Machine Learning (2024) 113:4245–4284
https://doi.org/10.1007/s10994-021-05981-0

Can metafeatures help improve explanations of prediction models when using behavioral and textual data?

Yanou Ramon[1] · **David Martens**[1] · **Theodoros Evgeniou**[2] · **Stiene Praet**[1]

Received: 1 March 2020 / Revised: 30 March 2021 / Accepted: 12 April 2021 /
Published online: 8 June 2021

Abstract
Machine learning models built on behavioral and textual data can result in highly accurate prediction models, but are often very difficult to interpret. Linear models require investigating thousands of coefficients, while the opaqueness of nonlinear models makes things worse. Rule-extraction techniques have been proposed to combine the desired predictive accuracy of complex "black-box" models with global explainability. However, rule-extraction in the context of high-dimensional, sparse data, where many features are relevant to the predictions, can be challenging, as replacing the black-box model by many rules leaves the user again with an incomprehensible explanation. To address this problem, we develop and test a rule-extraction methodology based on higher-level, less-sparse "metafeatures". We empirically validate the quality of the explanation rules in terms of fidelity, stability, and accuracy over a collection of data sets, and benchmark their performance against rules extracted using the fine-grained behavioral and textual features. A key finding of our analysis is that metafeatures-based explanations are better at mimicking the behavior of the black-box prediction model, as measured by the fidelity of explanations.

Keywords Explainable artificial intelligence · Interpretable machine learning · Metafeatures · Comprehensibility · Global explanations · Rule-extraction · Classification · Big behavioral data · Textual data

1 Introduction

Technological advances have allowed storage and analysis of large amounts of data and have given industry and government the opportunity to gain insights from thousands of digital records collected about individuals each day (Matz and Netzer, 2017). These "big behavioral data"—characterized by large volume, variety, velocity and veracity,

Editors: Tim Verdonck, Bart Baesens, María Óskarsdóttir and Seppe vanden Broucke.

✉ Yanou Ramon
yanou.ramon@uantwerp.be

[1] Department of Engineering, University of Antwerp, Antwerp, Belgium

[2] Decision Sciences and Technology Management, INSEAD, Fontainebleau, France

and defined as data that capture human behavior through the actions and interactions of people (Shmueli, 2010)—have led to predictive modeling applications in areas such as fraud detection (Vanhoeyveld et al., 2019), financial credit scoring (Martens et al., 2007; De Cnudde et al., 2018; Tobback and Martens, 2019), marketing (Verbeke, 2011; Matz and Netzer, 2017; Chen et al., 2017) and political science (Praet et al., 2018). Sources of behavioral data include, but are not limited to, transaction records, search query data, web browsing histories, social media profiles, online reviews, and smartphone sensor data (e.g., GPS location data). Textual data are also increasingly available and used. Example text-based applications are automatic identification of spam emails (Attenberg et al., 2009), objectionable web content detection (Martens and Provost, 2014) and legal document classification (Chhatwal et al., 2019), just to name a few examples.

Behavioral and textual data are very high-dimensional compared to traditional data, which are primarily structured in a numeric format and are relatively low-dimensional (Moeyersoms et al., 2016; Matz and Netzer, 2017; De Cnudde et al., 2020). Consider the following example to illustrate these characteristics: the prediction of personality traits of users based on the Facebook pages they have "liked" (Kosinski et al., 2013; Matz and Netzer, 2017). A user is represented by a binary feature for each unique Facebook page that exists, with a 1 if that page was liked by the user and 0 otherwise, which results in an enormous feature space. However, each user only liked a relatively small number of pages, which results in an extremely sparse data matrix (almost all elements are zero). In the literature, because of their specific nature, behavioral and textual data are often referred to as "fine-grained" (Martens and Provost, 2014; Martens et al., 2016; De Cnudde et al., 2020). For this reason, in this article we mathematically represent behavioral and textual data as $X_{FG} \subset \mathbb{R}^{n \times m}$, where FG stands for "fine-grained", and n and m refer to the number of instances and features respectively. These features can be binary (e.g., someone "liked" a Facebook page or not) or numerical (e.g., tf-idf vectorization for text documents).

Learning from behavioral and textual data can result in highly accurate prediction models (Junqué de Fortuny et al., 2013; De Cnudde et al., 2020). A drawback of prediction models trained on these types of data, however, is that they can become very complex. The complexity arises from either the learning technique (e.g., deep learning) or the data, or both. It is essentially impossible to interpret classifications of nonlinear techniques such as Random Forests or deep neural networks without using interpretation techniques like rule-extraction—on which the solution proposed in this article is based—or feature importance methods (e.g., LIME (Ribeiro et al., 2016) or TreeSHAP (Lundberg and Lee, 2017)). For linear models or decision trees, the most common approach to understand the model is to examine the estimated coefficients or to inspect the paths from root to leaf nodes. In the context of behavioral and textual data, however, even linear models are not straightforward to interpret because of the large number (thousands to millions) of features each with their corresponding weight (Martens and Provost, 2014; Moeyersoms et al., 2016). Moreover, one may question the comprehensibility of decision trees with thousands of leaf nodes. Alternatively, for linear models, we could inspect only the features with the highest estimated weights. But for sparse data, this means that only a small fraction of the classified instances are actually explained by these features, because of the low coverage of the top-weighted features (Martens and Provost, 2014; Moeyersoms et al., 2016). Kosinski et al. (2013), for example, explain models that predict personal traits using over 50,000 Facebook "likes" by listing the pages that are most related to extreme frequencies of the target classes. For example, the best predictors for high intelligence include Facebook pages "*The Colbert Report*", "*Science*" and "*Curly Fries*" (Kosinski et al., 2013). Because of the

extreme sparsity of the data (users liked on average 170 out of 55,814 possible pages), these pages are only relevant to a small fraction of users predicted as "highly intelligent", which questions the practicality of this approach for better understanding (global) model behavior.

It is important to note that the high-dimensional, sparse nature of behavioral and textual data alone does not necessarily lead to complex prediction models. If many behavioral or textual features are irrelevant for the prediction task, applying dimensionality reduction or feature input selection prior to modeling, or using strong model regularization can result in models having high predictive performance, while still being interpretable. However, previous research shows that all of these techniques result in worse predictive performance compared to models that exploit the full set of behavioral or textual features for making predictions (Joachims, 1998; Junqué de Fortuny et al., 2013; Clark and Provost, 2015; Martens et al., 2016; De Cnudde et al., 2020). By means of a learning curve analysis on a benchmark of 41 behavioral data sets, De Cnudde et al. (2019) demonstrate that, when mining text or behavior, many features contribute to the predictions. Similar results have been found by Clark and Provost (2015) and Junqué de Fortuny et al. (2013) for behavioral data, and by Joachims (1998) for the analysis of textual data. In other words, the dimensionality and sparsity of the data combined with many relevant features drive the "black-box" nature of any model trained on behavioral and textual data. We represent a classification model trained on behavioral or textual data X_{FG} as C_{BB}, where BB stands for "black-box".

Explainability has emerged recently as a key business and regulatory challenge for machine learning adoption. The relevance of global interpretability of classification models is well-argued in the literature (Andrews and Diederich, 1995; Diederich, 2008; Martens et al., 2007; Junqué de Fortuny and Martens, 2015).[1] In the process of extracting knowledge from data, the predictive performance of classification models alone is not sufficient as human users need to understand the models to trust, accept and improve them (Van Assche and Blockeel, 2007). Both the United States and the European Union are currently pushing towards a regulatory framework for trustworthy Articial Intelligence, and global organizations such as the OECD and the G20 aim for a human-centric approach (European Comission, 2020). In high-stake application domains, explanations are often legally required. In the credit scoring domain, for example, legislation such as the Equal Credit Opportunity Act in US Federal Law (US Federal Trade Commission, 2003) prohibits creditors from discrimination and requires reasons for rejected loan applications. Also in lower-stakes applications, such as (psychologically) targeted advertising (Matz and Netzer, 2017; Moeyersoms et al., 2016) or churn prediction (Verbeke et al., 2012), explanations are managerially relevant. Global interpretability allows to verify the knowledge that is encoded in the underlying models (Andrews and Diederich, 1995; Huysmans et al., 2006). Models trained on big data may learn incorrect trends, overfit the data or perpetuate social biases (Chen et al., 2017). Furthermore, explanations might give users more control of their virtual footprint. Matz et al. (2020) argue that insight into what data is being collected and the inferences that can be drawn from it, allow users to make more informed privacy

[1] Explanations for model predictions vary in scope: a method either generates global or instance-level explanations (Martens and Provost, 2014; Ramon et al., 2020). We focus on global explanations that give insight in the model's behavior over all possible feature values and for all instances. Instance-level explanations, on the other hand, explain a single model decision. For example, when mining behavior or text, an *Evidence Counterfactual* is an instance-level explanation that shows a minimal set of features such that, when changing their feature values to zero, the predicted class changes (Martens and Provost, 2014; Ramon et al. 2020).

decisions (Matz et al., 2020). Moreover, global model explainability can help to induce new insights or generate hypotheses (Andrews and Diederich, 1995; Shmueli, 2010).

Rule-extraction has been proposed in the literature to generate global explanations by distilling a comprehensible set of rules (hereafter referred to as "explanation rules") from complex classifiers C_{BB}. The complexity of the set of rules is largely restricted so that the resulting explanation is understandable to humans. Rule-extraction in the context of big behavioral and textual data can be challenging, and, to our knowledge, has thus far received scant attention. Because of the data characteristics, rule-extraction might fail in their primary task (providing insight on the black-box model) as the complex model needs to be replaced by a set of hundreds or even thousands of explanation rules (Huysmans et al., 2006; Sushil et al., 2018).

This article addresses the challenge of using rule-extraction to globally explain classification models on behavioral and textual data. Instead of focusing on rule-extraction techniques themselves[2], this article leverages an alternative higher-level feature representation $X_{MF} \subset \mathbb{R}^{n \times k}$, where MF stands for "metafeatures" and k represents the number of metafeatures. Metafeatures are expected to improve the fidelity (approximation of the black-box classification model), explanation stability (same explanations for slightly different training sessions—a concept we introduce, which we will be calling just stability) and accuracy (correct predictions of the original instances) of the extracted explanation rules. For simplicity, we only focus on classification problems. Our main claim is that metafeatures are more appropriate, in specific ways we discuss, for extracting explanation rules than the original behavioral and textual features that are used to train the model.

This article's main contributions are threefold: (1) we propose a novel methodology for rule-extraction by exploring how higher-level feature representations (metafeatures) can be used to increase global understanding of classification models trained on fine-grained behavioral or textual data; (2) we define a set of quantitative criteria to assess the quality of explanation rules in terms of fidelity, stability, and accuracy; and empirically study the trade-offs between these; and lastly, (3) we perform an in-depth empirical evaluation of the quality of explanations with metafeatures using nine data sets, and benchmark their performance against the explanation rules extracted with the behavioral or textual features. We aim to answer the following empirical questions:

- How do explanation rules extracted with metafeatures compare against rules extracted with the fine-grained behavioral and textual features across different evaluation criteria (fidelity, stability, accuracy)?
- How does the fidelity[3] of explanation rules vary for different complexity settings?
- To what extent do the fidelity and stability of explanation rules extracted with metafeatures depend on a key parameter of our metafeatures-based rule-extraction methodology, that is the parameter k that represents the number of metafeatures?

[2] In this article, we use the decision tree algorithm CART as the rule-extraction algorithm. We leave the comparison of different rule-extraction techniques (Ripper (Cohen,1995), C4.5 (Quinlan, 1993), and so on) as well as more advanced variants like active learning-based rule-extraction (Junqué de Fortuny and Martens, 2015; Craven and Shavlik, 1999) to future research. Our main focus is on empirically assessing the value of using metafeatures for extracting explanation rules.

[3] As the main goal of rule-extraction is to mimic the behavior of black-box models with a set of rules, we focus on fidelity instead of, say, accuracy, as discussed in Sect. 4.4. Our methodology and analysis can be adapted to also study the accuracy of explanations.

2 Related work

2.1 Rule-extraction

In the Explainable Artificial Intelligence (XAI) literature, rule-extraction falls within the class of post-hoc explanation methods that use "surrogate models" to gain understanding of the learned relationships captured by the trained model (Martens et al., 2007; Murdoch et al., 2019). The idea of surrogate modeling is to train a comprehensible surrogate model (the *white-box* C_{WB}) to mimic the predictions of a more complex, underlying *black-box* model[4] C_{BB} (Diederich, 2008). We define a black-box model as a complex model from which it is not straightforward for a human interpreter to understand how predictions are made. In this article, we consider *any* classification model (linear, rule-based or nonlinear) utilising a large number of features as a black-box (because of the specific nature of the data described in the Introduction, a large number of features are typically used in the final models); which is different from previous research that only considers highly-nonlinear models as black-boxes (Andrews and Diederich, 1995; Martens et al., 2007; Diederich, 2008; Martens et al., 2009).

In the machine learning literature, small decision trees and rule-based models with few rules have been argued to yield the most comprehensible classification models (Van Assche and Blockeel, 2007; Freitas, 2013), making them good candidates to use as white-box models C_{WB} to extract a set of explanation rules (known as "rule-extraction").[5] It is important to note that the complexity of the rules needs to be restricted so that the resulting explanation is comprehensible to humans (Martens et al., 2007, 2009).

Rule-extraction can be used for two purposes. First and foremost, one may be interested in knowing the rationale behind decisions made by a classification model C_{BB} and verify whether the results make sense in practice. The goal is to extract comprehensible rules that closely mimic the black-box, that is measured by what is called "fidelity". Alternatively, the goal can be to improve the "accuracy", namely, the generalization performance of a white-box model (e.g., a small decision tree or a concise set of rules) by approximating the black-box (Martens et al., 2009; Huysmans et al., 2006). In this article, we discuss most results in terms of fidelity instead of accuracy as our focus is on developing global explanations that "best mimic the black-box"—but all our analyses can also be done using accuracy as the main metric.

Rule-extraction methods use the mapping of the data to the predicted labels, i.e., the input-output mapping defined by the model C_{BB} (Andrews and Diederich, 1995; Martens et al., 2007; Huysmans et al., 2006). The idea behind this approach is that the similarity between the black-box and white-box model (the fidelity) can be substantially improved by presenting the labels predicted by the black-box model $\hat{\mathbf{y}} = \{\hat{y}_i\}_{i=1}^n$ to the white-box model, instead of the ground-truth labels $\mathbf{y} = \{y_i\}_{i=1}^n$ (Martens et al., 2009; Junqué de Fortuny and Martens, 2015).

[4] We will interchangeably refer to this model as the black-box model or the underlying model that we want to interpret globally.

[5] Note that, in the literature, also linear models with a small number of features have been proposed as surrogate models to approximate a prediction model's behavior (Ribeiro et al., 2016). In this article, we focus on rule-based models as surrogates.

