USING ENSEMBLE AND LEARNING TECHNIQUES TOWARDS EXTENDING THE KNOWLEDGE DISCOVERY PIPELINE

Yu-N Cheah, Sakthiaseelan Karthigasoo, Selvakumar Manickam
School of Computer Sciences, Universiti Sains Malaysia, 11800 USM Penang, Penang, Malaysia

Keywords: Knowledge discovery, Clustering ensemble, Neural network ensemble, Discretization, Rough set analysis

Abstract: Knowledge discovery presents itself as a very useful technique to transform enterprise data into actionable knowledge. However, their effectiveness is limited in view that it is difficult to develop a knowledge discovery pipeline that is suited for all types of datasets. Moreover, it is difficult to select the best possible algorithm for each stage of the pipeline. In this paper, we define (a) a novel clustering ensemble algorithm based on self-organizing maps to automate the annotation of un-annotated medical datasets; (b) a data discretization algorithm based on Boolean Reasoning to discretize continuous data values; (c) a rule filtering mechanism; and (d) to extend the regular knowledge discovery process by including a learning mechanism based on neural network ensembles to produce a neural knowledge base for decision support. We believe that this would result in a decision support system that is tolerant towards ambiguous queries, e.g. with incomplete inputs. We also believe that the boosting and aggregating features of ensemble techniques would help to compensate for any shortcomings in some stages of the pipeline. Ultimately, we combine these efforts to produce an extended knowledge discovery pipeline.

1 INTRODUCTION

The generation of a huge amount of data by an enterprise is of great concern to decision makers. This problem is compounded by the many environmental challenges that an enterprise faces in the effort to produce better products and services. It is highly crucial to know what goes on in its business transactions both internally and externally and to examine the heart of an enterprise's transactions, that is its data, and to transform it into actionable knowledge through the process of knowledge discovery.

Knowledge discovery is a series of processes, which can be likened to a pipeline to find hidden but potentially useful information and patterns in data (Fayyad et al., 1996). This series of processes involves preparation on the data, the application of data mining algorithms on the data and finally the interpretation and/or visualization of the data mining results. The number of ways that a pipeline can be developed is perhaps limitless in view that there are many ways in which data preparation, data mining and interpretation or visualization can be achieved.

Current knowledge discovery pipelines have been proven effective to a certain extent in discovering hidden knowledge from various datasets. However, their effectiveness is limited as it is difficult to develop a pipeline that is suited for all types of datasets. Moreover, it is difficult to select the best possible algorithm for each stage of the pipeline. It is challenging to develop an approach that would minimize the impact of a sub-optimal choice of algorithm at each stage of the pipeline.

Therefore, in this paper, we aim to define (a) a novel clustering ensemble algorithm based on self-organizing maps (SOM) to automate the annotation of un-annotated medical datasets; (b) a data discretization algorithm based on Boolean Reasoning to discretize continuous data values; (c) a rule filtering mechanism employing a Rule Quality Function; and (d) to extend the regular knowledge discovery process by including a learning mechanism based on neural network ensembles (NNE) to produce a neural knowledge base for decision support. We believe it is advantageous to produce a knowledge base that is trained from decision rules to develop a decision support system that is tolerant towards ambiguous queries, e.g. with incomplete inputs. We also believe that the boosting and aggregating features of ensemble techniques would help to compensate for any shortcomings in some stages of the pipeline. Ultimately, we combine

Cheah Y., Karthigasoo S. and Manickam S. (2005).
USING ENSEMBLE AND LEARNING TECHNIQUES TOWARDS EXTENDING THE KNOWLEDGE DISCOVERY PIPELINE.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems*, pages 408-411
DOI: 10.5220/0002538104080411

these efforts to produce an extended knowledge discovery pipeline.

1.1 Ensembling Techniques

Ensembling is a technique that harnesses the capabilities of a predetermined number of algorithms or processes that perform the same task to obtain an improved result.

