Sentiment Classification of Drug Reviews Using Fuzzy-rough Feature Selection

Tianhua Chen[a], Pan Su[b], Changjing Shang[c], Richard Hill[a], Hengshan Zhang[de] and Qiang Shen[c]

[a]Department of Computer Science, School of Computing and Engineering, University of Huddersfield, UK
[b]Cixi Institute of Biomedical Engineering, Chinese Academy of Sciences, China
[c]Department of Computer Science, Institute of Mathematics, Physics and Computer Science, Aberystwyth University, UK
[d]School of Computer Science, Xi'an University of Posts and Telecommunications, Xi'an, China
[e]Shanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'ian, China

Abstract—Sentiment analysis mines people's opinions and attitudes regarding a certain issue from source materials. Recently, it has drawn significant attention in a number of application areas. The sentiment analysis of healthcare in general and that of users' drug experience in particular could shed significant light on how to improve public health and make the right decisions. However, one of the major challenges in sentiment classification lies in the very large number of extracted features. Fuzzy-rough feature selection provides a means by which discrete or real-valued noisy data can be effectively reduced without human intervention. This paper proposes an implementation for automatic sentiment classification of drug reviews employing fuzzy rough feature selection. Experimental results demonstrate that the employment of fuzzy-rough feature selection can indeed significantly reduce the complexity of feature space and the classification run-time overheads while maintaining classification accuracy.

I. Introduction

Sentiment analysis, also called opinion mining, is the field of study that analyzes people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, events, and topics [1]. It is a type of subjectivity analysis which examines sentiment in a given textual unit with the objective of understanding the sentiment polarities (i.e., positive, negative, or neutral) of the opinions toward various aspects of a subject [2]. A lot of sentiment analysis work has been done in the general areas of electronic products, movies, and restaurants reviews, but not extensively in the public health and medical domains, possibly owing to the concerns regarding privacy and ethic issues [2], [3], [4]. However, with the ever increasing popularity of various social media, the online communities and forums make it possible for users to express their experiences and opinions anonymously and freely on drug reviews related to multiple aspects, including effectiveness, side effects, conditions, costs and dosages, which can be leveraged to obtain valuable insights to improve general public health and medical care [5].

Analyzing sentiments concerning various aspects of drug reviews can provide a wealth of information regarding user preferences and experiences, which help make informed decisions by medical professionals, e.g., through strengthened public health monitoring [5]. With the popularity of clinical decision support systems such as therapy recommender systems, which aim at helping to find an optimal personalized therapy

option for a given patient, the clinicians and the systems will greatly benefit from patients' feedback on therapy [6]. The sentiment analysis of post-marketing drug surveillance on the effectiveness and potential risks of adverse drug reactions also plays a major role concerning drug safety once a drug has been released [7].

Sentiment classification can be generally handled with a number of different approaches, including lexicon-based and machine learning-based. A fundamental requirement for lexicon-based approach is a readily available list of pre-labeled words of sentiment expressions in natural language text [4]. All the words in an unknown text are then compared to the words in the predefined lexicon, resulting in a polarity score through computing the difference between the numbers of positively and negatively assigned words [8]. However, lexicon-based techniques often do not consider the possibly different meanings of the words in different contexts. They usually underperform machine learning-based methods that implement sentiment analysis with a computational model learned by converting raw text into numerical features, typically via the bag of words [9] or alternative mechanisms such as the term frequency-inverse document frequency method [10].

One of the major challenges in sentiment classification, especially for machine-learning techniques lies in the very large number of extracted features that may be irrelevant, redundant or even misleading [11]. Feature selection is an important step in sentiment analysis that aims at selecting an optimum feature subset without disrupting their original meanings [12]. It helps minimise noise and redundancy in the feature space which would otherwise adversely increase the likelihood of model overfitting and prevent quality features from being incorporated [13]. Efficiency is another benefit from the utilization of feature selection. Having a reduced number of features helps to reduce run-time overheads, which also implies relaxed memory and storage requirements. In particular, Fuzzy-rough feature selection (FRFS) [14] provides a means by which discrete or real-valued noisy data (or a mixture of both) can be effectively reduced without the need for user-supplied information. It has been developed and applied in a number of applications (e.g., [15]). However, little attention has been paid to the work of FRFS with application to drug review analysis.

