Towards an FCA-Based Approach for Explaining Multi-label Classification

Hakim Radja, Yassine Djouadi, Karim Tabia

Towards an FCA-based approach for explaining multi-label classification

Hakim Radja[1][0000−0002−4345−426X], Yassine Djouadi[2,3][0000−0003−1151−5221], and Karim Tabia[3][0000−0002−8632−3980]

[1] Mouloud Mammeri University of Tizi-Ouzou, Computer science department, BP 17, RP, Tizi-Ouzou, Algéria, hakim.radja@ummto.dz
[2] University of Algiers 1, RIIMA Laboratory, Algeria, y.djouadi@univ-alger.dz
[3] Univ. Artois, CNRS, CRIL, F-62300 Lens, France tabia@cril.fr

Abstract. Multi-label classification is a supervised learning task where each data item can be associated with multiple labels simultaneously. Although multi-label classification models seem powerful in terms of prediction accuracy, they have however like mono-label classifiers certain limitations mainly related to their opacity. We propose in this preliminary work a novel approach for explaining multi-label classification models based on formal concept analysis (FCA). The proposed approach makes it possible to answer certain questions that a user may ask such as: *What are the minimum attribute sets allowing the classifier f to make a prediction ?* and *What are the attributes that contribute to a given prediction?*

Keywords: Explainable AI · FCA · multi-label classification

1 Introduction

Until recently, machine learning (ML) models mainly focused on making accurate predictions. ML is indeed widely used in several areas but regularly comes up against a major issue, its black-box side especially due to the complexity of the models used (eg. models based on deep learning can have several million parameters). Explainable AI and interpretable ML attempt to address these issues. They aim at equipping ML models with the ability to explain or present their behavior in understandable terms [1]. Most of explainable ML approaches try to assess the influence of attributes in the predictions made by classifiers.

We are interested in this preliminary work in the explanation of the predictions made by multi-label classifiers. To do this, we rely on a powerful mathematical framework, not yet used in explaining multi-label classifiers, that is the one of formal concept analysis (FCA) to answer questions such as:
- *What are the attributes that contributed to the prediction of the class set y predicted by the multi-label classifier f?*
- *What is the minimum set allowing the classifier f to make this prediction ?*
- *At what extent does an attribute influence the prediction of a given class belonging to the set y of classes predicted by the classifiers f?*

It should be noted that the majority of existing approaches in explainable ML are interested in the single-label case. This is the first novelty of our work. Moreover, the proposed approach is original insofar as it is based on formal concept analysis which has, to the best of our knowledge, never been used to explain the predictions of a multi-label classifier. Our approach aims to provide some forms of useful symbolic explanations. The other important advantage of our approach is that it is agnostic and can be applied to explain the predictions of any multi-label classifier.

2 Preliminaries

2.1 Multi-label classification

Multi-label classification is an extension of single-label classification, where classes are not mutually exclusive, and each instance can be assigned to several classes simultaneously. It is encountered in various modern applications such as text categorization [6], scene classification [5], video annotation and bio-informatics [14]. Let $x \in \mathcal{X}$ be a data instance denoted $x=(a_1, a_2, .., a_n)$ where a_i is a binary attribute (feature). Let $\mathcal{C}=\{c_1, c_2, .., c_m\}$ be a finite set of labels. A multi-label classifier f allows to predict for each instance $x \in \mathcal{X}$ a subset of labels $y \in 2^{|\mathcal{C}|}$.

2.2 Explainable AI

The existing methods in explainable ML mainly focus on how an explanation can be obtained and how the explanation itself can be constructed (for a survey, see [15]). Examples of common methods are : ordering the attributes contributions to a prediction [7], selection, construction and presentation of prototypes [13], summaries with decision trees [17] and decision rules [10]. The two most used methods are **LIME** (Local interpretable model agnostic explanation) [8] and **SHAP** (The Shapley Concept of Value) [4]. LIME is an explanatory technique that explains the predictions of any classifier through learning a locally interpretable model around the prediction. **SHAP** is based on game theory and assesses on average the contribution of each feature to the prediction. In this paper, we rather focus on an alternative and complementary category of explanations that are symbolic and may be very useful to the end-users. More precisely, we focus on some forms of sufficient reason explanations. These latter justify what is enough or necessary to trigger the prediction.

2.3 Formal concept analysis

Formal concepts analysis (FCA) [3] consists in learning pairs of subsets (objects, properties) called formal *concepts*, from a binary relation, called *formal context*, between a set of *objects* and a set of *properties*. Let $\mathcal{O}$ and $\mathcal{P}$ be sets of objects and properties respectively, and $\mathcal{R}$ be a binary relation between $\mathcal{O}$ and $\mathcal{P}$ verifying $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P}$. A pair $(x, a) \in \mathcal{R}$ (also denoted $x\mathcal{R}a$) means that the object $x \in \mathcal{O}$