2.2 Challenges of rule-extraction for high-dimensional data

The vast majority of the rule-extraction literature has focused on improving the fidelity and scalability of rule-learning algorithms. However, despite some very impressive and promising work (Andrews and Diederich, 1995; Martens et al., 2007, 2009; Diederich, 2008; Junqué de Fortuny and Martens, 2015), the rule-extraction techniques are mostly validated on low-dimensional, dense data, such as the widely used set of benchmark data from the UCI Machine Learning repository (Bache and Lichman, 2020). These data have feature dimensions going up to 50 features. We identify at least three challenges in regard to rule-extraction to explain classifiers on fine-grained behavioral and textual data:

1. *Complexity of the explanation rules.* In the context of high-dimensional data with many relevant features, rule-extraction might fail to provide insight on the black-box model as the black-box model needs to be replaced by a large set of rules (Martens et al., 2007, 2009; Huysmans et al., 2006). Sushil et al. (2018) applied rule-extraction on (real-world) textual data and show that rule learners can closely approximate the underlying model, but at the cost of being very complex (hundreds of rules). A related challenge that stems from rule-based learners not being very adept at handling high dimensionality, is their high variance profile that can result in overfitting (Kotsiantis et al., 2006; De Cnudde et al., 2020).
2. *Computational complexity.* It is not straightforward for every existing rule-learning algorithm to be used for high-dimensional data, because the learning task can become computationally too demanding (Andrews and Diederich, 1995; Sushil et al., 2018). Some algorithms, such as Ripper (Cohen, 1995), are not able to computationally deal with problem instances having large-scale feature spaces (Sushil et al., 2018).
3. *Fine-grained feature comprehensibility.* Diederich (2008) questions the usefulness of rules extracted from models trained on behavioral or textual data. For example, when rules are learned from a model initially trained on a "bag–of–words" representation of text documents, the antecedents in a rule include individual words taken out of their context. This can reduce the semantic comprehensibility of the explanations. Likewise, for digital trace data, we can question the comprehensibility of a single action (e.g., a single credit card transaction, a single Facebook "like") taken out of its context, that is, the collection of all behaviors of an individual.

Because of the above challenges, it is questionable *whether fine-grained behavioral and textual features are the best representation for extracting global explanation rules to achieve the best explanation quality in terms of fidelity, stability, and accuracy*. This motivates our approach to use a metafeatures representation instead. It is not clear a priori whether such a representation will improve the quality of explanations of models on behavioral and textual data, making this a key empirical question that we study in this article.

3 Metafeatures

3.1 Motivation

As previously introduced, behavioral and textual data suffer from high dimensionality and sparsity. For this reason, the features individually may exhibit little discriminatory

power to explain the black-box model. Because of the low coverage that characterizes such sparse features, a single feature is not expected to "explain" much of the classifications of the underlying model. The feature will only be active (nonzero) for a small fraction of all data instances, and therefore, the coverage of an explanation rule with a single behavioral or textual feature is likely to be low (Sommr, 1995; Martens and Provost, 2014; Chen et al., 2016; Sushil et al., 2018).[6]

We address the data sparsity by mapping the fine-grained, sparse features $X_{FG} \subset \mathbb{R}^{n \times m}$ onto a higher-level, less-sparse feature representation $X_{MF} \subset \mathbb{R}^{n \times k}$ (to which we refer as "metafeatures"), where m and k respectively represent the dimensionality of the original features and metafeatures. Existing research has experimented with the idea of using higher-level features other than the actual features used by the model to extract explanations (Ribeiro et al., 2016; Chen et al., 2016; Kim et al., 2018; Lee et al., 2019). In the field of image recognition, for example, the input pixels are not straightforward to interpret, hence researchers have proposed to use a patch of similar pixels (super-pixels) for generating explanations of image classifications (Ribeiro et al., 2016; Wei et al., 2018). Another example stems from the field of natural language processing, where Chen et al. (2016) cope with data sparsity by clustering similar features by their frequency in large data sets. All of these approaches have, however, not been used before in the context of rule-extraction for models on big behavioral and textual data.

3.2 Desired properties

We propose the following set of properties for engineering metafeatures:

1. *Low dimensionality.* We want the dimensionality k of the metafeatures to be smaller than the dimensionality m of the original feature space: $k << m$. A lower feature dimension may lead to more stable explanation rules (Alvarez-Melis and Jaakkola, 2018). Moreover, the computational burden for extracting rules with metafeatures is likely to be much lower compared to rule-extraction with high-dimensional data (Andrews and Diederich, 1995; Sushil et al., 2018).
2. *High density.* This property relates to the coverage of a metafeature, which we want to be higher compared to the coverage of fine-grained features (Chen et al., 2016). In other words, there should be more instances for which a metafeature is active (nonzero) compared to the fine-grained features. The higher density (lower sparsity) of the metafeatures is expected to increase the fidelity and accuracy of explanation rules resulting from the higher coverage of rules predicting the non-default target class (often the "class of interest").
3. *Faithful to the original feature representation.* This property is in line with prior research suggesting that the representation of the original data instances in terms of metafeatures should preserve relevant information to discriminate between the predicted labels $\hat{\mathbf{y}}$ (Alvarez-Melis and Jaakkola, 2018). The metafeatures should preserve the predictive information of the original features as the black-box model is trained on the latter. It is important that the extracted rules using metafeatures can reach a high level of discrimi-

[6] The coverage of a feature is defined as the number of data instances that have a nonzero value for this feature, whereas the coverage of a rule is defined as the number of instances that are classified by this rule. For sparse data, both feature and rule coverage tend to be low.

natory power in regard to the true predictions being made, because this will result in a better approximation of the underlying model as measured by fidelity. In the experiments, we use the test fidelity of the explanations as a proxy to measure the faithfulness of the metafeatures X_{MF} to the original features X_{FG}. In addition, we use the Gini index to measure the predictive information captured by each metafeature.[7]

4. *Semantic comprehensibility.* Metafeatures should have a human-comprehensible interpretation (Alvarez-Melis and Jaakkola, 2018). For example, Facebook "likes" can be grouped into semantically meaningful categories (e.g., "*Entrepreneurship*") and GPS location data can be categorized into venue types (e.g., "*Concert halls*"). This property is subjective in nature and depends on the application domain and the expectations of users who interact with the model (explanations) (Wood, 1986; Campbell, 1988; Huysmans et al., 2006; Huysmans et al., 2011). We do not explicitly measure the comprehensibility of explanations with (meta)features, as this would require experimentation with people, an important research direction to explore if indeed metafeatures improve the quality of explanations in the other dimensions we study here (fidelity, stability, accuracy). In this article, we make the assumption that the resulting metafeatures are semantically meaningful. In Sect. 4.5, we demonstrate how metafeatures generated with a data-driven method (e.g., Non-negative Matrix Factorization) can be interpreted, based on common practices described in the literature (Wang and Blei, 2011; O'Callaghan et al., 2015; Contreras-Pina and Sebastián, 2016; Kulkarni et al., 2018; De Cnudde et al., 2019). Note that the metafeatures that are manually crafted (the "domain-based" metafeatures described in Sect. 4.2) are, by design, comprehensible to humans.

4 Metafeatures-based explanation rules

We introduce and validate a methodology to extract explanation rules from a complex model C_{BB} trained on behavioral and textual data X_{FG}. The steps of the proposed methodology are summarized in Fig. 1 and discussed below.

4.1 Model building and predicting labels

From the behavioral and textual data X_{FG} (having n instances and m features) we train and test the black-box model C_{BB}. The model is trained on a subset of $\alpha \times \beta \times n$ instances (the training set $X_{FG,train}$ with corresponding labels $\mathbf{y}_{\boldsymbol{train}}$) and hyperparameters are optimized using a holdout set of $\alpha \times (1-\beta) \times n$ instances (the validation set $X_{FG,val}$ with labels $\mathbf{y}_{\boldsymbol{val}}$). Finally, the generalization performance of the black-box model is evaluated on an unseen part of the data (the test set $X_{FG,test}$ with labels $\mathbf{y}_{\boldsymbol{test}}$) that contains $(1-\alpha) \times n$ instances.[8] The trained black-box model is used to make predictions $\hat{\mathbf{y}}$ for all instances in the data set (training, validation and test data), which will thereafter be used to train, fine tune and test our white-box model C_{WB} (the explanation rules).

[7] The Gini index is the splitting criterion of the CART decision tree algorithm that we use in this article to extract explanation rules.

[8] For classification models that do not require hyperparameter tuning, β equals 1. In the experiments in Sect. 6, we set α and β to 0.8. Moreover, we use five-fold cross-validation (CV) to evaluate the generalization performance of the black-box model C_{BB} and to measure fidelity and accuracy of the explanation rules C_{WB} on the test data.

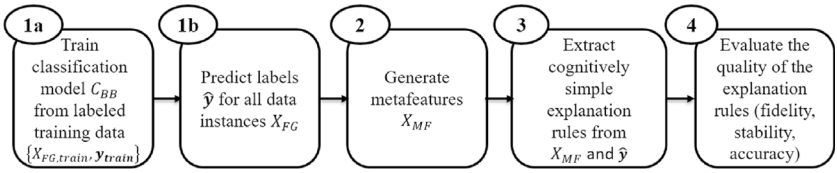

Fig. 1 Proposed rule-extraction methodology using metafeatures

4.2 Generating metafeatures X_{MF}

We need to specify a feature transformation process to group behavioral and textual features X_{FG} with similar properties together in metafeatures X_{MF} (Chen et al., 2016). There are various approaches for generating metafeatures from the original features, either by manually crafting them using domain knowledge (Murdoch et al., 2019) or by automatically obtaining them by means of data-driven feature engineering techniques, such as (un) supervised dimensionality reduction. In the following, we use DomainMF and DDMF as abbreviations for domain-based metafeatures and data-driven metafeatures respectively.

4.2.1 Domain-based metafeatures

One way of generating metafeatures from the original features is to group features together in domain-based categories that are manually crafted by experts (Murdoch et al., 2019; Alvarez-Melis and Jaakkola, 2018). For example, for the Facebook "like" data, individual Facebook pages can be grouped together in predetermined categories, for example, pages related to "*Machine Learning*". This human-selected set of metafeatures can then be used to extract simple rules to explain model predictions, which represent the relative importance of these domain-based metafeatures in the prediction model. In this article, we mathematically denote the domain-based metafeatures as $X_{DomainMF} \subset \mathbb{R}^{n \times k}$.

4.2.2 Data-driven metafeatures

Alternatively, metafeatures can be generated via a data-driven approach, such as matrix factorization-based dimensionality reduction.[9] The idea is to increase density by representing the data in a lower dimensional space without too much loss of information. The original data matrix X_{FG} with n unique instances and m unique features is split into two matrices $L_{n \times k}$ and $R_{k \times m}$ such that: $X_{FG} \approx L\ R$. The k columns of L represent the metafeatures, and each instance will have a representation in the new k-dimensional space. The matrix R,

[9] Note that there exist many other techniques that can be explored to generate the metafeatures representation, such as supervised dimensionality reduction (e.g., see De Cnudde et al. (2020)), embedding techniques (e.g., word2vec (Mikolov et al., 2013) or Fasttext (Joulin et al., 2016)) or language representation models (e.g., BERT (Devlin et al., 2019)). Although different methods exist, we do not aim to do a comparison among them in this work. Our focus is to first answer whether metafeatures can help for the problem we study. A positive answer would indicate that it is promising to study other metafeatures methods for this problem in the future. Moreover, if other approaches generate better results, that would only further support our findings about the value of metafeatures for explainability for behavioral and textual data, rendering our results more conservative.

represents the relationship between the new metafeatures and the original features (Clark and Provost, 2015).

Metafeatures group together related features. The quality of the metafeatures depends on the number of extracted metafeatures k: a value of k that is set too high results in many highly-similar categories, whereas a low value of k tends to generate overly-broad metafeatures. The intended goal of generating metafeatures in this article is to use them for rule-extraction, and consequently, we optimize the number of k such that the out-of-sample fidelity of the rules is maximal (we use a validation set to fine tune the value of k). We consider values of k from 10 up to 1000 (Clark and Provost, 2015). Note that we should not be concerned with generating too many metafeatures because we only need to interpret the ones that are part of the final explanation rules (this is demonstrated in Sect. 4.5).

For generating metafeatures based on matrix-factorization-based dimensionality reduction, we first approximate the original training data X_{FG} by two matrices L and R for a given number of metafeatures k (step 1 in Fig. 13 in "Appendix 1"). Matrix R maps each metafeature to the original fine-grained features. We ensure mutual exclusivity by transforming matrix R into a binary matrix R_{binary}, where 1 represents the maximum element for each column (fine-grained feature) of R and all other elements are 0 (step 2 in Fig. 13). In other words, each feature only belongs to one metafeature. Next, we map the original matrix X_{FG} to X' by multiplying X_{FG} with the transposed binary matrix R^T_{binary} (step 3 in Fig. 13). Finally, matrix X' is normalized over the number of active (fine-grained) features per instance (e.g., total number of behaviors or words) to become matrix X_{DDMF-k} that represents the metafeatures per instance (step 4 in Fig. 13). We found that the normalized matrix X_{DDMF-k} produced better results (as measured by test fidelity of the explanations) than utilizing the original matrix X' or even a binary matrix derived from X'. We apply the binarization of matrix R to make the resulting explanation rules more interpretable and semantically meaningful. For example, for the *Facebook* data, the explanation rules with the metafeatures can be interpreted in terms of the percentage of "liked" pages of a category (see Fig. 4). If we would immediately use the matrix L to represent instances in the metafeatures space, and ignore the binarization and normalization steps, the explanation rules would contain logical statements that are not immediately comprehensible. For example, it would be difficult to interpret a rule that says something like, "if the value of this metafeature is higher than 0.3, then the model predicts the person as Female", when the metafeature is not expressed in terms of an actual unit of measurement.[10]

In this article, we experimented with two well-established dimensionality reduction methods based on matrix factorization: Non-negative Matrix Factorization (NMF) and Singular Value Decomposition (SVD). NMF is applied in multiple domains to decompose a non-negative matrix into two non-negative matrices (Lee and Seung, 2001). In most real-life applications, negative components or subtractive combinations in the representation are physically meaningless. Incorporating the non-negativity constraint thus facilitates the interpretation of the extracted metafeatures in terms of the original data (Wang and Zhang, 2012; Kulkarni et al., 2018; Clark and Provost, 2015). SVD is a popular technique for matrix factorization across a wide variety of domains such as text classification (Husbands et al., 2001) and image recognition (Turk and Pentland, 1991). SVD is computed by

[10] We also experimented with using the continuous data matrix L, without doing binarization or normalization, to extract explanation rules. The difference in fidelity of the rules using this continuous metafeatures representation compared to using metafeatures with binarization and normalization, was marginal—given that the fidelity of explanations with the binarized metafeatures was already higher than explanations with the original data (see Sect. 6).

optimizing a convex objective function and the solution is equivalent to the eigenvectors of the data matrix. We implemented these dimensionality reduction techniques using Python's *Scikit-learn* package (Pedregosa et al., 2011).

