Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2021) 29: 1797 – 1820

doi:10.3906/elk-2008-148

A new approach: semisupervised ordinal classification

Ferda ÜNAL[1], **Derya BİRANT**[2,*], **Özlem ŞEKER**[1]
[1]The Graduate School of Natural and Applied Sciences, Dokuz Eylül University, İzmir, Turkey
[2]Department of Computer Engineering, Faculty of Engineering, Dokuz Eylül University, İzmir, Turkey

Received: 28.08.2020 • **Accepted/Published Online:** 08.12.2020 • **Final Version:** 31.05.2021

Abstract: Semisupervised learning is a type of machine learning technique that constructs a classifier by learning from a small collection of labeled samples and a large collection of unlabeled ones. Although some progress has been made in this research area, the existing semisupervised methods provide a nominal classification task. However, semisupervised learning for ordinal classification is yet to be explored. To bridge the gap, this study combines two concepts "semisupervised learning" and "ordinal classification" for the categorical class labels for the first time and introduces a new concept of "semisupervised ordinal classification". This paper proposes a new algorithm for semisupervised learning that takes into account the relationships between the class labels, especially class orderings such as low, medium, and high. We also performed an extensive empirical study that involves 10 benchmark ordinal datasets with different quantities of labeled samples varying from 15% to 50% with an increment of 5%, aiming to evaluate the performance of our method by combining different base learners. The experimental results were also validated with a nonparametric statistical test. The experiments show that the proposed method improves the classification accuracy of the model compared to the existing semisupervised method on ordinal data.

Key words: Semisupervised learning, ordinal classification, machine learning, classification

1. Introduction

Machine learning builds a model to be able to predict an outcome for a new data instance based on prior data. The main types of machine learning are *supervised learning* and *unsupervised learning* that perform learning with labeled data and unlabeled data, respectively. A typical supervised learning task is a *classification* that finds the relationship between the objects in the training set and then assigns the unknown data to the classes previously determined. The classification has been known with the ability to give high accuracy on data estimation due to using labeled data. Nevertheless, obtaining labeled samples is a costly process, since it is usually based on expert experience. Therefore, this study focuses on *semisupervised learning* (SSL) in which a small number of labeled data and a large number of unlabeled data are used together.

Ordinal classification is a special case of multiclass classification in which an inherent ordering among the classes exists. The aim of the ordinal classification is to predict the class label of an unseen data instance by considering ranking relationships between the classes. For instance, an ordinal class attribute can have the five ranking values such as "strongly disagree", "disagree", "neutral", "agree", and "strongly agree" that are ordered from the worst situation to the best. Hence, the first class label can be considered five times worse than the last one. Another example is that the size of an object can be one of the following labels: "large",

*Correspondence: derya@cs.deu.edu.tr

"medium", and "small". According to the literature [1], in case there is an order relationship among the class label, ignoring this relationship can affect classification performance in a negative way.

The main motivations of this study are two-fold. (i) The existing SSL algorithms are capable of processing nominal or numerical class labels; however, we also need to capture the orderings among the categorical class labels. (ii) The traditional ordinal classification is limited to using only labeled ordinal data to construct a classifier; however, we also need a learning paradigm dealing with the design of classifiers in the presence of both labeled ordinal data and unlabeled ordinal data.

The main novelty of this paper is that the ordinal classification is used as a new approach for semisupervised learning. It proposes a new algorithm, called *semisupervised ordinal classification* (SSOC), which relies on semantic knowledge on the class label ordering and uses both labeled and unlabeled data together for classification. Our algorithm is different from the traditional regression and multiclass classification algorithms since, in the former, numeric target values are predicted based on a metric and, in the latter, there is no ordering between classes.

The main contributions of this study can be summarized as follows: (i) This is the first study that combines two concepts "semisupervised learning" and "ordinal classification" for the categorical class labels and introduces a new concept of "semisupervised ordinal classification"; (ii) this paper proposes a new algorithm, called SSOC, for semisupervised learning that takes into account the relationships between the class labels, especially class orderings such as bad, regular, and good; (iii) our new method allows a standard base learner to be applied to an ordinal classification task. This study is also original in that it compares alternative base learners conjunction with the proposed method, including decision tree (DT), support vector machines (SVM), k-nearest neighbors (KNN), random forest (RF) and neural network (NN); (iv) it is the first time that different ratios of labeled data (from 15% to 50%) are explored for ordinal classification.

The proposed SSOC method consists of the following main stages: The first stage is to train multiple binary classifiers on the existing labeled ordinal data. After that, the unlabeled data is labeled by the constructed classifiers via their predictions to generate additional labeled data, which is commonly referred to as pseudo-labeled data. Lastly, the final inductive classifier is built by using both originally labeled and pseudo-labeled ordinal data.

In the experiments, the effectiveness of the proposed SSOC method was verified on 10 benchmark ordinal datasets by comparing with the standard ordinal classification algorithm [1]. In addition, it was also compared with the well-known semisupervised classification algorithm, called YATSI (yet another two stage idea) [2]. The main findings can be concluded as follows:

- The proposed SSOC method resulted in a significant improvement over the existing YATSI method.
- The accuracy of prediction slightly increased as the number of labeled data samples increased.
- When the SSOC method was tested in combination with popular classification algorithms (DT, SVM, KNN, RF, and NN), the SSOC-RF algorithm achieved significantly better accuracy (93.57%) than the rest.
- Semisupervised ordinal classification methods (SSOC-DT, SSOC-SVM, SSOC-RF, and SSOC-NN) generally exceeded their supervised ordinal counterparts (OC-DT, OC-SVM, OC-RF, and OC-NN) in terms of accuracy when labeling at most half of the ordinal instances.

The implications/novelty of the results can be summarized as follows. The SSOC method performed high accuracy by the use of a set of labeled ordinal data and a set of unlabeled ordinal data. The Wilcoxon statistical test results showed that the differences in the performances of the proposed method and the existing

method are statistically significant. As a result, the proposed SSOC method can be successfully used for the tasks in management decisions and assessments where both labeled and unlabeled data are available and the classes of the objects are discretely ordinal, instead of nominal or numerical.

The proposed method (SSOC) has a number of advantages that can be summarized as follows:

- The traditional ordinal classification is limited to using only labeled ordinal data to build a model. The traditional semisupervised learning disregards the ordering information in class labels. The main advantage of the SSOC method is that it overcomes these two limitations and deals with the design of classification models in the presence of both labeled ordinal data and unlabeled ordinal data.
- In many real-world applications, an extremely huge amount of unlabeled ordinal data is available. However, labeling ordinal data is an expensive, difficult, tedious, or time-consuming process, since it usually requires human efforts, sometimes domain expert knowledge. Data labeling by domain experts has both explicit costs such as financial resources and implicit costs such as time spent. This is especially true for real-world applications that include learning from a large number of class labels and distinguishing similar classes. The SSOC method addresses this inherent bottleneck by automatically allowing the model to integrate the available unlabeled ordinal data with little or no cost.
- An important advantage of the SSOC method is that it can be utilized with the combination of any supervised base learner such as DT, SVM, and NN. The base learner is entirely unaware of the semisupervised ordinal classification method, in fact, it simply learns from the ordinal labeled and pseudo-labeled samples as if they were regular labeled instances.
- The ordinal classification methods often address the problems where labeled ordinal data is scarce or expensive. However, it is difficult to build a strong classifier with high generalization ability by using limited labeled ordinal data. The main idea behind the SSOC method is to take advantage of a huge amount of unlabeled ordinal data when building the ordinal classifier. In addition to labeled ordinal data, the SSOC method also exploits unlabeled data to help improve classification performance. Thanks to the SSOC method, the unlabeled data instances provide additional knowledge that is relevant for ordinal classification, and they can successfully be used to improve the generalization ability of the learning system.
- Another advantage is that the SSOC method can be applied to any ordinal data without any prior information about the given dataset. It does not provide any specific knowledge and specific assumptions for the given data.
- A large amount of data generated in real-life is unlabeled. Since the SSOC method covers a relatively wide domain, it enables enormous applications, and so it expands the application field of the ordinal classification algorithm.

The remainder of this paper is organized as follows: Section 2 presents the previous works on semi-supervised learning as well as ordinal classification; Section 3 explains the method proposed in this article; Section 4 presents the empirical results with dataset descriptions. Finally, concluding remarks and possible future works are given in Section 5.

2. Related work

Since our study combines two concepts ("semi-supervised learning" and "ordinal classification") for the categorical class labels for the first time, we herein present a literature review of previous studies on both of them separately.

2.1. Literature review for semisupervised learning

In the field of semisupervised learning (SSL), some new methods have been developed, as well as some existing supervised methods have been adapted to the problem. Until now, the semisupervised learning methods have been successfully applied to problems in many different areas such as health [3], security [4], education [5], geology [6], biology [7], real estate [8], and energy [9]. A comprehensive survey on the SSL topic was conducted by Van Engelen and Hoos in 2020 [10]. They give a broad overview of the SSL methods by presenting a new taxonomy, explain recent advances, and provide basic assumptions underlying SSL.

Semisupervised learning can be grouped into two different categories [10]: inductive learning and transductive learning. The goal of *inductive learning* [11] is to produce a prediction function that is defined on the whole input space. On the other hand, the idea in *transductive learning* [6, 12] is to directly perform predictions only for the unlabeled data. In other words, given a dataset consisting of labeled (X_L) and unlabeled (X_U) samples such that X_L, $X_U \subseteq X$, with labels $Y_L \in Y$ for the labeled samples, the inductive methods consider both X_L and X_U to yield a model $\mathrm{f}: X \mapsto Y$, whereas the transductive methods output predicted labels $\widehat{Y}_U$ for the given unlabeled samples (X_U) only. Our proposed algorithm (SSOC) is based on the inductive learning paradigm.

