Enhancing multilabel classification for food truck recommendation

Article *in* Expert Systems · July 2018

DOI: 10.1111/exsy.12304

Adriano Rivolli
Federal Technological University of Paraná, Cornélio Procópio, Brazil

Carlos Soares
University of Porto

Andre de Carvalho
University of São Paulo

ORIGINAL ARTICLE

Journal Section

Enhancing multi-label classification for food truck recommendation

Adriano Rivolli[1,2] | Carlos Soares[3] | André C. P. L. F. de Carvalho[2]

[1]Universidade Tecnológica Federal do Paraná, Cornélio Procópio, Brazil

[2]ICMC, Universidade de São Paulo, São Carlos, Brazil

[3]Faculdade de Engenharia da Universidade do Porto, Porto, Portugal

Correspondence
Adriano Rivolli, Universidade Tecnológica Federal do Paraná, Cornélio Procópio, Brazil
Email: rivolli@utfpr.edu.br

Funding information

Food trucks are a widely popular fast food restaurant alternative, whose differentiating factor is their proximity to customers. Their popularity have stimulated the expansion of available options, which now includes several different types of cuisines, consequently making the choice by customers a challenging issue. From data obtained via a market research, in which hundreds of participants provided their food truck preferences, this paper focuses on the problem of food truck recommendation using a multi-label approach. In particular, it investigates how to improve the recommendation task regarding a previous work, where some labels have never been predicted. In order to address this problem, different alternatives were investigated. One of these alternatives, the Ensemble of Single Label, proposed in this paper, was able to reduce it. Despite its simplicity, good predictive results were obtained when they were used in the investigated task. Among other benefits, all labels were correctly predicted at least for few instances.

KEYWORDS

Food truck recommendation, Multi-label classification, Recommender system, Multi-label dataset

1 | INTRODUCTION

In the last decade, food trucks became a phenomenon, with a rapid growth in popularity, in many parts of the world (Weber, 2012). Generally, they sell food that belongs to a wide range of cuisines, and move to different locations of a region, without having to settle in a specific place. In festivals, music concerts and large-scale events, it is common to have more than one food truck. The availability of a large number of cuisines can make the food choice a challenging task. Recommender systems have been successfully used in similar problems, where a decision maker has different possibilities and can be helped by receiving a recommendation of suitable options to his/her taste. Thus, recommender systems, like personalized event app and iterative totems, can be used to suggest food options, helping the customers to select one or more options among the available alternatives.

Motivated by the importance of this emerging business, a food truck recommendation task is investigated in this paper, using data obtained from a market research regarding food truck preferences, which describes socioeconomic profiles and customers' personal information like their habits and preferences. While related works (Pazzani and Billsus, 2007; Fu et al., 2014; Zhang et al., 2016) recommend specific places (restaurants), in this work, the food truck cuisines are recommended. Some business opportunities that can benefit from this study are: food truck market research, discovery of trends in food truck preferences, food truck advertising and food truck recommendation apps.

In a previous work, this task was addressed as a multi-label classification (MLC) problem (Rivolli et al., 2017), when each different type of food truck cuisine was mapped to a class label and more than one type could be recommended for a given input. Although, overall, good predictive results were obtained, the MLC strategies evaluated presented a poor predictive performance in the least common class labels. All the MLC strategies investigated presented this problem. In order to address this limitation and improve the predictive performance in the least common classes, two different approaches were investigated: *(i)* removal of irrelevant attributes using an attribute selection technique, and *(ii)* increasing the number of instances in the minority classes using an oversampling technique.

By failing to obtain the expected improvement, a new alternative, the *Ensemble of Single Label* (ESL), is proposed. It induces several multi-class models from datasets where each instance is randomly associated with only one of its labels. By modeling the least common labels during the transformation process, the authors believe it might lead the induced MLC model to predict the correct label for instances from the least common classes more frequently. It must be observed that ESL is an ensemble of the single label strategy (Boutell et al., 2004), one of the first studied transformations in MLC, but due to the fact that it eliminates labels (de Carvalho and Freitas, 2009) and discards a large amount of information relative to the problem (Tsoumakas and Katakis, 2007), it has been criticized and is not used often nowadays. However, when multiple models related to the same task are combined, none of the labels are effectively ignored and, consequently, a new strategy for MLC is obtained.

Intuitively, ESL can address problems with few and imbalanced labels, like those investigated in this paper. Experiments were performed comparing different MLC strategies and base algorithms. Even being much simpler than the other strategies, ESL when combined with feature selection and data oversampling techniques, was able to correctly predict labels that were not predicted by other strategies.

The main contributions from this paper in relation to the previous work (Rivolli et al., 2017) are: *(i)* incorporation of new approaches to improve the predictive performance of MLC strategies for food truck recommendation; *(ii)* identification and formalization of three label problems, which are ignored by the current evaluation measures; *(iii)* proposal and experimental evaluation of the ESL strategy, a simple MLC transformation strategy able to deal with the under-represented labels in MLC tasks

The rest of this paper is organized as follows: In Section 2, the concepts relative to the multi-label classification are presented and the related works are analyzed. Next, Section 3 briefly describes the food truck dataset used in this study.

Afterwards, Section 4 presents the ESL strategy. Section 5 presents the methodology used in the experiments and Section 6 presents and discusses the obtained results. In Section 7, the paper is concluded with a highlight of relevant aspects observed in the experimental results and directions for future work.

2 | BACKGROUND

This section begins with the definition of the main concepts related to MLC (Subsection 2.1) necessary to follow this paper and ends with a brief description of the related works (Subsection 2.2).

2.1 | Multi-label Classification

In MLC tasks, an instance can be simultaneously classified in more than one of the existing class labels. Binary and multi-class classification tasks can be seen as special cases of MLC tasks (de Carvalho and Freitas, 2009) where a single class is predicted. To formally define MLC, let $\mathcal{L} = \{\lambda_1, \lambda_2, ..., \lambda_q\}$ be the set of q labels λ_j related to a particular problem $\mathcal{X}$, where $\mathcal{X}$ is the instance space with d attributes. The learning process results in a hypothesis $h : \mathcal{X} \rightarrow 2^{\mathcal{L}}$ that associates new instances with a subset of labels contained in $\mathcal{L}$.

The available data $\mathcal{D} \in \mathcal{X}$, contains a set of labeled instances, such that $\mathcal{D} = \{(x_1, Y_1), ..., (x_n, Y_n)\}$. Every labeled instance is composed of $x_i = (x_{i1}, x_{i2}, ..., x_{id})$, and $Y_i \subseteq \mathcal{L}$. For convenience, the labels associated with the i^{th} instance, also called labelset, can be seen as a binary vector $y_i = (y_{i1}, y_{i2}, \cdots, y_{in}) \in \{0, 1\}^q$, where $y_{ij} = 1 \iff \lambda_j \in Y_i$ and $y_{ij} = 0 \iff \lambda_j \notin Y_i$. Different strategies have been proposed to obtain the predictive model h, that is used to predict, for an unseen instance $(x_i, ?)$, the set of relevant labels $\hat{Y}_i$ (or $\hat{y}_i$ as a binarized prediction).

According to Tsoumakas et al. (2010), MLC strategies can be organized in two approaches: transformation approach and algorithm adaptation approach. The former transforms the original multi-label data set into a set of single-label data sets, where conventional machine learning algorithms can be used. The latter modifies existing machine learning algorithms to intrinsically support the multi-labeled data. This study evaluates the predictive performance obtained by different strategies from both approaches when applied to the food truck recommendation tasks: Binary Relevance (BR) (Boutell et al., 2004); Calibrated Label Ranking (CLR) (Fürnkranz et al., 2008); Dependent Binary Relevance (DBR) (Montañés et al., 2014); Ensemble of Classifier Chains (ECC) (Read et al., 2011); Multi-Label k-Nearest Neighbor (ML-kNN) (Zhang and Zhou, 2007); and, RAndom k-LabELsets (RAkEL) (Tsoumakas et al., 2011). These strategies were selected due to their popularity as well as to cover the two MLC strategy approaches.

BR is the simplest and most commonly used multi-label strategy (Luaces et al., 2012). It uses the one-versus-all multi-class classification approach (de Carvalho and Freitas, 2009) to generate q binary datasets and induce a binary model for each dataset. The final MLC prediction is the combination of all binary predictions. The DBR strategy is based on stacking generalization (Wolpert, 1992), where the label values are used to increase the feature space, to model the labels' dependencies in the learning process. The ECC strategy is an ensemble of distinct Classifier Chains models. It organizes the binary classifiers in a chain and increment the input space with the results obtained by the previous classifiers in the chain. The CLR strategy uses a pairwise transformation where each pair of labels generates a binary dataset, similar to one-versus-one multi-class classification approach. The prediction is defined using a voting scheme. RAkEL is an ensemble of multi-class models. Each model is induced using a subset of labels, named labelset, and each labelset is mapped to a class. Exploring a different perspective, for a given instance, ML-kNN finds the k nearest neighbors in the training set and uses them to apply the maximum a posteriori (MAP) principle to predict the labels. Contrary to the rest, ML-kNN does not require a base algorithm, because it is an adaptation of the traditional k-Nearest

Neighbor (kNN).

2.2 | Related Works

To the best of the authors' knowledge, the use of MLC for food truck recommendation has not been explored in the literature, apart from a previous work from the authors (Rivolli et al., 2017). However, restaurant recommendation has been the subject of several studies, some of them recent (Fu et al., 2014; Zhang et al., 2015a; Sun et al., 2015; Zhang et al., 2016). This problem has been studied in the recommender systems scientific community as a specialization of the location prediction problem (Pazzani and Billsus, 2007).

Restaurant recommender systems usually base their recommendation on a broad range of attributes, including user's feedback (restaurant visit via check-ins, reviews and ratings), geolocation information, demographic data (age, gender, etc.), friends' preferences and restaurant features. In some cases, the prediction is a ranking of restaurants (Fu et al., 2014) or the top-k most relevant restaurants (Zhang et al., 2015a).

The main approaches used for restaurant recommender systems are designed to recommend specific places and their solutions are developed in a dynamic and ubiquitous environment. This work follows a different, more generic approach, addressing food truck recommendation through the recommendation of cuisine categories.

3 | DATA DESCRIPTION

The food truck dataset was created from the answers provided by 407 participants in a food truck preference survey. The participants either were approached in local fast-food festivals[1] or online, where they anonymously filled-out a questionnaire, in Portuguese, describing their personal information and preferences when it comes to their selection from a set of food truck cuisine possibilities[2]. This section describes the questionnaire used and analyzes the multi-label dataset, which is openly available in the Cometa (Charte et al., 2018) and Mulan (Tsoumakas et al., 2010) repositories.

3.1 | The survey

The form used for the survey had 15 objective questions about the habits and preferences related to food truck cuisines and 6 objective questions about users' profile. These 21 questions were considered as predictive attributes, as summarized in Table 1. Some attributes were inherently organized in categories, like `gender` and `marital.status`. Since the implementation of the machine-learning algorithms used in this study do not support ordinal attributes, the questions with answers that consist of ordinal values were converted to numeric values, like `educational.level` and `age.group`, to avoid loss of information. In the table, the texts of questions and options are reduced, due to space limitation. The types *num*, *categ* and *bin* are, respectively, abbreviations for numeric, categorical and binary.

