Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-020-01942-y

ORIGINAL RESEARCH

Profile generation system using artificial intelligence for information recovery and analysis

Pablo Chamoso[1] · Álvaro Bartolomé[1] · David García-Retuerta[1] · Javier Prieto[1] · Fernando De La Prieta[1]

Received: 21 March 2019 / Accepted: 28 March 2020

Abstract
The advances in data computing and analysis methodologies have contributed to the added value of data. Several years ago it was difficult to imagine that we would ever be able to extract such a large amount of information from the Internet. All this thanks to the ability of current techniques to process large volumes of data in a short period of time. The Internet provides access to a large amount of unstructured or unlabelled data, which are hard to retrieve for any human due to the lack of knowledge of the available sources of information. Moreover, in many cases people are unaware of the online availability of their personal data. This article presents a system for retrieving personal information from the Internet on the basis of several input criteria. The system is capable of differentiating the information of different people with the same name by using artificial intelligence techniques. In the conducted case study, the information has been gathered from sources containing information about people living in Spain, but it could be adapted to the specific sources of information of other countries. The system has been validated in a case study which included several participants and the obtained results have been quite satisfactory.

Keywords Information recovery · Information fusion · Big data · Profiling

1 Introduction

Today, the internet provides us with access to a great deal of information. Thanks to the current techniques large volumes of information (Big Data) can be processed and analyzed to extract knowledge. Less than 10 years ago, when there was already a lot of information on the Internet, the extraction of valuable knowledge was an unthinkable process.

A lot of personal information can be found about both globally known and anonymous people. The fact that this information is public and that anyone can access it has generated considerable controversy. Internet search engines, such as Google, have been forced to implement mechanisms in multiple countries to offer people the possibility the "right to be forgetten", according to this ruling a person may wish to erase some of their data permanently from the public database and search engines are legally obliged to do so.

The goal of the proposed system is to search for a person's personal information made available by search engines such as Google or Bing and to use this information to create a personal profile. The input criteria for performing the search include a person's full name and a series of keywords associated with them. The information is retrieved from multiple sources and analysed. The purpose of this system is to allow anybody to find and track the publicly available information about them.

The system uses this input data to search social networks, phone numbers, emails, related news and images of the person. The information retrieval process is performed according to the Big Data ETL (extract, transform and load) methodologies that are mainly based on web crawlers and web scrapers, with the purpose of retrieving huge loads of data indexed on the Internet, as explained by Olston et al. (2010). In addition, Artificial Intelligence (AI) methodologies, such as clustering (Allahyari et al. 2017) or text analytics (Moreno

✉ Pablo Chamoso
chamoso@usal.es

Álvaro Bartolomé
alvarob96@usal.es

David García-Retuerta
dvid@usal.es

Javier Prieto
javierp@usal.es

Fernando De La Prieta
fer@usal.es

[1] BISITE Research Group, University of Salamanca, Calle Espejo s/n, 37007 Salamanca, Spain

Published online: 15 April 2020

and Redondo 2016) are applied to analyze the sources from which the information about a person is obtained. This analysis helps determine whether the information pertains to the person being search for, hereinafter called the person-to-search. In this case study, the text analytics techniques are designed for the analysis of Spanish texts only but it is possible to extend them to other languages.

The main novelty of the article lies in applying a set of well-known AI methodologies, which have been widely used for different purposes, but there is no published research that groups all of them together to achieve the results achieved in this work.

The methodology proposed to search for information is also a novelty, since not only the first results obtained in search engines or in selected websites are analyzed, as other information retrieval systems usually do, such as the presented by Barbosa and Freire (2010), but also (1) all the results obtained are recovered and analyzed, (2) the links to other pages that appear in those results, (3) text files that are linked and (4) the images with faces that are shown in the results are recovered as well as all the similar faces that appear between the results that a new search returns.

The proposed system achieves quite accurate results for profiles with peculiar names, but with very frequent names the accuracy of the results is lower, although it is still able to correctly identify the person whose information is being searched for. It is necessary to indicate that, due to the restrictions of the General Data Protection Regulation (GDPR), data gathered from third party sites are not stored, they are simply analyzed and the results are only presented to the user and do not persist. This also implies that each time a search is repeated, the whole process is carried out again, always obtaining results with updated information.

The rest of the article is structured as follows: Sect. 2 provides an overview of the background related to the present work. Section 3 discusses the methodology and describes the proposed system in detail. Section 4 outlines the process of implementing the proposed system and describes the case study. Section 5 summarizes the results obtained by the developed system. Finally, Sect. 6 draws conclusions from the conducted research and discusses future lines of research.

2 Background

The rapid advances in computing have provided us with a greater ability to retrieve information from the Internet, expanding the possibilities of information analysis.

Bahrami et al. (2015) speed up the search and retrieval of information from the Internet through the design of distributed architectures. Similarly, the emergence of technologies specifically aimed at both structured and unstructured distributed storage, such as NoSQL (Jose and Abraham 2017) or even HDFS (Hadoop Distributed File System), have been successfully used to process large volumes of textual information, such as Sun et al. (2017), contributing to advances in this line of research.

