Outlier Detection based Accurate Geocoding of Historical Addresses

Nishadi Kirielle, Peter Christen, and Thilina Ranbaduge

Research School of Computer Science, The Australian National University, Canberra, ACT 2600, Australia. nishadi.kirielle@anu.edu.au

Abstract. Research in the social sciences is increasingly based on large and complex databases, such as historical birth, marriage, death, and census records. Such databases can be analyzed individually to investigate, for example, changes in education, health, and emigration over time. Many of these historical databases contain addresses, and assigning geographical locations (latitude and longitude), the process known as *geocoding*, will provide the foundation to facilitate a wide range of studies based on spatial data analysis. Furthermore, geocoded records can be employed to enhance record linkage processes, where family trees for whole populations can be constructed. However, a challenging aspect when geocoding historical addresses is that these might have changed over time and therefore are only partially or not at all available in modern geocoding systems. In this paper, we present a novel method to geocode historical addresses where we use an online geocoding service to initially retrieve geocodes for historical addresses. For those addresses where multiple geocodes are returned, we employ outlier detection to improve the accuracy of locations assigned to addresses, while for addresses where no geocode was found, for example due to spelling variations, we employ approximate string matching to identify the most likely correct spelling along with the corresponding geocode. Experiments on two real historical data sets, one from Scotland and the other from Finland, show that our method can reduce the number of addresses with multiple geocodes by over 80% and increase the number of addresses from no to a single geocode by up to 31% compared to an online geocoding service.

Keywords: Geocode matching · String comparison · Open Street Map.

1 Introduction

The recent surge in the digitization of historical records, such as censuses, and birth, death, and marriage certificates, is enabling social and health scientists to explore human behavioral patterns across time at an unprecedented level of detail [8]. The economic, social, medical, and demographic history of people has been the interest that led to the growth of historical data analysis [7,18]. In this context, geospatial analysis plays an important role to uncover a great deal of hidden patterns in populations using geographical information such as residential addresses that are commonly available in historical databases.

Table 1: Sample historical addresses and the corresponding retrieved modern addresses using Open Street Maps along with their geocodes and address types.

Historical Address	Modern Address	Geocode [latitude, longitude]	Address Type
Kilmorie	No matching addresses	No geocode	-
Kilmore	Kilmore, Highland Scotland, IV44 8RG, UK	[57.0942387, -5.8720672]	Hamlet
Feorlig	Feorlig, Highland Scotland, IV55 8ZL, UK	[57.4020757, -6.4979426]	Hamlet
	Feorlig, A863, Feorlig Highland, IV55 8ZL, UK	[57.4052835, -6.4974833]	Post box

The process of geocoding aims to assign a geographic location (latitude and longitude) to a textual address string [2]. In order to obtain accurate geocodes for a large number of addresses, a comprehensive reference database consisting of addresses and their locations is required. Alternatively, online services, such as Google Maps or Open Street Maps (OSM) [10], some of which provide an application programming interface (API), can be employed for geocoding.

While geocoding modern addresses generally results in accurate locations being assigned to address strings [2,16], the process of geocoding historical addresses is quite challenging. This is due to address quality issues and differences between historical addresses and the addresses available in modern geocode reference databases or geocoding services. Address quality issues can occur because of spelling variations, missing values, and incomplete addresses [6], and many historical addresses do not follow modern address structures. For instance, evidence in the Digitising Scotland project [6] suggests that Nineteenth century addresses in census records mostly only provide township names [15]. This is in contrast to commonly used modern hierarchical address structures that generally consist of street numbers and names, postcodes, and town names [4].

Due to such imperfections in historical addresses, querying such address strings using modern geocoding services commonly leads to partial matches with multiple contemporary addresses regardless of the high quality, coverage, and efficiency of the used geocoding service or geocoding reference database. As a result, when querying historical addresses, existing geocoding services will return either a single, multiple, or an empty set of locations.

Table 1 shows an example of historical addresses from a real-world database we use in our experiments in Sect. 4. These addresses are extracted from Nineteenth century birth certificates from the Isle of Skye in Scotland [15]. Also shown are the results when geocoding these historical address strings using OSM, which returns no address, or a list of one or several matching contemporary addresses, their geocodes, and their corresponding address types.

Prior research in geocoding historical addresses involves establishing a separate *gazetteer* (geographical dictionary) by associating historical addresses with geocodes using existing gazetteer sources [4,13,20]. This approach, however, does not facilitate geocoding historical addresses in the absence of corresponding gazetteers with associated geocodes. To the best of our knowledge, no previous studies have investigated how to incorporate modern geocoding services such as OSM to geocode historical addresses.

