Extraction of Fuzzy Rules from Incomplete Data with "Do Not Care" and "Lost" Values by Rough Sets

Gülnur YILDIZDAN[1,*], Mehmet KAYA[2]

[1] *Kulu Vocational School, Selcuk University, Konya, Turkey*

[2] *Department of Computer Engineering, Fırat University, Elazığ, Turkey*

Abstract

Rough Set Theory (RST) is a mathematical method used in reasoning and information extraction for expert systems. RST makes incomplete, inadequate or ambiguous information appropriate for data analysis by editing it. Today, incomplete data are found in many datasets. Extracting rules from these incomplete data, which are frequently found in disease data, is extremely important in the diagnosis of diseases. In this study, an algorithm previously proposed and extracting fuzzy rules from datasets containing only “do not care” missing attribute value by RST was developed in a way that it can extract fuzzy rules from datasets containing missing attribute value in both “do not care” and “lost” type. The algorithm developed was applied to the dataset of thyroid disease and certain and possible fuzzy rules were obtained for the diagnosis of the disease. The performance of the algorithm was investigated on six different datasets that had “do not care” and “lost” kinds of missing attribute values in different numbers. It was found that the algorithm generally produced successful and consistent rules in the datasets that had “do not care” and “lost” missing attribute values.

Keywords: Rough set theory, incomplete data, fuzzy rule, rule extraction, thyroid disease.

1. Introduction

The Rough Set Theory (RST) (Pawlak, 1982), which was proposed by Pawlak in 1982, is a mathematical method used to extract information from indefinite and incomplete data (Pawlak, 2012). The RST, which is based on the equivalence class concept, has applications in many fields, such as medicine, finance, data mining, artificial intelligence, machine learning, classification, data reduction, rule discovery and incomplete data reasoning.

Many learning approaches extract rules from complete datasets. However, an existing dataset may not always be complete. If some attribute values are not known in a dataset, it is called an incomplete dataset. The techniques, which were developed to obtain decision rules from incomplete data systems, apply different methods to manage incomplete data. Examples of these methods include the replacement of the missing value with the most commonly used value for that attribute, replacement of it with the average value of that attribute for numeric attribute values, the assignment of all possible values of the attribute, or the deletion of the objects that have missing attributes (Grzymala-Busse, 2008). RST provides a natural method for dealing with inconsistent and incomplete data, which is the main problem with rule extraction and classification. In this theory, the rule extraction process is carried out with approximations, and the size of the dataset does not change.

In the literature, RST has applications in many different fields, which include determining the interest and preference of the TV viewers (Jayasuruthi et al., 2018), secure end-to-end network communication (Wu et al., 2020), noise cancellation in infrared images (Jia et al., 2021), medical diagnosis of heart disease (Nabwey, 2020), classification of texts (Cekik & Uysal, 2020), methods for identifying and using incomplete data (Çekik & Telçeken, 2018; Luo et al., 2020; Nakata et al., 2020), dynamic incomplete data processing (Luo et al., 2017), and eliminating the incomplete data in smart city services (Abdel-Basset & Mohamed, 2018).

In this study, an algorithm to obtain certain and possible fuzzy rules from a dataset with quantitative values and only "do not care" type of missing attribute values was developed to be applied to datasets with different types of missing attribute values ("do not care" and "lost"). This developed algorithm was applied to thyroid disease data. The algorithm was tested on six different datasets with "do not care" and "lost" attribute values in different numbers and the certain and possible fuzzy rules obtained were evaluated.

The rest of the paper is organized as follows: Section 2 gives background information about rough set. Section 3 defines the missing attribute values and characteristic relations. Section 4 presents the method which extracts of fuzzy rules from an incomplete quantitative dataset. Section 5 shows the experimental results. Finally, Section 6 concludes the paper.

2. Rough Set Theory

Proposed by Zdzislaw Pawlak (Pawlak, 1982) in 1982, RST is a mathematical method that overcomes uncertainties and doubts. As in fuzzy sets, RST does not accept exact limitations. Data are stored in RST as a table consisting of conditions and decision attributes. RST uses the concept of an equivalence class to separate training data into sections according to certain criteria. Two types of sections are created in its learning process, as lower approximation and upper approximation. From these concepts, which constitute the basis of RST, certain rules are obtained with the help of lower approximations, and possible rules are obtained with the help of upper approximations. In this theory, some basic concepts can be explained as follows (Walczak & Massart, 1999).