has the property $a \in \mathcal{P}$. The triplet $\mathcal{K}:=(\mathcal{O},\mathcal{P},\mathcal{R})$ is called a formal context. Let $\mathcal{K} := (\mathcal{O},\mathcal{P},\mathcal{R})$ be a formal context. For all $X \subseteq \mathcal{O}$ and $A \subseteq \mathcal{P}$, the Galois derivation set operator, denoted $(.)^{\Delta}$, is defined as follows: $\mathrm{X}^{\Delta} = \{p \in \mathcal{P} \mid X \subseteq \mathcal{R}(p),\ A^{\Delta} = \{x \in \mathcal{O} \mid A \subseteq \mathcal{R}(x)\}$. Intuitively, X^{Δ} is the set of properties common to all the objects of X and A^{Δ} is the set of objects having all the properties of A.
A formal concept is a pair (X, A) such that $X \subseteq \mathcal{O}$, $A \subseteq \mathcal{P}$, $X^{\Delta} = A$ and $A^{\Delta} = X$. X and A are respectively called *extent* and *intent* of the formal concept (X, A). In this case, we have also $(A^{\Delta})^{\Delta} = A$ and $(X^{\Delta})^{\Delta} = X$.

Example 1. Table 1 provides an example of a formal context $\mathcal{K}$ where the set of items $\mathcal{O}=\{x_1, x_2, x_3, x_4, x_5\}$ and the set of attributes $\mathcal{P}=\{a_1, a_2, a_3, a_4\}$. An

$\mathcal{R}$	a_1	a_2	a_3	a_4
x_1	×			×
x_2		×	×	
x_3	×	×	×	
x_4				×
x_5			×	

Table 1. Example of a formal context

example of a formal concept in Table 1 is $\langle X_1, A_1 \rangle = \langle \{x_2, x_3\}, \{a_2, a_3\} \rangle$ where $X_1 = \{x_2, x_3\}$ is the extent of the formal concept and $A_1 = \{a_2, a_3\}$ is its intent.

The set $\mathcal{B}(\mathcal{K})$ of all formal concepts of $\mathcal{K}$ is partially ordered by :
$(X_1, A_1) \preceq (X_2, A_2) \Rightarrow X_1 \subseteq X_2 (A_2 \subseteq A_1)$. The subsumption relation $\preceq$ organizes the formal concepts in a concepts lattice (Galois lattice) denoted by $\mathcal{B}(\mathcal{O},\mathcal{P},\mathcal{R})$ or $\mathcal{B}(\mathcal{K})$. FCA has several advantages for data analysis. In particular, thanks to the visual representation of the lattice of concepts, it is possible to visually analyze and explore the data and its structure. Another frequent use is the generation of association rules, which also allows data analysis and knowledge extraction. In [16], an overview of classification methods based on formal concepts analysis is provided. In [2], the authors propose a learning classifier system (LCS) based on FCA to generate and exploit multi-label association rules which highlight the different relationships between labels. In [9], a neural network architecture based on concept lattices is used to improve the model explainability.

3 From multi-label predictions to FCA representation

This section briefly presents our FCA-based approach for explaining the predictions of a multi-label classifier f. We first need to associate a *local* or *global* formal context to the classifier.

- *Local explanations* : If one needs to explain locally a prediction $f(x)$, then we need to build a local formal context representing the predictions of the classifier in the neighborhood of data instance x. In case we are given a dataset $\mathcal{D}$, the neighborhood of x, denoted $\mathcal{N}(x)$ is simply obtained by selecting m data instances from $\mathcal{D}$ that are close to x. More precisely, we associate a local formal context composed of data instances $x' \in \mathcal{N}(x)$ and their predictions $f(x')$. Otherwise, one can obtain $\mathcal{N}(x)$ by applying local perturbations to instance x as it is done in most explainability approaches such as LIME [8].
- *Global explanations* : This case corresponds to the use of all the predictions of f over a dataset $\mathcal{D}$ to explain the prediction at hand. Global explanations can also be used in order to explain the global functioning of a classifier in the general case.

Recall that a multi-label classifier f associates with each data instance x described by its feature vector $(a_1, a_2, .., a_n)$ a subset of classes y from $\mathcal{C}=\{c_1, c_2, ..., c_k\}$. Therefore, this relation implies three dimensions: a set of objects $\mathcal{X}$ (namely, data instances), a set of attributes $\mathcal{P}$ and a set of classes $\mathcal{C}$. Thus $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P} \times \mathcal{C}$. The transformation of multi-label data into a formal context consists in transforming this three-dimensional relation into a two-dimensional one to obtain a formal context with only two dimensions. This is the first contribution of this paper.

