Fuzzy Rule Based Systems for Interpretable Sentiment Analysis

Han Liu and Mihaela Cocea
School of Computing, University of Portsmouth
Buckingham Building, Lion Terrace, Portsmouth, United Kingdom PO1 3HE
Email: han.liu@port.ac.uk, mihaela.cocea@port.ac.uk

Abstract—Sentiment analysis, which is also known as opinion mining, aims to recognise the attitude or emotion of people through natural language processing, text analysis and computational linguistics. In recent years, many studies have focused on sentiment classification in the context of machine learning, e.g. to identify that a sentiment is positive or negative. In particular, the bag-of-words method has been popularly used to transform textual data into structured data, in order to enable the direct use of machine learning algorithms for sentiment classification. Through the bag-of-words method, each single term in a text document is turned into a single attribute to make up a structured data set, which results in high dimensionality of the data set and thus negative impact on the interpretability of computational models for sentiment analysis. This paper proposes the use of fuzzy rule based systems as computational models towards accurate and interpretable analysis of sentiments. The use of fuzzy logic is better aligned with the inherent uncertainty of language, while the "white box" characteristic of the rule based learning approaches leads to better interpretability of the results. The proposed approach is tested on four datasets containing movie reviews; the aim is to compare its performance in terms of accuracy with two other approaches for sentiment analysis that are known to perform very well. The results indicate that the fuzzy rule based approach performs marginally better than the well-known machine learning techniques, while reducing the computational complexity and increasing the interpretability.

Keywords—Data Mining; Machine Learning; Fuzzy Rule Based Systems; Text Classification; Sentiment Analysis

I. Introduction

Sentiment analysis is also known as opinion mining and aims to identify the emotion or attitude of people through natural language processing, text analysis and computational linguistics. In recent years, sentiment analysis has been mainly considered as a classification problem in the context of machine learning, e.g. to classify sentiments to be positive or negative. This has led to applications in broad areas such as cyberbullying detection [1] and movie reviews [2].

In the machine learning context, textual data must be transformed into structured data in order to enable the direct use of learning algorithms for sentiment classification. In particular, the bag-of-words method, which considers each single term in a document as an attribute in a structured data set, has been used as a popular approach for this transformation [3]. On this basis, support vector machine [4] and Naive Bayes [5] have been popularly used towards accurate classification of sentiment data. However, computational models built by using

these two algorithms are not easy to interpret due to the nature of the learning strategies of the two algorithms. In particular, models learned by support vector machines have limitations in transparency and depth of learning, and models learned by Naive Bayes are not sufficiently interpretable due to the required assumption that all input attributes are totally independently of each other. More detailed arguments can be seen in [6].

Sentiment analysis involves discovery of opinions from texts, which is an exploratory task in which the results need to be interpretable; however, this had been formulated as a machine learning task, with the focus on classification performance and virtually no attention paid to interpretability. Having interpretable models of sentiment analysis would facilitate the understanding of which aspects of a product lead to a positive or a negative review, opening the possibility of addressing these aspects.

Machine learning for textual data involves massively high dimensionality following the transformation from textual data to structural data by using the bag-of-words method. This high dimensionality, coupled with the incomprehensibility (i.e. "black box" approach) of many predictive models, makes the current models not only non-interpretable, but also highly complex, requiring considerable computational resources for their use.

We argue that the proposed fuzzy rule based approach can address both the interpretability and the computational complexity limitations, while preserving a classification performance in line with the best performing algorithms used for sentiment analysis.

The rest of the paper is organized as follows. Section II introduces theoretical preliminaries related to sentiment analysis and machine learning. In particular, concepts on fuzzy logic, rule based systems and sentiment classification are described. Section III proposes the use of fuzzy rule based systems towards advances in interpretability of computational models for sentiment classification. Section IV reports an experimental study to show that fuzzy rule based systems can demonstrate similar or even better accuracy of sentiment classification. Also, this section reports the actual dimensionality of each of the structural data sets transformed from the chosen sentiment data and the number of fuzzy rules generated from each of these data sets. Section V summarises the contributions of this

paper and suggests further directions towards further advances in this research area.

II. THEORETICAL PRELIMINARIES

We argue that fuzzy rule based systems can increase the interpretability of machine learning models for sentiment analysis. To highlight the characteristics of fuzzy logic and rule based systems that can contribute to higher interpretability, in contrast to the typical sentiment analysis approach using bag-of-words, this section describes theoretical preliminaries related to fuzzy logic, rule based systems and sentiment analysis.