An important assumption we make is that the resulting data-driven metafeatures are semantically meaningful. While the obtained metafeatures are not always guaranteed to be interpretable, especially NMF has been shown to provide interpretable results for fine-grained data applications (Contreras-Pina and Sebastián, 2016; Lee and Seung, 1999), as compared to other techniques like SVD. Usually, metafeatures are interpreted by looking at the top-weighted features (Wang and Blei, 2011; O'Callaghan et al., 2015; Contreras-Pina and Sebastián, 2016; Kulkarni et al., 2018; De Cnudde et al., 2019). It is important to note that only the metafeatures that are part of the final explanation rules need to be interpreted.[11]

4.3 Extracting explanation rules

Both rule and decision tree learning algorithms can be used for rule-extraction. Since trees can be converted to rules, we also use tree algorithms for rule-extraction (Martens et al., 2007; Huysmans et al., 2011; Martens, 2008). A full review of these techniques is beyond the scope of this article, but we will shortly describe CART[12], as this is the technique used in our experiments.

CART can be used for both classification and regression problems and it uses the Gini index as a splitting criterion, which measures the impurity of nodes. The best split is the one that reduces the impurity the most. We apply CART to the data where the target variable is changed to the black-box predicted class label $\hat{\mathbf{y}}$ instead of the ground-truth labels $\mathbf{y}$ (see Sect. 2.1).

The number of extracted explanation rules can be used as a proxy for human comprehensibility.[13] Restricting the complexity of the rule set is also motivated by research on how people make decisions: based on relatively simple rules to avoid excess cognitive effort (Gigerenzer and Goldstein, 2016; Hauser et al., 2009) due to cognitive limitations (Sweller, 1988). In the context of consumer decision-making for example, Hauser et al. (2009) argue that decision rules should incorporate "cognitive simplicity": Rule sets should consist of a limited number of rules, each with a small number of antecedents. Finally, it is important to note that the concept of comprehensibility in the context of explanation rules comprises many different aspects, such as the size of the explanation, but also the specific application context and subjective opinion and expectations of the end user, which makes it difficult to measure comprehensbility in a generic way (Huysmans et al., 2011; Campbell, 1988; Wood, 1986). In line with cognitive simplicity arguments (Hauser et al., 2009; Sweller, 1988), in the experiments in Sect. 6, we restrict the complexity of the explanations to at most 32 rules each consisting of at most five antecedents (this is equivalent to a tree depth of at most five).

[11] We will demonstrate in Sect. 4.5 how to interpret metafeatures that are part of the explanation by looking at the top-weighted features per metafeature.

[12] CART is readily available from the *Scikit-learn* library in Python.

[13] A general assumption in the literature is that linear models with few parameters or rule-based models with few rules are more comprehensible than linear models with many parameters or rule sets with many different rules (Freitas, 2013; Hauser et al., 2009; Huysmans et al., 2011).

4.4 Evaluating explanation rules

4.4.1 Fidelity

First and foremost, the explanation rules are evaluated on how well they approximate the classification behavior of the underlying model. Fidelity measures the ability of the rules to mimic the model's classification behavior from which they are extracted. Let $\{\mathbf{x}_i, y_i\}_{i=1}^n$ represent the labeled data instances and $\mathbf{y}^{\mathbf{WB}}$ and $\hat{\mathbf{y}}$ respectively the white-box and black-box predicted labels. Fidelity is expressed as the fraction of instances for which the label predicted by the explanation rules (the white-box predicted label) equals the black-box predicted label (Craven and Shavlik, 1999; Huysmans et al., 2006):

$$fidelity^{WB} = \frac{|\{\hat{y}_i = y_i^{WB} | \mathbf{x}_i\}_{i=1}^n|}{N} \tag{1}$$

While most of our analysis is based on fidelity, we can extend the fidelity to "*f-score fidelity*" (f-fidelity). The f-fidelity is defined as the harmonic mean between *precision* and *recall* of the white-box predictions (w.r.t. the predicted labels $\hat{\mathbf{y}}$ rather than the true labels $\mathbf{y}$). More precisely, the formula of f-fidelity is $\frac{2 \cdot precision \cdot recall}{precision + recall}$, where the *precision* of the classifier is the fraction of positively-predicted instances that is correctly classified and the *recall* refers to the fraction of positive instances that is correctly classified as a positive. Note that f-fidelity is less intuitive than fidelity, but we use it in the experiments as an additional metric to measure to what extent the explanation rules reflect the black-box model, that is especially interesting for imbalanced problems, i.e., prediction tasks where the distribution of the target variable is skewed (e.g., the *20news* data in the experiments[14]).

4.4.2 Stability

A second important factor is explanation stability—which we will call stability from here on. Users, businesses, or regulators may have a hard time accepting explanations that are unstable (meaning small changes in the data lead to large changes in explanations of the black-box model), even if the explanation has shown to have high fidelity and comprehensibility (Van Assche and Blockeel, 2007). Turney (1995) distinguishes two types of stability: syntactic and semantic stability. Semantic stability is often measured by estimating the probability that two models learned on different training sets, will give the same prediction to an instance. On the other hand, syntactic stability measures how similar two explanations are (e.g., the overlap of features in two different explanations), and is more specific to a particular explanation representation (Turney, 1995). We argue that syntactic stability is the most relevant type of stability in the context of explaining classification models. To the best of our knowledge, it remains an open question how to measure syntactic stability for different explanation representations, such as rules and trees. We propose a procedure based on the Jaccard coefficient to measure syntactic stability of explanation rules. More specifically, by measuring the overlap of features that are part of the explanations extracted

[14] For this data set and prediction task, predicting all news posts as positive (related to "atheism") using one explanation rule would already result in a fidelity of approximately 96%, whereas f-fidelity would give a better idea of the actual explanatory power of the explanation rules w.r.t. the classification model.

from slightly different subsets of training data.[15] To compute the stability of explanation rules extracted using different data representations $\forall\ X \in \{X_{FG}, X_{DDMF-k}, X_{DomainMF}\}$ and the black-box predicted labels $\hat{\mathbf{y}}$, we propose the following procedure:

- *Step 1* Generate B samples $\{X_{trainBS,j}\}_{j=1}^{B}$ from the training data X_{train} using bootstrapping.[16]
- *Step 2* Extract explanation rules $C_{WB,j}$ from each bootstrap training sample $X_{trainBS,j}$ (this can be the fine-grained or metafeature representation) and the corresponding labels $\hat{\mathbf{y}}_{\boldsymbol{trainBS,j}}$ predicted by the black-box model. It is important to note that the data-driven metafeatures X_{DDMF-k} need to be computed again for each bootstrap training sample. Obtain B explanations and keep track of the features that are part of the explanations in B sets of features $\{F_j\}_{j=1}^{B}$.
- *Step 3* Make $\frac{B!}{2!(B-2)!}$ pairwise comparisons of the extracted explanations using the Jaccard coefficient. For two sets of features F_v and F_w (respectively representing the features in explanations $C_{WB,v}$ and $C_{WB,w}$), the Jaccard coefficient is defined as: $J(F_v, F_w) = |F_v \cap F_w| / |F_v \cup F_w|$. The Jaccard coefficient equals 1 if the sets are equal (the explanations have perfect overlap of features) and 0 if they are disjoint (the explanations are completely different). For the data-driven metafeatures, two metafeatures are considered to have the same interpretation when the Jaccard coefficient computed for two metafeatures, as measured over the top features with the highest weight, exceeds a cut-off value of c.[17]
- *Step 4* Compute the average (pairwise) Jaccard coefficient over $\frac{B!}{2!(B-2)!}$ comparisons.

Prior research has shown that explanations that rely on high-dimensional data tend to be less robust compared to methods that operate on higher-level features (Alvarez-Melis and Jaakkola, 2018). For this reason, we expect that the extracted explanation rules with metafeatures will be more stable over different training sessions compared to the rules with the original behavioral and textual features. It is important to note, however, that for the metafeatures generated with a data-driven approach (X_{DDMF-k}), the computed stability of the explanations also depends on the metafeature generation method (e.g, NMF). For each bootstrap sample, the data-driven metafeatures are computed again. For the domain-based metafeatures $X_{DomainMF}$ and the original features X_{FG}, the features do not have to be computed for each bootstrap sample, making this part of the rule-extraction process relatively more stable.

4.4.3 Accuracy

Rule-extraction has also been used to increase the generalization performance of white-box models C_{WB}, as measured by accuracy. Martens et al. (2009) show that rules that mimic the behavior of an underlying, better-performing model can become more accurate compared

[15] The procedure is based on the work of Fletcher and Islam (2018) who compare sets of patterns using the Jaccard coefficient.

[16] In the experiments, we set the number of bootstrap samples B to 10.

[17] How many top-features to compare between metafeatures and the cut-off value c, are parameter values that need to be set in advance. We consider looking at the top-20 features in a metafeature and a cut-off value of $c = 0.5$ as suitable choices based on the literature on interpreting factors obtained from dimensionality reduction (De Cnudde et al. 2020).

to the rules learned from the original data and the corresponding ground-truth labels $\mathbf{y}$. Accuracy is defined as the fraction of correctly classified instances by the explanation rules (Huysmans et al., 2006):

$$accuracy^{WB} = \frac{|\{y_i = y_i^{WB} | \mathbf{x}_i\}_{i=1}^{n}|}{N} \tag{2}$$

4.5 Examples of explanation rules

In this subsection, we show examples of the explanation rules extracted from classification models (ℓ_2-regularized Logistic Regression[18]) that predict gender from Facebook "like" data (Praet et al., 2018) and movie viewing data (Harper and Konstan, 2015), and predict whether a news post is about "atheism" (Lang, 1995). Moreover, for the explanation rules based on data-driven metafeatures, we demonstrate how to interpret metafeatures that are part of the explanation. For the other data in the experiments (see Sect. 5.1), the semantic meaning of the features is not publicly available, and for this reason, we do not include examples for these prediction models.

Explanations of the gender prediction model on *Facebook* data with the fine-grained features (Facebook pages), the domain-based metafeatures (the manually crafted categories) and the data-driven metafeatures are respectively shown in Figs. 2, 3 and 4. Table 4 in "Appendix 2" shows the data-driven metafeatures part of the explanation in Fig. 4, and the top-20 Facebook pages with the highest weights for each. The cluster names at the bottom show our interpretation of each metafeature. There are four metafeatures that group similar Facebook pages like "Female media" and "Interior design". Note that the process of interpreting metafeatures might require domain-specific knowledge, but we mainly aim to demonstrate how the interpretation of metafeatures is usually done (Wang and Blei, 2011; O'Callaghan et al., 2015; Contreras-Pina and Sebastián, 2016; Kulkarni et al., 2018; De Cnudde et al., 2019).

Explanation rules for the Logistic Regression model on the *20news* data to predict the topic "atheism" are shown in Figs. 5 and 6 (explanations respectively with the words in a news post and with data-driven metafeatures). Table 5 in "Appendix 3" shows the top-20 words with the highest weight per metafeature (only those part of the explanation in Fig. 6 are shown). For example, there are five metafeatures that group similar words into subtopics like the "Israeli-Palestinian conflict". Interestingly, the explanation with the data-driven metafeatures shown in Fig. 6 reveals a problem with the model: it is overfitting on the posts from Bob Beauchaine and his signature quote containing words like "Manhattan" and the "Bronx" (Metafeature 1 in Table 5). Figure 7 shows an example news post from Beauchaine. When we generate an explanation with the words in the news posts (see Fig. 5), we are not able to diagnose the overfitting on the posts of Beauchaine so easily. This nicely illustrates a specific use case of metafeatures-based explanations of models trained on behavior or text that goes beyond the question whether DDMF-based explanations are more suitable for these models. It shows how DDMF-based explanations can serve as a "tool" for improving the model or gaining insight from it, that is a complement to, and not necessarily a replacement of, FG-based explanations.

[18] In the literature, Logistic Regression with ℓ_2-regularization has shown to be the best-performing classification model for behavioral and textual data (De Cnudde et al., 2020).

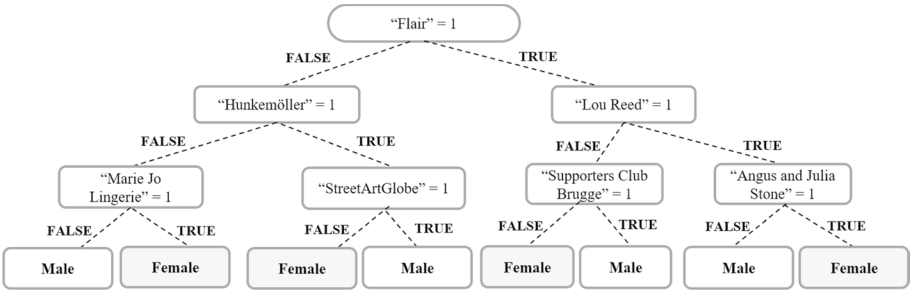

Fig. 2 Example of explanation rules using the fine-grained Facebook pages X_{FG} to explain predictions of the ℓ_2-LR model for *Facebook* data. The global explanation tells us, for example, that the ℓ_2-LR model tends to predict Facebook users as "Female" when they like the magazine "Flair", "Lou Reed", and "Angus and Julia Stone"

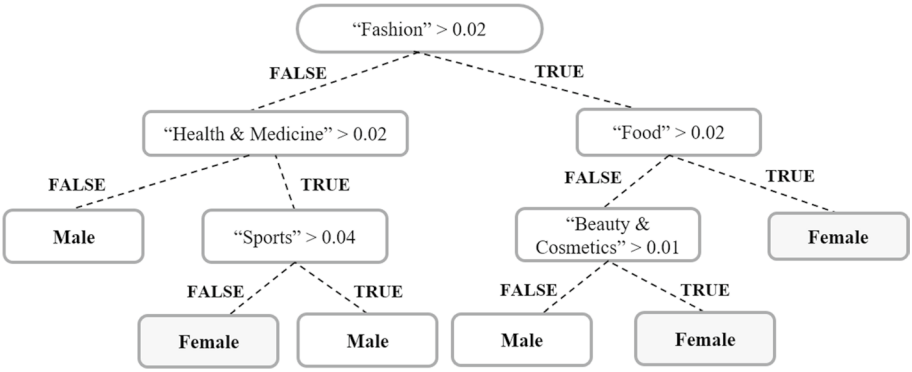

Fig. 3 Example of explanation rules using the domain-based metafeatures $X_{DomainMF}$ to explain predictions of the ℓ_2-LR model for *Facebook* data. The explanation tells us, for example, that the ℓ_2-LR model tends to predict Facebook users as "Female" when more than 2% of their likes belong to the category "Fashion" and more than 2% of their likes are about "Food"

