Decision-making process analysis and semantic explanation extraction from trained supervised machine learning models in medical expert systems

1st Aliaksandr Kurachkin
faculty of radiophysics and computer technologies
Belarusian State University
Minsk, Belarus
alex.v.kurochkin@gmail.com

2nd Vasili Sadau
faculty of radiophysics and computer technologies
Belarusian State University
Minsk, Belarus
sadov@bsu.by

3rd Tatiana Kachan
Belarusian State Medical University
Minsk, Belarus
tvk35@yahoo.com

***Abstract*—Supervised machine learning provides a mechanism for establishing an approximation of input-output relationship between arbitary dataset. However, semantic interpretation of an underlying decision-making process of a trained model is very hard, especially considering the probabilistic nature of machine learning. The paper discusses possible ways to semantically explain decision-making process of a trained supervised machine learning model in order to gain insights to the dataset and derive new expert knowledge from such models.**

***Keywords*—supervised machine learning, machine learning explanation, decision support systems, medical expert systems**

I. Introduction

Supervised machine learning has become a staple of data mining techniques in the recent years, solidly establishing itself as a separate field of research with a multitude of approaches and algorithms, suitable for solving a wide array of problems. One of the most important applications of such techniques is creating decision support systems – given a sufficiently large dataset, supervised machine learning algorithms are able to derive a decision path by establishing non-linear dependencies between input features and expected results. However, one of the main problems of supervised machine learning algorithms is a semantic interpretation of acquired results [1], [2].

Although some models, like predictive decision trees, are relatively easy to examine, the models they produce are usually too simple to correctly approximate datasets with non-linear dependencies. Because of this, most of the dependencies established by a more complex supervised machine learning model, like gradient-boosted decision trees, random forests or neural networks, are highly non-linear in nature and are usually very hard to interpret. Moreover, such models usually employ a degree of randomness in order to make the learning process more robust, making the learning itself probabilistic, which means that several retrainings of the same model architecture on the same dataset may yield different models, with their own underlaying decision-making processes.

II. Semantics of decision-making process

In many applications, supervised machine learning models used as part of decision-making process must not only be correct on the dataset provided, but also be able to generalize for new data and provide semantically correct results. While the former is usually solved by introducing a test and cross-validation dataset split, the latter is impossible to establish for a general case. In other words, a trained supervised machine learning model remains a black box that might produce correct decisions on the items provided in dataset, but the decision-making process itself remains obscure, making it harder to "trust" such a model from a semantic interpretation standpoint.

Another important problem is generation of new expert knowledge based on data, or gaining insights into the relationship between a particular data item input and output in a given model. Even if a model demonstrates a high prediction accuracy in a certain dataset, the prediction itself is sometimes not as important as the decision-making process behind making that prediction. This is especially true for medical expert systems and sympthomogenesis – semantic explanation of dependencies between some measurable inputs and the prediction of a certain pathology may become, in essence, the definition of a symptom – a piece of expert knowledge that establishes, based on dataset given, that specific types of inputs may, in fact, serve as indicators that a certain pathology is present. In order to establish such a dependency, the nature of relation must be explained semantically, i.e. in an understandable and human-readable form [1].

One of the main problems with establishing an understandable explanation for a trained model's decision-making process is the neccesity to create a reasonable semantically succint representation that doesn't hide away the complexity of the model itself. For this purpose, it's necessary to distinguish what kind of relationships are generally percieved as semantically understandable by a human being.

One of the more "understandable" decision-making classification techniques is a threshold-based linear binary classification using a single feature, i.e. feature exceeding a certain threshold signifies one class, and feature being lower than this threshold signifies the opposite class. More formally, given a dataset $\mathbf{T}$ of k elements for binary classification with n-dimensional input vectors

$$\mathbf{T} = \{\mathbf{X}, \vec{y}\}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{k \times n}$ – inputs matrix, $\vec{y} \in \mathbb{B}^k$ – expected outputs vector, $\mathbb{R}^{k \times n}$ – a set of rational-valued matrices with k rows and n columns, $\mathbb{B}^k$ – a set of k-dimensional vectors of boolean $\mathbb{B} = \{0, 1\}$ values, a threshold-based linear binary classification using a feature j is established as follows:

