PAPER • OPEN ACCESS

Natural Language Explanation Model for Decision Trees

Natural Language Explanation Model for Decision Trees

Jesús Silva[1], Hugo Hernández Palma[2], William Niebles Núñez[3], Alex Ruiz-Lazaro[4] and Noel Varela[5]

[1]Universidad Peruana de Ciencias Aplicadas, Lima, Perú.
[2] Universidad del Atlántico, Puerto Colombia, Atlántico, Colombia.
[3]Universidad de Sucre, Sincelejo, Sucre, Colombia.
[4]Universidad Simón Bolívar, Barranquilla, Atlántico, Colombia
[5]Universidad de la Costa, Barranquilla, Atlántico, Colombia

[1]**Email:** jesussilvaUPC@gmail.com

Abstract. This study describes a model of explanations in natural language for classification decision trees. The explanations include global aspects of the classifier and local aspects of the classification of a particular instance. The proposal is implemented in the ExpliClas open source Web service [1], which in its current version operates on trees built with Weka and data sets with numerical attributes. The feasibility of the proposal is illustrated with two example cases, where the detailed explanation of the respective classification trees is shown.

1. Introduction

The generalization of the use of new technologies has allowed to work and live surrounded by intelligent systems [2]. Terms like smart city, factory, home, car or smart phone are becoming more and more popular. In reality, there are many devices with a certain intelligence that assist people every day, often without being fully aware of it. Special mention deserves the mobile phone, which offers a multitude of applications for almost anything that can be imagined. It can be affirmed that, although in the past the world lived an industrial revolution, now a social revolution driven by Artificial Intelligence (AI) is taking place [3].

When an intelligent system makes decisions that affect people (e.g. filtering calls, medical diagnosis, granting a loan, etc.), a multitude of questions may arise like [4]: Who is responsible for the collateral sequences that might result from the decisions made? What are the ethical consequences? Can there be legal consequences?

From a legal point of view, the European Parliament approved a new General Regulation on Data Protection [5] which began on 25 May 2018. The new regulation emphasizes the right of citizens to ask for explanations, regardless of whether decisions affecting them are made by a person or a computer program. This means that citizens can ask companies to give them explanations associated with the decisions made by the intelligent systems they use.

From a technical point of view: can you explain to us the application that made a decision because I made that decision and not another? For this, there are basically two options [6]: (1) the intelligent system is built following an interpretable model (also called white box) that an expert operator can analyze and understand in order to elaborate an explanation; or (2) the system is built following an explainable model that generates explanations by itself. The DARPA raised the following technical issues in 2016 [7]:

Can an intelligent machine learn autonomously to explain its behavior? Is the current generation of intelligent systems ready to give explanations in a clear, unambiguous way to both specialized and non-specialized audiences? And there is a challenge of creating a new generation of intelligent systems that can be explained between 2017 and 2021. The challenge was initially launched to American universities and research centers, with emphasis on the creation of multidisciplinary teams that would address not only algorithmic aspects but also implementation and evaluation with people. The selected teams started working in May 2017 but today only very preliminary results have been found (e.g. [8], [9]).

In practice, the responsibility for generating explanations falls directly on the operator associated with the intelligent system, if it is available for it [10]. Although there are knowledge-based systems that are interpretable, in recent years, AI techniques for automatic learning and supervised and unsupervised (i.e., with or without human intervention) data mining have become increasingly popular [11]. These systems are certainly proving to be useful and versatile, but most of them do not usually have any explanatory capability nor can they be easily interpreted by people (in which case they are said to be black box systems).

Therefore, the new legal framework demands that AI experts develop new algorithms that automatically provide explanations.

This study presents a model for the interpretation of one of the most interpretable AI algorithms such as decision trees for classification.

