ASIA - An Investigation Platform for Exploiting Open Source Information in the Fight against Tax Evasion

Clara Bacciu[1], Fabio Valsecchi[1], Matteo Abrate[1], Maurizio Tesconi[1], Andrea Marchetti[1]
[1]*Institute of Informatics and Telematics, CNR, Pisa, Italy*
{*firstname.lastname*}*@iit.cnr.it*

Keywords: Open Source Intelligence, Information Extraction, Web Search, Tax Evasion

Abstract: Tax evasion is a widespread phenomenon confirmed by numerous European and American reports. To contrast it, governments already adopt software solutions that support tax inspectors in their investigations. However, the currently existing systems do not normally take advantage of the constant stream of data published on the Web. Instead, the ASIA project aims to prove the effectiveness of combining this kind of open source information with official data contained in Public Administration archives to fight tax evasion. Our prototype platform deals with two cases of investigation, people and businesses. Public officers have been involved throughout the project, and took part in a preliminary test phase which showed very promising results.

1 INTRODUCTION

Tax evasion is a widespread phenomenon. In fact, U.S. unreported incomes of 2009 are estimated to range between 390 and 537 billion dollars (Feige and Cebula, 2011). In 2010, the Italian National Statistics Institute (ISTAT) declared that the Italian tax evasion value was between 255 and 275 billion euro (ISTAT, 2010). Governments try to combat the problem through education, punishment and prosecution (Ducke et al., 2010), sometimes by promoting the use of software platforms that support tax inspectors in finding evaders (TOSCA, 2010; Sogei, 2010). However, these solutions usually require very costly, manual operations and rely on official, closed-source information[1] alone (e.g., civil registry, business registry, land registry records, energy bills, etc.).

The so-called *Open Source Information* (OSINF) is instead being widely exploited in the defense and intelligence fields (Johnson et al., 2007). The World Wide Web is a valuable source of OSINF, given the huge amount of people and businesses that every day publish information about their private life and commercial activities on websites, blogs, social networks, and so on. It is in fact estimated that about 10 new websites are published every second, more than 3 million blog posts are written every day (Internet Live Stats, 2015), and that in 2014 social media penetration was about 56% in North America, 40% in Europe, and between 42% and 54% in Italy (We Are Social Singapore, 2014). The term *Open Source Intelligence* (OSINT) is used to denote the retrieval, extraction and analysis of OSINF, as opposed to classified or closed sources, to acquire intelligence (Best, 2008). Our claim is that the founding principles of OSINT can be applied to fight tax evasion, thus providing help to address the general phenomenon of *shadow economy*. In fact, OSINF includes many advertised off-the-books activities that may lead directly to suspicious cases, and it is also valuable for acquiring knowledge about the context of the investigation.

In this article, we present a work-in-progress platform that exploits OSINF fetched from the Web to feed automatic and semi-automatic analyses, in order to give tax inspectors the ability to find, visualize and interact with data relevant for their investigation. We describe two different architectures considering two investigation targets: *people* and *businesses*. Furthermore, we present two prototypes that implement our designs, and the involvement of users in some preliminary tests.

1.1 Related Work

OSINF is a valuable resource used by several commercial and free services. Spokeo[2] is a search engine that aggregates data such as white pages (i.e.,

[1]In the following, we refer to closed-source information by using the acronym CSINF.

[2]http://www.spokeo.com/

phone directories), public records, mailing lists and social network information in order to search and learn more about people. Entitycube[3] is a prototype that allows everyone to search people, locations and organizations presenting the results as a summary of the information contained in the web pages collected. The research community also exploits OSINF. For instance, CLUO (Maciołek and Dobrowolski, 2013) is a prototype system for extracting and analyzing large amounts of OSINF such as web pages, blog posts and social media updates. Other approaches are focused on gaining intelligence from OSINF. Some works propose techniques to deal with the prevention of organised crime (Aliprandi et al., 2014) or to support the intelligence operative structures (Neri and Geraci, 2009). Other studies (Yang and Lee, 2012) perform automatic processing of OSINF relying on text mining techniques for detecting *events*, valuable pieces of information for domains like national security, personal knowledge management and business intelligence.

However, to the best of our knowledge, even though there are several works concerning OSINT, none of them is explicitly targeted to address the tax evasion phenomenon.

2 DESIGN AND ARCHITECTURE

The general purpose of our work is to provide investigators with a platform capable of automatically combining two kinds of data: closed source information (CSINF), i.e., validated and authoritative data, and unofficial and informal OSINF (e.g., user generated content) retrieved from the Web.