The semisupervised learning methods exist in the literature can be mainly categorized into generative models [10], cotraining [7, 8], self-training [5], semisupervised support vector machines (i.e., semi-SVM, S3VM) [9], disagreement-based methods [4] and graph-based methods [3, 11]. *Generative models* [10] build a joint probability model (i.e., Gaussian mixture model) depending on a distribution assumption, and the decision boundary is determined by using both labeled data samples and unlabeled data samples in the probabilistic framework such as expectation-maximization, maximum probability likelihood, or Bayes. The *cotraining methods* [7, 8] iteratively build two or more classifiers by using multiple different views of data, in which the most confident predictions of a classifier on the unlabeled samples are utilized as the labeled training samples by another classifier in each iteration. *Self-training* [5] is one of the widely used wrapper approaches, where a classifier firstly is trained with the initial labeled data for the purpose of classifying unlabeled samples, and then it is retrained by adding its predictions to the labeled data. The Semi-SVM method [9] is an extended version of the traditional SVM, which is used both labeled and unlabeled data to iteratively find a boundary that maximizes the margins between classes. The *disagreement-based* methods [4] train multiple learners and exploit the disagreements among the learners during the SSL process. The *graph-based* methods [3, 11] firstly construct a weighted graph, where each vertex refers to a data sample and the edge between two nodes represents the pairwise similarity of data samples, and then the methods use the graph to assign class labels to the unlabeled data samples. In our study, a new approach based on the self-training method is proposed to address the ordinal classification.

Similar to supervised learning, no technique has been discovered yet to determine prior information in which the SSL method is well-suited for a certain problem. Each method has some advantages and disadvantages, as well as each problem, even each dataset, has its own characteristics. Therefore, a combination of empirical evaluation and theoretical analysis should be used to determine a method that is best-suited to the given problem. For instance, Livieris et al. [5] compared four different semisupervised algorithms (cotraining, self-training, tri-training, and YATSI) to determine the best one for the student performance prediction problem. Similarly, Uylas Sati [13] tested many different SSL methods on real-world datasets.

A semisupervised learning problem is designed to reflect an assumption that the algorithm builds on.

The most commonly known assumptions are cluster assumption, manifold assumption, smoothness assumption, and low-density separation assumption [14]. The *cluster assumption* considers that samples belonging to the same group are likely to be of the same class. The *manifold assumption* states that data samples on the same low-dimensional manifold have a great probability of having the same class value. The *smoothness assumption* means that if samples close to each other in a high-density space, these samples may have the same label. The *low-density separation assumption* states that the class-decision boundary lies in the region where the data density is low in the input space. Since our proposed algorithm deals with the ordinal classification task, it is based on the assumption that if two samples are close, then, may have similar class rankings (orders).

Although most of the research on SSL has been centered on semisupervised classification, other problems such as *semisupervised clustering* [12] and *semisupervised regression* [8] have also been studied. The former focuses on building a model using labeled data and unlabeled data together to make a prediction based on continuous variables, whereas the latter aims to improve clustering results with the help of labeled data.

Since the aforementioned SSL algorithms focus on nominal classification, they are out of the scope of this paper. They do not capture and reflect the orderings among the class labels; hence, they may lead to construct inefficient models in terms of accuracy in the case of ordinal data. Unlike the previous studies, this paper proposes a new paradigm for semisupervised learning where ordinal data is used in the model construction process. More specifically, semisupervised learning is extended by considering the relationships between class labels.

2.2. Literature review for ordinal classification

Ordinal classification (OC) considers the problems where the class labels of the target attribute in the dataset follow a given order, such as *very hot, hot, warm, cold, and very cold* labels in weather prediction problem. Recently, the ordinal classification has received much attention in machine learning field with the development of a growing number of real-world applications, such as customer segmentation (i.e., gold, silver, and bronze), sentiment analysis (i.e., happy, natural, and sad), credit scoring (i.e., high, medium, and low risk levels), human age estimation (i.e., old, adult, and young), medical diagnosis (i.e., initiation, proliferation and progression stages of cancer), and survey analysis (i.e., disagree, neutral, and agree). In these contexts, the ability to capture the natural order of the labels is crucial to improve the classification performance of the model. The importance of taking into account inter-classes relations has already been proven in [1].

The OC approaches can be grouped under three categories: threshold approaches, naive approaches, and ordinal binary decomposition approaches [15]. The *threshold approaches* [16] obtain a set of thresholds by dividing the target variables into successive intervals, where each class label belongs to an interval limited by these thresholds. The *naive approaches* [17] use an appropriate simplifying assumption on the class labels to treat OC problems as if they were standard classification problems. For instance, one possible solution is to use different weights for different class labels, or another alternative solution is to map the class labels into numeric values and then implement a standard regression algorithm such as the support vector regression. The *ordinal binary decomposition* (OBD) *approaches* [1] transform the original ordinal classification task into a set of binary classification tasks. Each sub-task is separately solved by a binary classification algorithm, and then the final class labels are determined by ultimately combining the binary outputs into one label. In this study, we propose a new method based on the OBD approach.

The supervised ordinal classification approaches aforementioned above present limitations when only a small number of labeled ordinal data is available. The reason is that, in most ordinal classification problems,

data instances are labeled by a user and this process could be difficult, expensive, or time-consuming, especially when the number of class labels is high (i.e., more than three). In order to overcome this limitation, this article proposes a new paradigm for ordinal classification where both labeled ordinal data and unlabeled data are utilized during the model construction process.

Semisupervised ordinal regression (SSOR) has very little coverage in the literature, only several detailed analyses [18–21] has been performed. Table 1 shows the comparison of our SSOC study with the existing SSOR studies. Our method differs from the existing methods in three respects. First, they performed the prediction of numeric target values, while we focus on the classification of categorical target values. Second, they used different methods such as kernel discriminant learning [18], empirical risk minimization [19], max-coupled learning [20], and Gaussian processes [21]; whereas we utilized the ordinal binary decomposition method. Third, since their target values are numeric they used different evaluation metrics such as mean absolute error (MAE), mean squared error (MSE), and mean zero-one error (MZE), whereas we tested the performance with accuracy and F-score metrics. To the best of our knowledge, a semisupervised ordinal classification that will consider the ordinal categorical class labels has not been studied until now. To bridge this gap, in this study, a new algorithm for semisupervised learning is developed in the context of ordinal classification.

Table 1. Comparison of our study with the existing studies.

Ref	Year	Methods	Algorithm	Target class attribute		Evaluation metrics	Application domain
				Numeric	Categorical		
[18]	2016	Kernel discriminant learning (KDL)	Reduced empirical feature space semisupervised KDL for ordinal regression (ES-DL)	✓		MAE STD	Various domains
[19]	2019	Empirical risk minimization	Semisupervised ordinal regression with Gaussian kernel (SEMI-Kernel)	✓		MAE MSE MZE	Various domains
[20]	2011	Max-coupled learning	Semisupervised max-coupling algorithm	✓		MAE	Health
[21]	2013	Gaussian processes	Semisupervised Gaussian process ordinal regression (SSGPOR)	✓		MAE MZE	Various domains
Proposed approach		Ordinal binary decomposition	DT, SVM, KNN, RF, and NN		✓	Accuracy F-Score	Various domains

Cardoso and Domingues [20] propose a new learning paradigm, called max-coupled learning, and three different methodologies for the breast cancer application. Our study differs from their study in many aspects. First, they focused on the prediction of numeric target values (breast imaging reporting and data system - BIRADS scores such as 1,2,3), whereas we consider the categorical class labels where the metrical distances between adjacent categories are commonly unknown. For instance, the question of whether the difference between classes c_1 and c_2 is the same as between classes c_2 and c_3 is unknown. We propose a completely different approach that enables unlabeled data via the smoothness assumption (i.e., close samples are likely to have similar categorical class labels). Second, their method was proposed for handling data with two views (x and z), however, our method can be applied to any ordinal data without any restriction. Third, they presented their solution for a specific problem (breast cancer) by generating synthetic data, whereas we demonstrated the

generalization ability of our method on real-world datasets obtained from various domains. Fourth, they used a different method (max-coupling) which finds the maximum of the values obtained from the two "views" in prediction, while we utilized the ordinal binary decomposition method by considering upward and downward unions of classes. Fifth, to evaluate the performance, they used the MAE metric since their target values are numeric like a regression problem, whereas we assess the percentage of errors since the class labels are categorical.

3. Material and methods

3.1. Semisupervised learning

Semisupervised learning (SSL) is an important and useful type of machine learning concerned with using both labeled and unlabeled data to perform a particular learning task. It is conceptually situated between unsupervised learning and supervised learning and implies a more complex problem than both of them. The semisupervised learning topic has received much attention in many areas ranging from health [3] to education [5], where it is cheaper and easier to obtain unlabeled data rather than labeled data since it requires less expertise, effort, and time consumption.

The existing SSL algorithms are capable of processing nominal data; however, they do not capture and reflect the orderings among the class labels. Hence, they may lead to building inefficient models in the case of ordinal data. In order to address this limitation of the existing SSL algorithms, in this article, we propose a new SSL algorithm to exploit the presence of the orders among class labels in ordinal data. Our study aims to investigate how combining labeled and unlabeled ordinal data may improve the classification performance, and develop an algorithm that takes advantage of such a combination.

3.2. Ordinal classification

Ordinal classification is a special kind of supervised multi-class classification task that addresses the class attributes whose labels exhibit a form of ordering. For example, an ordinal class attribute can represent four wind-level categories in a wind speed prediction problem: *very high*, *high*, *moderate*, and *low*. It is clear that there exists an order among the class labels such that *very high* $\succ$ *high* $\succ$ *moderate* $\succ$ *low*, where $\succ$ represents that the former class label is better than the latter class label.

The conventional classification algorithms disregard the ordering information in class labels; however, it usually leads to a significant loss of performance. In a ordinal classification problem, a sample from the lowest class is remarkably different from a sample from the highest class, whereas two samples, from the lowest and middle classes, are relatively close to each other. Therefore, taking into account such ordering information aims to improve the performances of classifiers.

The traditional ordinal classification is limited to using only labeled ordinal data to construct a classifier. However, in this study, we propose a novel learning paradigm dealing with the design of classifiers in the presence of both labeled ordinal data and unlabeled ordinal data.

3.3. The proposed approach: semi-supervised ordinal classification

In this study, we propose a new approach "*semi-supervised ordinal classification*" (SSOC), which considers the hierarchical relationship among the target variables when using both labeled and unlabeled ordinal data for classification. In other words, we present a new classification strategy for incorporating ordinal data information into semisupervised learning.