2.1. Information system

An information system is defined as given in Equation 1, with U is a finite set of objects, and A is the set of attributes.

$$IS = (U, A) \tag{1}$$

Each attribute ($a \in A$) defines an information function. If V_a refers to the definition set of attribute a, the information function is shown as $f_a: U \rightarrow V_a$.

2.2. Indiscernibility relation

If $B \subset A$, the indiscernibility relation is shown as IND(B) for each set of attributes. For $\forall b \subset B$, if $b(x_i) = b(x_j)$, X_i and X_j objects are indistinguishable by the B set of attributes in A. The IND(B) equivalence class is called the elementary set in B since it forms the smallest discernible group of the objects.

2.3. Lower and Upper approximations

RST is established on two ideas as lower and upper approximation, which refer to elements that definitely belong to the set and elements that are likely to belong to the cluster. Let X show the subset of elements in U universe (X⊂U). The lower approximation of X in B is shown as $\underline{BX}$ if B⊆A, and is defined in Equation 2 as the combination of all the elementary sets in X.

$$\underline{BX} = \{x_i \in U \,|[x_i]_{IND(B)} \subset X\} \quad (2)$$

The upper approximation of X is shown as $\overline{BX}$ and is expressed in Equation 3 as the combination of elementary sets whose intersection with X is not an empty set.

$$\overline{BX} = \{x_i \in U|[x_i]_{IND(B)} \cap X \neq 0\} \quad (3)$$

3. Missing Attribute Values and Characteristic Relations

Decision tables, which define conditions by using attribute values and decisions, are used in RST. In many studies conducted on RST, it is considered that data are complete. However, the input values in the decision table might have missing attribute and decision values. In studies aimed to discover data from examples, an example that has missing decision values is unusable in terms of classification. For this reason, it is assumed that only attribute values may be missing. An attribute value may be missing for two reasons, the first is that the value is lost (deleted, for instance). In such a case, although the attribute value is useful, the value is not accessible, and is shown with the symbol "?". The second reason is that the value is insignificant and may be replaced by any possible value of attribute. Such values are called "do not care", and are shown with the symbol "*" (Grzymala-Busse, 2006).

Tables with missing attribute values are defined by characteristic relationships rather than indiscernibility relationships. Table 1 may be given as an example to an incomplete decision table referring to the a_1,a_2 and a_3 condition attributes; d is the attribute of the decision. As seen, Table 1 has both "do not care" and "lost" attribute values.

Table 1: A decision table that contains missing attribute values

	a_1	a_2	a_3	d

x_1	1	1	1	3
x_2	0	?	0	1
x_3	*	*	1	2
x_4	0	1	0	2
x_5	1	0	?	2

Let each decision table be defined as a function, such as "g". This function is expressed as the set of ordered pairs (case, attribute). For a decision table, such as the one given in Table 1, which contains "lost" and "do not care" missing attribute values, the characteristic relation $R(B)$ on set U is defined as given in Equation 4 (Grzymala-Busse, 2006).

$$(x, y) \in R(B) \text{ if and only g } (x, a) = \text{ g } (y, a) \text{ or g } (x, a) = * \text{ or g } (y, a) = *$$

$$\text{For all } a \in B \text{ so that g } (x, a) \neq ? \quad (4)$$

Let B be a non-empty subset of A ,and $x, y \in U$, the $R(B)$ characteristic relationship may be shown as $CB(x)$ characteristic set given in Equation 5(Grzymala-Busse, 2006).

$$CB(x) = \{y \mid (x, y) \in R(B)\} \quad (5)$$

For Table 1, $CA(x)$ characteristic sets are shown as follows:

CA(x_1) = {1, 3}, CA(x_2) = {2, 4}, CA(x_3) = {1,3}, CA(x_4) = {4}, CA(x_5) = {3, 5}.

4. Extraction of Fuzzy Rules from an Incomplete Quantitative Dataset

Hong et al. (Hong et al., 2002) proposed an algorithm using RST to obtain certain and possible fuzzy rules from the incomplete quantitative value dataset and to estimate the missing attribute values. This algorithm was proposed only for the datasets with "do not care" missing attribute values. In this study, the algorithm was developed in a way to be applied to datasets containing "do not care" and "lost" attribute values. It was ensured that the algorithm could also be applied to "lost" missing attribute values after editing the steps associated with the notation and estimation of incomplete data in the original algorithm.

