E-mail Address Categorization based on Semantics of Surnames

Suresh Veluru*, Yogachandran Rahulamathavan*, P. Viswanath†, Paul Longley‡, and Muttukrishnan Rajarajan*
*Information Security Group, School of Engineering and Mathematical Sciences,
City University London, London, EC1V 0HB, United Kingdom.
E-mail:{Suresh.Veluru.1, Yogachandran.Rahulamathavan.1, R. Muttukrishnan}@city.ac.uk
†Department of CSE, R.G.M.C.E.T, Nandyal, Andhra Pradesh, India
E-mail: viswanath.pulabaigari@gmail.com
‡Department of Geography, University College London, London, WC1E 6BT, United Kingdom.
E-mail: p.longley@ucl.ac.uk

***Abstract*—Surname (family name) analysis is used in geography to understand population origins, migration, identity, social norms and cultural customs. Some of these are supposedly evolved over generations. Surnames exhibit good statistical properties that can be used to extract information in names data set such as automatic detection of ethnic or community groups in names. An e-mail address, often contains surname as a substring. This containment may be full or partial. An e-mail address categorization based on semantics of surnames is the objective of this paper. This is achieved in two phases. First phase deals with surname representation and clustering. Here, a vector space model is proposed where latent semantic analysis is performed. Clustering is done using the method called average-linkage method. In the second phase, an email is categorized as belonging to one of the categories (discovered in first phase). For this, substring matching is required, which is done in an efficient way by using suffix tree data structure. We perform experimental evaluation for the 500 most frequently occurring surnames in *India* and *United Kingdom*. Also, we categorize the e-mail addresses that have these surnames as substrings.**

***Index Terms*—Vector space model; latent semantic analysis; surnames; average link clustering method; suffix tree;**

I. Introduction

Due to rapid growth of digital data, knowledge discovery and data mining have great potential which would turn data into useful information and knowledge. Text mining (sometimes called 'mining from text documents') is to extract knowledge from a set of text documents [6]. One such knowledge is to discover the clustering structure present in the data, *i.e.,* to find groups of documents [11]. This can be later used to categorize a new document in to one of these pre-discovered classes (clusters) or as an outlier (saying 'does not belong to any of these groups'). Similar to this, names, like first names, family names of individuals can be clustered to find inherent structure present in them which later can be used to classify a new name. This knowelge is shown to have importance in geography [5].

Broadly speaking, family names (surnames) represent ethnic, geographic, cultural and genetic structures that have been developed in human populations. It is a well known fact that people migrate from one location to other due to job prospects, economic prosperity, political unrest, etc. However, the surnames of migrants retain semantic similarity to surnames of the people at their original locations.

In future, an e-mail address can be used as a form of digital identify of an individual that often holds surname as a substring. Thus, a methodology is important to categorize an e-mail address based on the knowledge extracted from names data set. Hence, association among people can be predicted from the e-mail addresses. The objective of this paper is to extract information in names data set which can be used in classifying an e-mail address. Knowledge discovery in names data set involves identifying relationship or association among groups of people (surnames).

In text mining, latent semantic analysis (LSA) is used in finding semantic similarity between terms (words) across documents [15]. Here, a document is seen as a bag of words where the lower level structure (like phrases, sentences which shows a definite relationship between words) present in the document is neglected. It is shown that the phrase based approaches does not perform well since phrases do not repeat as the terms repeat in a set of documents. Hence, phrase based approaches do not capture good statistical information [14]. Vector space model is used popularly in text mining to represent documents. Similarly, surnames provide good statistical information at several location to extract knowledge from names data set using vector space model. Several surname analysis techniques have been developed in [3], [10], and [9], but do not explicitly use the vector space model to extract knowledge.

Our contribution: To the best of our knowledge, this is the first paper that represents surnames at different locations in a vector space model and applies classical text mining techniques such as latent semantic analysis (LSA), average-link clustering method, and suffix tree data structure appropriately to perform categorization of e-mail addresses based on semantics of surnames.

The proposed method in this paper has two phases. In the first phase, it represents surnames in a vector space model and applies LSA and an average link clustering method in order to cluster surnames which co-occur together in several locations. In the second phase, it constructs the suffix tree of an e-mail address which compactly represents all of the suffixes of the e-mail address. Further, it performs a substring matching technique such that if any surname is present as a substring in the e-mail address then the e-mail address is assigned into the cluster to which the surname belongs. This means that if two surnames that are in the same cluster are substrings of two different e-mail addresses then these two e-mail addresses will also be assigned into the same cluster.

This paper is organized as follows. Section II sets out the background and literature review for the proposed work. Section III describes proposed method for e-mail address categorization. Section IV presents the experimental results and finally Section V presents conclusion and future work of the paper.

II. Background and literature review

This section briefly explains some of the background techniques and literature review that are used in this paper to develop the proposed method.

A. Background

Vector space model is popularly used to extract information in text documents. Consider if a document contains a bag of words then each document could be represented with a d-dimensional vector where d- represents d most frequent terms (or words) in a set of documents. Each element of the vector represents either term frequency multiplied by the inverse document frequency (TF-IDF[1]) if the term is present in the document or 0 if the term is not present in the document. A set of documents are represented using a set of vectors in the form of a term-document matrix and techniques such as latent semantic analysis (LSA), clustering and classification of documents can be performed on the term-document matrix [2].