The target attribute, a set of class labels, was associated with food preferences and multiple alternatives could be simultaneously assigned, making the target prediction a MLC task. The form provided 12 alternatives: 1 - `arabic_food`, 2 -`brazilian_food`, 3 - `chinese_food`, 4 - `street_food`, 5 - `fitness_food`, 6 - `gourmet`, 7 - `healthy_food`, 8 - `italian_food`, 9 - `japanese_food`, 10 - `mexican_food`, 11 - `snacks`, and 12 - `sweets_desserts`.

[1] In Natal, the capital of Rio Grande do Norte, which is located in the northeast of Brazil.

[2] The survey was conducted between November 15th and 20th, 2015. The Google Form tool was used to collect the responses.

TABLE 1 Summary of dataset attributes. The questions and options used in the survey and their respective values mapped in the dataset.

Attribute	Type	Question	Options
Questions related to the consumers' habits and preferences			
frequency	num	Often eating out	0 - rarely, 1 - monthly, 2 - weekly, 3 - twice a week, 4 - almost daily/daily
time	categ	Day period of preference	afternoon, dawn, dinner, happy hour, lunch
expenses	num	How much to spend	15 - to R$15,00, 20 - to R$20,00, 30 - to R$30,00, 40 - to R$40,00, 50 - without limit
motivation	categ	What is the motivation	ads, by chance, friend, social network, web
taste	num	Importance of food flavor	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
hygiene	num	Importance of hygiene	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
menu	num	Importance of menu diversity	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
presentation	num	Importance of food presentation	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
attendance	num	Importance of service quality	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
ingredients	num	Importance of ingredients quality	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
place.to.sit	num	Importance of a place to sit	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
takeaway	num	Importance of takeaway option	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
variation	num	Importance of varying the choices	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
stop.trucks	num	Importance of food truck meetings	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
schedule	num	Importance of a fixed schedule	1 - very low, 2 - low, 3 - medium, 4 - high, 5- very high
Questions related to the consumers' profile			
gender	bin	Gender	F - Female, M - Male
age.group	num	Age group	1 - <19, 2 - 20-25, 3 - 26-30, 4 - 31-35, 5 - 36-40, 6 - 41-45, 7 - 46-50, 8 - >50
educational.level	num	Scholarity	0 - no diploma, 1 - high school, 1.5 - in graduation, 2 - graduation, 3 - specialization, 4 - master's degree, 5 - doctorate (PhD)
average.income	num	Average income	1 - around R$ 1.000,00, 2 - around R$ 3.000,00, 3 - around R$ 5.000,00, 4 - around R$ 10.000,00, 5 - around R$ 20.000,00, 6 - more than R$ 20.000,00
has.work	bin	Has a job	0 - No, 1 - Yes
marital.status	categ	Marital status	divorced, married, single

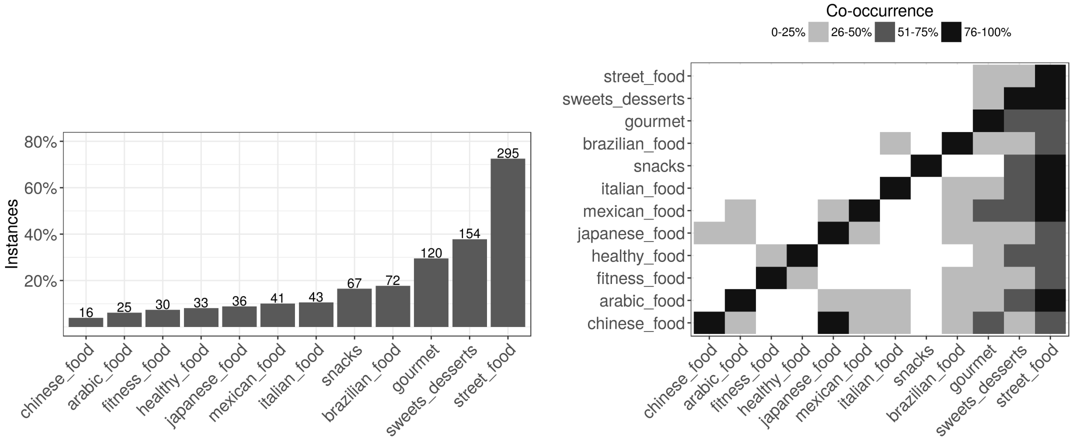
3.2 | Dataset Analysis

The 21 predictive attributes are composed of 16 numeric attributes, 3 categorical attributes and 2 binary attributes. The 12 labels are combined in 117 distinct ways (labelsets), where 74 of these combinations occur only once (single labelsets). Thereby, from the 407 survey-participants, 18% had an uncommon response regarding food-truck preferences. Given that the number of labelset is upper bounded by $min(n, 2^q) = min(407, 4096) = 407$, the labelset's diversity for the food truck dataset is $\approx$ 28%, which means, few labelsets were frequently observed while many of them were rarely observed. This characteristic may particularly affect the MLC strategies that explore label combinations (Tsoumakas et al., 2011).

Other characteristics observed in the food truck dataset were: label dependency, cardinality and density (Luaces et al., 2012). These characteristics describe, respectively, the average correlation among the labels, the average number of labels per instance and the average proportions of label frequencies. The label dependency is 0.13, which indicates that there is a low correlation between the labels. The cardinality and density are 2.28 and 0.19, respectively. Hence, on average, each instance is tagged with two labels and each label is related to 19% of the instances. From these observations, MLC strategies that explore label dependency can have their performance reduced by the low correlation among the food truck labels. Moreover, even though the number of labels per instance is low, their average occurrence (density) is high, when compared with other multi-label datasets, as can be observed in Luaces et al. (2012).

Figure 1a describes the frequency of each label in the dataset, which is the number of instances with each label. Most labels appear in more than 10% of the instances and the 3 most frequent labels appear, at least, in 30% of the instances. Particularly, the `street_food` appears in more than 70% of the instances, which is not very common in multi-label datasets[3]. Regarding the co-occurrences of the labels, shown in Figure 1b, it is possible to notice that, given their high frequency, the 4 most popular labels are strongly correlated. Other frequent pairs of labels were observed, such as:

`arabic_food` ⇔ `japanese_food`
`brazilian_food` ⇔ `italian_food`
`arabic_food` ⇔ `mexican_food`
`fitness_food` ⇔ `healthy_food`
`chinese_food` ⇔ `japanese_food`
`japanese_food` ⇔ `mexican_food`



(a) Labels frequency

(b) Labels co-occurrence

FIGURE 1 Frequency and co-occurrence of the labels in the food truck dataset. In Fig b, the rows indicate the presence of each label and the columns reflect the co-occurrence.

The food items in each pair are somewhat related, like spicy food (Arabic and Mexican), oriental cuisine (Chinese and Japanese), those with many pasta dishes (Italian and Brazilian) and those that are health-related (fitness and healthy). Finally, Figure 2 shows the Pearson correlation coefficient between each predictive attribute and each label. The most relevant attribute, with regards to correlation, is `average.income`, followed by `scholarity`, and both of which are considered part of an individual's personal information. `Gourmet` is the label mostly related to the attributes and `snacks` is the second, although it is inversely correlated with most of the attributes. All other pairs of labels and predictive attributes have some occurrences, however it was not possible to see a clear pattern.

[3] Usually, a label is associated with less than 50% of the instances.

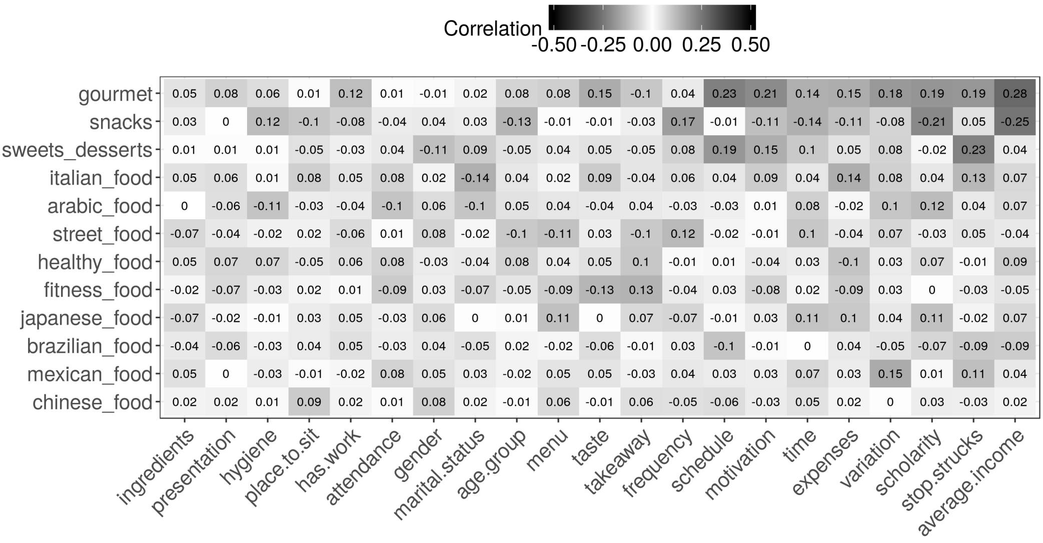


FIGURE 2 Correlation coefficient for each pair of predictive attributes (x-axis) and the labels (y-axis). The attributes are sorted from left to right by the absolute average of the correlations. Similarly, the labels are sorted from the bottom to the top.

4 | ENSEMBLE OF SINGLE LABELS

In (Rivolli et al., 2017), the investigated MLC strategies were not able to recommend the less frequent labels for the food truck dataset. In order to deal with this limitation, an *Ensemble of Single Label* (ESL) strategy, which uses a transformation approach that favors the choice of the less frequent labels at the expense of the most common is proposed here. ESL transforms the multi-label dataset into several multi-class datasets, by randomly selecting, for each training instance, a single label. Using a multi-class classification base learning algorithm, a model is induced from each transformed dataset. The multi-label prediction is obtained by combining the labels predicted by the multi-class models.

To illustrate the whole pipeline, Figure 3 presents the ESL training and prediction phases and Table 2 shows its transformation using a sample of the food truck dataset. Assuming that the instances in the sample are training instances, five multi-class models are created using a random label from each instance. The transformed datasets has different classes, as shown in the D_i columns of the table, which represent the class column in the multi-class datasets presented in Figure 3. When an instance is related to a single label, such as x_2, all five models are applied to it. When an instance has multiple labels, the less frequent labels are selected with higher probability. Even if, for a specific instance, a label is not selected for any model, like label 2 in x_1, it may be selected for other instances, as can be observed in x_2 and x_3. Since the multi-label prediction is the union of the multi-class models' prediction, in this example the models can predict the labels $M_1 = \{1, 2, 4, 9\}$, $M_2 = \{2, 8, 10, 11\}$, $M_3 = \{1, 2, 8, 9, 12\}$, $M_4 = \{2, 8, 9, 10\}$ and $M_5 = \{1, 2, 8, 10\}$, covering all the labels presented in the labels' column.

Algorithm 1 describes the training procedure used by the ESL strategy. As input, the strategy receives the training data ($\mathcal{D}$), the base algorithm (A), the number of internal models (m) and the weight given to the least frequent labels (ω). The output of the training phase is a set of multi-class models. The algorithm is simple, for each member of the ensemble,

TABLE 2 Example of food truck instances and ESL transformation process. The columns D_1 to D_5 indicate the label selected for each instance/model. Although the labels were randomly selected, the ELS favors the less frequent labels.

