Legal Knowledge and Information Systems
E. Francesconi et al. (Eds.)

doi:10.3233/FAIA220463

A Compression and Simulation-Based Approach to Fraud Discovery

Peter FRATRIČ [a,1], Giovanni SILENO [a], Tom VAN ENGERS [a,b] and Sander KLOUS [a]
[a] *Informatics Institute, University of Amsterdam, the Netherlands*
[b] *Leibniz Institute, TNO/University of Amsterdam, the Netherlands*

Abstract. With the uptake of digital services in public and private sectors, the formalization of laws is attracting increasing attention. Yet, non-compliant fraudulent behaviours (money laundering, tax evasion, etc.)—practical realizations of violations of law—remain very difficult to formalize, as one does not know the exact formal rules that define such violations. The present work introduces a methodological framework aiming to discover non-compliance through compressed representations of behaviour, considering a fraudulent agent that explores via simulation the space of possible non-compliant behaviours in a given social domain. The framework is founded on a combination of utility maximization and active learning. We illustrate its application on a simple social domain. The results are promising, and seemingly reduce the gap on fundamental questions in AI and Law, although this comes at the cost of developing complex models of the simulation environment, and sophisticated reasoning models of the fraudulent agent.

Keywords. fraud discovery, non-compliance detection, active learning, agent-based modelling, simulation, behavioural exploration

1. Introduction

Formalizing legislation into machine-readable artefacts is a traditional track of research in law and computer science [2]. However, the discussion on how representing and processing normative directives generally obfuscates a more fundamental problem of normative reasoning. In law, we often encounter rules such as: *Any person who willfully attempts in any manner to evade or defeat any tax imposed by this title or the payment thereof shall, in addition to other penalties provided by law, be guilty of a felony.* This rule, defining tax evasion in the United States, does not say anything about what types of behaviour can be deemed to be attempts to evade taxes. Yet, it implicitly assumes that any felony will consist of a sequence of actions, and refers (without defining it concretely) to some set of action sequences that are relevant to qualify or disqualify a certain behaviour as tax evasion. In this paper, we will focus on the problem of addressing these rules with implicit behavioural definitions (*implicit rules* for short), in particular concerning qualifications of *non-compliant* behaviour. This problem partially overlaps with the traditional *case-based reasoning* research track in AI & Law [1], aiming to reconstruct the structure

[1]Corresponding Author: Peter Fratrič, p.fratric@uva.nl. This work was partly funded by the Dutch Research Council (NWO) for the HUMAINER AI project (KIVI.2019.006).

of rationale behind case decisions, typically identifying relevant factors and their relative contributions to the conclusion; however, the "behavioural definition" issue studied here focuses primarily on capturing legally relevant behavioural *scripts*, rather than relevant contextual factors: the temporal sequence of actions will play the major role.

Two general modelling approaches can be identified (see eg. [9]): *rule based*, in which an expert identifies a set of rules that indicate evidence likely to be related to non-compliant activity (see eg. [8]); and *machine-learning based*, where a dataset of evidence related to usually both compliant or non-compliant activity is used to train a non-compliance classifier over the entire behavioural space [4]. Unfortunately, sample datasets of fraudulent behaviour suffer from class imbalance; there is only a relatively small amount of labeled instances of non-compliance compared to labeled instances of compliance. To face this issue, recent contributions have proposed to use either *inductive* (using both labeled and unlabeled instances in the training process) [3] or *transductive* (building upon local similarities among data points) [6] *semi-supervised learning* for non-compliance classification.

We propose a research direction that aims to combine recent trends into one simulation framework, where instances of non-compliance can be generated [5]. We will posit and elaborate on the following arguments: (a) the definition of what is non-compliance can be seen as a *compression* task on possible, relevant behaviours; (b) *tracking* of (intentional) non-compliance can be based on defining sound constraints and preferences (eg. expressed as pay-offs); (c) automated exploration of the behavioural space can be performed by means of *simulation*. We also observe that, despite the help of computational tools, (d) the role of human experts in directing the search and determining the legal status of the generated action sequence remains crucial.