The Algorithm Developed:

1. Divide the objects of the same C_L class into subsets according to class values, showing them as x_L.

2. If $i = 1, \dots, n$ and $j = 1 \dots m$, show the v_{ij}, which is the quantitative value of the object O_i for each A_j attribute as the f_{ij} fuzzy set by using the membership function given in Equation 6.

$$\frac{f_{ij_1}}{R_{j_1}} + \frac{f_{ij_2}}{R_{j_2}} + \cdots + \frac{f_{ij_l}}{R_{j_l}} \quad (6)$$

In Equation 6, the number of fuzzy areas for A_j attribute is shown with L, the *k.fuzzy area* is shown with R_{j_k}, and the fuzzy membership value of v_{ij} in R_{j_k} is shown with f_{ij} . If the O_i object has a missing value for A_j attribute, the value of the attribute is shown with * or ?.

3. Create fuzzy incomplete elementary sets for the attributes. If the O_i object has a certain fuzzy membership value in A_j, write the (O_i, c) expression to the fuzzy incomplete equivalence class; if it has a "do not care" attribute value (*), write (O_i, u) expression, and if it has "lost" attribute value(?), write (O_i, l). Calculate the membership value of the fuzzy incomplete class for $A_j = R_{jk}$ as O_i certain and $f_{ij_k} \neq 0$ according to Equation 7.

$$\mu_{A_{jk}} = \min_i f_{ij_k} \tag{7}$$

4. Start with s=1 (s is the counter referring to the number of attributes operated for fuzzy incomplete lower approximations at the time).

5. Calculate the fuzzy incomplete lower approximation of each B subset, which consists of s attributes for each X_L class according to Equation 8.

$$B_*(X_L) = \{\left(B_k(O_i), \mu_{B_k}(O_i)\right), 1 \leq i \leq n, O_i \in X_L, B_{ck}(O_i) \subseteq X_L, 1 \leq k \leq |B(O_i)|\} \tag{8}$$

In Equation 8, $B(O_i)$ refers to the set of the fuzzy incomplete equivalence class that includes O_i objects, and $B_{ck}(O_i)$ shows the certain part of the kth fuzzy incomplete equivalence class in $B(O_i)$.

6. Apply the following steps for each uncertain O_i object ("do not care" or "lost") in fuzzy incomplete lower approximation.

(6.a) If the O_i object is only in $B_{ck}(O_i)$ fuzzy incomplete equivalence class from the B attribute subset in fuzzy incomplete lower approximation, calculate the uncertain value of O_i according to Equation 9.

$$\frac{\sum_{O_r \in B_{ck}(O_i)} v_{rj} \times f_{rj_k}}{\sum_{O_r \in B_{ck}(O_i)} f_{rj_k}} \tag{9}$$

v_{rj} is the quantitative value of the object O_r for A_j attribute. f_{rj_k} is the fuzzy membership value of v_{rj} in R_{kB} which is the kth area combination. Convert the O_i value that is calculated according to Equation 9 into a fuzzy set. In other words, delete the expression (O_i,u) or (O_i,l) in fuzzy incomplete equivalence class and change as (O_i,c). Calculate the membership values of fuzzy incomplete equivalence classes that contain this attribute again. Go back to the fuzzy incomplete lower approximations.

(6.b) If the O_i object is included in more than one fuzzy incomplete equivalence class in a fuzzy incomplete lower approximation, defer the estimation until it is determined by more attributes.

7. Increase **s** (s=s+1). Repeat steps 5, 6 and 7 as long as $s \leq m$.

8. If the O^i object continues to be in more than one fuzzy incomplete equivalence class in a fuzzy lower approximation, equivalence class with the maximum scalar cardinality is used to estimate the value of the object. Estimation and processing are carried out as described in Step (6a).

9. For each B subset, create certain fuzzy rules from the fuzzy incomplete lower approximation, and identify the effectiveness measure value of the rule according to the membership values of the equivalence classes in lower approximation.

10. From the obtained certain fuzzy rules, delete the ones whose condition parts are more specialized, and those whose effectiveness measure values are equal to or less than other certain fuzzy rules.

