Web Content Mining for Alias Identification: a first step towards suspect tracking

Tarique Anwar*, Muhammad Abulaish*† and Khaled Alghathbar*
*Center of Excellence in Information Assurance, King Saud University, Riyadh, Saudi Arabia
†Jamia Millia Islamia (A Central University), New Delhi, India
Email: {tAnwar.c, mAbulaish, kAlghathbar}@ksu.edu.sa

***Abstract*—In this paper, we present the design of a web content mining system to identify and extract aliases of a given entity from the Web in an automatic way. Starting with a pattern-based information extraction process, the system applies n-gram technique to extract candidate aliases. Thereafter, various statistical measures are applied to identify feasible aliases from them. The extracted aliases can be used to generate profiles of suspects and keep track of their movements on the Web using different identities.**

***Index Terms*—Web content mining; Cyber security; Alias identification; Suspect profiling; Web monitoring.**

I. Introduction

Due to easy accessibility and availability of Internet, the World Wide Web (WWW) has become one of the soft and cheap platforms for most of the terrorist organizations to spread propaganda around the globe. The WWW is also being used by such organizations to provide training, recruit people and fund-raising. Moreover, with the prosperity of Internet and Web 2.0, many social networking and social media sites are emerging and people can easily connect to each other in the Cyber space. Thus, the exponential growth in interactions of people on the Web has made it ever since the largest repository of data and it has become very easy for an end user to access information about any matter of concern. For example, one can get the details about any person or place using web search engines. But, on the other hand, due to presence of tech-savvy non-social elements in our society, we have become heavily prone to thefts or illegal access of our private data or other issues making our data totally insecure. This insecurity calls urgently for information security mechanisms, and has drawn a serious attention of researchers towards Cyber security. As compared to other sources, due to its easy access and various security flaws in it, Web itself has become the most active medium for Cyber crimes. Terrorist and other criminal groups are using Web as a tool for a number of cyber crimes related to data thefts, hacking of on-line bank accounts etc. In these situations, the tracking of suspected intruders and their activities can be helpful to predict their future target attacks. But, one of the bottlenecks that hinders the tracking of suspects over Web is the use of aliases to represent themselves to their friends and hiding their original identity from rest of the population. It has been observed to be very common that in addition to the real name, people on the Web are being represented by multiple alternate names. Sometimes, these alternate names are their nicknames or titles whereas, at other times they are the names being used by different groups or networks of people the person is related to. For example: *Albert Einstein* on the Web is also known as the *father of modern science*, *Albert* and *Alby*. Sometimes, a person uses a secret name to represent himself to his friends but hiding his original identity from rest of the public. These multiple alternate names being used are called as *alias names* or sometimes also called as *mnemonic names*. Searching for information about people on the Web is a very common activity of most of the Internet users. According to Guha and Garg [1], nearly 30% of search queries on the Web account for name of any person.

In this paper, we present the design of a pattern-based web-content mining system to identify and extract the alias names for a given entity in an automatic way. Starting with the set of lexical patterns for aliases presented in [2] the system first identify relevant web pages using Google API (Application Programming Interface) and store them on local machine. Thereafter, n-gram technique is applied to generate candidate aliases from them. Finally, feasibility analysis using various similarity measures is applied on the candidate aliases to identify feasible ones.

The rest of the paper is organized as follows. Section II presents a review of related works. In section Section III, we present the functional detail of the proposed system. Section IV presents the experiment setup and results. Finally, we conclude the paper in section V with future enhancements of the proposed system.

II. Related Work

There are two problems being faced in profile summary generation. First, a single entity is designated by multiple names, whereas the other one is just the reverse of this, i.e., various different entities are designated by the same name. In second case, an entity could have two or more different meanings and the area that deals with this problem is called *named entity disambiguation* and a number of techniques exist in literature [3]. The first case needs to find out the several alternate names that are being used on the Web to represent the real name. Two analogous problems related to this problem are *named entity recognition* [4] and *cross-document co-reference resolution* [5]. In case there are two different entity names known to us from different documents, co-reference resolution

finds out if entity names are inter-related in any way. In [6], Hokama and Kitagawa have proposed a pattern-based web mining system that deals with the problem of alias mining from web documents written in Japanese language. They assumed that an alias whenever exists with the real name on Web, it follows a lexical pattern in which these are associated with a Japanese text pronounced as *"koto"* and translated into English as *"be called"*. In [2], a set of patterns along with their f-score values for alias appearance in English language texts is identified. They have used a dataset of real names as well as their aliases for fifty English persons, fifty English places and fifty Japanese persons to determine the lexical text pattern that usually exist in between a real name and its alias. Through this they found *f-scores* of the patterns by five-fold cross validation, and used them as search queries to extract candidate aliases. For ranking, a support vector machine (SVM) is trained on a set of 23 features. In contrast to their approach, we propose a light weight feasible approach for ranking candidate aliases statistically.