There are many cases where unlabeled data in addition to labeled ordinal data can help in building or improving a classifier. Consider, for instance, the case of movie classification, where we want to assign a rating to a collection of movies such as very good, good, average, poor, and terrible ratings. However, most users might not have an interest in rating movies, thus labeled ordinal data is scarce or difficult to obtain; on the other hand, a huge amount of unlabeled data is available. This is where semisupervised ordinal classification comes in. The proposed SSOC method is able to use unlabeled data along with the labeled ordinal data to construct more efficient models in terms of accuracy compared to the standard ordinal classification methods.

A simple solution for the SSOC can be converting class labels into real values, e.g., {1,2,3,4,5}, and then solve the problem as a standard regression problem. However, this solution may lead to building unreliable models, since the metrical distances between adjacent categories are commonly unknown. For instance, the question is whether the difference between saying "strongly agree" and "agree" is the same as between saying "neutral" and "disagree". The answer is that it is not possible to ensure this assumption. Therefore, in this paper, we propose a completely different approach. We extended the ordinal classification algorithm presented in [1] to enable unlabeled data via the smoothness assumption (i.e., close samples are likely to have similar categorical class labels).

The proposed SSOC method is based on the *self-training* principle (also known as *self-learning* or *self-labeling*). Firstly, an ordinal classifier is trained in a standard way with an initial small number of labeled samples on an ordinal scale. After that, the unlabeled data is labeled by the constructed ordinal classifier via its predictions. Lastly, a purely ordinal classification algorithm builds the final inductive classifier by using both originally labeled and pseudo-labeled ordinal data together.

3.3.1. The formal definition of the proposed method

Formally let $\mathcal{X}$ be an input space with d-dimensional features, $\mathcal{X} \subseteq \mathbb{R}^d$, and, $\mathcal{Y} = \{c_1, c_2, ..., c_k\}$ be a finite set of predetermined ordered classes such that $c_1 \prec c_2 \prec ... \prec c_k$. We denote two sample sets $\mathcal{X}_L$ and $\mathcal{X}_U$ as the collection of input instances for the ordinal labeled and unlabeled samples, respectively, where $\mathcal{X}_L, \mathcal{X}_U \in \mathcal{X}$ and $\mathcal{X} = \mathcal{X}_L \cup \mathcal{X}_U$. Given a set of l labeled instances $\{(x_i, y_i)\}_{i=1}^{l}$ and u unlabeled instances $\{x_j\}_{j=l+1}^{(l+u)}$, we denote labeled data as $D_L = \{(x_1, y_1), (x_2, y_2), , (x_l, y_l)\}$ and unlabeled data as $D_U = \{x_{l+1}, x_{l+2}, , x_{l+u}\}$. Each data point (x_i, y_i) in the labeled data D_L consists of an instance x_i from a given input space $\mathcal{X}$, such that $x_i \in \mathcal{X}$, and has an associated label y_i, where y_i is one of the ordinal categories such that $y_i \in \mathcal{Y}$. The two datasets D_L and D_U form the training set D together with n instances, where $n = l + u$. We are especially interested in cases where $u >> l$ since an unlabeled instance is abundant and easily to be obtained while labeling the instance is difficult and expensive since it usually requires expert knowledge.

Definition 1 (Semi-supervised ordinal classification) *Semi-supervised ordinal classification refers to the problem that utilizes both ordinal labeled data D_L and unlabeled data D_U and aims to find a decision function $f : \mathcal{X}_{L+U} \to \mathcal{Y}$ that can correctly predict the class labels y^* of some unseen inputs x^*.*

The proposed SSOC method consists of two main steps. In the first step, the SSOC method applies an ordinal classification algorithm to the labeled ordinal instances in $\mathcal{X}_L$ to build an ordinal classifier and then produces pseudo-labels $\hat{\mathcal{Y}} = \{y_{l+1}, y_{l+2}, , y_{l+u}\}$ for the unlabeled data instances in $\mathcal{X}_U$ by using the predictions of the resulting classifier. In the second step, the ordinal classifier is then retrained on the newly obtained pseudo-labeled ordinal data in addition to the originally labeled ordinal data.

Definition 2 (Ordinal decomposition) *Assume that the output space is defined as* $\mathcal{Y} = \{c_1, c_2, , c_k\}$ *and the labels are ordered according to the ranking structure* $c_k \succ ... \succ c_2 \succ c_1$, *where* $\succ$ *denote this order information. Ordinal decomposition is to decompose the original ordinal classification problem involving* k *classes into* $k-1$ *binary classification problems to encode the ordering information among the class labels. The* i^{th} *binary problem is defined by separating the classes from* 1 *to* i, *denoted by* $\mathcal{Y}_- = \{c_1, c_2, , c_i\}$, *and the classes from* $i+1$ *to* k, *denoted by* $\mathcal{Y}_+ = \{c_{i+1}, c_{i+2}, , c_k\}$.

Rather than the standard class probability $P(label = c_i)$, the i^{th} problem estimates the binary composition probability $P(label \succ c_i)$, which is the probability of a sample of having a class greater than c_i in the ordinal scale. In this way, it takes into account the order among the classes since the upward and downward unions of classes are progressively considered during the creation of the binary datasets. The final output is then predicted by multiple models ($k-1$ models), which are trained in conjunction with a base learner, i.e., using a decision tree method.

Definition 3 (Binary composition probability) *Let* M_i *for* $i = 1, 2, , k-1$, *denotes the model constructed for the ordinal classification problem. Given a search instance* x, *an estimation* $M_i(x)$ *is considered as a prediction of the probability* $P(L_x \succ c_i)$ *that the class label of* x *is interpreted in* $\mathcal{Y}_+$, *where* L_x *denotes the class label of* x. *Binary composition probabilities on* $\mathcal{Y}$ *are formulated as follows:*

$$\begin{aligned} P(c_1) &= 1 - P(L_x \succ c_1) \\ P(c_i) &= P(L_x \succ c_{i-1}) \times (1 - P(L_x \succ c_i)) \qquad 1 < i < k \\ P(c_k) &= P(L_x \succ c_{k-1}) \end{aligned} \tag{1}$$

To assign a class label (L) to a new sample x, we need to calculate the probabilities of the k ordinal classes using k - 1 binary models. The prediction of the probabilities for the first and last class depends on a single model. The probability of an example being of the first class $P(c_1)$ is given by $1 - P(L_x \succ c_1)$ since we will only consider the first model that discriminates between c_1 and the rest. Similarly, the probability of the last class $P(c_k)$ is computed from $P(L_x \succ c_{k-1})$ in accordance with the one-vs-followers strategy. For the intermediate classes $1 < i < k$, the probability depends on a pair of models since the ordinal structure of a middle-class c_i can be reflected by two indicators: greater than its previous class $P(L_x \succ c_{i-1})$ and not greater than itself $(1 - P(L_x \succ c_i))$. In this way, the method considers the order in the frontiers. For instance, when the model is queried with a sample x for the class c_2, a vote for the higher classes c_3 and a vote for the lower classes c_1 should be considered. Therefore, the upward and downward unions of classes are progressively considered. In other words, adjacent classes are grouped together to encode the ordering among classes. For example, assume that there are five ordered classes: very sad, sad, natural, happy, and very happy; then the probability of each class is calculated as follows:

Ordinal decomposition	Binary composition probability
{verysad} {sad, natural, happy, veryhappy}	P(verysad) = 1 - P(label $\succ$ verysad)
{verysad} {sad} {natural, happy, veryhappy}	P(sad) = P(label $\succ$ verysad) $\times$ (1 - P(label $\succ$ sad))
{verysad, sad} {natural} {happy, veryhappy}	P(natural) = P(label $\succ$ sad) $\times$ (1 - P(label $\succ$ natural))
{verysad, sad, natural} {happy} {veryhappy}	P(happy) = P(label $\succ$ natural) $\times$ (1 - P(label $\succ$ happy))
{verysad, sad, natural, happy} {veryhappy}	P(veryhappy) = P(label $\succ$ happy)

At the end of the binary decomposition, the label that has the highest probability is predicted as the final class for a given instance. This probability estimation is meaningful for ordinal classification since it typically assigns higher scores to the instances from higher classes. As an example, an instance x of class c_i will be in a positive class ($\mathcal{Y}_+$) of all i+1 binary classifiers, and therefore the model M_i should return a high score. Thereby, the model M_i yields reasonable probability estimation that is consistent in the sense of $M_i(x) \geq M_{i+1}(x)$. Essentially, the mapping $i \mapsto M_i(x)$ is similar to a decumulative distribution function.

Coming back to the problem of semisupervised ordinal classification, an ordinal classifier model is constructed by using labeled data until now. In the next step, the pseudo-labeling step, the resulting ordinal classifier is used to predict labels for the previously unlabeled instances. Since the SSOC method takes into account the relationships among the class labels, an unlabeled sample will be assigned to a pseudo-label by maximum ranking probability to which it is associated. The pseudo-labeled data is appended to the initial labeled data. Finally, the SSOC method retrains the underlying ordinal classifier with this augmented data.

3.3.2. The algorithm of the proposed method

The general framework of the proposed SSOC approach is given in Algorithm 1. The algorithm consists of five steps.

Step 1: In the first step (lines 9-18), called *ordinal binary decomposition*, the method converts the original ordinal labeled dataset D_L into a set of binary datasets $\{D_i\}_{i=1}^{k}$ to encode the ranking relation among the classes. In this process, the label y_j associated with the instance x_j is replaced with $y_j = 0, \forall\, y_j \preceq c_i$, and, $y_j = 1, \forall\, y_j \succ c_i$. In other words, if we consider the class c_i, the class values higher than c_i are labeled as 1 and the others are labeled as 0. In this way, the algorithm transforms the ordinal classification problem involving k classes into $k-1$ binary classification problems. For every $i \in \{1, 2, ..., k-1\}$, the dataset D_i is generated using classes $\{c_1, c_2, ..., c_i\}$ against $\{c_{i+1}, ..., c_k\}$.

Step 2: In the second step (lines 19–23), a separate model M_i is trained for each binary dataset D_i by using a base learner. The first model M_1 is built to estimate what is the probability of a given sample to belong to any of the classes that are located higher than class c_1 (class $\succ$ c_1), the second model M_2 is constructed to estimate the probability of belonging to any of the classes that are located higher than c_2 (class $\succ$ c_2), and so on. Finally, the last model M_k is trained to estimate the probability of belonging of a sample to the highest class c_k (class $\succ$ c_k). In the training phase, a standard classification algorithm (i.e., DT, SVM, and NN) can be used as a base learner.