Person	Food truck preferences (labels)	D_1	D_2	D_3	D_4	D_5
x_1	2-brazilian_food, 8-italian_food, 9-japanese_food	9	8	9	9	8
x_2	2-brazilian_food	2	2	2	2	2
x_3	2-brazilian_food, 8-italian_food	2	8	8	8	8
x_4	1-arabic_food, 4-street_food, 9-japanese_food, 11-snacks	1	11	1	9	1
x_5	4-street_food, 10-mexican_food, 12-sweets_desserts	4	10	12	10	10

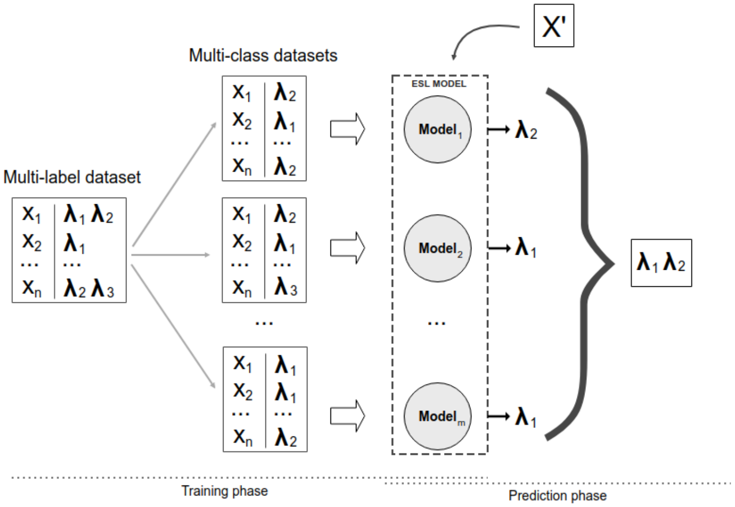


FIGURE 3 Illustration of the training and prediction phases in the ESL pipeline. A multi-class transformation is applied to the original multi-label dataset, generating *m* datasets, where *m* is a hyperparameter defined by the user. They are used to induce the multi-class models. For a given instance x' the multi-label prediction is the union of all labels predicted by the internal models.

the multi-label dataset is transformed (line 3), a model is induced from the new dataset (line 4) and then added to the list of models (line 5). Let *freq* and *random* be pre-defined functions that, respectively, return the frequency of a given label and a random number, both from an interval between 0.0 and 1.0. Equation 1 describes the transformation process and shows how the labels are selected. The ω hyperparameter is a weight value that gives more importance to the least frequent labels during the random selection. It must be observed that, when $\omega = 0$, the frequencies are ignored and the labels are selected by a completely random process. On the other hand, when $\omega = 1$ the inverse of the labels' frequency is used as the probability to select them. As ω increases, the least frequent labels have a higher chance to be selected.

$$\begin{aligned} \Phi(\mathcal{D},\omega) &= \left\{(x_i,\lambda^*) \mid 1 \le i \le n\right\}, \text{where} \\ \lambda^* &= \underset{\lambda_j \in Y_i}{\arg\max}\ (1 - freq(\lambda_j))\omega + random() \end{aligned} \tag{1}$$

The prediction procedure is described by Algorithm 2. The algorithm receives as input a set of multi-class models and the instance to be predicted. The algorithm's outputs are the set of labels predicted and a score associated with

Algorithm 1: ESL training

```
Input  : $\mathcal{D}$, A, $m$, $\omega$
Output : Models
Models $\leftarrow \varnothing$
for $i \leftarrow 1$ to $m$ do
    $D'_i \leftarrow \Phi(\mathcal{D}, \omega)$    // Equation 1
    $\theta_i \leftarrow$ Induces a model from $D'_i$ using A
    Models $\leftarrow$ Models $\cup \{\theta_i\}$
```

each label, showing the proportion of models that predicted it as relevant. In the lines 1 to 3, the variables are initialized; in 4 to 7, the multi-class models predict their relevant label and a vote for each predicted label is computed; finally, lines 8 and 9 compute the score associated with each label. It must be observed that 0 is the score for the irrelevant labels.

Algorithm 2: ESL prediction

```
Input  : Models, $x'$
Output : $\hat{Y}$, scores
1 $\hat{Y} \leftarrow \varnothing$
2 foreach $\lambda_j$ in $\mathcal{Y}$ do
3     votes$_{\lambda_j} \leftarrow 0$
4 foreach $\theta_i$ in Models do
5     lbl $\leftarrow \theta_i(x')$
6     $\hat{Y} \leftarrow \hat{Y} \cup \{$lbl$\}$
7     votes$_{\text{lbl}} \leftarrow$ votes$_{\text{lbl}} + 1$
8 foreach $\lambda_j$ in $\mathcal{Y}$ do
9     scores$_{\lambda_j} \leftarrow$ votes$_{\lambda_j}$ / |Models|
```

Although fairly simple, when a sufficient number of models are induced ($m \geq q$) any labelset can be predicted by the ESL strategy. In practice, the ESL is an extension of the single label strategy (Boutell et al., 2004) and what de Carvalho and Freitas (2009) named transformation by labels elimination. However, the use of multiple models mitigate the problems of loss of information due to the label elimination. As different labels are associated with the instances by each model, different decision boundaries are learned. Different from the strategies of the label powerset family (Read et al., 2008; Tsoumakas et al., 2010), that uses the labelset as classes, ESL directly uses a single label as class. Consequently, it has the drawback of not exploring the relationship of the labels, present in the powerset strategies. In addition, in scenarios with a large number of labels the ESL can generate datasets with many classes, which can make the learning process more difficult.

If suitable values were selected for m and ω, all labels will be represented in the training data. When the labels have similar frequencies, the probability of randomly selecting each label is also similar, regardless of the value of ω. In this scenario, given that $n \gg q$, any m larger than the label's cardinality is enough to have all labels represented in the training data. However, when labels occur with different frequencies, a suitable value of ω (at least 1) will favor the less common label, while the others with similar frequencies will be randomly selected for the rest of instances. Only two scenarios can avoid the suitable selection of labels and they are due to the selection of a very high ω value, which eliminates the randomization effect in the algorithm. The first is when an uncommon label only occurs together with another even more uncommon label, the former will never be selected. The second scenario is the opposite, since very common labels that never occur alone are discarded by another label.

5 | FOOD TRUCK RECOMMENDATION

This section describes the methodology adopted and the experiments carried out using the food truck dataset. It starts by explaining the MLC alternatives investigated and finishes with the experimental methodology, including procedures, evaluation measures, tools and setup configuration.

5.1 | MLC Approaches

In a previous work published by the authors (Rivolli et al., 2017), the food truck recommendation task was performed using MLC transformation strategies. Only a single base algorithm was evaluated for all strategies and no hyperparameter tuning was carried out. Despite getting better results than the adopted baseline, the strategies were not able to accurately predict the least common labels.

From this point, the question used as guideline for the investigation is: *How to predict the least common labels without harming the general predictive accuracy results?* Finding a good trade-off between the correct prediction of the least common labels and the most frequent labels leads to the investigation of distinct alternatives in the machine learning area. Besides the use of a different MLC strategy, a distinct base algorithm and the adoption of a tuning step to choose the hyperparameter values for the investigated strategies, other approaches, like attribute selection and data oversampling, were evaluated.

5.1.1 | Attribute selection

Even though the investigated problem has a relative small number of attributes, due to the low correlation of the attributes with the labels presented in Figure 2 the first hypothesis investigated consists of applying an attribute selection technique, to remove the irrelevant attributes.

As different attributes are correlated with different labels, a local solution, also called embedded (Pereira et al., 2018), for the transformation strategies is employed. Before inducing the internal models, the Relief algorithm (Robnik-Sikonja and Kononenko, 2003) is used to select the most relevant attributes for each label. Thus, no specific attribute selection technique for MLC is required. For instance, BR creates q binary datasets, one for each label, where all attributes are used indiscriminately. Using the Relief as a local solution, each dataset will have a distinct set of features. Figure 4 generalizes this scenario. Different numbers of attributes were selected as the most relevant, according to a specific hyperparameter defined as *attr.prop*, that indicates the proportion of attributes that can be selected.

5.1.2 | Data oversampling

MLC datasets usually present imbalanced class distributions. When classification algorithms are applied to imbalanced datasets, the induced models usually favor the majority class. There are several alternatives to reduce the class

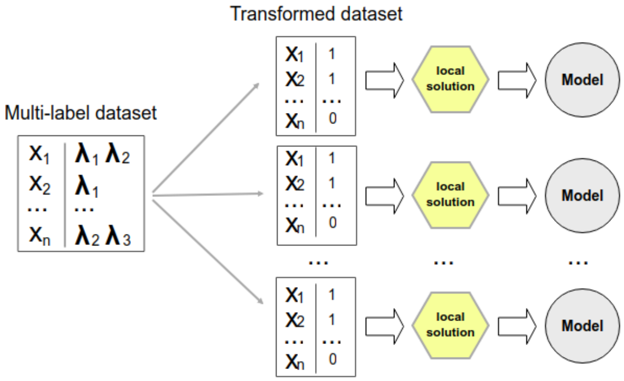


FIGURE 4 Example of the local solutions. After transforming the multi-label dataset and before inducting the learning model, a local solution is applied to the new data. Both Relief and SMOTE techniques were applied as a local solution, to select relevant attributes and create new instances to make the data less imbalanced, respectively.

imbalance. One of them increases the number of instances in the minority classes by adding artificial instances, known as data oversampling. Several techniques have been proposed for data oversampling. Among them, the SMOTE algorithm is one of the most popular (Bowyer et al., 2011). Some techniques were also proposed for MLC problems (Charte and Charte, 2015; Zhang et al., 2015b; Charte et al., 2017), to name a few, however, some of them are strategy-dependent while others favor a specific transformation approach. In this work, SMOTE is applied as a local solution, dispensing the use of an intrinsically multi-label oversampling technique.

Figure 4 also illustrates the use of SMOTE for the transformation strategies. In addition to the SMOTE hyperparameters, this work uses a hyperparameter to define when SMOTE should be applied to a dataset. In some situations, the dataset is not imbalanced and, consequently the use of SMOTE becomes unnecessary. For such, *min.freq* is used to define the minimum frequency of a class that characterizes a dataset as being imbalanced. As an example, if *min.freq = 0.1*, SMOTE is used only when the minority class is present in less than 10% of the training instances. For multi-class transformation, like the transformation used by RAkEL, where multiple classes are present, all classes with less than the respective minimum frequency are oversampled.

5.2 | Methodology

To assess the predictive performance of the strategies in the task of recommending food truck options, 10-folds cross-validation was used with paired and stratified folds, using the iterative stratification algorithm (Sechidis et al., 2011). It is a relaxed version of multi-class stratification, which obtains a similar distribution of labels in each fold, proportional to the distribution throughout the dataset. Hence, at least few instances related to the least frequent labels will be observed in the training and test sampling of each iteration. Six of the seven evaluated strategies require the use of a base algorithm. After investigating different possibilities, the Random Forest (RF) (Breiman, 2001) and the Support Vector Machines (SVM) (Amari and Wu, 1999) were employed due to their high predictive performance obtained in the experiments performed by the authors.

The grid search (Bergstra and Bengio, 2012) procedure was used for tuning the hyperparameters of the base algorithms and the adaptation strategies. Using 2-folds cross-validation in the training data, all combination of hyperparameter values were evaluated and those that obtained the best macro-F1 measure were selected. This measure was used because one of the goals is to obtain predictive models that are able to predict all labels correctly. Among the evaluated measures, macro-F1 indicates the averaged performance over all labels.