The combination of these techniques together with classic AI methodologies, such as K-Means, Term Frequency-Inverse Document Frequency (TF-IDF), Support Vector Machine (SVM), Artificial Neural Networks (ANNs), Histogram of Oriented Gradients (HOG) have all made it possible to develop the work proposed in this article.

These techniques are already being used in systems, similar to ours, that retrieve information from the Internet for different purposes. For example, Nguyen et al. (2019) and Roy et al. (2016) applied K-Means to create clusters of textual information represented with word embedded vectors and Ali et al. (2016) to design a system for image retrieval.

TF-IDF is a commonly applied technique that extracts frequently used words in a text such as documents or websites as detailed by Rivas et al. (2018). More specifically, it is a numerical statistic that identifies the keywords of a document or a website, in this case we used it to ensure that the text is related to the profile of the person-to-search.

SVM is a supervised learning model with associated learning algorithms that analyze data used for classification and regression analysis, which has been recently applied in works such as the presented by VenkateswarLal et al. (2019). It can be used to search and match facial patterns, making it possible to find more information about a profile by identifying those patterns. This methodology has been used in works such as the presented by Shah et al. (2017). Similarly, HOG algorithms (Dalal and Triggs 2005) or ANNs and their variations, such as Convolutional Neural Networks (CNNs) are also widely used today for facial recognition (Kasar et al. 2016).

All these existing techniques and methodologies have previously been used in different existing works which are related to the work described in this article, although they have not yet been applied together to solve a case study like the one described in this work.

Since the emergence of social networks there has been a trend in research that focuses on creating personal profiles on the basis of the information available about them on the Internet, such as the presented in Karidi et al. (2018). For more than a decade now, several researches have focused on developing user profiles, such as those presented by Balduzzi et al. (2010) and by Dang et al. (2016).

The purposes of those researches are multiple. Some researches are carried out for purely commercial purposes, such as the development of precise recommendation systems such as the presented by Davoodi et al. (2012) or systems for better segmentation of the marketing campaigns organized by companies such as the presented by Vasanthakumar

et al. (2016) that analyze the profiles of the most influential people on social networks determining the best target for an advertising campaign. According to Jayaram et al. (2015) it is necessary to extract these profiles to conduct more effective commercial campaigns. This has implied that numerous pages sell personal information to different companies due to the great value of personal data (Spiekermann et al. 2015).

However, the current trend in research is directed towards studying how publicly available information affects people's privacy. Fairly up-to-date studies address those issues such as the presented by Kandias et al. (2017) or by Ali et al. (2017).

Another trend is the monitoring of users on the web based on an identifier, so that if a user maintains the same username on different web pages, they can track their presence on the Internet to measure their level of activity, as well as their preferences. Falahrastegar et al. (2016) presented an interesting proposal focused on this line of research.

Since there is a great deal of research in this area, many tools have been designed that use keywords to search and track information in a network. Many of these tools are commercially available.

The best known tool is called Maltego, which has been designed for the discovery of data from open sources and their user-friendly visualization. Furthermore, although Maltego is a proprietary tool, different works such as the one presented by Marx (2014) have extended its functionalities to make it more personalized.

3 Proposed system

This section gives an overview of the platform, which is illustrated in Fig. 1 that identifies its modules and workflow. Implementation details are shown in Sect. 4.

It is common knowledge that the information we entrust the Internet with is stored and archived. Hence, the data extraction platform is mainly based on web crawling and web scraping, with the objective of retrieving personal information indexed and stored on the Internet directly or indirectly related to the person the final user is searching, from now on called person-to-search. The retrieved information is further going to be analysed in order to generate personal profiles from it after a filtering and analysis processing. Therefore, as input data to the platform, a form should be filled with the person-to-search basic information (name and surname) and some extra information such as keywords that can help the platform to single any person out. As keywords are a distinctive factor when it comes to automated search via either Google or Bing due to the high load of information indexed by them which means that the more keywords the less results, since both search engines base their resulting list of webs on both the ranking of the web and the coincidence between the input search and the web content. Thus, the output of the web crawlers and web scrapers should be all the information the system was able to retrieve, that later will be passed as an input to the Artificial Intelligence modules in order to analyse and filter it. Afterwards, once this process is completed, the analysed and filtered information will be shown in the dashboard, as a reply to the final user initial request.