11. Set s=1 (**s** is the counter referring to the number of attributes operated at that time for fuzzy incomplete upper approximations).

12. Calculate the fuzzy incomplete upper approximation of each B subset, which consists of s attributes for each X_L class according to Equation 10.

$$B^*(X_L) = \{\left(B_k(O_i), \mu_{B_k}(O_i)\right), 1 \le i \le n, B_{ck}(O_i) \cap X_L \ne \emptyset, B_{ck}(O_i) \not\subset X_L, 1 \le k \le |B(O_i)|\} \quad (10)$$

13. Follow the steps given below for each uncertain O_i object in fuzzy incomplete upper approximation.

(13.a) If the O_i object is only in $B_{ck}(O_i)$ fuzzy incomplete equivalence class from the B attribute subset in fuzzy incomplete upper approximation, calculate the uncertain value of O_i according to Equation 11.

$$\frac{\sum_{O_r \in B_{ck}(O_i) \& O^r \in X_L} v_{rj} \times f_{rj_k}}{\sum_{O_r \in B_{ck}(O_i) \& O^r \in X_L} f_{rj_k}} \quad (11)$$

Convert the O_i value that is calculated according to Equation 11 into a fuzzy set. Delete (O_i,u) or (O_i,l) expression in fuzzy incomplete equivalence class, and change as (O_i,c). Calculate the membership values of fuzzy incomplete equivalence classes that contain this attribute again. Go back to the fuzzy incomplete upper approximations.

(13.b) If the O_i object is in more than one fuzzy incomplete equivalence class in fuzzy incomplete upper approximation,defer the estimation until it is determined by more attributes.

14. Increase **s** (s=s+1). As long as $s \le m$, repeat steps 12, 13, and 14.

15. For each X_L class, calculate the plausibility measure of each fuzzy incomplete equivalence class in upper approximation according to Equation 12.

$$P(B_{ck}(O_i)) = \frac{\sum_{O_r \in B_{ck}(O_i) \& O_r \in X_L} f_{rj_k}}{\sum_{O^r \in B_{ck}(O_i)} f_{rj_k}} \quad (12)$$

In Equation 12, f_{rj_k} refers to the fuzzy membership value of O_r value for A_j attribute in R_{kB}, the kth area combination.

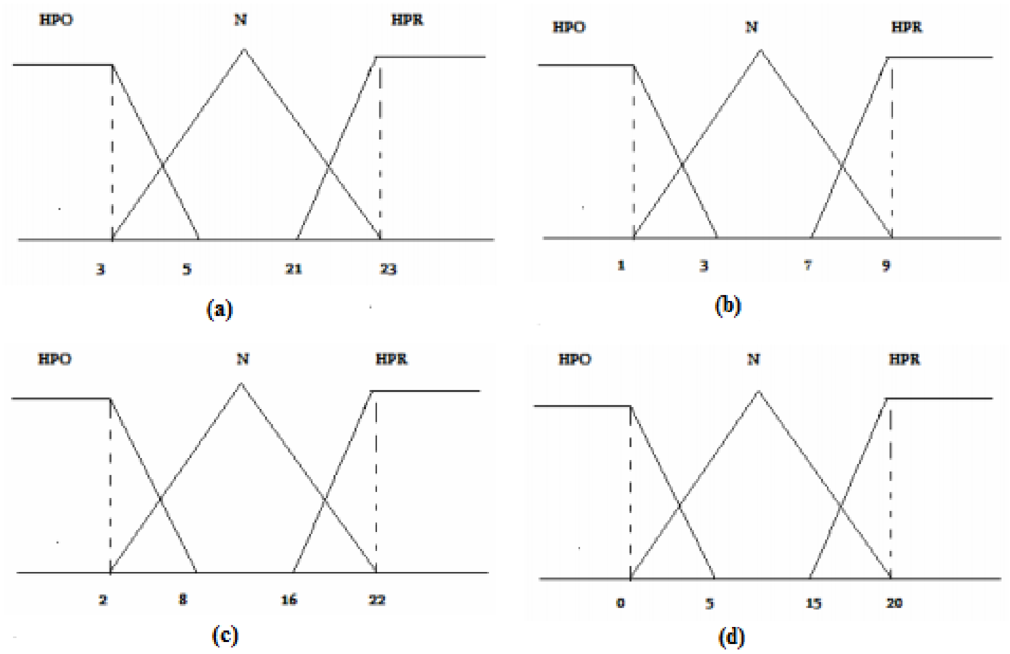
16. If the O_i object continues to be in more than one fuzzy incomplete equivalence class in the fuzzy incomplete upper approximation, the maximum plausibility measured equivalence class is used to estimate the value of the object. The estimating and process are carried out as described in Step (13a).

