The Uncertainty of Identity Toolset: Analysing Digital Traces for User Profiling

Muhammad Adnan
Department of Geography,
University College London, UK
m.adnan@ucl.ac.uk

Antonio Lima
School of Computer Science,
University of Birmingham, UK
a.lima@cs.bham.ac.uk

Luca Rossi
School of Computer Science,
University of Birmingham, UK
l.rossi@cs.bham.ac.uk

Suresh Veluru
School of Engineering and
Mathematical Sciences,
City University London, UK.
suresh.veluru.1@city.ac.uk

Paul Longley
Department of Geography,
University College London, UK
p.longley @ucl.ac.uk

Mirco Musolesi
School of Computer Science,
University of Birmingham, UK
m.musolesi@cs.bham.ac.uk

Muttukrishnan Rajarajan
School of Engineering and
Mathematical Sciences,
City University London, UK
r.muttukrishnan@city.ac.uk

ABSTRACT

People manage a spectrum of identities in cyber domains. Profiling individuals and assigning them to distinct groups or classes have potential applications in targeted services, online fraud detection, extensive social sorting, and cyber-security. This paper presents the Uncertainty of Identity Toolset, a framework for the identification and profiling of users from their social media accounts and e-mail addresses. More specifically, in this paper we discuss the design and implementation of two tools of the framework. The Twitter Geographic Profiler tool builds a map of the ethno-cultural communities of a person's friends on Twitter social media service. The E-mail Address Profiler tool identifies the probable identities of individuals from their e-mail addresses and maps their geographical distribution across the UK. To this end, this paper presents a framework for profiling the digital traces of individuals.

Categories and Subject Descriptors

K.6.5 Security and Protection

General Terms

Algorithms, Design, Human Factors

Keywords

Identity, Online social networks, Twitter, Geographic distribution, Substring matching, Suffix tree.

1. INTRODUCTION

Identity is a complex concept, reflecting issues of privacy and ownership, across real and cyber media [1]. We use different identities in the cyber-space to communicate with friends, family, and work colleagues. These identities are used on different social media services, blogs, instant messaging, and e-mail services. Current research on identity (for example SuperIdentity [2] and Uncertainty of Identity [3]) aim to combine a rich set of measures from real and cyber domains as a way to identify and authenticate individuals as legitimate users of different online services. In the past years, we have witnessed a rapid growth of the use of online services for various purposes e.g. online shopping, bank transactions, targeted online marketing, and increased use of the social networking services. The increased use of these online services has raised issues related to cyber-crimes, identity frauds, and hacking.

This paper presents a framework for the identification and profiling of individuals from their social media accounts and e-mail addresses. This paper presents two tools that are part of the identity management toolset we are currently developing. This identity management framework might be useful for profiling individuals for targeted marketing, online frauds, cyber security, and extensive social sorting. The first tool, the Twitter Geographic Profiler builds a map of the ethno-cultural communities of a person's friends. That is, it determines the distribution over the set of possible ethno-cultural groups of the friends of a given individual. This tool integrates information from two sources, namely Twitter and Onomap [4]. Onomap is a name classification system which assigns users into different cultural, ethnic and linguistic groups on the basis of their forename and surname pairs. Onomap is based on the cluster analysis of names extracted from electoral registers and telephone directories from different countries. The Onomap classification was created from a version of the 2007 Electoral Register for the United Kingdom. The register contains information about every individual who is eligible (or, in the case of 17 year olds, about to become eligible) to vote in UK or European elections, plus non-voters and other adults identified from consumer dynamics files by the data supplier (CACI Ltd., London [5]).

The second tool, the E-mail Address Profiler uses a database of family names to extract probable identities of individuals from their e-mail addresses. In most cases, an e-mail address encapsulates some kind of identity information, i.e., forename or surname. A forename or surname is a statement of the bearer's cultural, ethnic, and linguistic identity [4]. The tool uses an efficient approach to identify the presence of surnames as substrings in an e-mail address. Then, it predicts the probable ethnicity, and maps the geographical distribution of the surname in the UK. For this purpose, this tool uses data from three different data sources, namely Onomap, Worldnames [6], and the 2007 Register of Electors for the United Kingdom.