Step 3: In the third step (lines 24-33), *a* query instance $x_i \in D_U$ is submitted to all models M^*, and their estimations ($M^*(x_i)$) are compared, and the class label with the highest probability is assigned to x_i as a pseudo-label. This process is repeated for each instance x_i in the unlabeled dataset D_U. In the foreach-loop, the probability of belonging of the query instance x_i to each class is estimated by using Equation (1). While the probability of the first class $P(c_1)$ is computed (line 27), the upward union of classes $P(L_x \succ c_1)$ is subtracted from 1. While the probability of the last class $P(c_k)$ is computed (line 31), the upward union of classes $P(L_x \succ c_{k-1})$ is directly considered. On the other hand, the probabilities for the intermediate classes $2 \leq j \leq k-1$ are computed in the for-loop (lines 28-30) by considering both upward and downward unions of classes. In the end, the label that has the highest (MAX) probability is predicted as the final class (y) for the query instance x_i.

Step 4: In the fourth step (lines 34-35), the binary datasets are generated from the combined (labeled and

pseudo-labeled) data, similar to the first step. In other words, the algorithm repeats Step 1 by simply using both the originally labeled and pseudo-labeled ordinal data as if it was regular labeled ordinal data.
Step 5: In the fifth step (lines 36-37), the ordinal classifier is re-trained by using the newly obtained binary datasets, similar to the second step. In other words, the ordinal classifier is re-trained with its own estimations by enlarging its initial labeled set.

Inputs:
D_L: the labeled ordinal dataset $D = \{(x_1, y_1), (x_2, y_2), ., (x_l, y_l)\}$ with l intances
D_U: the unlabeled dataset $D = \{x_{l+1}, x_{l+2}, ., x_{l+u}\}$ with u intances
$\mathcal{Y}$: ordinal class labels $y \in \{c_1, c_2, ., c_k\}$ with an order $c_k \succ ... \succ c_2 \succ c_1$
k: the number of classes
Output:
M^* : semisupervised ordinal classification model
Begin:
// Step 1 - Construction of binary datasets from the labeled data, D_L
for $i \leftarrow 1$ **to** $k-1$ **do**
  foreach (x_j, y_j) ***in*** D_L **do**
    if $(y_j \preceq c_i)$ **then**
      $D_i.Add(x_j, 0)$ // The class values smaller than or eqaul to c_i are labeled as 0
    else
      $D_i.Add(x_j, 1)$ // The class values higher than c_i are labeled as 1
    end
  end
end
// Step 2 - Construction of unified binary models
for $i \leftarrow 1$ **to** $k-1$ **do**
  $M_i = Train(D_i)$ // Building a classifier on the training set using a base learner
  $M^* = M^* \cup M_i$
end
// Step 3 - Pseudo-labeling unlabeled data, D_U
foreach x_i ***in*** D_U **do**
  $y = M^*(x_i) =$ MAX (// The label with max probability is predicted as the final class
  $P(c_1) = 1 - P(L_i \succ c_1)$ // The probability of belonging to the first class
  for $j \leftarrow 2$ **to** $k-1$ **do**
    $P(c_j) = P(L_i \succ c_{j-1}) \times (1 - P(L_i \succ c_j))$ // The probability of belonging to a middle class
  end
  $P(c_k) = P(L_i \succ c_{k-1}))$ // The probability of belonging to the last class
  $D_L.Add(x_i, y)$
end
// Step 4 - Construction of binary datasets from the labeled and pseudo-labeled data
Repeat step 1
// Step 5 - Construction of semisupervised ordinal classification model
Repeat step 2
End Algorithm

Algorithm 1: Semi-supervised Ordinal Classification (SSOC)

The SSOC method builds $k-1$ models on the training set, where k is the number of classes. When we use a base learner with complexity T, the overall time complexity of the method for training is given by $O((k-1).T(l) + u + T(l+u))$, where l and u are the number of labeled and unlabeled ordinal instances, respectively.

3.3.3. An example of the proposed method

Figure 1 shows a concrete example of the proposed approach: semi-supervised ordinal classification (SSOC). Assume that each sample in the given dataset has three features (gender, age, and status) and is classified into one of the predefined four categories on an ordinal scale (none $\prec$ low $\prec$ medium $\prec$ high). The SSOC method consists of six consecutive steps grouped under two main stages. The first stage (Figure 1a) is to train ordinal classifiers on the existing labeled ordinal data, and after that, use the predictions of these classifiers on the previously unlabeled data to generate additional labeled data, which is commonly referred to as pseudo-labeled data. This process allows unlabeled data to be introduced to the training process in an efficient and straightforward manner. In the binary decomposition step, the ordinal classification problem involving k classes ($k = 4$ in this example) is converted into $(k-1)$ binary classification problems to encode the ordering among the class labels such that $P(class \succ none)$, $P(class \succ low)$, and $P(class \succ medium)$. In the second main stage (Figure 1b), the ordinal classifiers are retrained on the newly obtained pseudo-labeled data in addition to the originally labeled data. The method simply passes both the originally labeled and pseudo-labeled ordinal data to a supervised base classifier as if they were regular labeled ordinal instances.

4. Experimental studies

We conducted five experiments on 10 ordinal datasets for illustrating the efficiency and validity of the proposed SSOC algorithm.

Experiment 1 - We compared the SSOC method with the existing semi-supervised learning algorithm (YATSI) presented in [2] to show the superiority of our algorithm on ordinal data.

Experiment 2 - We investigated the performance of the SSOC method with different ratios of labeled data (from 15% to 50%) to determine its impact on the method.

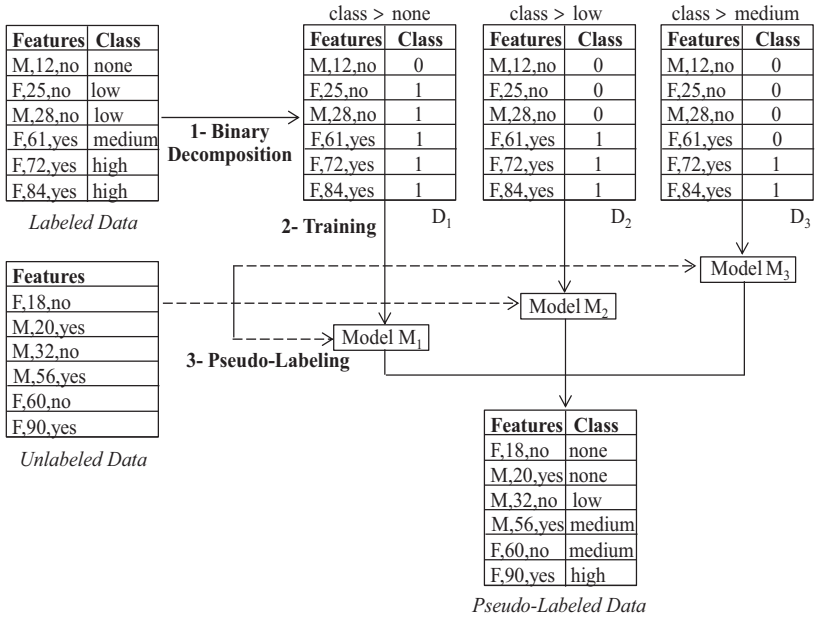
Experiment 3 - We evaluated the performance of the SSOC method with different base classifier combinations (DT, SVM, KNN, RF, and NN).

Experiment 4 - We compared the SSOC method with the standard ordinal classification algorithm presented in [1] to show the effectiveness of our method.

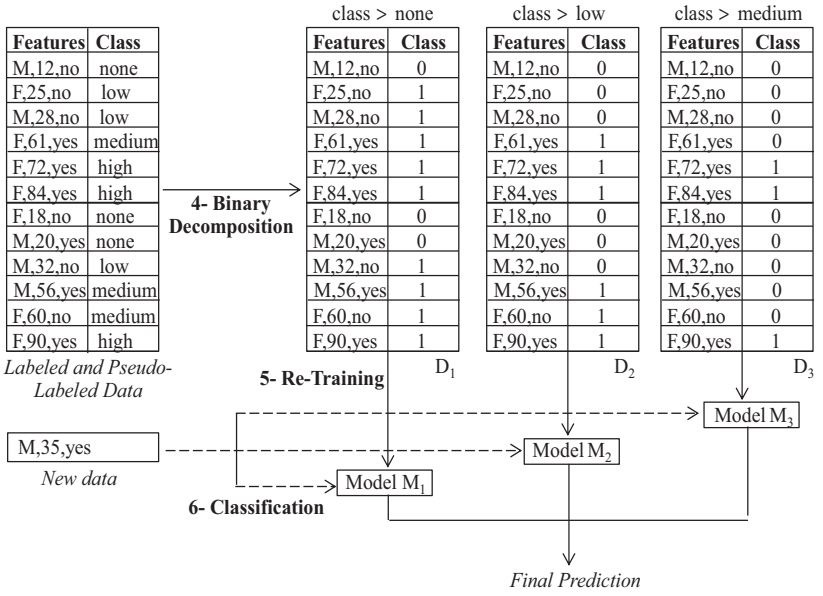
Experiment 5 - We investigated that after retraining the classifier with new pseudo-labels, how much the performance is increased compared to the performance with initial training labels.

In these experiments, the performances of the classification models were evaluated according to the accuracy metric, which is the percentage of correctly predicted samples. Accuracy is calculated by the formula: $Accuracy = (TP+TN)/(TP+TN+FP+FN)$, where true positive (TP) and true negative (TN) indicate the correct predictions for the positive samples and negative samples, respectively; whereas false positive (FP) and false negative (FN) represent the misclassified positive and negative samples, respectively. The SSOC and YATSI methods were compared by using the original dataset for testing. Ordinal classification accuracies were obtained by using the 10-fold cross-validation technique, in which the data is randomly divided into ten disjoint and equal-sized partitions, and one of the partitions is kept for the testing process, while the remaining partitions are utilized for the training process.