As mentioned previously, the proposed system relies on the results extracted from the Internet via web crawling and

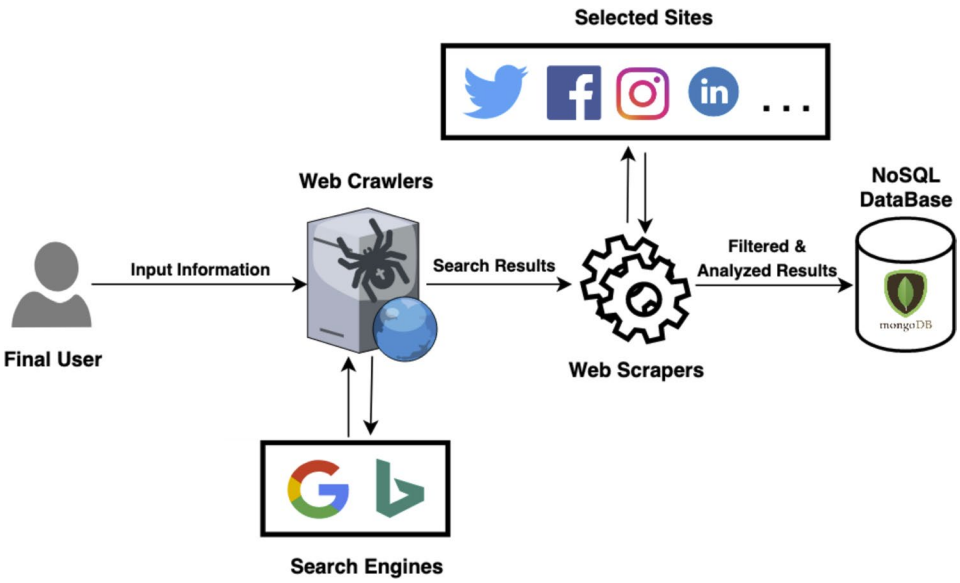

Fig. 1 Schema of the proposed system

web scraping, which is later sent to the Artificial Intelligence modules so that retrieved data can be processed and filtered for both text content and images. A clearer and more concise explanation of the modules that conform the main architecture of the platform is going to be presented as shown in Fig. 2, more detail is provided in Sect. 4.

The main engine is separated in two different modules, so that the processes involved in data retrieval via web crawling and web scraping from the ones involved on data analysis using AI, can be told apart. The data retrieval module is the one in charge of the extract, transform, and load (ETL) procedure referred to the information that is extracted directly from the Internet through automated search techniques and, so on, its later analysis and transformation so that it suits our proposed structured data model and the final load on a NoSQL database. The AI module is the one in charge of analysing and filtering the information either for Topic Modelling or Face Recognition, as its implementation supposes a big step forward when it comes to profile generation because it provides an additional value to the retrieved data. Another important fact while developing the proposed system was the use of a NoSQL database since a huge load of data is retrieved and needs to be stored in a database as to ensure persistence, the used database was MongoDB due to its fast access on read/write operations and its schema-less structure which allows every type of document insertions, not having to worry about structured data.

So on web crawlers are used to bulk download websites retrieved from any of both search engines Google and Bing, as the system automates web searches via the use of web crawlers, the web downloading process relies on them. Once all the webs resulting from the previous search query are downloaded, web scrapers are the ones in charge of parsing the HTML in order to retrieve just the needed content, which in this case is the plain text from every web (just content surrounded between HTML text tags) as it is common to every website. Web scrapers are used to retrieve specific information from HTML, but as a lot of different webs are retrieved and Web Scraping depends on how the HTML is structured, it can just be implemented if the sources are known, so web scrapers can be developed to retrieve specific data from specific webs. Additionally to the web crawlers resulting webs, some searches on know webs can be launched so additional data can be found on them, web scrapers can be implemented so to retrieve just the needed data from specific sources.

Topic Modelling is an AI process that classifies a set of documents into topics, where the content feature extraction of the documents is made by searching keywords on it, based on the rate of appearance which can later be used to classify them into topics. In this case Topic Modelling is useful for creating clusters of similar webs according to their text-plain content, so that the user can select only the webs which are likelihood related to the person-to-search, e.g. if the name of a known scientist is introduced, results related to sports and car sales can be discarded. Additionally when it comes to image filtering, a Face Recognition module can be implemented so that from a trust-worthy image (allegedly, the first image retrieved by Google Images search engine) which is chosen for obtaining the face landmarks of the person-to-search and among all the retrieved images from the HTML via web crawlers, only the images that match the same face landmarks as the identified one, are shown to the user.