Inspired the above observations, this paper proposes an

approach for the automatic sentiment classification of drug reviews employing FRFS. In particular, the proposed work first extracts a list of significant words as features from original drug review documents, whereby both the bag of words and term frequency-inverse document frequency techniques are employed for the generation of discrete and real-valued features, respectively. FRFS is then exploited to identify a minimal representation of the original information by returning a subset of previously extracted features, which is subsequently utilised as input to a number of popular classification algorithms. The case study is conducted on data collected from a popular website on drug information to both consumer and healthcare professionals, for the retrieval of users reviews on drug experience and ratings which are converted to three sentiment labels, i.e., positive, neutral and negative.

The reminder of this paper is organised as follows. Section II introduces the background of recent advances on drug review analysis and the basic concept of FRFS. Section III describes the pipeline whereby FRFS is employed to perform sentiment analysis on drug reviews. Section IV presents and discusses comparative experimental results. Section V concludes the paper and outlines ideas for further development.

II. BACKGROUND

Owing to the significance of mining drug reviews that could contribute to various healthcare stakeholders, a number of advancements have been reported in the recent literature. For example, a framework [8] has been developed to track user experiences regarding drugs and cosmetics on social media data, where learning classifiers are utilised to predict sentiment orientations. To consider the fact that one single sentence may contain multiple clauses discussing multiple aspects of a drug, a clause-level sentiment analysis algorithm [2] has been introduced, which adopts a pure linguistic approach from prior sentiment scores assigned to individual words, while simultaneously taking the grammatical relations between, and semantic annotation of, words into consideration. Apart from sentiment analysis concerning overall satisfaction, side effects and effectiveness of user reviews on drugs have also been investigated [5]. Multiple facets of sentiment in the context of medicine including drug reviews are characterised in [4], whereby a quantitative assessment is conducted with respect to word usage and sentiment distribution.

However, none of the existing literature in the relevant areas considers feature selection techniques that may extract feature subsets of minimal knowledge representation without degrading the performance of the learning classifiers. Yet, much work has been established to conduct feature selection. In particular, FRFS has originated from the development of rough set theory, which is able to find minimal knowledge representation through data-driven learning without thresholds or expert knowledge [14]. Being complementary to rough sets that are concerned with indiscernibility, fuzzy set theory is concerned with vagueness and has been successfully used in a wide range of domains [16], [17], [18], [19], [20]. The hybridization of both theories especially in the area of feature selection has led to the popularity of FRFS with robust solutions.

Specifically, a fuzzy-rough set is defined by two fuzzy sets, i.e., a fuzzy lower and a fuzzy upper approximation, obtained by extending the corresponding crisp rough set notions. In the crisp case, elements either belong to the lower approximation with absolute certainty or not at all. In the fuzzy-rough case, elements may have a membership in the range [0,1], allowing greater flexibility in handling uncertainty. Let $IS = (\mathbb{U}, \mathbb{A})$ be an information system, where $\mathbb{U}$ is a nonempty set of finite objects (the universe) and $\mathbb{A}$ is a nonempty finite set of attributes such that $a : \mathbb{U} \rightarrow V_a$ for every $a \in \mathbb{A}$. V_a is the set of values that the attribute a may take. For decision systems, $\mathbb{A} = \{\mathbb{C} \cup \mathbb{D}\}$, where $\mathbb{C}$ is the set of input features and $\mathbb{D}$ is the set of decision features. The following defines the fuzzy lower and upper approximations:

$$\mu_{\underline{R_B}X}(x_i) = \inf_{x_j \in \mathbb{U}} I(\mu_{R_B}(x_i, x_j), \mu_X(x_j)) \quad (1)$$

$$\mu_{\overline{R_B}X}(x_i) = \sup_{x_j \in \mathbb{U}} T(\mu_{R_B}(x_i, x_j), \mu_X(x_j)), \quad (2)$$

