International Advance Journal of Engineering, Science and Management (IAJESM)
ISSN -2393-8048, January-June 2022, Submitted in June 2022, iajesm2014@gmail.com

Applying Prescient Displaying Strategies to Modern Issues

Shivam Sharma, Department of Computer Science & Engineering, RDEC, Ghaziabad. Email- sharmashivam02@gmail.com
siteshwar Srivastava, Department of Computer Science & Engineering, RDEC, Ghaziabad

Abstract

There are more than 2.2 million significant street mishaps in the US consistently. Which is the most noteworthy on the planet. However, amazingly, street mishaps in India are multiple times not exactly in the US, yet the quantity of passings is mutiple and a half lakh a year. Simultaneously, the quantity of individuals who bite the dust in excess of 22 lakh street mishaps in America is just 34 thousand. As per the near report, the quantity of passings in 2.2 million street mishaps in the US is near 37 thousand in a year. In India, around 30% of the 480,000 street mishaps for example 1.5 lakh individuals lose their lives. This figure is number one in the entire world. Japan has the second-biggest number of street mishaps on the planet. There are 500,000 little and huge street mishaps in Japan and the quantity of individuals who bite the dust from street mishaps here is just 4,500 every year.

Keywords—Industy, safety

1. Introduction

Overall, this paper highlights the challenges that arise when applying predictive modeling techniques to industrial problems and proposes a novel approach for addressing these challenges using a general conceptual architecture that incorporates parameter cross-validation, ensemble techniques, and meta-learning. The proposed instance of this architecture is shown to be effective and robust when applied to real-life data sets.In this way, the data is transformed in favor of the modeling techniques (see [1] for a review of such case studies). However, the drawback of this approach is that because the data can dramatically change from case to case, each new case requires new time-consuming manual pre-processing. Furthermore, once the data is pre-processed the correct predictive method must be selected. This selection is critical for the performance of the whole model since different techniques have different strengths and weaknesses. Very often one cannot see a-priori which technique fits best the data and different methods and their parameters have to be tried. Even more critically, in an industrial environment, the model developers often have their favorite techniques and focus only on these without taking any other approaches into account which is not of advantage for the final performance of the model. The most applied techniques to industrial modElling problems are ranging from statistically based Principal PetrKadlec and Bogdan Garbs are with the Computational IntelliJgene Research Group, Bournemouth University, Bournemouth, BH12 5BB, United Kingdom . Component Regression [2], Partial Least Squares Regression [3] and Support Vector Machines [4] to techniques from computational intelligence like Multi-Layer Perceptron [5] and Neuro-Fuzzy Systems [6]. Although many applications of these techniques have been published (see e.g. [1], [7] for reviews) most of the authors claim that a certain effort must be spent on the preparation of the data (i.e., data pre-processing) as well as the techniques (i.e., parameter selection). Another problem is that one also cannot separate the two previously discussed tasks, i.e., data pre-processing and predictive technique selection and parametrization due to their mutual influence on each other. This fact further increases the number of possibilities to be tested in order to identify a well-performing model.Section II shows a brief overview of the conceptual architecture and outlines its most critical aspects necessary for the understanding of the proposed instance. This is followed by a methodology for the development of the model and the way in which the data is typically provided in an industrial environment in Section III. Section IV is the main contribution of this paper as it presents the actual instance of the architecture and shows the mechanisms applied in order to achieve high robustness and adaptive capabilities. The model is then evaluated in Section V by applying it to two real-life data sets. Finally, the paper is concluded in Section VI.

2.Architecture Overview

This section gives a brief overview of the architecture which is instantiated in this work. The architecture is in more detail discussed in [11]. Due to space limitations the figure showing the general structure of the architecture cannot be shown here however Fig. 6 showing an instance of architecture can be used to see its structure. The architecture consists of eight main modules which are together with their functions outlined

The eight main modules of the architecture include:

1. Data Collection and Pre-processing: This module is responsible for collecting data from various sources and pre-process it for further analysis.
2. Feature Extraction: This module extracts relevant features from the pre-processed data.
3. Feature Selection: This module selects the most relevant features for the analysis.
4. Classification: This module classifies the data into different classes based on the extracted features.
5. Evaluation: This module evaluates the performance of the classification model.
6. Visualization: This module visualizes the results of the analysis.
7. User Interface: This module provides a graphical user interface for users to interact with the system.
8. Persistence: This module handles the storage and retrieval of data for future use.