Worldnames is an online service which maps the geographical distribution of a searched surname around 26 different countries of the world. It was created by using the names data extracted from the telephone directories and electoral registers from different countries.

This paper is organized as follows. Section 2 of the paper describes the Twitter geographic profiler tool and discusses the use and privacy implications of the tool. Section 3 describes the E-mail Address Profiler tool and the underlying suffix tree construction algorithm. Finally, section 4 concludes the paper.

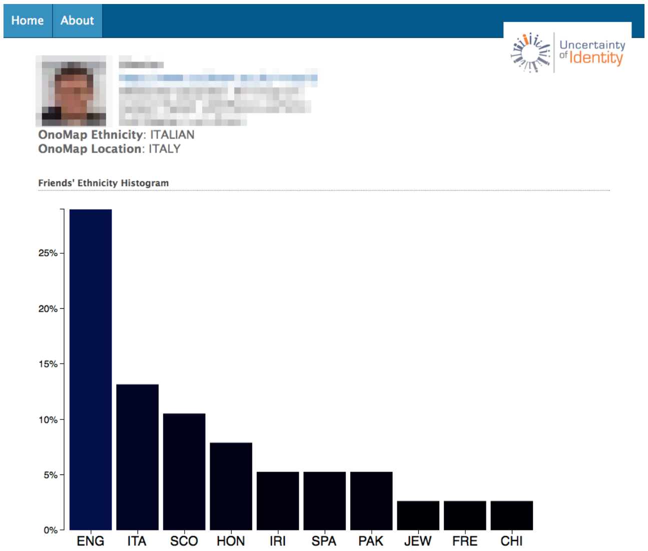
2. TWITTER GEOGRAPHIC PROFILER

This tool builds a map of the ethno-cultural communities of a person's friends. That is, we want to determine the distribution over the set of possible ethno-cultural groups of the friends of a given individual. To this end, we integrate information from two sources, namely Twitter and Onomap. Note, that the same ideas can be applied to data collected from other Online Social Networks (OSN), such as Facebook or Foursquare[1]. However, different OSN capture social interactions around different and sometimes specific themes, i.e., Foursquare’s venues. In this work, we decide to focus on Twitter data because of the general context of the interactions, i.e., they are not restricted to a specific theme or interest, and because, unlike Facebook, information is easily accessible through the Twitter API[2].

More specifically, given the Twitter username of the person being analysed, we download the list of *(surname, forename)* pairs of his or her friends. We then map this list of names to a list of ethno-cultural groups, according to the classification of Onomap. We also map the surnames to the most probable countries of origin. With these lists to hand, we estimate respectively the distribution of a user’s friends over the set of possible ethno-cultural groups and over the set of countries. In the following subsections we report the implementation details of the tool and its applications and implications in terms of users' privacy. Finally, note that the social graph of Twitter is directed, in the sense that the friendship relation is not necessarily reciprocated. As a consequence, there are two lists associated with each user, one for the accounts that the user is *following* and one for the accounts that follow the user, i.e., his or her *followers*. In this work, we consider the first as representing the list of a user's friends. Subsection 2.1 describes the system implementation of this tool and Subsection 2.2 discusses the use of the tool and its privacy implications.

2.1 System Implementation

Given the Twitter username of an individual, we first probe the list of his/her friends’ ids using the *GET friends/ids* method. As most of the methods in the API, the number of requests that can be performed in a certain time interval is bounded. More specifically, we are only allowed to send 15 requests every 15 minutes. Note also that with each query we can only get up to 5000 user ids. However, we find that generally one single request is sufficient to download the complete list of friends. Given the list of ids, we use the *GET users/lookup* method to fetch the *(surname, forename)* pair associated with each id. This returns up to 100 users profiles, given a list of ids as input. Note that request rate of the *GET users/lookup* method is currently limited to 180 requests every 15 minutes. We should stress that these rate limitations do not prevent us to parse the complete list of friends of a user, as the distribution of the number of accounts followed by Twitter users has been shown to be approximately a power law distribution [7]. As a consequence, the majority of the users actually follow a limited number of profiles, which are then accessible even with the rate limitation in place.