2. Decision trees classification

Within supervised learning from datasets, model-based methods are characterized by representing the knowledge learned in some representational formalism that makes that knowledge explicit. An important advantage of this approach is that, once the model is available, it can be applied directly to new instances (e.g., prediction problems, such as classification) without the need to maintain training data [12].

Decision trees use a tree as a representation formalism where the nodes represent conditions on the values of the attributes of the dataset, which are organized hierarchically, and where the branches of each node correspond to possible values of the attribute. There are different inductive methods [13], [14] for the construction of a decision tree, but all of them usually use "divide and conquer" strategies that build the tree from the root to the leaves by selecting, in each intermediate node, the attribute and the condition that partitions the dataset in the best possible way, usually based on entropy criteria and maximization of information gain [15].

In the specific case of classification trees, leaf nodes ideally contain a set of instances corresponding to the same class. The application for the classification of new instances starts evaluating the condition of the root node for the attributes of said instance and continuing the route through the corresponding branches and nodes. The classification process ends when a leaf node is reached, which indicates the class that corresponds to the instance. In practice, the condition that a leaf node contains only instances of the same class ("pure" node) is too restrictive, so the condition must be relaxed within purity margins. On the other hand, this results in trees incorrectly classifying just some cases (ideally very few), characteristic that is included in the confusion between classes matrix [16].

This model for the explanation of classification trees is based on the mentioned aspects. On the one hand, a global characterization of the classification problem and of the induced tree; on the other hand, an explanation of the route by the tree in the classification task.

3. Model for the Explanation Generation

The Natural Language text generation (popularly known as NLG by the acronym of "Natural Language Generation") constitutes an outstanding research line in the area of IA and Computational Linguistics [17].

This study focuses on the most popular NLG architecture, initially proposed by [18], and the Computational Theory of Perceptions proposed by [19]. The generation of explanations in Natural Language is done combining open source templates and libraries for the linguistic realization [20].

The explanation of classifiers through decision trees is proposed at two levels (global and local), as described below. All the examples used in the following sections to illustrate the proposal can be reproduced using the ExpliClas [1] web service.

3.1 Global explanation of a classifier

The first level is the global explanation, which is aimed at describing the general behavior of a given classification tree, learned from a particular dataset. The information included in the global explanation essentially refers to characteristics of the classification problem itself and its performance [21]. The input data for this explanation comes from the dataset and from the confusion matrix of the learned classifier.

The overall explanation planning contains the following elements:

- Contextualization of the problem, which lists the classes of the problem.
- Reliability of the classifier, which evaluates the overall percentage of correct classifications on the learning dataset, including a qualitative assessment according to an established definition of linguistic values.
- Confusion of the classifier, highlighting the classes that are most affected by this confusion. The confusion matrix of the classifier is interpreted as an adjacency matrix of a network, whose cycles are understood as possible closed paths of confusion between classes. It takes the longest road to be included in the explanation. If the level of confusion is low, this part of the explanation will be omitted. In order to enumerate the classes, the aim is to limit the length of the explanation, treating differently the cases in which the closed path of confusion is long (many confused classes) or short (reduced number of confused classes) so that the length of the explanation is as short as possible. Thus, in the first case, the classes for which there is no confusion are listed (expressing them as exceptions) and in the second case, the classes for which there is confusion are listed.
- High confusion between classes, where those pairs of classes that present a high level of confusion and are not included in the previous cycles are highlighted.

3.2 Local explanation of an instance

The second level is the local explanation, which is oriented to explain the result of the classification obtained when applying the classifier on a new instance. The information included in the local explanation refers to the route through the classification tree from root to a leaf, determined by the conditions fulfilled in the different nodes of the tree for the instance to be classified.

The current version of the model defined for the generation of explanations in natural language, is applied just to numerical attributes, which allows to give a certain flexibility in the explanation, for considering possible alternatives to the real classification. For this, a certain tolerance is included regarding the threshold values of the conditions, in order to contemplate that small variations can occur in the value of an attribute, which could result in a different classification. The input data for the local explanation are the instance to be classified, the classification tree and the allowed tolerance value (by default, 5 % on the value of each attribute) [22].