All these modules work together to provide a complete system for analyzing data. The architecture is highly modular and flexible, allowing it to be easily adapted to different data analysis tasks. The information processing within the model is structured in a hierarchical manner. At the lowest level of the architecture, there is a diverse set of data processing units called computational paths which are maintained in the Paths module (see Fig. 1 for the internal path structure). The paths consist of an arbitrary number of pre-processing methods and one computational learning method. At the next level, the paths are combined into path combinations which, apart from the fact that they operate in another data space, do not differ from the paths. At the highest level of complexity, management of the underlying levels which evolves the echotexture towards the global goal defined by the underlying task (e.g., best predictive performance in the Mean Squared Error sense), takes place.

3. Methodology

The availability of both historical and real-time data is crucial for the successful operation of our proposed model. The historical data is used for the initial training of the model, while the real-time data stream is used for the continuous adaptation of the model during the on-line phase. The ability to deal with incremental data, varying sample rates and potential data inconsistencies is also essential for the model to perform effectively in industrial settings. By leveraging both historical and real-time data, our proposed model can provide valuable insights and optimize industrial processes in real-time. the sampling rate between the input and the target data can differ and additionally there can also be sys between them. The correct target values can be applied to the evaluation of the model performance and its adaptation during the on-line phase.

4. Experiment

In this section, two soft sensors for the online prediction of the target variable are presented as a practical implementation of the architecture instance discussed in Section IV. For the experimental evaluation, we follow the methodology from Section III and split the available data into two sets. A set of historical data (30% of the available data sample) and online data which are the residual 70% of samples. This split of the available data is justified by the focus on the evolutionary properties of the model. A. The data set

5. Support Vector Regression (SVR): Non-linear regressionmodel [17]

The Soft Sensor module is responsible for generating the predictions given the incoming data. Both MLR and SVR models are trained and used for generating predictions. The Model Management module monitors the performance of the model on the online data (Online) and trains the models with the historical data (Deist) at regular intervals. The Monitoring and Feedback module receives feedback on the performance of the models and communication with other parts of the process control system. An instance of the soft sensor architecture for the prediction of the target variable from the online data stream of the drier and the thermal oxidizer data set. C. ResultsIn this section, we present the results of the experiments performed using the architecture instance developed for the two data sets.

For the drier data set, a comparison of the prediction results between the two models (MLR and SVR) are presented where the performance of the models is measured in terms of the root-mean-square error (RMSE) of the predictions with respect to the target variable. The results show that both models achieve good results, with SVR slightly outperforming MLR in this case. For the thermal oxidizer data set, Fig. 8 shows the results of the prediction task using both models. In this case, the performance of both models is comparable, with MLR slightly outperforming SVR. Prediction performance comparison between MLR and SVR for the drier data set Prediction performance comparison between MLR and SVR for the thermal oxidizer data set Overall, the results suggest that the soft sensors developed using the presented architecture achieve good performance in predicting the target variable in both data sets. The choice of the learning method (MLR or SVR) appears to depend on the specific characteristics of the data set. Further, the use of pre-processing methods (standardization, smoothing, PCA) is found to be useful in improving the performance of the models.

Conclusion

This work highlights the potential of the proposed architecture for the development of robust and adaptive data-driven models that can handle changing data and be applied across multiple modeling tasks. Its ability to automatically adapt its structure and parameters to new data sets without requiring extensive parameter optimization makes it a promising approach for practical applications. Further research can explore the scalability and generalization of the architecture to more complex modeling tasks and larger data sets. This work demonstrates the applicability of an architecture for the development of evolving data-driven models which was proposed earlier by the authors. An instance of the architecture, which makes use of some of the mechanisms provided for model development and maintenance, is shown to have adaptation ability at different levels. A model developed according to the architecture shows comparable performance to another adaptive model based on the Locally Weighted Projection Regression (LWPR) where the parameters of the LWPR method were adjusted to deliver optimal performance for the given modeling task. It is also presented that without any additional parameter optimization, the LWPR technique fails to deliver a working model on another data set. This contrasts with the instance of the architecture which succeeds to deliver a working model for the new data set without any parameter changes. These results demonstrate that the developed model can evolve on one hand with changing data and on the other hand is able to adapt its structure with the underlying data set and thus allows the application. It is also presented that without any additional parameter optimization, the LWPR technique fails to deliver a working model on another data set.