OnoMap Ethnicity: ITALIAN
OnoMap Location: ITALY

Friends' Ethnicity Histogram

Figure 1: Screenshot of the Twitter Geographic Profiler. The bottom part of the screen shows the histogram of the Twitter user's friends ethno-cultural groups.

With the list of *(surname, forename)* pairs to hand, we query Onomap to get the ethno-cultural classification associated with each *(surname, forename)* pair, and the *SearchSurnameTopCountries* method to get the list of the countries where an instance of a given surname was observed. Each element of the latter list is attributed with the relative frequency of the corresponding surname in that country, so as to take into account differences in population counts. Given this ranking, we then classify each surname as originating from the corresponding highest ranked country. As for *SearchEthnicity*, the method returns the most probable classification for both the surname and the forename, as well as the overall classification for the pair. Finally, in order to query the Onomap API in an asynchronous way, we make use of the grequest[3] package.

[1] Foursquare is a Location-Based Social Network (LBSN). LBSNs are based on the concept of check-in, where a user can register in a certain location and share this information with friends. Moreover, the user can leave recommendations and comments about the visited venues.

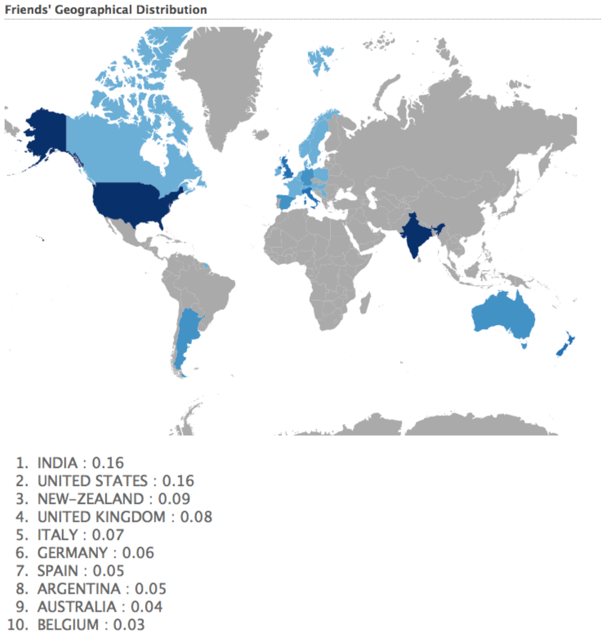
[2] https://dev.twitter.com/docs/api

[3] https://github.com/kennethreitz/grequests

However, we decide to limit the number of simultaneous asynchronous requests to 50 in order to avoid congesting Onomap's server.

Once the entire list of friends *(surname, forename)* pairs has been parsed, we can easily estimate the distribution over the set of possible ethno-cultural groups and over the set of countries of the Twitter user's friends. Figs. 1 and 2 respectively show the histogram of the ethno-cultural groups and a map visualising the countries of origin of the friends of a sample user. In the map a darker (brighter) color denotes a higher (lower) probability of having a friend originating from that country.

Note that, when we extract the *(surname, forename)* pairs using the *GET users/lookup* method, a filtering system needs to be put in place to discard invalid strings. In fact, while in other OSN such as Facebook the user is forced to enter the surname and the forename in two separated fields, in Twitter the users are required to enter their name (or any alternative identifying string) in a single *Username* field. As a consequence, we need to parse the username string to separate it into its constituent tokens. Then, we need to apply some heuristic to detect the *(surname, forename)* pair among the extracted tokens. In this work we mark as *invalid* any string that is composed of a single token. If this is the case, we skip the profile of the corresponding friend.



1. INDIA : 0.16
2. UNITED STATES : 0.16
3. NEW-ZEALAND : 0.09
4. UNITED KINGDOM : 0.08
5. ITALY : 0.07
6. GERMANY : 0.06
7. SPAIN : 0.05
8. ARGENTINA : 0.05
9. AUSTRALIA : 0.04
10. BELGIUM : 0.03

Figure 2: Map showing the geographical origin of the Twitter user's friends' surnames as assigned by our tool. Below the map the user is shown a list of the top 10 countries with the respective frequency.