Contribution We examine how to best utilize modern geocoding services to geocode historical addresses, and propose a novel geocoding method that uses Open Street Maps (OSM) [10] to geocode historical addresses. We employ two refinement phases to find a single geocode for those addresses where OSM returns either multiple or no geocodes in the first phase: We use outlier detection to find the most likely location for addresses that have multiple geocodes returned from OSM; and apply approximate string matching for address strings that did not receive any geocode to identify the most similar corresponding address string along with its geocode. We evaluate our method on two historical data sets showing how it can lead to significantly improved geocoding results compared to applying a basic online geocode service such as OSM only.

2 Related Work

We now describe research related to our work, including approaches to geocoding of historical addresses as well as the use of geocoding for record linkage.

St-Hilaire et al. [20] presented a historical address geocoding approach for Canadian census manuscripts from 1911 to 1951 by implementing a reference gazetteer which associates historical addresses with geocodes. Rather than geocoding at the level of addresses, the authors have geocoded at the level of census subdivisions (CSD), a small unit for which census returns were published, by associating each address with the corresponding CSD polygon as per the historical records. Due to variations in addresses over time, the CSD polygons are generated separately for each year by referencing and overlaying 2001 Statistics Canada digital maps onto historical maps. Logan et al. [13] have geocoded US census records from 1880 with a resolution of street-level addresses by associating street level historical addresses with contemporary TIGER (Topologically Integrated Geographic Encoding and Referencing) files which comprise geospatial information released by the US Census Bureau.

A recent approach by Lafreniere et al. [11] has implemented a framework for geocoding historical addresses using an address point locator created for each historical period by combining historical sources. All historical sources with images have been georeferenced using ArcGIS, a modern geocoding service. A similar study by Cura et al. [4] relaxes the need of complete gazetteers and instead employs geohistorical objects which contain information extracted from historical sources for the process of geocoding.

In 2015, Daras et al. [5] proposed a framework for geocoding historical addresses in the Digitising Scotland project [6]. In contrast to the previous work that used historical gazetteers, the authors employed exact and fuzzy string matching to map historical addresses to modern addresses. This framework compares historical to modern addresses and employs manual clerical review to geocode addresses that do not map to a corresponding modern address.

In the context of record linkage, several attempts have been proposed to incorporate geographical information when linking databases. Blakely et al. [1] have utilized geocodes in the blocking step when linking New Zealand census to

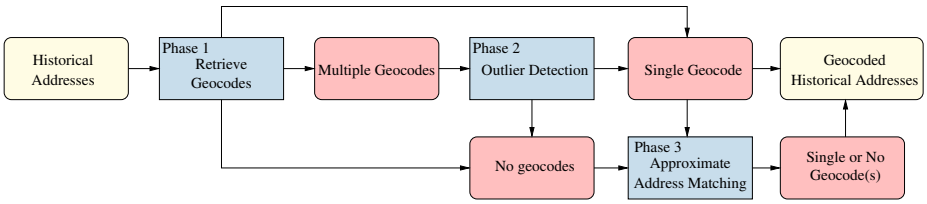

Fig. 1: Overview of geocoding historical addresses consisting of (1) geocode retrieval, (2) processing of addresses with multiple geocodes with outlier detection, and (3) approximate string matching for addresses with no geocode. The blue boxes indicate phases while the red boxes indicate intermediate results. Yellow boxes indicate input and output.

mortality data. Schraagen and Kosters [19] employed distance-based measures as consistency constraints applied on graphs for family reconstruction. More recently, in a genealogical network inferring algorithm proposed by Malmi et al. [14], the authors used a probabilistic record linkage model to construct family trees with attribute similarity features including a geographical distance.

Overall, these studies have shown that existing geocoding approaches for historical addresses are highly data dependent, where most of the research work uses existing gazetteers to conduct geocode matching. What is not yet clear is the impact of using available online geocoding services, such as OSM, for the geocoding process of historical addresses.

3 Geocoding Historical Addresses

In this section, we present our method to geocode historical addresses, as outlined in Fig. 1 and summarized next. The aim of geocoding historical addresses is to find a single and accurate geographical location for each address.