where X is the fuzzy concept being approximated, I is a fuzzy implicator, T is a t-norm, and R_B is the fuzzy similarity relation induced by the subset of features B, and $x_i, x_j \in X$ are two arbitrary objects in X. In particular,

$$\mu_{R_B(x_i, x_j)} = T_{a \in B}\{\mu_{R_a}(x_i, x_j)\} \quad (3)$$

where $\mu_{R_a}(x_i, x_j)$ is the degree to which the objects x_i and x_j are similar for the feature $a \in \mathbb{A}$. Many similarity relations [21] can be constructed for this purpose, for example:

$$\mu_{R_a}(x_i, x_j) = 1 - \frac{|a(x_i) - a(x_j)|}{a_{max} - a_{min}} \quad (4)$$

$$\mu_{R_a}(x_i, x_j) = \exp\left(-\frac{(a(x_i) - a(x_j))^2}{2\sigma_a^2}\right) \quad (5)$$

$$\mu_{R_a}(x_i, x_j) = \max\left(\min\left(\frac{a(x_j) - (a(x_i) - \sigma_a)}{a(x_i) - (a(x_i) - \sigma_a)},\right.\right.$$
$$\left.\left.\frac{(a(x_i) + \sigma_a) - a(x_j)}{(a(x_i) + \sigma_a) - a(x_i)}\right), 0\right) \quad (6)$$

where σ_a^2 is the variance of the feature a, and $a(x_i)$ is the value of a for the object x_i. The choices for I, T and the fuzzy similarity relation have great influence upon the resultant fuzzy partitions.

FRFS employs a quality measure termed the fuzzy-rough dependence function $\gamma_B(Q)$ that measures the dependency between two sets of attributes B and Q, which is defined by:

$$\gamma_B(Q) = \frac{\sum_{x \in \mathbb{U}} \mu_{POS_{R_B}(Q)}(x)}{|\mathbb{U}|}. \quad (7)$$

where the fuzzy positive region, which contains all objects of U that can be classified into classes of $\mathbb{U}/Q$ using the information in B, is defined as:

$$\mu_{POS_{R_B}(Q)}(x) = \sup_{X \in \mathbb{U}/Q} \mu_{\underline{R_B}X}(x). \quad (8)$$

That is to say, $\gamma_B(Q)$ may be viewed as a measure of quality for a given feature subset $B \in \mathbb{C}$, with respect to the set of decision features $\mathbb{D}$: $0 \leq \gamma_B(Q) \leq 1$, where $\gamma_\emptyset(\mathbb{D}) = 0$. A fuzzy-rough reduct R can then be defined as a subset of features that preserves the dependency degree of the entire data set, i.e., $\gamma_R(\mathbb{D}) = \gamma_\mathbb{C}(\mathbb{D})$.

III. Sentiment Classification of Drug Reviews

This section describes the major modules of the framework developed for sentiment classification of drug reviews using FRFS. Figure 1 depicts the flow chart of the general pipeline. This includes the learning phase where the sentiment classification model is trained with a collection of user online reviews on their drug experience. In particular, a preprocessing step is performed on the training data to prepare for clean documents such that text features can be extracted. FRFS is then executed to search for a minimal knowledge representation. The subset of features returned by FRFS is fed into a certain learning classification algorithm which performs classifier learning. Once trained, only those features selected by FRFS during the training phase are utilized in the learned sentiment model. Within the application or testing phase the functionality of the preprocessing module is the same to that of its counterpart in the training phase, but the feature extraction module is simpler than its counterpart in the training phase as it only needs to produce those features selected by FRFS during the training.
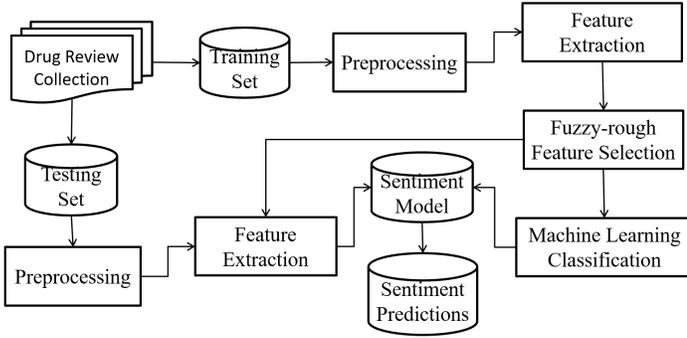