5.2.1 | Baseline

The MLC baseline General$_B$ (Metz et al., 2012) predicts the top most frequent labels based on the cardinality of the training data. For the dataset used in this work, only the 2 most frequent labels (`street_food` and `sweets_desserts`) were predicted as relevant. To also contemplate the ranking measures, the frequency of the labels was used to define the ranking order. Thus, the most frequent label is in the first position in the ranking and the least frequent is in the last. Equation 2 formalizes how to compute the ranking position for each label.

$$rank(\lambda_i) = |\mathcal{L}| - |\{\lambda_k \mid \lambda_k \in \mathcal{L},\ freq(\lambda_i) > freq(\lambda_k)\}| \tag{2}$$

5.2.2 | Evaluation measures

The evaluation of the predictive performance of MLC strategies requires the use of specific measures, that are able to explore the particularities of the MLC tasks (Tsoumakas et al., 2010). The measures can evaluate the quality of the labels predicted as relevant, called bipartition measures, or the quality of the ranking produced, the ranking measures. They can be instance or label-based, according to how they are computed. The label measures can be still organized in macro or micro averaged, where the former computes the measure for each label individually and then compute the mean of the values, and the latter creates a unique confusion matrix, using the results from all labels, which in turn is used to compute the measure (Zhang and Zhou, 2014). In this work, five measures that evaluate different predictive aspects were considered and are described next.

The F_1 *measure* (F1) is a bipartition instance based-measure used here as an index of accuracy. It computes the harmonic mean between precision and recall and does not take the true negative predictions into account, weighting the rate of relevant labels among those predicted and the rate of relevant labels that have been predicted over all relevant labels. The F1 is computed as

$$F1 = \frac{1}{n}\sum_{n=1}^{t} \frac{2|h(x_i) \cap Y_i|}{|h(x_i)| + |Y_i|}.$$

To complement the F1 measure, the *macro-based F1* (macro-F1) is a label based measure that computes the F1 individually for each label and uses the mean of these values. This measure gives equal weight to all labels and can be a suitable indicative that some labels have not been learned (Jackson and Moulinier, 2002), mainly when a very large difference in predictive performance is observed in relation to F1. To formally define this measure, let tp_j, tn_j, fp_j and fn_j be the true positive, true negative, false positive and false negative values of the confusion matrix for the label λ_j, respectively. The macro-F1 is computed as

$$macro\text{-}F1 = \frac{1}{q}\sum_{j=1}^{q} \frac{2tp_j}{2tp_j + fn_j + fp_j}.$$

The last bipartition measure considered is the *Subset Accuracy* (subset-accuracy), which is the strictest multi-label measure (Gibaja and Ventura, 2015). This measure only considers the proportion of fully correct instances, ignoring the partial hits. In other words, a partially correct prediction is the same as a completely incorrect one, since the labelset is considered the instance class (Zhang and Zhou, 2014). It can be computed as follows

$$\textit{subset-accuracy} = \frac{1}{n}\sum_{i=1}^{n} \mathcal{I}(h(x_i) = Y_i), \text{ where}$$

$$\mathcal{I}(\cdot) = \begin{cases} 1 & \text{if the predicate is true,} \\ 0 & \text{otherwise.} \end{cases}$$

Different from the previous measures, Ranking Loss (ranking-loss) considers the ranking of labels instead of the bipartition. It is an error measure that computes the averaged rate of label pairs that are inversely sorted, being a good indicator of the performance of the MLC strategy in predicting rankings. To formally define this measure, let $rank(x_i, \lambda_j)$ be the ranking position predicted for the label λ_j regarding the instance x_i, such that

$$ranking - loss = \frac{1}{n}\sum_{i=1}^{n} \frac{|\{(\lambda_j, \lambda_k)|rank(x_i, \lambda_j) > rank(x_i, \lambda_k), (\lambda_j, \lambda_k) \in Y_i \times \overline{Y}_i\}|}{|Y_i||\overline{Y}_i|},$$

$$\text{where } \overline{Y_i} = \mathcal{L} \setminus Y_i.$$

Another ranking and error measure is *One Error* (one-error). It indicates whether or not the most relevant label predicted should be really predicted, assessing the error of the most-relevant predicted label. This measure is computed as follows

$$one - error = \frac{1}{n}\sum_{i=1}^{n} \mathcal{I}(\arg\min_{\lambda_j \in \mathcal{L}} rank(x_i, \lambda_j) \notin Y_i)$$

5.2.3 | Label prediction problems

One problem with the previous measures is that when some of the labels are not correctly predicted for any given instance, the measures cannot identify these labels, unless the predictive performance of each label for each instance is individually identified.

From this scenario, this paper defines the inability of an MLC strategy to correctly predict one of the existing labels at least for one instance in the whole dataset, as the *Wrong Label Problem (WLP)*. A special case of this problem, named *Missing Label Problem (MLP)* occurs when the strategy never predicts a specific label. By contrast, when an MLC strategy predicts the same label for all instances, a problem named *Constant Label Problem (CLP)* is said to occur. Let TP_j, TN_j, FP_j and FN_j be, respectively, the true positive, true negative, false positive and false negative values of the confusion matrix for the label λ_j considering the whole dataset. Formally, WLP, MLP and CLP can be described by Equations 3, 4 and 5, respectively, which measure the proportion of labels presenting these problems. In an ideal scenario, zero is the expected value for them.

$$WLP = \frac{1}{q}\sum_{j=1}^{q} \mathcal{I}(TP_j == 0) \tag{3}$$

$$MLP = \frac{1}{q}\sum_{j=1}^{q} I(TP_j + FP_j == 0) \tag{4}$$

$$CLP = \frac{1}{q}\sum_{j=1}^{q} I(TN_j + FN_j == 0) \tag{5}$$

The use of these three measures can quantify and assess the occurrence or absence of the previously mentioned label problems. Some works related to label imbalanced (Charte et al., 2015, 2017) tried to increase the multi-label measures, especially the macro-F1. However, they did not investigate the occurrence of these problems. Consequently, there is no information about the ability of the respective approaches to solve them. Their formalization and investigation are contributions of this paper.

5.2.4 | Tools and setup

The experiments carried out for this paper used the R environment. Some of the strategies and algorithms require the hyperparameters to be defined by the user. In the beginning of this section, the methodology used to find the best set of hyperparameter values was detailed. Table 3 shows the set of values used in the grid search procedure for each strategy. As the MLC strategies require the use of a base algorithm, only the hyperparameters of these algorithms were tuned and the strategy default hyperparameter values defined in the original MLC papers were applied, reducing the complexity of the grid search.

TABLE 3 Values of the hyperparameters used in the experiments for the distinct algorithms. The multiple values of some hyperparameters were used in the grid search tuning.

Algorithm	Hyperparameters	R package
MLC strategies		
BR	-	`utiml`
CLR	-	`utiml`
DBR	-	`utiml`
ECC	`m=10`, `subsample=1`, `attr.space=1`, `vote.schema="maj"`	`utiml`
ESL10	`m=10`, `w=1`	developed
ESL30	`m=30`, `w=1.5`	developed
MLKNN	`k=[1,2,...,14, 15]`, `s=[0.1, 0.2, ..., 1.4, 1.5]`	`utiml`
RAkEL	`k=3`, `m=24`	`utiml`
Base algorithms		
RF	`ntree=[50, 100, ..., 450, 500]`	`randomForest`
SVM	`kernel="radial"`, gamma=[2^{-15}, 2^{-14}, ..., 2^{2}, 2^{3}], cost=[2^{-5}, 2^{-4}, ..., 2^{14}, 2^{15}]	`e1071`
Others		
Relief	`neighbours.count=5`, `sample.size=20`, `attr.prop=[0.3, 0.4, 0.5, 0.6, 0.70]`	`FSelector`
SMOTE	`k=[3,5,7]`, `perc.over=[150, 200, 300, 400]`, `min.freq=[0.05, 0.1, ..., 0.25, 0.3]`	`DMwR`

6 | RESULTS AND DISCUSSION

This section presents and analyzes the main results obtained from the experiments carried out in this study. Different perspectives are addressed covering the food truck recommendation task. The experiments also investigate the predictive performance obtained by the proposed ESL strategy to solve this task, and how they compare to the performance of existing strategies.

6.1 | MLC strategies and alternatives

Starting from the well known MLC strategies using the base algorithms RF and SVM, Table 4 shows the mean and standard deviation results for the previously mentioned evaluation measures and also their ranking position, when compared with the other strategies. The *AvgRank* column indicates the averaged ranking from all measures and offers an overview of the performance obtained by each strategy.

TABLE 4 Mean and standard deviation of performance measure values obtained by different evaluation measures for MLC strategies and baseline. The value inside parenthesis represents the position in the ranking. The AvgRank column is the averaged ranking for each strategy over all measures.

	F1 ↑		Macro-F1 ↑		Subset-accuracy ↑		One-error ↓		Ranking-loss ↓		
Strategy	Mean	Sd	Mean	Sd	Mean	Sd	Mean	Sd	Mean	Sd	AvgRank
Baseline	0.505 (10)	0.023	0.116 (9)	0.005	0.039 (12)	0.026	0.273 (7)	0.046	0.160 (8)	0.013	9.2
BR$_{RF}$	0.557 (2)	0.036	0.178 (3)	0.023	0.262 (6)	0.069	0.263 (3)	0.061	0.153 (3)	0.019	**3.4**
BR$_{SVM}$	0.528 (8)	0.037	0.122 (8)	0.024	0.269 (4)	0.053	0.272 (6)	0.072	0.156 (4)	0.016	6.0
CRL$_{RF}$	0.551 (3)	0.041	0.176 (4)	0.023	0.250 (9)	0.070	0.268 (4)	0.063	**0.140** (1)	0.016	4.2
CLR$_{SVM}$	0.507 (9)	0.016	0.090 (11)	0.016	**0.272** (1)	0.043	0.273 (7)	0.040	0.157 (6)	0.012	6.8
DBR$_{RF}$	0.532 (7)	0.037	0.150 (6)	0.018	**0.272** (1)	0.066	0.268 (4)	0.053	0.145 (2)	0.016	4.0
DBR$_{SVM}$	0.534 (6)	0.037	0.145 (7)	0.036	**0.272** (1)	0.049	0.273 (7)	0.065	0.156 (4)	0.014	5.0
ECC$_{RF}$	**0.574** (1)	0.043	**0.227** (1)	0.029	0.186 (10)	0.071	**0.260** (1)	0.059	0.203 (10)	0.026	4.6
ECC$_{SVM}$	0.540 (5)	0.047	0.221 (2)	0.024	0.159 (11)	0.060	0.292 (12)	0.081	0.195 (9)	0.027	7.8
ML-kNN	0.496 (12)	0.016	0.085 (12)	0.014	0.254 (8)	0.050	0.280 (11)	0.050	0.159 (7)	0.012	10.0
RAkEL$_{RF}$	0.542 (4)	0.044	0.169 (5)	0.030	0.257 (7)	0.076	0.261 (2)	0.049	0.210 (11)	0.037	5.8
RAkEL$_{SVM}$	0.502 (11)	0.050	0.099 (10)	0.021	0.267 (5)	0.071	0.273 (7)	0.055	0.264 (12)	0.026	9.0

In general, the predictive performance obtained by the MLC strategies were superior to the baseline, with few exceptions. In fact, only ML-kNN showed an averaged ranking worse than the baseline. Some transformation strategies using SVM did not outperform the baseline for the one-error measure. The ensembles ECC and RAkEL also did not overcome the baseline for the ranking-loss measure, which is explained by their use of a voting schema that produces poor rankings. Regarding the base algorithms, the strategies using RF had better ranking positions than those using SVM. The strategies with the best-averaged ranking position over all other measures used RF and when they were compared in pairs, those using RF also outperformed those using SVM. For this reason, only the strategies using RF will be considered in the next analysis.