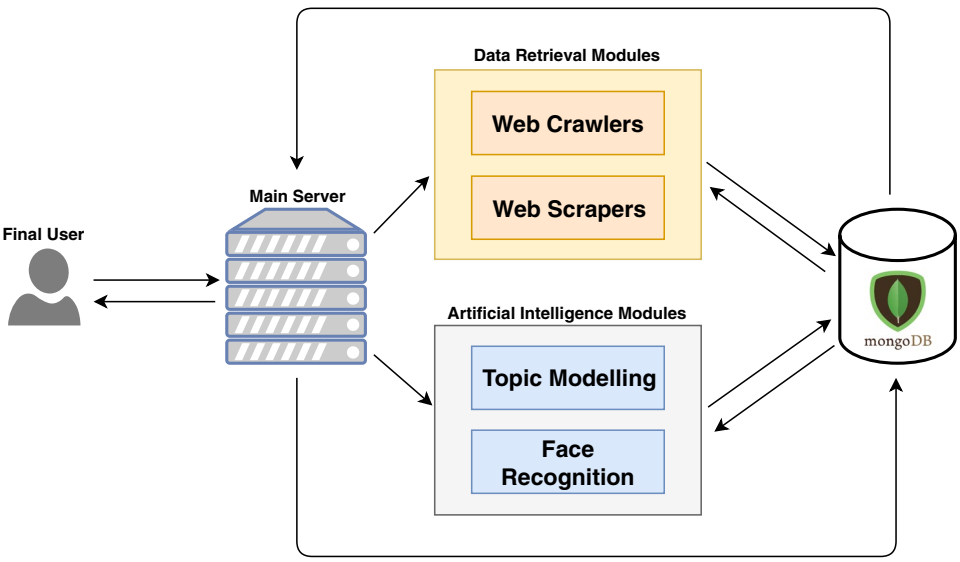

Fig. 2 Architecture of the proposed system

4 Implementation and case study

In this section, the modules that conform the proposed system are described individually and how do they operate and how they have been implemented. These modules have been depicted in Fig. 2 with the intention of providing readers a clear understanding on how each module contributes to the proposed system and more concretely to profile generation.

4.1 Web crawlers

Web crawling starts when the user sends a request to the system as to retrieve all the information indexed on the Internet from a person-to-search, sending as input parameters both the full name and related keywords; once the requests has been sent, the system triggers all the web crawlers in order to get all the information indexed on Google or Bing (Zhao et al. 2019). As web crawlers consume a lot of resources since they are bulk downloading the content of a lot of websites, the work has to be distributed among different threads to parallelize them, which can lead to a faster web crawling process as explained by Perisetla (2012), but with a bottleneck problem on the Internet connection issues. So on, the multi-threading distribution has been made in Java and, therefore, the web crawling too. Web crawlers have been developed using a Java library called Jsoup which is an open source library to parse, extract and manipulate data stored in HTML documents.

As there are huge loads of data stored on the Internet, in order to get the web crawlers get the best results or the most relevant ones according to both Google and Bing ranking system, an algorithm for automated web searches which combines the full name of the person-to-search and some keywords related to it, so that the results provided by search engines are more accurate in terms of relation to the target person-to-search. Nonetheless, the web scraping processes are described in more detail in Sect. 4.2.

The core objective of the developed algorithm is to combine the personal information and the related keywords of the person-to-search, both introduced as input to the system, so that more accurate information can be retrieved as search engines base their results on a website ranking and word coincidence of the introduced query with the indexed websites. Once the web crawling processes are completed, meaning by it their download and parse, a TF-IDF algorithm is applied, combined with a previous Stemming and stopword removal processes so that from the retrieved information of every resulting web new keywords can be generated based on TF-IDF results. So the generated keywords are the ones with more occurrences among all the filtered content of every web retrieved by the crawlers.

4.2 Web scrapers

As shown in Fig. 1 the web scraper instances are launched once the web crawling process finishes, as web scrapers take HTMLs as input. Web scraping is the process involves selecting the content from the HTML elements nested in the tree-structure of every HTML file. So on, web scrapers have been used for both general scraping of all the plain-text from any website by identifying HTML tags and getting their content, and also for retrieving data from a group of selected sources. Either web scrapers or APIs can be used in order to retrieve all the information available on a selected source, as the one on social networks like Twitter, Facebook, LinkedIn, etc. In this case web scrapers have been developed in Java using Jsoup just as web crawling, since these process were distributed in threads and each unique process involved both retrieval and its consequent scraping. Additionally, regular expressions have been used to retrieve certain content from which its structure was known so to tag it depending on the matched regular expression, e.g. mails whose structure is clear and can be properly defined.

Due to the amount of personal information that people rely on social networks, they can be reliable sources that can provide useful information when generating a personal profile. Additionally, some sources can be selected to scrape them via analysing their inner HTML and, after understanding how data is structured and what information is needed, retrieving its information. This selected sources need to be reliable and consistent so to their content. Like Drucker (2014) presents, *"to innovate is to find new or better uses to the resources that we already have"*, meaning that if a resource that matches our needs already exists, it is better to use it or improve it rather than creating something new, which justifies the use of APIs instead of developing Web Scrapers or API Wrappers.

Thus Web Scraping is used to retrieve information from the inner HTML of a selected source, in this case just social networks information is scraper through both APIs and web scrapers. Once the information is scraped and filtered it is inserted in a NoSQL database, because as the information is retrieved from a lot of different resources it does not have a defined structure. The filtering process is made once all the websites have been retrieved by the web crawlers and their plain-text content has been scraped via web scrapers, so on based on keywords, all the retrieved websites that have nothing to do with the person-to-search as they do not have any of the keywords related, they are discarded and classified as useless. After filtering by keywords, all the websites and their attached information is then stored on a NoSQL database.

4.3 Artificial intelligence for topic modelling

In this section, we present an algorithm which is responsible for selecting the most relevant websites by clustering the gathered texts. It contributes to filtering the collected data and provides a basic sum up of the main topics. The program has been written in Python and the Scikit-learn library has been used, the database is MongoDB.