17. Due to estimated objects, create possible fuzzy rules from fuzzy incomplete upper approximations for each recalculated B subset with plausibility value, and determine the effectiveness of the rule according to the membership values of the equivalence classes in upper approximation.
18. From the obtained possible fuzzy rules, delete those whose condition parts are more specialized and those whose measures of both effectiveness and plausibility are equal to or less than other possible fuzzy rules or certain fuzzy rules.
19. Display obtained rules.

5. Experimental Results

In this section, the algorithm that was developed was tested on thyroid disease data that were obtained from UCI Machine Learning Repository (Dua & Graff, 2019), and the resulting rules were evaluated. A total of 100 thyroid disease data were taken for the experimental study. This dataset consisted of one decision attribute and four condition attributes. The decision attribute received three different values, and was classified according to its value (1 = normal thyroid, 2 = hyper thyroid, 3 = hypo thyroid). There were four condition attributes, which are Total Serum Thyroxine amount (TST), Total Serum Triiodothyronine amount (TSTR), Basal Thyroid Stimulating Hormone amount (TSH), and the amount of maximum absolute difference of TSH (DTSH). These condition attributes refer to the quantitative values of the hormone amounts that have an effect on the disease. Since the developed algorithm works on incomplete data, “do not care” or “lost” attribute values, these were inserted in the thyroid data in random attribute values. Then, the data were made fuzzy by using the membership functions given in Figure 1. Normal thyroid was expressed as ‘N’, Hyper thyroid as ‘HPR’, and Hypo thyroid as ‘HPO’ in membership functions. Firstly, the elementary sets, and then the lower and upper approximations were calculated by increasing the number of attributes in each iteration. Based on the resulting approximations, the “do not care” and “lost” attribute values were estimated. According to the approximations obtained in each iteration, certain rules were extracted from lower approximations, and possible rules from upper approximations. Finally, the rules whose left side is more specialized, or those whose left side is the same and whose measure of effectiveness and plausibility is equal to or lower than other rules, were deleted.

Figure 1: Membership functions (a) Membership functions for TST attribute (b) Membership functions for TSTR attribute (c) Membership functions for TSH attribute (d) Membership functions for DTSH qualification



The algorithm was tested for six different datasets that were obtained by changing the numbers and types of missing attribute values in the dataset. The features for these datasets are given in Table 2.

Table 2: Features of the tested datasets

Datasets	***"do not care" attribute value count***	***"lost" attribute value count***
DS1	11	5
DS2	16	0
DS3	0	16
DS4	22	5
DS5	11	10
DS6	22	10

The certain and possible rules for DS1 dataset are listed in Figure 2. The f value refers to the effectiveness measure of the rule, and the p value refers to the plausibility measure of the rule.

Figure 2: Rules obtained from DS1

Certain Rules		
1. IF TST = HPR THEN CLASS = HPR	f= 0,25	
2. IF TSTR = N THEN CLASS = HPR	f= 0,05	
3. IF TSH = N THEN CLASS = HPO	f= 1	
4. IF DTSH = HPO THEN CLASS = HPO	f= 0,8	
5. IF TST = HPO AND TSH = HPR THEN CLASS = HPO	f= 1	
6. IF TST = HPO AND DTSH = HPR THEN CLASS = HPO	f= 1	
7. IF TSH = HPR AND DTSH = HPO THEN CLASS = N	f= 0,46	
8. IF TSH = HPR AND DTSH = HPR THEN CLASS = HPO	f= 1	
9. IF TSTR = N AND TSH = N AND DTSH = HPO THEN CLASS = N	f= 0,075	
10. IF TST = N AND TSH = N AND DTSH = HPO THEN CLASS = N	f= 0,41	
Possible Rules		
1. IF TST = N THEN CLASS = N	f= 0,04	p = 0,550
2. IF TST = HPO THEN CLASS = HPO	f= 0,15	p = 0,915
3. IF TSTR = N THEN CLASS = HPR	f= 0,025	p = 0,603
4. IF TSTR = HPR THEN CLASS = HPR	f= 0,05	p = 0,646
5. IF TSTR = HPO THEN CLASS = HPO	f= 0,15	p = 0,478
6. IF TSH = HPO THEN CLASS = N	f= 0,08	p = 0,479
7. IF TSH = N THEN CLASS = HPO	f= 0,01	p = 0,900
8. IF TSH = HPR THEN CLASS = HPO	f= 1	p = 0,554
9. IF DTSH = N THEN CLASS = N	f= 0,01	p = 0,462
10. IF DTSH = HPO THEN CLASS = HPR	f= 0,06	p = 0,520
11. IF DTSH = HPR THEN CLASS = HPO	f= 0,4	p = 0,907