The first phase, as described in Sect. 3.1, involves retrieving geocodes from an online geocoding service. In our work, we utilize the freely accessible geocoding service OSM [10]. We denote with $\mathbf{A}$ the set of unique historical addresses for which we are interested in finding geocodes. The retrieval of geocodes from the online geocoding service can result in three subsets: (1) addresses with a single geocode, $\mathbf{A}_S$, (2) addresses with multiple geocodes, $\mathbf{A}_M$, and (3) addresses with no geocodes, $\mathbf{A}_N$, where $\mathbf{A} = \mathbf{A}_S \cup \mathbf{A}_M \cup \mathbf{A}_N$. Addresses in $\mathbf{A}_M$ and $\mathbf{A}_N$ require further processing to obtain a valid single geocode of their locations.

Over time, conventions in address structures have evolved, and as a consequence, historical addresses often do not follow the hierarchical structure of contemporary addresses [4]. Existing historical sources might also only contain incomplete or partial addresses due to choices and inefficiencies in the digitizing processes. Accordingly, for a particular historical address, multiple matching contemporary addresses (each with a different location) may be returned. In the second phase of our method, as we describe in Sect. 3.2, we process the addresses

in $\mathbf{A}_M$. We use the type of each matching contemporary address, such as *building*, *village*, *hamlet*, or *camping area*, and the geographical distances between the geocodes for a given address to remove likely irrelevant geocodes.

As a result of spelling variations and differences between historical and contemporary addresses, existing geocoding services potentially do not contain location information for all historical addresses [4]. The third phase of our method, described in Sect. 3.3, therefore focuses on identifying the most similar correct spelling variation in $\mathbf{A}_S$ for the addresses in $\mathbf{A}_N$, assuming that those addresses contain spelling variations or missing tokens from their correct version, and assigning corresponding geocodes to the addresses in $\mathbf{A}_N$.

3.1 Retrieving Geocodes

The retrieval of geocodes for a historical address from a geocoding service is straight-forward when the queried address string returns either a single or multiple geocode(s). However, due to data quality issues in historical addresses discussed above, there are instances where a historical address string cannot be matched to any existing address known to the geocoding service. For historical addresses that comprise of multiple words (tokens), in the absence of any geocode for the full address, we tokenize the address (split a string at whitespaces) and obtain geocodes for different subsets of tokens to generalize the address.

However, the hierarchical inconsistency and incompleteness of historical addresses can complicate the process of identifying hierarchical information in an address. Therefore, we iteratively remove each token (starting from the first) and query the geocoding service with the remaining set of tokens. For instance, the address 'Brae Stein Waternish' can be queried with 'Stein Waternish', 'Brae Waternish' and 'Brae Stein', and we then consider the union of geocodes retrieved from all three queries. If multiple matching contemporary addresses with multiple geocodes are returned for these queries, further processing (as described next) can help to obtain the best matching geocode.

3.2 Geocoding Historical Addresses with Multiple Geocodes

The second phase of our method, as detailed in Algo. 1, focuses on processing each address in $\mathbf{A}_M$ to obtain a single valid geocode by removing one or more invalid geocodes. We use geographical distances between geocodes, types of contemporary addresses associated with each geocode, and an outlier detection based function to filter out invalid geocodes.

Depending on the application, we can decide which type of addresses, $\mathbf{T}$, to consider when multiple modern addresses are returned for a particular historical address string. For example, we consider an order of addresses of type $\mathbf{T}$=[*village*, *hamlet*, *residential area*, *building*] for the experimental evaluation as we are interested in places where people live. In the presence of addresses of types $\mathbf{T}$, we filter the addresses with the highest priority type in lines 3 to 5.

Algorithm 1: ***Processing multiple geocodes (Phase 2)***

Input:
- $\mathbf{A}_M$: Set of historical addresses with multiple geocodes
- $\mathbf{A}_S$: Set of historical addresses with a single geocode
- $\mathbf{A}_N$: Set of historical addresses with no geocode
- $\mathbf{T}$: List of appropriate address types ordered according to their priority
- t_{min}: Threshold for minimum distance between a valid set of geocodes
- t_{max}: Threshold for maximum distance between a set of geocodes with no outliers
- f: Outlier detection function
- z: Threshold for the outlier detection function

Output:
- $\mathbf{A}_S$: Set of historical addresses with a single geocode
- $\mathbf{A}_N$: Set of historical addresses with no geocode

