A Proposal of Web Data Mining Application for Mapping Crime Areas in the Czech Republic

Martin Lnenicka, Jan Hovad, Jitka Komarkova and Miroslav Pasler
Faculty of Economics and Administration, University of Pardubice, Studentska 84, 532 10, Pardubice, Czech Republic

Keywords: Web Data Mining, Geography of Crime, Decision Support, GIS, Python.

Abstract: In this paper, authors offer a new view on the issue of the geography of crime areas. The paper proposes an interface to support the decision-making process in the reasonable time. The proposed application is used to extract crime related information, find crime hotspots and present crime trends using web data mining techniques. The target area is the sum of all administrative districts of municipalities with extended powers in the Czech Republic. Every crime problem is related to some location, whether it is an address, street or district. The proposed application scans the selected mass media servers for the words connected with the crime type and the concrete municipality. This paper shows the detailed overview about the proposed and used architecture, methods and data structures.

1 INTRODUCTION

In recent times, while there was quick growth of urbanization and exponential increase in crimes, most authors focused on the geography of crime areas. But prevention and planning is important too. Related policies and planning concentrate on crime prevention and emphasize the backgrounds and contexts leading to crimes. A quick analysis of crime data can provide key information both to police organizations to improve safety at the operational level and to policies makers at strategic and tactical levels. Citizens also desire crime information to make informed decisions about where to live and work. However, to the authors' best knowledge, very few publications can be found, that discuss the issue of delivering the crime related results in minimum possible time.

Geographic Information Systems (GIS) are used in many different ways in the crime analysis. They bring geographical dimension into these analyses. Geographic profiling methodology, which is focused on geographical patterns of criminals, can be given as an example. Also variety of socioeconomic and crime opportunity factors such as the population density, economic investment and land use are likely to influence criminal behavior and it is necessary to take them into account when analyzing these data.

The main principle of the proposed application is to find a required crime type and the neighbor word, which occurs in the list of target cities. If there is any match, then get the location, compare it with JavaScript Object Notation (JSON) alternatives and draw into the map. The target level is represented by the 206 municipalities with extended powers in the Czech Republic (CR) and the four crime types based on the crime classification and definition in the CR: krádež (theft), loupež (robbery), vražda (murder), znásilnění (rape) and their different shapes in the Czech language. Authors are focused to design and code a web data mining application, which provides the basic layer to analyze web content for the crime references, locate them on the map and calculate the weights for target cities through the time.

The proposed application runs automatically in the specified time intervals on the server. The main used language is Python, which is presented e.g. in (Oliphant, 2007). This programming language is suitable for scientific purposes because it simplifies many complicated programming approaches into the unified and understandable form. An Application Programming Interface (API) is used to limit programming only to the computational tasks and use the suitable existing services to do the rest, for example the visualization of results. An API is an interface of a service, that is usually reachable by login information and it provides the possibility to outsource a specific activity. An API creation is covered in the work of Henning (2007). This job can be done using e.g. the PHP, which was widely used

Lněnička M., Hovad J., Komárková J. and Pásler M..
A Proposal of Web Data Mining Application for Mapping Crime Areas in the Czech Republic.
DOI: 10.5220/0005558104500455
In *Proceedings of the 10th International Conference on Software Engineering and Applications* (ICSOFT-EA-2015), pages 450-455
ISBN: 978-989-758-114-4

in this paper for the database tasks. Processing the HTML data (website raw string) response to the sent request was done by parsing the output tree, which is described by Jackson (2003). Accessing information like getting the specified container content usually results in the necessity to filter the text string can be done by using the regular expressions and special libraries that can perform operations like removing the unwanted JavaScript or comments in the source code (Baeza-Yates and Gonnet, 1996). Uniform data structure as a XML was replaced by a JSON, which is more suitable for Python and it is also used by many APIs as a data structure to transfer the optional adjustment arguments (Peng et al., 2011). JSON city dictionary was created by the help of Esri ArcGIS.

2 RELATED WORK

The geography of crimes has been deeply studied to understand distribution and patterns of the crimes and to identify the factors which influence them. Spatial image of crime distribution and comparing its data with characteristics of the place of crime and socioeconomic status can give the opportunity to identify the crime hotspots and predict the possible crime places in urban areas (Ardian et al., 2014).