Some patterns were also observed in the experimental results. The F1 and macro-F1 measures are highly correlated (Pearson correlation is around 0.9). Furthermore, the F1 values increase as the one-error values decrease (Pearson correlation is around -0.5), which in practice indicates that the strategies with a high F1 had a low one-error value, and vice-versa. Analyzing the rankings, strategies with a high (good) ranking positions for the macro-F1 obtained a low ranking position for the subset-accuracy and the strategies with a high ranking for the subset-accuracy also obtained a high ranking for ranking-loss.

The difference observed between the F1 and the macro-F1 indicates the inability of the strategies to correctly predict all labels. Table 5 shows the average binary F1 values obtained for each label by the different strategies, including the baseline. The value 0 indicates that the respective label was never correctly predicted. The ECC strategy predicted the largest number of labels and, consequently, has the best macro-F1. Most of the least frequent labels were not correctly predicted by any strategy, whereas other labels (brazilian_food and italian_food) were accurately predicted for a small number of instances and, consequently, obtained a low value for this measure.

TABLE 5 Averaged F1 obtained for each label by the different ML strategies. The labels are sorted by the frequency column. The macro-F1 of each strategy is the mean of such values.

Label	Frequency	Baseline	BR$_{RF}$	CLR$_{RF}$	DBR$_{RF}$	ECC$_{RF}$	ML-kNN	RAkEL$_{RF}$
chinese_food	0.04	0	0	0	0	0	0	0
arabic_food	0.06	0	0	0	0	0	0	0
fitness_food	0.07	0	0	0	0	0.050	0	0
healthy_food	0.08	0	0	0	0	0	0	0
japanese_food	0.09	0	0	0	0	0	0	0
mexican_food	0.10	0	0	0	0	0.073	0	0
italian_food	0.11	0	0.090	0.090	0.124	0.098	0	0.130
snacks	0.16	0	0.280	0.249	0.179	0.373	0.029	0.267
brazilian_food	0.18	0	0.075	0.095	0.075	0.197	0	0.075
gourmet	0.29	0	0.402	0.395	0.296	0.546	0.106	0.321
sweets_desserts	0.38	0.550	0.441	0.443	0.285	0.546	0.044	0.381
street_food	0.72	0.841	0.846	0.841	0.838	0.844	0.839	0.851

More specifically, Figure 5 shows the confusion matrix plot, where the performance for each label (x-axis) and strategy can be comparatively analyzed. The colors in each column indicate the amount of false negative (FN), false positive (FP), true negative (TN) and true positive (TP) values. An optimal predictive performance should have only two colors (TP and TN) and the black line shows where this division should be. The labels are sorted by their frequency and the strategies are arranged in alphabetical order. With the exception of ML-kNN, the strategies presented similar confusion matrices. However, none of the strategies was able to predict the presence of the labels arabic_food, chinese_food, fitness_food, healthy_food, japanese_food and mexican_food, even as false positive, characterizing the aforementioned labels' problems.

To assess the level of the previously defined labels problems in the MLC strategies, Table 6 shows the CLP, MLP and WLP values for each strategy. The baseline is also reported for comparative purposes. Apart from ECC, all other strategies had at least 50% of the labels with problems, usually MLP and WLP.

6.1.1 | Attribute selection and data oversampling

The authors assumed that the low predictive performance in the macro-F1 measure and the identified problems were caused by the class imbalance in the dataset and the low correlation between the predictive attributes and the

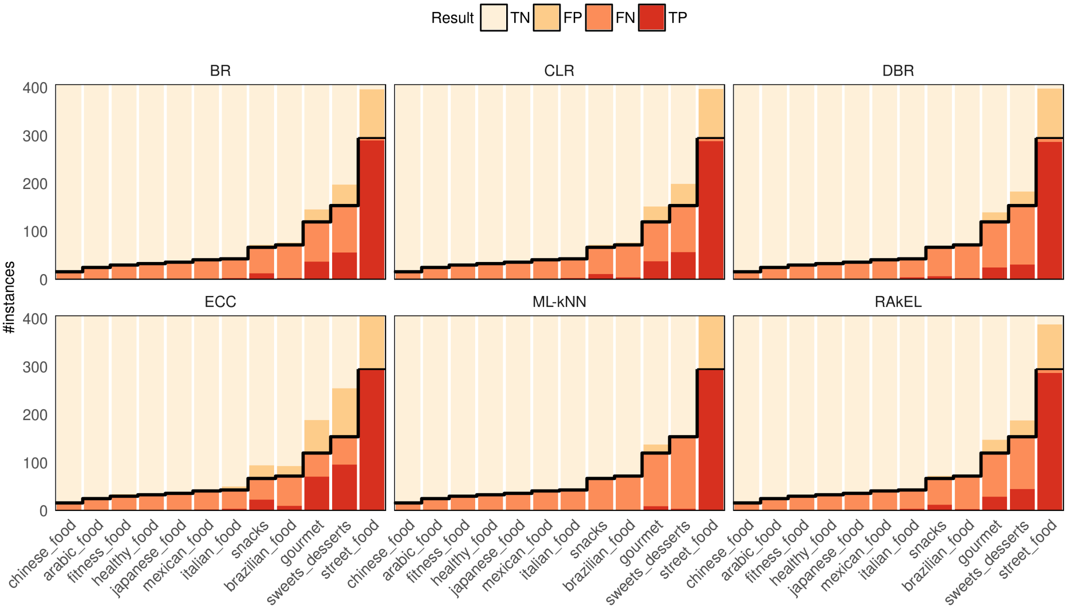


FIGURE 5 Confusion matrix of each label obtained for distinct strategies. The colors indicate the values False Negative (FN), False Positive (FP), True Negative (TN) and True Positive (TP). The line indicates the expected values to divide the TP from TN considering an optimal prediction. The RF is the base algorithm used for the transformation strategies.

TABLE 6 Label problems present in different MLC strategies for the food truck MLC dataset. Zero is the ideal result for each measure.

Problem	Baseline	BR_{RF}	CLR_{RF}	DBR_{RF}	ECC_{RF}	ML-kNN	RAkEL_{RF}
CLP	0.17	0	0	0	0	0	0
MLP	0.83	0.33	0.42	0.42	0.17	0.67	0.42
WLP	0.83	0.50	0.50	0.50	0.33	0.67	0.50

class labels. To deal with these problems, the Relief and SMOTE techniques were applied as local solution for the transformation strategies, as detailed in Section 5.1.

To facilitate a comparison with the results presented in Table 4, Figure 6 shows the difference obtained by using these pre-processing techniques. Both of them increased the macro-F1 measure for all strategies, as desired, at the cost of decreasing the values from the other measures, particularly, for subset-accuracy and one-error. It is worth noting, they subtly increased the ranking-loss results for the ensembles ECC and RAkEL. The highest macro-F1 improvement was obtained by SMOTE for DBR and RAkEL. However, even for these strategies, the absolute values were still low.

Regarding the label prediction problems, Table 7 shows how the use of Relief and SMOTE affected the MLC predictive performance. PCL was removed from this table because its value was zero for all strategies. Only ECC_{SMOTE} was able to eliminate one type of error, MLP. With few exceptions, the use of both techniques reduced the problems. In specific, ECC_{RELIEF} increased the WLP and RAkEL_{RELIEF} remained unchanged. DBR_{RELIEF} showed the largest improvement,

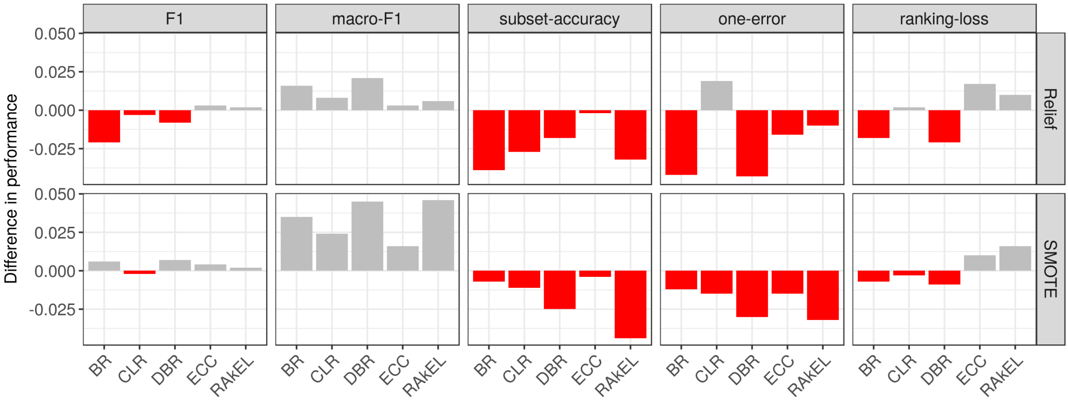


FIGURE 6 Results showing the effects of using pre-processing techniques. Relief and SMOTE were used, respectively for attribute selection and data oversampling. RF was used as the base algorithm for all strategies.

decreasing the MLP in 0.25 points.

TABLE 7 Results showing how the use of pre-processing techniques affected the prediction of labels with problems. In the table, Def, Rel and SMO are abbreviations for default (the strategy result), Relief and SMOTE, respectively.

	BR_{RF}			CLR_{RF}			DBR_{RF}			ECC_{RF}			$RAkEL_{RF}$		
Problem	Def	Rel	SMO	Def	Rel	SMO	Def	Rel	SMO	Def	Rel	SMO	Def	Rel	SMO
MLP	0.33	0.25	0.08	0.42	0.33	0.25	0.42	0.17	0.25	0.17	0.08	0	0.42	0.42	0.33
WLP	0.50	0.42	0.42	0.50	0.50	0.42	0.50	0.50	0.42	0.33	0.42	0.33	0.50	0.50	0.42

In summary, the solutions using SMOTE produced better results in comparison with the use of Relief. However, both techniques were able to improve the predictive performance in the investigated task and to minimize the label's prediction problems. Despite its inability to completely eliminate them, it is a good indicative that investing in similar techniques can mitigate the label prediction problems. Thus, the investigation of other MLC techniques for data oversampling (Charte et al., 2015, 2017) and attribute selection (Pereira et al., 2018) as solutions to the MLP and WLP problems are suggested for future work.

