Expert Systems with Applications 41 (2014) 4915–4928

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Learned lessons in credit card fraud detection from a practitioner perspective

Andrea Dal Pozzolo [a,*], Olivier Caelen [b], Yann-Aël Le Borgne [a], Serge Waterschoot [b], Gianluca Bontempi [a]

[a] Machine Learning Group, Computer Science Department, Faculty of Sciences ULB, Université Libre de Bruxelles, Brussels, Belgium
[b] Fraud Risk Management Analytics, Worldline, Brussels, Belgium

ARTICLE INFO

Keywords:
Incremental learning
Unbalanced data
Fraud detection

ABSTRACT

Billions of dollars of loss are caused every year due to fraudulent credit card transactions. The design of efficient fraud detection algorithms is key for reducing these losses, and more algorithms rely on advanced machine learning techniques to assist fraud investigators. The design of fraud detection algorithms is however particularly challenging due to non-stationary distribution of the data, highly imbalanced classes distributions and continuous streams of transactions.

At the same time public data are scarcely available for confidentiality issues, leaving unanswered many questions about which is the best strategy to deal with them.

In this paper we provide some answers from the practitioner's perspective by focusing on three crucial issues: unbalancedness, non-stationarity and assessment. The analysis is made possible by a real credit card dataset provided by our industrial partner.

1. Introduction

Nowadays, enterprises and public institutions have to face a growing presence of fraud initiatives and need automatic systems to implement fraud detection (Delamaire, Abdou, & Pointon, 2009). Automatic systems are essential since it is not always possible or easy for a human analyst to detect fraudulent patterns in transaction datasets, often characterized by a large number of samples, many dimensions and online updates. Also, the cardholder is not reliable in reporting the theft, loss or fraudulent use of a card (Pavía, Veres-Ferrer, & Foix-Escura, 2012). Since the number of fraudulent transactions is much smaller than the legitimate ones, the data distribution is unbalanced, i.e. skewed towards non-fraudulent observations. It is well known that many learning algorithms underperform when used for unbalanced dataset (Japkowicz & Stephen, 2002) and methods (e.g. resampling) have been proposed to improve their performances. Unbalancedness is not the only factor that determines the difficulty of a classification/detection task. Another influential factor is the amount of overlapping of the classes of interest due to limited information that transaction records provide about the nature of the process (Holte, Acker, & Porter, 1989).

Detection problems are typically addressed in two different ways. In the static learning setting, a detection model is periodically relearnt from scratch (e.g. once a year or month). In the online learning setting, the detection model is updated as soon as new data arrives. Though this strategy is the most adequate to deal with issues of non stationarity (e.g. due to the evolution of the spending behavior of the regular card holder or the fraudster), little attention has been devoted in the literature to the unbalanced problem in changing environment.

Another problematic issue in credit card detection is the scarcity of available data due to confidentiality issues that give little chance to the community to share real datasets and assess existing techniques.

2. Contributions

This paper aims at making an experimental comparison of several state of the art algorithms and modeling techniques on one real dataset, focusing in particular on some open questions like: Which machine learning algorithm should be used? Is it enough to learn a model once a month or it is necessary to update the model everyday? How many transactions are sufficient to train the model? Should the data be analyzed in their original unbalanced form? If not, which is the best way to rebalance them? Which performance measure is the most adequate to asses results?

In this paper we address these questions with the aim of assessing their importance on real data and from a practitioner

* Corresponding author. Tel.: +32 2 650 55 94.
E-mail addresses: adalpozz@ulb.ac.be (A. Dal Pozzolo), olivier.caelen@worldline.com (O. Caelen), yleborgn@ulb.ac.be (Y.-A. Le Borgne), serge.waterschoot@worldline.com (S. Waterschoot), gbonte@ulb.ac.be (G. Bontempi).

http://dx.doi.org/10.1016/j.eswa.2014.02.026
0957-4174/

perspective. These are just some of potential questions that could raise during the design of a detection system. We do not claim to be able to give a definite answer to the problem, but we hope to that our work serves as guideline for other people in the field. Our goal is to show what worked and what did not in a real case study. In this paper we give a formalisation of the learning problem in the context of credit card fraud detection. We present a way to create new features in the datasets that can trace the card holder spending habits. By doing this it is possible to present the transactions to the learning algorithm without providing the card holder identifier. We then argue that traditional classification metrics are not suited for a detection task and present existing alternative measures.

We propose and compare three approaches for online learning in order to identify what is important to retain or to forget in a changing and non-stationary environment. We show the impact of the rebalancing technique on the final performance when the class distribution is skewed. In doing this we merge techniques developed for unbalanced static datasets with online learning strategies. The resulting frameworks are able to deal with unbalanced and evolving data streams. All the results are obtained by experimentation on a dataset of real credit card transactions provided by our industrial partner.