3.1 Building a formal context for the classifier predictions

Let $\mathcal{MLD}=(\mathcal{O}, \mathcal{P}, \mathcal{C}, \mathcal{R})$ be a multi label dataset where $\mathcal{O}$ is a set of instances, $\mathcal{C}$ be the set of classes, $\mathcal{P}$ be the set of features and $\mathcal{R}$ a ternary relationship $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P} \times \mathcal{C}$. Transforming this multi-label data into a formal context consists in transforming the ternary relation $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{P} \times \mathcal{C}$ into a binary relation $\mathcal{R}' \subseteq \mathcal{O} \times \mathcal{M}$ where $\mathcal{M}=(\mathcal{C} \times \mathcal{P})$ similar to what was done in [12]. Hence, in our approach, a formal context is a triplet $\mathcal{K}'=(\mathcal{O}, \mathcal{M}, \mathcal{R}')$ where $\mathcal{O}$ represents the set of objects, $\mathcal{M}=(\mathcal{C} \times \mathcal{P})$ is a set of pairs (c_k, a_j) with $a_j \in \mathcal{P}$, $c_k \in \mathcal{C}$, and $\mathcal{R}' \subseteq \mathcal{O} \times \mathcal{M}$ a binary relation between the two sets $\mathcal{O}$ and $\mathcal{M}$. In other words, the transformation of a the multi-label data into a context formal $\mathcal{K}'$ is done by flattening (projecting) of the set of classes on all the attributes of the objects, as follows :
- $x_i \in \mathcal{O}$ the i-th instance of the set of objects $\mathcal{O}$, $i = 1, m$
- $a_j \in \mathcal{P}$ the j-th property of the set of properties $\mathcal{P}$, $j = 1, n$
- $c_k \in \mathcal{P}$ the k-th class of the set of classes $\mathcal{C}$, $k = 1, p$
- $f_k(x)$ the prediction for the class c_k for the instance x by the classifier f

$$f_k(x_i) = \begin{cases} 0 \;\; if \;\; \mathcal{R}(x_i, a_j, c_k) = 0 \\ 1 \;\; if \;\; \mathcal{R}(x_i, a_j, c_k) = 1 \end{cases}$$

Example 2. Table 2 illustrates the use of Algorithm 1,given below, to represent multi-label data in the form of a formal context. The triplet (x_1,a_1a_2, c_1c_3) means that the object x_1 where the attributes a_1 and a_2 are present is predicted in classes c_1 and c_3. In other words, object x_1 is predicted in classes c_1 and c_3 under the conditions a_1 and a_2.

Algorithm 1 Flattening multi-label data into a formal context

Require: Multi-label data $\mathcal{D}$
Ensure: Formal context $\mathcal{K}^{'}$
1: **for** $i \leftarrow 1, m$ **do**
2: **for** $j \leftarrow 1, n$ **do**
3: **for** $k \leftarrow 1, p$ **do**
4: **if** $\mathcal{R}(x_i, a_j, c_k)$=1 **then**
5: $\mathcal{R}^{'}\ (x_i, (a_j \times c_k)) \leftarrow 1$
6: **else**
7: $\mathcal{R}^{'}\ (x_i, (a_j \times c_k)) \leftarrow 0$

	$\mathcal{P}$			$\mathcal{C}$		
$\mathcal{O}$	a_1	a_2	a_3	c_1	c_2	c_3
x_1	1	1	0	1	0	1
x_2	1	0	0	1	0	0
x_3	1	0	1	1	0	0
x_4	0	0	1	0	1	0
x_5	1	1	1	1	0	1
x_6	1	1	0	1	0	1

	$\mathcal{M}$=$\mathcal{C}$×$\mathcal{P}$								
$\mathcal{O}$	c_1-a_1	c_1-a_2	c_1-a_3	c_2-a_1	c_2-a_2	c_2-a_3	c_3-a_1	c_3-a_2	c_3-a_3
x_1	1	1	0	0	0	0	1	1	0
x_2	1	0	0	0	0	0	0	0	0
x_3	1	0	1	0	0	0	0	0	0
x_4	0	0	0	0	0	1	0	0	0
x_5	1	1	1	0	0	0	1	1	1
x_6	1	1	0	0	0	0	1	1	0

Table 2. Example of flattening a 3D multi-label data (left side table) into a 2D formal context (right side table).

Clearly, flattening the 3D formal context ensures that in the new 2D formal context, any formal concept includes a set of objects that are all predicted in the same classes under the same conditions (features values). This allows using existing algorithms and implementations to generate formal concepts directly for explanation purposes. Once the flattened formal context built, one can use it to provide different forms of explanations.

3.2 Computing explanations as formal concepts

In order to compute explanations, let us first adapt the derivation operator to our formal context. Assume we are given a 2D formal context and that our objective is to provide local or global explanations. Let $\mathcal{K}^{'}$=($\mathcal{O}$,$\mathcal{M}$,$\mathcal{R}^{'}$) a formal context, $\mathcal{O}$ a set of instances, $\mathcal{M} = (\mathcal{C} \times \mathcal{P})$ is a set of pairs (c_i, a_j) with $a_j \in \mathcal{P}$, $c_k \in \mathcal{C}$. The triplet $(x_i, c_k, a_j) \in \mathcal{R}^{'}$ means that the instance $x_i \in \mathcal{O}$ has the class $c_k \in \mathcal{C}$ when the attribute $a_j \in \mathcal{P}$ is present.
For all $X \subseteq \mathcal{O}$ and $Y \subseteq \mathcal{M}$ with $Y = \{(c_k, a_j)/a_j \in \mathcal{P}, c_k \in \mathcal{C}\}$ we define the Galois derivation set operator $(.)^{\Delta}$, seen in the previous section, as follows:
$X^{\Delta} = \{(c_k, a_j) \in \mathcal{M} \mid X \subseteq \mathcal{R}'(c_k, a_j)\}$
$= \{(c_k, a_j) \in \mathcal{M} \mid \forall x \in \mathcal{O} x \in X \Rightarrow (x, (c_k, a_j)) \in \mathcal{R}'\}$
$Y^{\Delta} = \{x \in \mathcal{O} \mid Y \subseteq \mathcal{R}^{'}(x)\}$
$= \{x \in \mathcal{O} \mid \forall (c, a) \in \mathcal{M}((c, a) \in Y \Rightarrow (x, (c_k, a_j)) \in \mathcal{R}^{'}\ )\}$.