A. Fuzzy Logic

Fuzzy logic is generally an extension of deterministic logic, i.e. the truth value is ranged from 0 to 1 rather than a binary value. The theory of fuzzy logic is mainly aimed at turning a black and white problem into a grey problem [7]. In the context of set theory, deterministic logic is corresponding to crisp sets, which means that each element in a set has a full membership to the set, i.e. the element fully belongs to the set. In contrast, fuzzy logic is corresponding to fuzzy sets, which means that each element in a set only has a partial membership to the set, i.e. the element belongs to the set to a certain degree. The degree of a fuzzy membership is determined by a particular membership function such as trapezoidal membership function, triangular membership function and Gaussian membership function [8].

Fuzzy logic involves some logical operations that are slightly different from those being used in deterministic logic, such as conjunction, disjunction and negation. In terms of conjunction, the min function is used to get the smallest value among the values of the given fuzzy variables. For example, a, b and c are three fuzzy variables with the fuzzy truth values of 0.3, 0.5 and 0.7, respectively; in this case, $a \wedge b \wedge c = min(a, b, c) = 0.3$. For the same example, disjunction involves using the max function instead of the min function, i.e. $a \vee b \vee c = max(a, b, c) = 0.7$. In terms of negation, for the above example, $\neg a = 1 - a = 0.7$. More details on fuzzy operations can be found in [8].

Fuzzy logic has been involved in broad application areas. In particular, fuzzy logic can be used in machine learning to achieve fuzzy classification, regression or clustering towards reduction of bias [9]. In operational research, fuzzy logic can be used to support fuzzy decision making [10] towards reduction of judgement bias. In engineering, fuzzy logic can be used to enable fuzzy modelling [11]. In rule based systems, fuzzy logic can be used to enable the generation and representation of fuzzy rules towards more reasonable and interpretable predictions being made [12]. A more detailed description of fuzzy rule based systems is given in the next section.

B. Rule Based Systems

A rule based system is a special type of expert systems, which typically consists of a set of rules. Each rule also consists of a number of rule terms, which are also known
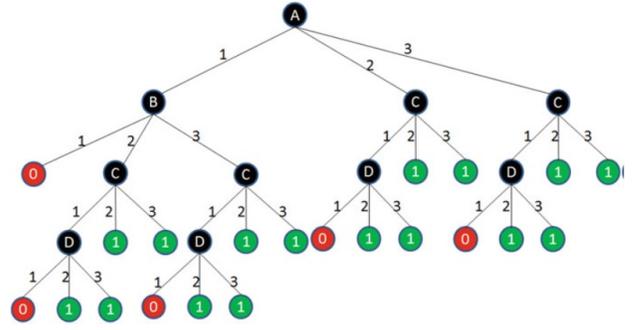

Fig. 1. Replicated Sub-tree Problem [6]

as conditions or antecedents. In general, rule based systems can be designed by using expert knowledge or through learning from real data. In the big data era, machine learning approaches have become increasingly popular for the design of rule based systems and this type of learning approaches is referred to as rule learning. In this context, there are two approaches that can achieve learning of rules, namely divide and conquer [13] and separate and conquer [14].

The divide and conquer approach is also known as Top-Down Induction of Decision Trees (TDIDT). This is because this approach aims to generate rules in the form of a decision tree. Examples of this approach include ID3 [15] and C4.5 [13]. This approach has a serious limitation, which is known as the replicated sub-tree problem [16], i.e. the decision tree generated following this approach may contain redundant terms that make up several identical sub-trees as illustrated in Fig. 1.

Due to the presence of the replicated sub-tree problem, the separate and conquer approach, which aims to generate if-then rules directly from training instances, has become increasingly popular. This approach is also known as the covering approach due to the fact that it generally involves learning one rule that covers some training instances and then learning the next rule on the basis of the remaining instances, i.e. the instances that are covered by the previously generated rules are removed prior to the learning of the next rule. Some examples of the separate and conquer approach include Prism [16] and Ripper [17].

Both of the above two approaches aim to generate deterministic rules, which means that the rules are assumed to have no uncertainty. However, in reality, it is fairly difficult to guarantee that the training data is complete for generation of deterministic rules. From this point of view, deterministic rules are considered to be biased and less reliable for predicting on unseen instances in practice. Therefore, the generation of fuzzy rules, which make up a fuzzy rule based system, has been used to address this problem.

There are three popular types of fuzzy rule based systems, namely Mamdani, Sugeno and Tsukamoto [8]. The first two types of fuzzy rule based systems apply to regression problems, as the output from such systems is a real value, and the third type generally applies to classification problems, as the

output is a discrete value. As we focus on classification, an illustrative example of a Tsukamoto system is provided below.