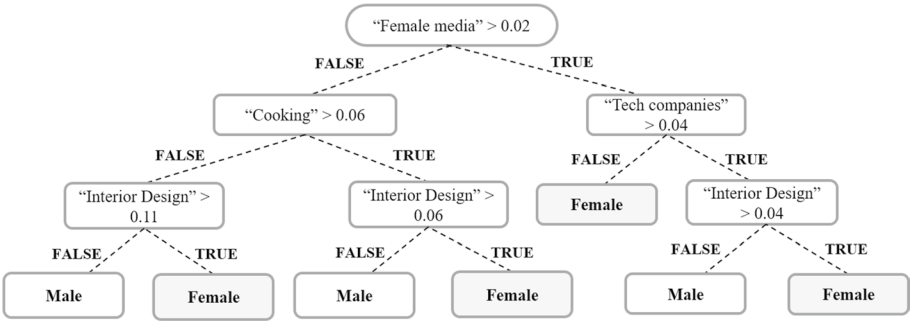

Fig. 4 Example of explanation rules using the data-driven metafeatures X_{DDMF-k} (k=70) to explain predictions of the ℓ_2-LR model for *Facebook* data. The explanation tells us, for example, that the ℓ_2-LR model tends to predict Facebook users as "Female" when less than 2% of their likes belong to the metafeature "Female media", less than 6% of their likes are about "Cooking" and more than 11% of their likes are related to "Interior Design"

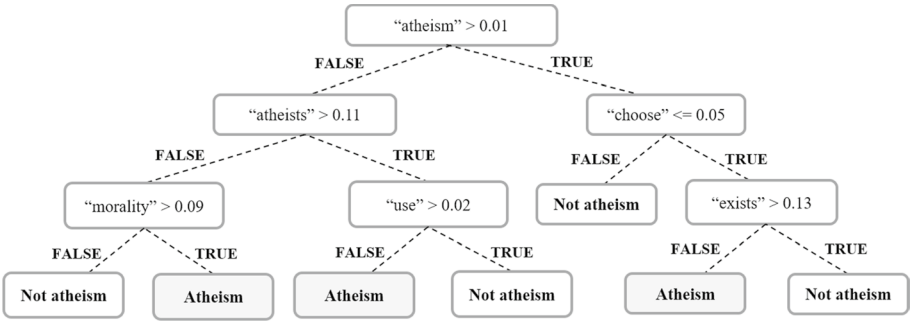

Fig. 5 Example of explanation rules using the fine-grained words X_{FG} to explain predictions of the ℓ_2-LR model for *20news* data. The explanation tells us, for example, that the ℓ_2-LR model tends to predict news posts as "Atheism" when the tf-idf values of "atheism", "atheists" and "morality" are respectively less than 0.01, less than 0.11 and more than 0.09

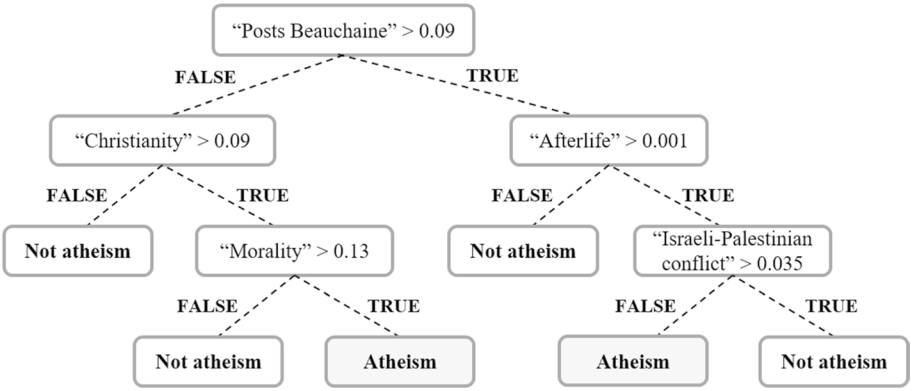

Fig. 6 Example of explanation rules using the data-driven metafeatures X_{DDMF-k} (k=30) to explain predictions of the ℓ_2-LR model for *20news* data. The explanation tells us, for example, that the ℓ_2-LR model tends to predict the topic of news posts as "atheism" when the values of the metafeatures "Posts from Bob Beauchaine", "Afterlife", and "Israeli-Palestinian conflict" are respectively more than 0.09, more than 0.1 and less than 0.035

'In the Old testement, Satan is RARELY mentioned, if at all.
Huh? Doesn't the SDA Bible contain the book of Job?
This is why there is suffering in the world, we are caught inthe crossfire.
(…)
Bob Beauchaine
They **said** that **Queens could stay**, they blew the **Bronx away**,
and **sank Manhattan** out at sea.
(…)'

Fig. 7 News post from the *20news* data with ground-truth topic "atheism". 7.29% of the posts about "atheism" are from Beauchaine and contain the same quote "They said that Queens could stay, they blew the Bronx away, and sank Manhattan out at sea". The words in Metafeature 1 in Table 5 are related to the posts of Beauchaine and clearly represent words of the quote. The model overfitted on the posts of Beauchaine, more specifically, on his name and signature quote, as the explanation in Fig. 6 shows, that contains Metafeature 1

In "Appendix 4", the explanations of a ℓ_2-LR model on movie viewing data (*Movielens1m*) to predict gender are shown. The explanation with data-driven metafeatures tells us that the LR model predicts users as "Female" when at least 2% of the movies they watched has female (lead) characters and at least 3% are drama movies (see Fig. 15). The interpretation of the data-driven metafeatures is shown in Table 6.

5 Experimental setup

The experiments in this article explore the performance of explanation rules with metafeatures versus the original features on which the model is trained. We make a distinction between domain-based metafeatures ($X_{DomainMF}$) and metafeatures generated with a data-driven method (X_{DDMF-k}). The dimensionality reduction parameter k determines the number of metafeatures. The parameter k is fixed for the domain-based, but a hyperparameter for the data-driven metafeatures. We evaluate the performance on a suite of classification tasks using nine behavioral and textual data sets. Figure 16 in "Appendix 5" summarizes the experimental procedure for evaluating the fidelity, f-fidelity and accuracy of explanations using five-fold cross-validation (CV), and the explanation stability using bootstrapping.

5.1 Data sets and models

Our experimental data comprise seven behavioral and two textual data sets. The data sets are summarized in Table 1. The *Movielens100* and *Movielens1m* (Harper and Konstan, 2015) data sets contain movie viewing data from the MovieLens website. We focus on the task of predicting the gender of these users. The *Airline* data[19] contains Twitter data about American airlines, and the task is to predict (positive) sentiment. The Facebook "like" data collected by Praet et al. (2018) (*Facebook*) contains likes from over 6000 individuals in Flanders (Belgium) and is used to predict gender. The *Yahoomovies* data[20] also contains movie viewing behavior, from which we predict gender. The *Tafeng* data (Hsu et al., 2004) consists of fine-grained supermarket transactions, where we predict the age of customers (younger or older than 30) from the products they have purchased. The *20news* data (Lang, 1995) contains about 20,000 news posts. For this data, the task is to predict whether a post belongs to the topic "atheism", based on the words of the post. Another behavioral data set is the *Libimseti* data (Brozovsky and Petricek, 2007), which contains data about profile ratings from users of the Czech social network Libimseti.cz. The prediction task is, again, the gender of the users. Lastly, the *Flickr* data (Cha et al., 2009) contains millions of Flickr pictures and the target variable is the popularity of a picture (the number of comments it has).

All data have a high-dimensional feature space with up to hundreds of thousands of features. *Movielens1m*, *Movielens100* and *Airline* have lower-dimensional feature spaces compared to the other data sets. The large sparsity values ρ for all data indicate that the number of active features is very small compared to the total number of features.

[19] Crowdflower (https://data.world/crowdflower/airline-twitter-sentiment).

[20] Yahoo Webscope Program (https://webscope.sandbox.yahoo.com/).

We train Logistic Regression models with *l*2-regularization (ℓ_2-LR)[21] and Random Forest (RF) models with the *Scikit-learn* library (Python). For training the classification models, we use 80% of the data, and the remaining 20% of the data is used for testing the models. For fine tuning hyperparameters of the model, we use a validation set (20% of the training data). More specifically, the regularization parameter C of the ℓ_2-LR model and the number of trees in the RF model are selected based on the accuracy on the validation set. For preprocessing the text data, we remove stopwords and lemmatize tokens using the *NLTK* package in *Python*, and then use tf-idf[22] vectorization (Joachims, 1998; Martens and Provost, 2014).

Measuring accuracy in practice requires discrete class label predictions, which we obtain by comparing the predicted probabilities to a threshold value t and assigning instances with a predicted probability that exceeds this threshold a positive predicted label. In practice, the choice of the threshold value t depends on the costs associated with false positives and false negatives. In this article, the exact misclassification cost are unknown, and for this reason we compute the threshold value t such that the fraction of instances that are classified as positive equals the fraction of positives in the training data (Lessman et al., 2015). Table 2 indicates the generalization performance of all models for each data set over five folds. In addition to the accuracy, we also report the f-score, precision and recall.

To extract explanation rules with the CART algorithm, we use the *DecisionTree* model of the *Scikit-learn* library. For controlling the complexity of the extracted rules, or equivalently the depth of the tree, we change the value of the *max_depth* parameter. We let the depth of the tree vary from 1 to 5 such that the explanations are cognitively simple (which we motivated in Sect. 4.3).

We extract explanation rules with the original features X_{FG} (on which the classification models are trained) and the metafeatures X_{MF}, and the predicted black-box labels $\hat{\mathbf{y}}$. In the experiments, we generate data-driven metafeatures X_{DDMF-k} based on two approaches (see Sect. 4.2): NMF and SVD. In the experimental results, we mainly discuss the explanations with DDMF generated via NMF (simply denoted by DDMF), that showed the best (fidelity) results. For the *Facebook* and *Movielens1m* data, we also extract explanations with domain-based metafeatures.

6 Experimental results

We compare explanation rules for black-box models extracted with metafeatures against those extracted with fine-grained features, across different classification tasks, data sets and evaluation criteria. As mentioned, our main goal is to better understand how metafeatures affect these different criteria and their trade-offs.

6.1 Are metafeatures better than the original features for explaining models on behavioral and textual data with cognitively simple explanation rules?

Table 3 shows the performance of explanation rules with FG features and metafeatures for the LR models. One of the first key questions related to the performance of the rules is "what is the fidelity", as we want our explanations to mimic the black-box as closely as possible. Overall, the results indicate that the fidelity is higher for DDMF than for the

[21] In the literature, Logistic Regression with ℓ_2-regularization has shown to be the best-performing classification model for behavioral and textual data (De Cnudde et al., 2020).

[22] Tf-idf is short for term frequency and inverse document frequency.

Table 1 Characteristics of the data sets: data type (Type: behavioral(B)/textual(T)), classification task (Target), number of instances (Instances), number of features (Features), number of domain-based metafeatures (DomainMF), balance of the target b (fraction of instances with a positive class label), and sparsity of the data ρ (fraction of zero feature values in the data X_{FG})

Data set	Type	Target	Instances n	Features m	DomainMF	b (%)	ρ (%)
Movielens100	B	Gender	943	1682	n.a.	71.05	93.69
Movielens1m	B	Gender	6040	3883	18	28.29	95.76
Airline	T	Sentiment	14,640	5183	n.a.	16.14	99.82
Facebook	B	Gender	6733	5357	50	32.42	98.19
Yahoomovies	B	Gender	7642	11,915	n.a.	71.13	99.76
Tafeng	B	Age	31,640	23,719	n.a.	45.23	99.90
20news	T	Topic	18,846	41,356	n.a.	4.24	99.87
Libimseti	B	Gender	137,806	166,353	n.a.	44.53	99.93
Flickr	B	Comments	100,000	190,991	n.a.	36.91	99.99

The data is sorted by increasing number of features m

FG-based rules (on average, across all data sets, 6.05%). The rules with DDMF achieve a higher number of wins for both the fidelity and f-fidelity (8 wins versus 1). We use a one-tailed Wilcoxon signed-rank test (Demsar, 2006) to make a statistical comparison between the fidelity of rules with DDMF vs FG features. The test is performed with a sample size of 9 data sets. We find a test statistic $T = 2$ (which is smaller than the critical value $T_c = 3$), hence the difference in fidelity between DDMF and FG is statistically significant at a 1% significance level. The difference in f-fidelity is statistically significant at a 5% level (test statistic of $T = 5$ compared to a critical value of $T_c = 8$).

One notable exception is the *20news* data: the FG-based rules outperform the DDMF-based rules, and the fidelity values are very high while the f-fidelity results are comparably low. This is likely because of the severe (predicted) class imbalance ($b = 4.24\%$ in Table 1) compared to the other data. For this reason, the fidelity criterion might be less suitable for this specific data set. Instead, we could have optimized the depth of the tree and the k of the DDMF on the f-fidelity as measured on the validation set.[23]

Another prominent observation that is, at least at first sight, unexpected is that the optimal tree depth (explanation complexity) does not always reach the maximum of 5. For example, for the *Flickr* data, the optimal complexity of FG-based explanations is a depth of 3. For the FG-based explanations of at most 32 rules, we observe only very small differences in fidelity when varying the complexity (see Fig. 8). However, when we let the complexity of the explanations grow (very large), the fidelity also increases. This is in line with what we would expect: because of the data sparsity and many features being relevant in the model, more complex explanations (larger rule sets) explain a larger fraction of the predictions, resulting in a higher fidelity as shown for the *Flickr* data in Fig. 8.

In order to better understand what drives some of the differences in the performance of explanations with FG features and DDMF, we conjecture this relates to the information

[23] However, for simplicity, we only used fidelity for all data sets.

held at and the coverage of the most predictive features[24]. We look at the Gini impurity reduction (used by the CART algorithm) for different features[25], which we plot in Figs. 17 and 18 in "Appendix 6". The results (visually) indicate that the ratio in Gini impurity reduction of the top-ranked metafeatures and the FG features relate to the difference in fidelity between rules using FG and DDMF features. For example, consider again the *Flickr* data, for which the explanation rules with metafeatures achieve a fidelity of 20.46% higher compared to the FG rules. From Fig. 18c we observe that the top-DDMF holds much more information (larger Gini impurity reduction) than the top-FG feature, which might explain the large fidelity difference between the explanations. Indeed, the correlation coefficient between this ratio (impurity reduction of top-ranked DDMF vs top-ranked FG) and the difference in fidelity between explanations with DDMF vs FG (from Table 3) is 0.81.