III. Proposed System

In this section, we present the functional detail of the proposed system. We have used the patterns learned in [2] to target only those pages that might be a candidate for containing an alias. Thereafter, we have applied n-gram technique to generate candidate aliases from them. For ranking, our statistical strategy integrates three diverse but salient properties (*associativity*, *similarity* and *co-occurrence*) of an alias to a single ranking value. We analyzed the dataset used in [2] along with some data of real names and their aliases collected from different sources. On analysis, we found that for a famous person (e.g. *David Hasselhoff*) there exist a large number of pages on the Web and their names are found as anchor texts a number of times. The co-occurrence graph used by Bollegala *et al.* [2] works fine for them. However, it may be the case that the person for whom we are interested to find the aliases, is an obscured one, such that there does not exist much web pages and rarely does any name as anchor text (e.g., different forums where some discussions are going on, complain logs, etc.). Therefore, when we are in need to find aliases of a suspected intruder, who is not necessarily a famous person, the co-occurrence graph fails to contribute to their ranking using SVM. Analyzing all these diverse conditions, we identified the above-mentioned parameters to establish the *aliasness* for a given candidate alias. Figure 1 presents the architecture of the proposed alias mining system. Further details are presented in the following sub-sections.

A. Target Web Page Retrieval

This module accepts the real name as input and uses the Google API to retrieve relevant web pages. Using a sample dataset of 50 persons Bollegala *et al.* [2] have discovered a set of 25 patterns for English texts. We have used only top ten *f-scored* patterns out of which five are given in table I. But, in contrast to [2] in which only the text snippets returned by the search engine are used, we have considered whole web page as using only text snippets is not suffice to consider the multiple occurrences of patterns in a web page.

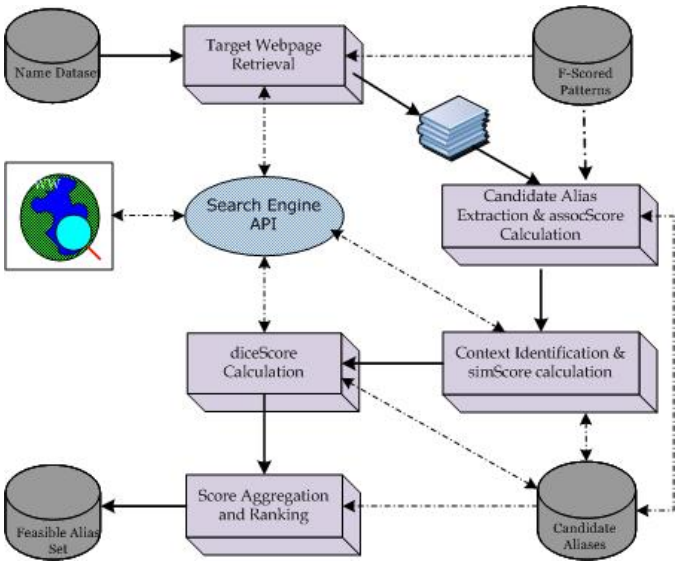

Fig. 1. Architecture of the proposed web content mining system

TABLE I
F-Scored Patterns [2]

Pattern Id	Pattern-Based Querries	F-Score
p_1	SearchQuery(*realName, aka, **)	0.335
p_2	SearchQuery(**, aka, realName*)	0.322
p_3	SearchQuery(*realName, better known as, **)	0.310
p_4	SearchQuery(*realName, alias, **)	0.286
p_5	SearchQuery(*realName, also known as, **)	0.281

B. Candidate Alias Extraction and Association Score Calculation

From each page, only those sentences containing the associated patterns are identified and considered for further processing. This improves the processing efficiency drastically. After cleaning, sentences are divided into record-size chunks on the basis of special characters like newline, full stop, etc. Thereafter, each chunk is subjected for n-gram generation where the value of n varies from 1 to 5. Depending on the position (left-most or right-most) of wildcard character, *, in query pattern n-grams are generated either from left-side texts or from right-side texts. All n-grams either having a complete match in the list of stop words or beginning or ending with a stop-word, numeric character or a special character are filtered out. Since a candidate alias, a, may exist with multiple patterns, an aggregated association score, $assocScore(a)$ is calculated using equation 1 in which i varies over the number of patterns associated with a, $F\text{-}score(p_i)$ represents the f-score value of the pattern p_i and $freq(a_i)$ represents the number of times a occurs in association with p_i.

$$assocScore(a) = \sum_i (F\text{-}Score(p_i) \times freq(p_i)) \quad (1)$$

C. Context Identification and Similarity Score Calculation

For each candidate alias and real name, we generate a small world to identify their contexts and then use them to find context similarity between an alias name and the real name. For this, we represent the contexts of both alias and real name using vector space model. To generate small world, a query is searched on the Web for both real name and candidate aliases. The web pages returned by the search engine are chunked and cleaned using the same process explained earlier in this paper. From each chunk, w neighboring unigrams from both left and right sides of the query string are collected and added to the set of small world. Then a combined set of small world is generated using set-theoretic union operation. This combined set is used to generate vector-space representation of the real name rn and an alias a as V_{rn} and V_a respectively. The similarity between V_{rn} and V_a is calculated using Cosine similarity function given in equation 2.