The paper is organized as follows. Section 2 presents relevant concepts and the proposed method: the task of constructing implicit definitions of non-compliant behaviour as a combination of utility maximization and an active learning component. Section 3 presents an illustrative example. The paper ends with a note on future work.

2. Method

Simulation Environment Let $A = \Gamma \cup \Gamma^c$ be the space of all possible behaviour consisting of sequences of elementary actions. Consider a simulation environment where the agents generate action sequences, such that each instance $a \in A$ is a finite sequence of actions $(a_1, ..., a_n)$ generated by an agent observing a sequence of states $(s_0, s_1, ..., s_n)$ (as in standard Markov-decision process formalisms). The actions available to the agent may be constrained by rules that represent *hard constraints* on the action space. These may be for instance physical constraints, or explicit legal constraints that would cause non-compliance if violated. The advantage of dealing with economic crime is that a monetary gain motivating the non-compliant behaviour is usually present, which means we can assume a utility function $u : A \to \mathbb{R}$ evaluating the quality of the action sequence.[2]

[2]Not all non-compliance can be quantified by monetary gain, or even well-defined utility function might not be available. One can relax this assumption by considering more general preferential structure.

Locally optimal forms Under hard constraints of the action space, the fraudulent agent can be seen as optimizing the utility function to discover sequences that yield high utility. Fraud schemes as such do not relate only to a specific type of behaviour, but to a class of behaviours that follows a certain higher-level behavioural pattern. Intuitively, one can expect that each scheme might have its representative form, that is, a form that illustrates the essence of a particular fraud scheme. With these representative forms, one does not need to list all the instances of Γ, achieving a compression of possibly infinite set Γ into a finite number of classes. We speak of sequences that attain a local maximum as *locally optimal forms*.

Exploration We consider as given an initial dataset D consisting of behavioural instances labeled as fraudulent, non-fraudulent or unknown. These instances can be organized into a graph $\mathscr{G}$, where each edge is weighted by the similarity function d_Γ. This graph structure can then be extended by an artificial *fraudulent agent*, which is an entity (standing for an individual agent, or a group or coalition of coordinated agents) capable of generating fraudulent behaviour in the simulation environment, following a certain rationality (eg. utility maximization), such that most, if not all, locally optimal forms are discovered.

Querying the oracle and learning The fraudulent agent is generating action sequences with high utility, but since implicit rules are not formalized, the only way how to know if an action sequence $a \in A$ violates them is to query an *oracle* $Q : A \to \{-1, 1\}$. A prototypical oracle would be a human legal analyst with the competence to pass a judgement on the input behavioural instance.

Since the number of queries of the oracle is relatively small compared to all possible sequences that can be generated by the agent, one needs to choose queried sequences wisely, such that maximal information gain is obtained by the query. A selection rule R_D is applied to select which unlabelled instances of $\mathscr{G}$ are likely to provide the most relevant information for the compressed representation, as a strategy for active learning. The chosen instances are then labelled by the oracle, and used to find locally optimal forms of Γ by utilizing local similarities. The compressed representation can be used as a classifier, by checking the membership of the instance to the compressed set Γ.

3. Illustrative example

Tracing the boundary between tax planning, tax avoidance, and the role of tax havens is recognized to be a debated topic both in the academic literature and in policy circles [7], and offers therefore a prototypical domain of application of the proposed method. In our example environment, a fraudulent agent is used to generate instances of behaviour. The system then aims to compress the fraudulent instances. The aim of the example is to illustrate the proposed framework, and to show how different choice of the selection rule R_D are influencing the search process of the true compressed representation Γ.

Action space and constraints Suppose organizations are able to move certain assets (eg. goods, capital, data...) from a place to another, and can decide how to act depending on the economic payoff (eg. transactional cost, income) and the norms in place. Let us represent the movement of a single asset of a single organization as a transaction system. Places fall in one of four categories denoted by letters $T = \{r, g, b, y\}$ (standing for red,

green, blue, and yellow). Elementary (moving) actions are pairs belonging to $T \times T$. Finite sequences of these actions form the action space A. Viewing the sequence as a string, suppose that there are three hard constraints (derived from a partially formalized legal system) restricting A only to strings that do not contain gr, bg and ry as substrings (ie. moving the asset from green to red is not allowed, etc.). Moreover, the string rb cannot occur more than twice as a substring (ie. moving the asset twice from red to blue is not allowed).