If the string contains two or more tokens, we take the first one to be the forename and the last one to be the surname. Moreover, when a *(surname, forename)* pair is sent to Onomap, an error message will indicate if the system is unable to parse the surname or the forename, or both. In the latter case we stop the computation and we proceed to parse the next *(surname, forename)* pair.

2.2 Discussion & Privacy Implications

Although relatively simple, the above tool can be used in a number of applications that leverage the ethno-cultural information of a person's friends. To start with, note that in addition to what described above we can query Onomap to classify the *(surname, forename)* pair of the Twitter user whose friends list is being analysed. Hence, given a large enough sample of users, we can estimate the average friendship distribution of a given ethno-cultural group.

This in turn can be used to measure the *multiculturality* of a given community. For example, one may compute the Shannon entropy of the ethno-cultural distribution of a community (or an individual) to get a readably interpretable measure of how *open* the community (the individual) is to other groups, in terms of bondings. Intuitively, the more peaked the distribution is, the lower is the Shannon entropy and the less prone a community is to bond with a large number of groups. Similarly, by computing the Jensen-Shannon divergence between the average distributions [8] and applying multidimensional scaling [9] to the resulting distance matrix one can embed these communities in the Euclidean space for the purpose of visualising and grouping similar ethno-cultural groups.

However, note that we expect the level of exposure to different ethno-cultural groups to vary across the geographical space. That is, on average a resident of London is likely to have friendships spanning a wider spectrum of communities rather than a resident of Swansea[4], due to the substantial mixture of ethnic groups living in London. As a consequence, the above analysis should be performed within a limited geographical space. Luckily, it has been shown that roughly 50% of Twitter users have a location assigned in their profile, and the vast majority of these locations are at town level [10], thus such an analysis would indeed be feasible.

Given the friendships distribution of a given ethno-cultural group, it is also possible to use outlier detection techniques [11] to identify individuals or group of individuals that stand out in terms of the ethno-cultural groups they bond with. Potentially, one can also infer the ethnicity of an individual whose name is unknown but for which a list of friend names is available.

To understand the extent of the privacy implications of our tool, we should stress that the default behaviour of Twitter is to set the profile of a user as *public*. Although the setting can be changed to *private*, thus making it impossible for our tool to operate on the profile, when testing our tool we did not encounter any private profile. Consequently, we can safely assume to be able to download the list of names of a user’s friends and perform our ethno-cultural profiling.

As for the limitations of the current implementation of our tool, we observed that the Twitter data contains a large amount of noise, which can considerably affect the results of the computation. The source of this noise is twofold. Firstly, the need of extracting the surname and forename tokens from a single string introduces unwanted uncertainty. In this sense, more sophisticated natural language processing techniques should be investigated to extract the correct *(surname, forename)* pair. Secondly, we note that a considerable number of accounts followed by the Twitter users is actually represented by news feeds, i.e., BBC, CNN, etc., celebrities, notable academics and

[4] http://www.swansea.gov.uk/index.cfm?articleid=44946

other popular figures. Thus, in a sense the Twitter social network reflects the interests of a user, rather than his or her friendships. Note however that this effect may be partly reduced by considering only reciprocated ties, *followers* of a user that are *followed* by that user.

3. E-MAIL ADDRESS PROFILER

This tool extracts probable identities of individuals from their e-mail addresses and maps their geographical distribution. In most cases, an e-mail address encapsulates some kind of identity information in the form of surname or forename as substring. These forenames and surnames reflect ethnic, geographic, cultural, and genetic structures of the individuals. This tool uses an efficient approach to identify surnames present as substring in e-mail addresses. The proposed approach builds a suffix tree for an e-mail address and checks against a database of surnames, the 2007 Electoral Register in this case, to identify which surname is present as substring in it.

Suffix tree data structures have been extensively used in natural language processing, bioinformatics, computational biology, and text mining [12]. Suffix trees solve a wide range of problems such as exact and inexact matching problems, substring problems, data compression, subsequence problems, longest common substring, string kernels, and circular strings. Recently, suffix trees have been used in surname correction and identification in a corpus of names [13] and e-mail address categorization based on semantics of surnames [14].

