Explaining Meta-Features Importance in Meta-Learning Through Shapley Values

Moncef Garouani, Adeel Ahmad and Mourad Bouneffa
Univ. Littoral Côte d'Opale, LISIC, Laboratoire d'Informatique Signal et Image de la Côte d'Opale, France

Keywords: Explainable Artificial Intelligence, Meta-Learning, Shapley Values, Autoencoder, Meta-Features Importance.

Abstract: Meta-learning, or the ability of a machine learning model to adapt and improve on a wide range of tasks, has gained significant attention in recent years. A crucial aspect of meta-learning is the use of meta-features, which are high-level characteristics of the data that can guide the learning process. However, it is a challenging task to determine the importance of different meta-features in a specific context. In this paper, we propose the use of Shapley values as a method for explaining the importance of meta-features in meta-learning process. Whereas, Shapley values is a well-established method in game theory. It has been used for fair distribution of payouts among a group of individuals, based on the separate contribution of meta-features to the overall payout. Recently, these have been also applied to machine learning to understand the contribution of different features in a model's prediction. We observe that a better understanding of meta-features, using the Shapely values, can be gained to evaluate their importance. In the context of meta-learning it may aid to improve the performance of the model. Our results demonstrate that Shapley values can provide insight into the relative importance of different meta-features and how they interact in the learning process. This can fairly optimize the meta-learning models, resulting in more accurate and effective predictions. Overall, this work conclude that Shapley values can be a useful tool in guiding the design of meta-features and these can be used to improve the performance of the meta-learning algorithms.

1 INTRODUCTION

Meta-learning, or learning to learn, refers to the process of adapting a machine learning (ML) model to a new task based on experience with similar tasks (Garouani. et al., 2021). Meta-learning algorithms are often useful in data-limited environments especially, when the data for a particular task keep changing over time (Nural et al., 2017). One key aspect of meta-learning is the use of meta-features, which are high-level characteristics of the data that can guide the learning process and aid in selecting the most appropriate learning algorithm. However, determining the most important meta-features for a specific context can be challenging (Garouani et al., 2023a). Recent research in the field of meta-learning has focused on identifying the most important features in the data, but the process of involving or discarding a family of meta-features can still be difficult (Garouani et al., 2023a; Alcobaça et al., 2020). Hence, understanding the importance of meta-features in meta-learning is crucial for effectively designing and deploying meta-learning algorithms, but this still requires further research and investigation.

Meta-features are high-level characteristics of a dataset that play a key role in the meta-learning process. These features, such as the size of the dataset, the complexity of the data, and the number of classes, help meta-learning algorithms to determine which machine learning algorithms are most suitable for a given dataset. However, without a proper understanding of the importance of these meta-features, it can be challenging to effectively design and deploy meta-learning algorithms. This lack of understanding can lead to sub-optimal performance and difficulty in interpreting the results of meta-learning algorithms, as highlighted in recent research studies (Shao et al., 2022; Garouani et al., 2022c; Shao et al., 2023). Therefore, gaining insight into the importance of meta-features in meta-learning is crucial for optimizing the performance and interpretability of meta-learning algorithms.

Our review of the literature on meta-feature importance in meta-learning revealed that while there is a growing interest in this topic, the field is still in its early stages (Shao et al., 2023; Shao et al.,

Garouani, M., Ahmad, A. and Bouneffa, M.
Explaining Meta-Features Importance in Meta-Learning Through Shapley Values.
DOI: 10.5220/0011986600003467
In *Proceedings of the 25th International Conference on Enterprise Information Systems (ICEIS 2023) - Volume 1*, pages 591-598
ISBN: 978-989-758-648-4; ISSN: 2184-4992

2022; Woźnica and Biecek, 2020). One common approach to improve the explainability of meta-learning is the use of interpretable models, such as decision trees or XGBoost (Woźnica and Biecek, 2020). However, these models often do not achieve the same level of performance as more complex models, which can be a trade-off for improved explainability (Garouani, 2022). Other methods to gain insight into meta-feature importance include the use of visualization techniques and the development of more interpretable meta-features (Garouani et al., 2023a; Samek et al., 2019). As can be witnessed in the literature, there are various approaches to better understand meta-feature importance, the field is yet evolving and there is a need for further research and developments to optimize the performance and interpretability of meta-learning algorithms.