The LSA method computes the semantic similarity among words in the term-document matrix. It performs corpus based statistical analysis that finds words which co-occur together in several documents [15]. LSA represents a vector for each word and hence the cosine similarity between two vectors can be used to measure semantic similarity between corresponding words. Popular clustering methods can be applied to group words that are semantically similar. Clustering methods can be divided into two types, *i.e., hierarchical* and *partitional* clustering methods. Hierarchical methods represent clusters and subclusters in a hierarchy. If π_i and π_{i+1} are two successive levels, then normally, either π_i is a refinement of π_{i+1} or π_{i+1} is a refinement of π_i . Single-link, complete-link and average-link clustering methods are the most widely used methods of this category which produce arbitrarily shaped clusters when compared with partitional clustering. The single-link clustering method is sensitive to noisy patterns and may merge two clusters if they are connected by a chain of noisy patterns [1]. In this sense, average link clustering method can be used to find good clusters.

B. Literature review

Surname analysis have been developed in geography such as identifying spatial concentration of surnames [3], migrant surname analysis [8], uncertainty in the analysis of ethnicity classification [10], and ethnicity and population structure analysis [9]. However, the degree of similarity between surname mixes has been developed by comparing relative frequencies of surnames at different locations such as *isonymy* [7] and *Lasker distance* [13]. These measures are complementary measures such that the inverse natural logarithm of the *isonymy* creates a more intuitive measure called *Lasker distance*. These are applicable to study inbreeding between marital partners or social groups, but do not explicitly address the semantic similarity between surnames. E-mail address categorization based on semantics of surnames is proposed in the following section.

III. E-mail address categorization

This section describes the proposed e-mail address categorization method. Figure 1 presents a block diagram for an e-mail address categorization technique which has two phases represented using dotted lines. Figure 1 also documents each phase as follows. In the first phase, the semantics of surnames are identified by representing a set of names at each location using a vector space model followed by latent semantic analysis as shown in the three blocks and as explained in Subsection A. Further, clustering of surnames is shown as an average-link clustering method and is explained in Subsection B. In the second phase, suffix tree construction of an e-mail address is shown in two blocks and is explained in Subsection C. Surname identification in an e-mail address is shown in matching algorithm and is explained in Subsection D.

A. Semantics of surnames

We adapt methods used in information retrieval in order to represent each location which contains a bag of surnames as a vector, and this is used to identify the semantics of surnames.

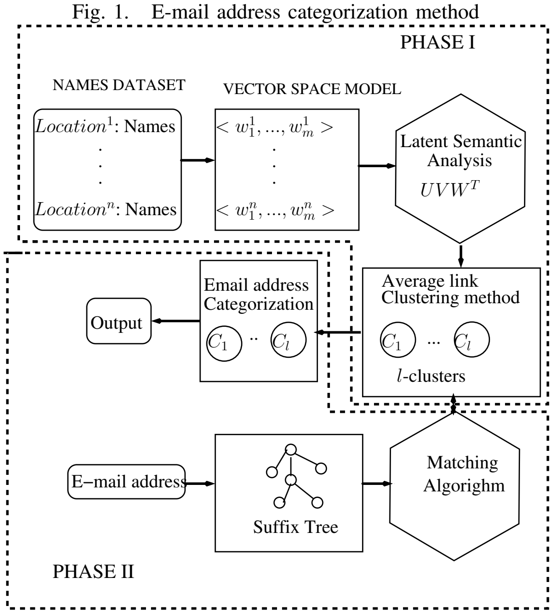
Consider the location space of a region or a country consisting of a set of locations where each location has a bag of surnames. Let there be n locations that are represented as $\mathcal{L}^1, \ldots, \mathcal{L}^n$. A typical vector space model represents each location with a vector consisting of m entries where m represents the top m frequently occurring surnames in a region or a country. Let these top m frequently occurring surnames be $\mathcal{S} = s_1, \ldots, s_m$. The vector space model for each location $\mathcal{L}^i$ is represented with a m-dimensional vector, for $i = 1$ to n is given in (1) where $w^i_{s_j}$ represents the weight of the surname s_j in location $\mathcal{L}^i$.

$$\mathcal{L}^i =< w^i_{s_1}, w^i_{s_2}, \ldots, w^i_{s_m} > \qquad (1)$$

We assign weight $w^i_{s_j}$ that represent the weight of surname s_j in location $\mathcal{L}^i$. The weight depends upon the number of occurrences of surname s_j in location $\mathcal{L}^i$ called *surname frequency* and a global weight for each surname s_j called *inverse location frequency* (ILF). The weight $w^i_{s_j}$ is given

[1] A popular representation scheme in information retrieval weights each term using a global weight IDF which is inversely proportionate to the number of documents that contains the term

Fig. 1. E-mail address categorization method



in (2). Here $f^i_{s_j}$ is the *frequency of surname* s_j at location $\mathcal{L}^i$ and $ILF(s_j)$ is the *inverse location frequency*.

$$w^i_{s_j} = \begin{cases} f^i_{s_j} * ILF(s_j) & \text{if surname } s_j \text{ in location } \mathcal{L}^i \\ 0 & \text{if no surname } s_j \text{ in location } \mathcal{L}^i \end{cases} \tag{2}$$

$ILF(s_j)$ provides the importance of surname s_j that retrieve locations using surname s_j. If surname s_j appears only in a particular location then it is easy to retrieve that location given the surname s_j. If a surname appears in one location then it is of greater importance than a surname that appears in several locations. If n_j is the number of locations in which the surname s_j appears and n is the total number of locations then $ILF(s_j)$ is given in (3).