In general, a suffix tree is a rooted directed tree data structure that can be constructed in a linear time and it is used to store all suffixes of a given string. The proposed work builds a suffix tree of an e-mail address and uses matching algorithm that checks if any of the surnames present in the dictionary is a substring or not. The matching algorithm has a computational complexity that is linear with the length of the surnames used to match against an e-mail address.

The rest of this section is organized as follows. Subsection 3.1 describes the suffix tree construction algorithm, followed by Subsection 3.2 that present the substring matching algorithm. Finally, Subsection 3.3 describes the implementation of the E-mail Address profiler tool.

3.1 Suffix Tree Construction Method

A suffix tree is a versatile data structure that stores all suffixes of a given string that can be constructed in a linear time [15]. It has been used in many applications [16], [17].

Let z be a string of length n over a finite alphabet Λ. Let \$ be a symbol from Λ and it is not present in z then an enhanced string can be represented as z\$ to make sure that every suffix is unique. A suffix tree which is constructed for a string z of length n has exactly $n+1$ leaves. Each internal node, other than root, has at least two children. Each edge is labeled with non-empty substring of $z\$$ and no two edges of a node can have edge -labels start with the same character. Let $z[i..n-1]$ be i th nonempty suffix of string $z\$$, then for any leaf i of the suffix tree, the concatenation of edge-labels on the path from root to leaf i is the i th suffix of string is $z[i..n-1]$.

The suffix tree of the enhanced string is represented as $\Gamma(z)$. Each node represents , which denotes a substring w that is the concatenation of edge-labels on path from root to the corresponding node. Each edge in suffix tree $\Gamma(z)$ is a substring of z\$. Let represents the subtree rooted at node. The root of suffix tree is denoted as ($\Gamma(z)$). A suffix link is an auxiliary unlabeled edge between two nodes , such that → where is a character. Suffix links are used significantly to speed up the insertion of each new suffix. Each suffix shares the prefix of previous suffix and suffix links are useful to jump quickly to another node in the suffix tree and hence suffix tree construction algorithm is linear. Each non-leaf node of a suffix tree $\Gamma(z)$ has a suffix link [15] and the suffix link for a root is root itself. If the set of all non-empty strings u such that uv belongs to nodes in the suffix tree for some string v (possibly empty) then the set contains all possible substrings of $z\$$. The suffix tree data structure is useful for computations on substrings of a string. Each leaf node represents a suffix of the given string and the dotted lines represent the suffix links. Figure 2, represents suffix tree data structure for the enhanced string *aamalam*\$.

3.2 Surname Matching Method in an E-mail Address

This method returns null if it finds none of the surnames as substring in an e-mail address. However, if the algorithm finds any surname as a substring in the e-mail address then it returns the matched surname.

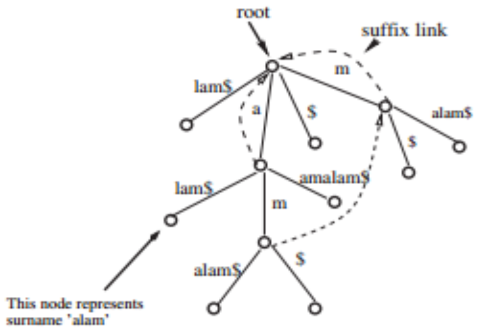


Figure 3, Suffix tree for an e-mail address aamalam@dom.com. The surname is *alam*\$, it is a substring in the e-mail address and it has been shown at a leaf node.

If it finds more than one surname as substring then it returns the longest surname, which is substring of the e-mail address. In general, it is unusual, since our assumption is that one identify is embedded in each e-mail address.

The algorithm *SurnameMatching* takes surname , suffix tree of an e-mail address $\Gamma(z)$, and empty string *match* which represents the matching part of surname in the e-mail address. The *SurnameMatching* algorithm compares the surname with the string associated to the edge of each child of the root node. If the surname matches the prefix of the edge then it returns surname, which is identified in the e-mail address. If there is no edge that matches with the prefix of then it returns a *null* (It says there is no substring present). Otherwise, if a prefix of surname is matched then the prefix is copied into the *match* string, eliminates the prefix from surname, and calls *SurnameMatching* algorithm at child node to check whether or not the remaining surname as substring in the e-mail address recursively. The detailed algorithm is given in Algorithm 2.