Ensembling involves two main steps. The first is the execution of the individual algorithm. Two popular approaches for this are boosting (Freund and Schapire, 1995) and bagging (bootstrap aggregation) (Breiman, 1996). In boosting, the output of a particular individual algorithm becomes the input for another algorithm. The outputs are boosted through re-sampling or re-weighting until all the algorithms have had a hand in processing the input. In bagging, several subsets are derived from the original input and each one is fed to a different algorithm to be processed separately. The second step involves the combination of the outputs to produce a single consolidated output, i.e. as if only a single instance of the algorithm was used. Techniques such as voting and averaging are popularly used for this purpose.

An example of an ensembling technique is the NNE (Hansen and Salamon, 1990) where the learning capabilities of a predetermined number of neural networks is utilized to obtain improved generalizations or predictions (Zhou, et al., 2003). Another example of the ensembling technique is in the area of clustering (Yang and Kamel, 2003).

1.2 Knowledge Discovery Pipelines

An example in developing a knowledge discovery pipeline for symbolic rules extraction from un-annotated datasets is by Abidi and Hoe (2002) which applies rough set analysis. This pipeline, or workbench as they have called it, includes steps to pre-process and cluster un-annotated datasets resulting in annotated versions of the original datasets. The workbench then proceeds to discretize the data and generate rules. These rules are finally filtered.

For clustering, the workbench employs the K-Means algorithm while the Chi Squared and Entropy-MDL algorithms were used to discretize the annotated version of the data. Rough set analysis was then used to compute the reducts and generate symbolic rules. Here, the reducts were derived using genetic algorithm. The rules were then filtered using a rule quality index computation.

Another work on a knowledge discovery pipeline involves the discretization of numerical attributes for machine learning (Risvik, 1997). This pipeline includes steps to pre-process and then discretize annotated datasets using discretization algorithms such as Chi Squared, Entropy, Naïve and Orthogonal Hyperlanes. Rough set analysis was used to generate rules from the discretized data. Reducts is generated using Johnson's Algorithm (1974) and followed by a rule generation process.

From our observation, the stages involved in knowledge discovery are more or less standardised and normally ends with a filtering stage. It appears that much of the research in knowledge discovery pipelines involves the exploration of various algorithms that can be applied at each stage of the pipeline. We believe that existing pipeline construction can be extended further by including a learning stage and that the algorithms can be made more effective by employing ensemble techniques, i.e. by being made to 'work harder' to compensate for any shortcomings as mentioned earlier.

2 THE EXTENDED KNOWLEDGE DISCOVERY PIPELINE

Our extended knowledge discovery pipeline consists of six stages (see Figure 1): data preparation, clustering, discretising, rough set analysis, rule filtering and learning.

2.1 Stage I: Data Preparation

Data preparation ensures the cleanliness or completeness of the dataset. Various strategies are currently employed to remove redundant data, missing values and other errors. For our purposes, we use the mean/mode fill technique to address the issues of missing values. We observed that mean/mode fill produces better results when compared to other data cleansing techniques such as combinational completion in terms of accuracy in

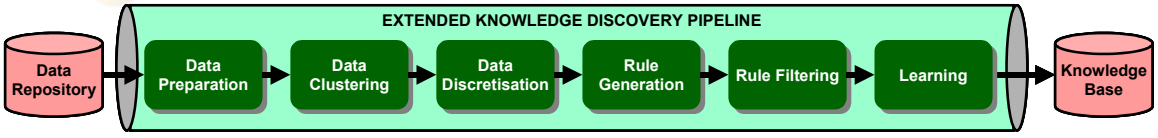

Figure 1: The extended knowledge discovery pipeline

view that the mean/mode fill substitutes missing values with the mean value for numerical attributes of all observed entries for that attribute. For string attributes, missing values are substituted by the mode value, i.e. the most frequently occurring value among the observed entries for that attribute. In contrast, combinational completion is perceived to be more complex in view that it expands each missing value for each object into the set of possible values. That is, an object is expanded into several objects covering all possible combinations of the object's missing values. This could lead to the number of possible combination for objects with multiple missing values to grow very rapidly.