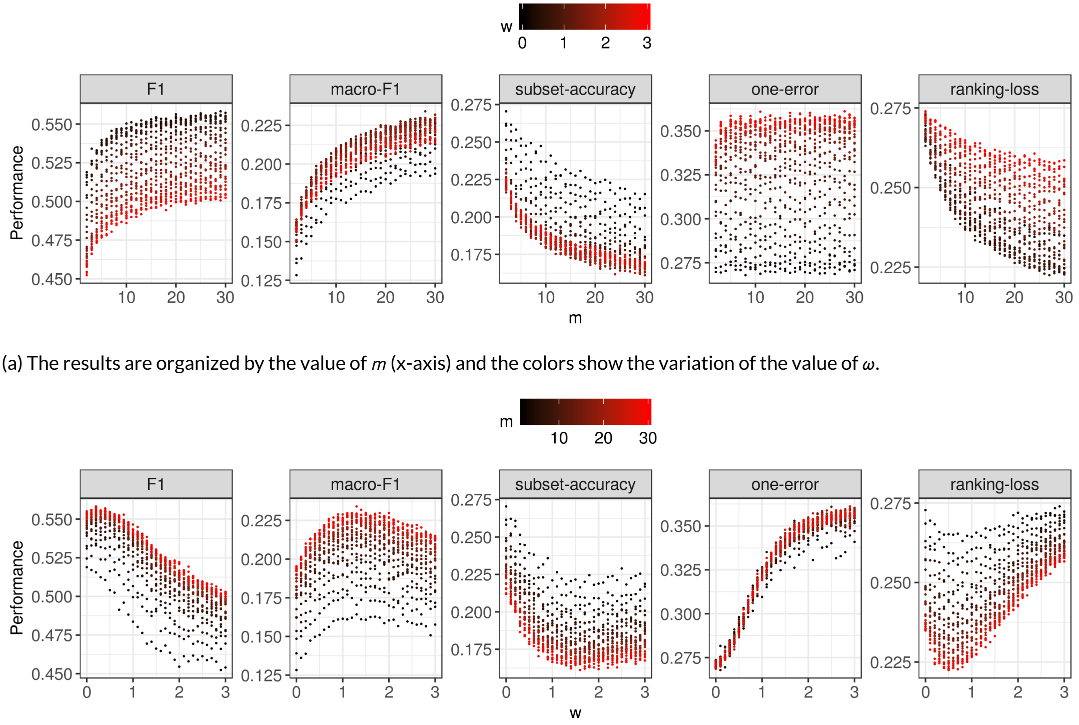
6.2 | ESL Results

The ESL is an MLC strategy designed to predict a more diversified set of labels, including the least common labels. Therefore, it is expected that its use can reduce the MLP and WLP problems presented in the solution of the food truck task by the other strategies. Initially, the overall behavior of the ELS strategy for the food truck dataset was investigated. Next, the predictive performance obtained by ESL was compared with those obtained by the other investigated approaches.

6.2.1 | Sensitivity of the hyperparameters

The MLC strategies have hyperparameters that affect their predictive performance. Using a grid search approach, different combinations of their values were evaluated for 30 distinct subsets of the food truck dataset, that was generated using different sampling approaches (Sechidis et al., 2011) and sizes. The following ranges of values were adopted for the hyperparameter: $m = [2, 3, ..., 29, 30]$ and $\omega = [0, 0.1, ..., 2.9, 3]$, such that, m defines the number of internal models used and ω the weight given by the less frequent labels in the random process of selecting them to create the transformed dataset.

Figure 7 shows the average predictive performance for different evaluation measures obtained for each combination of m and ω values. Figure 7a shows the variation due to changes in m, whereas Figure 7b shows the variation due to changes in ω. For each measure, a different configuration of these hyperparameters leads to the best predictive performance. The values for measures F1, macro-F1 and ranking-loss improved as m increased, while for the subset-accuracy, the opposite occurred. Regarding the ω hyperparameter, low to median values (between 0.5 to 2) produced the best predictive performance for all evaluation measures. For the one-error measure, the value of m did not have any clear effect. Only the ω value influenced the predictive performance.



(a) The results are organized by the value of m (x-axis) and the colors show the variation of the value of ω.

(b) The results are organized by the value of ω (x-axis) and the color shows the variation of the value of m.

FIGURE 7 ESL averaged result over 30 repetitions using different combinations of m and ω hyperparameters.

Looking specifically at the macro-F1 measure, the measure of interest, m affects the predictive performance slightly more than ω. For instance, the worst macro-F1 obtained by the best choice of m is higher than the worst value obtained by the best choice of ω. However, m is directly related to the computational complexity of the solution, since it increases the number of learning models. By contrast, ω impacts the quality of the measure and does not increase the solution's complexity. In practical terms, from a specific value of m, to choose a suitable ω is more important than to increase m. For this specific dataset, a ω value between 1 and 2 resulted in the best macro-F1 performance. For m, the same was seen when its value was higher than 25. As macro-F1 gives the same importance to all labels and the learning algorithms tend to generalize to the most frequent labels, these values of ω increase the probability of predicting the most frequent labels. A larger number of models also increases the chance of ESL predicting an infrequent label, since the prediction of the label by at least one of the individual prediction models is enough for ESL to predict the label.

6.2.2 | Comparative results

Using the same methodology presented in Section 5.2, two instances of the ESL using different hyperparameters values were evaluated, ESL10 and ELS30. The former demands less computational resources, using $m = 10$ and $\omega = 1$, whereas the latter is defined based on the analysis presented in the previous section, in which $m = 30$ and $\omega = 1.5$. Table 8 presents the results for both versions and includes the values obtained from the use of Relief and SMOTE.

TABLE 8 Mean and standard deviation of distinct measures obtained from the evaluation of the ESL strategy and the investigated alternatives. The number inside the parenthesis represents the position in the ranking. The AvgRank column is the averaged ranking for each item for all measures.

	F1 ↑		macro-F1 ↑		subset-accuracy ↑		one-error ↓		ranking-loss ↓		
Strategy	Mean	Sd	Mean	Sd	Mean	Sd	Mean	Sd	Mean	Sd	AvgRank
Baseline	0.505 (5)	0.023	0.116 (7)	0.005	0.039 (7)	0.026	**0.273** (1)	0.046	**0.160** (1)	0.013	4.2
ESL30$_{RF}$	0.554 (2)	0.047	0.239 (5)	0.037	0.181 (2)	0.058	0.310 (3)	0.072	0.221 (3)	0.028	**3.0**
ESL10$_{RF}$	**0.557** (1)	0.041	0.227 (6)	0.041	**0.201** (1)	0.050	0.298 (2)	0.064	0.222 (5)	0.031	**3.0**
ESL30$_{RELIEF}$	0.506 (4)	0.046	0.274 (2)	0.040	0.089 (6)	0.045	0.373 (5)	0.094	0.217 (2)	0.034	3.8
ESL10$_{RELIEF}$	0.513 (3)	0.042	0.257 (3)	0.046	0.109 (3)	0.067	0.340 (4)	0.067	0.225 (6)	0.026	3.8
ESL30$_{SMOTE}$	0.489 (6)	0.049	**0.280** (1)	0.043	0.094 (4)	0.040	0.430 (6)	0.117	0.221 (3)	0.032	4.0
ESL10$_{SMOTE}$	0.477 (7)	0.042	0.246 (4)	0.033	0.093 (5)	0.027	0.431 (7)	0.086	0.230 (7)	0.028	6.0

According to the experimental results, ESL was able to outperform the baseline in all situations for two measures (macro-F1 and subset-accuracy), partially in one (F1) and was outperformed in the two ranking measures (one-error and ranking-loss). With regards to the averaged ranking, only the ESL10$_{SMOTE}$ presented an overall performance that is worse than the baseline. As expected, the bias of the ESL tends to optimize the macro-F1 at the cost of having a small degradation in the predictive performance for the other analyzed measures.

In order to allow a comparison with the other strategies and approaches, Figure 8 shows, for each measure, the performance obtained by ESL10 against another similar strategy for each fold[4]. For instance, ESL10 against BR is a point in the plot, ESL10$_{SMOTE}$ against ECC$_{SMOTE}$ is another, and so on. With few exceptions, ESL obtained macro-F1 values higher than all other strategies. Despite that, the other strategies obtained better results than ESL in the other measures. Considering that ESL was designed to increase the macro-F1 measure, ESL achieved its goal for the dataset used. In fact, only ECC (including ECC$_{RELIEF}$ and ECC$_{SMOTE}$) was able to have a similar result. However, when comparing the number of models used, ESL demanded less computational resources[5].

[4] ESL30 obtained similar results, and for this reason, it was omitted.

[5] In this experiment, ECC induced 10 models for each label, resulting in 120 binary models, whereas, ESL10 and ESL30 induced only 10 and 30 multi-class

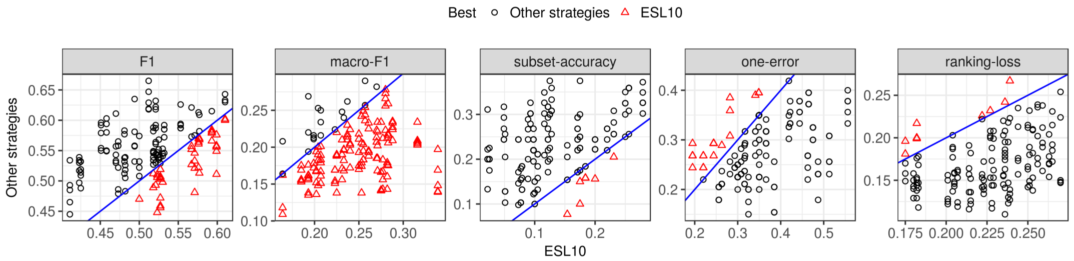


FIGURE 8 Comparison between the performance obtained by the ESL10 strategy (x-axis) and the performance obtained by the other strategies (y-axis). Each point shows the predictive performance of the ESL10 against another similar strategy for a specific fold. The line indicates the point where both strategies had a similar result. The color and shape of the point indicate which strategy obtained a better predictive performance according to the type of measure, which is also determined by the position of the point in relation to the line.

Even though the improvement in the macro-F1 is a good indicator that there was an improvement in the labels, it is not enough to guarantee that all of them were correctly predicted at least once and the label problems WLP and MLP were eliminated. In this sense, Table 9 presents such measures in comparison with the best results obtained for the other strategies. Only the strategies ESL10 and ESL30 were not able to avoid the WLP problem, and only the ESL30 was able to entirely eliminate the MLP. However, both had better or similar results than the other strategies using SMOTE. Moreover, the combination of ESL with Relief and SMOTE removed the labels' problems, with exception of ESL10_{SMOTE} that could not improve the WLP.

TABLE 9 Results showing how the ESL strategy affected the prediction of labels with problems. In the table, Def, Rel and SMO are abbreviations for default (the strategy result), Relief and SMOTE, respectively.

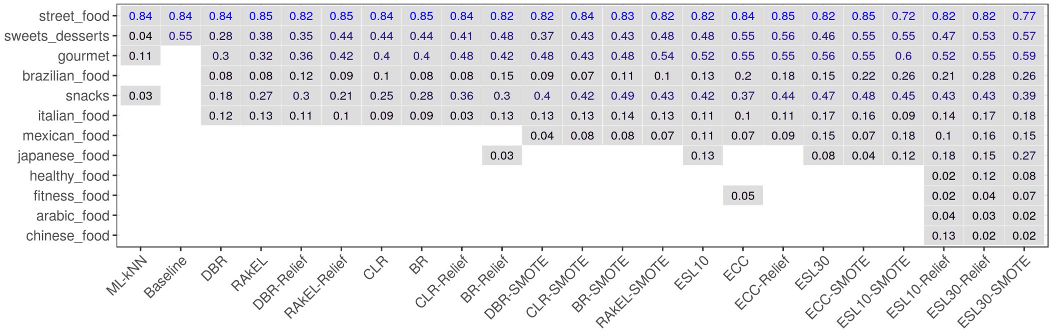
	ESL10			ESL30			BR_{RF}	CLR_{RF}	DBR_{RF}	ECC_{RF}	RAkEL_{RF}
Problem	Def	Rel	SMO	Def	Rel	SMO	SMO	SMO	SMO	SMO	SMO
MLP	0.08	0	0	0	0	0	0.08	0.25	0.25	0	0.33
WLP	0.33	0	0.33	0.33	0	0	0.42	0.42	0.42	0.33	0.42

Finally, Figure 9 shows the binary F1 result for all labels and strategies. The labels are sorted by their frequency in the dataset (from the bottom to the top) and the strategies by the value of the macro-F1 measure (from the left to the right). Three approaches were able to correctly predict instances for all labels (ESL10_{RELIEF}, ESL30_{RELIEF} and ESL30_{SMOTE}) without harming the F1 result of the most frequent labels, which can be seen as an improvement in the predictive performance for the food truck recommendation task.