We implemented the proposed method in the Java programming language by using the WEKA machine learning library [22]. We also compared our method with the existing methods [1, 2] which are available as packages for the WEKA tool. The comparison results were evaluated by using Wilcoxon statistical test to ensure the significance of the performance results obtained by the methods on the ordinal datasets.



(a)



(b)

Figure 1. An example of the proposed SSOC method.

In each experiment, all the input parameters of the algorithms were left as default values, except the KNN algorithm. The number of neighbors (the parameter k) was selected as $log_2(n)$ based on the previous study [23], where n is the number of objects in the respective dataset. The default value of this parameter is 1; however, it is too small and it generally does not make sense to choose the parameter k so small when a large number of instances are available in the dataset.

4.1. Dataset description

In order to demonstrate the validity and effectiveness of the proposed SSOC method, the experiments were carried out on 10 ordinal datasets available in the UCI (the University of California at Irvine) and OpenML repositories. Table 2 gives the names and main characteristics of the datasets, including the number of features, classes, instances, the number of instances per class, and the source repository.

Table 2. Summary description of the datasets.

No	Dataset	#Features	#Instances	#Classes	Class Distribution	Source
1	Automobile	26	205	7	$(0, 3, 22, 67, 54, 32, 27)$	UCI
2	Car Evaluation	7	1728	4	$(384, 69, 1210, 65)$	UCI
3	Employee Selection (ESL)	5	488	9	$(2, 12, 38, 100, 116, 135, 62, 19, 4)$	OpenML
4	Eucalyptus	20	736	5	$(180, 107, 130, 214, 105)$	OpenML
5	Nursery	9	12960	5	$(4320, 2, 328, 4266, 4044)$	UCI
6	Squash-Stored	25	52	3	$(8, 21, 23)$	OpenML
7	User knowledge modeling (UKM)	6	403	5	$(50, 129, 122, 102)$	UCI
8	Volcanoes on venus (B6)	4	10130	5	$(88, 86, 58, 68, 2952)$	UCI
9	Wine quality-red	12	1599	6	$(10, 53, 681, 638, 199, 18)$	UCI
10	Wine quality-white	12	4898	7	$(20, 163, 1457, 2198, 880, 175, 5)$	UCI

Each dataset is divided into two parts: labeled and unlabeled sets because the main purpose of the SSOC method is to utilize unlabeled ordinal data to improve learning performance. We used a random selection of instances that were marked as labeled samples, and the class labels of the rest of the samples were removed. To investigate the effect of the amount of labeled ordinal data, we considered different percentage values when dividing the dataset, varying from 15% to 50% with an increment of 5%. For example, assuming a dataset that contains 1000 instances, when the labeled data rate is 10%, 100 samples were put into the labeled dataset D_L with their class labels, while the remaining 900 samples were placed into the unlabeled dataset D_U without their class labels.

4.2. Experimental results

In this study, the proposed SSOC methods were tested with different base classifiers (i.e., DT, SVM, KNN, RF, and NN) on the benchmark datasets. In other words, the SSOC method with each different base classifier is built as an independent classifier. From here onwards, the abbreviation of the method followed by the abbreviation of the base classifier technique is used to refer to the related approach. For example, SSOC-SVM refers to the SSOC method with the SVM base classifier.

4.2.1. The results of experiment 1

In this experiment, we compared the proposed method (SSOC) with the existing method (YATSI) [2] on various ordinal datasets in terms of classification accuracy (%). YATSI (yet another two stage idea) is a collective classifier for semisupervised learning and it can be used in the WEKA machine learning tool [22] with additional Collective Classification package. Table 3 shows the comparison results. The selection of initial labeled data was repeated five times for the proposed method and the average results were reported here. It can be observed from Table 3 that the proposed SSOC method resulted in performance improvement over the existing YATSI method on 10 benchmark ordinal datasets. For example; for the "UKM" dataset, the SSOC-DT algorithm (90.07%) is significantly better than the YATSI-DT algorithm (83.13%). The predictive performance difference between SSOC and YATSI is considerable for almost all datasets, for different percentages of labeled data (i.e., 25%, 50%, and 75%), and for all machine learning algorithms (DT, SVM, KNN, RF, and NN).

Hence, the results indicated that semisupervised based ordinal classification could generally produce a robust model with high prediction accuracy.

Although the SSOC method usually outperforms YATSI, in some cases the performance of the YATSI is better than SSOC, especially for the Car Evaluation and Nursery datasets. This is probably true when the data is noisy or some classes are similar to each other. The reason is probably because of the fact that YATSI is a collective classification algorithm that uses an of-the-shelf classifier in a first step and benefits from the weighted nearest neighbor approach. The main idea behind collective classification is that the predicted class label of a test-sample is influenced by the predictions made by related samples. However, if the user does not determine the optimal number of neighbors as well as the optimal weights, YATSI may lose this advantage, especially when the data is noisy.

Although accuracy is a widely used evaluation metric, sometimes it is not enough alone due to its strong bias against the rare (minority) class, especially when the data is imbalanced. Since even considering all examples as the majority class, high accuracy can be achieved. Since some of the datasets used in this study is imbalanced we applied the synthetic minority oversampling technique (SMOTE) [24] and we used the F-Score measure to evaluate the results. Options for setting hyperparameters were left as default values from the WEKA software package, except the percentage of SMOTE instances to be created for the minority class. It was set according to the ratio of instances between the majority and minority classes in the particular dataset. Table 4 shows the comparison results in terms of *F-Score* which is the harmonic mean of precision and recall. The selection of initial labeled data was repeated five times and the average results were reported here. The value of the F-Score metric is ranged between 0 and 1, where 1 is the best value. As seen from Table 4; the F-score values obtained by the SSOC-RF algorithm are close to 1, especially when 75% of the data was considered as labeled. The F-score values slightly increased as the number of labeled data samples increased. As a result, it can be concluded that it is possible to achieve high F-score values by the proposed method for the ordinal data.

Even though the proposed SSOC method has higher accuracy than the YATSI method on average, the results were evaluated by using a formal statistical test to ensure the differences in performance are statistically significant. Table 5 demonstrates the P-values computed through the Wilcoxon test by pairwise comparisons between the methods. Based on the reported P-values, it can be noted that the results are statistically significant since almost all the obtained p-values are very smaller than the significance level ($\alpha = 0.05$). Hence, the statistical test strongly demonstrated the existence of significant differences between the methods.

4.2.2. The results of experiment 2

In this experiment, we considered different ratios of labeled data since the classification performance can be varied with the amount of labeled data. Figure 2 shows the accuracy results obtained by the proposed SSOC method with different ratios of labeled data varying from 15% to 50% with an increment of 5%. The results indicated that considerable improvement could be expected from our method in circumstances where the ratio of labeled data grows. The best accuracy on average (79.35%) was achieved when 50% of the data was considered as labeled. The second-best accuracy on average (78.12%) was obtained under the circumstance: 45% labeled and 55% unlabeled data. The percentages of 40% and 35% followed them with the average accuracy values of 76.98% and 75.54%, respectively. Hence, it can be concluded that, in ordinal classification tasks, a set of labeled ordinal data can be employed by the proposed SSOC algorithm to consistently improve performance. This result indicates that the initial small number of labeled ordinal data can be insufficient enough to learn the model, and therefore, additional labeled data can improve the classification performance.

Table 3. Comparison of the proposed method (SSOC) and the existing method (YATSI) [2] in terms of accuracy (%).

Dataset	Labeled data (%)	SSOC-DT	SSOC-SVM	SSOC-KNN	SSOC-RF	SSOC-NN	YATSI-DT	YATSI-SVM	YATSI-KNN	YATSI-RF	YATSI-NN
Automobile	25	56.78	63.02	48.39	**71.71**	66.15	48.78	46.34	50.24	47.32	47.80
	50	73.56	74.54	58.83	**87.80**	81.95	57.07	57.56	50.73	56.10	55.12
	75	76.88	80.20	67.41	**96.10**	91.12	60.49	64.39	59.02	61.95	63.41
Car Evaluation	25	81.42	82.13	75.21	89.28	**95.30**	83.39	89.12	76.33	84.38	91.78
	50	86.79	80.31	82.35	96.06	**98.62**	91.32	92.94	82.99	92.71	94.91
	75	91.63	81.31	91.86	98.34	**99.76**	91.90	92.01	88.54	92.82	93.23
ESL	25	62.83	58.98	59.92	68.93	**71.15**	60.25	51.23	60.45	61.68	63.11
	50	68.57	63.81	66.68	**75.33**	74.18	67.21	58.40	65.16	68.44	68.03
	75	72.75	65.53	69.47	**79.67**	76.43	64.96	59.22	65.16	64.55	65.16
Eucalyptus	25	61.90	65.76	47.93	67.42	**68.13**	52.04	53.53	48.23	51.63	50.82
	50	66.85	70.22	57.04	**81.20**	79.08	54.35	56.39	54.21	56.25	55.98
	75	72.80	72.55	61.82	**90.35**	85.90	50.00	54.08	55.03	55.43	54.76
Nursery	25	92.78	91.88	91.87	96.07	**99.58**	93.74	94.05	92.35	95.00	96.25
	50	96.30	92.64	95.84	98.74	**99.98**	96.99	95.44	95.81	97.38	98.74
	75	97.43	93.09	97.26	99.60	**99.98**	96.63	94.93	96.03	97.38	97.57
Squash-Stored	25	64.62	65.00	58.08	**70.00**	67.69	38.46	40.38	42.31	42.31	40.38
	50	69.23	77.31	62.31	81.54	**82.69**	57.69	48.08	53.85	55.77	51.92
	75	78.46	87.31	68.46	**91.15**	90.38	57.69	51.92	44.23	51.92	50.00
UKM	25	90.07	81.84	72.95	90.82	**95.53**	83.13	77.42	72.95	83.13	83.37
	50	95.53	83.13	84.42	**96.92**	96.87	85.36	83.37	84.37	84.86	86.10
	75	96.97	91.41	87.39	**98.86**	98.21	84.62	82.63	82.88	84.12	84.12
Volcanoes	25	96.20	96.21	96.23	**97.11**	96.48	96.47	96.20	96.23	96.41	96.42
	50	96.25	96.21	96.42	**98.17**	96.62	96.54	96.20	96.43	96.54	96.58
	75	96.30	96.21	96.53	**98.89**	96.67	96.52	96.21	96.51	96.54	96.60
WQ-Red	25	61.79	57.66	57.36	**69.79**	60.38	57.22	57.16	57.47	58.04	57.29
	50	68.09	58.09	61.71	**81.78**	63.63	58.16	58.16	59.16	58.85	57.72
	75	77.24	58.41	63.09	**91.56**	67.03	57.79	57.35	60.54	59.47	60.23
WQ-White	25	56.57	45.87	53.44	**67.92**	53.84	52.10	46.96	53.63	54.23	51.12
	50	65.66	51.67	57.76	**81.76**	56.61	55.59	53.74	56.57	56.94	55.96
	75	73.44	50.32	59.58	**91.19**	57.92	55.45	53.12	56.98	56.17	54.02

Table 4. Comparison of the methods in terms of F-Score.