Crime can be analyzed at many different levels of aggregation in time and space. Criminal activities tend to concentrate in certain places for reasons that have to be explained. Areas of concentrated crime are often referred to as hotspots (Wang et al., 2013).

Web mining lies in between data mining and text mining, and copes mostly with semi-structured data and/or unstructured data. Web content mining is performed by extracting useful information from the content of a web page (Zhang and Segall, 2008). The one aspect of web mining is the data collection. Web mining provides automated mechanism to collect the relevant data for predefined objectives from the enormous web data (Hussain and Asghar, 2013).

Shelley (1980) for example identified that the influence of the deep inhabitants control including inner passport system in the Soviet Union resulted into the different crime distribution in comparison to other industrialized countries. Wang et al. (2013) combine the ideas from spatial data mining and introduce a new hotspot mapping tool to improve the identification of crime hotspots through the mining of spatial patterns composed of crime related factors. Along similar lines Nath (2006) developed the claim that a data mining approach can help detect the crimes patterns and speed up the process of solving crime. For the support of this claim, the author used k-means clustering technique and developed the scheme for weighting the significant attributes.

Similarly, systems dynamics in connection with GIS technologies was used to simulate impacts of various social policies (education, environment and unemployment policy) on crime rates (Quijada et al., 2005). Erdogan et al. (2013) used exploratory spatial analysis to demonstrate the utility of spatial analysis and geographically weighted regression to detect important geographical dimensions and crucial geographical aspects of property crimes.

Friebel and Friebelova (2012) have evaluated the life quality in the Czech districts using the method of Data Envelopment Analysis (DEA). Criminality belonged to one of the four basic inputs of the DEA model. However, several practical questions arise when dealing with the identification of the crime in the real time or at least in the reasonable time.

All above described approaches are focused on web content mining methods. The spatial factor and speed of queries processing are not understood as so significant factors. Authors focus on these aspects.

3 PROBLEM STATEMENT AND USED METHODS

The paper is written with the primary aim of creating the object oriented Python application for the identification of the crime areas in the CR in the reasonable time. For the purpose of this goal, the proposed application is directed into the mass media servers, press and a newspaper area. The sources (articles) in the selected media servers are analyzed with the effort to find the references to the target crime type with the connection to the concrete city.

The main tool used is Python in the version 2.7. Version 3.3 is not used because of the limited compatibility with some web services for example Google. IDE was set to PyScripter, which provides the basic debugging options and it keeps the work environment simple. Python is used along with basic built-in libraries like time / datetime, JSON (data structure handling), urllib2 (processing the HTML requests), re (matching or replacing text by regular expressions), HTML-Parser (basic tag filtering) or collections (counting the results, statistics). The other libraries are installed additionally. Then the MySQLdb connector is used to save the history data during the script runs. The beautiful soup (BS) library is used for accessing the data. Google Geohart API and Maps API with basic functions are used to utilize the graphing API.

The code is written in objective way and it can be extended to cover new location-based situations and handle number of different tasks, such a social media monitoring, crisis management, customer segmentation, etc. The raw class structure of the proposed web data mining application can be seen from the Figure 1.

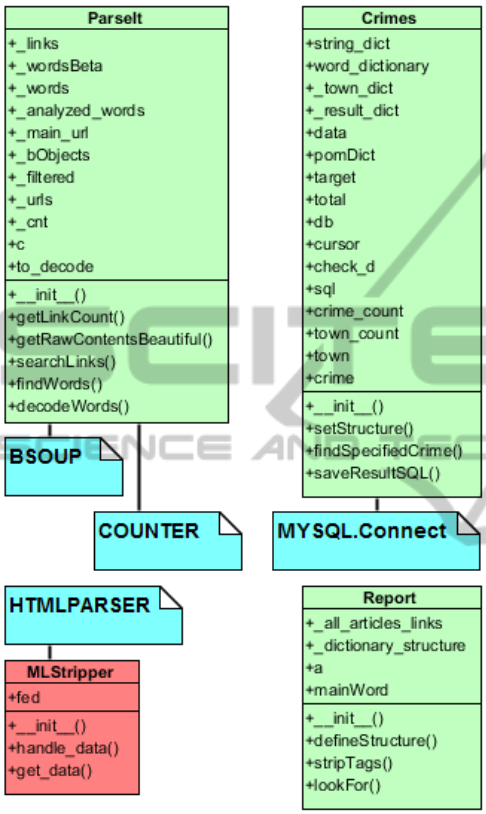


Figure 1: Class diagram of the proposed application.