The Tsukamoto system has two input variables x_1 and x_2 and one output variable y. The variable x_1 has two linguistic terms 'Good' and 'Bad', x_2 has two linguistic terms 'High' and 'Low' and y has two linguistic terms 'Positive' and 'Negative'. The fuzzy membership functions for the above linguistic terms are defined as follows:

Good: $0.25/1, 0.5/2, 0.75/3, 0.5/4, 0.25/5$
Bad: $0.75/1, 0.5/2, 0.25/3, 0.5/4, 0.75/5$
High: $0.3/1, 0.4/2, 0.6/3, 0.7/4, 0.5/5$
Low: $0.7/1, 0.6/2, 0.4/3, 0.3/4, 0.5/5$
Positive: equals to the rule firing strength
Negative: equals to the rule firing strength

There are four rules as follows:
Rule 1: If x_1 is 'Good' and x_2 is 'High' then y = 'Positive';
Rule 2: If x_1 is 'Good' and x_2 is 'Low' then y = 'Positive';
Rule 3: If x_1 is 'Bad' and x_2 is 'High' then y = 'Negative';
Rule 4: If x_1 is 'Bad' and x_2 is 'Low' then y = 'Negative';

In practice, for each rule the firing strength is derived based on the given input values, e.g. if both x_1 and x_2 are assigned the value of 2, then the firing strength of Rule 1 is 0.4, as the fuzzy truth value for 'Good' and 'High' are 0.5 and 0.4, respectively. In this case, Rule 1 provides the output value 'Positive' with the fuzzy truth value of 0.5 towards predicting an unseen instance. Each of these four rules works in the same way and the final output value is determined by taking the output value from the rule with the highest firing strength. The advantages of fuzzy rule based systems are discussed in more detail in Section III-A.

C. Sentiment Analysis

Sentiment analysis generally consists of five stages [18]: enrichment, transformation, preprocessing, vectoring and mining.

The enrichment stage aims to have semantic information added through recognition and tagging of named entities in order to support the term filtering in the later stages. Popular taggers include POS Tagger, Abner Tagger and Dictionary Tagger. More details on text enrichment can be found in [18].

Transformation aims to have textual data transformed into structured data in order to enable the direct use of machine learning algorithms for sentiment classification. In particular, the bag-of-words method is one of the most popular ways to achieve such transformation [1], [19] by turning each single term in a document into a single attribute in the transformed data set. Following the use of the bag-of-words method, it is also necessary to have the frequency of each term calculated in order to enable the filtering of those less frequently occurring terms. In this way, the data dimensionality can be reduced significantly leading to more efficient processing in the later stages.

Preprocessing aims to filter those irrelevant terms such as stop words, punctuation, numbers and terms that contain no

more than n characters [18]. Also, it is necessary to covert upper cases to lower cases and remove endings by using stemming [18]. Usually the terms extracted through creating a bag of words are filtered in the preprocessing stage except for those highly relevant ones that are used further to create a vector of terms in the next stage referred to as vectoring [18].

In the vectoring stage, each term is turned into a numerical attribute and the value reflects the relative frequency of the term appearing in a particular document or the absolute frequency of the term appearing in all the collected documents.

Mining is the last stage of sentiment analysis, which aims to adopt machine learning algorithms to build computational models and classify sentiments on the basis of the structured data set extracted following the previous stages.

III. FUZZY RULE BASED CLASSIFICATION OF SENTIMENTS

We propose the use of fuzzy rule based systems for sentiment analysis towards more accurate and interpretable classifications being made. This section presents the key features of this proposed approach and justifies its significance in both theory and practice.

A. Key Features

The proposed approach involves use of the Tsukamoto fuzzy rule based system due to the fact that this type of fuzzy systems applies to classification problems as introduced in Section 2.1. For each input attribute, the trapezoid fuzzy membership function is adopted to convert numerical values into fuzzy linguistic terms due to its popular use in practice [20].

When the trapezoid fuzzy membership function is used, each linguistic term T involves four key points a, b, c, d regarding the change pattern of a membership degree. An example is illustrated below and in Fig. 2:

$$f_T(x) = \begin{cases} 0, & \text{when } x \leq a \text{ or } x \geq d; \\ (x-a)/(b-a), & \text{when } a < x < b; \\ 1, & \text{when } b \leq x \leq c; \\ (d-x)/(d-c), & \text{when } c < x < d; \end{cases}$$

In the training stage, the values of the above four parameters a, b, c, d are derived for each single attribute so that fuzzy rules are generated. In the testing stage, fuzzy classification involves the following five steps: fuzzification, application, implication, aggregation and defuzzification [8]. The example given in Section 2.2 includes the following rules:

Rule 1: If x_1 is 'Good' and x_2 is 'High' then y = 'Positive';
Rule 2: If x_1 is 'Good' and x_2 is 'Low' then y = 'Positive';
Rule 3: If x_1 is 'Bad' and x_2 is 'High' then y = 'Negative';
Rule 4: If x_1 is 'Bad' and x_2 is 'Low' then y = 'Negative';

For this example, when $a = 2$, $b = 6$, $c = 8$ and $d = 12$ for the fuzzy linguistic terms 'Good' and 'High', and if $x_1 = 3$ and $x_2 = 4$, then the five steps are executed as follows.