Dataset	Labeled data (%)	SSOC-DT	SSOC-SVM	SSOC-KNN	SSOC-RF	SSOC-NN
Automobile	25	0.5206	0.5967	0.3481	0.6837	0.6227
	50	0.7708	0.7315	0.5584	0.9098	0.8448
	75	0.8402	0.7414	0.6222	**0.9785**	0.9458
Car Evaluation	25	0.5679	0.5359	0.4350	0.7488	0.8857
	50	0.6494	0.5038	0.4899	0.9106	0.9718
	75	0.8580	0.5074	0.7885	0.9737	**0.9945**
ESL	25	0.5451	0.3863	0.4882	0.5980	0.5942
	50	0.5823	0.4355	0.5616	0.6817	0.6377
	75	0.6316	0.4806	0.5731	**0.8035**	0.6699
Eucalyptus	25	0.5971	0.6359	0.4645	0.6582	0.6660
	50	0.6501	0.6821	0.5455	0.8016	0.7818
	75	0.7173	0.7060	0.5931	**0.8989**	0.8535
Nursery	25	0.7079	0.7283	0.6976	0.8127	0.8907
	50	0.8003	0.7740	0.7339	0.8837	0.9125
	75	0.8229	0.7856	0.7785	0.9003	**0.9125**
Squash-Stored	25	0.6107	0.6715	0.5325	0.709	0.6976
	50	0.6302	0.7769	0.5729	0.8081	0.8232
	75	0.806	0.8703	0.6899	**0.9083**	0.9011
UKM	25	0.9043	0.7772	0.7324	0.9112	0.9539
	50	0.9543	0.8053	0.8369	0.969	0.9699
	75	0.968	0.9093	0.8702	**0.9891**	0.9844
Volcanoes	25	0.4388	0.5810	0.43815	0.5207	0.5482
	50	0.4673	0.5810	0.5174	0.7300	0.5536
	75	0.5648	0.5810	0.5295	**0.9035**	0.7099
WQ-Red	25	0.3973	0.3305	0.3523	0.5676	0.4160
	50	0.4659	0.3325	0.3938	0.7639	0.4697
	75	0.5321	0.3341	0.3935	**0.9265**	0.5210
WQ-White	25	0.3565	0.2365	0.3164	0.5372	0.3446
	50	0.4487	0.2729	0.3386	0.7606	0.3800
	75	0.5152	0.2704	0.3638	**0.8928**	0.4116

Table 5. Wilcoxon test results.

Compared algorithms	P-value	Significance level
SSOC-DT vs. YATSI-DT	0.0000971	very strong
SSOC-SVM vs. YATSI-SVM	0.0349977	strong
SSOC-KNN vs. YATSI-KNN	0.0028530	very strong
SSOC-RF vs. YATSI-RF	0.0000017	very strong
SSOC-NN vs. YATSI-NN	0.0000017	very strong

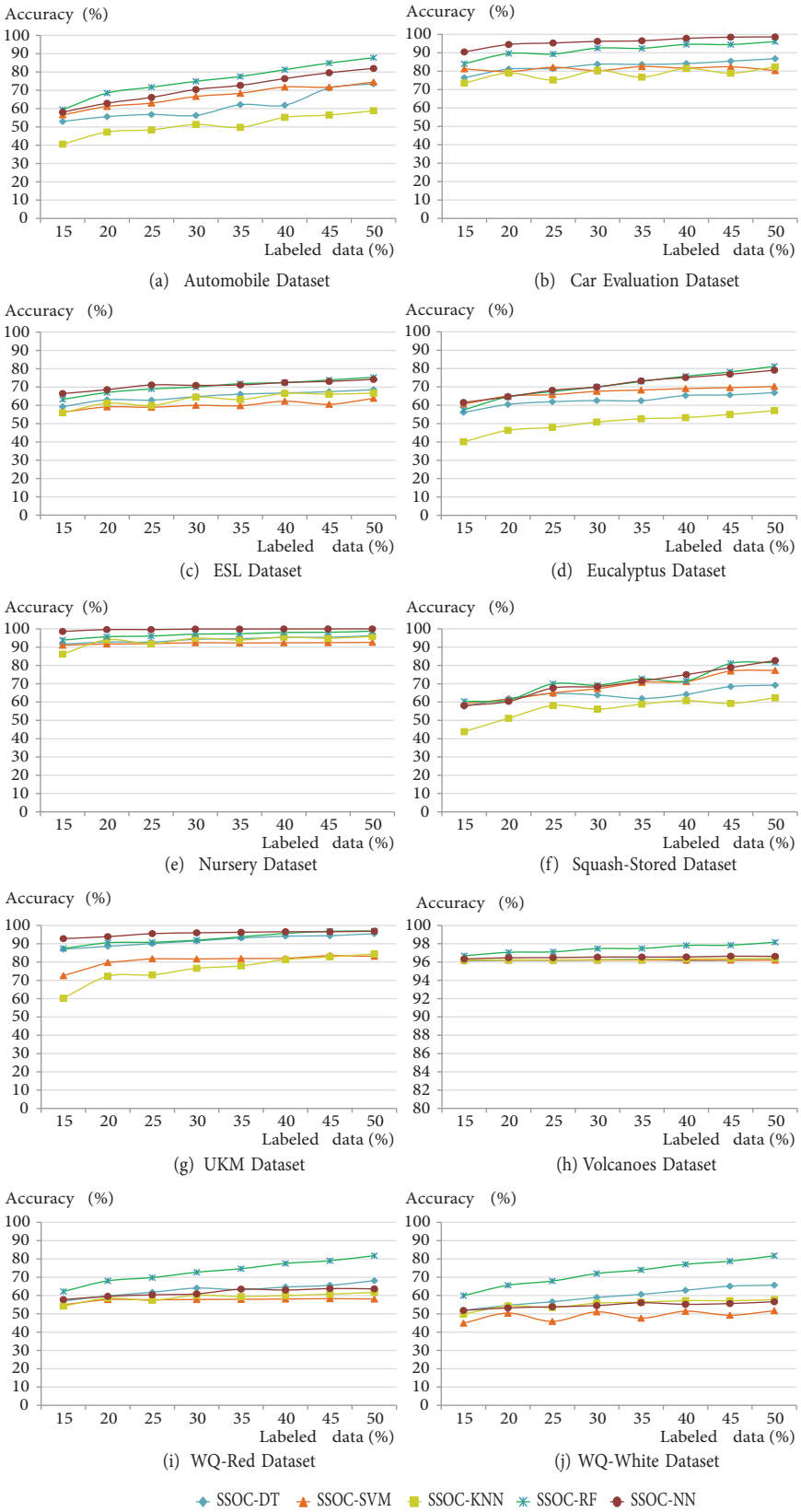


Figure 2. Accuracy results obtained with different ratios of labeled data.

As has been observed in the Nursery, UKM, and Volcanoes datasets, it is possible to provide good generalization ability for the ordinal classification problem by applying a small number of labeled data and a large number of unlabeled data. This is probably because of the following reasons. The Nursery and Volcanoes datasets are the largest datasets among others, hence the initial labeled ordinal data is sufficient enough to learn the model. This makes the model more robust, and therefore, unlabeled data can be labeled correctly by using this initial model. The Volcanoes dataset has the least number of features among the other datasets, and therefore, it is free from irrelevant and redundant features. The presence of irrelevant or redundant features may mislead the algorithm to produce irrelevant patterns, leading to classification errors. The UKM dataset is a relatively balanced dataset, and therefore, providing a representative investigation and being able to reliably classify all classes. When the dataset is imbalanced, the instances from the minority class may not exist in the small labeled data enough for learning. Furthermore, the model may fail to converge to a correct solution and cannot give good results, if labeled data does not represent patterns in the data. Performance declines could be observed when there was a mismatch between the classes available in the labeled ordinal data and the classes present in the unlabeled data. Therefore, the selection of optimal labeled data ratio is restricted so that at least one sample per class should be labeled. In multiclass classification, the number of labeled data samples can be increased linearly with respect to the number of classes.

It experimentally confirmed that the characteristics of the ordinal data and their partitioning may have an important impact on the classification performance of different machine learning algorithms. To provide a realistic evaluation, researchers should run alternative algorithms on available datasets with different ratios of labeled and unlabeled ordinal data. For example, in this study, it was observed from Figure 2 that the SSOC-RF and SSCO-NN algorithms were usually worked well even if a small number of labeled data is available.

The results obtained from the Nursery and Volcanoes datasets are quite stable, no matter what ratio of labeled data is used. This is probably because of the fact that they are the largest datasets and so the initial labeled data were supplied sufficiently, and therefore, the classification accuracy did not change so much by adding more unlabeled data. On the other hand, the labeling process is quite important for other datasets. For example, for the Automobile dataset, after adding 5% labeled data, we found that the classification accuracy of the SSOC-RF algorithm increased from 84.88% to 87.80%.

4.2.3. The results of experiment 3

A key property of our SSOC method is that it can be applied to any ordinal data in combination with any given base learner. The performance of the method is dependent on the selection of the appropriate base learning algorithm. Therefore, in this experiment, the proposed approach was tested on various datasets in combination with five popular classification algorithms, including DT, SVM, KNN, RF, and NN.