4 DESCRIPTION OF THE WEB DATA MINING APPLICATION

4.1 Targets and HTML Structure

Authors use the three biggest media servers in the CR. Sources are: www.novinky.cz, www.ct24.cz, www.m.idnes.cz. This fact is crucial, because the Czech language is characterized by hard inflection of individual words and their meanings. Desktop versions usually need more focus in the case of clearing the content, mobile versions are clearer.

The HTML request is made for each of these sources and the result is then saved in the string variable. This variable then contains everything: comments, JavaScript, tags and also a plain text of the articles. This text string is then transformed by the BS library into the BS object (BO). After that, each source is loaded by an object constructor and the title page is scanned for the relevant links that lead to the individual articles. Every tag in the document could be easily accessible, e.g.:

```
<title>
  Media server webpage
</title>
BO.title.string
"Media server webpage"
```

The only manual step before running the script is to explore the HTML structure of the source for the identification of the target content. This analysis is shown in the Figure 2. If the web structure of the target media server is changed, the code of the proposed application has to be also modified. However, it is a simple task, only the function, which deals with the HTML containers (mostly with their names), has to be changed.

Figure 2: The analysis of the basic HTML structure.

4.2 Filtering Content and Data Structure

Initially, the title page of the first mass media server is scanned for the all <*h3*> occurrences, because the new articles are characterized by this tag. All the content inside is scanned again for the <*a*> tag that contains links to the full texts. Then the content of this tag is returned and the string of the attribute was saved into the associative array (dictionary). Every article then also gets an ID to ensure that it is scanned only once. The shortened version of this step for a selected source is shown in the Figure 3. The complete code is 350 lines long.

Another iteration of this operation processes the founded links once again. They are converted into the BO one by one while the pure text of the article was extracted. Pure text is placed in the <*div*

```
def searchLinks(self):
    for key, value in self._bObjects.iteritems():
        if key == "http://www.novinky.cz":
            print "*****NOVINKY "*5
            for item in value.find_all("h3"):
                for x in item.find_all("a"):
                    pom = x.get("href")
                    self._links[key].append(pom)
            for item in value.find_all("h2"):
                for x in item.find_all("a"):
                    pom = x.get("href")
                    self._links[key].append(pom)
            for item in value.find_all('div', attrs={'class':'mainArticle hasIlust'}):
                for x in item.find_all("a"):
                    pom = x.get("href")
                    self._links[key].append(pom)
```

Figure 3: The function to search through the HTML structure for links.

id="articleBody"> and this structure is global for the whole website. The content is returned and saved as a string. MLStripper class inherits from the HTMLParser and instantiates an object that clears a string from the HTML tags and Javascript. Data structure is characterized as a key-value pair, where the value can be another associative array. Words are stored in the JSON structure.

4.3 Dictionaries and Debugging

Czech language has many possible shapes / options for a single word. Therefore, two dictionaries had to be created to cover more than one inflection of the concrete word: the list of cities and the list of crimes. The structure is *word(key):[shapes(args)]*. This had to be done for all the 206 municipalities (cities) with extended powers and 4 crime types. The focus of the dictionaries can be easily changed or extended.