6.3 | Statistical analysis

To complement the analysis of results, the Friedman's test was applied to the experimental results to assess whether the differences observed in the predictive performance of the investigated strategies are statistically significant. The Bonferroni-Dunn post-hoc test for pairwise comparisons was employed in the cases where the null hypothesis was rejected, at 95% confidence level. They are non-parametric statistical tests suitable to handle two or more dependent

models, respectively.



	ML-kNN	Baseline	DBR	RAkEL	DBR-Relief	RAkEL-Relief	CLR	BR	CLR-Relief	BR-Relief	DBR-SMOTE	CLR-SMOTE	BR-SMOTE	RAkEL-SMOTE	ESL10	ECC	ECC-Relief	ESL30	ECC-SMOTE	ESL10-SMOTE	ESL10-Relief	ESL30-Relief	ESL30-SMOTE
street_food	0.84	0.84	0.84	0.85	0.82	0.85	0.84	0.85	0.84	0.82	0.82	0.84	0.83	0.82	0.82	0.84	0.85	0.82	0.85	0.72	0.82	0.82	0.77
sweets_desserts	0.04	0.55	0.28	0.38	0.35	0.44	0.44	0.44	0.41	0.48	0.37	0.43	0.43	0.48	0.48	0.55	0.56	0.46	0.55	0.55	0.47	0.53	0.57
gourmet	0.11		0.3	0.32	0.36	0.42	0.4	0.4	0.48	0.42	0.48	0.43	0.48	0.54	0.52	0.55	0.55	0.56	0.55	0.6	0.52	0.55	0.59
brazilian_food			0.08	0.08	0.12	0.09	0.1	0.08	0.08	0.15	0.09	0.07	0.11	0.1	0.13	0.2	0.18	0.15	0.22	0.26	0.21	0.28	0.26
snacks	0.03		0.18	0.27	0.3	0.21	0.25	0.28	0.36	0.3	0.4	0.42	0.49	0.43	0.42	0.37	0.44	0.47	0.48	0.45	0.43	0.43	0.39
italian_food			0.12	0.13	0.11	0.1	0.09	0.09	0.03	0.13	0.13	0.13	0.14	0.13	0.11	0.1	0.11	0.17	0.16	0.09	0.14	0.17	0.18
mexican_food											0.04	0.08	0.08	0.07	0.11	0.07	0.09	0.15	0.07	0.18	0.1	0.16	0.15
japanese_food										0.03					0.13			0.08	0.04	0.12	0.18	0.15	0.27
healthy_food																					0.02	0.12	0.08
fitness_food																0.05					0.02	0.04	0.07
arabic_food																					0.04	0.03	0.02
chinese_food																					0.13	0.02	0.02

FIGURE 9 Comparison of the F1 measure for all labels (x-axis) and evaluated strategies (y-axis). The labels are sorted by frequency in the dataset (from the bottom to the top) and the strategies by the value of the macro-F1 measure (from the left to the right). The background's color indicates if the strategy was able to predict at least one label correctly.

samples (Sheskin, 2007).

The statistical tests were performed for two analyses: *(i)* to identify if the attribute selection (Relief) and data oversampling (SMOTE) approaches were able to improve the default strategy, three versions of each MLC strategy were statistically compared among themselves (default, with attribute selection and with oversampling). *(ii)* to find out if there are statistical differences in the predictive performance of different strategies. For this comparison, only the default version of each strategy was used. The tests were performed for all evaluation measures, including the label problems' measures, using 10-fold cross validation. Only the strategies BR, CLR, DBR, ECC, RAkEL and ESL using the RF as base algorithm were considered.

Table 10 presents the statistical differences obtained for the comparison of different approaches of the same strategy, with the p-value reported in parenthesis. For two measures, ranking-loss and CLP, no differences were observed, whereas for macro-F1 both DBR and RAkEL using SMOTE were superior to their default version. For all other measures, differences were observed only for the ESL strategy. For the bipartition and ranking measures, the default version outperformed the use of SMOTE and Relief indiscriminately. Regarding the label problems measures (PML and PWL), the use of Relief and SMOTE produced, with statistical significance, better results than the default. Finally, the comparison of ESL10 and ESL30 presented no differences, regardless of the evaluation measure used. In the next analysis, only ESL10 is employed, since it is simpler than ESL30.

In the comparison among the various strategies, no statistical differences were found for the measures F1, subset-accuracy, one-error and WLP. The statistical differences observed for the other measures are reported in Table 11. The strategies ECC and ESL were superior to the other strategies for macro-F1 and MLP, which shows that they predicted better results for all labels, on average. In contrast, ECC was worse than the other strategies, including ESL, for the CLP measure, which is the opposite of the MLP measure. In spite of ESL being simpler than ECC, it was the only one that could predict labels in a balanced way. As a drawback, ESL presented the lowest performance, together with ECC and RAkEL, for the ranking-loss measure, indicating that, for this measure, MLC ensembles are worse than non-ensemble strategies.

In summary, ESL addressed its specific purpose, predicting all food truck labels in a balanced way, without compromising the quality of the predictions. However, further studies, using similar scenarios, are necessary in order to confirm this finding.

TABLE 10 Statistical test results comparing different approaches for each strategy. The p-value is reported in parenthesis.

Measure	Statistical differences	
F1	ESL10$_{RF}$ $\prec$ ESL10$_{SMOTE}$ (0.0041), ESL30$_{SMOTE}$ (0.0371)	ESL30$_{RF}$ $\prec$ ESL10$_{SMOTE}$(0.0042), ESL30$_{SMOTE}$ (0.0378)
macro-F1	DBR$_{SMOTE}$ $\prec$ DBR$_{RF}$ (0.0018)	RAkEL$_{SMOTE}$ $\prec$ RAkEL$_{RELIEF}$ (0.0245), RAkEL$_{RF}$ (0.0064)
subset-accuracy	ESL10$_{RF}$ $\prec$ ESL10$_{RELIEF}$ (0.0183), ESL10$_{SMOTE}$ (0.0034), ESL30$_{RELIEF}$ (0.0017), ESL30$_{SMOTE}$ (0.0032)	
	ESL30$_{RF}$ $\prec$ ESL10$_{SMOTE}$ (0.0273), ESL30$_{RELIEF}$ (0.0151), ESL30$_{SMOTE}$ (0.0256)	
one-error	ESL10$_{RF}$ $\prec$ ESL10$_{SMOTE}$ (0.007), ESL30$_{SMOTE}$ (0.0251)	ESL30$_{RF}$ $\prec$ ESL10$_{SMOTE}$ (0.0168)
MLP	ESL10$_{RELIEF}$ $\prec$ ESL10$_{RF}$ (0.0009)	ESL30$_{SMOTE}$ $\prec$ ESL10$_{RF}$ (0.0002), ESL30$_{RF}$ (0.0174)
	ESL10$_{SMOTE}$ $\prec$ ESL10$_{RF}$ (0.0308)	ESL30$_{RELIEF}$ $\prec$ ESL10$_{RF}$ (0.0001), ESL30$_{RF}$ (0.0091)
WLP	ESL30$_{SMOTE}$ $\prec$ ESL10$_{RF}$ (0.0037)	ESL30$_{RELIEF}$ $\prec$ ESL10$_{RF}$ (0.0089)

TABLE 11 Statistical results obtained in the comparison among the strategies. The p-value is reported in parenthesis.

Measure	Statistical differences	
macro-F1	ECC$_{RF}$ $\prec$ BR$_{RF}$ (0.046), CLR$_{RF}$ (0.0342), DBR$_{RF}$ (0), RAkEL$_{RF}$ (0.0112)	
	ESL10$_{RF}$ $\prec$ DBR$_{RF}$ (0.0001), RAkEL$_{RF}$ (0.0279)	
ranking-loss	BR$_{RF}$ $\prec$ ECC$_{RF}$ (0.0316), ESL10$_{RF}$ (0.002) , RAkEL$_{RF}$ (0.0232)	
	CLR$_{RF}$ $\prec$ ECC$_{RF}$ (0.0015), ESL10$_{RF}$ (0.0001), RAkEL$_{RF}$ (0.001)	
	DBR$_{RF}$ $\prec$ ECC$_{RF}$ (0.0058), ESL10$_{RF}$ (0.0003), RAkEL$_{RF}$ (0.0041)	
CLP	CLR$_{RF}$ $\prec$ ECC$_{RF}$ (0.0197)	ESL10$_{RF}$ $\prec$ ECC$_{RF}$ (0.0034)
	DBR$_{RF}$ $\prec$ ECC$_{RF}$ (0.0197)	RAkEL$_{RF}$ $\prec$ ECC$_{RF}$ (0.0005)
MLP	ECC$_{RF}$ $\prec$ CLR$_{RF}$ (0.0425), DBR$_{RF}$ (0.0299), RAkEL$_{RF}$ (0.0354)	
	ESL10$_{RF}$ $\prec$ BR$_{RF}$ (0.0167), CLR$_{RF}$ (0.0023), DBR$_{RF}$ (0.0015), RAkEL$_{RF}$ (0.0018)	

7 | CONCLUSIONS

This paper investigated the task of food truck recommendation using MLC, where different types of cuisines were used as class labels and the predictive attributes were the personal information and preferences of individuals. Popular MLC strategies were evaluated as solutions to this task and they obtained a similar predictive performance. Although the obtained results were superior to a baseline, most of the strategies were not able to correctly predict the least common labels. Regarding this scenario, three label prediction problems were formalized and the two observed in the results were investigated, the wrong label (WLP) and, its specialization, the missed label problem (MLP).

Two frequent situations that affect the predictive performance of machine learning algorithms, and therefore machine learning based solutions to MLC are irrelevant attributes and class imbalance. The respective use of the Relief and SMOTE pre-processing techniques to remove irrelevant attributes and oversample the instances related to the least frequent labels were investigated. Despite observed improvements, these techniques were not able to satisfactorily address the label prediction problems. Thus, a new MLC transformation strategy, named Ensemble of Single Label (ESL), was proposed as a new alternative. The empirical results showed that its use was able to significantly reduce the WLP and MLP problems, while remaining competitive with the other evaluation measures.

Concerning the food truck recommendation task, none of the strategies was able to induce predictive models with high predictive performance. The authors believe that this difficulty was due to the distribution of values in the dataset, with a small number of instances for some labels and subjective class labels assigned by those participating in the survey. Even the strategies that are capable of incorporating the labels' dependencies in the learning process (DBR, ECC, LIFT and RAkEL) and the other investigated alternatives were not able to recommend food trucks with high predictive accuracy. These results also suggest that this MLC task is not easy to model with the currently used predictive attributes.

Future studies will further explore the labels' dependencies (Loza Mencía and Janssen, 2016; Papagiannopoulou et al., 2015), particularly, by using Bayesian Networks to model it beforehand (Wang et al., 2014) and also through specific recommender algorithms like those based on collaborative filtering (Adomavicius and Tuzhilin, 2005). Additionally, the authors plan to further investigate the prediction of the labels' problems and assess the proposed ESL strategy in other MLC domains.