Fig. 1.   Flow chart of proposed framework

A. Pre-processing

Once the raw drug reviews are collected, a number of preprocessing steps are necessary for the generation of clean documents for further processing. These include the following:

1) Tokenize the reviews such that each review is represented as a collection of words for text analysis;
2) Convert all text data to lowercase, so that the words of different cases could be treated the same to remove redundancy;
3) Erase punctuation and symbols, which can safely be ignored without sacrificing the meaning of the sentence;
4) Remove a list of stop words such as 'and' and 'the' that does not add much meaning to a sentence;
5) Lemmatize the words to reduce words to their dictionary forms such that for example, 'am', 'are' and 'is' can all be converted to 'be'.

B. Text Feature Extraction

Once the preprocessing of raw reviews is completed, each resultant tokenized review is represented as a matrix of the length that is equal to the number of unique terms in the returned corpus. The value of each term in the corresponding document is determined by the application of either the bag of words (BoW) method or the term frequency-inverse document frequency (tf-idf). BoG is a simplifying representation used in natural language processing whereby each drug review is represented as a multiset of words, and the frequency of occurrence of each word is subsequently used as the feature value for training the sentiment model.

Different from BoG, the tf-idf method is intended to reflect how important a word is to a document in a certain collection of documents. The tf-idf value of a term t in a document d is defined as:

$$\text{tf-idf}(t, d, D) = tf(t, d) \times idf(t, D) \qquad (9)$$

where $tf(t, d)$ is the term frequency, i.e., the raw count of t in d, and $idf(t, D)$ is the inverse document frequency which measures how common or rare of a particular word is across all documents D. In particular, $idf(t, D)$ can be defined as:

$$idf(t, D) = log\frac{N}{1 + |\{d \in D : t \in d\}|} \qquad (10)$$

where $N = |D|$ is the total number of documents considered, $|\{d \in D : t \in d\}|$ is the number of documents where the term t appears. In general, the tf-idf value increases proportionally to the number of times a word appears in the document concerned and is offset by the number of documents in the corpus that contain the word.

C. Fuzzy-Rough Feature Selection (FRFS)

Following on the above generation of feature values, an artificial data set can be constructed for subsequent feature selection as generally illustrated in Table I, where l_i is the label or sentiment of a certain document d_i and v_{ij} is the value of the term $t_j, j \in \{1, \ldots, M\}$ in $d_i, i \in \{1, \ldots, N\}$. Note that it is the v_{ij} that can be computed by the use of either BoW or tf-idf.

TABLE I.   Data Set Matrix

	t_1	$\ldots$	t_j	$\ldots$	t_M	L
d_1	v_{11}	$\ldots$	v_{1j}	$\ldots$	v_{1M}	l_1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_i	v_{i1}	$\ldots$	v_{ij}	$\ldots$	v_{iM}	l_i
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_N	v_{N1}	$\ldots$	v_{Nj}	$\ldots$	v_{NM}	l_N

As outlined previously in Section II, the evaluation of $\gamma_R(\mathbb{D})$ results in the popular hill climbing-based FRFS algorithm that is termed fuzzy-rough QuickReduct [22]. It works by adding to the current candidate feature subset a feature that leads to the highest fuzzy-rough dependency improvement. The application of fuzzy-rough QuickReduct to drug review analysis is shown in Algorithm 1. It terminates when the addition of any remaining feature does not result in an increase in dependency. Note that with the QuickReduct, for a problem with the total number of terms being M in the corpus, the worst case will result in $(M^2 + M)/2$ evaluations of the dependency function. However, as FRFS is used for dimensionality reduction prior to any involvement of a given application which will exploit those features belonging to the resultant reduct, this operation has no negative impact upon the run time efficiency of the system.