4.4 Results of Searching and Their Storage

This function takes an input word (e.g. murder) and set of *args (murdered, murdering, murders, etc.) and compares all these words with the words in the target source (string of a pure text). If any of input word or *args occurred, the key, which was represented by the input word, is returned. After that, the article is scanned for the location (the city and its *args) from the list of cities. If the city is found, the presence is saved along with weights / counts. Google Geochart API and Maps API are used to present the results of the selected crime and its appropriate city and also a weighted count in the specified interval, which leads the circle size. The time attribute represents the borders for the article on the title page in the selected source. The function that found the reference to the concrete crime and the city is shown in the Figure 4.

```
def findSpecifiedCrime(self,target_word,*optional_alternatives):
    pomDict = {}
    target = target_word.lower().decode("utf-8")
    total = 0
    for article_link,word_array in self.word_dictionary.iteritems():
        c = 0
        for wrd in word_array:
            if wrd==target:
                c+=1
                print article_link
            if len(optional_alternatives)>0:
                for opt in optional_alternatives:
                    if opt.lower().decode("utf-8") == wrd:
                        print article_link
                        c+=1
        pomDict[article_link] = (target_word,c)
        total+=c
        if c > 0:
            for wrd in word_array:
                for town,town_option in self._town_dict.iteritems():
                    if wrd==town.lower():
                        pomDict[article_link]+=(town,)
                        break
                    for i in town_option:
                        if i.lower()==wrd:
                            pomDict[article_link]+=(town,)
    print "Total number of occurences for word: "+target+" and its
    self._result_dict[target_word] = pomDict
```

Figure 4: The function for the finding of the crime.

Finally, the results are saved into the database table. The much shortened version of this function is shown in the Figure 5.

```
def saveResultSQL(self):
    db = MySQLdb.Connect(host=app_cfg.host, port=app_cfg.port, user=
    cursor = db.cursor()
    check_d = cursor.execute("""SELECT * FROM information_schema.ta
    db.commit()
    if (check_d):
        print("CRIMES table is ready, I can insert the data ... ")
    else:
        print("CRIMES table is not ready, trying to create a new one.")
        sql = """CREATE TABLE realmining.CRIMES (CRIME  VARCHAR(
        cursor.execute(sql)
        db.commit()
    crime_count = 0
    town_count = 0
    town = ""
    crime = ""
    for target_w, indiv_dicts in self._result_dict.iteritems():
        for link,tpl in indiv_dicts.iteritems():
            part = 0
            towns_occurances=Counter()
            if len(tpl)>2:
                print "\n PRINTING MATCHED TOWN OCCURANCES \n"
                crime=tpl[0]
                crime_count = tpl[1]
                towns = tpl[2:]
                for town in towns:
                    towns_occurances[town]+=1
                sum = 0
                for key,val in towns_occurances.iteritems():
                    sum += val
                part = 1.0/sum
```