$$ILF(s_j) = log_2(\frac{n}{n_j}) \tag{3}$$

Let the location space which contains a set of locations be $\mathcal{L}$, represented by a matrix consisting of location-surnames. For our convenience, let $\mathcal{L}^T$ be a transpose matrix of $\mathcal{L}$ having m rows and n columns given below.

$$\mathcal{L}^T = \begin{array}{c} \\ s_1 \\ . \\ . \\ s_m \end{array} \begin{array}{c} \begin{array}{ccccc} \mathcal{L}^1 & . & . & \mathcal{L}^n \end{array} \\ \begin{pmatrix} w^1_{s_1} & . & . & w^n_{s_1} \\ . & . & . & . \\ . & . & . & . \\ w^1_{s_m} & . & . & w^n_{s_m} \end{pmatrix} \end{array}$$

We apply LSA to $\mathcal{L}^T$ which in-turn applies a SVD technique that decomposes $\mathcal{L}^T$ into three matrices U, V and W^T such that $\mathcal{L}^T_{m\times n} = U_{m\times k}V_{k\times k}(W_{n\times k})^T$. The matrices $U_{m\times k}$ and $(W_{n\times k})^T$ correspond to surnames and locations respectively which consist of *orthonormal* columns. The matrix $V_{k\times k}$ is a diagonal matrix that containing the singular values in descending order where the i^{th} singular value indicates the amount of variation along the i^{th} axis. We focus on matrix $U_{m\times k}$ which corresponds to surnames that have k columns represented with $dim_1, \ldots, dim_k$. This means that each surname s_i is a vector of k dimensions such that $s_i =< u_{i1}, \ldots, u_{ik} >$, for $i = 1$ to m which is given as below.

$$U_{m\times k} = \begin{array}{c} \\ s_1 \\ . \\ . \\ s_m \end{array} \begin{array}{c} \begin{array}{cccc} dim_1 & . & . & dim_k \end{array} \\ \begin{pmatrix} u_{11} & . & . & u_{1k} \\ . & . & . & . \\ . & . & . & . \\ u_{m1} & . & . & u_{mk} \end{pmatrix} \end{array}$$

The semantic similarity between two surnames s_i and s_j is a cosine similarity between two vectors s_i and s_j which is given by (4). Further, the clustering of surnames can be performed using the semantic similarity given by (4) to identify semantic clusters of surnames which is explained in the following subsection.

$$COS(s_i, s_j) = \frac{\sum_{t=1}^{k} u_{it} \times u_{jt}}{\sqrt{(\sum_{t=1}^{k} u_{it} \times u_{it})}.\sqrt{(\sum_{t=1}^{k} u_{jt} \times u_{jt})}} \tag{4}$$

B. Clustering of surnames

We used average-link clustering method to develop a good semantic clusters of surnames.

Average-link clustering produces a hierarchy of clusters. Let $\mathcal{C}_i$, $\mathcal{C}_j$ be two clusters of surnames then the average-link similarity ($AvgSim$) between two clusters of surnames is defined by (5). Here $|\mathcal{C}_i|$ is number of surnames in the cluster $\mathcal{C}_i$.

$$AvgSim(\mathcal{C}_i, \mathcal{C}_j) = \frac{1}{|\mathcal{C}_i||\mathcal{C}_j|} \sum_{s_p\in\mathcal{C}_i, s_q\in\mathcal{C}_j} COS(s_p, s_q) \tag{5}$$

Initially, the average-link clustering method assumes that each surname $s_i \in \mathcal{S}$ is a separate cluster and proceeds by merging two clusters at each iteration of the clustering process. If the average-link clustering method reaches the desired number of clusters then the merging process ceases. Otherwise, at each iteration, it finds two clusters such that the average link similarity ($AvgSim$) between these two clusters is a maximum and merges them into a single cluster. The algorithm for average-link clustering method is given in Algorithm 1.

C. Suffix tree construction method

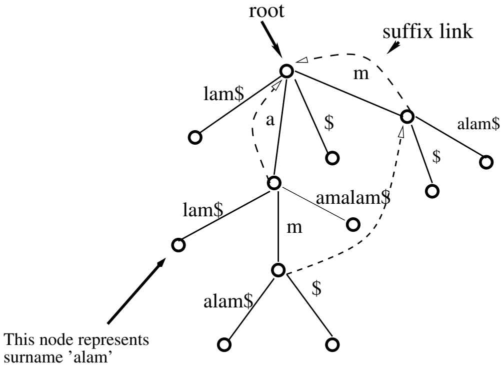
A suffix tree is a versatile data structure that stores all suffixes of a given string that can be constructed in linear time [12]. It has been used in many applications [16], [4]. Given a string z, an enhanced string is represented as $z\$$ to make sure that every suffix is unique. The suffix tree of the enhanced string is represented as $\Gamma(z)$. Each node represents $\overline{w}$ which denotes a string w that is the path from root to the corresponding node. Each edge in suffix tree $\Gamma(z)$ is a substring of $z\$$. Let $T_{\overline{w}}$ represents the subtree rooted at node $\overline{w}$. The root of suffix tree is denoted as $root(\Gamma(z))$.

A suffix link is an *auxiliary* unlabeled edge between two nodes $\overline{z_iw}$, $\overline{w}$ such that $\overline{z_iw} \rightarrow \overline{w}$ where z_i is a character.