Secondly, moving to the stability of the explanations, we observe from Table 3 that the rules with DDMF are similar in stability compared to the FG features (5 wins versus 4). The difference in stability is not statistically significant. It is important to note that for the DDMF-based explanations, there are two sources of instability: computing metafeatures from different bootstrap samples, and extracting explanation rules for different bootstrap samples. When we would "fix" the data-driven metafeatures, and not compute them for different bootstrap samples, the stability of the DDMF-based explanations increases, and is comparable to the DomainMF-based explanations. Furthermore, the stability results can be closely tied to the parameter k. When the optimal dimensionality k of the DDMF is low (for example *Movielens1m* and *Tafeng*), the same DDMF are likely to appear in the global explanation, resulting in more stable explanations over the bootstrap samples. When the selected value of k is higher (for example *20news* and *Airline*), the stability of the explanations with DDMF decreases.

Thirdly, when we compare the accuracy between the rules with DDMF and FG, we observe that the metafeatures-based explanations result in more accurate predictions in regard to the true labels $\mathbf{y}$ (7 wins versus 2). However, using a Wilcoxon signed-rank test, we find that the difference in accuracy is not significant at a 5% level. One data set that stands out is *Libimseti*. For this data set, the fidelity and accuracy for DDMF-based explanations as compared to FG-based explanations is respectively better and worse. Stronger even: the accuracies of the explanations with FG, DDMF and DDMF-SVD (94.38%, 87.94% and 99.63%) are better compared to the accuracy of the black-box model (82.71% in Table 2). Despite the sparsity of this data, there are features that have a large coverage and that are very predictive in regard to the (predicted) target values. For *Libimseti*, there exists a prediction model that has a small number of features (e.g., tree-based model with a depth of 5) that is more accurate compared to models on the full set of behavioral features. As a consequence, this seems not to be a problem instance that requires post-hoc explanations using rule-extraction. This example illustrates that one should always carefully verify first that there are black-box models on the *full* behavioral or textual data that, indeed, outperform intrinsically-interpretable models (e.g., small decision trees or linear models with a small number of features). If not, it may not help to use a black-box model and then compute post-hoc explanations from it (Rudin, 2019). Leaving out the *Libimseti* data and performing the Wilcoxon test on the eight remaining data sets, we find that the differences

[24] The features are either the fine-grained behavioral or textual features, or the metafeatures. With "predictive" we mean predictive in regard to the predicted labels of the black-box model.

[25] We compute the average Gini impurity of the top-FG features and top-MF over five folds.

Table 2 Average performance of black-box classification models (ℓ_2-LR and RF) on the test data using five-fold CV

Data set		Accuracy (%)	f-score (%)	Precision (%)	Recall (%)
Movielens100	ℓ_2-LR	72.75	80.87	80.69	81.08
	RF	73.17	81.19	80.89	81.52
Movielens1m	ℓ_2-LR	78.79	62.60	62.58	63.08
	RF	77.10	59.64	59.56	60.15
Airline	ℓ_2-LR	89.28	66.62	68.07	70.42
	RF	88.05	62.83	64.16	66.37
Facebook	ℓ_2-LR	85.22	77.07	77.53	76.83
	RF	84.79	76.42	76.82	76.25
Yahoomovies	ℓ_2-LR	76.78	83.51	82.70	84.33
	RF	76.54	83.46	83.71	83.24
Tafeng	ℓ_2-LR	67.69	64.98	67.59	62.55
	RF	62.07	57.95	58.19	57.93
20news	ℓ_2-LR	96.59	59.83	59.84	60.03
	RF	96.58	59.70	59.69	59.92
Libimseti	ℓ_2-LR	82.71	82.89	85.61	89.05
	RF	79.29	81.78	84.62	89.31
Flickr	ℓ_2-LR	82.28	76.00	76.02	76.15
	RF	81.17	74.50	74.59	74.61

in fidelity and accuracy between DDMF and FG explanations are statistically significant at a 5% level.

Instead of generating metafeatures using a data-driven method, we can also rely on domain-based metafeatures, crafted by experts. The prominent advantage of this approach is that the resulting metafeatures are (by design) comprehensible. However, they may not always be available. For example, we have such metafeatures for only two of the nine data sets: *Facebook* and *Movielens1m*. When comparing DDMF with domain-based metafeatures for these two data sets, we see again that the fidelity is higher for the DDMF compared to the DomainMF (Table 3 shows that the rules with domain-based metafeatures achieve, at best, test fidelities of 77.66% for *Facebook* and 71.47% for *Movielens1m*), providing further support for using DDMF when developing global explanations for black-boxes. However, when a straightforward semantic meaning of the metafeatures is key, one might still prefer to use DomainMF if they can also increase the fidelity relative to the explanation with FG features (for example for the *Facebook* data).

Finally, we also generate explanations for Random Forest models, for which the performance results are shown in Table 7 in "Appendix 7". We find similar results when explaining RF models compared to explaining LR models, which increases the generalizability of our experimental findings, and further supports using metafeatures to explain models on behavioral and textual data. In the experiments, we also compute explanations with data-driven metafeatures based on the SVD approach. For both the LR models and RF models, the results in Tables 3 and 7 indicate that, overall, the explanations with DDMF-SVD also have a higher fidelity and accuracy than FG-based explanations, but the differences are slightly less prominent compared to the DDMF-based explanations using NMF.

Table 3 Evaluation of explanation rules for ℓ_2-LR model using fine-grained features (FG) and data-driven metafeatures (DDMF) with optimal dimensionality reduction parameter k in parentheses

Data set	Representation	Complexity: tree depth ≤ 5				
		Fidelity(%)	f-fidelity(%)	Stability(%)	Accuracy(%)	Optimal depth
Movielens100	FG	72.43	81.99	**5.91**	68.93	3
	DDMF (100)	**75.29**	**84.06**	5.52	**72.00**	4
	DDMF-SVD (10)	72.54	82.47	65.38	70.09	5
Movielens1m	FG	75.53	34.43	8.53	73.29	5
	DDMF (10)	**78.92**	**57.78**	**75.18**	**74.24**	4
	DDMF-SVD (30)	77.89	54.12	30.81	73.08	4
	DomainMF	71.47	21.06	46.01	70.29	3
Airline	FG	90.53	64.04	**33.13**	87.43	4
	DDMF (700)	**90.79**	**65.53**	17.66	**88.03**	5
	DDMF-SVD (700)	89.62	57.88	17.66	87.08	5
Facebook	FG	75.08	41.36	11.71	74.83	5
	DDMF (70)	**81.73**	**67.99**	**18.81**	**79.65**	5
	DDMF-SVD (10)	78.30	63.74	65.32	75.91	5
	DomainMF	77.66	59.38	50.44	76.04	5
Yahoomovies	FG	77.32	85.59	**22.62**	72.85	5
	DDMF (100)	**80.41**	**86.81**	18.91	**74.05**	5
	DDMF-SVD (50)	77.48	85.25	25.88	72.43	5
Tafeng	FG	68.72	58.27	20.43	57.43	5
	DDMF (10)	**69.39**	**62.15**	**76.62**	**58.70**	5
	DDMF-SVD (300)	69.19	62.07	14.28	57.75	5
20news	FG	**96.38**	**32.68**	**26.48**	**96.12**	3
	DDMF (70)	96.11	27.67	15.78	95.75	3
	DDMF-SVD (700)	95.90	20.62	4.01	95.74	4
Libimseti	FG	77.18	76.72	24.34	**94.38**	5
	DDMF (10)	**94.52**	**94.11**	**65.39**	87.94	5
	DDMF-SVD (10)	93.12	92.59	68.03	99.63	5
Flickr	FG	63.23	0.84	38.36	63.20	3
	DDMF (30)	**83.69**	**78.09**	**38.69**	**79.74**	5
	DDMF-SVD (30)	83.87	78.12	32.43	78.87	5
	# wins DDMF vs FG	$8-1$	$8-1$	$5-4$	$7-2$	
	Mean difference DDMF vs FG (σ)	6.05 (7.18)	16.47 (23.87)	8.82 (37.65)	2.40 (5.77)	

The best performance values (FG vs DDMF) are indicated in bold. The average fidelity, f-fidelity and accuracy on the test data are reported over five-fold CV. The stability is measured over 10 bootstrap samples. The optimal complexity (tree depth) is shown in the last column. We also report the results for explanations with DDMF generated via SVD (*DDMF-SVD*). For *Facebook* and *Movielens1m*, we also report results for the domain-based metafeatures (*DomainMF*)

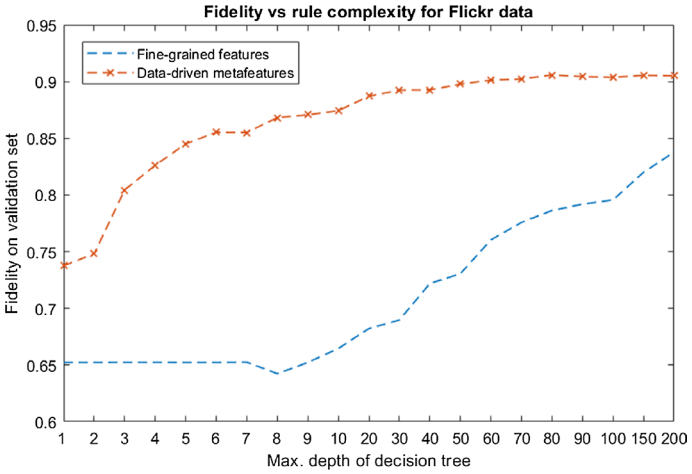

Fig. 8 Fidelity on validation set of explanation rules for varying complexity settings (tree depths ranging from 1 to 200) when explaining the LR model on the *Flickr* data

6.2 How does the fidelity of explanation rules vary for different complexity settings?

Figure 9 plots the difference in average test fidelity between rules with DDMF and rules with FG features against the maximum allowed explanation complexity (equivalent to the tree depth). Points above the horizontal line are data sets for which the rules with DDMF perform better. The graph clearly shows that for the majority of data, and for varying complexity settings, the DDMF representation performs better than the FG (differences larger than 0). Only for the *Tafeng*, *Airline* and *20news* data, the differences are sometimes not positive, indicating that for these complexity settings, the average test fidelity for the rules with FG features is best. In general, from this plot, we can conclude that the findings of Table 3 hold for varying complexity settings, and that the fidelity is generally higher for explanations with the DDMF representation compared to the FG representation.

Figure 10 plots the average test fidelity against the maximal allowed explanation complexity for FG (10a) and DDMF (10b) explanation rules. We observe that, as one would

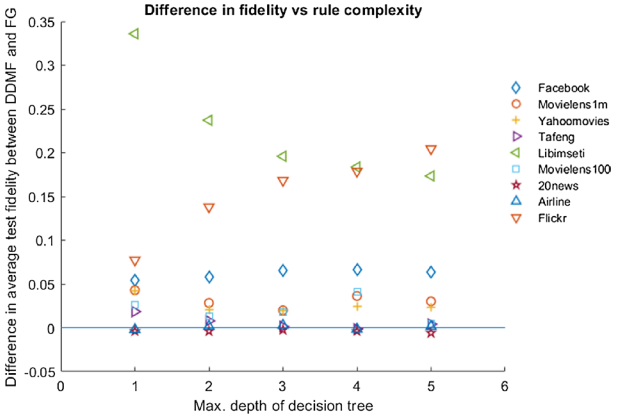

Fig. 9 Difference in average test fidelity of rules with DDMF and FG features in percentage points for varying complexity settings (tree depths from 1 to 5)

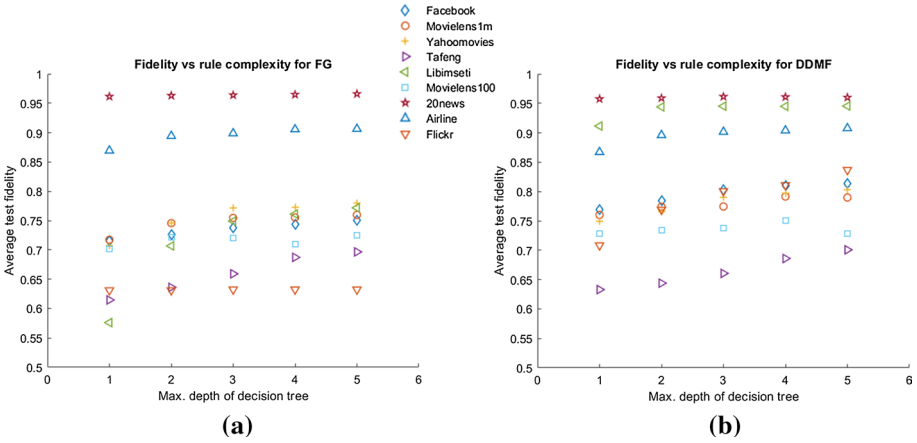

Fig. 10 Average test fidelity of rules with (**a**) FG features and (**b**) DDMF for varying complexity settings (tree depths from 1 to 5)

expect, for all data sets, there is generally an increasing fidelity when we increase the depth of the decision tree, or equivalently, the complexity of the explanation rules. Interestingly, for some data sets, this fidelity-complexity trade-off is less severe. For example, for the *20news* and *Movielens100* data, the slopes of the curves are relatively flat. These results also indicate that in some cases, there may not be much to gain by using a relatively "more complex" explanation. Therefore, once one is willing to trade-off fidelity for complexity, in some cases, one might as well choose an "extremely" simple explanation. For the *20news* data, we already pointed at the f-fidelity being a more suitable measure than fidelity because of the class imbalance, which explains the marginal increase in fidelity when increasing complexity.

6.3 How does the number of generated data-driven metafeatures (dimensionality reduction parameter *k*) impact fidelity and stability?

A key parameter in our metafeatures-based rule-extraction methodology is the dimensionality reduction parameter k. For DomainMF, this k is fixed. For DDMF, we have been selecting the value of k that maximizes the fidelity of the explanation rules on the validation data. As this k may be an important parameter that defines the dimensionality of the space where rule-extraction methods operate (and their performance), we also investigate to what extent the quality—both fidelity and stability—of rules extracted using DDMF depends on this parameter.[26] Although fidelity can be considered the most important evaluation metric, in practice, one may wish to tune parameters such as k on a desired combination of fidelity, stability and accuracy[27] depending on the context.

[26] One can do such an analysis for other parameters, too, in general.

[27] As mentioned earlier, we focus on fidelity—namely how well we can mimic the black-box—instead of accuracy. All analyses can be done for either of the two—or for both—although trade-off decisions become more complex when one uses many criteria.

Figures 11 and 12 show the average fidelity and the stability for varying values of k used and explanation complexities. Firstly, looking at the figures depicted on the left, we observe that, for most data, the fidelity increases with a higher number of metafeatures up until a certain point, after which fidelity decreases again. This turnover point varies per data set, and also depends on the complexity of the explanation (the tree depth). Therefore, an important implication is to select the optimal number of metafeatures on a separate validation set, as we also do. Interestingly, fidelity behaves similarly to how (out-of-sample) accuracy typically does as complexity increases: for both measures there is some sort of "overfitting" to the black-box training data in case of including too many metafeatures.