Oracle To make the example easier to work with, the oracle is not a human as it would be in a practical setting. The oracle is defined by the ability to decide (non)compliance, and this decision is made by applying a regular expression `g[by]+g`. Any $a \in A$ that is matching the regular expression is regarded as a violation of implicit rule, which could be thought eg. as a known scenario of non-compliance by analysts in that social domain.

Utility function Consider an agent moving a certain capital $u_0 = 1000$ of assets from a place G to a place H, acting as initial and terminal places, with both of them being of type g. Each movement incurs a cost (eg. a tax) given by a transaction table M with values listed in the following table:

	r	g	b	y
r	0.050	0.05	-0.00525 (first time), ∞ (others)	∞
g	∞	0.40	0.001	0.40
b	0.005	∞	0.10	0.00
y	0.05	0.0001	0.40	0.05

Hard constraints, such as gr, can be conveniently represented in the table as transactions with infinite cost. Note that rb has a negative cost, ie. the agent derives a benefit; we also hard-constrain it to be applied only once (otherwise all action sequences with high utility would be only claiming benefits). The value of the asset is updated depending on the transaction as $u_{t+1} = u_t(1 - M_{i,j})$, where $i, j \in T$. The agent aims to find such a sequence of transactions that maximize the value of u_n.

(Non)compliance knowledge base and compression The data sample D forms a graph as the one illustrated in Figure 1. A vertex $x \in \Gamma$ is linked to an unlabelled vertex y if and only if $d_\Gamma(x,y) \leq \tau$, where d_Γ is Levenshtein distance (most commonly used string edit distance). The query strategy $R_D(u, \mathscr{G})$, necessary for active learning of the compressed representation of Γ, is evaluated on four different strategies by always querying either (i) **random**: a random unlabelled instance; (ii) **max utility**: an unlabelled instance with the highest utility value; (iii) **max degree**: an unlabelled instance with the highest node degree in the graph; (iv) **min degree**: an unlabelled instance with the lowest node degree in the graph.

Once the knowledge base of the possible behavioural instances is updated, the system performs a search (eg. brute force) of a regular expression with a maximal fixed length (eg. 10) over a hypothesis space given by the alphabet $\{r, g, b, y, [,], +, *\}$ to obtain a classifier expression that maximizes accuracy of classification. The accuracy is determined by oracle, that has the access to true labels of all instances.[3]

[3] The oracle here is used out of convenience, as it is possible to evaluate the true accuracy of a compression model. In practice, this evaluation would be done in a more standard way by splitting the labelled data into a training set and a test set.

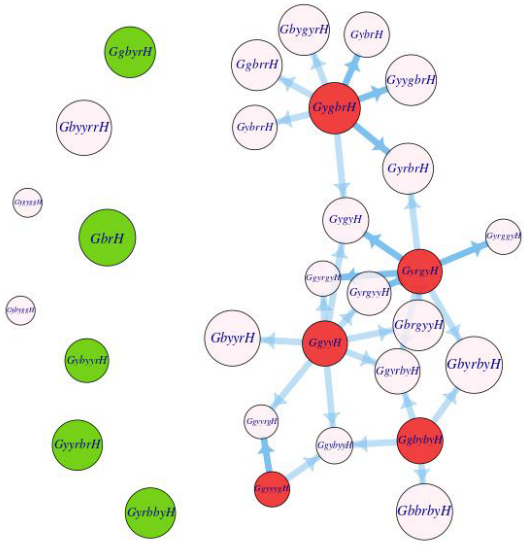

Figure 1. Vertexes coloured in green are compliant behaviours, while the red are non-compliant. The other vertexes are unlabelled. The bigger the size of the vertex, the greater the utility value of the sequence. Edge transparency depends on Levenshtein distance between the instances of behaviour.