X^{Δ} is the set of pairs (c_k, a_j) common to all the objects of X and Y^{Δ} is the set of objects having all the pairs (c_k, a_j) of Y. The obtained set of all formal concepts is named $\mathcal{L}(K')$

Example 3. Following the new definition of the Galois operator to the formal context of Table 2, we obtain the set $\mathcal{L}(K')$ of the following formal concepts:
$fc_1 : \langle\{x_1, x_2, x_3, x_4, x_5, x_6\}, \{\}\rangle$
$fc_2 : \langle\{x_4\}, \{(c_2, a_3)\}\rangle$
$fc_3 : \langle\{x_1, x_2, x_3, x_5, x_6\}, \{(c_1, a_1)\}\rangle$
$fc_4 : \langle\{x_1, x_5, x_6\}, \{(c_1, a_1), (c_1, a_2), (c_3, a_1), (c_3, a_2)\}\rangle$
$fc_5 : \langle\{x_3, x_5\}, \{(c_1, a_1), (c_1, a_3)\}\rangle$
$fc_6 : \langle\{x_5\}, \{(c_1, a_1), (c_1, a_2), (c_1, a_3), (c_3, a_1), (c_3, a_2), (c_3, a_3)\}\rangle$
$fc_7 : \langle\{\}, \{(c_1, a_1), (c_1, a_2), (c_1, a_3), (c_2, a_1), (c_2, a_2), (c_2, a_3), (c_3, a_1), (c_3, a_2), (c_3, a_3)\}\rangle$

The formal concept fc_4 can be rewritten as follows fc_4:$\langle\{x_1, x_5, x_6\}$, $\{c_1, c_3\}$, $\{a_1, a_2\}\rangle$ meaning that the classes $\{c_1, c_3\}$ are predicted for the instances $\{x_1, x_5, x_6\}$ when attributes $\{a_1, a_2\}$ are present. In this example, $\{x_1, x_5, x_6\}$, $\{c_1, c_3\}$, $\{a_1, a_2\}$ represent respectively, the extension, the intention and the condition of the formal concept fc_4.

Reducing the number of formal concepts The number of formal concepts can be very large, then a question arises regarding the relevance of some formal concepts for explanation purposes and whether one can not reduce their number. The particularity of our formal concepts reduction algorithm (Algorithm 2) lies in the fact that it does not only rely on attributes and instances but also on classes, as a third parameter, in the reduction process. For instance, in Example 3, the intention $\{c_1\}$ appears in several formal concepts:
- In fc_3: the class c_1 is predicted for the objects x_1, x_2, x_3, x_4, x_5 and x_6 when the attribute a_1 is present.
- In fc_5: the class c_1 is predicted for the object x_3 and x_5 when the attributes a_1 and a_3 are present.
Clearly the attribute a_1 is sufficient to predict the object x_3 and x_5 in the class c_1 since $\{x_3,x_5\}\subseteq\{x_1,x_2,x_3,x_5,x_6,x_7\}$ and $\{a_1\}\subseteq \{a_1, a_3\}$. Hence, the attribute a_3 is not a necessary condition to predict c_1 for objects $\{x_1,x_5\}$. Then from an explanation point of view, a_3 is not relevant and it suffices to keep only the formal concept fc_3. Based on this observation, Algorithm 2 allows to reduce the number of formal concepts: Let $\mathcal{L}(K')=\{fc_1, fc_2, ..., fc_n\}$ be the set of all formal concepts, $\mathcal{I}nt(fc_i)$ be the intention of the formal concept fc_i, $\mathcal{E}xt(fc_i)$ be the extension of the formal concept fc_i, and $\mathcal{C}ond(fc_i)$ be it condition.

Example 4. Applying Algorithm 2 to the set $\mathcal{L}(K^{'})$ of formal concepts obtained above (Example 3), we obtain the following reduced set of formal concepts $\mathcal{L}'(K^{'})=\{fc_2, fc_3, fc_4\}$.

Up to now, we showed how to build a formal context for explanation purposes and how to reduce the number of the formal concepts. The following section presents explanation generation from the obtained formal concepts.

