Inference algorithm for knowledge bases with rule cluster structure

Agnieszka Nowak - Brzezińska[1[0000−0001−7238−1170]] and Igor Gaibei[1[0000−0002−4708−9036]]

Institute of Computer Science, Faculty of Science and Technology,
University of Silesia, Bankowa 12, Katowice 40-007 , Poland;
agnieszka.nowak-brzezinska@us.edu.pl

Abstract. This paper presents an inference algorithm for knowledge bases with a rule cluster structure. The research includes the study of the efficiency of inference, measured by the number of cases in which the inference was successful. Finding a rule whose premises are true and activating it leads to extracting new knowledge and adding it as a fact to the knowledge base. We aim to check which clustering and inference parameters influence the inference efficiency. We used four various real datasets in our experimental stage. Overall, we proceeded with almost twenty thousand experiments. The results prove that the clustering algorithm, the amount of input data, the method of cluster representation, and the subject of clustering significantly impact the inference efficiency.

Keywords: inference algorithm · rule-based knowledge bases · clustering algorithms

1 Introduction

Expert systems are a significant artificial intelligence (AI) branch that has been developed for several decades. Recently, we have seen a lot of movement in this field. Many applications are being developed to have the role of an assistant with access to knowledge and share this knowledge with the user. Such an assistant is often simply an expert system with built-in expert knowledge and implemented one of two inference algorithms: forward and backward. Expert knowledge is often stored in the form of IF-THEN rules, and classical inference requires analysis of each rule, one by one, to assess which one can activate since all premises are true. When there are a lot of such rules, a reasoning process take a very long time, which may not be acceptable to the user who needs information without a delay. Our concept is based on the idea that we will cluster similar rules into groups and assign representatives to these rule clusters. These representations will later be reviewed in the inference process. The paper presents the following topics: rule clustering algorithms, comparing classical inference with inference which operates on rule cluster representations, studying the efficiency of inference (measured by the number of cases with successful inference), assessing the impact of various parameters on this efficiency.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

1.1 Literature review

In the literature, one can find many articles on the analysis and application of inference algorithms, analyzing expert systems based on the knowledge representations, the comparison of the *K - Means* and the *AHC* algorithm, the use of different distance measures, methods of combining clusters or methods of analyzing the quality of clustering. In [1], the authors present a rule-based forward chaining system that decides the measurement category. In [2], the authors propose an expert system that helps cyclists decide whether to fix the issue themselves or search for an expert. The paper [3] presents an expert system built using the forward chaining method for rules *IF-THEN*, but not for rule group representations. The paper [4] presents the design of an expert system based on a forward inference for plant disease identification. [5] presents a project designed for medical diagnosis based on the provided symptoms. The expert system performs inference using *IF-THEN* production rules. In [6], the authors evaluate the effectiveness of various clustering methods. In [7], authors compare and assess five clustering algorithms, however, not for rule-based knowledge representation. The authors of [8] compare *K-Means* and *AHC* algorithms in terms of the number of clusters, the number of objects in clusters, the number of iterations, and clustering time for small and large data. In [9], the authors compare clustering algorithms and methods of cluster quality assessment (F-measure, Entropy) for different values of the number of clusters. In our previous work [10], we examined the *K-Means* and the *AHC* clustering algorithm in the context of rule-based knowledge representation. Although, it is impossible to find papers that would combine these issues in one study. We wanted to investigate which algorithm (*AHC* or *K-Means* algorithm) and method of creating a cluster representative (mean or median) is more effective in terms of the efficiency of inference on clusters.

1.2 Article structure

Section 2 describes the clustering algorithms used in this research. The inference algorithm is presented in Section 3. The procedure for creating rule clusters and representatives of groups is presented in Section 4. Then, in Section 5, the description of the experiments was included with the results and their analysis. Section 6 contains the summary of the research.