Among others, Shapley Values (Lundberg and Lee, 2017a) has achieved a great popularity in recent years. It is a method of attributing the importance to different features or variables in a model or decision-making process. These values have been used to understand the contribution of different features to a model's prediction. These are often used to identify the most important features in a model, or to understand how different features interact to influence the prediction (Olsen et al., 2022). Shapley values are calculated by considering all possible combinations of features and averaging the predicted output change when a particular feature is removed. This allows Shapley values to capture the marginal contribution of each feature, taking into account the interactions between features. In the context of meta-learning, Shapley values can be used to explain the importance of meta-features that are used to help a machine learning model to learn from other models or datasets. Shapley values can be used to assign a numerical value to each meta-feature, indicating its relative importance in the meta-learning process. This can be helpful in understanding which meta-features are most important in determining the success of a meta-learning algorithm, and can help inform decisions about which meta-features to prioritize when designing meta-learning algorithms.

In this paper, we present a new method based on Shapley values, to explain the meta-features importance in meta-learning context. The method could be beneficial to gain a better understanding of which meta-features are most important for improving the performance of the model or in contrast which ones may be less important. This can help to better design and optimize meta-learning models, resulting in more accurate and effective predictions. The contributions of our work to the field are as follows :

- We developed a method to explain meta-features importance revealed by an autoencoder-KNN meta-model. The method explains the meta-features with the highest reconstruction errors using Shapley values. This is the first study that uses a model-agnostic method to explain meta-feature selection in meta-learning to the best of our knowledge.
- We conducted a preliminary experiment with real-world meta-learning environment on 400 real word datasets.

The rest of this paper is organized as follows : an overview on meta-learning for the automatic algorithms selection and Shapley values for explaining features contribution is given in Section 2. The motivation behind the proposed approach is detailed in section 3. The proposed explanatory approach is described In section 4, while the Section 5 describes the experiments illustrating the effectiveness of the proposed approach. Finally, section 6 provides the brief conclusion and points out the directions for the future work.

2 RESEARCH BACKGROUND

2.1 Meta-Learning

Meta-learning, also known as "learning to learn," is a sub-field of machine learning that focuses on the development of algorithms that can adapt and improve their performance over time through experience (Garouani et al., 2022d). The goal of meta-learning is to enable machine learning systems to acquire new skills or knowledge more efficiently, by leveraging the information learned from previous learning tasks. This is in contrast to traditional ML approaches, which require a large amount of data and compute resources to learn a new task from scratch.

The challenge in meta-learning involves using prior experiences in a systematic and data-driven way to improve the performance of machine learning algorithms on new tasks. This process, illustrated in figure 1, has three main phases : first, a meta-learning space is created using meta-data that describes prior learning tasks and previously learned models. This includes characteristics of the datasets and a performance measure (meta-responses) for data mining algorithms. Next, a predictive meta-model is generated from the meta-dataset to extract and transfer knowledge that guides the search for optimal models for new tasks. Finally, when a new dataset arises, its characteristics are extracted and the predictive meta-

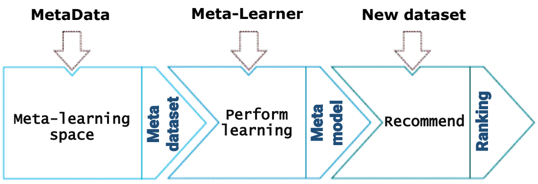

Figure 1: The meta-learning process.

model is used to recommend the most promising ML algorithms with related HPs configurations.

Meta-learning algorithms typically operate by learning a meta-representation of the gained knowledge from previous tasks, which can then be used to quickly adapt to new tasks with only a small amount of additional data (Nural et al., 2017; Garouani et al., 2022b). This meta-representation can take many forms, such as a set of weights shared across multiple tasks, a set of task-specific optimization algorithms, or a high-level representation of the structure of the tasks themselves (Kalousis and Hilario, 2001). Meta-learning has the potential to greatly improve the efficiency of machine learning in a number of applications, including transfer learning, continual learning, and multi-task learning. It has been applied to a wide range of tasks, including natural language processing (Garouani and Zaysa, 2022), computer vision (Bennequin, 2019), adaptive artificial intelligence and automated machine learning (Garouani et al., 2022a; Garouani et al., 2022d) with promising results.