2.2 Stage II: Data Clustering

With un-annotated datasets in mind, this next phase involves the utilization of a novel clustering ensemble algorithm to annotate the previously un-annotated dataset. This step involves the application of Kohonen's SOM as the basic technique for clustering the un-annotated dataset. It is not our aim to explore in detail any enhancements to SOMs. However, here, we aim to boost the results of the SOM clustering through re-sampling. Training instances that were wrongly predicted in the kth classifier will play a more important role in the $k+1$th classifier to produce optimum clustering results, i.e. the best possible annotation of the previously un-annotated dataset. We argue that the application of boosting and SOM techniques in a clustering mechanism is novel in view that it is more commonly applied within the context of other neural networks applications. Moreover, previous work on clustering ensembles focused more on using aggregation (Dimitriadou et al., 2003) and hypergraph partitioning (Strehl and Ghosh, 2002) strategies.

2.3 Stage III: Data Discretization

Following the clustering stage, we then discretize the annotated results. The objective of discretization is basically to clearly differentiate between continuous values that are likely to be present in the dataset. Continuous values that lie within each interval are then mapped to the same discrete value. This process would result in better rule generation. Here, we use Boolean Reasoning and we have observed that it best suits our purpose over other techniques that we have explored such as the Entropy/Minimum Description Length (MDL) technique. This is because Boolean Reasoning computes a larger interval range compared to Entropy/MDL. The interval range computed by Entropy/MDL is 0.1, e.g. 1.15-1.25. We believe that small interval range will not be sufficiently succinct and practical to discretize data as there will be very few datasets that are categorised in that interval. In this sense, Boolean Reasoning computes larger interval cuts which lead to a lower number of categories being computed and, hence, allow clearer rules to be generated in the next stage.

2.4 Stage IV: Rule Generation

For the data mining proper, we choose to carry out rule generation as the product of the pipeline based on rough set analysis. We compute reducts using genetic algorithm, a technique that is widely used for this purpose in rough set analysis. We then look into a number of sub-tables which will be randomly sampled from the datasets. Proper reducts are computed from each of these samples. The reducts that occur the most often across these sub-tables are in some sense the most stable and can be categorised as dynamic reducts. Based on the computation of the reducts, rules will be generated. This would constitute knowledge as it is potentially capable of capturing complex relationship between attributes and decision values of the dataset.

2.5 Stage V: Rule Filtering

This rule filtering phase follows the induction of rules. This stage basically cleanses the output of the rule generation stage to ensure that weak rules are removed. For this purpose, we define a quantitative rule evaluation technique leading towards the definition of what we call a Rule Quality Function which would indicate the quality of an induced rule. Here, the Rule Quality Function is based on the support, consistency and coverage measurements used by Michalski (1983). After determining the quality of the rules, the actual filtering process can then be carried out. We observed that statistical methods like the mean and Receiver Operating Characteristic curve (ROC) serves as a good indicator in the rule filtering. This can be viewed as an initial step towards ensuring that the rule base generated by the rule induction process using the rough set theory is of good quality.

2.6 Stage 6: Learning

The link between this learning phase and the rule filtering phase is novel where the filtered rules will be trained. We argue that the inclusion of this learning stage to the overall knowledge discovery pipeline is novel in view that most pipelines would

stop after rule filtering. Here, we employ NNE techniques to effectively learn the filtered rules. This is done by using different neural network algorithms, i.e. Multilayer Perceptron, Generalized Feedforward Network, Modular Neural Network and Radial Basis Function Networks. For this NNE, the bagging technique would be employed and this involves re-sampling or re-weighting the training instances where all the results of each neural network are aggregated to produce better predictions. Here, we have done the aggregation via averaging. This will produce a neural knowledge base (weights) that would be considered the best abstraction of the knowledge from the rules and would serve as a robust repository for decision support even in the event that users do not provide sufficient input.