Algorithm 1 Average-link($\mathcal{S}$, d)

```
{$\mathcal{S}$ is a set of surnames, each $s_i \in \mathcal{S}$ is a vector of $k$ dimensions, $d$ is a desired number of clusters}
Place each surname $s_i \in \mathcal{S}$ in a separate cluster. Let it be $\pi_j = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m\}$ and $j = 1$.
{ Let $|\pi_j|$ be the number of clusters at iteration $j$}
while $|\pi_j| > d$ do
  Select two closest clusters $\mathcal{C}_p, \mathcal{C}_q \in \pi_j$ such that $AvgSim(\mathcal{C}_p, \mathcal{C}_q)$ is a maximum.
  Form a new cluster $\mathcal{C} = \mathcal{C}_p \cup \mathcal{C}_q$.
  Next set of clusters is $\pi_{j+1} = \pi_j \cup \{\mathcal{C}\} \backslash \{\mathcal{C}_p, \mathcal{C}_q\}$.
  j=j+1;
end while
Output final clustering $\pi_j$
```

Fig. 2. Suffix Tree for an email address *aamalam$ (e.g aamalam@yahoo.com)*. The surname is *alam$*, it is a substring in the e-mail and it has been shown at a leaf node



Suffix links are used significantly to speedup 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 [12] and the suffix link for a root is root itself. If the set of all non-empty strings u such that $\overline{uv}$ 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.

D. Surname matching method in an e-mail address

The proposed e-mail address categorization method uses surname matching method and semantic clusters of surnames which is proposed in phase one. It constructs a suffix tree for an e-mail address. Further, it identifies any substring in the e-mail address that matches with a surname. If it finds a surname as a substring in the e-mail address then the e-mail address is assigned to the cluster to which the surname belongs. If it does not find any surname then it returns a *null*. Similarly, it checks for each surname and if any surname matches as a substring in the e-mail address then the e-mail address is assigned to the cluster to which the surname belongs. Since the e-mail address is represented in a compact trie of suffixes, the proposed method is a fast one which verifies against all surnames to identify which surname is present as substring in it. If there are two surnames present as substrings in an e-mail address then it is categorized into any one of the two surname's clusters. In general, it is unusual, however, in such cases it categorizes the e-mail address into the cluster of surname that occurs first.

Algorithm 2 SurnameMatching(s_i, $\Gamma(z)$, $match$)

```
{$s_i \in \mathcal{S}$ is a surname $i$ in a set of surnames $\mathcal{S}$. Let $|s_i|$ be number of characters in surname $s_i$. 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=$\phi$;
{let $T$ be next child of $root(\Gamma(z))$ and $T.edge$ be it's edge. let $s_i[j]$ be a character at position $j$ of string $s_i$}
while $root(\Gamma(z))$ has next child do
  k=0;
  while $k < |T.edge|$ & $k < |s_i|$ & $T.edge[k]$=$s_i[k]$ do
    k++;
  end while
  if k $\neq$ 0 & k=$|s_i|$ then
    $match = match + s_i$;
    return $match$;
  else
    if k $\neq$ 0 & i=$|T.edge|$ then
      {let $s_i[l, m]$ be a substring between position $l$ to $m$ of $s_i$}
      $match = match + s_i$[0, k];
      $s_i$=$s_i$[k+1,length($s_i$)];
      return SurnameMatching($s_i$,$T$,$match$);
    else
      return temp;
    end if
  end if
end while
return temp;
```

Surname matching method in an e-mail address identifies whether or not the surname present as substring in the e-mail address. The algorithm *SurnameMatching* takes surname s_i, 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 s_i with the string associated to the edge of each child of the root node. If the surname s_i matches with 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 s_i then it returns a *null* (It says there is no substring present). Otherwise, if a prefix of surname s_i 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 2 .

Figure 2 denotes the suffix tree for an e-mail address *aamalam$* and *alam$* is the surname. Given a surname s_i and a suffix tree $\Gamma(z)$ where z is an e-mail address, the proposed method finds weather s_i is a substring or not in $O(|s_i|)$ 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 assigns the cluster to which the surname belongs. If there are m surnames then the time complexity of the proposed method to categorize an e-mail address is $O(m \times \max\{|s_i|\}_{i=1}^{m})$.

IV. Experimental Results

This section describes experimental results. We have two countries names and e-mail addresses data sets, viz., *India* and *United Kingdom*. *India* corpus has 17.4 million names and 14.9 million e-mail addresses collected over 277 locations which covered 28 provinces and 6 union territories. *United Kingdom* corpus has 0.924 million names and 1.048 million e-mail addresses collected over 115 locations in *United Kingdom*. Location information is not used in e-mail address, perhaps, it is used to identify semantics of surnames which in-turn used to categorize e-mail addresses. In Figures 3 and 4, the horizontal axis represents surname or e-mail address domain and vertical axis represents frequency of surnames or e-mail addresses in log scale. The frequency of 40 most frequent surnames and 40 most frequent e-mail address domains for *India* and *United Kingdom* data sets are given in Figures 3 and 4 respectively.

In phase 1, we extracted the 500 most frequently occurring surnames which are represented in a vector space model for *India* and *United Kingdom* names data set. For *India* names data set, 277 vectors were generated correspond to 277 locations and for *United Kingdom* names data set, 115 vectors were generated correspond to 115 locations. After applying LSA, we chose 60 dimensions for each surname in a decomposed matrix which corresponds to surnames in order to find semantic similarity among surnames and clustered them into 30 groups.

