

Automatic Identification of Addresses

Subjects: [Computer Science](#), [Artificial Intelligence](#) | [Geography](#), [Physical](#)

Contributor: Paula Cruz

Address matching continues to play a central role at various levels, through geocoding and data integration from different sources, with a view to promote activities such as urban planning, location-based services, and the construction of databases like those used in census operations. Closely associated to address matching is the task of address parsing or address segmentation, which consists of decomposing an address into its different components, such as a street name or a postal code. However, these tasks continue to face several challenges, such as non-standard or incomplete address records or addresses written in more complex languages.

address matching

address parsing

machine learning

deep learning

natural language processing

address geocoding

1. Introduction

An address is a reference to a unique location on Earth and is usually expressed according to a certain addressing system (a combination of components such as street names, building numbers, units, levels, unit directions, postal codes, etc.), which can be distinguished from others based on its structure as well as on the types of used components ^[1]. Due to the hierarchical nature of the fields that compose an address, the association between addresses and address fields can be formally modelled, thereby taking into account the semantic characteristics of address fields ^[2].

In general terms, address matching consists of the process of identifying pairs of records through the comparison of full addresses or address fields, with the aim of obtaining the best matching result in relation to a searched address ^[3]. Address matching is also described as the process of relating the literal description of an address to its corresponding location on a map ^[4]. In this process, known as geocoding, addresses (up to the street name or street name and door number, combined with a postal code and/or an administrative division) are matched with a reference database in order to obtain the corresponding spatial geographic coordinates ^[5]. In the absence of a unique identifier (such as the social security number, for instance), addresses can also be used as quasi-identifiers in the linking of records related to the same entity in one or more data collections ^[6]. As such, address matching main areas of application include, among others, the enrichment of data quality ^[3], named entity recognition ^[6], and location-based analyses in general ^[7], which are central to express delivery and take-out services, to disaster risk management and response, as well as in the construction of databases such as those used in census operations ^[3].

Closely associated to address matching is the task of address parsing or address segmentation, which consists of decomposing an address into its different components, such as a street name or a postal code. Basically, through parsing, it is possible to convert unstructured or semi-structured input addresses into structured ones, helping to overcome imprecise or vague addresses [5].

Regarding the matching of address fields or addresses, three types of similarities should be considered: string similarity, semantic similarity, and spatial similarity [2]. String similarity is mainly focused on finding common substrings or characters between address records or elements, whereas semantic similarity tries to capture the linguistic relations between words, such as synonyms (for instance, “street” and “road” both consist of street types but would be considered highly dissimilar based on a string similarity approach). Lastly, spatial similarity can be measured based on street numbers, when available. The combination of multiple similarities generally increases address matching accuracy, even though there is a tendency to overlook semantic characteristics and spatial proximities [2].

Traditional address matching methods are organized into two major categories: string similarity-based methods (calculation of the text-similarity between two addresses) and address element-based methods (comparison of the hierarchical results and the matching rate between each address element) [8]. However, these methods do not always manage to tackle non-standard address records, with redundant or missing address elements and few literal overlaps or written in more complex languages [9].

2. Discussion and Future Research

2.1. Detailed Literature Review

The present section aims to perform a more detailed discussion of the different address matching algorithms based on the full text of the selected articles (also summarized in [Appendix A](#)), with a view to extend the previously presented keyword-based analysis. This more detailed literature review will be organized around the three main methods which have been found to be the most relevant: string similarity-based methods, address element-based methods, and deep learning methods [9].

String similarity-based methods consist of a standard approach for address matching and generally involve the computation of a similarity metric between the addresses under comparison. Three main methods can be identified: character-based, vector-space based, and hybrid approaches [10]. Character-based methods comprehend edition operations, like sub-sequence comparisons, deletions, insertions, and substitutions. One of the best-known character-based methods is the Levenshtein edit distance metric [11], consisting of the minimum number of insertions, substitutions, or deletions which are required to convert a string into another (for instance, the edit distance between the toponyms Lisboa and Lisbonne is three, since it requires two insertions and one substitution) [10]. Another example of a character-based method is the Jaro metric [12], specifically conceived for matching short strings, like person names, with a more advanced version being latter proposed (Jaro–Winkler similarity) [13], in order to give higher scores to strings matching from the beginning up to a given prefix length.