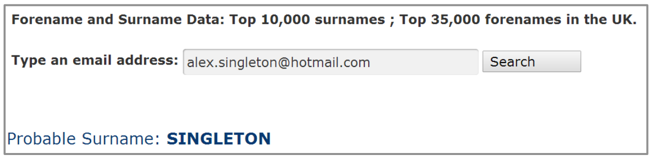
Algorithm 2 SurnameMatching(*s, Γ(z), match*)

```
{$s \in S$ is a surname in a set of surnames S. Let $|s|$ be number of characters in surname $s$. Let $\Gamma(z)$ be suffix tree of an e-mail address. Let $match$ be the string matched with the surname in the e-mail address and it is empty initially.}
Let string $temp = \varphi$;
{let $T$ be next child of $root(\Gamma(z))$ and $T.edge$ be its edge. Let $s$ be a character at position $j$ of string $s$}
while $root(\Gamma(z))$ has next child do
    $k = 0$;
    while $k < |T.edge|$ & $k < |s|$ & $T.edge[k] = s[k]$ do
        $k$++;
    end while
    if $k = 0$ & $k = |s|$ then
        $match = match + |s|$;
        return match;
    else
        if $k = 0$ & $k = |T.edge|$ then
            {let $s[l,m]$ be a substring between position $l$ to $m$ of $|s|$}
            $match = match + s[0,k]$
            $s = s[k+1, length(s)]$
            return SurnameMatching($|s|$, $T$, $match$);
        else
            return temp;
        endif
    endif
endwhile
return temp;
```

Figure 3 shows the suffix tree for an e-mail address *aamalam*$ and *alam*$ is the surname. Given a surname and a suffix tree Γ(*z*) where *z* is an e-mail address, the proposed method finds weather is a substring or not in *O*(|s|) time. In the example, the edge *'a'* is matched with the prefix of surname *alam*$ and the algorithm finds a child node attached to *'a'* and traverses that child node. It finds the edge *lam*$ that matches with the remaining characters of the surname (i.e., *lam*$) and hence returns the surname.

3.3 System Implementation

The tool was implemented as a web application. Given an e-mail address, the tool identifies a surname contained in the e-mail address by using the suffix tree method. Figure 4 shows a screenshot of the tool, where a search was conducted by using an e-mail address (alex.singleton@hotmail.com) and the system was able to identify the identity (i.e. surname: Singleton) from the e-mail address.



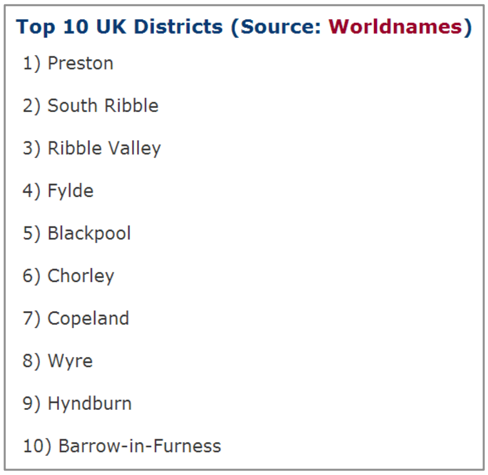
Forename and Surname Data: Top 10,000 surnames ; Top 35,000 forenames in the UK.

Type an email address: alex.singleton@hotmail.com

Probable Surname: **SINGLETON**

Figure 4: Identification of a surname from an e-mail address.

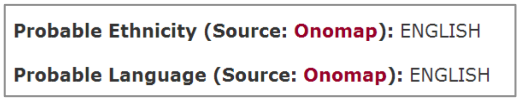
Then, the tool extracts the ethno-linguistic characteristic and maps the geographical distribution of the surname. The tool does that by using data from Onomap, Worldnames, and 2007 Electoral Register. Figures 5-7 show the results of the search. The tool returns the top 10 UK Districts of the identified surname, probable ethnicity and language of the surname, and the geographical distribution of the surname in the UK. Worldnames and Onomap were used to identify the top 10 UK districts and probable ethnicity and language of the surnames respectively. 2007 Electoral Register was used to map the geographical distribution of the surname.