2.1.1 Data Characterization

In meta-learning, data characterization refers to the process of understanding and describing the properties of the data that can be used for meta-learning. This includes understanding the distribution of the data, the relationship between different features or variables, and any patterns or trends that may exist within the data. Data characterization is important in meta-learning because it helps inform the design of the meta-learning algorithm and the choice of meta-representation (Kalousis and Hilario, 2001). For example, if the data exhibits strong patterns or trends, the meta-representation may need to be able to capture these in order to effectively learn from the data. On the other hand, if the data is highly variable or unpredictable, the meta-representation may need to be more flexible in order to adapt to these changes.

Meta-features, also known as auxiliary features or side information, are additional features that are used in the meta-learning process to make predictions about the performance of machine learning algorithms on a new task. These features can include properties of the data itself, such as the number of samples, the dimensions, and the noise level, as well as characteristics of the learning algorithm, such as its time and space complexity. Meta-features can be extracted from both the training and test sets, and are used to inform the selection of the most appropriate machine learning algorithm for a given task. In addition to informing the design of the meta-learning algorithm, meta-features can also help to identify any potential challenges or biases in the data that may impact the performance of the meta-learning system. This can be especially important in applications where the data may be highly imbalanced or may contain sensitive information. Overall, data characterization plays a crucial role in the success of meta-learning systems, as it helps to ensure that the meta-learning algorithm is well-suited to the characteristics of the data and can effectively learn from it.

One recent trend in the use of meta-features in meta-learning has been the development of algorithms that can learn to automatically select the most relevant meta-features for a given task. These methods can be trained on a large dataset of tasks and meta-features, and use this information to select the most predictive meta-features for a new task. There has also been a focus on the use of meta-features for lifelong learning, where a machine learning system continually learns from new tasks and experiences. In this setting, meta-features can be used to prioritize which tasks should be learned first, or to identify when it is necessary to transfer knowledge from previous tasks to a new one.

2.2 Shapley Values

Shapley values are a mathematical concept used to distribute the "importance" or "influence" of each feature in a machine learning model among all the features (Hart, 1989). They were developed by Lloyd Shapley, a Nobel laureate in economics, and are often used in the field of game theory. The basic idea behind Shapley Values is to assign a value to each member of the group based on the contributions they make to the overall group. The values are calculated using a complex mathematical formula, which takes into account the number of members in the group, the number of resources available, and the relative contributions of each member.

In the context of machine learning, Shapley values can be used to explain the contribution of each feature to the model's predictions, or to identify the most important features in the model. They can be calculated using the formula 1 :

$$\phi_i = \sum_{S \subseteq N \setminus i} \frac{|S|!(|N|-|S|-1)!}{|N|!}[f(S \cup i) - f(S)] \quad (1)$$

Where, N is the set of all features in the model, i is a specific feature, S is a subset of N that does not include i, and $f(S)$ is the prediction made by the model when using only the features in S.

The Shapley value for feature i is the average of the difference that adding feature i makes to all possible subsets of features. This can be computationally expensive to calculate, especially for large models with many features. However, there are approximate methods that can be used to compute Shapley values more efficiently. Shapley values have a number of desirable properties, such as being fair (they respect the symmetry of the model's predictions) and being able to handle both categorical and continuous features. They are often used in combination with techniques like feature selection and model interpretation to better understand the behavior of ML models.

2.2.1 Shapley Values in Model Explanation

Shapley Values can be used to explain the individual contributions of each feature to a model's predictions. For example, suppose we have a machine learning model that predicts the price of a house based on a number of features such as the size of the house, the location, the number of bedrooms, etc. We can use Shapley values to determine the relative importance of each of these features in determining the final prediction. To do this, we can compute the Shapley values for each feature, and then sort the features according to their Shapley values. The features with the highest Shapley values will be the most important ones in determining the model's predictions.

Another way to use Shapley values for model interpretation is to compute them for a specific prediction made by the model. This can help us to better understand which features were most influential in causing the model to make that particular prediction (see Figure 2). Therefore, in addition to being used for model interpretation, Shapley values can also be useful for feature selection, where we try to identify the most important features to include in the model. By ranking features according to their Shapley values, we can identify the ones that have the biggest impact on the model's predictions and choose to include only those in the model, potentially leading to simpler and more interpretable models.

2.2.2 Data Characterization Importance

Meta-features play a crucial role in the meta-learning process, as they are used to inform the selection of the most appropriate machine learning algorithm for a given task. As discussed earlier, meta-features can be used to capture important characteristics of the data and the learning algorithm, hence these can be used to predict the performance of different algorithms on the task. By using meta-features to guide the selection of the learning algorithm, meta-learning can improve the efficiency and effectiveness of the machine learning process, particularly in cases where the task or data is not well understood.