$$C_j(\vec{x}, p) = [x_j > p], \tag{2}$$

where x_q denotes a feature with the number q in input vector, and parameter p is some threshold value. The classifier (2) assumes that each individual feature of the input vector is sufficient to perform an entire classification. It is also possible to use binary classifier metrics, like precision, recall and F1-score, in order to establish the best threshold value p with respect to a dataset (1). Threshold value can also be adjusted to perform ROC-analysis and compare individual features for statistical significance using AUC (area under ROC-curve) metric [2], [3].

Such a classifier is quite primitive – it only uses a single feature and assumes a direct linear discrimination for classification. However, it has the advantage of being intuitively easy to explain and extract semantic meaning from. While it is rare for a single feature to serve as a good discriminator for output on the entire dataset, it is quite possible that a simple linear discriminative dependency in a local input neighbourhood would be representative of a more complex classifier behavior, while maintaining the ease of explanation.

III. Feature-dropping as a way to establish the most statistically significant features

Within the problem of deriving decision-making process from a trained supervised machine learning model it is possible to establish a sub-problem of determining the statistical significance of individual features in terms of how much their specific values affect the predicted output. While impossible to derive from a single model, it is possible to use ensembling to evaluate individual feature significance. In order to do this on a dataset (1), the following algorithm is proposed:

- A neural-network based feedforward model M is created for the dataset. Model's hyperparameters can be optimized at this stage using grid search methods in order to find the model that best fits the dataset (1) with respect to its F1-score $F_1(M)$.
- For each feature j of total n input features, the same model architecture (i.e. with the same set of hyperparameters) is used to train the model M_j on the dataset (1) with feature j excluded from the dataset.
- For each model M_j, F1-score $F_1(M_j)$ is calculated.
- Amongst n models M_k and base model M, the best model M^* is found based on its F1-score.
- For every feature, a metric is calculated as follows:

$$Q(M_k) = 1 - (F_1(M^*) - F_1(M_k)) \tag{3}$$

The process can be repeated for any number of combinations of individual features – i.e. for pairs, triples, etc. of input features in the dataset, constrained by computational complexity and combinatorial explision of working with individual features. In such a case, it's possible to aggregate the metric (3) per feature.

The metric (3) can be used as a relative measure of statistical significance amongst the features considered. The idea behind it is that a statistically significant feature would have a large enough impact on the overal model that its removal would lead to a greater decrease in the F1-score whenever this particular feature is not used in the next model.

Features with high enough statistical significance are candidates for being discriminative enough to try and establish a single-feature threshold classifier (2), although without traversing all the possible input feature combinations, it's usually impossible to determine whether a specific feature is representative by itself, or if it relies on relationships (possibly non-linear) with other features in order to make a decision.

IV. Instance explanation using local linear approximations

While simple linear threshold binary classifiers (2) aren't usually sufficient to approximate dependencies in real data, their defining feature of being susceptible to semantic analysis makes them a useful tool for generating approximate explanation that may be used to describe a local behavior of a more complex model [3].

The paper proposes using local linear approximations as a method for explaining a specific instance prediction result for a trained supervised machine learning model. The explanation itself takes a form of a threshold condition for an arbitary classifier (2) that is representative of a model's behavior in proximity to a specific instance data point, while not necesserily being a good approximation of the model's behavior as a whole on the entire input parameter space.

Given a trained supervised machine learning model for binary classification $C_f(\vec{x}, \mathbf{\Theta}) : \mathbb{R}^n \to \mathbb{B}$ with a set of parameters $\mathbf{\Theta}$ that are determined during model's learning on dataset (1), a local linear approximation of this model at data point $\vec{x}^*$ can be derived using the following algorithm:

1) A subset of the input vectors from source dataset (1) are selected based on their proximity to the data point $\vec{x}^*$. For proximity measure, it's possible to use any n-dimensional vector measure, for instance, Euclidean distance:

$$d(\vec{x_1}, \vec{x_2}) = \sqrt{\sum_{i=1}^{n} (x_{1_i} - x_{2_i})^2}. \tag{4}$$

It's possible to select an arbitary number of vectors closest to the data point, or select vectors within a certain threshold distance.