3. State of the art in credit card fraud detection

Credit card fraud detection is one of the most explored domains of fraud detection (Chan, Fan, Prodromidis, & Stolfo, 1999; Bolton & Hand, 2001; Brause, Langsdorf, & Hepp, 1999) and relies on the automatic analysis of recorded transactions to detect fraudulent behavior. Every time a credit card is used, transaction data, composed of a number of attributes (e.g. credit card identifier, transaction date, recipient, amount of the transaction), are stored in the databases of the service provider.

However a single transaction information is typically not sufficient to detect a fraud occurrence (Bolton & Hand, 2001) and the analysis has to consider aggregate measures like total spent per day, transaction number per week or average amount of a transaction (Whitrow, Hand, Juszczak, Weston, & Adams, 2009).

3.1. Supervised versus unsupervised detection

In the fraud detection literature we encounter both supervised techniques that make use of the class of the transaction (e.g. genuine or fraudulent) and unsupervised techniques. Supervised methods assume that labels of past transactions are available and reliable but are often limited to recognize fraud patterns that have already occurred (Bolton & Hand, 2002). On the other hand, unsupervised methods do not use the class of transactions and are capable of detecting new fraudulent behaviours (Bolton & Hand, 2001). Clustering based methods (Quah & Sriganesh, 2008; Weston, Hand, Adams, Whitrow, & Juszczak, 2008) form customer profiles to identify new hidden fraud patterns.

The focus of this paper will be on supervised methods. In the literature several supervised methods have been applied to fraud detection such as Neural networks (Dorronsoro, Ginel, Sgnchez, & Cruz, 1997), Rule-based methods (BAYES Clark & Niblett, 1989, RIPPER Cohen, 1995) and tree-based algorithms (C4.5 Quinlan, 1993 and CART Olshen & Stone, 1984). It is well known however that an open issue is how to manage unbalanced class sizes since the legitimate transactions generally far outnumber the fraudulent ones.

3.2. Unbalanced problem

Learning from unbalanced datasets is a difficult task since most learning systems are not designed to cope with a large difference between the number of cases belonging to each class (Batista, Carvalho, & Monard, 2000). In the literature, traditional methods for classification with unbalanced datasets rely on sampling techniques to balance the dataset (Japkowicz & Stephen, 2002).

In particular we can distinguish between methods that operates at the data and algorithmic levels (Chawla, Japkowicz, & Kotcz, 2004). At the data level, balancing techniques are used as a pre-processing step to rebalance the dataset or to remove the noise between the two classes, before any algorithm is applied. At the algorithmic level, the classification algorithms themselves are adapted to deal with the minority class detection. In this article we focus on data level techniques as they have the advantage of leaving the algorithms unchanged.

Sampling techniques do not take into consideration any specific information in removing or adding observations from one class, yet they are easy to implement and to understand. *Undersampling* (Drummond & Holte, 2003) consists in down-sizing the majority class by removing observations at random until the dataset is balanced.

SMOTE (Chawla, Bowyer, Hall, & Kegelmeyer, 2011) over-samples the minority class by generating synthetic minority examples in the neighborhood of observed ones. The idea is to form new minority examples by interpolating between examples of the same class. This has the effect of creating clusters around each minority observation.

Ensemble methods combine balancing techniques with a classifier to explore the majority and minority class distribution. *EasyEnsemble* is claimed in Liu, Wu, and Zhou (2009) to be better alternative to undersampling. This method learns different aspects of the original majority class in an unsupervised manner. This is done by creating different balanced training sets by *Undersampling*, learning a model for each dataset and then combining all predictions.

3.3. Incremental learning

Static learning is the classical learning setting where the data are processed all at once in a single learning batch. *Incremental learning* instead interprets data as a continuous stream and processes each new instance "on arrival" (Oza, 2005). In this context it is important to preserve the previously acquired knowledge as well as to update it properly in front of new observations. In incremental learning data arrives in chunks where the underlying data generation function may change, while static learning deals with a single dataset. The problem of learning in the case of unbalanced data has been widely explored in the static learning setting (Chawla et al., 2011; Drummond & Holte, 2003; Japkowicz & Stephen, 2002; Liu et al., 2009). Learning from non-stationary data stream with skewed class distribution is however a relatively recent domain.

In the incremental setting, when the data distribution changes, it is important to learn from new observations while retaining existing knowledge form past observations. Concepts learnt in the past may re-occur in the future as new concepts may appear in the data stream. This is known as the stability-plasticity dilemma (Grossberg, 1988). A classifier is required to be able to respond to changes in the data distribution, while ensuring that it still retains relevant past knowledge. Many of the techniques proposed (Chen, He, Li, & Desai, 2010; Polikar, Upda, Upda, & Honavar, 2001; Street & Kim, 2001) use ensemble classifiers in order to combine what is learnt from new observations and the knowledge acquired before. As fraud evolves over time, the learning framework has to adapt to the new distribution. The classifier should be able