$$simScore(a) = \frac{\sum_{i=0}^{n} V_{rn}(i) \times V_a(i)}{\sqrt{\sum_{i=0}^{n} (V_{rn}(i))^2 \times \sum_{i=0}^{n} (V_a(i))^2}} \quad (2)$$

D. Dice Score Calculation

The *dice score* measure is applied to boost up the candidate aliases that are more frequently found in web pages along with their real names, as compared to those with less frequent co-occurrences. Although, it is also a type of associativity measure between them but, the prime focus here is not their association because there does not exist any specific pattern or link to create a strong relationship. Despite their placement fashion or their lexical and syntactic structure, it just depends on the counts of their co-occurrences on the same page. We performed experiments with several different types of score values, but finally arrived at *Dice Coefficient* producing the best results. This measure is also used by Bollegala *et al.* [2] as one among several association measures. The dice score value for a candidate alias a and its real name rn is calculated using equation (3).

$$diceScore(a) = \frac{2 \times hits(rn\ AND\ a)}{hits(rn) + hits(a)} \quad (3)$$

E. Score Aggregation and Ranking

The candidate aliases extracted in section III-B are those that somehow possess some key property in them to get marked as aliases, for which they can also be called as *"to be aliases"*. However, due to the unstructured nature of Web, we can also be misleading in extracting aliases by the previous step. That is why we employ here additional statistical measure to determine the candidacy of an alias and rank them in accordance with their aggregate relevance value. Therefore, all different scores calculated in previous sections are aggregated to generate a single score value to determine the most feasible and promising aliases. The aggregate value that can be termed as *aliasness* of a candidate is calculated using equation 4 in which AS stands for *assocScore*, SS stands for *simScore* and DS stands for *diceScore*.

$$Aliasness(a) = AS(a) \times SS(a) + SS(a) \times DS(a) + DS(a) \times AS(a) \quad (4)$$

Finally, based on this aggregated value of *aliasness*, candidate aliases are ranked and r top aliases are declared as feasible and most promising aliases that are being used on the Web as an alternate to the persons' real name.

IV. Experimental Setup and Results

To evaluate our proposed system, we have used the publicly available dataset used in [2]. This consists of 50 English person name, 50 Japanese person names and 50 English place names of US. As we are concerned only with the aliases of English person names, we ignored the other data in the set. We have set the values of n for candidate alias extraction to 5, w for context identification to 10, and a maximum of 200 web pages for each pattern. After applying the filtering and cleaning process discussed earlier, candidate aliases are generated using n-grams technique. Then, for each candidate alias we have generated three different type of scores as described in section III. Finally, they are ranked after aggregating their scores. To measure the accuracy of the proposed system we have used *Mean Reciprocal Rank* (MRR) metric defined using equation 5 in which G is the set of gold standard and $rank_i$ is the rank of i^{th} alias of G in the ranked list of extracted aliases. The average MRR value is found to be as 0.58.

$$MRR = \frac{1}{|G|} \times \sum_{i=1}^{|G|} \frac{1}{rank_i} \quad (5)$$

V. Conclusion

In this paper, we have proposed a light-weight web content mining system to identify aliases of a person on the Web. The proposed method is computationally efficient and seems to work for names of concealed persons, for whom there are less number of hits on the Web and there are negligible occurrences as anchor text. Presently, we are exploiting the learned aliases to develop a suspect tracking and monitoring system.

References

[1] J. Artiles, J. Gonzalo, and F. Verdejo, "A testbed for people searching stategies in the www," in *Proceedings of SIGIR'05*, 2005, pp. 569–570.

[2] D. Bollegala, T. Honma, Y. Matsuo, and M. Ishizuka, "Identification of personal name aliases on the web," in *Proceedings of 17th International Conference on World Wide Web WWW'08*, Apr. 2008.

[3] C. Whitelaw, A. Kehlenbeck, N. Petrovic, and L. Ungar, "Web-scale named entity recognition," in *Proceedings of 17th ACM Conference on Information and Knowledge Management (CIKM 2008)*, 2008.

[4] G. D. M. Rennie and T. Jaakkola, "Using term informativeness for named entity detection," in *Proceedings of ACM SIGIR 2005*, 2005.

[5] A. Bagga and B. Baldwin, "Entity-based cross document coreferencing using vector space model," in *COLING'98*, 1998, pp. 79–85.

[6] T. Hokama and H. Kitagawa, "Extracting mnemonic names of people from the web," in *Proceedings of 9th Intlernational Conference on Asian Digital Libraries (ICADL'06)*, 2006, pp. 121–130.