The choice of meta-features depend on the specific problem and the available data, and it is often necessary to carefully select and engineer relevant meta-features in order to achieve good performance. Thence, meta-features are an important aspect of meta-learning that can significantly improve the efficiency and effectiveness of the ML process.

3 MOTIVATION

Explainability of meta-learning as an approach for automating the algorithms selection and parametrization process is important because it allows us to better understand why certain decisions were made and how they affect the performance of the system. Explainability makes it easier to debug, monitor, and improve upon existing models by providing insights into important research questions such as what works well or not so well in a particular context. Additionally, explainability can help identify areas where further research could be beneficial, such as which parameters are most influential in achieving optimal results. Eventually, understanding why certain algorithms work better than others may lead to more informed decisions when choosing which algorithm should be used for a given task.

In this paper, we propose a framework for explaining the importance of meta-features in meta-learning using Shapley values. As opposed to existing explainability methods which explain predictions (supervised), we develop a method for explaining a meta-feature vector revealed by an autoencoder meta-model (unsupervised). By using the autoencoder meta-model (Garouani et al., 2023b), the reconstruction error is used as a basis for extracting important meta-features from the learned representation. Those instances with high reconstruction error scores are considered unimportant. A reconstruction score is defined as the difference (error) between an input value and an output value (see Figure 3). If an unimportant meta-feature exists, it resides in the input values, and the explanatory model must explain why this instance did not reconstructed well, and the error must be connected to the explanation. Thus, our method computes the SHAP values of the reconstructed meta-features and compares them to their true input values.

Figure 2: SHapley Additive exPlanation.

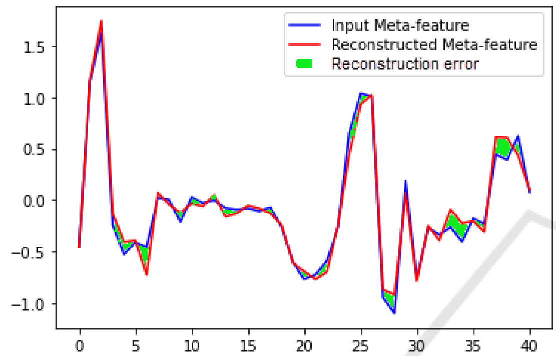

Figure 3: The reconstruction error of an instance from the meta-features set after it's encoded and decoded by the Autoencoder.

4 THE PROPOSED APPROACH

To calculate the Shapley values for meta-features in a meta-learning context, we would first need to define the meta-features that we want to evaluate and the meta-learning model that we may use to make predictions. Next, we would need to create a meta-dataset that contains the meta-features and the corresponding predictions made by the meta-learning model. Once we have this dataset, we can use the Shapley values to calculate the contribution of each meta-feature to the overall accuracy of the meta-learning model. The Shapley values formula takes into account the value of each meta-feature when it is included in the model and when it is excluded, as well as the interactions between different meta-features.

To construct the meta-dataset, we made use of the PyMFE tool (Alcobaça et al., 2020) for the *general*, *statistical*, *info-theoretical*, *model-based*, *landmarking*, and *data complexity* meta-features. These features are extracted from a large set of 400 datasets used in a meta-learning context in (Garouani, 2022; Garouani et al., 2022d). Consequently, we generate a meta-dataset of 400 meta-instances and 41 meta-feature (characteristics) that describe the datasets. The process of meta-features extraction is formalized by (Alcobaça et al., 2020) as the following function:

$$\mathcal{F} : \mathcal{D} \rightarrow \mathbb{R}^k$$

that receives a dataset $\mathcal{D}$ as input, and returns a features vector of k values characterizing the dataset, and that are predictive of algorithms performance when applied to the dataset.