1: **for** $a \in \mathbf{A}_M$ **do:** // Loop over addresses
2: $\mathbf{g} = a.geocode\_set$ // Get initial geocode set for address a
3: $\mathbf{g}_T = GetPriorityGeocodes(\mathbf{g}, \mathbf{T})$ // Get set of geocodes filtered by priority types
4: **if** $\mathbf{g}_T \neq \emptyset$ **then**: // Check if prioritized geocodes are available
5: $\mathbf{g} = \mathbf{g}_T$ // Update geocode set with the prioritized geocodes if available
6: $d_{min} = GetMinimumDistance(\mathbf{g})$ // Retrieve minimum distance among geocodes
7: **if** $d_{min} > t_{min}$ **then**: // Check minimum distance
8: $\mathbf{A}_N = \mathbf{A}_N \cup \{a\}$ // All geocodes are too far apart, add address to set $\mathbf{A}_N$
9: **else:**
10: $d_{max} = GetMaximumDistance(\mathbf{g})$ // Get maximum distance among geocodes
11: **if** $(d_{max} > t_{max})$ and $(|\mathbf{g}| > 2)$ **then**: // Check possibility of outliers
12: $\mathbf{g} = GetOutlierRemovedGeocodes(\mathbf{g}, f, z)$ // Get outlier removed geocodes set
13: **if** $\mathbf{g} \neq \emptyset$ **then**: // Check if geocode set is not empty after outlier removal
14: $a.geocode = GetAverageGeocode(\mathbf{g})$ // Get average geocode
15: $\mathbf{A}_S = \mathbf{A}_S \cup \{a\}$ // Add to set of addresses with single geocode
16: **else:**
17: $\mathbf{A}_N = \mathbf{A}_N \cup \{a\}$ // Add to set of addresses with no geocode
18: return $\mathbf{A}_S, \mathbf{A}_N$

For a given address, we validate the set of geocodes by exploiting the minimum and maximum geographical distance between them. If the minimum distance, d_{min}, between any geocode pair in its set $\mathbf{g}$ is above a certain threshold t_{min}, then these geocodes are geographically too scattered. As we are employing an unsupervised process, it is not possible to decide which geocode is correct in a scattered set of geocodes. We therefore consider them as an invalid set of geocodes and add the address to $\mathbf{A}_N$ (lines 6 to 8).

In lines 9 to 17, we then aim to identify any outlying geocodes in the set of geocodes for a given address. We only employ outlier detection if the maximum distance, d_{max}, between any pair of geocodes in $\mathbf{g}$ is greater than the threshold t_{max} (lines 10 to 12). The thresholds t_{min} and t_{max} can be set by the user based on the expected circular proximity of a valid set of geocodes.

We employ outlier detection to identify any geocodes that are far away from others for a given address. Because the number of multiple geocodes for a given address is usually small, and the set of geocodes are not a set of numerical values, we use modified versions of standard statistical outlier detection functions such as the z-score normalization [9] and the robust variation of z-score normalization [17]. In z-score normalization, if a data point deviates more than z standard deviations from the mean of the data distribution, it is considered as an outlier. The values for z used vary in the range of $2 \leq z \leq 4$ [12]. The robust variation of z-score replaces the mean and standard deviation in the normalization process with the median and median absolute deviation to avoid the effect of outliers on the statistical measures [17].

To apply these statistical outlier detection functions in the context of geocodes, we use the distances between all geocodes in the set $\mathbf{g}$ for a given address, rather than the geocodes themselves. The pair-wise distances between geocodes are calculated using the great circle distance, which reflects the shortest distance between two points on the Earth measured along the surface using the Haversine equation [21]. Let us assume the radius of Earth is R and the geocodes g_1 and g_2 have longitude and latitude values as (x_1, y_1) and (x_2, y_2), respectively. Then the distance $d_{1,2}$ between these two geocodes can be calculated as:

$$h_{1,2} = \sin^2\left(\frac{x_2 - x_1}{2}\right) + \cos(x_1) \times \cos(x_2) \times \sin^2\left(\frac{y_2 - y_1}{2}\right)$$
$$d_{1,2} = R \times 2 \times \arcsin(\min(1, \sqrt{h_{1,2}})) \quad (1)$$

Now let us define the set $\mathbf{D}$ as the $n(n-1)/2$ pair-wise distances calculated between the n geocodes in $\mathbf{g}$, with $n = |\mathbf{g}|$, returned for one given address, where $n > 2$. For a given geocode $g_i \in \mathbf{g}$, we denote its set of distances to all other geocodes in $\mathbf{g}$ as $\mathbf{D}_i$. We calculate the average of a set of distances as avg(), the standard deviation as std(), the median as med(), and the median absolute deviation as mad(). The z-score, z_i, and robust z-score, rz_i, for geocode g_i are then calculated as:

$$z_i = \frac{|avg(\mathbf{D}) - avg(\mathbf{D}_i)|}{std(\mathbf{D})} \qquad rz_i = \frac{|med(\mathbf{D}) - med(\mathbf{D}_i)|}{mad(\mathbf{D})} \quad (2)$$

If the value z_i or rz_i is greater than the predefined threshold value, z, then geocode g_i is considered as an outlier.