Our approach is to define an *investigation pipeline*, described in Figure 1. OSINF and CSINF are searched and retrieved from the respective sources according to the query issued by the investigator. Then, data is automatically integrated and analysed (e.g., relevant entities such as addresses and fiscal codes can be extracted from text). The user can interact with this phase by validating and correcting the results, and then by issuing an update command to let the system run another round of analysis based on his new inputs. The last step is the presentation and visualization of the investigation results, through which the user can find clues, grasp insights and gain new knowledge about the target specified in its query. The investigator can also have an overview of the data, filter out some results and load more details.

[3]http://entitycube.research.microsoft.com/

2.1 Investigating people

A first architecture (Figure 2) is defined to tailor the pipeline concept to the specific goal of investigating individuals, i.e., natural people, while simplifying the traditional inquiry process adopted by investigators. Subjects of investigations are usually chosen because they officially report to have little or no income. The investigator should be able to learn which is the job of a specific subject, who are his known associates or family members, which are the places, the phone numbers, the nicknames on social networks, etc. associated to him. This information is important to let the investigator create a better profile of the suspect, complementing the data he already has from official archives, and possibly discovering an unreported commercial activity.

Thus, the system must make the user able to: (i) Issue a query about a certain person, starting an investigation; (ii) Understand which are the entities connected to that person; (iii) Learn that person's profession; (iv) Correct and update the priority with which the system shows the results; (v) Keep track of the performed investigations.

The proposed solution specialises the three steps of the investigation pipeline in the following way:

1. *Search and Retrieval.* This module retrieves a set of web pages related to a target person selected by the investigator. The *query construction* component builds a set of queries by retrieving information about the family of the subject from the civil registry. Given a family F, each member m_i is described by a set of character strings providing personal information such as fiscal code, first name, last name, birth date, address and city. Multiple query templates are prepared for each family:

(a) *Queries featuring the attributes of a single subject.* For each member $m_i \in F$ the system prepares queries composed by one or more attributes, combined with the quotation marks operator, such as:

```
"fiscal_code"
"firstname lastname"
"firstname lastname" "address"
```

(b) *Queries featuring attributes of various subjects in F.* For each pair $(m_i, m_j) \in F \times F$, with $i \neq j$:

```
"firstname_i lastname_i" "firstname_j
 lastname_j"
```

Each query template is given a score $c \in [0,1]$, which measures how likely it is for those queries to fetch pages containing information concerning the family F.

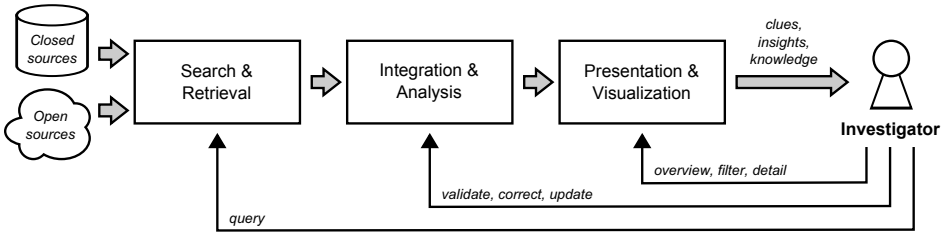

Figure 1: Our general approach to the problem is to define an *investigation pipeline*, where investigators can interact with three consecutive steps of closed and open source information processing (see Section 2 for more details).

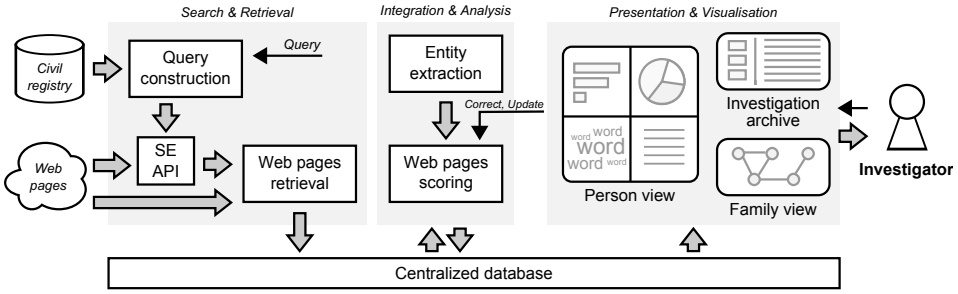

Figure 2: A specific architecture is defined to adapt the pipeline to the goal of investigating natural *people*. The system exploits data from civil registries to perform queries to web search engines, extracting relevant entities (e.g., names, phone numbers) from the retrieved pages. All information is assigned a manually adjustable score, in order to prioritize the investigator's access to it in the visual representation step (Subsection 2.1).