On the other hand, for stability (all figures shown on the right), we observe that, overall, the stability of the extracted rules decreases with a higher number of k, especially when allowing for a larger explanation complexity. For example, the stability of rules with DDMF with $k = 10$ is generally larger than the stability with DDMF with $k = 700$. For a lower value of k, the dimensionality and sparsity of the metafeatures are lower, making the same metafeature more likely to appear in explanations from different bootstrap samples (as also explained in Sect. 6.1). Interestingly, these figures also show that there is a fidelity-stability trade-off. While fidelity generally increases (at first) with the number of metafeatures and the explanation complexity, stability does not. This may also impact the "optimal" number of generated metafeatures k, or any parameter selection for any explanation methodology.

7 Conclusion

The fine-grained level of the features that are typically observed in behavioral and textual data sets are of great value for predictive modeling. Feature selection methods or dimensionality reduction techniques to come to a reduced set of "metafeatures" have been shown in the literature to lead to lower accuracies (Junqué de Fortuny et al., 2013; Clark and Provost, 2015; De Cnudde et al., 2019) for models mining these types of data. On the other hand, we have shown empirically using a number of data sets, and for Logistic Regression and Random Forest models as black-boxes, that these metafeatures are of great value to *explain* the complex prediction models built on the fine-grained features. The results indicate that explanation rules extracted with data-driven metafeatures are better able to mimic the black-box models than those extracted using the behavioral or textual features on which the model was trained. As such, metafeatures help to improve the fidelity: concise rule sets that explain a large(r) percentage of the black-box's predictions (higher fidelity) can be obtained.

Exploring *when* our solution of metafeatures-based rule extraction works best, our empirical results show a strong indication that the relative gain of using metafeatures to extract explanations is positively related to the sparsity of the most important fine-grained predictors in the model. When the black-box model is not characterized by the problems we try to address (high dimensionality, sparsity, and many relevant predictors for the classification task at hand), explanation rules with metafeatures will be as good as or worse than explanations with the original features. However, they can still provide the user with different types of insights on the model's behavior, that would not (as easily) be identified when looking at rules extracted with the original data, rendering them also valuable in this context.

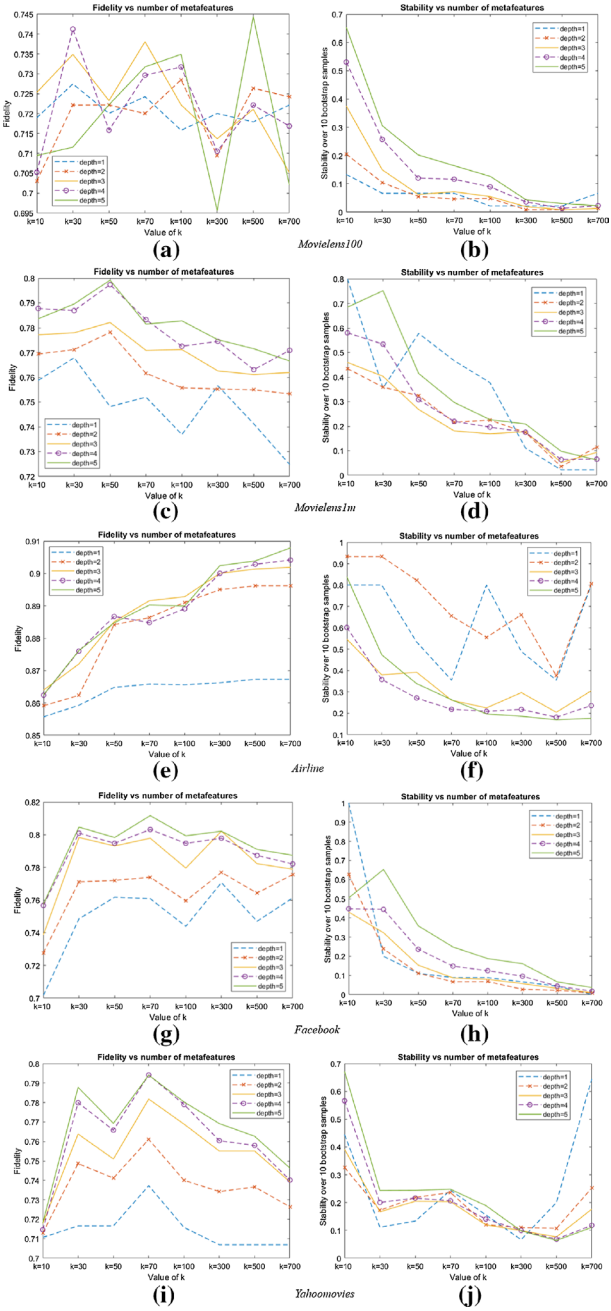

Fig. 11 Average test fidelity and stability for rules with DDMF for varying values of k (number of metafeatures) and complexities for data sets *Movielens100*, *Movielens1m*, *Airline*, *Facebook*, and *Yahoomovies*

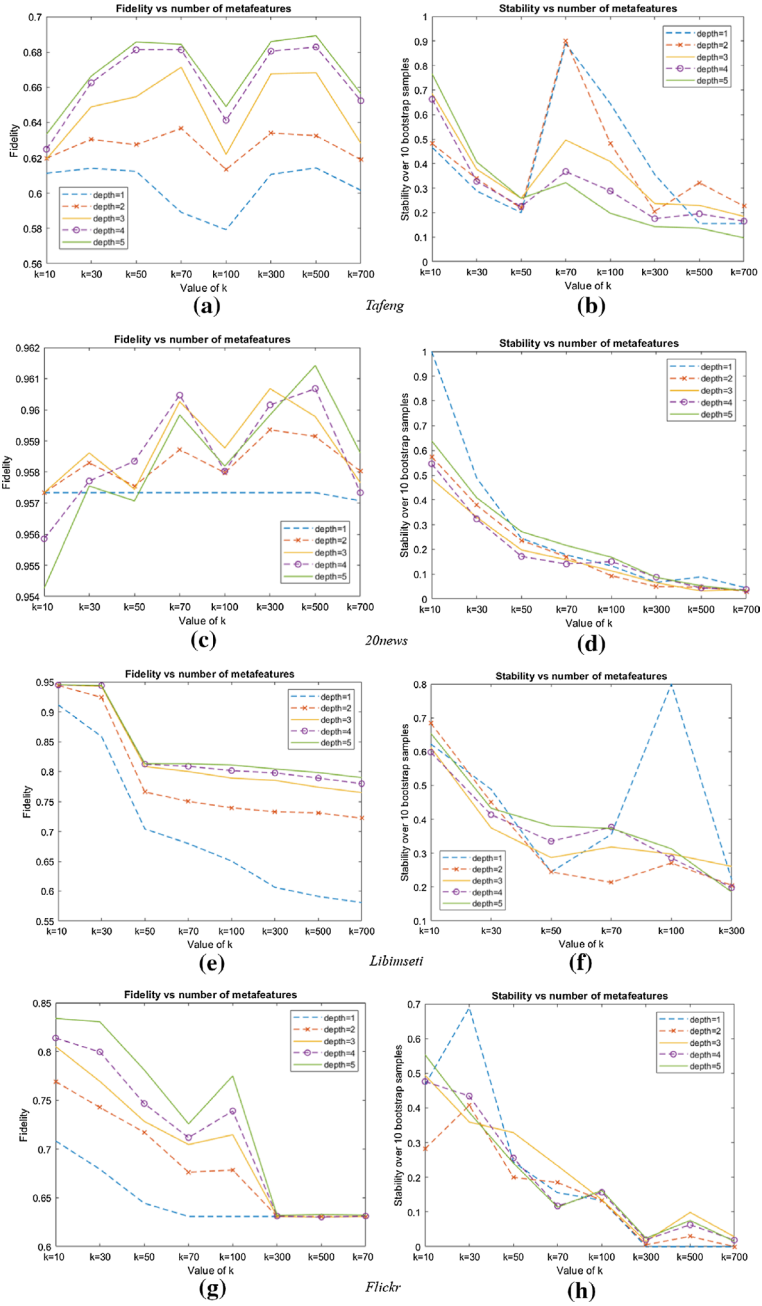

Fig. 12 Average test fidelity and stability for rules with DDMF for varying values of k (number of metafeatures) and complexities for data sets *Tafeng*, *20news*, *Libimseti* and *Flickr*

Our empirical results also show important trade-offs between the quality measures of the explanation rules that we considered. For example, more complex explanations

(larger rule sets) tend to lead to higher fidelity but lower stability. An interesting implication of our empirical findings is that one should carefully fine tune any parameter of their explainability method, such as the number of generated metafeatures in our methodology, in order to obtain the desired trade-offs. In our case, increasing the number of generated metafeatures has shown to result in lower stability of the extracted rules, whereas the impact on fidelity is not straightforward and depends on the data set and the complexity setting.

In this article, we mainly focused on the fidelity of explanation rules in regard to the black-box model. For future research, there are some other important directions to explore for evaluating post-hoc explanations of prediction models: the computational cost to achieve the explanations, the cost of having an explanation rule set with an accuracy that is lower than the black-box model, or the issue of presenting only one rule set as explanation, while other rule sets with similar fidelity and accuracy might exist. Although these aspects are implicitly addressed in our article, a more qualitative study on how these "costs" are perceived by users can be another interesting issue for future research. On a methodological level, this study could spur future research on the use of other feature engineering techniques such as embeddings to be used in metafeatures-based explanation rules. One interesting approach is to include the fidelity, stability, accuracy, and complexity measures explicitly when constructing the metafeatures.

Finally, our metafeatures-based explanation approach for high-dimensional, sparse behavioral and textual data has important practical implications for any setting where such data is available and explainability is an important requirement, be it for model acceptance, validation, insight, or improvement. This article could therefore potentially lead to a wider use of valuable behavioral and textual data in different domains, among others, marketing and fraud detection.

Appendix

Appendix 1: Procedure for generating data-driven metafeatures

See Fig. 13.

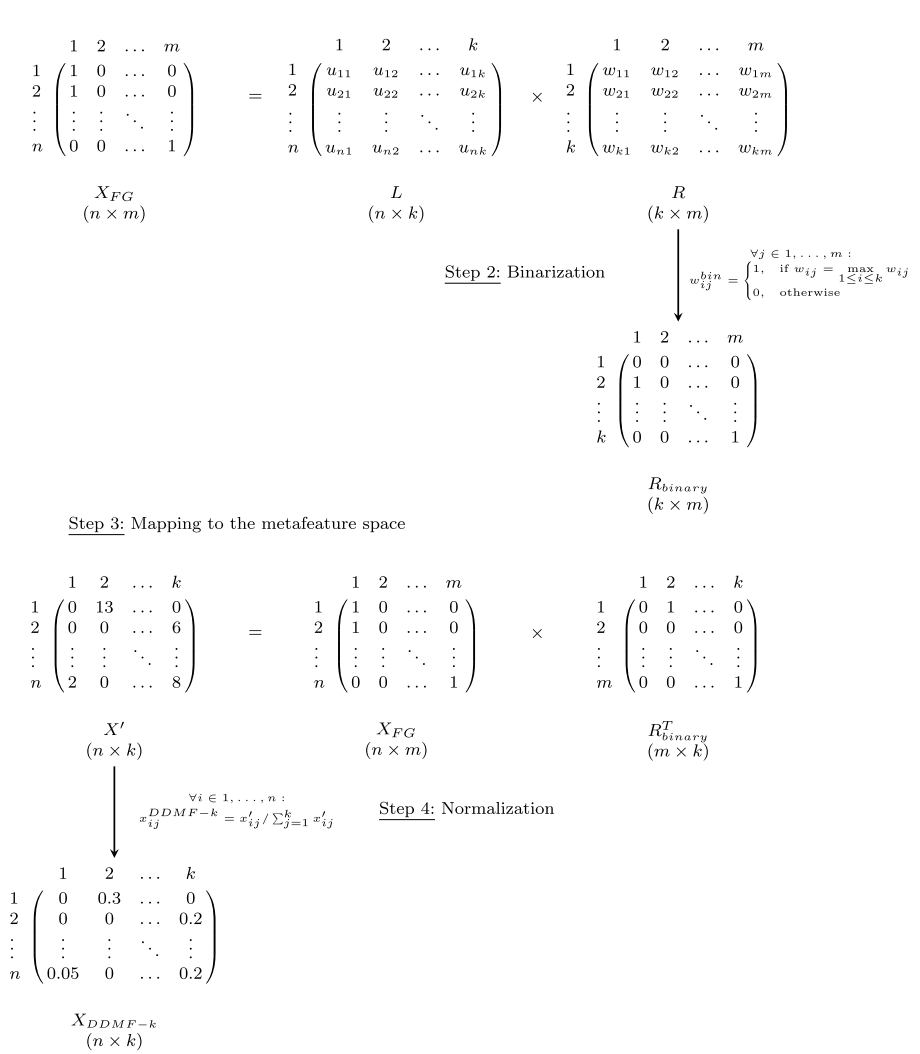

Fig. 13 Procedure for generating DDMF using dimensionality reduction with matrix factorization. Note that X_{FG} can also contain numerical features

Appendix 2: Explaining the ℓ_2-LR model on *Facebook* data to predict gender using fine-grained features and metafeatures

See Table 4.

Table 4 Interpretation of data-driven metafeatures ($k = 70$) of *Facebook* data by investigating top-20 features with highest coefficient. We only interpret the metafeatures that are part of the explanation for the ℓ_2 -LR model to predict gender (see Fig. 4). The "cluster names" at the bottom show our interpretation of the metafeatures based on the Facebook pages with the highest coefficient

Metafeature 1	Metafeature 2	Metafeature 3	Metafeature 4
H&M	Dagelijkse kost	Coolblue	IKEA
Flair	Standaard Uitgeverij	Microsoft	Decovry.com
Gossip Girl	Lidl Belgium	Samsung	Eva Mouton
Adele	Lekker van bij ons	Windows	Tasty
Humans of New York	ZOO Planckendael	Telenet	Espoo.design store
Ed Sheeran	Libelle.be	Mobile Vikings	Brussels Airlines
Tasty	Alpro	bol.com	MADE.com
De Slimste Mens ter Wereld	bol.com	OnePlus	Furnified
ZARA	Radio 2	Takeaway.com	newplacestobe.com
Jamie's World	Vente-Exclusive.com	Google	LILY—Life's Little Luxuries
The fault in our stars	Gezondheid.be	Netflix	Knack Weekend
ELLE België	Jobat.be	PlayStation België	Bloovi
Bokken voor bij het blokken	Ish Ait Hamou	Proximus	Sandra Bekkari
Ellie Goulding	Vlaanderen Vakantieland	BMW Belgium	VakantiePiraten.nl
Loïc	IKEA	Telenet vaste klanten	Charlie
The Notebook	Zin in meer	iBOOD.be	Belmodo
Pretty Little Liars	Bose	Smartmat	ELLE België
VIJF	She.be	Audi	MONOQI
Awkward	UiTinVlaanderen	Game Mania	Ugly Belgian Houses
Belgian Red Devils	Uitgeverij Lannoo	Volvo Car BeLux	Woonblog
Female media	**Cooking**	**Tech companies**	**Interior Design**

Appendix 3: Explaining the ℓ_2-LR model on *20news* data to predict the topic "atheism" using fine-grained features and metafeatures

See Table 5.

