within the frameworks of SOO and MOO for chemical name identification and classification. As the base learning algorithms, we used CRF and SVM. These classifiers were trained with a diverse set of features, most of which were generated without using any domain-specific external resources and/or tools. For CRF, we have optimized two parameter values, namely hyper-parameter and cut-off threshold of features. For SVM, we have optimized five parameter values, namely the kernel function, degree of kernel, gamma value of kernel function, coefficient value of kernel function and the epsilon parameter. The proposed SOO and MOO based approaches generate a set of solutions on the final best population and the final Pareto optimal front, respectively. In the second stage of our algorithm we employ ensemble learning algorithms to combine the outputs of all the solutions obtained in the first stage. In case of SOO the solutions on the final best population are combined using a SOO based classifier ensemble technique. In case of MOO the solutions of these final Pareto front are then combined using a MOO based classifier ensemble technique. The proposed systems are evaluated with three benchmark datasets. Overall performance of the SOO and MOO based approaches show the *F*-measure values of 74.05% and 75.55%, respectively for the 2008 patent dataset. The same methods exhibit 74.00% and 76.07% *F*-measures, respectively for the patent 2009 test data set. Evaluation also yields state-of-the-art performance with the overall *F*-measure values of 88.49% and 89.77%, respectively for SOO and MOO based approaches on Medline test data set. Our proposed method performs superior over the six baseline models, constructed with the various features, default parameters and classical ensemble techniques. Comparisons with the other existing state-of-the-art systems show the efficacy of our proposed models. We tried to preserve the system as much domain-independent as possible, and hence we limited ourselves to use only two domain-specific features that were extracted from an external resource. In future, we would like to include more domain dependent features. Future works also include the use of some other well known classifiers like decision tree and memory based learner. We also plan to evaluate our proposed approaches for the other domains.

[11] http://apidoc.ch.cam.ac.uk/oscar3/.

References

[1] Hisham Al-Mubaid, Hoa A. Nguyen, Using MEDLINE as standard corpus for measuring semantic similarity in the biomedical domain, in: Proceedings of the Sixth IEEE Symposium on BionInformatics and BioEngineering, BIBE '06, IEEE Computer Society, Washington, DC, USA, 2006, pp. 315–318.

[2] Peter Corbett, Peter Murray-Rust, High-throughput identification of chemistry in life science texts, in: Proceedings of the Second International Conference on Computational Life Sciences, CompLife'06, Springer-Verlag, Berlin, Heidelberg, Cambridge, UK, 2006, pp. 107–118.

[3] H. Cunningham, GATE, a general architecture for text engineering, Comput. Humanit. 36 (2002) 223–254.

[4] Kalyanmoy Deb, Multi-Objective Optimization Using Evolutionary Algorithms, John Wiley and Sons, Ltd., England, 2001.

[5] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 181–197.

[6] Asif Ekbal, Sriparna Saha, Weighted vote-based classifier ensemble for named entity recognition: a genetic algorithm-based approach, ACM Trans. Asian Lang. Inf. Process. 10 (2) (2011) 9.

[7] Asif Ekbal, Sriparna Saha, Simultaneous feature and parameter selection using multiobjective optimization: application to named entity recognition, Int. J. Mach. Learn. Cybern. (2014) 1–15, http://dx.doi.org/10.1007/s13042-014-0268-7.

[8] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, New York, 1989.

[9] Tiago Grego, Piotr Pezik, Francisco M. Couto, Dietrich Rebholz-Schuhmann, Identification of chemical entities in patent documents, in: Proceedings of the 10th International Work-Conference on Artificial Neural Networks: Part II: Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living, IWANN '09, Springer-Verlag, Berlin, Heidelberg, Salamanca, Spain, 2009, pp. 942–949.

[10] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten, The WEKA data mining software: an update, SIGKDD Explor. Newsl. 11 (1) (2009) 10–18.

[11] Md. Hasanuzzaman, Sriparna Saha, Asif Ekbal, Feature subset selection using genetic algorithm for named entity recognition, in: Ryo Otoguro, Kiyoshi Ishikawa, Hiroshi Umemoto, Kei Yoshimoto, Yasunari Harada (Eds.), PACLIC, Institute for Digital Enhancement of Cognitive Development, Waseda University, 2010, pp. 153–162.

[12] J.H. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, AnnArbor, 1975.

[13] Md Rahat Hossain, Amanullah Maung Than Oo, A.B.M. Shawkat Ali, The combined effect of applying feature selection and parameter optimization on machine learning techniques for solar power prediction, Am. J. Energy Res. 1 (1) (2013) 7–16.

[14] Cheng-Lung Huang, Jian-Fan Dun, A distributed PSO-SVM hybrid system with feature selection and parameter optimization, Appl. Soft Comput. 8 (4) (2008) 1381–1391.

[15] Cheng-Lung Huang, Chieh-Jen Wang, A GA-based feature selection and parameters optimization for support vector machines, Expert Syst. Appl. 31 (2) (2006) 231–240.

[16] David Jessop, Sam Adams, Egon Willighagen, Lezan Hawizy, Peter Murray-Rust, OSCAR4: a flexible architecture for chemical text-mining, J. Cheminformatics 3 (1) (2011) 41.

[17] Roman Klinger, Corinna Kolarik, Juliane Fluck, Martin Hofman-Apitius, Christoph M. Friedrich, Detection of IUPAC and IUPAC-like chemical names, Bioinformatics 24 (13) (2008) 268–276.

[18] John D. Lafferty, Andrew McCallum, Fernando C.N. Pereira, Conditional random fields: probabilistic models for segmenting and labeling sequence data, in: ICML, 2001, pp. 282–289.

[19] Kuan-Cheng Lin, Hsu-Yu Chien, CSO-based feature selection and parameter optimization for support vector machine, in: Proceedings of Joint Conferences on Pervasive Computing (JCPC), Taipei, 2009, pp. 783–788.

[20] Huan Liu, Hiroshi Motoda, Feature Selection for Knowledge Discovery and Data Mining, Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[21] Huan Liu, Lei Yu, Toward integrating feature selection algorithms for classification and clustering, IEEE Trans. Knowl. Data Eng. 17 (4) (2005) 491–502.

[22] M. Narayanaswamy et al., A biological named entity recognizer, in: Proceedings of the Pacific Symposium on Biocomputing, 2003, pp. 427–438.

[23] NCBI, Pubchem data, 2007.

[24] Sriparna Saha, Asif Ekbal, Combining multiple classifiers using vote based classifier ensemble technique for named entity recognition, Data Knowl. Eng. (0) (2012).

[25] Vladimir N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag New York, Inc., New York, NY, USA, 1995.