Dataset point selection is used to determine a radius d_{min} around the point $\vec{x}^*$ that is representative of a model's behavior without bias introduced by other points. The simplest selection method is to assume d_{min} to be the distance to the closest point in the dataset.

2) The local proximity $P_{\vec{x}^*} = \{\vec{x}_i\}$ of $\vec{x}^*$ is sampled within d_{min}: $\forall \vec{x}_i | d(\vec{x}^*, \vec{x}_i) < d_{min}$. For generated sample points $\{\vec{x}_i\}$ respective classifier responses are obtained by evaluating the model C_f:

$$\{y_i\} = \{C_f(\vec{x}_i)\} \tag{5}$$

3) An explanatory classifier $C_{exp}^{(\vec{x}^*)}$ with a linear model is defined:

$$C_{exp}^{(\vec{x}^*)}(\vec{x}) = \theta_0^* + \vec{\theta}^* \cdot \vec{x} \tag{6}$$

The linear classifier $C_{exp}^{(\vec{x}^*)}$ represents a hyperplane in feature space that is used to discriminate between two classes in the original classification problem solved by C_f. Hyperplane parameters θ_0^* and $\vec{\theta}^*$ are initialized randomly.

4) Point sets $\{\vec{x}_i\}$ and $\{y_i\}$ are used as a training set for explanatory classifier $C_{exp}^{(\vec{x}^*)}$ in order to obtain specific hyperplane parameters.

The resulting linear classifier $C_{exp}^{(\vec{x}^*)}$ is, essentially, a local linear approximation of a more complex behavior exhibited by the original model C_f. However, this local approximation is guaranteed to be representative of the original classifier within d_{min} radius of the original data point $\vec{x}^*$, with the added benifit of being semantically representative and understandable by a human being. It is possible to analyze projections of the hyperplane defined by classifier model (6) in order to establish specific threshold values for individual features, and use the projection angle as a measure of this particular feature impact. For example, if a local approximation hyperplane is perpendicular to a feature axis k, that means that original classifier C_f in the proximity of an explained data point relies only on feature k to produce a prediction. With the threshold value t_k, it is possible to produce a semantic form of an explanation using the following statement:

For inputs $\vec{x}^*$ classification result is $y^* = C_f(\vec{x}^*)$, because x_k is greater (lesser) than t_k.

For arbitary plane alignment, similar statements can be produced for any input feature, while their impact can be determined by the cosine of the angle between a feature axis and the plane.

In other words, given a trained supervised machine learning model for binary classification $C_f(\vec{x}, \mathbf{\Theta}) : \mathbb{R}^n \to \mathbb{B}$ with a set of parameters $\mathbf{\Theta}$ that are determined during model's learning on dataset (1), a local linear approximation $C_{exp}^{(\vec{x}^*)}$ of this model at data point $\vec{x}^*$ can be used to create a hyperplane, that defines threshold values t_k (axis intersection points) and impact values $w_k = cos\varphi_k$ (where φ_k is the angle between feature axis and the plane) of an arbitary k-th feature. These individual values can be used to produce semantic form of an explanation with the following statements:

For inputs $\vec{x}^*$ and classification result $y^* = C_f(\vec{x}^*)$, the fact that x_k is greater (lesser) than t_k affects the output with impact w_k.

V. Example data analysis in ophtolmological decision support system

One of the domain areas where proposed approaches were used to generate expert knowledge is ophtolmological disease diagnosis, specifically optical nerve disorders like multiple sclerosis associated optic neuritis and glaucoma. The most common diagnostic tool in this area is optical coherent tomography (OCT) and scanning laser polarimetry (SLP), producing optical nerve and retina images and allowing to produce structural features. On the other hand, clinical practice also uses certain functional features. One of the unsolved problems is discriminating feature selection – some papers suggest that specific OCT or SLP features are the most impactful, while the others indicate that functional features should be used instead [5].