Fraud compression results Let us assume that the fraudulent agent has generated all action sequences of length at most 7 with starting address G and ending address H that satisfy the hard constraints. Out of these 345 sequences, only one has a label at the start. This knowledge is meant only to facilitate the bootstrapping of the method. Then we can

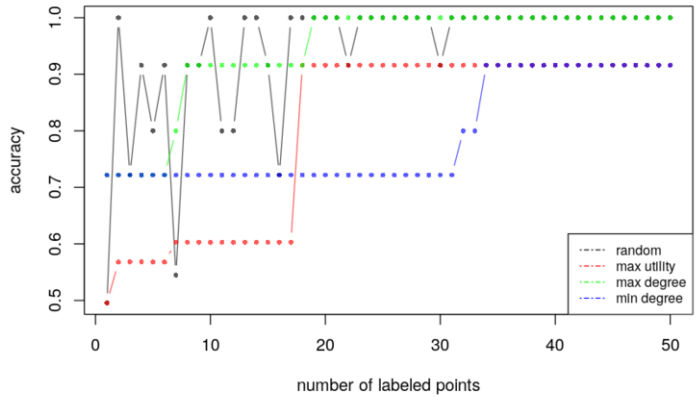

Figure 2. Min degree and **max utility** decision rules fail to converge to true expression with less than 50 oracle queries, which means the rules are biased for the less informative properties of the instance. The **random** decision rule converges, but seems to be very volatile, as instances are queried at random. The most suitable decision rule looks to be the **max degree** rule, where the agent aims to acquire labels from the oracle on instances that are most similar to already known non-compliant instances.

update the knowledge graph by querying the oracle.

Each query provides more information for the inference of the compression model. After a sufficient amount of queries, the system finds the optimal compression expression `g[by]+g`. For this compressed representation, the locally optimal form is *gbyg*. It is easy to see that any extension of this sequence that is still a member of Γ will have lower utility. For example, the non-compliant instance *gbygbyg* has the second-highest utility.

On Figure 2 one can observe how the classification accuracy of the classifier expression is separating the hard constrained space A into Γ and Γ^c. As the fraudulent agent is obtaining more knowledge by querying the oracle more times, the results converge to 100% accuracy for two out of four decision rules considered.

4. Perspectives

The theoretical considerations presented in this study provide initial foundations for a more structured approach to non-compliance detection via compression. In combination with active learning and graph-based semi-supervised learning, the framework is capable to discover compressed representations of non-compliant space, that can be later integrated into the formal legal system. The goal of the future research is to improve scalability of the framework by considering more efficient, but still explainable, compression methods over the non-compliant space, and more sophisticated models of the fraudulent agent.

References

[1] Ashley, K.D.: Case-based reasoning and its implications for legal expert systems. Artificial Intelligence and Law **1**(2-3), 113–208 (1992)

[2] Bench-Capon, T.J., Coenen, F.P.: Isomorphism and legal knowledge based systems. Artificial Intelligence and Law **1**(1), 65–86 (1992)

[3] van Engelen, J.E., Hoos, H.H.: A survey on semi-supervised learning. Machine Learning **109**(2), 373–440 (feb 2020)

[4] Fursov, I., Morozov, M., Kaploukhaya, N., Kovtun, E., Rivera-Castro, R., Gusev, G., Babaev, D., Kireev, I., Zaytsev, A., Burnaev, E.: Adversarial Attacks on Deep Models for Financial Transaction Records. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. vol. 1, pp. 2868–2878. ACM, New York, NY, USA (aug 2021)

[5] Hemberg, E., Rosen, J., Warner, G., Wijesinghe, S., O'Reilly, U.M.: Detecting tax evasion: a co-evolutionary approach. Artificial Intelligence and Law **24**(2), 149–182 (2016)

[6] Lebichot, B., Braun, F., Caelen, O., Saerens, M.: A graph-based, semi-supervised, credit card fraud detection system. In: Int. Workshop on Complex Networks and their Applications. pp. 721–733 (2016)

[7] Merks, P.: Tax evasion, tax avoidance and tax planning. Intertax pp. 272–281 (2006)

[8] Sileno, G., Boer, A., van Engers, T.: Reading agendas between the lines, an exercise. Artificial Intelligence and Law **25**(1), 89–106 (2017)

[9] Zhu, X., Ao, X., Qin, Z., Chang, Y., Liu, Y., He, Q., Li, J.: Intelligent financial fraud detection practices in post-pandemic era. The Innovation **2**(4), 100176 (2021)