The process of clustering is described below:

1. *Preprocessing the plain text* The raw inputs are transformed so that only the words which contribute with the biggest amount of information are used.

   The text of each web is divided into an array of words, all letters are transformed to lowercase, and stop words are removed. As a consequence, words are embedded into a homogeneous structure, and words which cannot provide any information are dismissed. The NLTK (Natural Language Toolkit) stop words dictionary (Perkins 2010) has been used in our implementation, albeit any other dictionary is equally valid.

   Afterwards, the 1000 words with the maximum number of occurrences are selected and stored. This process is used for extracting the most important features of the corpus and reducing the future processing time.
2. *Clustering* Several web groups are formed, and the words that best define each cluster will be selected.

   Firstly, the TF-IDF (term frequency-inverse document frequency) algorithm is used for transforming the words into numerical variables and our texts are consequently converted to matrices. Its good performance has been observed in Ramos (2003) and a detailed description of its implementation can be found in this paper as well.

   Eventually, the k-means algorithm is applied in order to the create clusters from the previously processed data. Subsequently, the closest words to the centroid of each cluster can be extracted, as they will provide a good sum up of the elements of each cluster (Singh et al. 2011). In addition, words with less than three letters or with all equal letters are not extracted, since they do not provide any valuable information neither about their text nor about their cluster. As a result, the information-to-noise ratio is increased and the output is easily understandable. An example of the achievements of this technique can be found in Fig. 3.

4.4 Artificial intelligence for face recognition

This part of the system is responsible of selecting images of the person-to-search, and consists in identifying their face from among all the images retrieved by the crawlers and stored into the NoSQL database. The applied deep learning algorithm is based on CNNs (convolutional neural networks), as their performance has been proven superior to previous techniques such as the presented in Liyew (2017). Furthermore, the use of SVMs and CNN together has been found to provide a better performance as shown by Mori et al. (2005). The algorithm is based on dlib's state-of-the-art face recognition, implemented in the Python library *face_recognition*. This model obtained an accuracy of 99.38% in the 'Labeled Faces in the Wild' benchmark.

Fig. 3 Wordcloud of the two most important clusters. Source texts are in Spanish

The process of face recognition is described in detail below:

1. *Face identification* We make use of the HOG algorithm due to its good performance for human image detection as presented by Dalal and Triggs (2005).

   The process is as follows: we convert the image to gray-scale and calculate the gradient of each pixel. As a result, a common ground is created for all images and changes in brightness do not affect the algorithm anymore. Afterwards, we store the gradients in an array, divide it into 16×16 pixel squares and select the direction of the greatest gradients of each square. Finally, we use a trained linear SVM for finding face patterns as previously described in Soentpiet et al. (1999). The faces are extracted from the original image and stored.
2. *Placing and projecting faces* We make use of the face landmark estimation algorithm, as formulated by Kazemi and Sullivan (2014). This algorithm ensures that the positions the faces are similar in all images. 68 special points (the so-called 'landmarks') are found, and affine transformations are used to center the mouth and eyes as much as possible. The resulting image is stored.
3. *Codifying faces* We make use of deep learning algorithms to find 128 unique measures of each face. A CNN trained by OpenFace is used to extract these features (Amos et al. 2016), and positions of such measures are stored.
4. *Matching faces with the codified image* A SVM is trained to classify images on the basis of their similitude to the codified image. This results in a positive or negative

evaluation. An example of positively evaluated images is shown in Fig. 4.

5 Results

This section outlines the results obtained by the developed system. Firstly, the visual dashboard is presented, which has been designed in order to make it easier for the user to understand the performed analysis of the retrieved information (Sect. 5.1). Then an analysis of the performance and accuracy of the developed system is presented below in Sect. 5.2.

5.1 Data visualization

Data visualization is a relevant aspect as it is necessary to present the obtained results to the user in a clear and concise way, as the output information of the system is structured as a JSON file which is not clear enough for user-level. So on, a visualization platform has been created in a Node.js server, which is an open source JavaScript runtime environment for server-side scripting, combined with with Angular, a framework for web application design, and communicated using Express.js, which is a web application framework for Node.js which has been used as an API between the server-side and the client-side. Previously mentioned process which were both web crawlers and web scrapers and AI techniques have been executed as processes in a pipeline where the conclusion of each process led to an entry point to the next one as described in Fig. 1 where the stored information in MongoDB will later be analysed via the different AI techniques to produce an unique result.

Thus the resulting information of the system when searching for a person is visualized on a dashboard as shown in Fig. 5. The dashboard takes part on all the process of the system as it presents the form which needs to be filled in order to look for the person-to-search with name, surname and related keywords (at least one keyword is mandatory), while processing the dashboard waits until all the information is retrieved and filtered.

As shown in Fig. 5, once all the webs are retrieved and parsed via web crawlers and web scrapers, respectively, and then the retrieved information is filtered and analysed using Artificial Intelligence, the resulting generated profile as a JSON file is shown on the dashboard. In order to keep the persistence, the generated profile is then stored on a NoSQL database and so on shown on the dashboard.