After outliers are identified and removed (lines 11 and 12 in Algo. 1), the remaining geocodes are averaged to obtain a single geocode for the given historical address (we use the average instead of the median due to the generally very small numbers of geocodes in $\mathbf{g}$). The computational complexity of Algo. 1 is $O(|\mathbf{A}_M| \cdot g^2)$, where g is the average size of the sets of geocodes, $\mathbf{g}$.

3.3 Geocoding Historical Addresses with No Geocodes

In the third phase of our method, as outlined by Algo. 2, we employ approximate string matching to identify the most similar address string for the addresses in $\mathbf{A}_N$ where no geocode was found. To identify the most similar address string, we can either use addresses for which we have already identified a single geocode, $\mathbf{A}_S$, or alternatively use existing historical gazetteers [4,13,20].

Algo. 2 requires a string similarity function $sim()$, a similarity threshold value s_t, and two sets of addresses: those without geocodes, $\mathbf{A}_N$ (possibly due to spelling variations), and those for which a single geocode is available, $\mathbf{A}_S$. For each address $a_N \in \mathbf{A}_N$, if the highest similarity score s_{max} of a_N with any address $a_S \in \mathbf{A}_S$ is above the similarity threshold s_t (which decides if two addresses are matching or not), then we assign the geocode of the best matching address, a_S^{best}, to the non-geocoded address a_N in line 9. Otherwise, the geocode of a_N is left as unknown in $\mathbf{A}_N$ and kept for manual review. The computational complexity of Algo. 2 is $O(|\mathbf{A}_N| \cdot |\mathbf{A}_S|)$.

Algorithm 2: *Approximate Address Matching (Phase 3)*

Input:
- $\mathbf{A}_N$: Set of historical addresses with no geocodes
- $\mathbf{A}_S$: Set of historical addresses with a single geocode
- $sim()$: String similarity function
- s_t: Threshold for string similarity calculation

Output:
- $\mathbf{A}_N$: Set of historical addresses with no geocodes
- $\mathbf{A}_S$: Set of historical addresses with a single geocode

for $a_N \in \mathbf{A}_N$ **do:** // Loop over non-geocoded addresses
  $s_{max} = 0$ // Initialize maximum similarity value to 0
  for $a_S \in \mathbf{A}_S$ **do:** // Loop over geocoded addresses
    $s = sim(a_N, a_S)$ // Calculate the similarity between addresses
    if $s \geq s_{max}$ **then**: // Check if similarity is above the maximum similarity
      $s_{max} = s$ // Update the highest similarity score
      $a_S^{best} = a_S$ // Update the most similar record
  if $s_{max} \geq s_t$ **then**: // Check if the maximum similarity score is above the threshold
    $a_N.geocode = a_S^{best}.geocode$ // Assign the geocode of most similar address
    $\mathbf{A}_N = \mathbf{A}_N \setminus \{a_N\}$ // Remove the record from non-geocoded address set
    $\mathbf{A}_S = \mathbf{A}_S \cup \{a_N\}$ // Add the record to geocoded address set
return $\mathbf{A}_S$, $\mathbf{A}_N$

Many different approximate string similarity functions have been developed [3]. One commonly used such function specific for English names is Jaro-Winkler [22]. This function calculates a similarity between 0 (strings are completely different) and 1 (strings are the same) by counting the numbers of common and transposed characters. Given address strings commonly contain several tokens, we adapted this comparison function where we first sort all tokens in the addresses to be compared and then apply Jaro-Winkler on the sorted tokens. A set of pre-experiments showed good results using this approach. However, our method can use any string similarity function to match addresses. Deciding on the string similarity threshold depends on the expected similarity of matching addresses.