Fuzzification:
Rule 1: $f(Good) = 0.25$, $f(High) = 0.5$;
Rule 2: $f(Good) = 0.25$, $f(Low) = 0.5$;
Rule 3: $f(Bad) = 0.75$, $f(High) = 0.5$;
Rule 4: $f(Bad) = 0.75$, $f(Low) = 0.5$;

Application:
Rule 1: $f(Good) \wedge f(High) = Min(0.25, 0.5) = 0.25$;
Rule 2: $f(Good) \wedge f(Low) = Min(0.25, 0.5) = 0.25$;
Rule 3: $f(Bad) \wedge f(High) = Min(0.75, 0.5) = 0.75$;
Rule 4: $f(Bad) \wedge f(Low) = Min(0.75, 0.5) = 0.75$;

Implication:
Rule 1: $f_1(Positive) = Min(0.25, 1) = 0.25$;
Rule 2: $f_2(Positive) = Min(0.25, 1) = 0.25$;
Rule 3: $f_3(Negative) = Min(0.75, 1) = 0.75$;
Rule 4: $f_4(Negative) = Min(0.75, 1) = 0.75$;

Aggregation:
$f(Positive) = f_1(Positive) \vee f_2(Positive)$
$= max(0.25, 0.25) = 0.25$;
$f(Negative) = f_1(Negative) \vee f_2(Negative)$
$= max(0.75, 0.75) = 0.75$;

Defuzzification:
$f(Positive) < f(Negative) => y = Negative$ when $x_1 = 3$ and $x_2 = 4$.

The next subsection outlines the suitability of fuzzy logic combined with rule based systems for sentiment classification problems.

B. Justification

We propose the use of fuzzy rule based systems for sentiment classification due to the advantages of fuzzy logic and rule based systems, as well as their suitability for the problem, as outlined below.

Firstly, fuzzy logic is well capable of dealing with uncertainty in terms of linguistics. In particular, it considers
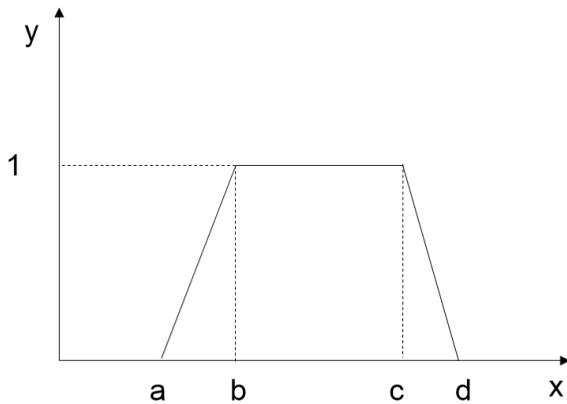

Fig. 2. Trapezoid Fuzzy Membership Function

a classification problem to be a 'degree of grey' problem rather than a 'black and white' problem (currently used in sentiment analysis). This way of defining the problem leads to a reduction of bias on both positive and negative sides. For example, popular algorithms for sentiment classification, such as C4.5 and Naive Bayes, deal with continuous attributes by putting numerical values into different intervals, each of which is used as a condition judgement towards the final classification. This way of handling continuous attributes have been generally criticised as judgement bias in fuzzy logic literature, which can be resolved by using fuzzy linguistic terms instead. In addition, through the use of fuzzy logic, the classification outcome is provided with a certainty factor (fuzzy truth value) rather than an absolute truth.

Secondly, as argued in [6], rule based systems are generally considered to be more interpretable than computational models built by using other popular learning algorithms in sentiment analysis, such as support vector machine learning. This is due to the fact that rule based models are working in a white box manner and thus fully transparent in terms of the mapping relationship between an input and an output.

Thirdly, the combination of fuzzy logic and rule based systems can make rules represented in a form that is highly similar to natural language and can thus make the information extracted from rules more understandable. This will result in higher confidence (i.e. the degree of trust) in the results for people who would like to see the reasoning process of sentiment analysis by machine learning techniques. In particular, to demonstrate a high level of interpretability, fuzzy rules can be represented in the following form (taking the example given in Section 3.1):

When $x_1 = 3$ and $x_2 = 4$:

Rule 1: If x_1 is 'Good' (truth value: 0.25) and x_2 is 'High' (truth value: 0.5) then y = 'Positive' (certainty factor: 0.25);

Rule 2: If x_1 is 'Good' (truth value: 0.25) and x_2 is 'Low' (truth value: 0.5) then y = 'Positive' (certainty factor: 0.25);

Rule 3: If x_1 is 'Bad' (truth value: 0.75)and x_2 is 'High' (truth value: 0.5) then y = 'Negative' (certainty factor: 0.5);

Rule 4: If x_1 is 'Bad' (truth value: 0.75) and x_2 is 'Low' (truth value: 0.5) then y = 'Negative' (certainty factor: 0.5).