Figure 3 shows the radar and bar charts that represent the accuracy values obtained from each algorithm for the ordinal datasets considered when using a 75%-25% labeled-unlabeled ratio. We selected this ratio since the accuracy of prediction increased as the number of labeled data samples increased, and the characteristics underlying the data cannot be suitable for training with a small ratio of labeled data due to many different reasons, i.e., small data size, imbalanced data, noisy data, the presence of similar classes, and a large number of classes. The radar chart (Figure 3a) illustrates the classification performance as the distance from the center; thus, a higher area indicates a better classification performance. It helps us to comparatively visualize the accuracy of the methods for each dataset separately. Figure 3b shows the same results in a different way aiming to compare the accuracy values on all the datasets.

According to the results, SSOC-RF achieved slightly better accuracy (93.57%) than the rest. This is probably because RF is an ensemble learning method that constructs many decision trees to improve the classification performance. The SSOC-NN and SSOC-DT methods follow the SSOC-RF method with the accuracy values of 86.34% and 83.39%, respectively. It seems that KNN is not a good choice as a base learner for SSOC since their combinations showed the worst performance (76.29%) on almost all datasets. Furthermore, it was observed that SSOC-SVM provided better results than SSOC-KNN, although the differences between them are less remarkable than in the cases of other methods adopted as base learner.

As has been empirically observed, the characteristics of the ordinal data may have an important impact on the classification performance of different base learners. While the best average accuracy (96.92%) was achieved for the Volcanoes dataset, the worst average values were generally obtained for the WQ-red and WQ-white datasets. Some base learners may work well on particular types of datasets and may perform poorly on others. Therefore, a combination of empirical evaluation and theoretical analysis should be used to determine the best base learner for the given problem.

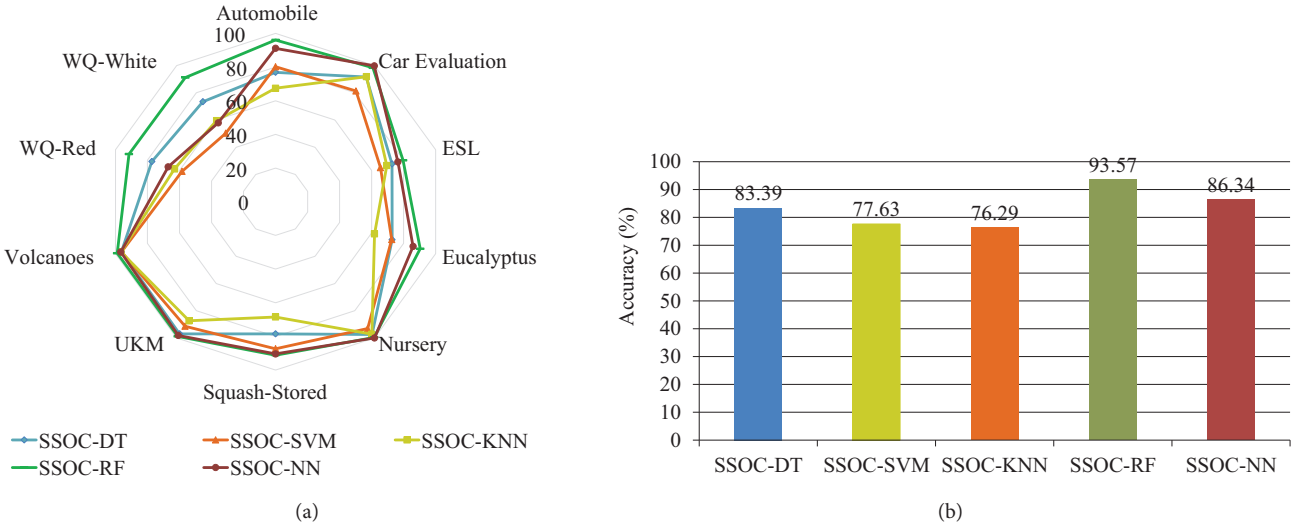


Figure 3. Accuracy results obtained with different ratios of labeled data.

In this study, we considered different ratios of labeled data since the classification performance can be varied with the amount of labeled data. For example, when 25%–75% labeled-unlabeled ratio was considered, the results given in Table 3 and Figure 2 showed that the SSOC-DT, SSOC-RF, and SSOC-NN methods achieved the high accuracy values ($\geq$80%) on some datasets such as the Car Evaluation, Nursery, UKM, and Volcanoes datasets. However, a small ratio of labeled data could sometimes be insufficient enough to train the model. This is especially true in the following cases: (i) When the dataset size is small, the insufficient labeled data may not represent patterns in the data; (ii) when the dataset is imbalanced, the instances from the minority class may not exist in the small labeled data enough for learning; (iii) when the data is noisy, a small portion of labeled data may not tolerate the noise itself; (iv) when some classes are similar to each other, a small portion of labeled data may not contain useful information for distinguishing them; (v) when the number of classes is large, a small number of labeled data may not include at least one sample per class.

Since SSOC is a metaalgorithm, it allows any supervised base learner to be applied, such as DT, SVM, and NN. Since deep learning (DL) has been attracting attention in the machine learning community in recent years,

our method can also be combined with it. Since deep learning has been proven to be a powerful ML technique in many studies, it can be used to improve the classification performance of our method. We performed an experiment to investigate the performance of SSOC-DL by using the WekaDeeplearning4j package [25]. The first experimental results showed that SSOC-DL outperformed the existing OC-DL on average, similar to other base learners. The optimal hyperparameters could be found and increased to build models with higher performance. However, we can have challenges of high computational time and high power consumption. Deep learning has generally been used to solve complex problems such as image classification, speech recognition, and natural language processing since it has ability to automatically to extract useful features (such as edges in images). Actually, the datasets used in this study contain simple transactional data and are not complex to be analyzed by deep learning. However, the proposed SSOC method can take advantage of deep learning when classifying ordinal image, voice, text, and video data.

4.2.4. The results of the experiment 4

This experiment was conducted to evaluate semisupervised learning in the case of ordinal classification. The central question in this experiment is: whether the introduction of unlabeled ordinal data yields a learner better than the traditional learner or not – shortly, be it semisupervised or supervised. To answer this question, we compared the proposed SSOC algorithm with the existing ordinal classification algorithm presented in [1]. Table 6 shows the comparison results obtained when using a 50%–50% labeled-unlabeled ratio. It is clearly seen that the semisupervised ordinal classification methods (SSOC-DT, SSOC-SVM, SSOC-RF, and SSOC-NN) were exceeded their supervised ordinal counterparts (OC-DT, OC-SVM, OC-RF, and OC-NN) in terms of accuracy on average. The results presented in Table 6 are promising since they indicate that, in ordinal classification, unlabeled ordinal data can be employed by the machine learning algorithms to consistently improve performance. This is because of the fact that when unlabeled data is available besides the labeled ordinal data, semisupervised learning can better exploit and learn existing structures in the explanatory features than supervised learning would be. For instance, the SSOC-RF method (87.93%) remarkably outperformed the existing OC-RF algorithm (80.61%) on average. Similarly, the SSOC-NN method (83.02%) achieved better performance than the existing OC-NN algorithm (77.43%) on average. It experimentally confirmed that the introduction of unlabeled ordinal data can improve the generalization and so classification performances of any particular learning algorithm. Hence, our semi-supervised ordinal classification algorithms showed significant potential and competitiveness against supervised ordinal ones. Consequently, the experimental results provided good synergy for the ordinal version of semisupervised learning.

The obtained results are promising indeed because, in real-world applications, unlabeled ordinal data are more available than labeled ordinal data, and labeling ordinal data is a difficult, expensive, or time-consuming process and it usually requires human efforts. Therefore, it is usually possible to construct a proper classification model by labeling at most half of the ordinal instances, instead of all.

4.2.5. The results of experiment 5

The proposed SSOC method consists of two main steps. In the first step, a classifier is trained with the initial labeled data on an ordinal scale. In the second step, the unlabeled data is labeled by the constructed ordinal classifier via its predictions, called pseudo-labels, and then the classifier is retrained with this augmented data. In this experiment, we investigated that after retraining the classifier with new pseudo-labels, how much the performance is increased compared to the performance with initial training labels. To answer this question, we compared the performances of the initial model and retrained model.

Figure 4 shows the average accuracy values obtained from the initially trained model and retrained model for all the ordinal datasets considered when using 25%-75%, 50%-50%, and 75%-25% labeled-unlabeled ratios. It can be clearly seen from the results that the pseudo-labeled data can be employed by the algorithm to consistently improve performance. For example, when 50%-50% labeled-unlabeled ratio was considered, the SSOC-NN method achieved a higher accuracy value (83.02%) than the initial model (76.07%) on average. It experimentally confirmed that the introduction of unlabeled ordinal data can improve the generalization and so classification performances of any particular learning algorithm. This is because of the fact that when unlabeled data is available besides the labeled ordinal data, the algorithm can better exploit and learn existing structures in the explanatory features. The unlabeled data instances provide additional knowledge that is relevant for ordinal classification, and they can successfully be used to improve the generalization ability of the learning system. The best improvement (14.81%) was observed for the SSOC-RF algorithm when the labeled data rate is 75% probably due to its ensemble structure. Based on the results, we can conclude that the introduction of unlabeled ordinal data yields a learner better than the initial learner. Unlabeled data in addition to labeled ordinal data can help in improving a classifier in terms of accuracy.

Table 6. Comparison of the proposed method (SSOC) and the existing method [1] in terms of accuracy (%).