2 Clustering algorithms

This part presents a concise description of clustering algorithms applied to the rules in the knowledge base. Among the available clustering techniques, *non-hierarchical* and *hierarchical* methods can be used. When we have a lot of rules in the knowledge base, then, unfortunately, the efficiency of an inference decreases because the inference time increases and the system user has problems interpreting too many newly generated facts. Rules can be grouped, and we only

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

need to search the representatives of rule clusters and select the cluster that best matches the given facts. Two clustering algorithms were used in our work: the non-hierarchical algorithm (*K-Means*) with the hierarchical algorithm *AHC*. Both are presented in the following subsections.

2.1 Partitional *K-Means* clustering algorithm

K-Means clustering algorithm tries to group similar rules in the form of K clusters. Each cluster is represented by its center (arithmetic mean of data points). In each iteration we try to divide N original rules into K rule clusters so well that each rule belongs to the cluster to which it is most similar. The main idea of the algorithm is as follows:

1. Select the number of rule clusters (K) and assign K centers.
2. For each rule, the nearest cluster center is determined.
3. The rule cluster representative is created - an average value for each attribute value (conditional and decisional) or a mode value for qualitative attributes.
4. New cluster center is formed.
5. The 3rd and 4th steps repeat iteratively.
6. The algorithm ends when no rule cluster changes occur at some iteration.

2.2 Hierarchical *AHC* clustering algorithm

AHC hierarchical clustering algorithm for rules creates a tree of rule clusters. The *AHC* algorithm works as follows:

1. Each rule forms a separate cluster. We must calculate the distance between each pair of rule clusters.
2. Find and join the two most similar rule clusters.
3. Repeat the second step until obtaining the declared final number of rule clusters (K) or combining all rules into one big cluster.

Each algorithm will create an entirely different structure of the focus of rules.

3 Forward chaining inference

There are two inference algorithms: forward chaining (from premises to conclusions) and backward chaining (from hypothesis/conclusion to premises). In this work, we have only dealt with the first method. Each rule is analyzed to determine whether all its' premisses are satisfied. The rule is activated, and its conclusion is added to the set of facts. The algorithm stops where no more rules can be activated or when the starting hypothesis is added to the set of facts. The more rules, the longer the inference time. Therefore, we aim to cluster similar rules, hoping we reduce the inference time significantly when we divide such a large rule set into clusters of similar rules. This is possible because we only need to search the representatives of rule clusters and select the cluster that best

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

matches the given facts. The rule cluster representatives are achieved as follows: at the cluster level, we calculate the similarity of the representative of each cluster to the fact vector, using Euclidean distance to calculate the similarity. We select the most promising cluster. Within the selected cluster of rules, we calculate the similarity of facts to each rule. We then apply the forward inference algorithm only on the selected cluster. The next step is to look within the loop for rules to fire. If more than one rule could be fired at a time, we chose one of them using the appropriate rule selection strategy. In our experiments we fire the first rule in the list of all rules to be fired (sequential strategy). Firing (activation) of a rule leads to adding the conclusion of such rule to the fact base and blocking this rule from being activated again. The inference ends when there are no more rules to be fired or the set of facts (after the last activation) includes the inference target set at the beginning.

3.1 Group representative: Methods of creating a group representative

In our experiments, we used two methods for creating a representative of a group of rules - the mean and the median. Our goal is to verify which method provides better inference efficiency. Figure 1 shows an example of a hierarchical algorithm where the choice of a rule cluster depends on the similarity of the fact representative to the cluster representative. As we can see, in some cases, the mean or median method can lead to entirely different branches of the binary tree. Therefore, it is crucial to study the influence of the cluster representative on the efficiency of inference.