Through the fuzzy representation, given a test instance, people can clearly see the extent to which each of the conditions as part of a rule antecedent is satisfied and how certain a rule is towards classifying the test instance.

In sentiment analysis, it is generally inappropriate to consider all types of classification problems to be 'black and white'. For example, in the context of multi-class classification, it is not always the case that different classes are totally mutually exclusive. In movie classification, it is fairly possible that the same movie can be put into different categories

without conflicts. Also, in emotion classification, it is quite sensible that two different emotions can be identified from the same person. From this point of view, fuzzy rule based systems can be useful to indicate that an item may belong to two or more categories as the item has very high degrees of fuzzy memberships to these categories.

On the other hand, it is also necessary to consider sentiment classification problems to be various degrees of grey. This is because people usually have different criteria for judging that a review is positive or negative, which involves a high degree of subjectivity. In fact, it is fairly difficult to consider anything to be perfect, i.e. everything in general may have both positive and negative aspects. For people who seek for things to be perfect, it is more likely that they would judge a review to be negative. In contrast, some other people may judge a review to be positive even if they can only find a few positive aspects, but consider those aspects to be the most important, thus outweighing the negative ones. It is also fairly possible that a sentence has no negative words but actually aim to point out negative aspects in a positive/constructive way.

In the big data era, the judgement bias on both positive and negative sides can be effectively reduced through using generated fuzzy rules with weighted voting towards classification of sentiments. As argued in [21], the presence of big data can generally help reduce the overfitting of predictive models, especially when fuzzy rules are used as the predictive models provided with certainty factors.

In addition, by using fuzzy rules the judgement process can be interpreted, allowing people to understand how the final classification was arrived at. Moreover, the fuzzy rules would allow people to understand in more detail the positive and negative aspects, which, in turn, would enable them to act to make improvements – e.g. for travel industry (hotels, restaurants).

The use of the proposed fuzzy rule based learning approach can also help reduce the computational complexity in comparison with popular machine learning methods such as C4.5. The fuzzy approach also has the same computational complexity as Naive Bayes.

C4.5 is a popular example of decision tree learning, which follows the divide and conquer approach, as introduced in Section 2.2. This learning algorithm involves recursive learning in terms of attribute selection for the root node and each internal node. Therefore, the computational complexity for C4.5 is $O(mn^2)$, where m is the sample size of training data and n is the number of attributes [22].

Naive Bayes is a popular example of Bayesian learning, which employs the Bayes theorem (illustrated in Equation 1) for identification of the probabilistic relationship between an attribute-value pair and a class label. In training stage, it is necessary to go through the whole data set, attribute by attribute, in order to identify all of the probabilistic relationships between attribute-value pairs and class labels. Therefore, the computational complexity for Naive Bayes is $O(mn)$, where m is the sample size of training data and n is the number of attributes [22].

$$P(Y|X) = \frac{P(Y)P(X|Y)}{P(X)} \qquad (1)$$

where X and Y are two events:
- $P(X)$ is read as the probability that event X occurs to be used as evidence supporting event Y.
- $P(Y)$ is read as the prior probability that event Y occurs on its own.
- $P(Y|X)$ is read as the posterior probability that event Y occurs given that event X truly occurs.
- $P(X|Y)$ is read as conditional probability that event X occurs subject to that event Y must occur.

The above fuzzy approach is a special type of rule learning approach. However, in contrast to the separate and conquer approach introduced in Section 2.2, the fuzzy approach does not involve iterative learning in terms of selection of an attribute-value pair to be appended as a term into the left hand side of a rule. In other words, the fuzzy approach only involves one iteration to have all rules generated by going through the whole data set attribute by attribute. Therefore, the computational complexity for the fuzzy rule learning approach is $O(mn)$, where m is the sample size of training data and n is the number of attributes.

IV. EXPERIMENTAL STUDY

This section reports an experimental study to validate empirically the significance of fuzzy rule based systems in terms of accuracy and interpretability of sentiment classification.

In particular, the experiments aim to compare the fuzzy rules approach with the computational models learned by using other popular machine learning algorithms in sentiment classification, in terms of classification accuracy. The number of rules in our approach is also reported showing the suitability of this approach for interpretable analysis of sentiments, comparing with the number of rules produced by one of the most popular machine learning approaches, i.e. decision trees.

This experimental study is conducted by using four polarity data sets on movie reviews. The data sets with the number of instances in the positive and negative categories are listed in Table 1 and more details can be found in [23]–[25].