Given an input meta-features instance M with a set of dataset characteristics $\{m_1, m_2, ..., m_n\}$ and its corresponding output M' and reconstructed values $\{m'_1, m'_2, ..., m'_n\}$, using an autoencoder model A, the reconstruction error of the instance is the sum of errors of each feature $L(M, M')$:

$$L(M,M') = \frac{1}{2}\sum_{i=1}^{n} \| m_i - m'_i \|_2^2 \, | i \in \{1, \ldots, n\} \tag{2}$$

Let

- $\{m_{(1)}, m_{(2)}, \ldots, m_{(n)}\}$ be a reordering of the features in *error list*, such that : $|m_{(1)} - m'_{(1)}| \geq ... \geq |m_{(n)} - m'_{(n)}|$
- $TopMetafeatures = \{m_1, \ldots, x_m\}$ contains a set of features for which the total corresponding errors $ErrorList : \{|m_1 - m'_1|, \ldots, |m_n - m'_n|\}$ represent an adjustable percent of $L(M, M')$.

By using SHAP values, we can explain which meta-features affected each of the *high* reconstruction errors in *TopMetafeatures*. Algorithm 1 presents the pseudo-code for the process. First, we extract the meta-features with the highest reconstruction error from the *ErrorList* and save them in the $TopMetafeatures$ list (line 5). Next, for each feature m'_i in $TopMetafeatures$, we use Kernel SHAP (Lundberg and Lee, 2017b) to obtain the SHAP values, i.e, the importance of each meta-feature $m_1, m_2, \ldots, m_n$ in predicting the examined feature m'_i. The result of this step is a list

Algorithm 1: The proposed algorithm's pseudo-code.

1: **Input :** A- autoencoder meta-model, M- set of meta-features
2: **Output :** Contributing, Restricting ▷ Lists of MF that contribute to / restrict the good recommendation
3: $M' \leftarrow A.predict(M)$ ▷ The technical study on the design of A is detailed in (Garouani, 2022)
4: $ErrorList \leftarrow (m_1 - m'_1)\ldots(m_n - m'_n)$
5: $TopMetaFeatures \leftarrow$`top values from` $ErrorList$
6: **for** each $i \in TopMetaFeatures$ **do**
7: $\quad explainer \leftarrow shap.KernelExplainer(A)$
8: $\quad ShapTopMetaFeatures[i] \leftarrow explainer.shapvalues(m_i)$
9: **for** each $i \in ShapTopMetaFeatures$ **do**
10: $\quad$ **if** $m_i > m'_i$ **then**
11: $\quad\quad Contributing[i] \leftarrow ShapTopMetaFeatures[i]$ **if** > 0
12: $\quad\quad Restricting[i] \leftarrow ShapTopMetaFeatures[i]$ **if** < 0
13: $\quad$ **if** $m_i < m'_i$ **then**
14: $\quad\quad Contributing[i] \leftarrow ShapTopMetaFeatures[i]$ **if** < 0
15: $\quad\quad Restricting[i] \leftarrow ShapTopMetaFeatures[i]$ **if** > 0

ShapTopMetaFeatures, in which each row represents the SHAP values for one meta-feature from the *ErrorList*.

We divide the SHAP values into values contributing to the good recommendation - those pushing the predicted (reconstructed) value towards the true value, and values restricting the good recommendation - those pushing the predicted value away the true value. For each feature (line 9), we check if the input meta-feature value is greater than the reconstructed one (line 10); the contributing meta-features are the features with a positive SHAP value (line 11), and the restricting features are the negative (line 12). If the reconstructed meta-feature value is greater than the actual (input) value (line 13), then the contributing features are the features with a negative SHAP value, and the restricting features are the positive. This algorithm returns two lists, *Contributing* and *Restricting*, that contain the contributing and restricting -meta-features, along with their reconstruction errors, for each of the *TopMetafeatures*. Figure 4 shows a sample content of the resulting lists.

The next step is selecting the meta-features with high SHAP values of each of the features in the *TopMetafeatures* list; so from each row in *Contributing* and *Restricting*, we extract the highest values as shown in figure 5. The implementation of the explanation method can be found in the Notebook code [1].

Meta-feature	reconstruction loss	contribut	restrict
noise_signal_ratio	14.040792	1	0
equiv_nr_feat	10.090109	1	0
norm_entropy_min	-4.645205	0	1
norm_entropy_std	4.128490	1	0
rho_std	3.107853	1	0
nr_features	2.922161	1	0
prob_std	-2.743658	0	1
Landmarker_gnb	-2.624745	0	1
Landmarker_lda	-2.606783	0	1
skew_std	-2.491297	0	1
nr_instances	2.369843	1	0
dataset_ratio	-2.204865	0	1

Figure 4: Resulted list (sample) of the contributing and the restricting meta-features.

features that restrict the good recommendation using Shapley values with those obtained using traditional meta-features and found that Shapley values provided a more comprehensive and accurate assessment of meta-feature importance.