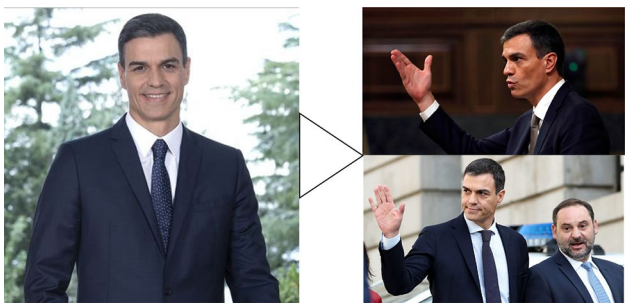

Fig. 4 Original image on the left, two of the matching images on the right

5.2 Performance and accuracy

As a lot of information is stored on the Internet, being able to retrieve just the one that is needed can be a quite hard task and, so on, the creation of a personal profile out of it. A lot of problems have been addressed associated with the filtering of information since the system aims to retrieve just the reliable and relevant information attached to a person and indexed on the Internet, which in this case is the main problem when trying to generate a personal profile.

In order to solve the problems attached to information filtering, an initial comparison between both search engines, Google and Bing, has been made so to identify which of them provides more results in less time when launching automated web searches. Our comparison study was based on determining which of both search engines provided more results in less time. As a case study, some profiles have been generated for spanish people indexed on the Internet, which the resulting time performance will depend on both the number of results per page scraped plus the amount of results depending on the Internet fame of the person-to-search. So on, Table 1 has been generated comparing the retrieval and scraping time of every Google and Bing search depending on the amount of results and number of retrieved results. It shows that Google outperforms Bing when it comes to results retrieved per search in less time, as a lot of searches are launched as a result of the combination of input parameters as already described. Anyways, quantity is not synonymous with quality so the amount of data retrieved does not mean that a search engine is better than another one; but in this case, as the main objective is to retrieve all the possible data stored on the Internet, so that a filtering and analysis can be made in order to generate a personal profile. The more data the more possible the system can analyse and filter without overfitting as all the retrieved information is directly or indirectly attached to the person-to-search.

However, one of the disadvantages of Google is that it imposes a restriction on the amount of automated searches per day; this is a risk factor and it is necessary to use proxy servers in order to avoid those limitations. The advantage of Bing over Google is that the monthly search limit is higher than Google's, but the indexed content on Bing is scarce. For this reason, Google remains the chosen search