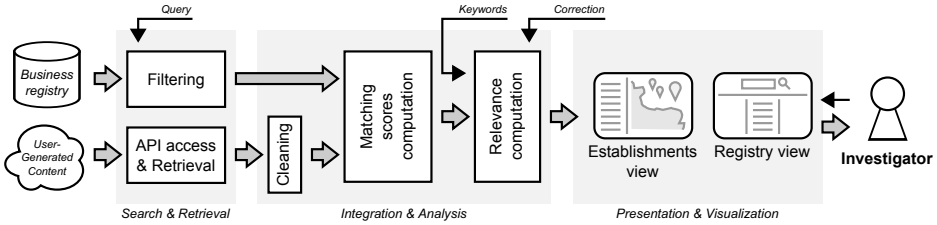

Figure 3: A second architecture is defined to tailor the pipeline to the case of legal people, i.e., *businesses*. The records obtained from business registries are matched with information published by users on the Web, in order to find which advertised establishments or services are not official. The system scores the results according to its confidence about the relevance of the entries, and uses this adjustable estimate to provide a prioritized view of potential suspects (Subsection 2.2).

The *Search Engine API* component executes the queries, retrieving a set of URLs. Then, the *Web Pages Retrieval* component actually downloads the web pages, storing them in a centralized database as a set of attributes having the following structure: *URL*, *title*, *snippet* (i.e., an excerpt of the page that matches the query), *plain text* and *query*.

2. *Integration and Analysis*. Firstly, the *Entity Extraction* component identifies entities contained in the plain-text version of the web pages, and stores them in the database. Named Entity Recognition and Classification (NERC) techniques are used for extracting the entities and assigning one of the following classes to each of them: *person*, *e-mail*, *telephone number*, *VAT registration number*, *fiscal code*, *IBAN code*, *social network nickname*, *price* and *profession*.

   After that, the *Web pages Scoring* component assigns a score to each web page equal to the corresponding query template score c. Then, it refines this score, according to predefined criteria that consider the number and the type of entities in common between web pages.

3. *Presentation and Visualisation*. This module collects the data processed by the previous ones and defines a web interface that embraces three main views:

(a) *Investigation Archive*. It is the starting point of our investigation process. In fact, it allows the user to keep track of the performed investigations and examine the corresponding web pages

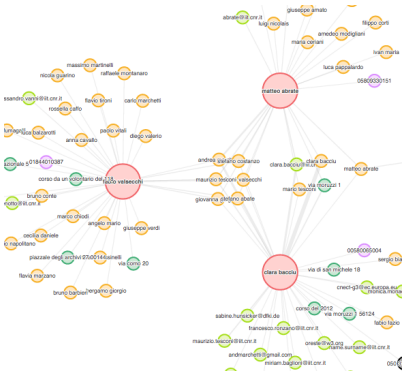

Figure 4: A portion of the node-link diagram included in the *Family View*. Two clusters can be identified in the center of the image, composed by entities (smaller nodes) connected to 2 and 3 family members (larger, red nodes).

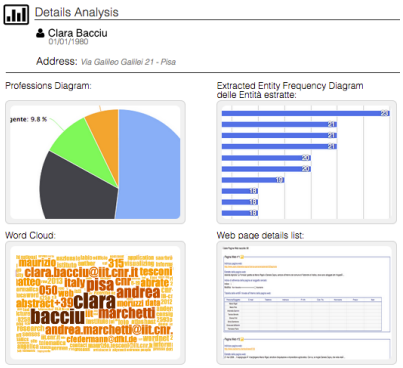

Figure 5: The *Person View* provides four different diagrams showing different aspects of the analysis on a single person.

through an interactive list that filters them according to various criteria (e.g., their score, the family to which they are connected or the query that has generated them).

(b) *Family View* (Figure 4). It consists in a node-link diagram that describes the relations between the members of a certain family and the entities extracted from the web pages related to them. This graph allows to easily identify clusters of entities connected to a single member or shared by some of them.

(c) *Person View* (Figure 5). It represents the core of an investigation. This dashboard view comprises different diagrams related to the information extracted from the web pages connected to a single person. A pie chart summarizes the professions identified by the system; An interactive bar chart shows which are the most recurrent entities for each class; A word cloud allows to identify which are the most frequent words inside snippets. Moreover, a list of the retrieved web pages is provided, allowing the investigator to see the entities they contain and to manually change the page score, triggering an update of all the diagrams if a new value is set.