Algorithm 2 Formal concepts reduction

Require: $\mathcal{L}(K')=\{fc_1, fc_2, ..., fc_n\}$
Ensure: $\mathcal{L}'(K')$ ▷ Reduced set of formal concepts
1: **for** $i \leftarrow 1, n-1$ **do**
2: **for** $j \leftarrow i+1, n$ **do**
3: **if** $\mathcal{I}nt(fc_i) = \mathcal{I}nt(fc_j)$ **then**
4: **if** $\mathcal{E}xt(fc_i) \subseteq \mathcal{E}xt(fc_j)$ **then**
5: **if** $\mathcal{C}ond(fc_i) \supseteq \mathcal{C}ond(fc_j)$ **then**
6: $\mathcal{L}'(K')=\mathcal{L}(K') \setminus fc_i$

4 Multi-label prediction explanations

Recall that our approach for explaining multi-label classification is agnostic and it allows to provide both *symbolic* and *numerical* of explanations. For lack of space, the presentation is limited to some forms of *symbolic* explanations.

Example 5. In this section, we will illustrate through a real example from the medical field. This example deals with diagnosing some respiratory deceases (labels) given some symptoms of patients (attributes). For the sake of simplicity, we consider only four deceases that are: Asthma (d_1), Bronchiolitis (d_2), COPD (d_3), Covid (d_4). It has been found that the manifestation of these diseases is generally made by the following symptoms: Dry cough (s_1), loose cough (s_2), shortness of breath (s_3), wheezing (s_4), fever(s_5), headaches (s_6), loss of taste (s_7), curvatures (s_8) and Dyspnoea (s_9). Table 3 presents the predictions of a black-box model f.

	Symptoms									*Diseases*			
$\mathcal{O}$	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	d_1	d_2	d_3	d_4
1	1	0	1	1	1	0	0	0	0	1	1	0	0
2	0	1	0	1	1	0	0	1	1	1	0	1	0
3	1	0	1	0	1	1	1	1	0	1	0	0	1
4	1	0	0	0	1	1	1	1	0	0	0	0	1
5	0	1	1	0	0	0	0	1	1	0	0	1	0
6	1	0	1	0	1	1	1	1	0	1	1	0	1
7	0	0	0	0	1	1	1	1	0	0	0	0	1
8	1	0	1	1	1	1	0	1	0	1	1	0	1
9	1	0	1	0	0	0	0	1	0	0	1	0	0
10	1	0	1	0	1	1	1	1	0	0	1	0	1
11	0	1	1	1	1	0	0	0	1	1	1	1	0
12	0	1	0	0	1	0	0	0	1	0	0	1	0
13	1	0	1	0	1	1	1	1	0	0	0	0	1
14	1	0	1	1	0	0	0	0	0	1	0	0	0

Table 3. Multi-label data from medical field

Table 3 contains data for 14 patients (instances). Each patient is described by a set of symptoms and the model f diagnosed him with a few diseases (compatible and

probable for the observed symptoms). After transforming these multi-label data into a formal context, as shown in the previous section, we obtain the following reduced formal concept set $\mathcal{L}'(K')$:
-fc_1:$\langle\{11\}, \{'Asthma', 'Bronchiolitis', 'COPD'\}, \{'fever', 'Dyspnoa', 'Shortnessofbreath', 'Wheezing', 'Loosecough'\}\rangle$
-fc_2:$\langle\{2, 11\}, \{''Asthma', 'COPD'\}, \{'Fever', 'Loosecough', 'Dyspnoa', 'Wheezing'\}\rangle$
-fc_3 : $\langle\{8, 6\}, \{'Asthma', 'Bronchiolitis', 'Covid'\}, \{'Fever', 'Headache', 'Shortnessofbreath', 'Drycough', 'Curvature'\}\rangle$
-fc_4:$\langle\{8, 3, 6\}$,{'Asthma', $'Covid'\}, \{'Fever', 'Headache', 'Shortnessofbreath', 'Drycough', 'Curvature'\}\rangle$
-fc_5 :$\langle\{8, 10, 6\}, \{'Bronchiolitis', 'Covid'\}, \{'fever', 'Headache', 'Shortnessofbreath', 'Drycough', 'Curvature'\}\rangle$
-fc_6 : $\langle\{8, 1, 11, \}, \{'Asthma', 'Bronchiolitis''\}, \{'Fever', 'Shortnessofbreath'\}\rangle$
-fc_7 : $\langle\{2, 11, 12, 5\}, \{'COPD''\}, \{'Loosecough', 'Dyspnoa'\}\rangle$
-fc_8 : $\langle\{1, 2, 8, 11, 14\}, \{'Asthma''\}, \{'wheezing'\}\rangle$
-fc_9 : $\langle\{1, 2, 3, 6, 8, 11\}, \{'Asthma'\}, \{'Fever'\}\rangle$
-fc_{10} : $\langle\{1, 3, 6, 8, 11, 14\}, \{'Asthma'\}, \{'Shortnessofbreath'\}\rangle$
-fc_{11} : $\langle\{1, 6, 8, 9, 10, 11\}, \{'Bronchiolitis'\}, \{'Shortnessofbreath'\}\rangle$
-fc_{12} : $\langle\{3, 4, 6, 7, 8, 10, 13\}, \{'Covid\}, \{'Fever', 'Headache', 'Curvature'\}\rangle$.

Given a data instance x and the prediction $f(x)$, we are first interested in symbolic explanations that are *sufficient reasons*. Broadly speaking, they refer to the subset of features in x that are sufficient to predict $y{=}f(x)$.