All the experiments were conducted using the following procedure:
- Step 1: The textual data is enriched by using POS Tagger and Abner Tagger [18];

TABLE I
DATA SETS ON MOVIE REVIEW WITH NUMBER OF POSITIVE AND
NEGATIVE INSTANCES

Data Set	Positive	Negative
$Polarity Dataset V 0.9$	700	700
$Polarity Dataset V 1.1$	700	700
$Polarity Dataset V 1.0$	700	700
$Polarity Dataset V 2.0$	1000	1000

Data Set	Naive Bayes	C4.5	Fuzzy Rules
$Polarity Dataset V 0.9$	0.936	0.942	0.951
$Polarity Dataset V 1.1$	0.942	0.945	0.951
$Polarity Dataset V 1.0$	0.939	0.939	0.962
$Polarity Dataset V 2.0$	0.913	0.938	0.943

- Step 2: The enriched data is transformed through using the Bag-of-Words method;
- Step 3: For each word, its relative frequency, absolute frequency, inverse category frequency and inverse document frequency are calculated towards filtering out words with low frequency;
- Step 4: The words left following the last stage are preprocessed by filtering stop words, words with no more than N Chars, and numbers, stemming porter and erasing punctuation;
- Step 5: Each document is turned into a vector that consists of all the words appearing in the textual data set, each of which is turned into a numerical attribute that reflects the frequency of the word;
- Step 6: All the document vectors are classified to be positive or negative through using machine learning algorithms.

In the text classification stage, the structured data set is divided into a training set and a test set in the ratio of 70:30.

The classification accuracy of using fuzzy rule based systems is compared with the ones of using computational models learned by Naive Bayes and C4.5, respectively. This is in order to test the performance of the fuzzy rule based system in terms of classification accuracy in comparison with popular learning algorithms that are known to perform well on sentiment analysis tasks.

The results are shown in Table IV and they show that the fuzzy rule learning approach performs slightly better than the well known Naive Bayes and C4.5 algorithms, thus indicating the suitability of fuzzy rules approaches for sentiment analysis tasks.

In addition, the experimental study also involves investigation of the number of rules generated by C4.5 and the fuzzy approach on the basis of the high dimensionality of the chosen textual data. This is in order to investigate the level of complexity of the generated fuzzy and non-fuzzy rule based models, which is tightly linked to the issue of interpretability.

As reported in [6], interpretability of computational models can be affected by four factors: model transparency, model complexity, model redundancy and human characteristics. The first three factors indicate the extent to which the model is transparent to people (transparency), the extent to which the model is easy for people to follow (complexity) and the extent to which different parts of the model are redundant (redundancy).

Model transparency highly depends on the nature of learning algorithms. In general, rule based models, both fuzzy and non-fuzzy, work in a white box manner by showing the particular rules that map inputs to outputs. Naive Bayes models also work in a white box manner by showing the posterior probabilities of mapping an input to all the possible outputs, i.e. the list of probabilistic correlations between the inputs and the outputs. Therefore, all three methods, i.e. Naive Bayes, C4.5 and fuzzy rules, are transparent.

The *complexity of models*, learned by any one of the three learning methods, highly depends on the number of attributes and the attributes complexity. For example, lets consider three attributes a, b and c with values of 2, 3 and 4, respectively. In this case, for Naive Bayes, the number of probabilistic correlations is $24 = 2 \times 3 \times 4$, and each of the correlations consists of a conjunction of three attribute-value pairs, i.e. $a = 1 \wedge b = 1 \wedge c = 1$, and a disjunction of all the possible classes with their probabilities, i.e. $class = 0$ ($P = 0.7$) $\vee class = 1$ ($P = 0.3$).

For fuzzy rule learning methods, the number of fuzzy rules generated is also $24 = 2 \times 3 \times 4$. In fact, the order (the number of rule terms) of each fuzzy rule can only be the number of attributes from a data set. In contrast, the number of rules generated by using C4.5 could be up to $59 = 9 + 26 + 24$. In particular, the number of first order rules (with one rule term) is $9 = 2 + 3 + 4$; the number of second order rules (with two rule terms) is $26 = 2 \times 3 + 2 \times 4 + 3 \times 4$, and the number of third order rules (with three rule terms) is $24 = 2 \times 3 \times 4$.

In addition, fuzzy rule learning methods involve replacing numerical values with several fuzzy linguistic terms and thus can reduce the complexity of attributes, which indicates its advantage in comparison with Naive Bayes and C4.5, especially when there is a large number of intervals involved in dealing with continuous attributes by Naive Bayes and C4.5. In sentiment analysis, each variable could be given only one linguistic term, e.g. the term "frequent". In other words, sentiment analysis generally involves measuring the degree to which each word is frequently present in a text instance or document, towards classifying a sentiment instance. This again indicates the advantage of fuzzy rule learning approaches comparing with Naive Bayes and C4.5.

Model redundancy depends on the nature of learning algorithms. In this aspect, decision tree learning algorithms, such as C4.5, have this drawback due to the replicated subtree problem as illustrated in Fig.1, which is a disadvantage comparing with Naive Bayes and Fuzzy rule learning methods.