Algorithm 1 FRQuickReduct for drug review analysis

Input:
T, set of all extracted text features, i.e., terms $t_j, j \in \{1, \ldots, M\}$;
L, attribute that indicates sentiment label of a given drug review;

1: $S \leftarrow \{\}$ % set of final selected text features;
2: $\gamma_{best} = 0; \gamma_{prev} = 0$
3: **repeat**
4:    $Temp \leftarrow S$
5:    $\gamma_{prev} = \gamma_{best}$
6:    **for all** $t_j \in (T - S)$ **do**
7:      **if** $\gamma_{S \cup \{t_j\}}(L) > \gamma_{Temp}(L)$ **then**
8:        $Temp \leftarrow S \cup \{t_j\}$
9:        $\gamma_{best} = \gamma_{Temp}(L)$
10:      **end if**
11:    **end for**
12:    $S \leftarrow Temp$
13: **until** $\gamma_{best} == \gamma_{prev}$
14: **return** S

D. Sentiment Classification

The execution of FRFS on the originally preprocessed data results in a set of instances $D_i = (v_{i1}, \ldots, v_{iM'}, l_i)$ whose dimensionality is typically significantly reduced from that of the artificially generated data set. Here, D_i is the i-th review with the sentiment label l_i and is composed of M' attributes. To demonstrate the efficiency and effectiveness of the feature subset returned by FRFS, four popular machine learning algorithms are employed to train the sentiment classification model, including:

- Naive Bayes [23] is a probalistic learning classifier, based on a direct application of the Bayesian theorem with strong independence assumptions. It has often been adopted as a baseline in text classification problems.

- C4.5 Decision Tree [24] is one of the most popular learning classifiers. The tree is constructed by starting with a givenl data set at the root node and iteratively expanding each of the branches, until all instances in the branch belong to the same class or no further information gain is provided by adding more features.

- Random Forest [25] is a meta estimator that combines the idea of bagging and random selection of features to fit a number of decision tree classifiers. It works on various sub-samples of a data set and averages the results to improve the final classification accuracy and control over-fitting.

- JRip [26] is a popular crisp classification rule learning algorithm that follows a divide-and-conquer strategy. Crisp rules are created incrementally one at a time, followed by an immediate simplification procedure. Once a set of rules for a given class is completed, an optimisation process is further imposed to fine tune the rules.

Through the use of any of the above learning mechanisms, the trained model is then utilised to perform classification given an unknown document.

IV. EXPERIMENT

This section presents and discusses the results of an experimental investigation into the proposed framework, starting with a brief introduction to the data set used.

A. Data Set

The data set used for drug review analysis is collected by scraping from the raw HTML files using the Beautiful Soup Library in Python from Druglib.com (which provides drug-related information for both consumers and healthcare professionals). This data set has initially been exploited in [5], which includes a total number of 4142 reviews and which has been split into fixed training and testing partitions based on a stratified random sampling scheme with a splitting ratio of 75% to 25%. In particular, the reviews given by the users are collected from the perspective of side effect, effectiveness and comments, which are then merged together to form the overall reviews. The ratings in the range of 1 to 10 are supplied by users for their overall satisfaction, which are converted into sentiment labels as *Positive* for ratings between 7 to 10, *Neutral* for ratings with 5 or 6, and *Negative* for ratings between 1 to 4, following the exact convention as the collected data was first applied [5]. The sentiment distribution is summarised as shown in Figure 2.
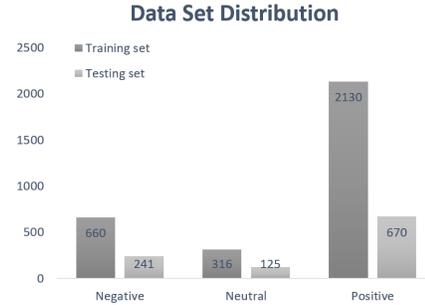
Fig. 2. Label distribution of training and testing data sets

B. Experimental Results

The extraction of a collection of raw training drug reviews results in a total number of 903 features, which can be summarised as shown in Figure 3 by a word cloud that creates a visual representation of the text data, where the prominence of individual terms is reflected by size and font.