Regarding vector-space approaches, the calculation of the cosine similarity between representations based on character n-grams (i.e., sequences of n consecutive characters) consists of a common approach, alongside the Jaccard similarity coefficient [10]. Lastly, hybrid metrics, while combining the advantages of the two previous approaches, also allow for small differences in word tokens and are more flexible in what concerns to word order and position [10]. Nevertheless, in terms of performance, there is not a best technique. The available metrics are task-dependent and, according to the study developed by Santos et al. [10], involving the comparison of thirteen different string similarity metrics, the differences in terms of performance are not significant, even when combined with supervised methods, to avoid the manual tuning of the decision threshold (one of the most important factors to obtain good results).

Address element-based methods, on their turn, rely on address parsing, a sequence tagging task which has been traditionally approached using probabilistic methods mainly based on Hidden Markov Models (HMM) and Conditional Random Fields (CRF) [3], alongside other less common approaches not always involving machine learning methods.

In what concerns the application of HMMs in the context of residential addresses, the hidden states correspond to each segment of the address and the observations consist of the tokens assigned to each word of the input address string (after the application of some cleaning procedures), which may be based on look-up tables and hard-coded rules [6]. For instance, the address “17 Epping St Smithfield New South Wales 2987”, after cleaning and tokenization, would turn into the following:

‘17’ (NU), ‘epping’ (LN), ‘street’ (WT), ‘smithfield’ (LN), ‘nsw’ (TR), ‘2987’ (PC), where ‘NU’ would stand for other numbers, ‘LN’ for locality (town, suburb) names, ‘WT’ for wayfare type (street, road, avenue, etc.), ‘TR’ for territory (state, region), and ‘PC’ for postal (zip) code [6] (p. 6). In order to determine, by statistical induction, the most likely arrangement of hypothetical “emitters” behind the observed sequence, a set of training examples is used to learn both the transition matrix and the observation matrix, through the maximum likelihood approach. Since it is computationally infeasible to evaluate the probability of every possible path (for N states and T observations, there would be N^T different paths), the Viterbi algorithm is used to find the most probable path through the model [48]. As such, the most probable sequence of states, based on previously trained transition and emission matrices, will present the highest probability of occurring, as illustrated below, in which the observation symbols are in brackets and the emission probabilities are underlined [6] (p. 7):

Start -> Wayfare Number (NU) -> Wayfare Name (LN) -> Wayfare Type (WT) -> Locality (LN) -> Territory (TR) -> Postal Code (PC) -> End

$$0.9 \times 0.9 \times 0.95 \times 0.1 \times 0.95 \times 0.92 \times 0.95 \times 0.8 \times 0.4 \times 0.94 \times 0.8 \times 0.85 \times 0.9 = 1.18 \times 10^{-2}$$

One of the main drawbacks of traditional HMMs is the fact that they do not support multiple simultaneous observations for one token. Even in more advanced versions of HMMs such as entropy Markov Models [14], in which the current state depends both on the previous state and on existing observations, there is a weakness

called the label bias problem [15]: “transitions leaving a given state to compete only against each other, rather than against all transitions in the model” [16] (p. 2). Within the present literature review, four of the considered articles propose HMM-based methods: the already mentioned one by Churches et al. [6], aiming at the preparation of name and address data for record linkage purposes, through a combined approach using lexicon-based tokenization and HMMs, with the obtained experimental results confirming it as an alternative to rule-based systems that is both feasible and cost-effective; a second paper by the same authors [17], in which a geocoding system based on HMMs and a rule-based matching engine (*Febrl*) for spatial data analysis is proposed and tested on small datasets of randomly selected addresses from different sources, with experimental results pointing to exact matches rates between 89% and 94%, depending on the source and considering the total exact matches obtained at various levels (address level, street level and locality level); the paper by X. Li et al. [18], in which an HMM-based large scale address parser is proposed, obtaining an accuracy of 95.6% (F-measure), after being tested on data from various sources with varying degrees of quality and containing billions of registers, of which 20% were synthetically rearranged in order to reproduce normal address variations; and, finally, the paper by Fu et al. [19], in which an HMM-based segmentation and recognition algorithm is proposed for the development of automatic mail-sorting systems involving handwritten Chinese characters (a problem which will be further addressed in the present literature review), with experimental results confirming its effectiveness.