Based on historical data analysis, a simple feedforward binary classifier was created that uses all structural and functional features measured during the research for confirmed cases pathology and control group of healthy people. Separate classifiers were implemented for multiple sclerosis and glaucoma. The initial research demonstrated that the classifiers were performing better than linear thresholding using any single feature, with F1 scores of 0.75 versus 0.68 for multiple sclerosis, and 0.93 versus 0.81 for glaucoma [5].

The neural network architecture used in proposed classifiers comprised 26 input features, 3 hidden layers of 30 neurons each with ReLU activation function and dropout, and single-neuron output layer with variable classification threshold.

Black-box nature of proposed classifiers made it hard to justify their use in clinical practice, because semantic correctness of the underlaying decision-making process was impossible to determine.

The statistical significance analysis based on the proposed metric (3) allowed to confirm that certain sturctural and functional features were more impactful than the others. The general domain knowledge about the impact of individual features coincided with acquired impact predictions, confirming their correctness [5].

When examining individual samples using proposed local linear approximations method, it was possible to determine a clinical explanation for many of the inconclusive samples. Established input-output dependencies explained in an understandable way were used in order to propose new insights into clinical pathologenesis and confirm certain forms and characteristics of researched pathologies.

Conclusion

Explanation of decision-making process in trained supervised machine larning models provides an effective way to evaluate datasets used for training the model. The methods proposed in this paper include individual feature statistical significance evaluation based on the reduction of F1-score when this particular feature is excluded from training, as well as local approximations method that allows to explain the local behavior of a trained model within the proxmity of an individual data point.

Proposed algorithms can be used in order to gain insights into the dataset used to train the data, evaluate decision-making processes established within the models to derivate new expert knowledge, and semantically validate individual predictions. In certain applciations, such analysis allows to mitigate the issue of black-box behavior of supervised machine learning algorithms and enable their applications to domain fields where transparency of a decision-making process is required, like complex control systems or medical decision support systems.

The evaluation of proposed feature impact analysis and semantic instance explanation were used to evaluate a medical decision-support system for diagnosing optical disorders. Feature impact analysis based on the reduction of F1-score on the same model showed results that are similar to the general domain knowledge, while smenatic interpretation of the decision-making process allowed to produce new domain knowledge and found a better understanding of specific dosorder's pathologenesis.

References

[1] А. В. Курочкин, В. С. Садов, “Методы анализа и семантической интерпретации процессов принятия решения в классификационных нейросетевых моделях машинного обучения с учителем,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, 2019.

[2] M. A. Nielsen, “Neural Networks and Deep Learning,” Determination Press, 216 p., 2019

[3] T. Hastie, R. Tibshirani, J. Friedman, “The Elements of Statistical Learning: Data Mining, Inference and Prediction,” Springer Series in Statistics, Springer-Verlag New York, 2nd ed., 745 p., 2009.

[4] D. M. W. Powers, “From Precision, Recall and F-measure to ROC, Informedness, Markedness & Correlation,” Journal of Machine Learning Technologies, vol. 2, iss. 1, BioInfo Publications, pp. 37–63, 2011.

[5] Т. В. Качан, А. В. Курочкин, Е. А. Головатая и др., “Роль искусственных нейронных сетей в выявлении ранней гибели ганглионарных клеток сетчатки у пациентов с дегенеративными оптиконейропатиями,” Офтальмология. Восточная Европа. – 2019. – Т. 9, № 4. – С. 445–458.

Анализ процессов принятия решений и извлечение семантического описания в обученных моделях машинного обучения с учителем в медицинских экспертных системах

Курочкин А.В., Садов В.С., Качан Т.В.

Модели машинного обучения с учителем предоставляют механизм установления аппроксимации взаимодействия между входными и выходными значениями произвольных наборов данных. Тем не менее, семантическая интерпретация лежащего в основе этих моделей процесса принятия решения является сложной задачей, особенно в контексте вероятностного характера некоторых методов машинного обучения с учителем. В статье рассматриваются методы семантического объяснения процесса принятия решения обученной модели машинного обучения с учителем, что позволяет выделить сложные зависимости из наборов данных и вывести с их помощью новые экспертные знания.

Received 30.12.19