The last factor that impacts on *model interpretability* typically depends on people's preference and cognitive capacity, i.e. the extent to which people like and are able to look at the model in detail. In fact, people in different domains would usually have different preferences and cognitive capacity in terms of reading particular information. For example, mathematical formulas are hard to interpret by people without background in mathematics and thus, these people would prefer a more accessible form of representation of the information in the models. From this point of view, fuzzy rules would

Data Set	Count(words)	Count(words(left))
$PolarityDatasetV0.9$	523456	1014
$PolarityDatasetV1.1$	515503	1027
$PolarityDatasetV1.0$	517567	1030
$PolarityDatasetV2.0$	726250	1030

be generally easier to interpret due to the fact that the rules are represented in the form of natural languages as illustrated in Section 3.2. Also, as natural languages are used as the most common way of communication between people, the representation of fuzzy rules would be preferred over other ways of information representation.

As mentioned above, model complexity, which is one of the impact factors for interpretability, can be affected by both the nature of learning algorithms and the characteristics of data. Table IV is used to show that the characteristics of sentiment data could result in the issue of interpretability from the perspective of model complexity. In other words, if the empirical results show that sentiment data is generally of massively high dimensionality (a huge number of attributes), then it would indicate the necessity to deal with interpretability issues by using more suitable algorithms, towards reduction of model complexity leading to advances in interpretability.

The results in Table IV show empirically that sentiment data is generally of massively high dimensionality following the data transformation through using the bag-of-words method. Even after any irrelevant terms (attributes) have been filtered, the data dimensionality is still very high (over thousands), which provides the general indication that interpretability is really a problem that is worth to be dealt with through more research in depth.

On the other hand, in order to show how the nature of learning algorithms can impact on interpretability from the perspective of model complexity, we compare the number of rules and the number of terms generated by C4.5 and the fuzzy rule learning approach, respectively. The results shown in Table IV indicate that the rule learning approach generates fewer rules than C4.5 in all the cases and fewer terms than C4.5 in three out of the four cases.

As mentioned earlier in this section, classifying a sentiment instance to a particular category typically depends on the frequent presence of key words in a text comment or document.

Data Set	C4.5		Fuzzy Rules	
$PolarityDatasetV0.9$	50	1225	48	1014
$PolarityDatasetV1.1$	50	1225	42	1027
$PolarityDatasetV1.0$	44	946	36	1030
$PolarityDatasetV2.0$	61	1890	51	1030

In this situation, when the fuzzy approach is used, people can choose to give only one linguistic term, e.g. the term "frequent". In this context, the number of fuzzy rule terms generated would be the data dimensionality (the number of words involved in the training stage). However, each variable may be given multiple fuzzy membership functions towards measuring the degree to which the corresponding word is frequently present in an instance. This is very similar to the case that different countries have different standards for judging the degree to which a person of a particular height belongs to tall people. From this point of view, the number of fuzzy rules generated depends on the number of fuzzy membership functions defined for each single variable.

In terms of the nature of different rule representation techniques, Fig.3 (which only displays a part of the tree due to the huge size) indicates that the decision tree generated by C4.5 through learning from Polarity Dataset V1.0 is huge and complex, i.e. the longest branch is of the length 1030. Also, as argued in [6], decision tree representation requires that all the rules are connected together as different branches of a tree, and thus it is difficult to interpret specific information that needs to be extracted from a particular rule, especially when a tree is large and complex and a particular branch from which information needs to be extracted is very long. In other words, it is not possible to move a node from one position to another one in a tree, people must have a depth-first search towards extracting relevant information from a tree. The similar argument has also been made in [16]. Due to the same constraint on the structure of a tree, it is difficult to show the ranking of rules according to their importance, which is another weakness of the decision tree representation. In addition, it is difficult to show which attribute is more important towards classifying instances. In sentiment analysis, it is required to show which words would be used as keywords for judging the category to which a text instance belong.

In contrast, fuzzy rules are represented in the form of natural languages, which allow ranking of rules and can show explicitly which attributes are more important than others. The example shown below indicates that each of the rule terms can be put in a separate line as a condition of assigning an instance the label "Positive" and can be given a weight showing its importance towards classifying instances. Also, this way of representation can achieve that the rules terms are listed in a descending order according to their importance, i.e. the most important rule term is put in the first line; the second important rule term is put in the second line, and so on. Furthermore, it is easy to interpret why the certainty factor of the rule shown below is 0.4 through looking at the weight of the rule term (least important one) in the last line.

IF car chases is frequently present (weight =0.9)
AND her cat is frequently present (weight =0.8)
AND twist is frequently present (weight =0.7)
....
....
AND petersen is frequently present (weight =0.4)

Fig. 3. Decision Tree Example

THEN the document belongs to Positive (certainty factor= 0.4)

The experimental results show that fuzzy rule based systems can perform as well as, if not better, than well known sentiment analysis learning approaches, while also providing the advantage of inherently handling uncertainty. These results are also relevant in the context of big data when considering the high dimensionality and ambiguity of textual data.