4 Experimental Evaluation

We evaluated our method to geocoding historical addresses on two real data sets. The Scottish (Isle of Skye) data set[1] [15] consists of 17,614 birth records from the Isle of Skye from 1861 to 1901 with 1,268 unique addresses. The Finnish data set[2] [14] contains 4,962,236 birth records from 1600 to 1917, with only 9,392 unique addresses (most of these only the name of a hamlet or village). For this data set we therefore considered the combination of village and parish names as the full address because the same village name commonly occurs across different parishes. The Finnish data set contains ground truth locations for most addresses. No ground truth is available for the Isle of Skye data set. We ran experiments for different values of the thresholds, t_{min} and t_{max}, the types of addresses, $\mathbf{T}$, and the two outlier detection functions discussed in Eqn. (2) with different threshold values, z. We implemented our method in Python 2.7, and the program is available at: `https://dmm.anu.edu.au/histrl/` to facilitate repeatability.

[1] Not publicly available, for similar data see: `https://www.scottish-places.info`
[2] Available at: `http://hiski.genealogia.fi/hiski?en`

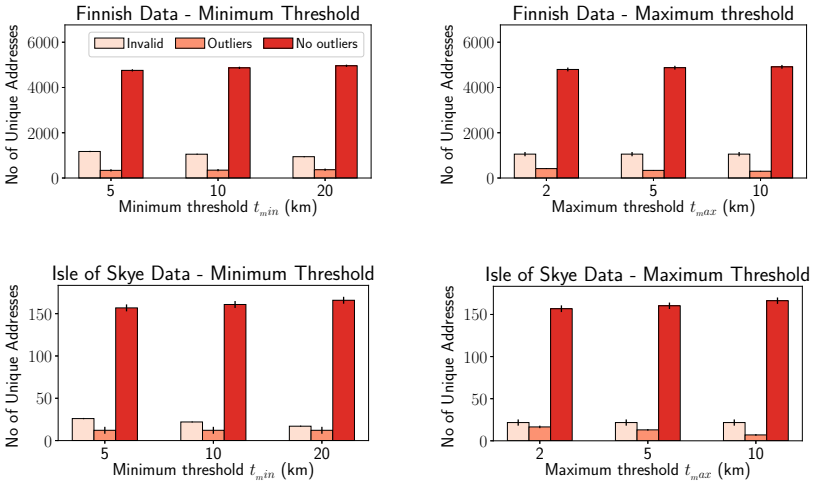

Fig. 2: Variation of multiple geocoded address categories with respect to different t_{min} and t_{max} values (as discussed in Sect. 3.2).

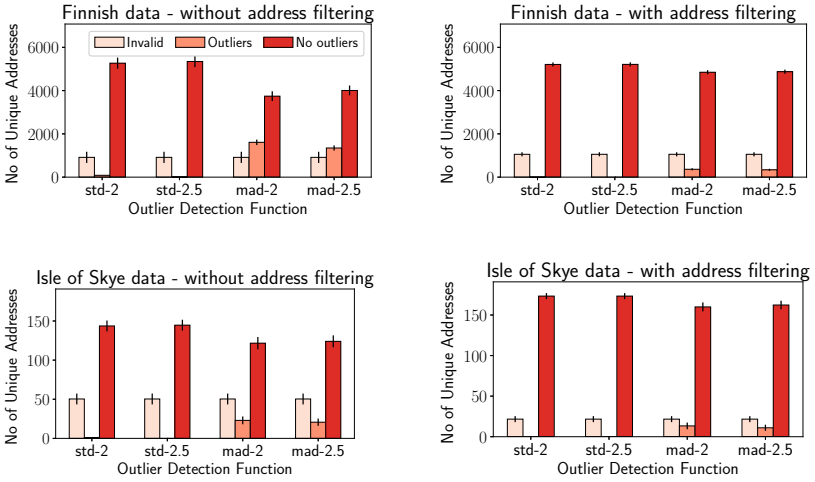

Fig. 3: Variation of multiple geocoded address categories with respect to different outlier detection functions f and thresholds z (as discussed in Sect. 3.2).