Table 5 Interpretation of data-driven metafeatures ($k = 30$) of *20news* data by investigating top-20 features with highest coefficients. We only interpret the metafeatures that are part of the explanation for the ℓ_2-LR model (see Fig. 6). The "cluster names" at the bottom show our interpretation of the generated metafeatures based on the words with the highest coefficient

Metafeature 1	Metafeature 2	Metafeature 3	Metafeature 4	Metafeature 5
Could	God	Christian	Israel	System
Find	Faith	Book	Jews	Moral
Said	Sin	True	Israeli	Morality
Give	Bell	Christians	Arab	Systems
Away	Angels	Evidence	Jewish	Running
Tell	Lord	Faith	Arabs	Operating
Bobbe	Existence	Read	Israelis	Objective
Ico	Bible	Christianity	Palestinians	Memory
Someone	Heaven	Even	Peace	Ini
Beauchaine	Man	Life	Lebanon	Duo
Queens	Love	Bible	Lebanese	Change
Bronx	Eternal	Truth	Palestinian	Necessary
Tek	Must	Word	State	Based
Sank	Belief	Find	Land	Information
Manhattan	Believe	Kent	Adam	Ram
Com	Exists	Love	Gaza	Gateaway
Nlew	Exist	Cheers	Palestine	Ranking
Bob	Christ	Meaning	War	Control
Vice	Satan	Claim	Killed	Ntfs
Stay	Word	Quite	Anti	Dialing
Posts from Bob Beauchaine	**Afterlife**	**Christianity**	**Israeli-Palestinian conflict**	**Morality**

Appendix 4: Explaining the ℓ_2-LR model on *Movielens1m* data to predict gender using fine-grained features and metafeatures

See Figs. 14, 15 and Table 6.

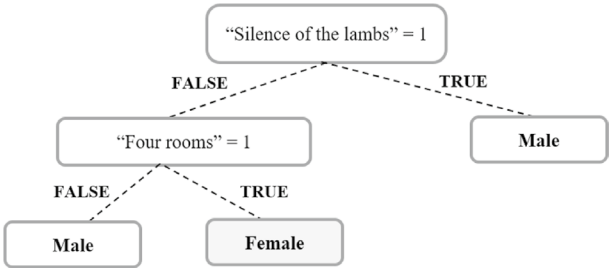

Fig. 14 Example of explanation rules using the movies X_{FG} to explain predictions of the ℓ_2-LR model for *Movielens1m* data. The explanation tells us, for example, that the ℓ_2-LR model likely predicts users as "Female" when they watched the movie "Four rooms" but they didn't watch "Silence of the lambs"

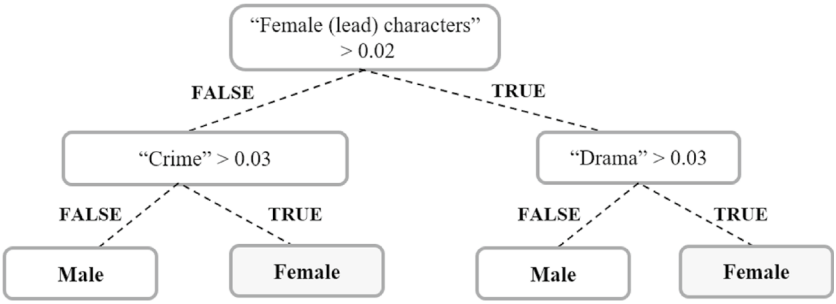

Fig. 15 Example of explanation rules using the data-driven metafeatures X_{DDMF} to explain predictions of the ℓ_2-LR model for *Movielens1m* data. The explanation tells us, for example, that the ℓ_2-LR model likely predicts users as "Female" when at least 2% of the movies they watched have female lead characters and at least 3% are drama movies

Table 6 Interpretation of data-driven metafeatures ($k = 70$) of *Movielens1m* data by investigating top-20 features with highest coefficients. We only interpret the metafeatures that are part of the explanation for the ℓ_2-LR model (see Fig. 15). The "cluster names" at the bottom show our interpretation of the generated metafeatures based on the movies with the highest coefficient

Metafeature 1	Metafeature 2	Metafeature 3
My fair lady	l'enfer	So I married an axe murderer
Four rooms	The untouchables	The harmonists
All about eve	Outbreak	The wooden man's bride
An affair to remember	Cross of iron	The kindred
The rescuers down under	Chariots of fire	My own private idaho
The amityville horror	The sexual life of the belgians	The baby-sitters club
Deadly friend	Ghost dog: the way of the samurai	Evita
Rebecca	Little buddha	I love trouble
Anna	October sky	Tough and deadly
The manchurian candidate	The ghost of frankenstein	Dracula: dead and loving it
The women	Homeward bound 2	Muppet treasure island
The apple dumpling gang rides	Castle freak	Sphere
Heathers	Disclosure	It could happen to you
Twisted	Love! valour! compassion!	An American werewolf in Paris
Charade	Seven days in may	Nil by mouth
Naked	Titanic	A goofy movie
The adventures of robin hood	White man's burden	Tom and Huck
It could happen to you	Selena	The good, the bad and the ugly
Army of darkness	A league of their own	Mouth to mouth
The alarmist	Seven	When a man loves a woman
Female (lead) characters	**Crime**	**Drama**

Appendix 5: Experimental procedure for evaluating fidelity, f-fidelity, accuracy, and stability of explanation rules

See Fig. 16.

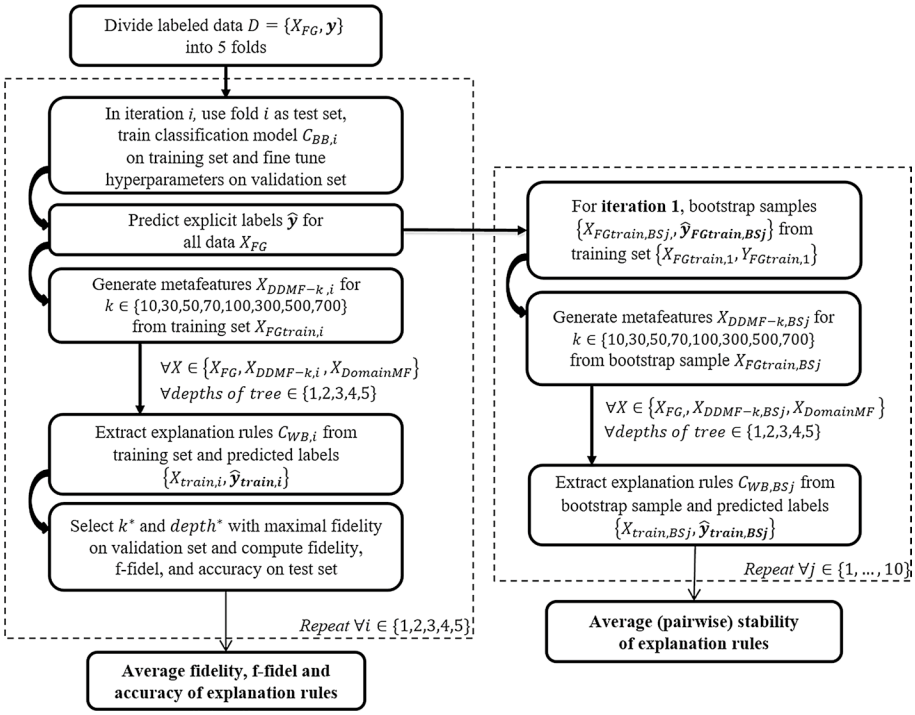

Fig. 16 Experimental procedure of evaluating fidelity, f-fidelity and accuracy of explanation rules C_{WB} with five-fold CV, and stability using bootstrap samples, using fine-grained features (FG), data-driven metafeatures (DDMF) and domain-based metafeatures (DomainMF), and varying complexity settings for explanations (this is equivalent to the tree depth)

Appendix 6: Gini impurity reductions

See Figs. 17 and 18.

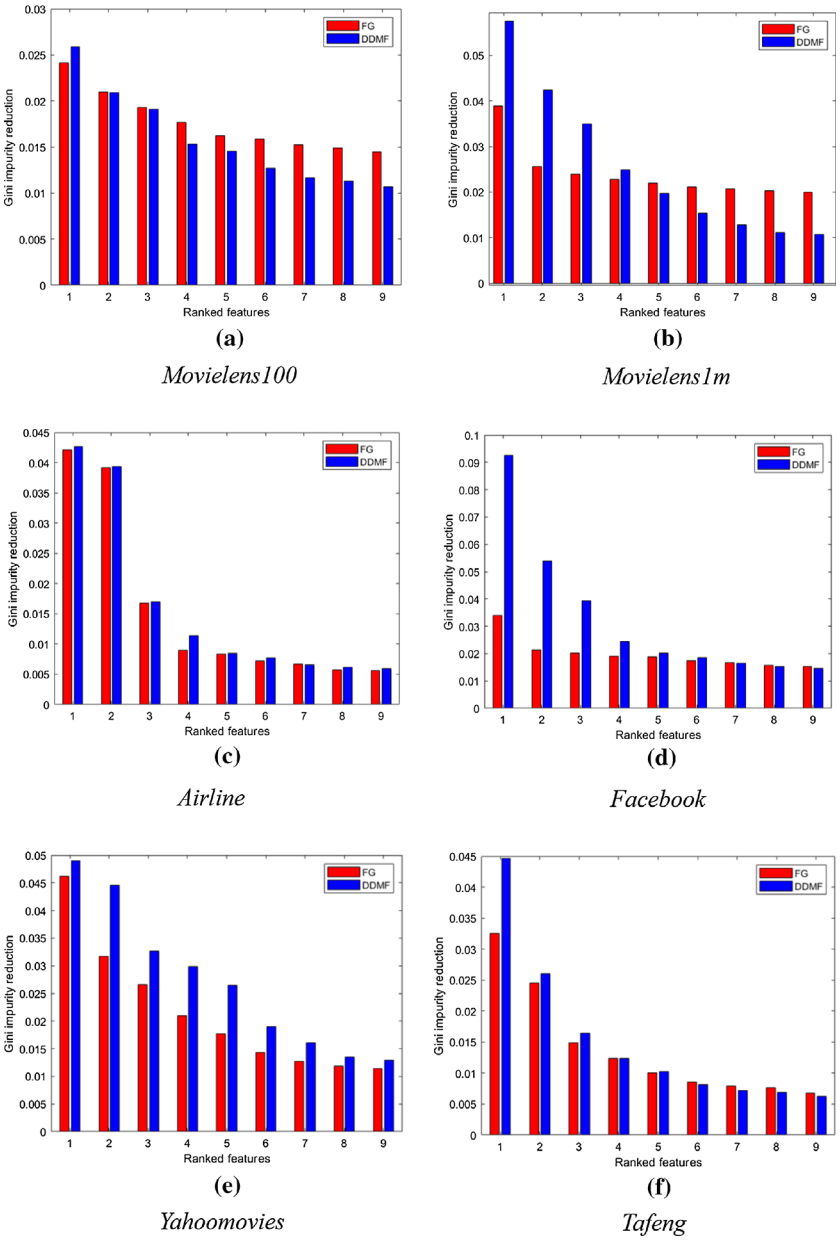

Fig. 17 Top-ranked features with highest Gini impurity reduction for each data representation for the ℓ_2-LR model as the black-box model. The reductions are averaged over five folds

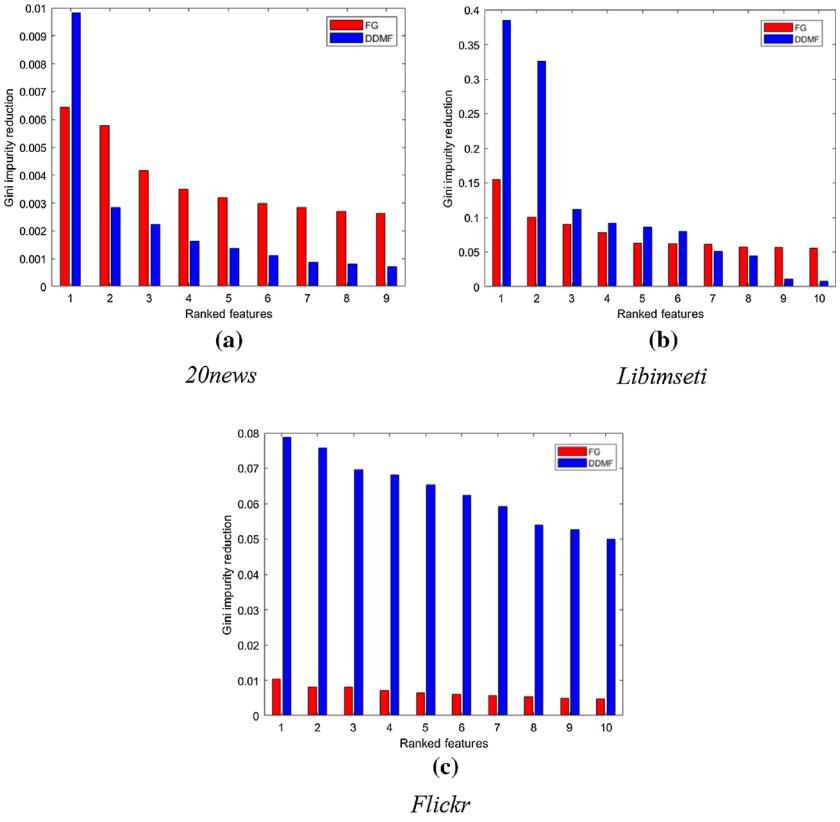

Fig. 18 Top-ranked features with highest Gini impurity reduction for each data representation for the ℓ_2-LR model as the black-box model. The reductions are averaged over five folds

Appendix 7: Experimental results of explanations for the Random Forest models

See Table 7.