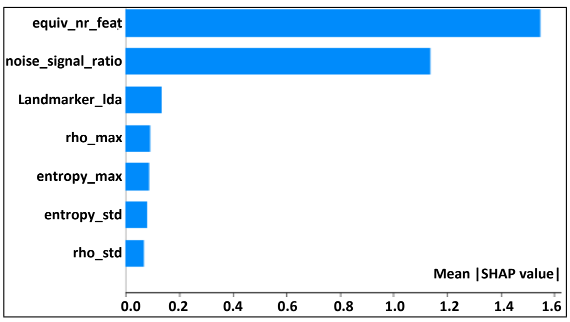

Figure 5: Importance of the top meta-features.

5 EVALUATION

To demonstrate the effectiveness of this approach, we applied Shapley values to a variety of meta-learning tasks. We examined the affect of getting rid the meta-

[1] Meta-Features_SHAP.ipynb

To validate and assess the competitiveness provided by the proposed approach for selecting the important and more informative meta-features, we perform a comparative study to the state-of-the-art and traditional ones (Garouani, 2022) with an oversampling approach using the 20-benchmark datasets using the KNN meta-model developed in (Garouani et al., 2022d).

The table 1 shows the results of the K-Nearest Neighbors (KNN) meta-model for recommending optimal pipelines for test data. The meta-model uses important meta-features and traditional one. The table shows the accuracy of the recommended ML algorithms on the benchmarked datasets, as well as the gain or loss obtained with the important meta-features compared to the traditional ones.

The results in the table are color-coded, with green indicating an improvement in accuracy when using important meta-features and red indicating a decrease in accuracy. Overall, the KNN meta-model show an improvement in accuracy when it is based on the most important dataset characteristics rather than on the whole set of all traditional ones. This means that the high-level meta-features obtained by the proposed approach provide more relevant information than those obtained by the state-of-the-art characteristics.

Table 1: Results of the KNN meta-model for recommending optimal pipelines for test data. The triangles (▲,▼) denote the gain/ loss % obtained with KNN with the selected important meta-features compared to the traditional ones used in (Garouani, 2022).

Dataset	Accuracy of the meta-model using	
	important MF	traditional MF
APSFailure	**0.9915** (0.05) ▲	0.9910
Higgs	**0.7319** (1.89) ▲	0.7130
CustSat	**0.8605** (0.46) ▲	0.8559
car	**0.9842** (0.88) ▲	0.9754
kr-vs-kp	0.9736 (2.04) ▼	**0.9976**
airlines	0.7274 (2.92) ▲	0.6982
vehicle	0.8817 (0.63) ▼	**0.8880**
MiniBooNE	0.9432 (2.13) ▼	**0.9645**
jannis	**0.6890** (1.71) ▲	0.6719
nomao	**0.9971** (2.63) ▲	0.9708
Credi-g	0.7747 (1.74) ▼	**0.7921**
Kc1	**0.9274** (4.81) ▲	0.8793
Cnae-9	**0.9702** (0.31) ▲	0.9671
albert	**0.8837** (0.78) ▲	0.8759
Numerai28.6	**0.5796** (5.82) ▲	0.5207
segment	**0.9867** (1.32) ▲	0.9735
Covertype	**0.8741** (3.97) ▲	0.8344
KDDCup	**0.9879** (1.39) ▲	0.9740
shuttle	0.9680 (0.32) ▼	**0.9649**
Gas_Sens-uci	**0.9917** (1.78) ▲	0.9739

6 CONCLUSION AND PROSPECTIVE

In this research paper, we proposed the use of Shapley values, a mathematical concept that are usually used to determine the importance of each feature in a cooperative game. We demonstrate that these can also be used as a method for understanding the importance of meta-features in meta-learning. For this purpose, through a series of experiments, it is found that Shapley values can effectively identify the most important meta-features and provide a more comprehensive understanding of their contribution to the overall performance of a meta-learning algorithm. This study also highlights that the relative importance of meta-features may vary depending on the task or dataset being used, and that certain meta-features may be more important than others. Furthermore, we also observe that the relative importance of meta-features may vary depending on the specific task or dataset being used. The findings of this study provides valuable insights into the use of Shapley values as a method for understanding the importance of meta-features in meta-learning and it has the potential to inform on the development of more effective meta-learning algorithms in the future. In future work, we plan to (1) expand the range of meta-learning algorithms and tasks that the Shapley values method is applied to, in order to gain a more comprehensive understanding of its usefulness in different contexts; (2) explore the use of Shapley values in combination with other features importance methods, such as features permutation or features elimination, to gain a more robust understanding of the importance of meta-features; (3) investigate the relationship between the Shapley values of meta-features and the performance of meta-learning algorithms under different conditions, such as varying amounts of training data or different types of noise in the data.