3 CONCLUSION

We would like to highlight that the knowledge discovery pipeline leaves much to be explored in terms of the algorithms and techniques that can be applied at each stage of the pipeline. For our extended pipeline, we have proposed mean/mode fill for data preparation, clustering ensemble with SOM for clustering or data annotation, Boolean Reasoning for discretization, rough set analysis for rule extraction, Rule Quality Function (based on support, consistency and coverage) for rule filtering and finally, a NNE for rule learning. In addition to contributing towards an extended knowledge discovery pipeline, we believe that this featured work will provide an alternative inductive approach to support diagnosis and decision support especially in the medical domain.

With the incorporation of ensembling techniques for clustering and learning, we hope to minimize the need, or to reduce the temptation, to switch to other algorithms by making the most out to the selected algorithm, i.e. SOM and our 'cocktail' of neural network algorithms. We are currently evaluating each stage of our extended knowledge discovery pipeline using continuous, discrete and also possibly mixed (continuous and discrete) medical datasets such as those on breast cancer and thyroid disease. We will also be exploring other methods of clustering ensembles in the second stage to be integrated into our extended pipeline in future. We believe this extended pipeline would result in more accurate knowledge-based predictions in our effort to make medical diagnosis and decision support more reliable and trustworthy.

REFERENCES

Abidi, S.S.R. and Hoe, K.M., 2002. Symbolic Exposition of Medical Data-Sets: A Data Mining Workbench to Inductively Derive Data-Defining Symbolic Rules. In *Proceedings of the 15th IEEE Symposium on Computer Based Medical Systems (CBMS 2002)*. Maribor, Slovenia.

Breiman, L., 1996. Bagging Predictors. *Machine Learning*. Vol. 24, pp. 123-140.

Dimitriadou, E., Weingessel, A., and Hornik, K., 2003. A cluster ensembles framework. In Abraham, A., Köppen, M., and Franke, K. (eds.), *Design and Application of Hybrid Intelligent Systems*. Frontiers in Artificial Intelligence and Applications. Vol. 104, pp. 528-534. IOS Press.

Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P., 1996. From Data Mining to Knowledge Discovery in Databases. *AI Magazine*. Vol. 17, No. 3, pp. 37-54.

Freund, Y. and Schapire, R.E., 1995. A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. In *Proceedings of the Second European Conference on Computational Learning Theory*. Barcelona, Spain, pp. 23-37.

Hansen, L.K. and Salamon, P., 1990. Neural Network Ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 12, pp. 993-1001.

Johnson, D.S., 1974. Approximation Algorithms for Combinational Problems. *Journal of Computer and System Sciences*, Vol. 9, pp. 256-278.

Michalski, R.S., 1983. A Theory and Methodology of Inductive Learning. In Michalski, R., Carbonell, J. and Mitchell, T. (eds.), *Machine Learning. An Artificial Intelligence Approach*, pp. 83-134. Springer-Verlag.

Risvik, K.M., 1997. Discretization of Numerical Attributes - Preprocessing for Machine Learning. *Project Report*. Knowledge Systems Group, Department of Computer Systems and Telematics, Norwegian Institute of Technology, University of Trondheim, Norway.

Strehl, A., and Ghosh, J., 2002. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal on Machine Learning Research*. Vol. 3, pp. 583-617.

Yang, Y and Kamel, M., 2003. Clustering Ensemble Using Swarm Intelligence. In *IEEE Swarm Intelligence Symposium*. Indianapolis, Indiana, USA.

Zhou, Z.H., Jiang, Y. and Chen, S.-F., 2003. Extracting Symbolic Rules from Trained Neural Network Ensembles. *AI Communications*. Vol. 16, No. 1, pp. 3-15.