Analysis of the spatial concentration of surnames has been developed in *Great Britain* [5] using the Location Quotient (LQ) to measure the concentration of any surname at different locations. Let $\mathcal{P}_i^j$ be the frequency of surname i in location j and let $\mathcal{Q}_i$ be the frequency of surname i in *Great Britain*. Let m be the total number of surnames then the LQ is defined by (6)

$$LQ_i^j = \frac{\frac{\mathcal{P}_i^j}{\sum_{i=1}^{m} \mathcal{P}_i^j}}{\frac{\mathcal{Q}_i}{\sum_{i=1}^{m} \mathcal{Q}_i}} \tag{6}$$

For each i, we represented surname i in the location j which has maximum LQ_i^j value in order to analyse the semantic clusters of surnames.

Semantic clusters of surnames for *India* names data set and *United Kingdom* names data set are plotted in Figure 5 and 6 respectively. We have calculated LQ values of each surname at all locations and taken the location that has the maximum LQ value [2]. For a surname, if the LQ value in a location is a maximum means the surname concentration in that location is the highest. Also, we have eliminated a few surnames that have relatively low maximum LQ value. Hence, we have plotted 123 surnames for *India* names data set and 118 surnames for *United Kingdom* names data set in which the horizontal axis represents surnames and the vertical axis represents locations where the surname concentration is a maximum. The size of the circle represents the LQ value and the number represents the semantic cluster number to each surname from 1 to 30.

From Figure 5, it is clear that clusters 3, 6, 11, and 28 contain surnames that are each heavily concentrated in a single province. It can be observed that surnames in clusters 21, 28 belong to a single community. Many of the surnames in cluster 29 are highly concentrated in *West Bengal* and many of the surnames in clusters 9 and 25 are highly concentrated in *Goa, Maharashtra, Dadra & Nagar Haweli, Daman & Diu and Andaman & Nicobar*, but are split between two clusters. Hence, it can be concluded that surnames found in cluster 9 and 25 are the result of migration between *Goa, Maharashtra, Dadra & Nagar Haweli, Daman & Diu and Andaman & Nicobar*, but, highly concentrated in *Goa*.

From Figure 6, since the LQ values are measured at each location and hence all clusters are heavily concentrated in two and more locations which are limited to a few locations for some clusters. For example, surnames in cluster 28 are heavily concentrated in *Zetland, Belfast*, and *Uxbridge*. It can be observed that surnames in clusters 18 and 10 belong to a single community which are heavily concentrated in a single location, *viz., Bradford and Uxbridge*.

TABLE I
CATEGORIZATION OF E-MAIL ADDRESSES FOR *India* DATA SET

No	E-mail address	surname	category
1	anal.chatterjee@domain1.com	chatterjee	29
2	anshukataria@domain1.com	kataria	8
3	anwesha.bakshi@domain1.com	bakshi	4
4	arnabghoshd@domain1.com	ghosh	29
5	binitmishra8@domain1.com	mishra	12
6	chawlaarvinder@domain1.com	chawla	8
7	eesatish.kumar@domain1.com	kumar	1
8	feroj_khan@domain2.com	khan	1
9	bedgautam@domain2.com	gautam	23

In the second phase, we analysed 14.9 million e-mail addresses and found that 3.7 million e-mail addresses have 500 most frequent surnames as substrings for *India* e-mail address data set. We categorized these 3.7 million e-mail addresses into 30 groups based on the clusters of surnames obtained in the phase 1 of the method. We analysed 1.048 million e-mail addresses and found 318,867 of e-mail addresses have

[2] For *India* names data set, the provinces are considered to calculate LQ value whereas for *United Kingdom* names data set, the locations themselves considered to calculate LQ values

Fig. 3. Frequency of 40 most frequent surnames and 40 most frequent e-mail address domains for *India* data set

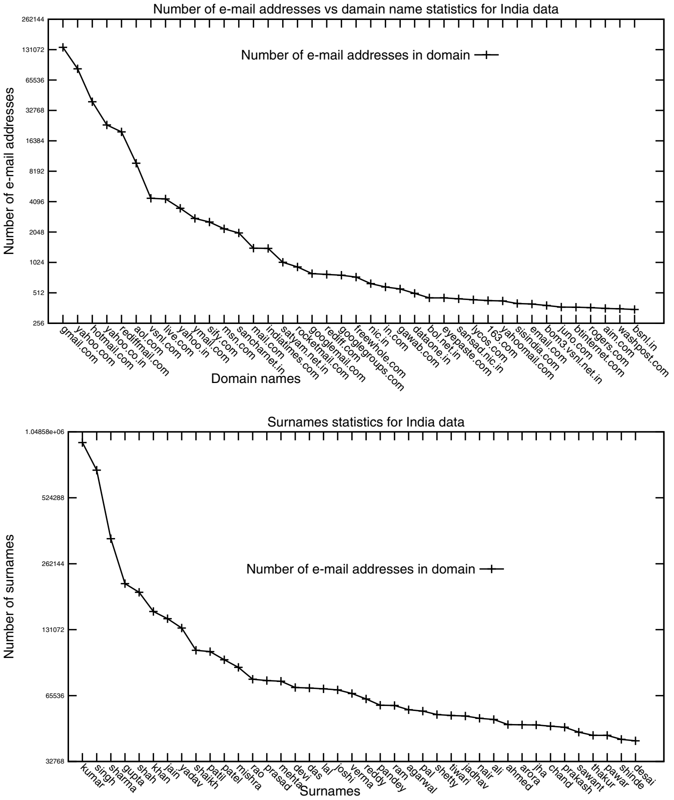


Fig. 4. Frequency of 40 most frequent surnames and 40 most frequent e-mail address domains for *United Kingdom* data set