REFERENCES

Alcobaça, E., Siqueira, F., Rivolli, A., Garcia, L. P. F., Oliva, J. T., and de Carvalho, A. C. P. L. F. (2020). MFE: Towards reproducible meta-feature extraction. *Journal of Machine Learning Research*, 21(111):1–5.

Bennequin, E. (2019). Meta-learning algorithms for few-shot computer vision.

Garouani, M. (2022). Towards efficient and explainable automated machine learning pipelines design.

Garouani, M., Ahmad, A., Bouneffa, M., and Hamlich, M. (2022a). AMLBID: An auto-explained automated machine learning tool for big industrial data. *SoftwareX*, 17:100919.

Garouani, M., Ahmad, A., Bouneffa, M., and Hamlich, M. (2022b). Scalable meta-bayesian based hyperparameters optimization for machine learning. In *Communications in Computer and Information Science*, pages 173–186. Springer International Publishing.

Garouani, M., Ahmad, A., Bouneffa, M., and Hamlich, M. (2023a). Autoencoder-knn meta-model based data characterization approach for an automated selection of ai algorithms. *Journal of Big Data*, 10(14).

Garouani, M., Ahmad, A., Bouneffa, M., and Hamlich, M. (2023b). Autoencoder-knn meta-model based data characterization approach for an automated selection of ai algorithms. *Journal of Big Data*.

Garouani, M., Ahmad, A., Bouneffa, M., Hamlich, M., Bourguin, G., and Lewandowski, A. (2022c). Towards big industrial data mining through explainable automated machine learning. *The International Journal of Advanced Manufacturing Technology*, 120:1169–1188.

Garouani, M., Ahmad, A., Bouneffa, M., Hamlich, M., Bourguin, G., and Lewandowski, A. (2022d). Using meta-learning for automated algorithms selection and configuration: an experimental framework for industrial big data. *Journal of Big Data*, 9:1169–1188.

Garouani., M., Ahmad., A., Bouneffa., M., Lewandowski., A., Bourguin., G., and Hamlich., M. (2021). Towards the automation of industrial data science: A meta-learning based approach. In *Proceedings of the 23rd International Conference on Enterprise Information Systems - Volume 1: ICEIS*, pages 709–716.

Garouani, M. and Zaysa, K. (2022). Leveraging the automated machine learning for arabic opinion mining: A preliminary study on AutoML tools and comparison to human performance. In *Digital Technologies and Applications*, pages 163–171. Springer International Publishing.

Hart, S. (1989). Shapley value. In *Game Theory*, pages 210–216. Palgrave Macmillan UK.

Kalousis, A. and Hilario, M. (2001). Feature Selection for Meta-learning. In Cheung, D., Williams, G. J., and Li, Q., editors, *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, pages 222–233. Springer.

Lundberg, S. and Lee, S.-I. (2017a). A unified approach to interpreting model predictions.

Lundberg, S. M. and Lee, S.-I. (2017b). A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA. Curran Associates Inc.

Nural, M. V., Peng, H., and Miller, J. A. (2017). Using meta-learning for model type selection in predictive big data analytics. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 2027–2036.

Olsen, L. H. B., Glad, I. K., Jullum, M., and Aas, K. (2022). Using shapley values and variational autoencoders to explain predictive models with dependent mixed features. *Journal of Machine Learning Research*, 23(213):1–51.

Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., and Müller, K.-R., editors (2019). *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer International Publishing.

Shao, X., Wang, H., Zhu, X., and Xiong, F. (2022). Find:explainable framework for meta-learning.

Shao, X., Wang, H., Zhu, X., Xiong, F., Mu, T., and Zhang, Y. (2023). EFFECT: Explainable framework for meta-learning in automatic classification algorithm selection. *Information Sciences*, 622:211–234.

Woźnica, K. and Biecek, P. (2020). Towards explainable meta-learning.