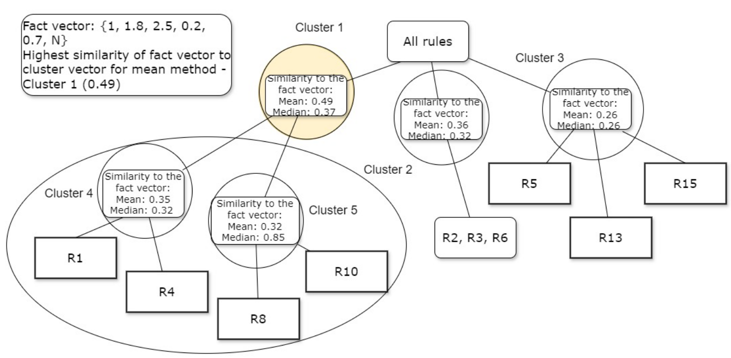

Fig. 1. Binary tree of rule clusters

To find the median value for a rule cluster representative with an odd number of numbers, one would find the number in the middle with an equal number of numbers on either side of the median. To find the median, we should first arrange the numbers, usually from lowest to highest. Mean is the average of

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

all the numbers within each group of rules. Therefore, to represent the rules in vector format, we should use the values of the attributes of each rule and zero out the attributes that have no value and are not considered in the rule. This is an essential operation for the correct calculation of vector similarity. It may turn out that the use of each of these representation methods can finally lead us to different results. We want to examine it in this work. We will perform the same experiments for both methods and compare the results in the experimental part.

4 Methodology

The experiments proceeded as follows. The source dataset was loaded into the *RSES* tool, where decision rules were generated using the *LEM 2* algorithm. The rules are then loaded into the Python environment in the customized software and were grouped using two clustering algorithms: *K-Means* and *AHC*. We tested different distance measures, different clustering methods (for the *AHC* algorithm), and different values for the parameter representing the number of rule groups created, different numbers of generated facts (inputs), and different methods for creating a representative. We studied the clustering time and cluster quality indices, the *Dunn* and the *Davies-Bouldin* indexes. The next step was to calculate the vector of initial facts. In our experiments, we generate the initial facts that constitute 5%, 25%, and 50% of all unique descriptors in the premises for each algorithm. We used two methods for creating a representative of a group of rules - the mean and the median. The next step was to calculate the representations of all rule clusters. Let us assume that the example rule cluster contains the foolowing rules: $R1$: IF $a1 = 1$ THEN $dec = T$, $R4$: IF $a1 = 2$ THEN $dec = T$, $R8$: IF $a2 = 2$ THEN $dec = T$ and $R10$: IF $a3 = 3$ THEN $dec = T$. If the number of attributes equals 5 rule vectors are as follows: $[1, 0, 0, 0, 0, T]$, $[2, 0, 0, 0, 0, T]$, $[0, 2, 0, 0, 0, T]$, $[0, 0, 3, 0, 0, T]$ and the rule cluster representative is the following *Representative mean* = {1.5 2 3 0 0 T} or *Representative median*= {1 1 0 0 0 T}.

5 Results of experiments

This section presents the course of experiments and the analysis of the results achieved. In the experiments, we included real knowledge bases with different structures. There were the following datasets: kb_1 with 4435 instances, 37 attributes and 937 rules [11], kb_2 with 7027 instances, 65 attributes and 4125 rules [12], kb_3 with 527 instances, 38 attributes and 123 rules [13] and kb_4 with 17898 instances, 9 attributes, 6432 rules [14]. The runtime for the experiments had the following configuration: Spyder compiler with *Python* version 3.9 from the Anaconda platform. The computer parameters on which all experiments were carried out are as follows: Intel Core i5-7500K, 16 Gb RAM. To run the experiments, we used self-written software in the Python programming language. The following libraries were used: *Pandas* for data processing and analysis and *NumPy* for basic operations on n-arrays and matrices. Finally, we used the *RSES* system and

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

the $LEM2$ algorithm to generate rules, although we also checked the *exhaustive* algorithm.