Fig. 2 shows the results of changing t_{min}, which determines if a set of geocodes is invalid or not, and t_{max}, which determines the maximum distance between geocodes in a set before outlier detection is applied. As can be seen, when t_{min} is increased, the number of addresses having invalid geocodes becomes lower ($\approx 32\%$) while the number of addresses having valid geocodes increases ($\approx 4\%$). When t_{max} is increased, the number of addresses having no outliers increases

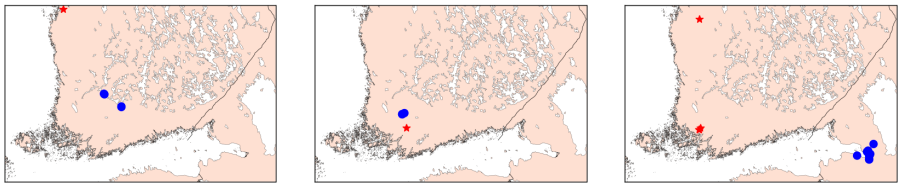

Fig. 4: Example addresses from the Finnish data set with geocode sets of 3 (left), 5 (middle), and 17 (right). Outliers are detected using the robust z-score function and shown as red stars.

Table 2: Address match percentages with different similarity threshold values s_t.

	Jaro-Winkler similarity						Sorted token Jaro-Winkler similarity					
s_t	0.7	0.75	0.8	0.85	0.9	0.95	0.7	0.75	0.8	0.85	0.9	0.95
%	99.4	94.2	74.1	49.5	29.4	11.0	99.0	93.5	73.6	49.7	29.4	10.5

slightly ($\approx 4\%$) because most received geocodes are in closer range, while the addresses with outliers are decreasing ($\approx 57\%$). Overall, however, our proposed method is robust with regard to settings of both these threshold parameters.

Fig. 3 shows results of using the two different outlier detection functions to identify outlying geocodes. The maximum number of multiple geocodes retrieved for an address was 40 for both data sets, while the average and median were 4.2 and 2, respectively. Because of these small numbers of geocodes for each address, the robust z-score function, using median, tends to perform better in identifying outliers compared to the average based z-score function. Furthermore, the figure provides strong evidence of the significance of using address type filtering. The numbers of invalid addresses and addresses with outliers are considerably higher if no address type filtering is applied compared to with address type filtering. This is because the geocodes are selected for each address for the highest priority address type while other geocodes that are unlikely to match are removed. Fig. 4 shows examples of how robust z-score correctly identifies outliers of geocode sets of different sizes.

We evaluated the effect of approximate string matching and the similarity threshold s_t, as discussed in Sect. 3.3, using the Isle of Skye data set. Table 2 shows a clear decrease in the number of matches when s_t is increased. However, a higher s_t more likely identifies the correct variation of a misspelled address due to the high similarity between the pair of address strings.

Finally, Table 3 presents the number of unique addresses of the two data sets when geocoded only with OSM, and the averages and standard deviations when different parameter settings of our geocoding method are applied. As can be seen, our method is able to find a valid single geocode for over 80% of addresses when multiple geocodes were retrieved from OSM. The approximate string matching phase is also capable of identifying correct spelling variations for misspelled addresses and identify a single valid geocode for no geocoded addresses for up to

Table 3: Summary of geocoded addresses after geocoding with OSM, after applying our geocoding algorithm and an analysis of proximity with ground truth.

	Finnish	Isle of Skye
Total number of unique address strings	9,392	1,268
Number of addresses with a single geocode from OSM	2,654	298
Number of addresses with multiple geocodes from OSM	6,268	195
Number of addresses with no geocodes from OSM	470	775
Number of addresses with multiple geocodes resulted in:		
A valid geocode with the proposed method	5,283 $\pm$207	159 $\pm$15
An invalid geocode with the proposed method	985 $\pm$ 207	36 $\pm$ 15
Number of addresses with no geocodes resulted in:		
A valid geocode with the proposed method	9 $\pm$6	331 $\pm$191
No geocode with the proposed method	1,447 $\pm$204	479 $\pm$191
Ground truth analysis:		
Number of addresses with a valid geocode within 1 km	2,284 $\pm$175	-
Number of addresses with a valid geocode within 5 km	3,226 $\pm$127	-
Number of addresses with a valid geocode within 10 km	3,682 $\pm$138	-
Number of addresses with a valid geocode within 20 km	4,065 $\pm$126	-

31% for the Isle of Skye data set. However, as the Finnish data set is normalized to contain unique addresses, in the third phase our method is unable to identify valid spelling variations for most of the Finnish addresses.

The final section of Table 3 shows the number of addresses located with a calculated geocode within $1, 5, 10$, and 20 km when compared with the ground truth location. Due to the variations in geocoding datums and the accuracy of OSM, the proximity of calculated geocodes and ground truth geocodes varies.