Table 3: Comparison of rules obtained from datasets

	Certain rule count	Possible rule count	Certain rule count after deletion	Possible rule count after deletion	Possible rule count consistent with certain rules	Possible rule count inconsistent with certain rules	Newly found possible rule count	Rate of consistent rules	Rate of inconsistent rules	Rate of new rules	Success rate
DS1	70	239	10	11	6	1	4	%55	%9	%36	%91
DS2	75	255	12	12	6	2	4	%50	%17	%33	%83
DS3	80	251	23	11	4	7	0	%36	%64	%0	%36
DS4	85	263	13	12	9	2	1	%75	%17	%8	%83
DS5	65	257	10	11	5	2	4	%46	%18	%36	%82
DS6	75	287	13	12	9	2	1	%75	%17	%8	%83

The rules obtained by running the algorithm on six different datasets whose features are given in Table 2 were evaluated, and the results are given comparatively in Table 3. The certain and possible number of rules obtained from the datasets is shown in columns 1 and 2 in the table. The rules whose left sides were more specialized or the ones with the same left side, which had equal or lower effectiveness or plausibility value, were identified and deleted. The remaining number of certain and possible rules is shown in columns 3 and 4 in the table. After these rules were examined, the number of possible rules that were consistent with certain rules, those that were inconsistent and those that were newly found (i.e. different) were determined, and are given in columns 5, 6, and 7, respectively in the table. The rate of possible rules that were consistent, inconsistent and new are shown in columns 8, 9, and 10 in the table, respectively, and the possible rule success rate for each dataset is shown in the last column. When the rules obtained from DS1 shown in Figure 2 were examined, it was seen that the 10th possible rule was inconsistent with the certain rules, the possible rules 4, 5, 6, and 9 were new rules, and the remaining rules were consistent with the certain rules.

When Table 3 is examined, it is seen that the inconsistent rule rate increased and the rate of consistent and new rules decreased in DS2, where there was only "do not care" missing value. In DS3, in which all the missing values were "lost", no new rules were obtained, consistent rule rate decreased and the highest inconsistent rule rate was achieved compared to the other datasets. The consistent and inconsistent rule rate increased and the new rule rate decreased in DS4, which contained twice as much "do not care" missing data compared to DS1. The consistent rule rate decreased, the inconsistent rule rate increased, and the new rule rate remained the same in DS5, which contained twice as much "lost" data as DS1. Consistent and inconsistent rule rates increased and the new rule rate decreased in DS6, which contained twice as much "do not care" and "lost" data compared to DS1. The highest success rate was achieved in DS1, and the lowest one in DS3. The same success rate (83%) was achieved in DS2, DS4, and DS6 in which the number of "do not care" missing attributes was increased. In DS5, in which the number of "lost" attributes was increased, a very close success rate (82%) was achieved to the rate of the datasets that were obtained by increasing the number of "do not care" attributes.

According to these results, it can be argued that the algorithm increases the inconsistent rule rates in the datasets which only have the "lost" type of missing attribute values and decreases the rate of consistent and new rules. It was determined that the increase in the number of "do not care" missing data causes increased rates of inconsistent rules and decreased rates of new rules. "Lost" missing values are those that are required but somehow unreachable. For this reason, the increase in such missing values in the dataset resulted in decreased performance. Although the increase in "do not care" values did not affect performance as much as the increase in "lost" values, it increased the rate of inconsistent rules. For this reason, it caused a decreased success rates.