4.1 Sufficient reason explanations

In FCA terms, a sufficient reason corresponds to the concept of a *decisive attribute set (Das)*. A $\mathcal{D}as$ is the smallest subset of features $a_j{\in}\mathcal{P}$ that allows a model f to predict $y{\subseteq}\mathcal{C}$. This subset is said to be decisive in the sense that this decision remains valid regardless of the values of the other attributes.
Let $x{=}(a_1, a_2, ..., a_n)$ be the instance to classify and $c_k{\in}y{\subseteq}C$ be a class predicted among the labels composing the multi-label prediction $y{=}f(x)$ where f is the multi label classifier to explain. First, we will define the set $\mathcal{B}(c_k)$ of all the formal concepts having in their intentions the class c_k. Let $\mathcal{L}^{'}(\mathcal{K}^{'}){=}\{fc_1, fc_2, ...\}$ be the set of all formal concepts obtained after reduction. Namely, $\mathcal{B}(c_k){=}\{fc_i{\in}\mathcal{L}^{'}(\mathcal{K}^{'})|c_k{\in}\mathcal{I}nt(fc_i)\}$. The $\mathcal{D}as$ that allows the model f to predict the classes c_k is defined as follows: Let $\mathcal{B}(c_k)$ be the set of all formal concepts having in their intentions the class c_k, $\mathcal{E}xt(fc_i){=}\{x_i \in \mathcal{O}|fc_i \in \mathcal{B}(c_k)\}$ be the extension of the formal concept whose intention contains the class c_k, $\mathcal{C}ond(fc_i){=}\{a_i \in \mathcal{P}|fc_i \in \mathcal{B}(c_k)\}$ be the condition of the formal concept whose intention contains the class c_k, and $\mathcal{D}iff(c_k){=}\bigcup_{i=1,n} \mathcal{E}xt(fc_i) \setminus \bigcap_{i=1,n} \mathcal{E}xt(fc_i)$ is the set difference between the union and the intersection of extensions having the class c_k in intention. The decisive Attribute Set ($\mathcal{D}as$) for the class c_k is given as follows :

$$\mathcal{D}as_j(c_k){=}\{\bigcup \mathcal{C}ond(fc_i)|x_j \in \mathcal{E}xt(fc_i) \text{ and } x_j \in \mathcal{D}iff(c_k)\}$$

The following algorithm summarizes this procedure :

Algorithm 3 Compute $\mathcal{D}as(c_k)$

Require: $\mathcal{B}(c_k)$
Ensure: $\mathcal{D}as(c_k)$
1: $\mathcal{D}as(c_k) = \emptyset$
2: **if** $|\mathcal{B}(c_k)|$=1 **then**
3: $\quad\mathcal{D}as(c_k) = \mathcal{C}ond(fc_k)$
4: **else**
5: $\quad$**for** $i \leftarrow 1, n$ **do**
6: $\quad\quad\mathcal{D}iff(c_k)$=$\bigcup \mathcal{E}xt(fc_i)\setminus \bigcap\mathcal{E}xt(fc_i)$
7: $\quad\quad$**for each** $x \in \mathcal{D}iff(c_k)$ **do**
8: $\quad\quad\quad$**for** $i = 1 \leftarrow 1, n$ **do**
9: $\quad\quad\quad\quad$**if** $x \in \mathcal{E}xt(fc_i)$ **then**
10: $\quad\quad\quad\quad\quad\mathcal{D}as_x(c_k)$=$\bigcup \mathcal{C}ond(fc_i)$
11: $\quad\quad\quad\quad\quad$**if** $\mathcal{D}as_x(c_k) \notin \mathcal{D}as(c_k)$ **then**
12: $\quad\quad\quad\quad\quad\quad\mathcal{D}as(c_k) = \mathcal{D}as(c_k) \cup \mathcal{D}as_x(c_k)$ $\quad\triangleright$ Add a subset to $\mathcal{D}as(c_k)$)

It should be noted that a decisive attribute set ($\mathcal{D}as$), which allows to predict a class c_k, is not necessarily unique and that it is possible to find several decisive attribute sets which allow the same class to be predicted. The set $\mathcal{D}as(c_k)$ is made up of subsets $\mathcal{D}as_x(c_k)$ such that $x \in \mathcal{D}iff(c_k)$.