Table 7 Evaluation of explanation rules for RF model using fine-grained features (FG) and data-driven metafeatures (DDMF) with optimal dimensionality reduction parameter k in parentheses

Data set	Representation	Complexity: tree depth ≤ 5				
		Fidelity(%)	f-fidelity(%)	Stability(%)	Accuracy(%)	Optimal depth
Movielens100	FG	71.05	82.43	3.74	70.31	4
	DDMF (70)	**75.19**	**84.70**	**10**.05	**71.69**	4
	DDMF-SVD (10)	71.79	81.78	18.29	69.78	2
Movielens1m	FG	78.05	46.49	8.83	73.49	5
	DDMF (10)	**81**.16	**59**.66	**43**.07	**74**.62	3
	DDMF-SVD (10)	78.06	48.03	54.16	73.77	4
	DomainMF	72.35	8.89	46.61	71.77	3
Airline	FG	91.67	67.91	**29**.49	87.47	5
	DDMF (700)	**92.31**	**72.23**	16.53	**87.60**	5
	DDMF-SVD (700)	90.25	60.79	17.59	86.81	5
Facebook	FG	77.29	47.02	**13**.38	75.09	5
	DDMF (300)	**82**.40	**70**.67	5.44	**79**.85	5
	DDMF-SVD (10)	78.87	64.54	65.77	76.19	5
	DomainMF	78.91	60.51	43.69	76.44	3
Yahoomovies	FG	78.57	86.43	**19**.03	73.27	5
	DDMF (100)	**79**.85	**86**.88	14.76	**74**.39	5
	DDMF-SVD (300)	76.98	85.34	9.29	72.78	5
Tafeng	FG	60.74	37.36	19.04	57.00	5
	DDMF (50)	**63**.19	**49.18**	**24**.51	**58**.45	5
	DDMF-SVD (50)	64.52	54.68	25.68	58.25	5
20news	FG	96.17	29.11	**13**.11	**96.11**	5
	DDMF (500)	**96**.24	**31**.22	4.69	95.89	5
	DDMF-SVD (70)	95.89	18.69	14.05	95.64	4
Libimseti	FG	73.74	75.84	14.45	**94**.33	5
	DDMF (10)	**91**.15	**92**.36	**67**.20	87.93	5
	DDMF-SVD (10)	91.06	92.27	66.58	87.85	5
Flickr	FG	63.12	0.18	**46**.67	63.05	1
	DDMF (30)	**81.49**	**74.78**	40.69	**79.97**	5
	DDMF-SVD (30)	80.91	75.36	34.34	79.33	5
	# wins DDMF vs FG	9 – 0	9 – 0	4 – 5	7 – 2	
	Mean difference DDMF-FG (σ)	5.84 (6.62)	16.55 (21.79)	6.58 (21.06)	2.25 (5.88)	

The best performance values (FG vs DDMF) are indicated in bold. The average fidelity, f-fidelity and accuracy on the test data are reported over five-fold CV. The stability is measured over 10 bootstrap samples. The optimal complexity (tree depth) is shown in the last column. We also report the results for explanations with DDMF generated via SVD (*DDMF-SVD*). For *Facebook* and *Movielens1m*, we also report results for the domain-based metafeatures (*DomainMF*)

Funding Funding was provided by Research Foundation – Flanders (Grant No. 11G4319N).

References

Alvarez-Melis, D., & Jaakkola, T. S. (2018). Towards Robust Interpretability with Self-Explaining Neural Networks, arxiv:1806.07538

Andrews, R., & Diederich, J. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems, 8*(6), 373–389.

Attenberg, J., Weinberger, K., Smola, Q., Dasgupta, A., Zinkevich, M. (2009). Collaborative email-spam filtering with the hashing-trick. In *Proceedings of the 6th conference on email and anti-spam*.

Bache, K, Lichman, M. UCI machine learning repository. *School Inf. Comput. Sci. Univ. California*. http://archive.ics.uci.edu/ml

Brozovsky, L., & Petricek, V. (2007). Recommender system for online dating service. In *Proceedings of conference Znalosti, VSB, Ostrava*. Czech Republic.

Campbell, D. (1988). Task complexity: a review and analysis. *Academy of Management Journal, 13*(1), 40–52.

Cha, M., Mislove, A., & Gummadi, K. P. (2009). A measurement-driven analysis of information propagation in the flickr social network. In *Proceedings of the 18th international world wide web conference*. https://doi.org/10.1145/1526709.1526806

Chen, D., Fraiberger, S. P., Moakler, R., & Provost, F. (2017). Enhancing transparency and control when drawing data-driven inferences about individuals. *Big Data, 5*(3), 197–212.

Chen, W., Zhang, M., Zhang, Y., & Duan, X. (2016). Exploiting meta features for dependency parsing and part-of-speech tagging. *Artificial Intelligence, 230,* 173–191.

Chhatwal, R., Gronvall, P., Huber, N., Keeling, R., Zhang, J., & Zhao, H. (2019) Explainable text classification in legal document review: A case study of explainable predictive coding, CoRR, abs/1904.01721.

Contreras-Pina, C., & Sebastián, A.-R. (2016). An empirical comparison of latent sematic models for applications in industry. *Neurocomputing, 179,* 176–185.

Clark, J., & Provost, F. (2015). Dimensionality reduction via matrix factorization for predictive modeling from large, sparse behavioral data. *Data Mining and Knowledge Discovery, 33*(4), 871–916.

Cohen, W.W. (1995). Fast effective rule induction. In A. Prieditis & S. Russell (Eds.), *Proceedings of the 12th international conference on machine learning* (pp. 115–123). Morgan Kaufmann.

Craven, M., & Shavlik, J. (1999). Rule extraction: Where do we go from here? In *Proceedings of machine learning research group working paper* (pp. 1–6).

De Cnudde, S., Martens, D., Evgeniou, T., & Provost, F. (2020). A benchmarking study of classification techniques for behavioral data. *International Journal of Data Science and Analytics, 9*, 131–173. https://doi.org/10.1007/s41060-019-00185-1

De Cnudde, S., Moeyersoms, J., Stankova, M., & Martens, D. (2018). What does your Facebook profile reveal about your creditworthiness? Using alternative data for microfinance. *Journal of the Operational Research Society, 70*(10), 1–10.

De Cnudde, S., Ramon, Y., Martens, D., & Provost, F. (2019). Deep learning on big. *Sparse, Behavioral Data, Big Data, 7*(4), 286–307.

de Fortuny, E. J., & Martens, D. (2015). Active learning-based pedagogical rule extraction. *IEEE Transactions on Neural Networks and Learning Systems, 26*(11), 2664–2677.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *JMLR, 7*(1), 1–30.

Devlin, J., Chang, M. -W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Human language technologies, Volume 1 (Long and Short Papers)* (pp. 4171–4186).

Diederich, J, (2008). Rule extraction from spport vector machines: An introduction. In: Diederich J. (Ed.), *Rule extraction from support vector machines. Studies in computational intelligence*, vol 80. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-75390-2_1

European Commission White Paper. (2020). On artificial intelligence—A European approach to excellence and trust.

European Union, Counil Directive 2004/113/EC, art.3; European Union, Council Directive 2004/43/EC of 29 June 2000 implementing the principle of equal treatment between persons irrespective of racial or ethnic origin, OJ L 180 (19 July 2000), art.3.

Fletcher, S., & Islam, M. Z. (2018). Comparing sets of patterns with the Jaccard index. *Australasian Journal of Information Systems*. https://doi.org/10.3127/ajis.v22i0.1538

Freitas, A. A. (2013). Comprehensible classification models: A position paper. *ACM SIGKDD Explorations*, *15*(1), 1–10. https://doi.org/10.1145/2594473.2594475.

Gigerenzer, G., & Goldstein, D. G. (2016). Reasoning the fast and frugal way: Models of bounded rationality. *Psychological Review, 103*(4), 650–669.
Harper, F. M., & Konstan, J. A. (2015). The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*. https://doi.org/10.1145/2827872
Hauser, J. R., Toubia, O., Evgeniou, T., Befurt, R., & Silinskaia, D. (2009). Disjunctions of conjunctions, cognitive simplicity and consideration sets. *Journal of Marketing Research*.
Hsu, C.-N., Chung, H.-H., & Huang, H.-S. (2004). Mining skewed and sparse transaction data for personalized shopping recommendation. *Machine Learning, 57*(1), 35–59.
Husbands, P., Simon, H., & Ding, C. (2001). On the use of the singular value decomposition for text retrieval. *Computational Information Retrieval, 5,* 145–156.
Huysmans, J., Baesens, B., & Vanthienen, J. (2006). Using rule extraction to improve the comprehensibility of predictive models. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.961358.
Huysmans, J., Dejaeger, K., Mues, C., Vanthienen, J., & Baesens, B. (2011). An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. *Decision Support Systems, 51*(1), 141–154.
Joachims, T. (1998). *Text categorization with support vector machines: Learning with many relevant features* (cit. on pp. 19,104,108,123,132). Springer.
Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification, arXiv preprint arXiv:1607.01759
Junqué de Fortuny, E., Martens, D., & Provost, F. (2013). Predictive modeling with big data: Is bigger really better? *Big Data, 1*(4), 215–226.
Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., & Sayres, R. (2018). Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV), ICML 2018. arXiv:1711.11279
Kosinski, M., Stillwell, D., & Graepel, T. (2013). Private traits and attributes are predictable from digital records of human behavior. *National Academy of Sciences, 110*(15), 5802–5805.
Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: A review of classification and combining techniques. *Artificial Intelligence Review, 26,* 159–190.
Kulkarni, V., Kern, M. L., Stillwell, D., Kosinski, M., & Matz, S. (2018). Latent human traits in the language of social media: An open-vocabulary approach. *PLoS One*. https://doi.org/10.1371/journal.pone.0201703.
Lang, K. Newsweeder: Learning to filter netnews. In *Proceedings of the twelfth international conference on machine learning* (pp. 331–339
Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature, 401,* 788–791. https://doi.org/10.1038/44565.
Lee, D. D., & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems* (pp. 556–562).
Lee, K., Sood, A., & Craven, M. (2019). Understanding learned models by identifying important features at the right resolution. arXiv:1811.07279
Lessman, S., Baesens, B., Seow, H. V., & Thomas, L. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *EJOR*, *247*(1), 124–136. https://doi.org/10.1016/j.ejor.2015.05.030.
Lundberg, S. M., Lee, S. -I., & C. (2019). Consistent feature attribution for tree ensembles. arXiv:1706.06060
Martens, D., Baesens, B., Van Gestel, T., & Vanthienen, J. (2007). Comprehensible credit scoring models using rule extraction from support vector machines. *EJOR, 183,* 1466–1476.
Martens, D., Baesens, B. B., & Van Gestel, T. (2009). Decompositional rule extraction from support vector machines by active learning. *IEEE Transactions on Knowledge and Data Engineering, 21*(2), 178–191. https://doi.org/10.1109/TKDE.2008.131.
Martens, D., Huysmans, J., Setiono, R., Vanthienen, J., & Baesens, B. (2008). Rule extraction from support vector machines: An overview of issues and application in credit scoring. *Studies in Computational Intelligence (SCI), 80,* 33–63.
Martens, D., & Provost, F. (2014). Explaining data-driven document classifications. *MIS Quarterly, 38*(1), 73–99.
Martens, D., Provost, F., Clark, J., & Junqué de Fortuny, E. (2016). Mining massive fine-grained behavior data to improve predictive analytics. *MIS Quarterly, 40*(4), 869–888.
Matz, S. C., Appel, R., & Kosinski, M. (2020). Privacy in the age of psychological targeting. *Current Opinion in Psychology*, *31*, 116–121. https://doi.org/10.1016/j.copsyc.2019.08.010.
Matz, S. C., & Netzer, O. (2017). Using big data as a window into consumer psychology. *Current Opinion in Behavioral Science, 18,* 7–12.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Moeyersoms, J., d'Alessandro, B., Provost, F., & Martens, D. (2016). Explaining classification models built on high-dimensional sparse data. In *Workshop on human interpretability, machine learning: WHI 2016, June 23, 2016, New York, USA/Kim, Been [edit.]* (pp. 36–40).

Murdoch, W.J., Singh, C., Kumbier, K., Abbasi-Asl, R., & Yu, B. (2019). Interpretable machine learning: Definitions, methods, and applications. arXiv:1901.04592

O'Callaghan, D., Greene, D., Carthy, J., & Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. *Expert Systems with Applications, 42,* 5645–5657.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12,* 2825–2830.

Praet, S., Van Aelst, P., & Martens, D. (2018). I like, therefore I am Predictive modeling to gain insights in political preference in a multi-party system, Research paper, University of Antwerp, Faculty of Business and Economics (pp. 1–34).

Quinlan, J. R. (1993). C4.5 programs for machine learning. Morgan Kaufmann Publishers Inc.

Ramon, Y., Martens, D., Provost, F., & Evgeniou, T. (2020). A comparison of instance-level counterfactual explanation algorithms for behavioral and textual data: SEDC, LIME-C and SHAP-C, Adv Data Anal Classif. https://doi.org/10.1007/s11634-020-00418-3

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you? Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144).

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. arXiv:1811.10154

Shmueli, G. (2010). To explain or to predict? *Statistical Science, 25*(3), 289–310.

Sommer, E. (1995). An approach to quantifying the quality of induced theories. In C. Nedellec (Ed.), *Proceedings of the IJCAI workshop on machine learning and comprehensibility*.

Sushil, M., Suster, S., & Daelemans, W. (2018). Rule induction for global explanation of trained models. arXiv:1808.09744

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science, 12*(2), 257–285.

Tobback, E., & Martens, D. (2019). Retail credit scoring using fine-grained payment data. *Journal of the Royal Statistical Society*, *182*(4), 1227–1246. https://doi.org/10.1111/rssa.12469.

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience, 3*(1), 71–86.

Turney, P. (1995). Technical note: Bias and the quantification of stability. *Machine Learning, 20,* 23–33.

US Federal Trade Commission, Your Equal Credit Opportunity Rights, Consumer Information (2003).

Van Assche, A., & Blockeel, H. (2007). Seeing the forest through the trees: Learning a comprehensible model from an ensemble. In: Kok J.N., Koronacki J., Mantaras R.L.., Matwin S., Mladenič D., Skowron A. (Eds.), *Machine Learning: ECML 2007. ECML 2007. Lecture Notes in Computer Science, vol 4701*. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-74958-5_39

Vanhoeyveld, J., Martens, D., & Peeters, B. (2019). Customs fraud detection: Assessing the value of behavioural and high-cardinality data under the imbalanced learning issue. Pattern analysis and applications, ISSN 1433–7541 (pp. 1–21). Springer.

Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2012). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. *European Journal of Operational Research, 218*(1), 211–229.

Verbeke, W., Martens, D., Mues, C., & Baesens, B. (2011). Building comprehensible customer churn prediction models with advanced rule induction techniques. *Expert Systems with Applications, 38,* 2354–2364.

Wang, Y. X., & Zhang, Y. J. (2012). Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering, 25*(6), 1336–1353.

Wang, C., & Blei, D. M. (2011). *Collaborative topic modeling for recommending scientific articles*. Association for Computing Machinery. https://doi.org/10.1145/2020408.2020480

Wei, Y., Chang, M. C., Ting, T., Lim, S. N., & Lyu, S. (2018). Explain Black-box Image Classifications Using Superpixel-based Interpretation. In *IEEE, 24th international conference on pattern recognition (ICPR)*.

Wood, R. (1986). Task complexity: Definition of the construct. *Organizational Behavior and Human Decision Processes, 37,* 60–82.