5 Conclusion

We have presented a novel method to geocoding historical addresses using an online geocoding service. We apply outlier detection and approximate string matching to identify accurate locations for those addresses where multiple or no geocodes were retrieved. Our evaluation on two real historical data sets showed significant improvements in geocoding historical addresses using our method compared to an online geocoding service. As future work, we plan to improve the geocode retrieval from an online geocoding service by recognizing the tokens in addresses using Hidden Markov model-based approaches [3], and we aim to compare our method with prior methods. We also aim to explore how a suitable threshold for outlier detection can be learned from the data and explore alternative outlier detection functions for geocoding of historical addresses.

Acknowledgements

This work was partially funded by the ARC under DP160101934. We like to thank Alice Reid, Ros Davies and Eilidh Garrett for their work on the Isle of Skye data set, especially Eilidh for her helpful advice on historical demography of the Isle of Skye.

References

1. Blakely, T., Woodward, A., Salmond, C.: Anonymous linkage of New Zealand mortality and census data. ANZ Journal of Public Health **24**(1), 92–95 (2000)
2. Christen, P., Churches, T., Willmore, A.: A probabilistic geocoding system based on a national address file. In: Australasian Data Mining Conference. Cairns (2004)
3. Christen, P.: Data Matching. Springer, Heidelberg (2012)
4. Cura, R., Dumenieu, B., Abadie, N., et al.: Historical collaborative geocoding. ISPRS International Journal of Geo-Information **7**(7), 262 (2018)
5. Daras, K., Feng, Z., Dibben, C.: Hag-gis: A spatial framework for geocoding historical addresses. In: GIS Research UK Conference. Leeds (2015)
6. Dibben, C., Williamson, L., Huang, Z.: Digitising Scotland (2012), `http://gtr.rcuk.ac.uk/projects?ref=ES/K00574X/2`
7. Garrett, E., Reid, A.: Introducing movers into community reconstructions: Linking civil registers of vital events to local and national census data: A Scottish experiment. In: Population Reconstruction. Springer (2015)
8. Georgala, K., van der Burgh, B., Meeng, M., Knobbe, A.: Record linkage in medieval and early modern text. In: Population Reconstruction. Springer (2015)
9. Grubbs, F.: Procedures for detecting outlying observations in samples. Technometrics **11**(1), 1–21 (1969)
10. Haklay, M., Weber, P.: OpenStreetMap: User-generated street maps. IEEE Pervasive Computing **7**(4), 12–18 (2008)
11. Lafreniere, D., Gilliland, J.: All the World's a Stage: A GIS framework for recreating personal time-space from qualitative and quantitative sources. Transactions in GIS **19**(2), 225–246 (2015)
12. Ley, C., Klein, O., et al.: Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. Journal of Experimental Social Psychology **49**(4), 764–766 (2013)
13. Logan, J., Jindrich, J., Shin, H., Zhang, W.: Mapping America in 1880: The urban transition historical GIS project. Historical Methods **44**(1), 49–60 (2011)
14. Malmi, E., Gionis, A., Solin, A.: Computationally inferred genealogical networks uncover long-term trends in assortative mating. In: World Wide Web Conference. Lyon (2018)
15. Reid, A., Davies, R., Garrett, E.: Nineteenth-century Scottish demography from linked censuses and civil registers. History and Computing **14**(1-2) (2002)
16. Roongpiboonsopit, D., Karimi, H.: Quality assessment of online street and rooftop geocoding services. Cartography and Geographic Information Science **37**(4) (2010)
17. Rousseeuw, P., Hubert, M.: Robust statistics for outlier detection. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **1**(1), 73–79 (2011)
18. Ruggles, S., Fitch, C.A., Roberts, E.: Historical census record linkage. Annual Review of Sociology **44**(1), 19–37 (2018)
19. Schraagen, M., Kosters, W.: Record linkage using graph consistency. In: Workshop on Machine Learning and Data Mining in Pattern Recognition. St. Petersburg (2014)
20. St-Hilaire, M., Moldofsky, B., Richard, L., Beaudry, M.: Geocoding and mapping historical census data: The geographical component of the Canadian Century Research Infrastructure. Historical Methods **40**(2), 76–91 (2007)
21. Van Brummelen, G.: Heavenly mathematics: The forgotten art of spherical trigonometry. Princeton University Press, Princeton (2012)
22. Winkler, W.: String comparator metrics and enhanced decision rules in the Fellegi-Sunter model of record linkage. In: Survey Research Methods ASA (1990)