Example 6. Let us continue our running example. Assume we want to compute the set of all decisive attribute sets that allow f to predict the class c_k=$\{Covid\}$ (label $Covid$ is also denoted (d_4)). First, select the formal concepts having in their intention the class c_k= $\{Covid\}$: $\mathcal{B}(\{Covid(d_4)\})$=$\{fc_3, fc_4, fc_5, fc_{12}\}$. Then, we compute the set $\mathcal{D}iff(Covid)$=$\bigcup\mathcal{E}xt(fc_i)\setminus\bigcap\mathcal{E}xt(fc_i)$ with $fc_i \in \mathcal{B}(\{Covid(d_4)\})$: $\mathcal{D}iff(Covid)$=$\{3, 4, 7, 10, 13\}$. For each instance x∈$\mathcal{D}iff(Covid)$, we compute the set $\mathcal{D}as_x(Covid)$. For objects $\{10, 3\}$, $\mathcal{D}as_{10}(Covid)$ =Das$_3(Covid)$ we write: $\mathcal{D}as_{10,3}(Covid)$=$\{'Fever', 'Headache', 'Shortnessofbreath', 'Drycough', 'Curvature'\}$. For objects $\{4, 13, 7\}$, $\mathcal{D}as_{4,13,7}(Covid)$=$\{'Fever', 'Headache', 'Curvature'\}$. Hence the set of all decisive attribute sets allowing the model f to predict the class y=$\{$ Covid$\}$ is : Das(Covid)=$\{$Das$_{4,13,7}(Covid), \mathcal{D}as_{10,3}(Covid)\}$

4.2 Significant attributes set

A significant attributes set for a class c_k (denoted $\mathcal{S}sa(c_k)$) is the set of attributes that appear in at least in one decisive attribute set. Namely, $\mathcal{S}sa(c_k)$ is equivalent to the set of attributes appearing in the union of all decisive attributes sets for this class ($\mathcal{D}as(c_k)$). The significant attributes set for a set of classes y is the union of all significant attributes set of the set of classes:

$$\mathcal{S}sa(Y)=\bigcup_{c_k\in y}\mathcal{S}sa(c_k)$$

Example 7. The significant attributes set for y=$\{Covid\}$ is the union of all decisive attributes sets of the class $Covid$. Thus,

$\mathcal{S}sa(Covid)$=$\{'Fever', 'Headache', 'Shortnessofbreath', 'Drycough', 'Curvature'\}$, same for the class $COPD$ $\mathcal{S}sa(COPD)$=$\{'Wheezing', 'fever', 'Dyspnoa', 'Loosecough'\}$, and $\mathcal{S}sa(covid, COPD)$=$\mathcal{S}sa(covid) \cup \mathcal{S}sa(COPD)$= $\{'Fever', 'Headache', 'Shortnessofbreath', 'Drycough', 'Curvature', 'Wheezing', 'Dyspnoa', 'Loosecough'\}$

4.3 Beyond $\mathcal{D}as$ and $\mathcal{S}sa$ explanations

For lack of space, we have limited our presentation to $\mathcal{D}as$ and $\mathcal{S}sa$ explanations. However, our approach can go further to give other types of explanations. For instance, it can directly provide scoped rules commonly known in explainable AI as *Anchors* [11] (rules are in the form *IF feature1=1 AND feature2=1.. THEN the prediction is y*). The necessary features to trigger a prediction are simply the intersection of $\mathcal{D}as$. Moreover, in addition to symbolic explanations, our FCA-based approach can also provide numerical explanations such as feature importance which can be assessed through the frequency of features in sufficient reasons for instance.

5 Case study

We present in this section a case study on the well-known Stack Overflow collection of coding questions and answers. It features questions and answers on a wide range of topics in computer programming. Based on the type of tags assigned to questions, the most discussed topics on the site are: Java, PHP, Android, Python, HTML, etc. A question in Stack Overflow contains three segments: *Title, Description* and *Tags* as illustrated in Fig. 2. We consider a prediction task

	Title	Tags
0	Flask-SQLAlchemy - When are the tables/databas...	['python', 'mysql']
1	Combining two PHP variables for MySQL query	['php', 'mysql']
2	Counting' the number of records that match a c...	['php', 'mysql']
3	Insert new row in a table and auto id number. ...	['php', 'mysql']
4	Create Multiple MySQL tables using PHP	['php', 'mysql']
...	...	...

Fig. 1. Illustration from Stack Overflow collection

consisting in predicting the tags (labels) from the title and description of the question. We selected five hundred questions that were preprocessed into 100 binary features and 5 labels (C=$\{Asp.net.1, java.1, mysql.1, php.1, python.1\}$. In our study, we considered three well-known multi-label techniques that are *Label powerset (LPS)*, *MLkNN* and *RAKEL* classifiers which represent the three multi-label classification methods, namely, the problem transformation method, the problem adaptation method and that combines both methods, respectively

(note that in our approach a classifier is considered as a black box). For each of them, we computed the decisive attributes sets (Dfs) as well as the percentage of significant attributes (PF) for each class.

Table 4. Number of decisive feature sets per class (nbDfs) and percent of significant attributes per class(Psf)

$nbDfs$	$Asp.Net.1$	$java.1$	$Mysql.1$	$Php.1$	$Python.1$
$MLKNN$	1	1	78	37	29
LPS	1	1	76	33	26
$RAKEL$	1	1	76	51	40
Psf	$Asp.Net.1$	$java.1$	$Mysql.1$	$Php.1$	$Python.1$
$MLKNN$	1%	1%	53%	29%	18%
LPS	1%	1%	51%	25%	19
$RAKEL$	1%	1%	50%	33%	21%

From these results, we notice that the number of decisive feature sets (nbDfs) as well as the percentage of significant features for each class differs from one classifier to another. This means that each of the Rakel and MLkNN classifiers (considered as black-boxes) rely on different attributes for their predictions concerning a given class or a set of given classes. For example, for the php.1 class the MLkNN classifier used 29% of the features with 37 decisive feature sets, RaKel classifier used 33% of the features with 51 decisive feature sets, while the LPS classifier only used 25% of the features with 33 decisive feature sets.