2.2 Investigating businesses

A second architecture (Figure 3) is a specialization of the investigation pipeline for the task of investigating businesses (i.e., legal people) that are advertised on the Web, but are not registered in official Public Administration archives. The user must be able to: (i) Retrieve a set of commercial activities that are advertised on the Web for a certain administrative area; (ii) See where they are located; (iii) Spot the more relevant ones, in terms of how likely it is that they need a deeper inspection; (iv) Correct and update the priority with which the system shows the results.

As in the previously described architecture, the steps of the investigation pipeline are specialised:

1. *Search and Retrieval.* Official data are retrieved directly from an official business registry, while User Generated Content (UGC) is accessed trough the APIs provided by social networks or websites (i.e Facebook, Google Places, Foursquare, advertisement websites, etc.). The user can issue a query to *filter* establishments by administrative area.

2. *Integration and Analysis.* UGC undergoes a *cleaning* process that extracts only the relevant information about businesses (e.g., denomination, address, coordinates) then it is analysed together with official data.

   Records from the two sources are compared to obtain a *matching score*. We follow an approach inspired by approximate reasoning (Zadeh, 1975), allowing us to tackle the intrinsic uncertainty of automatic data integration while also specifying the core formula of the score computation in a logic proposition. Each establishment found on the web is compared to each official record. A set of similarity scores is computed for each pair (i, j). Each similarity score is treated as a fuzzy variable, and the scores are combined through the following formula to compute an overall matching value:

$$M_{ij} = IN_{ij} \wedge (N_{ij} \wedge (\underline{SN_{ij}} \vee CN_{ij} \vee (SN_{ij} \wedge SA_{ij})))$$

   IN expresses if the names of the businesses are exactly *identical* in both the sources; *N* expresses if the businesses are located *near* each other; *SN* expresses if the names of the businesses are *similar* ($\underline{SN}$ stands for *very similar* - fuzzy intensification); *CN* expresses if one of the names is entirely

Figure 6: A portion of the *Establishment View* showing the most relevant businesses proposed by the system.

contained in the other name; *SA* expresses if the addresses of the businesses are *similar*. The overall matching score of the establishment is computed as the maximum score $M_i = \bigvee_j M_{ij}$.

Another component computes the *relevance* R_i of each establishment as the complement of M_i, combined with a variable P_i expressing the pertinence of the establishment to the business register[4] : $R_i = (1 - M_i) \wedge P_i$. The user can control the pertinence by specifying keywords describing businesses that are not considered important (e.g., if the user doesn't want a "lawyer" or a "dentist" to have a high relevance score), and can also correct the final computation and assign a new value of relevance, if needed.

3. *Presentation and Visualization.* The last step of the pipeline defines a series of views for the user to examine the results:

(a) *Establishments View.* This view is divided into two sections: a list that shows the commercial activities retrieved from the Web in order of descending relevance; and a map that shows the location of each establishment. The user can move a slider on the map to filter the placemarks by relevance. For each element of the list, the user can see the three corresponding entries of the business registry having the higher matching scores. A diagram on the leftmost side of the interface shows the trend of the relevance values throughout the whole dataset.

(b) *Registry View.* It has the purpose to let the user perform a simple keyword search on the official business registry, in order to manually check the validity of the results of the automatic system.

3 WORK IN PROGRESS

This section describes the current development of two prototypes implementing the architectures discussed in Section 2. Since we used CSINF coming from Public Administrations, both prototypes take into account the Italian law about the processing of sensitive data.

As for the people investigation prototype, we acquired the data of the Civil Registry of Tuscany from the TOSCA[5] database. After an analysis of the freely available search engine APIs, we chose the Google Custom Search Engine (CSE) API since it allows the use of powerful operators for making more specific queries and provides a larger amount of relevant results. Depending on the class, we extract entities from web pages employing different NERC techniques such as dictionaries, regular expressions and the third-party API of Alchemy[6].

Since the prototype is not complete at the moment, a formal testing activity has not been carried out yet. Nevertheless, the users where involved in some preliminary tests, in order to early spot and correct errors, both in the data and in the interface. The system has been tested by searching for data about 20 people, some belonging to our research laboratory, some being known to the users as suspect tax evaders. In 10 cases the official starting data were complete (i.e., name, surname, fiscal code, date and place of birth), while for the other 10 cases some official data was missing. The users where generally satisfied by the results, while in some cases the problem of homonymy has proven to be an issue. As expected, the retrieved information about the complete cases were more relevant and precise than those of the incomplete ones. A remark needs to be done about the Google CSE API: the users expect the system to retrieve the same exact pages that Google shows as result when a keyword search is made, but in some cases the API returns only a fraction of them. In one case, the returned results were so few that the system failed in getting relevant information.

