High Quality True-Positive Prediction for Fiscal Fraud Detection

S. Basta, F. Fassetti, F. Giannotti, M. Guarascio. G. Manco
G. Papi, D. Pedreschi, S. Pisani, L. Spinsanti
Outline

- Scenario and Motivation
- DIVA Overview
  - Solution Proposed
  - Scoring Criteria
  - Multi-purpose objectives
- Sniper Core
  - Generating Rule
  - Merging Rule
- Evaluation
- Conclusion
The Context: VAT frauds in Italy

- DIVA - A joint initiative involving academic researchers, experts on fiscal laws, IT Professionals
- Main objective:
  - To tackle the VAT Fraud Detection issue raised by the credit mechanism via the adoption of data mining techniques.
Scenario

- Several challenges, both from a scientific and a practical point of view:
  - Sample selection bias
    - Audited subjects are not randomly chosen
    - Highly skewed data
      - Positive subjects larger than non-defrauders in audit data
  - Imprecise settings
    - Inaccurate, incomplete, and irrelevant data attributes
  - Only 0.004% of population audited
Motivation

- Classical approaches to the problem of fraud detection are not very effective:
  - Rule-Based classifiers are preferable for interpretability, but
    - Poor predictive accuracy in highly imprecise learning settings
    - Class-imbalance problem
  - Cost-sensitive classification and meta-learning approaches suffer from low interpretability
The proposal: Sniper as a meta-learner

- The core of the Sniper technique is the extraction of a binary rule-based classifier able to identify X topmost defrauders.
- Based on the combined use of local models and the definition of multi-objective functions.
DIVA Overview

- The data made available by the agency consisted of about 34 million VAT declarations spread over 5 years.
- Data contain general ‘demographic’ information, plus specific information about VAT declarations.
- As a result of a data understanding process conducted jointly with domain experts, we chose a total of 135 such features and 45,442 audited subjects.
Scoring individuals

- A multi-purpose modeling strategy, aiming at characterizing the exceptionalness and interestingness of an individual
  - **PROFITABILITY:** The amount of VAT fraud
    - The higher, the better
  - **EQUITY**
    - Low amounts do not necessarily correspond to meaningless fraudsters. The amount of fraud is relevant related to their business volume (1.000eur on 10.000eur is better than 1.000eur on 100.000eur)
  - **EFFICIENCY**
    - Scoring and detection should be sensitive to total/partial frauds (underclaring 200eur declaring 2.000eur is less significant than underclaring 200eur declaring 200eur)
Issues

- Need to face a trade-off among profitability, equity and efficiency
  - Solution: a combination of baseline functions
  - AND, OR, FUZZY_AND, FUZZY_OR
The Fuzzy combination

- Two different objective functions, four main classes

\[ \mathcal{F}_\Pi(o) = \prod_{i \in [1,k]} (\mathcal{N}(f_i(o)))^{p_i} \]

\[ \mathcal{F}_\Sigma(o) = \sum_{i \in [1,k]} p_i \cdot \mathcal{N}(f_i(o)) \]

Score function results

- Subject partitioning:
  - 48.65%
  - 25.67%
  - 17.97%
  - 7.70%

- Retrieved fraud:
  - 84.69%
  - 11.42%
  - 2.97%
  - 0.94%
Generating rules

- Sniper builds a hybrid classifier, resulting from the combination of the whole set of classifiers trained over the training set

- Advantages:
  - Separate model construction from model selection
  - Model construction
    - Several different strategies are attempted to build models focused on local peculiarities of the top class
  - Model selection
    - Several local fragments can be selected or discarded if the global accuracy improves
Merging Rules

- A candidate ruleset $R$ is obtained by merging all the rules returned by $h$ classifiers modeling the top class.

$$R = \left\{ r \in \bigcup_{i \in [1,h]} R_i \mid r.class = top \right\}$$

- $R$ still represents a classifier, and class $top$ is assigned to a non-labeled object $o$ if and only if there exists at least a rule in $R$ that activates it.

- The model is distilled from $R$ by selecting accurate rules, and removing inaccurate rules from $R$ in a principled (confidence-based) way.
Building Ruleset

Why we cannot just collect all the “good” rules from our classifiers?

\[ \text{conf}_{\text{min}} = 0.8 \]

Rule 1: sup=20 conf=0.9

well classified: 18  misclass: 2

Rule 2: sup=20 conf=0.8

well classified: 16  misclass: 4
Building Ruleset

- Why we cannot just collect all the “good” rules from our classifiers?

\[ \text{conf}_{\text{min}} = 0.8 \]

Rule 1: sup=20  conf=0.9

- well classified: 18
- misclass: 2

Rule 2: sup=20  conf=0.8

- well classified: 16
- misclass: 4

Subset!!
Building Ruleset

- Why we cannot just collect all the “good” rules from our classifiers?

\[ \text{conf}_{\text{min}} = 0.8 \]