5.1 Experiments procedure

We perform clustering sequentially for each algorithm (*K-Means, AHC*) for $K = 2, 3, \ldots, 22$, using one of the three distance measures (Euclidean, Chebyshev, Manhattan). For the *AHC*, we repeated the algorithm for single, complete, and average linkages. We repeat each algorithm for three different inputs: the conditions alone, the conclusions alone, and the conditions and conclusions of the rules together. For each algorithm, we generate the initial facts constituting 5%, 25%, and 50% of all unique descriptors in the premises. We used two methods for creating a representative of a group of rules - the mean and the median. For four knowledge bases, this gives a total of $18,144$ experiments.

5.2 Results

This section presents the results of selected experiments. We decided to examine whether the following parameters affect more or less inference efficiency, and thus the frequency of successful inference process: selected clustering algorithm (*K-Means* or *AHC*), number of input facts (5%, 25%, and 50%), what was clustered (premises, conclusions, or both), rule cluster representative method (mean, median). Table 1 presents the inference efficiency achieved for two clustering algorithms. You can see much greater effectiveness of the inference process when

Table 1. Inference efficiency vs. clustering algorithms

	Succeed	Failed
$K - Means$	3618 (79.76%)	918 (20.24%)
AHC	7291 (53.58%)	6317 (46.42%)
sum	10909 (60.12%)	7235 (39.88%)

using the *K-Means* method. This may be surprising at first glance. Generally, the *AHC* method clusters the objects naturally by connecting the most similar pair of rules (or rule clusters) in each algorithm iteration. But it is a structure that is further searched as a binary tree, selecting only one child node from two given nodes at every level. It can cause cases when we wrongly choose the node for further searching. In the case of the *K-Means* algorithm, we achieve a flat rule cluster structure, which means that we search every rule in a selected cluster. Therefore, the chance that we will wrongly select a proper rule cluster (and then a proper rule) is much smaller than in the case of the *AHC* algorithm. In the case of the *AHC* algorithm, the average time is $O(log2N)$, while in the case of the *K-Means*, it is linear $O(K) + O(N/K)$[1], which generally lasts longer than the *AHC* searching time. The results presented in Table 2 show the effectiveness

[1] N is the rule number and K is the number of rule clusters.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

Table 2. Inference efficiency vs. clustering algorithms and the amount of input facts

	% of facts	Succeed	Failed	clustering object	Succeed	Failed
	5%	73.61%	26.39%	cond	72.29%	27.71%
$K - Means$	25%	71.63%	28.37%	dec	90.15%	9.85%
	50%	94.05%	5.95%	cond + dec	76.85%	23.15%
	5%	35.45%	64.55%	cond	70.26%	29.74%
AHC	25%	54.94%	45.06%	dec	16.40%	83.60%
	50%	70.35%	29.65%	cond + dec	74.07%	25.93%

of the proposed idea, except that it depends on how the rules were grouped (what algorithm); however, how many input facts were used is crucial. It can be seen that as the number of input facts increases, the effectiveness of inference increases. In the *K-Means* algorithm, with 50% of facts, the effectiveness of inference is about 95%, while for the same number of facts, the *AHC* algorithm is only 70%.

Table 3. Inference efficiency vs. clustering algorithms and representative method

	representative method	Succeed	Failed
$K - Means$	mean	1849 (81.53%)	419 (18.47%)
	median	1769 (78.00%)	499 (22.00%)
AHC	mean	3672 (53.97%)	3132 (46.03%)
	median	3619 (53.19%)	3185 (46.81%)

Table 3 showed that the grouping algorithm and the selected method of creating a representative (out of two proposed methods of concentration representation: mean and median) significantly impact the inference's effectiveness. We know that using the *K-Means* algorithm is successful in about 80% of cases. Still, when we consider the representation method, you can see that it will be more effective than the median when the representative is created using the mean method. The *K-Means* algorithm will behave entirely differently than *AHC*. When we use the *K-Means* algorithm and we only cluster by rule decisions (so large groups of rules are created), the inference is just over 90% effectiveness; when we group by the rule conditions, this efficiency is over 72%, and when we cluster. For *AHC*, it is entirely different. When we only group under decisions, the effectiveness of inference is only 16.4%. At the same time, clustering by rule conditions results in achieving efficiency at over 70%, and clustering by both conditions and decisions brings the best results, 74%, respectively.