The prototype developed for investigating businesses has been implemented by retrieving official data from the Italian Business Registry. Unofficial

[4]Registration is not compulsory for many of the commercial activities found on the Web.

[5]The Tuscan platform that supports the Public Administrations in fighting tax evasion (TOSCA, 2010)

[6]http://www.alchemyapi.com

data is instead gathered through the Google Places API for the whole province of Lucca, Italy. Users involved in the preliminary tests are quite satisfied, both with the interface and the results the system shows.

An analysis of raw data that comes out from the matching score computation is ongoing. In Figure 7 a fraction of the matching scores is shown. Each row of

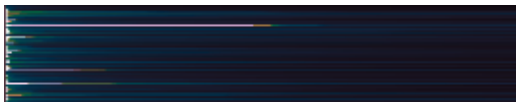

Figure 7: A visual analysis of a portion of the matching scores, showing fading lines and uniform streaks.

pixels represents a commercial activity retrieved from Google. Each cell of the row is a color-coded matching score describing the similarity between a commercial activity retrieved through Google and a registered one. We assign colors to the scores using a continuous color scale, clear for high scores and dark for low ones. For each line we order the scores from higher to lower, so it is possible to notice fading lines of color. Fading lines are expected, since they are due to the intrinsic uncertainty of the system. Streaks of the same color for a large number of cells mean instead that the same exact matching score was computed for a large number of couples. This shows that the system can be made more fuzzy in order to become less ambiguous.

For estimating the reliability of the matching score algorithm, we randomly selected a set of 1.000 establishments from those extracted from Google Places, whereof 606 have been manually annotated as having a sure match in the Business Registry. We then ran the algorithm on the same 1.000 entries and computed that 83%, 78% and 62% of correct matches can be found respectively in the top 10, 3 and 1 relevant results.

4 CONCLUSION

In this article, we presented an ongoing project consisting in the design and development of an *investigation platform* that supports tax inspectors in their tax-evasion inquiries. The prototypes can be improved upon. More sophisticated NLP techniques may be adopted to obtain more accurate results in the entity extraction phase. Machine Learning clustering algorithms may be tested for limiting the problem of people homonymy on the Web. The fuzzy calculation introduced in Section 2.2 may be changed by modifying the logic formula and even by adding new fuzzy variables. The preliminary tests are promising and show that the use of OSINF in the investigation of tax-evaders can be effective. Nevertheless, a massive testing phase involving users is fundamental for validating and refining the overall platform.

ACKNOWLEDGEMENTS

We would like to thank the municipality of Fabbriche di Vallico and ANCI Toscana for funding this work, Andrea D'Errico, Sergio Bianchi and Alessandro Prosperi for their contribution in the project.

REFERENCES

Aliprandi, C., Irujo, J. A., Cuadros, M., Maier, S., Melero, F., and Raffaelli, M. (2014). Caper: Collaborative information, acquisition, processing, exploitation and reporting for the prevention of organised crime. In *Intelligence and Security Informatics Conference*.

Best, C. (2008). Open source intelligence. *Mining Massive Data Sets for Security: Advances in Data Mining, Search, Social Networks and Text Mining, and Their Applications to Security*.

Ducke, D., Kan, M., and Ivanyi, G. (2010). *The Shadow Economy-A Critical Analysis*.

Feige, E. L. and Cebula, R. (2011). America's underground economy: measuring the size, growth and determinants of income tax evasion in the us. *Crime Law and Social Change*.

Internet Live Stats (2015). `www.internetlivestats.com/`.

ISTAT (2010). La misura dell'economia sommersa secondo le statistiche ufficiali. `http://www3.istat.it/salastampa/comunicati/non_calendario/20100713_00/testointegrale20100713.pdf`.

Johnson, L. et al. (2007). *Handbook of intelligence studies*.

Maciołek, P. and Dobrowolski, G. (2013). Cluo: Web-scale text mining system for open source intelligence purposes. *Computer Science*.

Neri, F. and Geraci, P. (2009). Mining textual data to boost information access in osint. In *IEEE 13th International Conference in Information Visualisation*.

Sogei (2010). Serpico. `http://goo.gl/yV7YNF`.

TOSCA (2010). Tosca project. `http://www.regione.toscana.it/imprese/innovazione/progetto-tosca`.

We Are Social Singapore (2014). Global digital statistics. `http://www.slideshare.net/wearesocialsg/social-digital-mobile-around-the-world-january-2014`.

Yang, H.-C. and Lee, C.-H. (2012). Mining open source text documents for intelligence gathering. In *IEEE International Symposium on Information Technology in Medicine and Education*.

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoningi. *Information sciences*.