Conditional Random Fields (CRFs) consist of a recent innovation in the field of text segmentation. CRFs are conditional by nature and assume no independence between output labels, illustrating real world addresses, in which zip codes, for instance, are related to city names, localities, and even streets [3]. Having all the advantages of Maximum entropy Markov models (MEMMs), CRFs also solve the label bias problem by letting the probability of a transition between labels also depend on past and future elements and not only on the current address element [3]. “The essential difference between CRFs and MEMMs is that the underlying graphical model structure of CRFs is undirected, while that of MEMMs is directed” [16] (p. 2). Considering, as an example, the address “3B Records, 5 Slater Street, Liverpool L1 4BW”, an HMM parser would erroneously predict the first and second labels as standing for number (“3B”) and street (“Records”), respectively, whereas the CRF parser, when reaching the actual property number (5), would give a higher score to the current label in order to revise it to a property number and the previous label (3B Records) to a business name [3]. Another recent approach to address parsing is based on so-called “word-embeddings”, the name given to the vector representation of words [3]. An implementation of such method is word2vec [20], an unsupervised neural network language which aims to make predictions about the next words by modeling the relationships between a given word and the words in its context, based on two possible architectures: the continuous skip-gram model (Skip-Gram) and the continuous bag-of-words model (CBOW) [20]. The latter is usually chosen over the former, since it is trained by inferring the meaning of a particular word from its context [9].

A practical comparison between HMMs, CRFs, and a CRF augmentation with word2vec is undertaken in Comber and Arribas-Bel [3]. The VGI based *Libpostal* library (<https://github.com/openvenues/libpostal> (accessed on 9 November 2021)), which trains a CRF model on 1 billion street addresses from OSM data, was used for the segmentation task. Although the obtained results are broadly consistent in terms of precision, the classifiers using the HMM technique present lower recall values than the ones obtained by the CRF, meaning that both methods are

capable of distinguishing true positives from false positives, but the CRF is able to classify a greater proportion of matches [3]. The augmented version of the CRF model does not outperform the results obtained by the original one but presents the advantage of not committing the user to a particular string distance and its biases [3]. In another recent work by the same author [21], a predictive model for address matching is proposed, based on recent innovations in machine learning and on a CRF parser for the segmentation of address strings. The biggest contribution of the paper at hand, however, is the thorough documentation of all the steps required to execute the proposed model's workflow. In other papers included in the present literature review, CRFs are used as a benchmark model, e.g.,: Dani et al. [22] or in combination with other methods, which will be further addressed [23][24][25].

Other less recent approaches have been proposed for address parsing/segmentation, namely within address standardization studies aiming at minimizing the size of labelled training data. One such example is the work by Kothari et al. [26], in which a nonparametric Bayesian approach to clustering grouped data, known as hierarchical Dirichlet process (HDP) [27], is used with a view to discover latent concepts representing common semantic elements across different sources and allow the automatic transfer of supervision from a labeled source to an unlabeled one. The obtained latent clusters are used to segment and label address registers in an adapted CRF classifier, with experimental results pointing to a considerable improvement in classification accuracy [26]. A similar approach is proposed by Guo et al. [28], in which paper a supervised address standardization method with latent semantic association (LaSA) is presented, with a view to capture latent semantic association among words in the same domain. The obtained experimental results show that the performance of standardization is significantly improved by the proposed method. Expert systems have also been proposed, namely by Dani et al. [22], in which paper a Ripple Down Rules (RDR) framework is proposed with a view to enable a cost-effective migration of data cleansing algorithms between different datasets. RDR allows the incremental modification of rules and to add exceptions without unwanted side effects, based on a failure driven approach in which a rule is only added when the existing system fails to classify an instance [22]. After comparison with traditional machine learning algorithms and a commercial system, experimental results show that the RDR approach requires significantly less rules and training examples to reach the same accuracy as the former methods [22].