According to the experimental results, the proposed strategy, ESL, can mitigate the problem in which the less frequent labels are either never or rarely predicted. Thus, ESL is a good alternative when this characteristic is observed, despite of the labels having similar importance in the business perspective, such as in the food truck recommendation. For instance, an interactive totem that suggests food options to customers in a food truck festival should not favor a specific type of cuisine, but, on the other hand, it should make relevant suggestions for demanding customers. Even though this issue is not new to the recommender system community, it is still not addressed by multi-label classification strategies.

ACKNOWLEDGEMENTS

This work is supported **by Intel** and by the Brazilian funding agencies CAPES, CNPq and FAPESP. It was used the computational resources of CeMEAI-FAPESP, Proc. 13/07375-0.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest and that the research was done in accordance with the policies of the journal.

REFERENCES

Adomavicius, G. and Tuzhilin, A. (2005) Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, **17**, 734–749. URL: https://doi.org/10.1109/TKDE.2005.99.

Amari, S. and Wu, S. (1999) Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, **12**, 783–789. URL: https://doi.org/10.1016/S0893-6080(99)00032-5.

Bergstra, J. and Bengio, Y. (2012) Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, **13**, 281–305. URL: http://dl.acm.org/citation.cfm?id=2188395.

Boutell, M. R., Luo, J., Shen, X. and Brown, C. M. (2004) Learning multi-label scene classification. *Pattern Recognition*, **37**, 1757–1771. URL: https://doi.org/10.1016/j.patcog.2004.03.009.

Bowyer, K. W., Chawla, N. V., Hall, L. O. and Kegelmeyer, W. P. (2011) SMOTE: synthetic minority over-sampling technique. *CoRR*, **abs/1106.1813**. URL: http://arxiv.org/abs/1106.1813.

Breiman, L. (2001) Random forests. *Machine Learning*, **45**, 5–32. URL: https://doi.org/10.1023/A:1010933404324.

de Carvalho, A. C. P. L. F. and Freitas, A. A. (2009) A tutorial on multi-label classification techniques. In *Foundations of Computational Intelligence - Volume 5: Function Approximation and Classification* (eds. A. Abraham, A. E. Hassanien and V. Snásel), vol. 205 of *Studies in Computational Intelligence*, 177–195. Springer. URL: https://doi.org/10.1007/978-3-642-01536-6_8.

Charte, F. and Charte, D. (2015) Working with multilabel datasets in R: The mldr package. *The R Journal*, **7**, 149–162. URL: https://journal.r-project.org/archive/2015-2/charte-charte.pdf.

Charte, F., Rivera, A. J., Charte, D., del Jesus, M. J. and Herrera, F. (2018) Tips, guidelines and tools for managing multi-label datasets: The mldr.datasets r package and the cometa data repository. *Neurocomputing*.

Charte, F., Rivera, A. J., del Jesús, M. J. and Herrera, F. (2015) MLSMOTE: approaching imbalanced multilabel learning through synthetic instance generation. *Knowl.-Based Syst.*, **89**, 385–397. URL: https://doi.org/10.1016/j.knosys.2015.07.019.

Charte, F., Rivera, A. J., del Jesus, M. J. and Herrera, F. (2017) Remedial-hwr: Tackling multilabel imbalance through label decoupling and data resampling hybridization. *Neurocomputing*. URL: http://www.sciencedirect.com/science/article/pii/S0925231217315187.

Fu, Y., Liu, B., Ge, Y., Yao, Z. and Xiong, H. (2014) User preference learning with multiple information fusion for restaurant recommendation. In *Proceedings of the 2014 SIAM International Conference on Data Mining, Philadelphia, Pennsylvania, USA, April 24-26, 2014* (eds. M. J. Zaki, Z. Obradovic, P. Tan, A. Banerjee, C. Kamath and S. Parthasarathy), 470–478. SIAM. URL: https://doi.org/10.1137/1.9781611973440.54.

Fürnkranz, J., Hüllermeier, E., Loza Mencía, E. and Brinker, K. (2008) Multilabel classification via calibrated label ranking. *Machine Learning*, **73**, 133–153. URL: https://doi.org/10.1007/s10994-008-5064-8.

Gibaja, E. and Ventura, S. (2015) A tutorial on multilabel learning. *ACM Comput. Surv.*, **47**, 52:1–52:38. URL: http://doi.acm.org/10.1145/2716262.

Jackson, P. and Moulinier, I. (2002) *Natural Language Processing for Online Applications: Text Retrieval, Extraction & Categorization*. John Benjamins.

Loza Mencía, E. and Janssen, F. (2016) Learning rules for multi-label classification: a stacking and a separate-and-conquer approach. *Machine Learning*, **105**, 77–126. URL: https://doi.org/10.1007/s10994-016-5552-1.

Luaces, O., Díez, J., Barranquero, J., del Coz, J. J. and Bahamonde, A. (2012) Binary relevance efficacy for multilabel classification. *Progress in AI*, **1**, 303–313. URL: https://doi.org/10.1007/s13748-012-0030-x.

Metz, J., de Abreu, L. F. D., Cherman, E. A. and Monard, M. C. (2012) On the estimation of predictive evaluation measure baselines for multi-label learning. In *Advances in Artificial Intelligence - IBERAMIA 2012 - 13th Ibero-American Conference on AI, Cartagena de Indias, Colombia, November 13-16, 2012. Proceedings* (eds. J. Pavón, N. D. Duque-Méndez and R. Fuentes-Fernández), vol. 7637 of *Lecture Notes in Computer Science*, 189–198. Springer. URL: https://doi.org/10.1007/978-3-642-34654-5_20.

Montañés, E., Senge, R., Barranquero, J., Quevedo, J. R., del Coz, J. J. and Hüllermeier, E. (2014) Dependent binary relevance models for multi-label classification. *Pattern Recognition*, **47**, 1494–1508. URL: https://doi.org/10.1016/j.patcog.2013.09.029.

Papagiannopoulou, C., Tsoumakas, G. and Tsamardinos, I. (2015) Discovering and exploiting deterministic label relationships in multi-label learning. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015* (eds. L. Cao, C. Zhang, T. Joachims, G. I. Webb, D. D. Margineantu and G. Williams), 915–924. ACM. URL: http://doi.acm.org/10.1145/2783258.2783302.

Pazzani, M. J. and Billsus, D. (2007) Content-based recommendation systems. In *The Adaptive Web, Methods and Strategies of Web Personalization* (eds. P. Brusilovsky, A. Kobsa and W. Nejdl), vol. 4321 of *Lecture Notes in Computer Science*, 325–341. Springer. URL: https://doi.org/10.1007/978-3-540-72079-9_10.

Pereira, R. B., Plastino, A., Zadrozny, B. and Merschmann, L. H. C. (2018) Categorizing feature selection methods for multi-label classification. *Artif. Intell. Rev.*, **49**, 57–78. URL: `https://doi.org/10.1007/s10462-016-9516-4`.

Read, J., Pfahringer, B. and Holmes, G. (2008) Multi-label classification using ensembles of pruned sets. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy*, 995–1000. IEEE Computer Society. URL: `https://doi.org/10.1109/ICDM.2008.74`.

Read, J., Pfahringer, B., Holmes, G. and Frank, E. (2011) Classifier chains for multi-label classification. *Machine Learning*, **85**, 333–359. URL: `https://doi.org/10.1007/s10994-011-5256-5`.

Rivolli, A., Parker, L. C. and de Carvalho, A. C. P. L. F. (2017) Food truck recommendation using multi-label classification. In *Progress in Artificial Intelligence - 18th EPIA Conference on Artificial Intelligence, EPIA 2017, Porto, Portugal, September 5-8, 2017, Proceedings* (eds. E. C. Oliveira, J. Gama, Z. A. Vale and H. L. Cardoso), vol. 10423 of *Lecture Notes in Computer Science*, 585–596. Springer. URL: `https://doi.org/10.1007/978-3-319-65340-2_48`.

Robnik-Sikonja, M. and Kononenko, I. (2003) Theoretical and empirical analysis of relieff and rrelieff. *Machine Learning*, **53**, 23–69. URL: `https://doi.org/10.1023/A:1025667309714`.

Sechidis, K., Tsoumakas, G. and Vlahavas, I. P. (2011) On the stratification of multi-label data. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part III* (eds. D. Gunopulos, T. Hofmann, D. Malerba and M. Vazirgiannis), vol. 6913 of *Lecture Notes in Computer Science*, 145–158. Springer. URL: `https://doi.org/10.1007/978-3-642-23808-6_10`.

Sheskin, D. J. (2007) *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, 4 edn.

Sun, J., Xiong, Y., Zhu, Y., Liu, J., Guan, C. and Xiong, H. (2015) Multi-source information fusion for personalized restaurant recommendation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015* (eds. R. A. Baeza-Yates, M. Lalmas, A. Moffat and B. A. Ribeiro-Neto), 983–986. ACM. URL: `http://doi.acm.org/10.1145/2766462.2767818`.

Tsoumakas, G. and Katakis, I. (2007) Multi-label classification: An overview. *IJDWM*, **3**, 1–13. URL: `https://doi.org/10.4018/jdwm.2007070101`.

Tsoumakas, G., Katakis, I. and Vlahavas, I. P. (2010) Mining multi-label data. In *Data Mining and Knowledge Discovery Handbook, 2nd ed.* (eds. O. Maimon and L. Rokach), 667–685. Springer. URL: `https://doi.org/10.1007/978-0-387-09823-4_34`.

— (2011) Random k-labelsets for multilabel classification. *IEEE Trans. Knowl. Data Eng.*, **23**, 1079–1089. URL: `https://doi.org/10.1109/TKDE.2010.164`.

Wang, S., Wang, J., Wang, Z. and Ji, Q. (2014) Enhancing multi-label classification by modeling dependencies among labels. *Pattern Recognition*, **47**, 3405–3413. URL: `https://doi.org/10.1016/j.patcog.2014.04.009`.

Weber, D. (2012) *The Food Truck Handbook: Start, Grow, and Succeed in the Mobile Food Business*. John Wiley & Sons.

Wolpert, D. H. (1992) Stacked generalization. *Neural Networks*, **5**, 241–259. URL: `https://doi.org/10.1016/S0893-6080(05)80023-1`.

Zhang, F., Yuan, N. J., Zheng, K., Lian, D., Xie, X. and Rui, Y. (2016) Exploiting dining preference for restaurant recommendation. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016* (eds. J. Bourdeau, J. Hendler, R. Nkambou, I. Horrocks and B. Y. Zhao), 725–735. ACM. URL: `http://doi.acm.org/10.1145/2872427.2882995`.

Zhang, F., Zheng, K., Yuan, N. J., Xie, X., Chen, E. and Zhou, X. (2015a) A novelty-seeking based dining recommender system. In *Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015* (eds. A. Gangemi, S. Leonardi and A. Panconesi), 1362–1372. ACM. URL: `http://doi.acm.org/10.1145/2736277.2741095`.

Zhang, M., Li, Y. and Liu, X. (2015b) Towards class-imbalance aware multi-label learning. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015* (eds. Q. Yang and M. Wooldridge), 4041–4047. AAAI Press. URL: http://ijcai.org/Abstract/15/567.

Zhang, M. and Zhou, Z. (2007) ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, **40**, 2038–2048. URL: https://doi.org/10.1016/j.patcog.2006.12.019.

— (2014) A review on multi-label learning algorithms. *IEEE Trans. Knowl. Data Eng.*, **26**, 1819–1837. URL: https://doi.org/10.1109/TKDE.2013.39.