Top 10 UK Districts (Source: Worldnames)

1) Preston
2) South Ribble
3) Ribble Valley
4) Fylde
5) Blackpool
6) Chorley
7) Copeland
8) Wyre
9) Hyndburn
10) Barrow-in-Furness

Figure 5: Top 10 UK Districts of the surname 'Singleton'.



Probable Ethnicity (Source: Onomap): ENGLISH

Probable Language (Source: Onomap): ENGLISH

Figure 6: Probable ethnicity and language of the surname 'Singleton'.

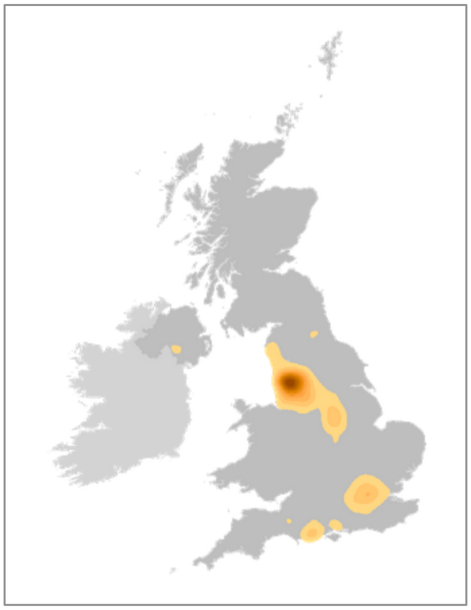


Figure 7: Geographical distribution of the surname 'Singleton' in the UK.

Figure 8, below, shows the geographical distribution of another surname 'Keay' identified from the e-mail address (james.keay@gmail.com).

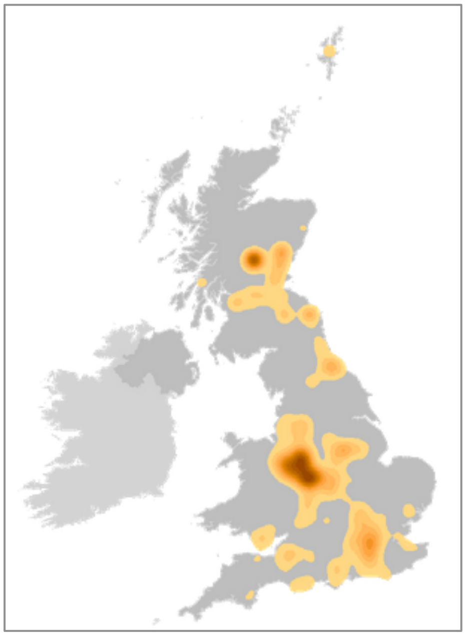


Figure 8: Geographical distribution of the surname 'Keay' in the UK.

This tool could be useful in a number of different applications. At the moment, the tool processes a single e-mail address at a time. However, the suffix tree method can be used to identify individuals by means of the analysis of their e-mail addresses on a large scale. The surname geographical distribution functionality can also be extended to multiple surnames. From a cyber-crime prevention point of view, the tool can be extended to analyse e-mail address book of individuals. In turn, the tool could be enhanced to process a large corpus of e-mail addresses to analyse their probable identities.

4. CONCLUSION AND FUTURE WORK

In this paper we have presented two tools of the Uncertainty of Identity framework. These tools identify and profile the ethno-cultural characteristics of individuals from their social media accounts and e-mail addresses. We used Twitter as a case study for the Twitter Geographic Profiler tool which builds a map of the ethno-cultural communities of a person's friends on Twitter. The E-mail Address Profiler tool identifies the probable surnames of individuals from their e-mail addresses, extracts the ethno-cultural characteristics, and maps the geographical distribution of the surname across the UK.