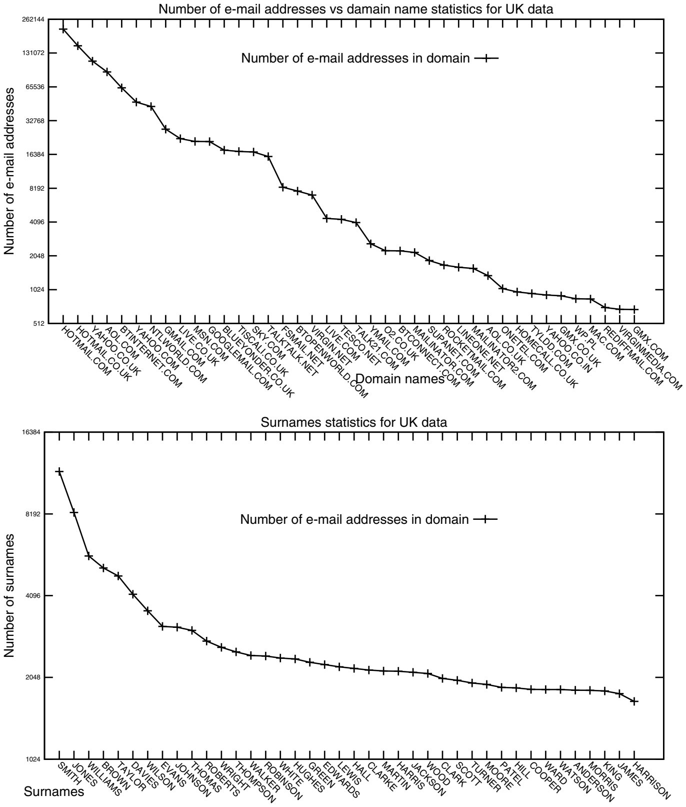


TABLE III
NUMBER OF E-MAIL ADDRESSES CATEGORIZED INTO DIFFERENT GROUPS FOR *India* AND *United Kingdom* DATA SETS

CNo	Country Name	
	India	United Kingdom
1	403485	34022
2	132674	35843
3	75559	5094
4	249669	18088
5	52414	27827
6	12742	4093
7	28689	829
8	117376	13908
9	94766	13245
10	35499	1439
11	35941	16833
12	109691	501
13	133299	3814
14	96657	14422
15	21952	227
16	407543	9003
17	103426	5152
18	82564	9655
19	19912	9827
20	602131	16347
21	99292	886
22	60759	2969
23	79369	32415
24	39220	297
25	56875	13942
26	16194	133
27	37509	8796
28	70889	13611
29	475388	4892
30	38975	757

500 most frequent surnames as substrings for *United Kingdom* e-mail address data set. Table I and II present the sample results of 9 e-mail addresses and their categories based on the semantics of their parent surnames for *India* and *United Kingdom* data sets respectively. For example, e-mail addresses 1 and 4 suggest surnames *chatterjee, ghosh* respectively which belong to the same cluster 29 and hence these two e-mail addresses are assigned to group 29. Similarly, we can see that e-mail address 7 and 8 assigned to group 1. Table III presents the categorization of 3.7 million and 318,867 of e-mail addresses for *India* and *United Kingdom* data sets respectively and the number of e-mail addresses belonging to each category is presented.