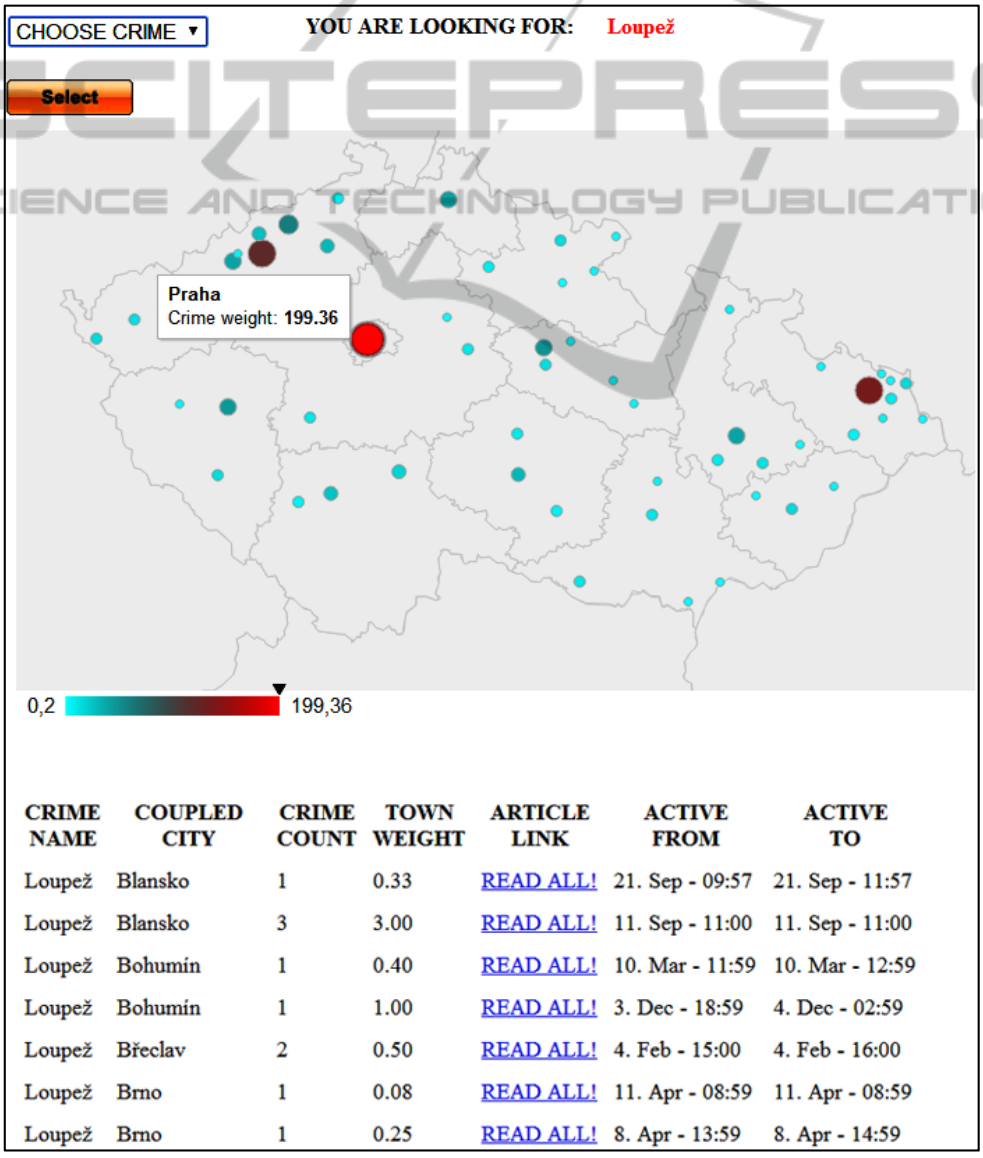
Figure 5: The function to save the crime into the table.

5 RESULTS AND DISCUSSION

In this paper, authors describe an effective approach

to analyze publicly available content on the Web to locate crime areas and support the decision-making process in the reasonable time. The proposed web data mining application utilizes Python and Google API, which significantly simplifies the front-end visualizations. All the background work is done by PHP (database listing, control elements), HTML and CSS (basic web layout), JavaScript (Google API communication) and MySQL (basic selects and operations over the data table). The target mass media servers in the CR are scanned every hour and the obtained data are saved into the database table since the deployment of the application. The output of the proposed application can be seen from the Figure 6.

The user can choose the crime type from the list box. Only one crime type can be selected and shown in the map. The results of query consist of crime name, coupled city (crime hotspot), crime count, town weight, article link and how long was the link active on the title page of the selected mass media server (after that, the article was moved in the archive). The algorithm for the calculation of the weight is the sum of the target word in the selected article (CRIME NAME in the Figure 6) for the monitored city (COUPLED CITY). If the article contains only the one city and only the one target word, then the weight is 1 (TOWN WEIGHT), for the two cities it is 0.5, etc. The final weight for the selected city is the sum of the previous weights



CRIME NAME	COUPLED CITY	CRIME COUNT	TOWN WEIGHT	ARTICLE LINK	ACTIVE FROM	ACTIVE TO
Loupež	Blansko	1	0.33	READ ALL!	21. Sep - 09:57	21. Sep - 11:57
Loupež	Blansko	3	3.00	READ ALL!	11. Sep - 11:00	11. Sep - 11:00
Loupež	Bohumín	1	0.40	READ ALL!	10. Mar - 11:59	10. Mar - 12:59
Loupež	Bohumín	1	1.00	READ ALL!	3. Dec - 18:59	4. Dec - 02:59
Loupež	Břeclav	2	0.50	READ ALL!	4. Feb - 15:00	4. Feb - 16:00
Loupež	Brno	1	0.08	READ ALL!	11. Apr - 08:59	11. Apr - 08:59
Loupež	Brno	1	0.25	READ ALL!	8. Apr - 13:59	8. Apr - 14:59

Figure 6: The output of the proposed web data mining application.

calculated. Under the map, there is also a list of the actual crime related news (hourly updated), which is the most important part in the support of the decision-making process and delivering the crime related results in the minimum possible time.