The fuzzy-rough QuickReduct algorithm is then performed to search for a subset of features that maintain full dependency (in theory or as high dependency as possible in practice), as a complete set of features does. In particular, two different strategies may be taken while searching for the feature subset, namely, the forward approach that starts with an empty candidate set and iteratively adds a feature that results in the highest dependency improvement to the current subset, and the backward approach that starts with the full feature set and iteratively removes a feature that does not lead to any decrease in dependancy.

By the use of BoW for specifying feature values, the completion of the feature selection process leads to a significant reduction in the resultant feature space. FRFS helps to

reduce the original 903 features to just 42 when the forward search strategy is adopted, and 56 when the backward search is employed. These results are depicted by the corresponding word clouds as shown in Figure 4 and 5. Importantly, despite both returned feature subsets are substantially smaller, the resulting word clouds are able to identify those key words such as 'take' and 'day' that have also been prominent in the word cloud generated by the use of the complete feature set. The two reduced clouds also share a lot of words in common with the list of top 10 frequent terms as extracted from the established work of [4].

Fig. 3.   Word cloud on preprocessed training data without feature selection

Fig. 4.   Word cloud on data searched forward with fuzzy-rough QuickReduct

Fig. 5.   Word cloud on data searched backward with fuzzy-rough QuickReduct

The performance of sentiment classification employing the two feature subsets is then verified using four popular machine learning approaches that have been briefly explained in Section III-D. Note that the Weka Machine Learning Toolkit [27] is utilised to implement both the learning classifiers and fuzzy-rough QuickReduct, all involving the use of default parameters unless otherwise explicitly specified. Table II lists the experimental results when BoW is employed to compute the values of extracted features, where Trn shows the training accuracy (%), Tst demonstrates the performance of running independent hold-out test (%), and Time presents the runtime (seconds) overheads of constructing the model while using the corresponding feature subset as indicated.

TABLE II.   Results with BoW

Algorithms	Original 903 features			42 features by forward search			56 features by backward search		
	Trn	Tst	Time	Trn	Tst	Time	Trn	Tst	Time
Naive Bayes	61.37	57.34	0.23	62.11	59.56	0.01	62.20	61.97	0.02
Decision Tree	92.21	58.11	16.91	84.48	56.95	0.27	78.72	61.97	0.41
RandomForest	100.00	66.41	10.10	100.00	66.41	1.51	100.00	66.41	2.25
Ripper	70.19	65.73	8.57	69.12	64.96	0.62	70.22	65.06	0.58
Average	80.94	61.90	8.95	78.93	61.97	0.60	77.78	**63.85**	0.82

C. Experimental Analysis

Whilst being one of the most commonly used baselines in text classification, Naive Bayes has generally performed the worst in comparison to others in this study. However, the testing performances using feature subsets searched either forward or backward have made an over 2% and 4% improvement, respectively. Using C4.5 Decision Tree, it is also shown that the employment of the feature subset searched backwards is able to outperform the use of the full original feature set by nearly 4%. For Random Forest, the two testing performances using fuzzy-rough QuickReduct feature selection are exact the same as that using the complete set of features. Together, these results may appear counter intuitive as the use of less features actually performs better. However, this is not a surprise as many of the features contain noise. Their removal can indeed help improve the classification accuracy, as generally proven in the relevant feature selection literature [28].

Similar phenomenon occurs when tf-idf is used to calculate feature values, as shown in Table III. The results of utilising the feature subsets returned by both forward and backward search are shown in Figure 4 and 5. Collectively, these results demonstrate that fuzzy-rough QuickReduct works well for both nominal and real-valued features. It is therefore not a surprise to notice that the averaged testing performances of employing the two reduced feature subsets are also better than those attainable by the use of the complete feature set. That is to say, the reduced feature subset generated by removing those features that may be redundant and even noisy is able to improve upon or at least maintain the existing performances. More importantly, this does not come along with the sacrifice of additional run-time overheads, which is reduced even more significantly over 10 times on average in comparison to that required if the complete feature set is adopted.