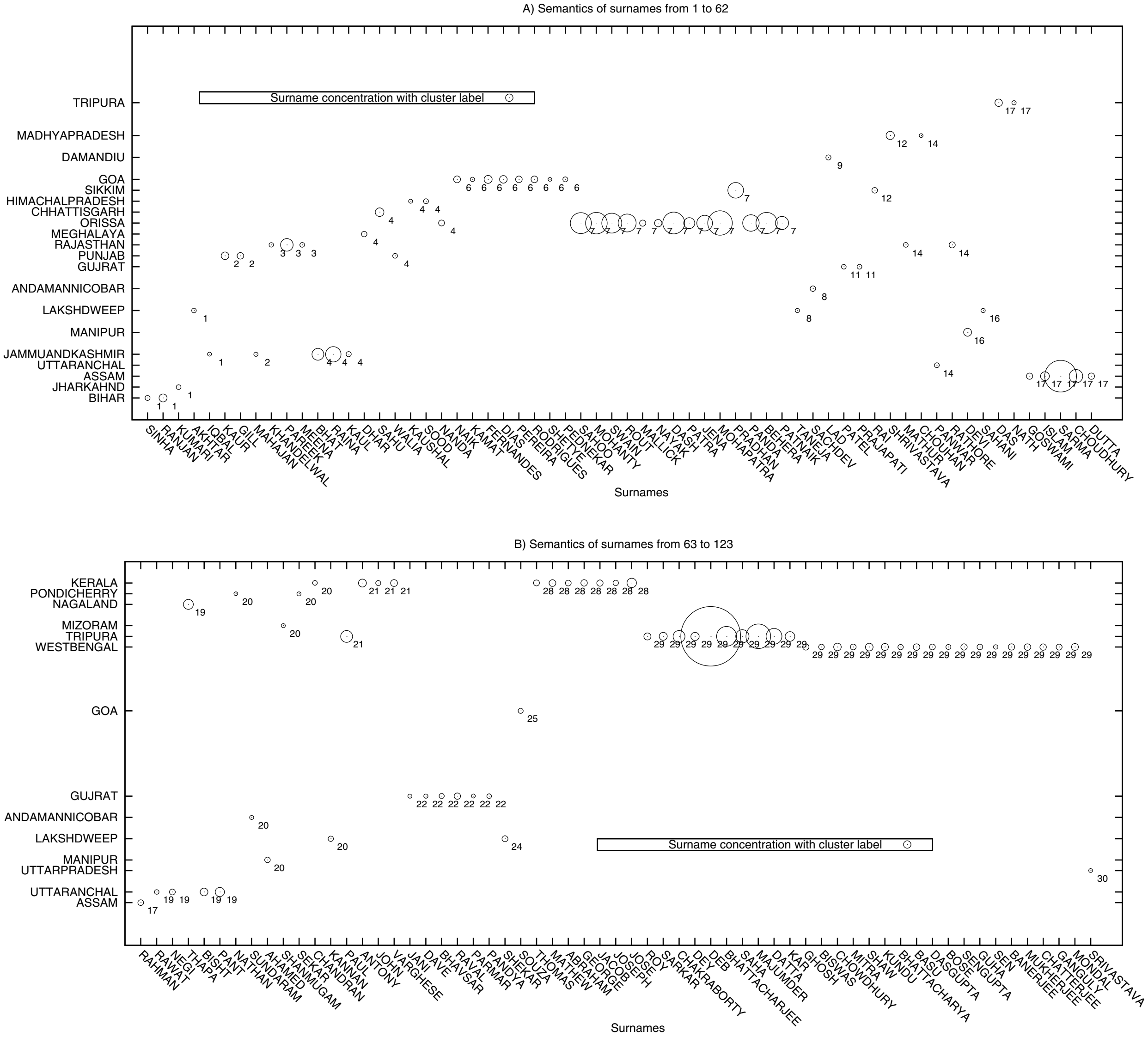
TABLE II
CATEGORIZATION OF E-MAIL ADDRESSES FOR *United Kingdom* DATA SET

No	E-mail address	surname	category
1	glennis.middleton@domin1.com	middleton	1
2	emily.curtis1@domin1.com	curtis	11
3	amanda.francis@domin1.com	francis	11
4	darrenmbates@domain2.com	bates	1
5	georgeamos44@domin3.com	george	23
6	johnnysingh1971@domain2.com	singh	30
7	michaelburton1983@domin3.com	burton	23
8	manishakaur2000@domin4.com	kaur	30
9	emmasjones2@domin2.com	jones	2

V. CONCLUSION AND FUTURE WORK

In general, an e-mail address is created to reflect the identity of an entity and it is common to see surnames as substring in

Fig. 5. Semantics of Surnames for India Data set A) Surnames from 1 to 62, B) Surnames from 63 to 123



the e-mail address of an identifiable individuals. In this paper, we have analysed statistical relationships among surnames and clustered them into several groups using a vector space model in the first phase. We used latent semantic analyses to identify semantic similarity among surnames and used the average-link clustering method to allocate surnames between 30 clusters. In the second phase, the categorization of an e-mail address has been carried out. If the e-mail address contains a surname identifiable as a substring and can thus be assigned to one of the surname clusters. This is done efficiently by using the suffix tree of an e-mail address.