The highest crime weight in the Figure 6 has the capital city of the CR. The other crime hotspots are situated in the North Bohemia region (Most and Teplice) and in the North Moravia region (Ostrava). These results were validated by comparison with the Police of the CR official database and they showed no significant difference between the compared data.

Although the geographical area for the proposed application is the CR, this research can be of general interest to practitioners in criminology, data mining and geographical mapping in other parts of the world as well, because the demonstration of how an application can be used to map crime can be adapted to other environments and scenarios.

6 CONCLUSIONS

The main objective and partial goals are completed successfully. The attention of authors is in this case concentrated only on the mass media servers. Nevertheless, the approach outlined by this paper can be integrated to provide much broader analyses.

Organizations using this application can improve decision-making capabilities in a rapidly changing environment and have a direct impact on the safety of the selected cities. In addition, it can help the police and other public sector institutions view and understand underlying crime hotspots, movements and patterns. It can also help to increase cooperation between them and the citizens they serve.

On the basis of the promising findings presented in this paper, work on the remaining issues is continuing and will be presented in the future papers. As a future extension of this paper, authors will focus on the improvement of the proposed web data mining application. Users will have an option to select the month (date) or the concrete mass media server together with the selected crime type as a list box and the choice to filter the results. Nonetheless, the first step in this process should be the definition of the framework and identification of patterns.

ACKNOWLEDGEMENTS

This paper was supported by the SGSFES_2015001 fund.

REFERENCES

Ardian, N. et al., 2014. The Spatial Analysis of Hot Spots in Urban Areas of Iran. The Case Study: Yazd, *Revista de Cercetare şi Intervenţie Socială*, 44, 103–115.

Baeza-Yates, R. A., Gonnet, G. H., 1996. Fast text searching for regular expressions or automaton searching on tries, *Journal of the ACM*, 43(6), 915–936.

Erdogan, S., Yalcin, M., Dereli, M.A., 2013. Exploratory spatial analysis of crimes against property in Turkey, *Crime Law And Social Change*, 59(1), 63–78.

Friebel, L., Friebelova, J., 2012. Quantitative evaluation of life quality of Czech districts, In *Proceedings of 30th International Conference Mathematical Methods in Economics*, 190–195, Silesian University in Opava.

Henning, M., 2007. API design matters, *Queue*, 5(4), 24–36.

Hussain, T., Asghar, S., 2013. Web mining: Approaches, applications and business intelligence, *International Journal of Academic Research*, 5(2), 211–217.

Jackson, Q. T., 2003. Efficient formalism-only parsing of XML/HTML using the §-calculus, *ACM SIGPLAN Notices*, 38(2), 29–35.

Nath, S. V., 2006. Crime pattern detection using data mining, In *2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Workshops*, 41–44, ACM.

Oliphant, T. E., 2007. Python for scientific computing, *Computing in Science and Engineering*, 9(3), 10–20.

Peng, D., Cao, L., Xu, W., 2011. Using JSON for data exchanging in web service applications, *Journal of Computational Information Systems*, 7(16), 5883–5890.

Quijada, S. E., Arcas, J. F., Renner, C., Rabelo, L., 2005. A spatio temporal simulation model for evaluating delinquency and crime policies, In *Proceedings of the 2005 Winter Simulation Conference*, 1328–1334, IEEE.

Shelley, L., 1980. The Geography of Soviet Criminality, *American Sociological Review*, 45(1), 111–122.

Wang, D. et al., 2013. Crime hotspot mapping using the crime related factors – a spatial data mining approach, *Applied Intelligence*, 39(4), 772–781.

Zhang, Q., Segall, R. S., 2008. Web mining: a survey of current research, techniques, and software, *International Journal of Information Technology and Decision Making*, 7(4), 683–720.