The local explanation planning contains the following elements:

- Description of the class, which expresses the result of the classification and a linguistic summary of the values of attributes that have led to such classification. The summary includes, for each attribute X, expressions of the type "X is A", where A is a predefined linguistic value.
- Alternative explanations, which are built on the basis of the tolerance threshold mentioned above. A margin of tolerance of 5 % has been established for each of the node conditions that justify the classification, so that possible alternative classifications are explored and included in the explanation in case the values of the attributes meet the conditions within the margin of tolerance.
- Finally, the alternative explanation also includes those classes for which there is a high level of general confusion with the original class. For this purpose, the confusion matrix is considered

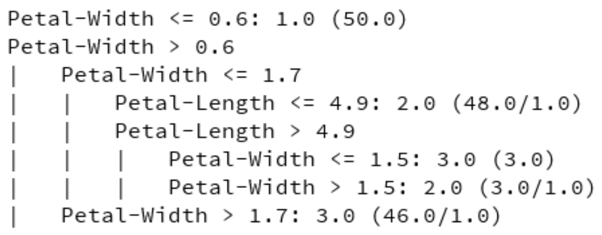
regarding to the classes involved, thus adopting a certain global perspective. Thus, if the classes have, in general, a high level of confusion, the explanation emphasizes this aspect; while, if the level of confusion is low, it will be presented as an exceptional case.

4. **Application**

Once described the elements that make up each explanation, this section presents a complete example to illustrate the operation of the proposal step by step. In this case, classifiers are learned using algorithm C4.5 [20], in the implementation available in Weka (J48) [16], [17].

4.1 IRIS dataset

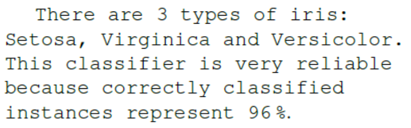
The IRIS data set (one of the best known in the repository [18]) is made up of 150 instances, 4 numerical attributes and 3 classes. The classification tree generated by Weka (Figure. 1) is formed by 10 total nodes, 6 of them leaf nodes that decide the classification and the 4 remaining nodes with the conditions (comparisons on the values of the attributes) to decide the classification. It is, therefore, a simple tree that will be used as an example.



```
Petal-Width <= 0.6: 1.0 (50.0)
Petal-Width > 0.6
|   Petal-Width <= 1.7
|   |   Petal-Length <= 4.9: 2.0 (48.0/1.0)
|   |   Petal-Length > 4.9
|   |   |   Petal-Width <= 1.5: 3.0 (3.0)
|   |   |   Petal-Width > 1.5: 2.0 (3.0/1.0)
|   Petal-Width > 1.7: 3.0 (46.0/1.0)
```

Figure 1. Classification tree corresponding to the IRIS data set.

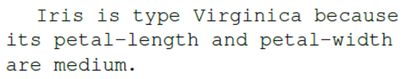
The overall explanation generated in this case can be seen in Figure 2.



```
   There are 3 types of iris:
Setosa, Virginica and Versicolor.
This classifier is very reliable
because correctly classified
instances represent 96%.
```

Figure 2. Global explanation of the example

The local explanation for the instance (Sepal- Length: 5.7, Sepal-Width: 4, Petal-Length: 5.1, Petal-Width: 1.4) is shown in Figure 3.

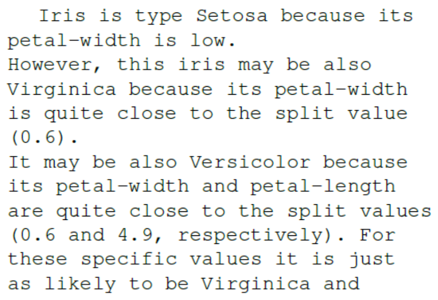


```
   Iris is type Virginica because
its petal-length and petal-width
are medium.
```

Figure 3. Global explanation of the example - short version

In this case, the explanation consists of indicating the linguistic values corresponding to the numerical values of the attributes that have given rise to the classification, as detailed in the figure.