As part of the E-mail Profiler Tool, this paper has proposed an efficient technique to extracts identities (surnames or forenames) from their e-mail addresses. The proposed technique constructs suffix tree of an e-mail address in linear time and matches against each identity efficiently since all suffixes of an e-mail address are represented in a compact data structure. It improves the computation burden caused due to large volumes of identities to verify against each e-mail address such that the ethnic, geographic, and cultural behaviours of the individuals can be established from their email addresses.

This is a work in progress and the final output will be an Uncertainty of Identity framework for identity management and profiling. There are a number of avenues to improve both tools in the future. The Twitter Geographic Profiler can be improved by implementing it as a Facebook application to gather less noisy profile information. We plan to study the data gathered by the tool in order to investigate the privacy implications of the proposed ethno-cultural profiler. The E-mail Address Profiler can be improved by implementing the functionality to process multiple e-mail addresses at a time. This will enable the tool to analyse a large corpus of e-mail addresses to analyse their probable identities. The tool can also be improved by extending it to use the surname geographical distribution data of other countries in addition to the UK. Worldnames contains the surname distribution data of 26 countries around the world and will be used to extend the E-mail Profiler tool in the future.

5. ACKNOWLEDGEMENTS

This work was completed as part of the EPSRC research Grant "The Uncertainty of Identity: Linking Spatiotemporal Information in the Real and Virtual Worlds" (EP/J005266/1). We also like to thank Jens Kendt from UCL Department of Geography for helping us in the production of maps for the E-mail Address Profiler tool.

6. REFERENCES

[1] S. Black, S. Creese, R. Guest, B. Pike, S. Saxby, D. S. Fraser, S. V. Stevenage, M. T. Whitty. SuperIdentity: Fusion of Identity across Real and Cyber Domains. *In ID360 – The Global Forum on Identity*, Austin, US, 23-24 April, 2012.

[2] SuperIdentity. http://www.southampton.ac.uk/superidentity/. Retrieved 2nd July, 2014.

[3] The Uncertainty of Identity Project. http://www.uncertaintyofidentity.com/. Retrieved 2nd July, 2014.

[4] P. Mateos, P. A. Longley, and D. O'Sullivan. Ethnicity and population structure in personal naming networks. *PloS one*, 6(9):e22943, 2011.

[5] CACI. http://www.caci.co.uk/. Retrieved 2nd July, 2014.

[6] Worldnames. http://worldnames.publicprofiler.org/. Retrieved 2nd July, 2014.

[7] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? *In Proceedings of the 19th International Conference on World Wide Web*, pages 591-600. ACM, 2010.

[8] J. Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145-151, 1991.

[9] J. B. Kruskal and M. Wish. Multidimensional Scaling. Volume 11, Sage, 1978.

[10] A. Lima and M. Musolesi. Spatial dissemination metrics for location-based social networks. *In Proceedings of the 4th International Workshop on Location-Based Social Networks. Colocated with UbiComp'12,* pages 972-979, 2012.

[11] V. J. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85-126, 2004.

[12] D. Gusfield. Algorithms on Strings, Trees and Sequences. Cambridge University Press, 1997.

[13] S. Veluru, Y. Rahulamathavan, M. Rajarajan. Surname identification and correction in a corpus of forename surname data set. *In Proceedings of the UK Workshop on Computational Intelligence*. September 2012.

[14] S. Veluru, Y. Rahulamathavan, P. Viswanath, P. Longley, M. Rajarajan. E-mail Address Categorization based on Semantics of Surnames. In *Proceedings of the IEEE Symposium on Computation Intelligence and Data Mining (CIDM)*, 222-229, April 2013.

[15] R. Giegerich and S. Kurtz. From Ukkonen to McCreight and Weiner: A unified view of linear-time suffix tree construction. *Algorithmica*, 19:331-353,1997.

[16] C. Hui and S. V. N. Viswanathan. Fast and space efficient string kernels using suffix arrays. In *Proceedings of the 23rd International Conference on Machine Learning*, 929-936, 2006.

[17] F. Raheed, M. Alshalalfa, R. Alhajj. Efficient periodically mining time series databases using suffix trees. *IEEE Transactions on Knowledge and Data Engineering*, 23: 79-94, 2011.