TABLE III.   Results with tf-idf

Algorithms	Original 903 features			42 features by forward search			56 features by backward search		
	Trn	Tst	Time	Trn	Tst	Time	Trn	Tst	Time
Naive Bayes	61.88	57.43	0.20	62.43	59.75	0.03	62.07	61.68	0.02
Decision Tree	92.21	58.11	18.36	84.51	56.85	0.31	78.72	61.87	0.53
Random Forest	100.00	67.08	10.36	100.00	66.41	1.64	100.00	65.73	2.29
Ripper	72.12	64.86	13.52	69.12	64.86	0.53	69.32	64.77	0.60
Average	81.55	61.87	10.61	79.02	61.97	0.63	77.53	**63.51**	0.86

An interesting observation from the above is that performances using the feature subset returned by backward search are generally better than those returned by forward search. However, when the number of original features is very large, searching backward is computationally more expensive and may even be impractical. After all, running FRFS is mainly to reduce computational costs. Hence, in practical applications of the proposed framework, it is necessary to trade off between effectiveness and efficiency, while deciding on whether forward or backward search is to be employed. As a general guideline, unless it is affordable to run FRFS in a backward

search manner, the forward search-based version should be used.

As an initial work here to test the efficacy of running FRFS on collected drug reviews (in the scale of hundreds of text features), the above experimental investigation shows promising results. In particular, the classification performance is maintained while saving massive amounts of run-time overheads. The proposed framework can be expected to bring forward more significant cost-efficiency savings to real-world healthcare analysis on large scale data, although this is subject to further experimental confirmation. Note again, of course that running FRFS for dimensionality reduction is independent of application problems. Running FRFS simply exploits those feature subsets belonging to the resultant reduct during the training phase, prior to the application phase and hence, has no negative impact upon the run time efficiency of the trained system.

V. Conclusion

The exploitation of drug reviews is able to shed light on the understanding of users' preference and drug experience, which may be exploited to help with decision making by the medical professionals and improve public health. This paper has proposed a sentiment classification approach using fuzzy-rough feature selection with a focussed application on drug review analysis. The case study has shown that supported with FRFS, popular machine learning approaches to learning classifiers are able to produce preserved or event improved performance while significantly reducing feature space and run-time overheads, as compared to the results achievable without FRFS.

Considering the different effects due to the use of different feature subsets that are returned by different search mechanisms, future work will be set to exploit search strategies that may be of most beneficial to the overall performance. It would also be interesting to investigate the use of alternative approaches for learning classifiers (e.g., using the recently proposed fuzzy rule-based models like the one proposed in [29], [30]) which may work better while dealing with the uncertainty inherent in natural language processing.

References

[1] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[2] J. Na and W. Y. M. Kyaing, "Sentiment analysis of user-generated content on drug review websites," *Journal of Information Science Theory and Practice*, vol. 3, no. 1, pp. 6–23, 2015.

[3] T. Chen, C. Shang, P. Su, G. Antoniou, and Q. Shen, "Effective diagnosis of diabetes with a decision tree-initialised neuro-fuzzy approach," in *UK Workshop on Computational Intelligence*. Springer, 2018, pp. 227–239.

[4] K. Denecke and Y. Deng, "Sentiment analysis in medical settings: New opportunities and challenges," *Artificial Intelligence in Medicine*, vol. 64, no. 1, pp. 17–27, 2015.

[5] F. Gräßer, S. Kallumadi, H. Malberg, and S. Zaunseder, "Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning," in *Proceedings of the 2018 International Conference on Digital Health*. ACM, 2018, pp. 121–125.

[6] F. Gräßer, H. Malberg, S. Zaunseder, S. Beckert, D. Küster, J. Schmitt, and S. Abraham, "Neighborhood-based collaborative filtering for therapy decision support," in *International Workshop on Health Recommender Systems co-located with ACM RecSys*, 2017, pp. 22–26.