6 Summary

This paper presents an inference algorithm for knowledge bases with a rule cluster structure. The research includes the study of the efficiency of inference, which

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9

will be measured by the number of cases in which the inference was successful. We aim to check which clustering and inference parameters influence the inference efficiency. We proceeded with almost twenty thousand experiments. The results prove that the clustering algorithm, the amount of input data, the method of cluster representation, and the subject of clustering significantly impact the inference efficiency. We observed a much greater effectiveness of the inference process when using the *K-Means* method compared to *AHC*. As the number of input facts increases, the effectiveness of inference increases.

References

1. D. Wang and Y. Bai, Data Processing and Information Retrieval of Atmospheric Measurements, 2022 IEEE, CIVEMSA, Germany, 2022, pp. 1-5, 10.1109/CIVEMSA53371.2022.9853650.
2. Hernández, M.P., et. (eds) Advances in Computational Intelligence. MICAI 2023, LNCS vol 14502. Springer, Cham. https://doi.org/0.1007/978-3-031-51940-6_8.
3. H. Mustafidah, B. R. Alfiansyah, Suwarsito, Purnomo and N. Hidayat, Expert System Using Forward Chaining to Determine Freshwater Fish Types Based on Water Quality and Area Conditions, ICIC, Indonesia, 2023, pp. 1-5, 10.1109/ICIC60109.2023.10382066.
4. Delima Sitanggang et. Application of forward chaining method to diagnosis of onion plant diseases, 2018 J. Phys.: Conf. Ser. 1007 012048, 10.1088/1742-6596/1007/1/012048.
5. Jabeen S.H., Zhai G. A prototype design for medical diagnosis by an expert system, WCSE 2017, pp. 1413 - 1417.
6. R. K. M.V.N.M, V. Sharma, K. Gupta, A. Jain, B. Priya and M. S. R. Prasad, Performance Evaluation and Comparison of Clustering Algorithms for Social Network Dataset, IC3I, India, 2023, pp. 111-117, 10.1109/IC3I59117.2023.10397806.
7. D. Teslenko, A. Sorokina, K. Smelyakov and O. Filipov, Comparative Analysis of the Applicability of Five Clustering Algorithms for Market Segmentatio, IEEE Open Conference of Electrical, Electronic and Information Sciences, Lithuania, 2023, pp. 1-6, 10.1109/eStream59056.2023.10134796.
8. Saleena, T.S.; Sathish, A.J. Comparison of K-Means Algorithm and Hierarchical Algorithm using Weka Tool. Int. J. Adv. Res. Comput. Commun. Eng. 2018, 7, 74–79.
9. Steinbach M.S., Karypis G. and Kumar V.: A Comparison of Document Clustering Techniques, Department of Computer Science and Egineering, Computer Science, (2000).
10. Nowak-Brzezińska, A., Gaibei, I. (2023). Decision Rule Clustering—Comparison of the Algorithms. IJCRS 2023. LNCS vol 14481. Springer, Cham. https://doi.org/10.1007/978-3-031-50959-9_27.
11. https://archive-beta.ics.uci.edu/dataset/146/statlog+landsat+satellite, [access December 2023]
12. https://archive-beta.ics.uci.edu/dataset/365/polish+companies+bankruptcy+data, [access December 2023]
13. https://archive-beta.ics.uci.edu/dataset/106/water+treatment+plant, [access December 2023]
14. https://archive-beta.ics.uci.edu/dataset/372/htru2, [access December 2023]

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63759-9_9