Dataset	SSOC-DT	SSOC-SVM	SSOC-KNN	SSOC-RF	SSOC-NN	OC-DT	OC-SVM	OC-KNN	OC-RF	OC-NN
Automobile	73.56	74.54	58.83	**87.80**	81.95	66.34	66.83	61.95	85.85	72.68
Car evaluation	86.79	80.31	82.35	96.06	98.62	90.16	80.32	94.27	95.72	**99.42**
ESL	68.57	63.81	66.68	**75.33**	74.18	65.78	66.80	68.24	66.60	68.85
Eucalyptus	66.85	70.22	57.04	**81.20**	79.08	65.49	66.03	54.48	62.77	63.32
Nursery	96.30	92.64	95.84	98.74	**99.98**	97.04	93.13	97.77	99.14	99.95
Squash-stored	69.23	77.31	62.31	81.54	**82.69**	69.23	71.15	53.85	67.31	65.38
UKM	95.53	83.13	84.42	**96.92**	96.87	92.56	93.55	85.86	93.55	95.29
Volcanoes	96.25	96.21	96.42	**98.17**	96.62	96.08	96.18	96.49	96.42	96.61
WQ-red	68.09	58.09	61.71	**81.78**	63.63	61.60	58.41	58.04	69.48	59.29
WQ-white	65.66	51.67	57.76	**81.76**	56.61	57.62	51.72	54.10	69.27	53.51
Avg.	78.68	74.79	72.34	**87.93**	83.02	76.19	74.41	72.51	80.61	77.43

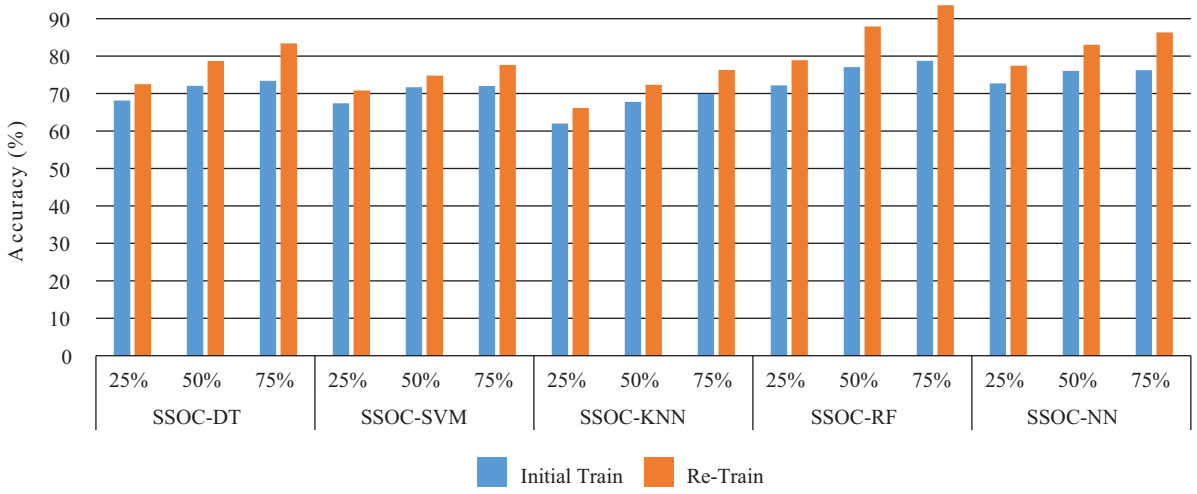


Figure 4. The performance improvement provided by retraining compared to the initial training.

As a result of five experiments aforementioned, it can be concluded that the proposed SSOC algorithm looks quite promising in achieving high accuracy for the ordinal version of semi-supervised learning. The experiments demonstrated that exploiting the order among class labels in semi-supervised learning could lead to building better models, compared to the traditional nominal and supervised learning.

5. Conclusion and future work

Ordinal classification differs from multiclass (nominal) classification in that there exists some ordering among the class labels such as low, medium, and high. The existing semisupervised methods provide a nominal classification task. The key issue for our study, thus, is to design a new method that can effectively combine "semisupervised learning" and "ordinal learning" paradigms for the categorical class labels for the first time. The purpose of our study is to effectively use a small number of labeled ordinal data and a large number of unlabeled ordinal data for classification model construction. From this perspective, our study is very important since unlabeled ordinal data is cheap and abundantly available, and labeling ordinal data is a time-consuming and costly process requiring expert knowledge. It aims to maximize the classification performance of the model through appending unlabeled samples to the labeled ordinal data while minimizing the human effort.

This paper introduces a new concept of "semi-supervised ordinal classification" for the categorical class labels. It proposes a new algorithm, called SSOC, which takes into account the relationships between the class labels during semi-supervised learning. In the proposed SSOC method, a classifier is firstly trained on an initial small number of labeled ordinal samples for the purpose of classifying unlabeled instances. After that, the resulting classifier is re-trained with its own estimations by enlarging its initial labeled set.

In the experimental studies, we evaluated the performance of the proposed algorithm (SSOC) on various ordinal datasets. The SSOC method resulted in a significant improvement over the existing YATSI method.

As future work, an ensemble semi-supervised ordinal classification approach can be implemented by using the proposed method as an ensemble member.

References

[1] Frank E, Hall M. A simple approach to ordinal classification. In: European Conference on Machine Learning; Freiburg, Germany; 2001. pp. 145-156.

[2] Driessens K, Reutemann P, Pfahringer B, Leschi C. Using weighted nearest neighbor to benefit from unlabeled data. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining; Singapore; 2006. pp. 60-69.

[3] Lang R, Lu R, Zhao C, Qin H, Liu G. Graph-based semi-supervised one class support vector machine for detecting abnormal lung sounds. Applied Mathematics and Computation 2020; 364: 1-10. doi: 10.1016/j.amc.2019.06.001

[4] Li W, Meng W, Au MH. Enhancing collaborative intrusion detection via disagreement-based semi-supervised learning in IoT environments. Journal of Network and Computer Applications 2020; 161: 1-9. doi: 10.1016/j.jnca.2020.102631

[5] Livieris IE, Drakopoulou K, Tampakas VT, Mikropoulos TA, Pintelas P. Predicting secondary school students' performance utilizing a semi-supervised learning approach. Journal of Educational Computing Research 2019; 57 (2): 448-470. doi: 10.1177/0735633117752614

[6] Xu P, Lu W, Wang B. A semi-supervised learning framework for gas chimney detection based on sparse autoencoder and TSVM. Journal of Geophysics and Engineering 2019; 16 (1): 52-61. doi:10.1093/jge/gxy004

[7] Stanescu A, Caragea D. An empirical study of ensemble-based semi-supervised learning approaches for imbalanced splice site datasets. BMC Systems Biology 2015; 9 (5): 1-12. doi: 10.1186/1752-0509-9-S5-S1

[8] Yang Y, Liu J, Xu S, Zhao Y. An extended semi-supervised regression approach with co-training and geographical weighted regression: a case study of housing prices in beijing. ISPRS International Journal of Geo-Information 2016; 5 (1): 1-12. doi:10.3390/ijgi5010004

[9] Wang X, Yang I, Ahn SH. Sample efficient home power anomaly detection in real time using semi-supervised learning. IEEE Access 2019; 7: 139712-139725. doi: 10.1109/ACCESS.2019.2943667

[10] Van Engelen JE, Hoos HH. A survey on semi-supervised learning. Machine Learning 2020; 109 (2): 373-440. doi: 10.1007/s10994-019-05855-6

[11] Dornaika F, Dahbi R, Bosaghzadeh A, Ruichek Y. Efficient dynamic graph construction for inductive semi-supervised learning. Neural Networks 2017; 94: 192-203. doi: 10.1016/j.neunet.2017.07.006

[12] Rahangdale A, Raut S. Clustering-based transductive semi-supervised learning for learning-to-rank. International Journal of Pattern Recognition and Artificial Intelligence 2019; 33 (12): 1-27. doi: 10.1142/S0218001419510078

[13] Sati N. A novel semisupervised classification method via membership and polyhedral conic functions. Turkish Journal of Electrical Engineering & Computer Sciences 2020; 28 (1): 80-92. doi:10.3906/elk-1905-45

[14] Chen K, Wang S. Semi-supervised learning via regularized boosting working on multiple semi-supervised assumptions. IEEE Transactions on Pattern Analysis and Machine Intelligence 2010; 33 (1): 129-143. doi: 10.1109/TPAMI.2010.92

[15] Gutierrez PA, Perez-Ortiz M, Sanchez-Monedero J, Fernandez-Navarro F, Hervas-Martinez C. Ordinal regression methods: survey and experimental study. IEEE Transactions on Knowledge and Data Engineering 2016; 28 (1): 127-146. doi: 10.1109/TKDE.2015.2457911

[16] Perez-Ortiz M, Gutierrez, PA, Hervas-Martinez C. Projection-based ensemble learning for ordinal regression. IEEE Transactions on Cybernetics 2014; 44 (5): 681-694. doi: 10.1109/TCYB.2013.2266336

[17] Sanchez-Monedero J, Gutierrez PA, Tino P, Hervas-Martinez C. Exploitation of pairwise class distances for ordinal classification. Neural Computation 2013; 25 (9): 2450-2485. doi: 10.1162/NECO_a_00478

[18] Perez-Ortiz M, Gutierrez PA, Carbonero-Ruz M, Hervas-Martinez C. Semi-supervised learning for ordinal kernel discriminant analysis. Neural Networks 2016; 84: 57-66. doi: 10.1016/j.neunet.2016.08.004

[19] Tsuchiya T, Charoenphakdee N, Sato I, Sugiyama M. Semi-supervised ordinal regression based on empirical risk minimization. Computing Research Repository 2019; arXiv:1901.11351.

[20] Cardoso JS, Domingues I. Max-coupled learning: application to breast cancer. In: 10th International Conference on Machine Learning and Applications; Honolulu, Hawaii, USA; 2011. pp. 13-18.

[21] Srijith PK, Shevade S, Sundararajan S. Semi-supervised Gaussian process ordinal regression. In: Blockeel H, Kersting K, Nijssen S, Zelezny F (editors). Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science. Berlin, Germany: Springer, 2013, pp. 144-159. doi: 10.1007/978-3-642-40994-3_10

[22] Witten IH, Frank E, Hall MA, Pal CJ. Data Mining: Practical Machine Learning Tools and Techniques. 4th ed. Cambridge, MA, USA: Morgan Kaufmann, 2016.

[23] Jiang D, Ma P, Su X, Wang T. Distance metric based divergent change bad smell detection and refactoring scheme analysis. International Journal of Innovative Computing, Information and Control 2014; 10 (4): 1519-1531.

[24] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic smaller over sampling technique. Journal of Artificial Intelligence Research 2002; 16: 321-357. doi: 10.1613/jair.953

[25] Lang S, Bravo-Marquez F, Beckham C, Hall M, Frank E. WekaDeeplearning4j: a deep learning package for Weka based on deepLearning4j. Knowledge-Based Systems 2019; 178: 48-50. doi: 10.1016/j.knosys.2019.04.013