However, for an instance whose values are precisely those of the intermediate node thresholds (Sepal-Length: 5.9, Sepal-Width: 5, Petal-Length: 4.4, Petal-Width: 0.7), the explanation is more extensive (see figure 4).



```
  Iris is type Setosa because its
petal-width is low.
However, this iris may be also
Virginica because its petal-width
is quite close to the split value
(0.6).
It may be also Versicolor because
its petal-width and petal-length
are quite close to the split values
(0.6 and 4.9, respectively). For
these specific values it is just
as likely to be Virginica and
```

Figure 4. Global explanation of the example - extended version

In this case, the classification carried out is of Setosa class. However, as the values of the instance are the same of the thresholds, and fall within the established tolerance range of 5%, the two branches of the root node and those of the node that classifies by length are considered as alternatives. All these alternatives lead to the Virginica and Versicolor classes. In both cases, the threshold value that justifies it is indicated and the situation is valued as being indistinctly either one or the other. However, a global character nuance is introduced, since according to the confusion matrix of the classifier, the confusion of the Setosa class with the Virginica and Versicolor classes is very rare (see Figure 5).

$$\left(\begin{array}{lccr} & Set. & Virg. & Vers. \\ Set. & 49 & 1 & 0 \\ Virg. & 0 & 47 & 3 \\ Vers. & 0 & 2 & 48 \end{array}\right)$$

Figure 5. Global matrix

5. Conclusions and Future Studies

This paper presented a model for the generation of explanations (global and local) in natural language about classifications made for decision trees with numerical attributes. The model is implemented in the ExpliClas [1] web service. As future study, an exhaustive validation of the model will be carried out with real users and will refine the explanations according to the received feedback. Additionally, the explanation model will be extended to consider categorical attributes and gray box classification algorithms, such as fuzzy decision trees, among others.

References

[1] B. Lopez-Trigo, J. M. Alonso, and A. Bugar´ın, “ExpliClas: Web service for the automatic explanation in natural language of classification models in data mining,” 2018, http://demos.citius.usc.es/ExpliClas/.

[2] Jain, Mugdha, and Chakradhar Verma. "Adapting k-means for Clustering in Big Data." International Journal of Computer Applications 101.1 (2014): 19-24.

[3] S. Ramiırez-Gallego, A. Fernandez, S. Garcıa, M. Chen, and F. Herrera, “Big data: Tutorial and guidelines on information and process fusion for analytics algorithms with mapreduce,” Information Fusion, vol. 42, pp. 51 – 61, 2018

[4] M. Hamstra, H. Karau, M. Zaharia, A. Konwinski, and P. Wendell, Learning Spark: Lightning-Fast Big Data Analytics. O’Reilly Media, 2015.

[5] Lis-Gutiérrez JP., Gaitán-Angulo M., Henao L.C., Viloria A., Aguilera-Hernández D., Portillo-Medina R. (2018) Measures of Concentration and Stability: Two Pedagogical Tools for Industrial Organization Courses. In: Tan Y., Shi Y., Tang Q. (eds) Advances in Swarm Intelligence. ICSI 2018. Lecture Notes in Computer Science, vol 10942. Springer, Cham

[6] J. Lin, "Mapreduce is good enough? if all you have is a hammer, throw away everything that's not a nail!" Big Data, vol. 1, no. 1, pp. 28–37, 2013.

[7] Viloria, A., & Gaitan-Angulo, M. (2016). Statistical Adjustment Module Advanced Optimizer Planner and SAP Generated the Case of a Food Production Company. Indian Journal Of Science And Technology, 9(47). doi:10.17485/ijst/2016/v9i47/107371.