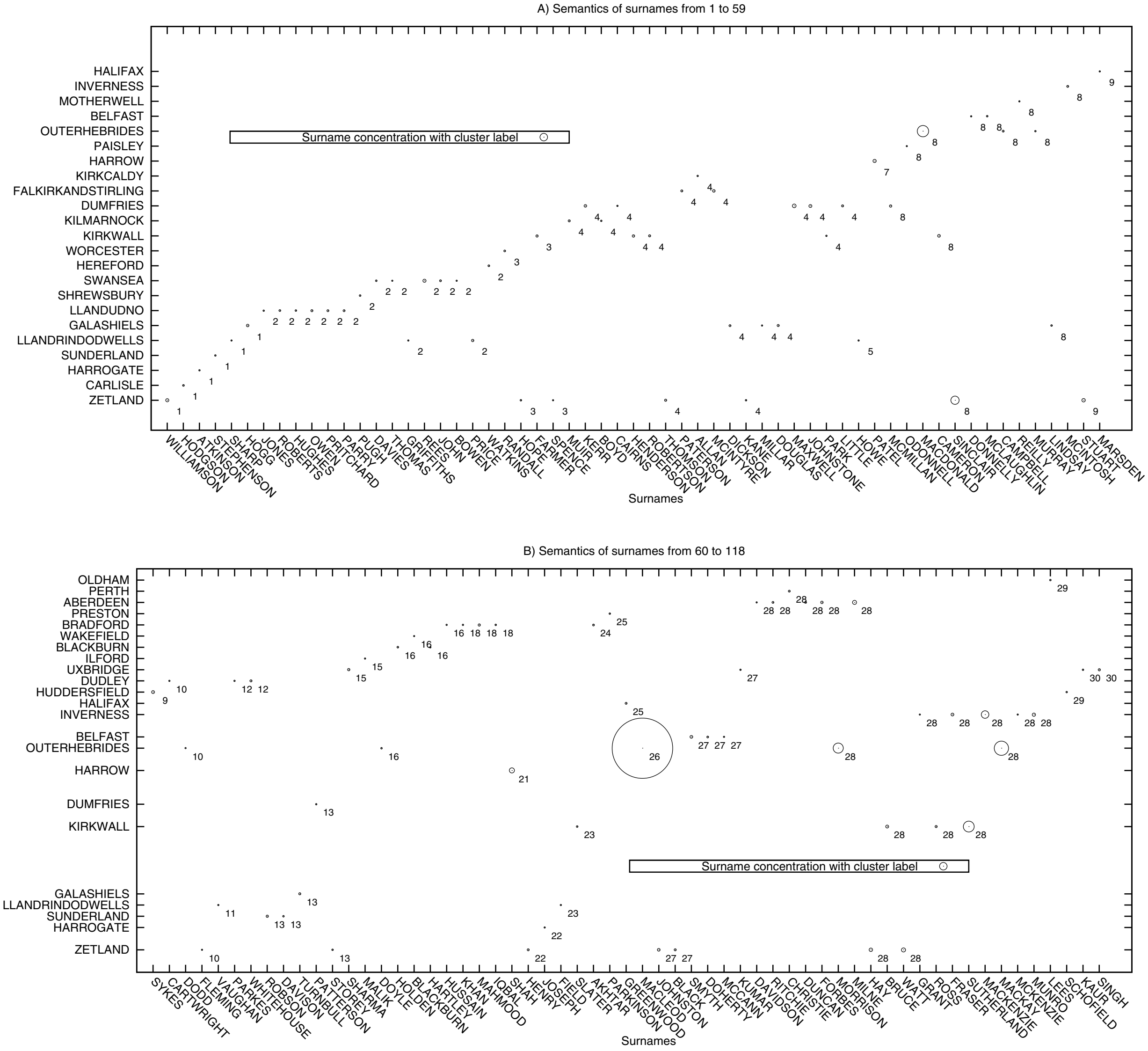
Through the experimental evaluations it is shown here that the surnames present as substring in an e-mail address can be retrieved which can be useful in the future to link individuals multiple digital identities to their physical identities. The e-mail addresses can then be assigned to locations because the geographic distributions of most surnames are far from random. From *India* and *United Kingdom* data sets, this is clearly the case from the results of our analysis of the 500 most frequently occurring surnames and the assignment of 3.7 million and 318,867 corresponding e-mail addresses into 30 groups for two data sets.

The future directions of this work will include i) finding an optimal number of clusters using the average-link clustering method; and ii) developing an efficient and fast approach for e-mail address database mining in order to find frequent sub-patterns that occur in the e-mail addresses.

Acknowledgements

This work was supported through the EPSRC Grant- “The Uncertainty of Identity: Linking Spatiotemporal Information Between Virtual and Real Worlds” (EP/J005266/1).

Fig. 6. Semantics of Surnames for UK Data set A) Surnames from 1 to 59, B) Surnames from 60 to 118



References

[1] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[2] C. D. Manning, and H. Schutze. *Foundations of Statistical Natural Language Processing*. The MIT Press, 2003.

[3] James A. Cheshire and Paul A. Longley. Identifying spatial concentrations of surnames. *International Journal of Geographical Information Science*, DOI:10.1080/13658816.2011.591291:1–17, 2011.

[4] F. Rasheed, M. Alshalalfa, and R. Alhajj. Efficient periodicity mining in time series databases using suffix trees. *IEEE Transactions on Kowledge and Data Engineering*, 23:79–94, 2011.

[5] J. Burt, G. Barder, and D. Rigby. *Elementary statistics for geographers*. Guilford Press, 2009.

[6] K. Aas and L. Eikvil. Text categorisation: A survey. In *Technical Report 941*. Norwegian Computing Center, 1999.

[7] G. Lasker. Using surnames to analyse population structure. *Naming, Society and Regional Identity*, pages 3–24, 2002.

[8] Paul A. Longley, James A. Cheshire, and P Mateos. Creating a regional geography of britain through the spatial analysis of surnames. *Geoforum*, doi:10.1016, 2011.

[9] P Mateos, Paul A. Longley, and David O'Sullivan. Ethnicity and population structure in personal naming networks. *PLoS ONE*, 6(9):1–12, 2011.

[10] P Mateos, A Singleton, and P A Longley. Uncertainty in the analysis of ethnicity classifications: Issues of extent and aggregation of ethnic groups. *Journal of Ethnic and Migration Studies*, 35(9):1437–1460, 2009.

[11] P. Viswanath, B. K. Patra, and V. Suresh Babu. Document clustering methods: Recent trends along with some efficient and fast approaches. In *Handbook of Research on Text and Web Mining Technologies*, volume 1, pages 181–188. IGI Global, 2008.

[12] R. Giegerich and S. Kurtz. From ukkonen to mccreight and weiner: A unifying view of linear-time suffix tree construction. *Algorithmica*, 19:331–353, 1997.

[13] A Rodriguez-Larralde, A. Pavesi, G. Siri, and I. Barrai. Isonamy and the genetic structure of sicily. *Journal of Biosocial Science*, 26:9–24, 1994.

[14] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1 – 47, 2002.

[15] T. K. Landauer, P. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.

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