Acknowledgment

I'm thankful to everyone I've worked with on this project and others like it. Each member has shared their views and a lot of personal and professional advice and taught me a lot about both scientific research and life in general.

I would like to express my sincere gratitude to my guide, Rishav Raj, for their invaluable guidance and support throughout this research project.

Remember to always check the specific requirements and guidelines provided by your institution or professor, as they may have different formatting preferences for research papers.

References

1. Adorno, T. &Horkheimer, M. (2002) The Dialectic of Enlightenment, Stanford: Stanford University Press.

"Random forests," Machine Learning, vol. 45, no. 1, pp.5–32, 2001.

[2] A. Liwa and M. Wiener, "Classification and regression by random forest," R News, vol. 2, no. 3, pp. 18–22, 2002.

[3] N. K. Kasimov, "Evolving Connectionist Systems - Methods andApplications," IEEE Press, 2002.

[4] J. K. Kumar and A. L. N. Reddy, "Fuzzy-based decision supportfor air pollution

monitoring and control," in Advances in SoftComputing, vol. 39, pp. 152–164, Springer Berlin Heidelberg, 2006.
[5] S. S. Sengupta and N. G. Leveson, "Data mining in software engineering," IEEE Transactions on Software Engineering, vol. 28, no.December, pp. 1051–1070, 2002.
[6] B. Garbs and M. Lech, "Real-time soft sensors for multiple outputs:a case study," in Soft Computing and Industry, Studies in Fuzzinessand Soft Computing, vol. 66, pp. 339–362, Springer Berlin Heidelberg,2002.
[7] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggi, andV. Vatnik, "Feature selection for SVMs," in Advances in NeuralInformation Processing Systems, 2003, pp. 668–674.
[8] C. E. Bradley and P. Smyth, "Rule induction with CN2: Some recentimprovements," in Machine Learning: Proceedings of the TwelfthInternational Conference, 1995, pp. 151–163.
[9] G. Weiss, "Mining with rarity: A unifying framework," ACM SIGKDDExplorations Newsletter, vol. 6, no. 1, pp. 7–19, 2004.
[10] J. R. Quinlan, C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.
[11] B. Garbs, A. Bargiel, and R. J. Rabinowicz, "Feature selection forrule induction," in Advances in Soft Computing - AFSS 2004, vol. 3131of Lecture Notes in Computer Science, pp. 16–24. Springer BerlinHeidelberg, 2004.
[12] Y. Dote and S. J. Vaska, "Industrial applications of soft computing: A review," in Proceedings of the IEEE, vol. 89, 2001, pp. 1243–1265.
[13] P. Kadlec and B. Garbs, "Adaptive local learning soft sensor for inferential control support," in CIMCA 2008. Vienna: IEEE, 2008.
[14] R. O. Dude, P. E. Hart, and D. G. Stork, Patten Classification. John Wiley and Sons, 2001.
[15] C. Croix and A. Ruiz-Gazan, "High breakdown estimators for principal components: the projection-pursuit approach revisited," Journal of Multivariate Analysis, vol. 95, no. 1, pp. 206–226, 2005.
[16] A. J. Dobson, An Introduction to Generalized Linear Models. Chapman Hall/CRC, 2002.
[17] E. Frank, M. Hall, and B. Farlinger, "Locally weighted naive bayes," in Proceedings of the Conference on Uncertainty in Artificial Intelligence, 2003, pp. 249–256.
[18] S. Vijayakumar, A. D'Souza, and S. Schaad, "Incremental online learning in high dimensions," 2005