6. CONCLUSIONS

Incomplete data are found in many datasets in the present day. One of the techniques that can be used to extract rules from incomplete data is the Rough Set Theory. There might be missing values with different features in a dataset. Changes in the type and number of missing attribute values affect the accuracy of the rules to be obtained. With the algorithm that was developed in this study, certain and possible rules were obtained with six different datasets that had different numbers and types of missing attribute values. The algorithm generally produced successful and consistent rules in the datasets that had "do not care" and "lost" attribute values. The increase in the number of missing attribute values in the datasets also reduced the possible rule success of datasets, which was more significant only in the dataset with "lost" missing attribute values. Further studies can be conducted to increase the success of the algorithm in datasets that have "lost" attribute values.

References

Abdel-Basset, M., & Mohamed, M. (2018). The role of single valued neutrosophic sets and rough sets in smart city: Imperfect and incomplete information systems. *Measurement*, *124*, 47-55. https://doi.org/https://doi.org/10.1016/j.measurement.2018.04.001

Cekik, R., & Uysal, A. K. (2020). A novel filter feature selection method using rough set for short text data. *Expert Systems with Applications*, *160*, 113691.

Çekik, R., & Telçeken, S. (2018). New method based on rough set for filling missing value. In *Recent Developments and the New Direction in Soft-Computing Foundations and Applications* (pp. 41-48). Springer.

Dua, D., & Graff, C. (2019). UCI Machine Learning Repository [http://archive. ics. uci. edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, zuletzt abgerufen am: 14.09. 2019. *Google Scholar*.

Grzymala-Busse, J. W. (2006). Rough set strategies to data with missing attribute values. In *Foundations and novel approaches in data mining* (pp. 197-212). Springer.

Grzymala-Busse, J. W. (2008). Three approaches to missing attribute values: A rough set perspective. In *Data mining: Foundations and practice* (pp. 139-152). Springer.

Hong, T.-P., Tseng, L.-H., & Chien, B.-C. (2002). Learning fuzzy rules from incomplete quantitative data by rough sets. 2002 IEEE World Congress on Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems. FUZZ-IEEE'02. Proceedings (Cat. No. 02CH37291),

Jayasuruthi, L., Shalini, A., & Kumar, V. V. (2018). Application of rough set theory in data mining market analysis using rough sets data explorer. *Journal of Computational and Theoretical Nanoscience*, *15*(6-7), 2126-2130.

Jia, W., Yongkang, Z., Jianchuan, H., Chao, P., Zemin, L., Bangze, Z., & Deli, Z. (2021). Infrared Image Denoising Algorithm Based on a Rough Set Approach. *红外技术 43*(1), 44-50.

Luo, C., Li, T., & Yao, Y. (2017). Dynamic probabilistic rough sets with incomplete data. *Information Sciences*, *417*, 39-54. https://doi.org/https://doi.org/10.1016/j.ins.2017.06.040

Luo, J., Fujita, H., Yao, Y., & Qin, K. (2020). On modeling similarity and three-way decision under incomplete information in rough set theory. *Knowledge-Based Systems*, *191*, 105251. https://doi.org/https://doi.org/10.1016/j.knosys.2019.105251

Nabwey, H. A. (2020). An intelligent mining model for medical diagnosis of heart disease based on rough set data analysis. *International Journal of Engineering Research and Technology*, *13*(2), 355-363.

Nakata, M., Sakai, H., & Hara, K. (2020, 2020//). Rough Sets Based on Possibly Indiscernible Classes in Incomplete Information Tables with Continuous Values. Proceedings of the

International Conference on Advanced Intelligent Systems and Informatics 2019, Cham.

Pawlak, Z. (1982). Rough sets. *International journal of computer & information sciences*, *11*(5), 341-356.

Pawlak, Z. (2012). *Rough sets: Theoretical aspects of reasoning about data* (Vol. 9). Springer Science & Business Media.

Walczak, B., & Massart, D. L. (1999). Rough sets theory. *Chemometrics and Intelligent Laboratory Systems*, *47*(1), 1-16. https://doi.org/https://doi.org/10.1016/S0169-7439(98)00200-7

Wu, B., Nazir, S., & Mukhtar, N. (2020). Identification of Attack on Data Packets Using Rough Set Approach to Secure End to End Communication. *Complexity*, *2020*, 6690569. https://doi.org/10.1155/2020/6690569