V. CONCLUSION

This paper has proposed the use of fuzzy rule based systems for sentiment classification. In particular, an experimental study has been conducted investigating the performance of fuzzy rule based systems in terms of accuracy and interpretability of sentiment classification.

The results also show that the proposed fuzzy approach can manage effectively to reduce the judgement bias on classifying sentiment data through checking the classification accuracy and that the fuzzy rule based systems learned from the sentiment data with massively high dimensionality have acceptable levels of model complexity in terms of the impact on interpretability through checking the number of generated fuzzy rules.

In addition, fuzzy rule based systems can represent rules in a form that is highly close to natural languages and thus are generally more interpretable than computational models learned by other popular machine learning algorithms. On the basis of the descriptions and justifications throughout this paper, fuzzy rule based systems are recommended to be used more popularly towards advances in dealing with uncertainty and interpretability.

Due to the presence of massively high dimensionality of sentiment data, each rule extracted from a fuzzy rule based system would have a large number of rule terms. In fact, the number of rule terms for a fuzzy rule is equal to the number of attributes in a structural data set, which indicates the necessity to investigate the use of feature extraction to reduce the data dimensionality further (through combination of highly correlated attributes) towards advances in interpretability of fuzzy rule based models for sentiment analysis.

ACKNOWLEDGMENT

REFERENCES

[1] K. Reynolds, A. Kontostathis, and L. Edwards, "Using machine learning to detect cyberbullying," in *Proceedings of the 10th International Conference on Machine Learning and Applications*, December 2011, pp. 241–244.

[2] A. Tripathy, A. Agrawal, and S. K. Rath, "Classication of sentimental reviews using machine learning techniques," *Procedia Computer Science*, vol. 57, pp. 821–829, 2015.

[3] J. Sivic, "Efficient visual search of videos cast as text retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 591–605, April 2009.

[4] N. Cristianini, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge: Cambridge University Press, 2000.

[5] I. Rish, "An empirical study of the naive bayes classifier," *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22, pp. 41–46, August 2001.

[6] H. Liu, M. Cocea, and A. Gegov, "Interpretability of computational models for sentiment analysis," in *Sentiment Analysis and Ontology Engineering: An Environment of Computational Intelligence*, W. Pedrycz and S.-M. Chen, Eds., vol. 639, March 2016, pp. 199–220.

[7] L. Zadeh, "Fuzzy logic: A personal perspective," *Fuzzy Sets and Systems*, vol. 281, pp. 4–20, December 2015.

[8] T. Ross, *Fuzzy Logic with Engineering Applications*. West Sussex: Wiley, 2010.

[9] E. Hllermeier, "Does machine learning need fuzzy logic," *Fuzzy Sets and Systems*, vol. 281, pp. 292–299, December 2015.

[10] S.-M. Chen and L.-W. Lee, "Fuzzy decision-making based on likelihood-based comparison relations," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 3, pp. 613–628, June 2010.

[11] A. Gegov, N. Petrov, B. Vatchova, and D. Sanders, "Advanced modelling of complex processes by fuzzy networks," *WSEAS Transactions on Circuits and Systems*, vol. 10, no. 10, pp. 319–330, October 2011.

[12] L.-X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 6, pp. 1414–1427, December 1992.

[13] R. J. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco: Morgan Kaufmann Publishers, 1993.

[14] J. Furnkranz, "Separate-and-conquer rule learning," *Artificial Intelligence Review*, vol. 13, pp. 3–54, 1999.

[15] R. J. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, March 1986.

[16] J. Cendrowska, "Prism: An algorithm for inducing modular rules," *International Journal of Man-Machine Studies*, vol. 27, pp. 349–370, May 1987.

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

[18] K. Thiel and M. Berthold, "The knime text processing feature: An introduction," Tech. Rep., 2012.

[19] R. Zhao, A. Zhou, and K. Mao, "Automatic detection of cyberbullying on social networks based on bullying features," in *Proceedings of the 17th International Conference on Distributed Computing and Networking*, 2016.

[20] S.-M. Chen, "A fuzzy reasoning approach for rule-based systems based on fuzzy logics," *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, vol. 26, no. 5, pp. 769–778, October 1996.

[21] H. Liu, A. Gegov, and M. Cocea, *Rule Based Systems for Big Data: A Machine Learning Approach*. Switzerland: Springer, 2016.

[22] J. Su and H. Zhang, "A fast decision tree learning algorithm," in *Proceedings of the 21st National Conference on Artificial Intelligence*, 2006, pp. 500–505.

[23] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," in *Proceedings of EMNLP 2002*, 2002, pp. 79–86.

[24] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity," in *Proceedings of ACL*, 2004, pp. 271–278.

[25] ——, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of ACL*, 2005, pp. 115–124.