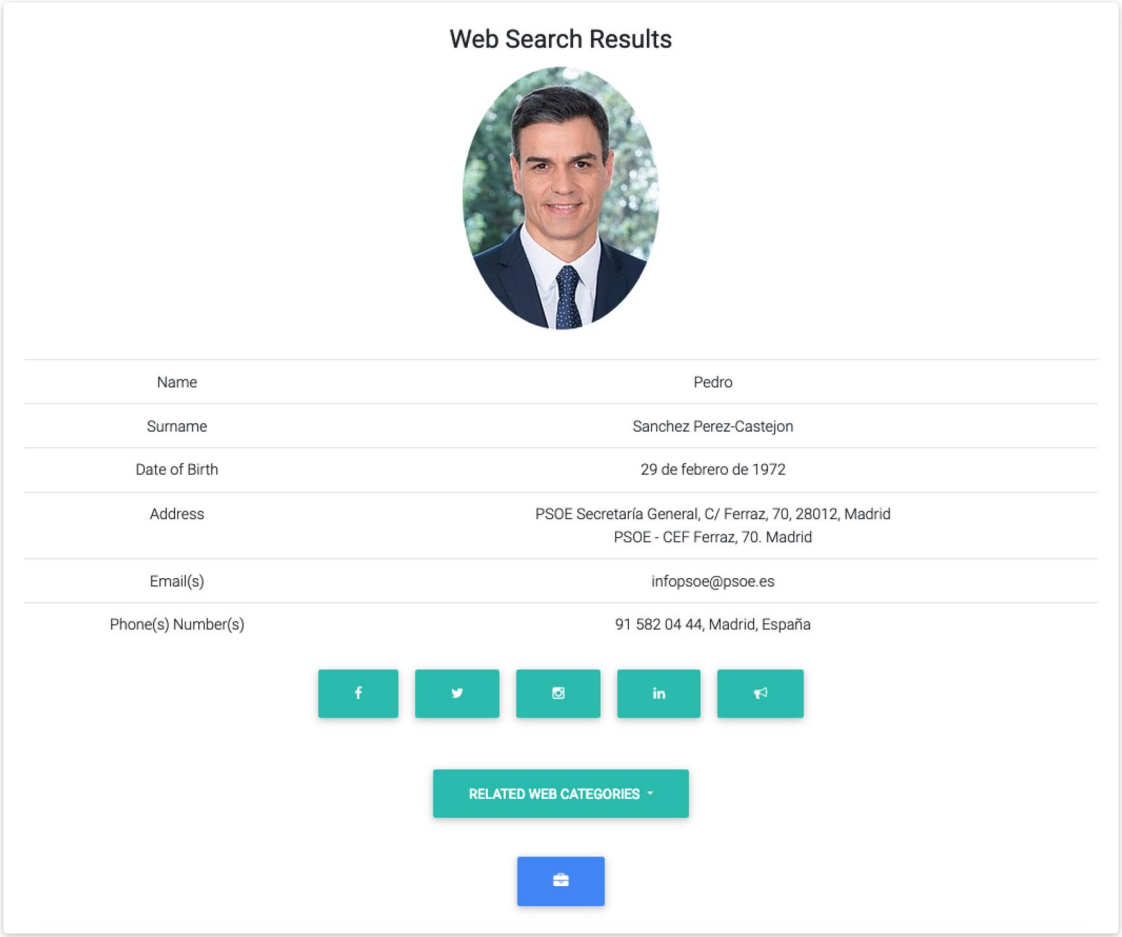

Fig. 5 Dashboard panel for data visualization

Table 1 Comparison between search engines performance

	Full name	Search engine	# of search results			
			3	5	7	10
World famous	Pau Gasol	Google	57.36	61.67	60	61.8
		Bing	75.24	74.28	73.35	70.66
	Pedro Sánchez Castejón	Google	63.82	58.68	65.52	65.01
		Bing	74.58	75.71	78.76	83.38
Well Known	Pablo Carreño	Google	59.7	57.25	61.77	60.41
		Bing	71.31	72.97	76.76	75.67
	Alex Márquez	Google	62.55	59.45	56.78	60.84
		Bing	72.3	81.81	72.74	76.41
Known	Juan Manuel Corchado	Google	60.56	59.91	60.62	58.28
		Bing	78.34	70.12	74.23	78.69
	Antonio Martínez Ares	Google	57.13	56.21	59.47	60.38
		Bing	70.8	67.18	71.87	68.13

engine for the proposed system; as to generate a personal profile a search engine that can provide as much data as possible in the shortest possible time period is needed.

As already mentioned above, as to generate personal profiles out of the indexed information on both Google and Bing the main need is to dispose of a lot of results in

the shortest time possible, but even though these factors are crucial for the proposed system as it is supposed to be a fast and reliable platform, the main attention focus should be on the quality of the results, ensuring accurate information and as a result a reliable personal profile based on it. So on, to check the accuracy of the proposed system, some searches were launched for known people in Spain, which implies that there was a previous knowledge that a lot of information related to them was indexed on the Internet, so as to do a supervised check on whether the proposed system generated profile was reliable or not. As tests have been supervised, two problems have been faced, both related to the amount of results that search engines provided concluding on a problem of over-information where a lot of information was displayed, and the opposite, a problem of less-information which leads to a bare information generated profile.

Conclude that due to the need of generated profiles by the proposed system of being supervised, tests can only be completed if there is a person supervising if they are accurate enough and provide relevant information. So on, the relevance of keywords should be point out as they provide more reliable results because of their relation with the person-to-search. Results retrieved via Web Crawling and Web Scraping by combining input parameters as to have better results and the further use of Artificial Intelligence in order to filter and analyse the retrieved information lead to the generation of a profile with consistent and reliable information of the person-to-search.

6 Conclusion and future work

In conclusion, our study has identified the most useful and reliable tools when it comes to web crawling and web scraping. Also different AI methodologies have been tested for both face recognition and topic modelling in order to determine which are the ones with the best accuracy and overall performance.

As a future work the migration of the developed system to a commercial cloud environment with no (practical) limits of the computing capacity is planned. Our goal is to release the tool for public use while controlling and limiting its use so that the system does not gets overloaded, so on, the limitations will depend on the amount of available servers and which computing capacity do they have. Once the system is launched, if anyone is going to use of the system for research purposes, they will need to contact the authors so that a personalised research plan can be adjusted to their needs. Moreover, once in the market, the system will continue improving and new functionalities will be added so to fit the needs of the target users.

References

Ali N, Bajwa KB, Sablatnig R, Mehmood Z (2016) Image retrieval by addition of spatial information based on histograms of triangular regions. Comput Electr Eng 54:539–550

Ali S, Rauf A, Islam N, Farman H, Khan S (2017) User profiling: a privacy issue in online public network. Sindh Univ Res J (Sci Ser) 49:1

Allahyari M, Pouriyeh S, Assefi M, Safaei S, Trippe ED, Gutierrez JB, Kochut K (2017) A brief survey of text mining: classification, clustering and extraction techniques. arXiv:170702919 (arXiv preprint)

Amos B, Ludwiczuk B, Satyanarayanan M et al (2016) Openface: a general-purpose face recognition library with mobile applications. CMU School of Computer Science, Pittsburgh

Bahrami M, Singhal M, Zhuang Z (2015) A cloud-based web crawler architecture. In: 2015 18th international conference on intelligence in next generation networks, IEEE, pp 216–223