[8] D. Garcia-Gil, S. Ramıırez-Gallego, S. Garcia, and F. Herrera, "Principal Components Analysis Random Discretization Ensemble for Big Data," Knowledge-Based Systems, vol. 150, pp. 166 – 174, 2018.

[9] N. Sapankevych y R. Sankar, "Time Series Prediction Using Support Vector Machines: A Survey", IEEE Computational Intelligence Magazine, vol. 4, núm. 2, pp. 24–38, may 2009.

[10] Viloria A., Lis-Gutiérrez JP., Gaitán-Angulo M., Godoy A.R.M., Moreno G.C., Kamatkar S.J. (2018) Methodology for the Design of a Student Pattern Recognition Tool to Facilitate the Teaching - Learning Process Through Knowledge Data Discovery (Big Data). In: Tan Y., Shi Y., Tang Q. (eds) Data Mining and Big Data. DMBD 2018. Lecture Notes in Computer Science, vol 10943. Springer, Cham.

[11] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell, "Generating visual explanations," in Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 2016, pp. 3–19.

[12] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," Journal of Artificial Intelligence Research, vol. 61, pp. 65–170, 2018.

[13] Ruß G. Data Mining of Agricultural Yield Data: A Comparison of Regression Models, In: Perner P. (eds) Advances in Data Mining. Applications and Theoretical Aspects, ICDM 2009. Lecture Notes in Computer Science, vol 5633.

[14] S. Barocas and D. Boyd, "Computing ethics. engaging the ethics of data science in practice," Communications of the ACM, vol. 60, no. 11, pp. 23–25, 2017.

[15] Hernández, J. A., Burlak, G., Muñoz Arteaga, J., y Ochoa, A. (2006). Propuesta para la evaluación de objetos de aprendizaje desde una perspectiva integral usando minería de datos. En A. Hernández y J. Zechinelli (Eds.), Avances en la ciencia de la computación (pp. 382-387). México: Universidad Autónoma de México.

[16] N. Sapankevych y R. Sankar, "Time Series Prediction Using Support Vector Machines: A Survey", IEEE Computational Intelligence Magazine, vol. 4, núm. 2, pp. 24–38, may 2009.

[16] S. Gang Wu, F. Sheng Bao, E. You Xu, Y.-X. Wang, Y.-F. Chang, and Q.-L. Xiang, "A leaf recognition algorithm for plant classification using probabilistic neural network," in IEEE International Symposium on Signal Processing and Information Technology, 2007, pp. 1–6..

[17] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, Data Mining: Practical Machine Learning Tools and Techniques, 4th ed. Morgan Kaufmann, 2016.

[18] D. Gunning, "Explainable Artificial Intelligence (XAI)," Defense Advan- ced Research Projects Agency (DARPA), Arlington, USA, Tech. Rep., 2016, DARPA-BAA-16-53

[19] Scheffer, T. (2004). Finding Association Rules that Trade Support Optimally Against Confidence. Intelligent Data Analysis, 9(4), 381-395.

[20] J. M. Alonso, A. Ramos-Soto, E. Reiter, and K. van Deemter, "An exploratory study on the benefits of using natural language for ex- plaining fuzzy rule-based systems," in IEEE International Conferen- ce on Fuzzy Systems (FUZZ-IEEE), Naples, Italy, 2017, pp. 1–6, http://dx.doi.org/10.1109/FUZZ-IEEE.2017.8015489.

[21] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauly, M. J. Franklin, S. Shenker, and I. Stoica, "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing," in Procee- dins of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12). San Jose, CA: USENIX, 2012, pp. 15–28

[22] S. Verbaeten and A. Assche, "Ensemble methods for noise elimination in classification problems," in 4th International Workshop on Multiple Classifier Systems, ser. Lecture Notes on Computer Science, vol. 2709. Springer, 2003, pp. 317–325.