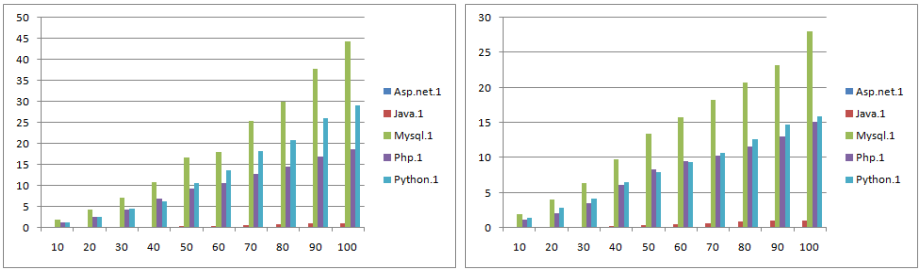

Fig. 2. Average decisive attribute set number per class (left side) and average significant attributes number per class (right side). On the X axis the size (# of instances) of the neighborhood considered (experiments are done of 10,20,...,100 neighbors).

5.1 Concluding remarks

In this preliminary work, we proposed an approach based on formal concept analysis to explain multi-label classification, and this by defining the minimal

attribute subsets allowing a multi-label classifier to make a given prediction. We have also defined the set of all significant attributes that influence the model predictions concerning a class or a set of classes. As a perspective, we intend to measure the importance of each attribute in the prediction of a set of classes. This work, can also be extended to counterfactual explanations to define, for example, the smallest perturbation (modification) of attribute values that modifies the predictions into a predefined output.

References

1. M Berrada. A Adadi. Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access, 2018 - ieeexplore.ieee.org.*, 2018.
2. Tzima F. Mitkas P. Allamanis, M. Effective rule-based multi-label classification with learning classifier systems. In Adaptive and natural computing algorithms. *11th international conference, ICANNGA (pp. 466–476), .*, 2013.
3. R. Wille. B. Ganter. Formal Concept Analysis. *Springer-Verlag*, 1999.
4. Jurgen Bajorath. Interpretation of machine learning models using shapley values:application to compound potency and multi-target activity predictions. *Journal of Computer-Aided Molecular Design (2020) 34:1013–1026*, 2 May 2020.
5. Luo J. Shen X. Boutell, M. R. and C. M. Brown. Learning multi-label scene classification. *Learning multi-label scene classification.*, 2004.
6. Yan J. Zhang B. Chen Z. Chen, W. and Q. Yang. Document Trans-formation for Multi-label Feature Selection in Text Categorization. *In Seventh IEEE International Conference on Data Mining, IEEE, pp. 451–456.*, 2007.
7. Sen S. Datta, A. and Y. Zick. Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems. In Proc. 2016 IEEE Symp. Secur. Priv. (SP 2016), pp. 598–617. IEEE, 2016.
8. Dominique Guegan. A Note on the Interpretability of Machine Learning Algorithms ,. July 6, 2020.
9. Makhazhanov N. Ushakov M. Kuznetsov, S.O. On neural network architecture based on concept lattices.
10. Varshney K. R. Emad A. Malioutov, D. M. and S. Dash. Learning Interpretable Classification Rules with Boolean Compressed Sensing. Transparent Data Min. Big Small Data. Stud. Big Data, 32, 2017. doi: 10.1007/ 978-3-319-54024-5. .
11. Carlos Guestrin. Marco Tulio Ribeiro., Sameer. S. "Anchors: high-precision model-agnostic explanations". *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.
12. Emamirad K. Missaoui, R. Lattice Miner 2.0 : A Formal Concept Analysis Tool. *In Supplementary Proc. of ICFCA, Rennes, France, 2017 (2017), pp. 91–94.*, 2017.
13. Dosovitskiy A. Yosinski J. Brox T. Nguyen, A. and J. Clune. Synthesizing the Preferred Inputs for Neurons in Neural Networks via Deep Generator Networks. Adv. Neural Inf. Process. Syst., 29. 2016.
14. Hua X.-S. Rui Y. Tang J. Mei T. Qi, G.-J. and H.-J. Zhang. Correlative multi-label video annotation. *In Proceedings of the 15th international con-ference on Multimedia - MULTIMEDIA '07, ACM Press, p. 17.*, 2007.
15. S. Ruggieri F. Turini D. Pedreschi R. Guidotti, A. Monreale and F. Giannotti. A survey of methods for explaining black box models.arXiv preprint. *arXiv: 1802.01933.*, 2018.

16. Meddouri N. Maddouri M.. Trabelsi, M. New taxonomy of classification methods based on Formal Concepts Analysis. *What can FCA do for Artificial intelligence,113-120*, 2016.
17. Y. Zhou and G. Hooker. Interpreting Models via Single Tree Approximation. arXiv preprint arXiv:1610.09036. 2016.