Balduzzi M, Platzer C, Holz T, Kirda E, Balzarotti D, Kruegel C (2010) Abusing social networks for automated user profiling. In: International workshop on recent advances in intrusion detection, Springer, pp 422–441

Barbosa L, Freire J (2010) Siphoning hidden-web data through keyword-based interfaces. J Inf Data Manag 1(1):133

Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: Computer vision and pattern recognition, 2005. CVPR 2005. IEEE computer society conference on, IEEE, vol 1, pp 886–893

Dang NC, De la Prieta F, Corchado JM, Moreno MN (2016) Framework for retrieving relevant contents related to fashion from online social network data. In: International conference on practical applications of agents and multi-agent systems, Springer, pp 335–347

Davoodi E, Afsharchi M, Kianmehr K (2012) A social network-based approach to expert recommendation system. In: International conference on hybrid artificial intelligence systems, Springer, pp 91–102

Drucker P (2014) Innovation and entrepreneurship. Routledge, London

Falahrastegar M, Haddadi H, Uhlig S, Mortier R (2016) Tracking personal identifiers across the web. In: International conference on passive and active network measurement, Springer, pp 30–41

Jayaram D, Manrai AK, Manrai LA (2015) Effective use of marketing technology in eastern europe: web analytics, social media, customer analytics, digital campaigns and mobile applications. J Econ Financ Admin Sci 20(39):118–132

Jose B, Abraham S (2017) Exploring the merits of nosql: a study based on mongodb. In: 2017 international conference on networks and advances in computational technologies (NetACT), IEEE, pp 266–271

Kandias M, Mitrou L, Stavrou V, Gritzalis D (2017) Profiling online social networks users: an omniopticon tool. IJSNM 2(4):293–313

Karidi DP, Stavrakas Y, Vassiliou Y (2018) Tweet and followee personalized recommendations based on knowledge graphs. J Ambient Intell Humaniz Comput 9(6):2035–2049

Kasar MM, Bhattacharyya D, Kim T (2016) Face recognition using neural network: a review. Int J Secur Appl 10(3):81–100

Kazemi V, Sullivan J (2014) One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1867–1874

Liyew MBT (2017) Applying a deep learning convolutional neural network (CNN) approach for building a face recognition system: a review. J Emerg Technol Innov Res 4(12):1104–1110

Marx M (2014) The extension and customization of maltego data mining environment into anti-phishing system. South Africa

Moreno A, Redondo T (2016) Text analytics: the convergence of big data and artificial intelligence. IJIMAI 3(6):57–64
Mori K, Matsugu M, Suzuki T (2005) Face recognition using SVM fed with intermediate output of CNN for face detection. In: MVA, pp 410–413
Nguyen THT, Dinh DT, Sriboonchitta S, Huynh VN (2019) A method for k-means-like clustering of categorical data. J Ambient Intell Human Comput 20:1–11
Olston C, Najork M et al (2010) Web crawling. Found Trends Inf 4(3):175–246
Perisetla KK (2012) Mutual exclusion principle for multithreaded web crawlers. Edit Preface 3:9
Perkins J (2010) Python text processing with NLTK 2.0 cookbook. Packt Publishing Ltd, Birmingham
Ramos J (2003) Using tf-idf to determine word relevance in document queries. Proc First Instr Conf Mach Learn 242:133–142
Rivas A, Martín L, Sittón I, Chamoso P, Martín-Limorti JJ, Prieto J, González-Briones A (2018) Semantic analysis system for industry 4.0. In: International conference on knowledge management in organizations, Springer, pp 537–548
Roy D, Ganguly D, Mitra M, Jones GJ (2016) Representing documents and queries as sets of word embedded vectors for information retrieval. arXiv:160607869 (arXiv preprint)
Shah JH, Sharif M, Yasmin M, Fernandes SL (2017) Facial expressions classification and false label reduction using LDA and threefold SVM. Pattern Recogn Lett 20:20
Singh VK, Tiwari N, Garg S (2011) Document clustering using k-means, heuristic k-means and fuzzy c-means. In: Computational intelligence and communication networks (CICN), 2011 international conference on, IEEE, pp 297–301
Soentpiet R et al (1999) Advances in kernel methods: support vector learning. MIT Press, London
Spiekermann S, Acquisti A, Böhme R, Hui KL (2015) The challenges of personal data markets and privacy. Electron Mark 25(2):161–167
Sun S, Gong J, Zomaya AY, Wu A (2017) A distributed incremental information acquisition model for large-scale text data. Cluster Comput 20:1–12
Vasanthakumar G, Shenoy PD, Venugopal K (2016) Ptib: profiling top influential blogger in online social networks. Int J Inf Process 10(1):77–91
VenkateswarLal P, Nitta GR, Prasad A (2019) Ensemble of texture and shape descriptors using support vector machine classification for face recognition. J Ambient Intelli Human Comput 20:1–8
Zhao X, Zhang W, He W, Huang C (2019) Research on customer purchase behaviors in online take-out platforms based on semantic fuzziness and deep web crawler. J Ambient Intell Human Comput 20:1–15