[7] I. Korkontzelos, A. Nikfarjam, M. Shardlow, A. Sarker, S. Ananiadou, and G. H. Gonzalez, "Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts," *Journal of Biomedical Informatics*, vol. 62, pp. 148–158, 2016.

[8] H. Isah, P. Trundle, and D. Neagu, "Social media analysis for product safety using text mining and sentiment analysis," in *UK Workshop on Computational Intelligence*. IEEE, 2014, pp. 1–7.

[9] M. McTear, Z. Callejas, and D. Griol, *The Conversational Interface: Talking to Smart Devices*. Springer, 2016.

[10] S. Robertson, "Understanding inverse document frequency: On theoretical arguments for idf," *Journal of Documentation*, vol. 60, no. 5, pp. 503–520, 2004.

[11] R. M. Elawady, S. Barakat, and N. M. Elrashidy, "Different feature selection for sentiment classification," *International Journal of Information Science and Intelligent System*, vol. 3, no. 1, pp. 137–150, 2014.

[12] C. Shang and Q. Shen, "Aiding classification of gene expression data with feature selection: A comparative study," *International Journal of Computational Intelligence Research*, vol. 1, no. 1, pp. 68–76, 2005.

[13] A. Abbasi, S. France, Z. Zhang, and H. Chen, "Selecting attributes for sentiment classification using feature relation networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 3, pp. 447–462, 2011.

[14] R. Jensen and Q. Shen, "New approaches to fuzzy-rough feature selection," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 4, pp. 824–838, 2009.

[15] P. Su, C. Shang, Y. Zhao, T. Chen, and Q. Shen, "Fuzzy rough feature selection based on owa aggregation of fuzzy relations," in *IEEE International Conference on Fuzzy Systems*. IEEE, 2017, pp. 1–6.

[16] T. Chen, P. Su, C. Shang, and Q. Shen, "Reliability-guided fuzzy classifier ensemble," in *Fuzzy Systems, 2017 IEEE International Conference on*. IEEE, 2017, pp. 1–6.

[17] L. Yang, F. Chao, and Q. Shen, "Generalized adaptive fuzzy rule interpolation," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 4, pp. 839–853, 2017.

[18] P. Su, C. Shang, T. Chen, and Q. Shen, "Exploiting data reliability and fuzzy clustering for journal ranking," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 5, pp. 1306–1319, 2017.

[19] L. Yang and Q. Shen, "Adaptive fuzzy interpolation," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 6, pp. 1107–1126, 2011.

[20] T. Chen, P. Su, C. Shang, and Q. Shen, "Weighted fuzzy rules optimised by particle swarm for network intrusion detection," in *IEEE International Conference on Fuzzy Systems*. IEEE, 2018, pp. 1–7.

[21] P. Su, Q. Shen, T. Chen, and C. Shang, "Ordered weighted aggregation of fuzzy similarity relations and its application to detecting water treatment plant malfunction," *Engineering Applications of Artificial Intelligence*, vol. 66, pp. 17–29, 2017.

[22] R. Jensen and Q. Shen, "Fuzzy-rough sets assisted attribute selection," *Fuzzy Systems, IEEE Transactions on*, vol. 15, no. 1, pp. 73–89, 2007.

[23] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[24] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008.

[25] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[26] W. W. Cohen, "Fast effective rule induction," in *Proceedings of the 12th International Conference on Machine Learning*, 1995, pp. 115–123.

[27] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[28] R. Jensen and Q. Shen, "Are more features better?" *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 6, pp. 1456–1458, 2009.

[29] T. Chen, Q. Shen, P. Su, and C. Shang, "Induction of quantified fuzzy rules with particle swarm optimisation," in *Fuzzy Systems, 2015 IEEE International Conference on*. IEEE, 2015, pp. 1–7.

[30] T. Chen, C. Shang, P. Su, and Q. Shen, "Induction of accurate and interpretable fuzzy rules from preliminary crisp representation," *Knowledge-Based Systems*, vol. 146, pp. 152–166, 2018.