Rule 1 AND 2: sup=24 conf=0.75
Merging Rules

**Input:** A set of non-exclusive positive rules $\mathcal{R}$, a confidence threshold $\gamma_{\text{min}}$, an integer $X$

**Output:** A model $\mathcal{M}$

**Method:**
1. $\mathcal{M} := \emptyset$
2. $\mathcal{R} := \left\{ r \in \mathcal{R} \mid \gamma(r) \geq \gamma_{\text{min}} \right\}$
3. while $\mathcal{R} \neq \emptyset$ do //first stop condition
   4. $r^* := \arg \max_{r \in \mathcal{R}} \left\{ \gamma(r) \right\}$ //select the best rule
   5. $\mathcal{M} := \mathcal{M} \cup \left\{ r^* \right\}$ //update the current model
   6. if $\mathcal{M}(D) \geq X$ then //second stop condition
      7. return $\mathcal{M}$
8. $\mathcal{R}$ is updated by removing $r^*$ and by replacing each rule $r$ other than $r^*$ with the rule $r'$ if $\gamma(r') = \gamma_{\text{min}}$, otherwise $r$ is just removed from $\mathcal{R}$
9. return $\mathcal{M}$
Merging Rules: Example

- Assume $\gamma_{\text{min}} = 60%$
- Initially, $R = \{R1,R2,R3,R4,R5\}, M = \{\}$

<table>
<thead>
<tr>
<th>Rule_ID</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>87,50%</td>
</tr>
<tr>
<td>R2</td>
<td>75%</td>
</tr>
<tr>
<td>R3</td>
<td>71,4%</td>
</tr>
<tr>
<td>R4</td>
<td>60%</td>
</tr>
<tr>
<td>R5</td>
<td>58,30%</td>
</tr>
</tbody>
</table>

- Positive Example
- Negative Example
Merging Rules: Example

\[ R = \{R2, R3, R4, R5\}, M = \{R1\} \]

<table>
<thead>
<tr>
<th>Rule_ID</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>R2</td>
<td>66,6%</td>
</tr>
<tr>
<td>R3</td>
<td>75%</td>
</tr>
<tr>
<td>R4</td>
<td>60%</td>
</tr>
<tr>
<td>R5</td>
<td>50%</td>
</tr>
</tbody>
</table>

- Positive Example
- Negative Example
Merging Rules: Example

- \( R = \{R2, R4, R5\}, \ M = \{R1, R3\} \)

<table>
<thead>
<tr>
<th>Rule_ID</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>R2</td>
<td>66.6%</td>
</tr>
<tr>
<td>R4</td>
<td>50%</td>
</tr>
<tr>
<td>R5</td>
<td>42.8%</td>
</tr>
</tbody>
</table>

- Positive Example
- Negative Example
Merging Rules: Example

- $R = \{R4, R5\}, M = \{R1, R3, R2\}$

<table>
<thead>
<tr>
<th>Rule_ID</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>R4</td>
<td>50%</td>
</tr>
<tr>
<td>R5</td>
<td>25%</td>
</tr>
</tbody>
</table>

- Positive Example
- Negative Example
Evaluation

We compared the results obtained from a single classifier against those obtained by Sniper in terms of confidence and support of the rules generated.

<table>
<thead>
<tr>
<th>classifier</th>
<th>supp (%)</th>
<th>conf (%)</th>
<th>dataset subjects</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_1$</td>
<td>1.01</td>
<td>84.90</td>
<td>1,910</td>
</tr>
<tr>
<td>$C_2$</td>
<td>1.10</td>
<td>82.97</td>
<td>2,240</td>
</tr>
<tr>
<td>$C_3$</td>
<td>3.11</td>
<td>77.28</td>
<td>4,955</td>
</tr>
<tr>
<td>$C_4$</td>
<td>3.44</td>
<td>77.12</td>
<td>5,675</td>
</tr>
<tr>
<td>$C_5^*$</td>
<td>6.36</td>
<td>62.26</td>
<td>10,056</td>
</tr>
<tr>
<td>$C_6^*$</td>
<td>6.81</td>
<td>60.80</td>
<td>8,875</td>
</tr>
<tr>
<td>$C_7^*$</td>
<td>7.07</td>
<td>59.72</td>
<td>9,059</td>
</tr>
<tr>
<td>$C_8^*$</td>
<td>5.22</td>
<td>52.64</td>
<td>9,950</td>
</tr>
<tr>
<td>$C_9^*$</td>
<td>4.56</td>
<td>49.18</td>
<td>12,584</td>
</tr>
<tr>
<td>$S$</td>
<td>8.78</td>
<td>80.41</td>
<td>9,840</td>
</tr>
</tbody>
</table>
(Partial) Results

- 1475 subjects identified
  - 276 subjects audited (feb-2010)
    - 147 in class 3 (53.26%)

Mean Values:
- Proficiency: 77.514,14
- Equity: 32.5738
- Efficiency: 0.4252