Tree-based models have been proposed to handle automatic handwritten address recognition, which consists of a particular address parsing/segmentation task mostly studied by Chinese researchers, due to the greater complexity of the Chinese language (larger character set, different writing styles, great similarity between many of the characters) [29]. In the paper by Jiang et al. [29], a suffix tree is proposed to store and access addresses from any character. In relation to previous approaches also based on a tree data structure, the proposed suffix tree is able to deal with noise and address format variations. Basically, a hierarchical substring list is firstly built, after which the obtained input radicals are compared with candidate addresses (filtered by the postcode) with a view to optimize a cost function, combining both recognition and matching accuracy [29]. A correct classification rate of 85.3% is obtained in the experimental results. However, according to Wei et al. [30], the recognition accuracy of character-level-tree (CLT) models is dependent on the completeness of the address list on which they are based. In order to overcome this limitation, the authors propose a structure tree built at the word level (WLT), in which each node consists of an address word and the path from the root to the leaf corresponds to a standardized address format.

After initial recognition by a character classifier, segment candidate patterns are mapped to candidate address words based on the WLT database. In the final phase (path matching), candidate address words' scores are summed in order to obtain the address recognition result [30]. The obtained experimental results show that the proposed method outperforms four benchmarking methods, including the previously mentioned suffix tree. Address tree models for address parsing and standardization are also proposed in the papers by Tian et al. [31], Liu et al. [32] and Li et al. [33]. In the first two, the address tree model is mainly used for rule-based validation and error recognition, by providing information about the hierarchy of Chinese addresses and, in the case of the paper by Li et al. [33], latent tree structures are designed with a view to capture rich dependencies between the final segments of an address, which do not always follow the same order.

Within the address element-based methods, it is also worth highlighting geocoding as a means to enhance address standardization, through the correction of misspellings and the filling of missing attributes, some of the most common errors found in postal addresses [34]. After successful matching with a record from a standardized reference database (like Google Maps or OSM), reverse geocoding can be performed to obtain a valid and complete representation of the queried address. In the case of geocoded databases like GNAF, for instance [17], geographic coordinates can be used to calculate the spatial proximity between different records for conducting distance-based spatial analyses and for record linking purposes between different databases (up to the house number). Another important application of address geocoding relates to the matching of historical address records (such as census records) with contemporary data, by attaching grid references to the former in order to perform longitudinal spatial demographic analyses [35]. However, the successful automated geocoding of residential addresses depends on a number of factors, namely population densities (with positional error increasing as population density decreases) [35][36], the completeness of an address (existence or not of a number and street name), and changes in street names, among others [35]. These limitations can be tackled by the previous standardization and enrichment of addresses [37] and the choice of the most adequate geocoding method, including the use of property data [36] or the use of hybrid geocoding approaches [38].

With the advancement of deep learning methods, various authors have recently proposed the adoption of the previously mentioned extensions to RNNs (namely, LSTMs, GRUs, and GCNs), in order to better cope with nonstandard address records and highly dissimilar toponyms. LSTMs and GRUs are both composed of gates, which consist of neural networks that regulate the flow of information from one time step to the next, thereby helping to solve the short memory problem. In particular, GRUs have two gates—update and reset gates—and LSTMs, three gates—input, forget, and output gates [39]. The amount of fresh information added through the input gate in LSTMs is unrelated to the amount of information retained through the forget gate. In GRUs, the retention of past memory and the input of new information to memory are not independent [39].

Within the present literature review, several of the considered papers propose these types of methods, namely the ones by Santos et al. [40], Lin et al. [9], J. Liu et al. [25], Shan et al. [7][41], P. Li et al. [42], and Chen et al. [43]. To take into account contextual information both from previous and future tokens, by processing the sequence in two directions, bidirectional LSTM (BiLSTM) or GRU layers are also being employed in the great majority of these studies. The best performing models further connect the encoder and decoder through an attention mechanism in

order to assign higher weights to the most important features [7][9][41]. With a view to reduce overfitting and enhance the classification models' generalization abilities, a dropout regularization layer is also normally added [9][40][25]. The ESIM model [44] consists of an illustrative example of a deep learning architecture based on the principles previously described. After address tokenization (with the help of gazetteers and dictionaries, in the case of more complex languages, with no natural separators) and the obtaining of vector representations of the different (labelled) address pairs (based on word2vec), the ESIM model is employed through the following four layers [9]:

- An input encoding layer, that encodes the input address vectors and extracts higher-level representations using the bidirectional long short-term memory (BiLSTM) model;
- A local inference modelling layer, that makes local inference of an address pair using a modified decomposable attention model [45];
- An inference composition layer, responsible for making a global inference between two compared address records based on their local inference, in which average and max pooling are used to summarize the local inference and output a final vector with a fixed length;
- Finally, a prediction layer, based on a multilayer perceptron (MLP) composed of three fully connected layers with rectified linear unit (ReLU), tanh and softmax activation functions, is used to output the predictive results of address pairs (that is, whether there is a match or not).

In terms of performance, all of the previously presented deep learning methods achieve a greater matching accuracy than the traditional text-matching models. In the case of the BiLSTM model proposed by Lin et al. [9], the precision, recall, and F1 score on the test set all reached 0.97, against the 0.92 scores achieved by the second-best performing model (Jaccard similarity coefficient + RF method). The deep neural network based on GRUs, to categorize toponym pairs as matches or non-matches, proposed by Santos et al. [40], also outperforms traditional text-matching methods, achieving an increase of almost 10 points in most of the evaluation metrics (namely, accuracy, precision, and F1). The LM-LSTM-CRF+BP neural networks model proposed by J. Liu et al. [25] achieves an accuracy and F1 score of 87%, compared with average scores of 70% by the benchmark methods (word2vec and edit distance). The address GCN method proposed by Shan et al. [7] also presents better results, on both precision (up to 8%) and recall (up to 12%), than the existing methods, which include the DeepAM model previously proposed by the same author, based on an encoder-decoder architecture with two LSTM networks [41]. The Bi-GRU neural network proposed by P. Li et al. [42] presents a similar performance to that shown by a Bi-LSTM neural network (F1 score of 99%) and a higher performance than unidirectional GRU and LSTM neural networks (F1 score of 93%), as it would be expected. Finally, the attention-Bi-LSTM-CNN network (ABLC) proposed by Chen et al. [43] achieves an improvement of 4–10% more accuracy than the baseline models, which include the previously mentioned ESIM model, presenting the second-best overall performance.

In two of the most recent studies included in the present literature review also based on deep learning methods, bidirectional encoder representations from Transformers (BERT) are proposed instead. The first one is the study by

Xu et al. [46], which proposes a method for fusing addresses and geospatial data based on BERT (in what concerns the learning of addresses' semantics) and a K-Means high-dimensional clustering algorithm, enhanced by innovative fine-tuning techniques, to create a geospatial-semantic address model (GSAM). The computational representation extracted from GSAM is then employed for predicting address location, based on a neural network architecture to minimize the Euclidean distance between the predicted and real coordinates. In the second study [47], a new address element recognition method is proposed, for dealing with address elements with shortchange periods (streets, lanes, landmarks, points of interest names, etc.) which still have not been included in a segmentation dictionary. A model based on BERT is first applied to obtain the vector representations of the dataset and learn the contextual information and model address features, followed by the use of a CRF to predict the tags, with new address elements being recognized according to the tag [47].

In terms of performance, the GSAM model [46] achieves a classification accuracy above 0.97, against a minimum expected accuracy of 0.91 by other methods; the BERT-CRF model [47] achieves the highest F1 score on generalization ability (0.78), when compared to benchmark models combining word2vec, BiLSTM, and CRF methods (with an average F1 score of 0.41), as well as an equally high F1 score on the testing dataset (0.95).

Although related to POIs' locations and descriptions, two final articles (both published in 2021) are worth mentioning, due to the combined use of the previously presented approaches and spatial correlation/reasoning methods. The first of this studies [2] presents a method for identifying POIs in large POI datasets in a quick and accurate manner, based on: an enhanced address-matching algorithm, combining string, semantic, and spatial similarity, within an ontology model describing POIs' locations and relationships, in order to support the transition from semantic to spatial; a grid-based algorithm capable of achieving compact representations of vast qualitative direction interactions between POIs and performing quick spatial reasoning, through the fast retrieval of direction relations and quantitative calculations. The second of the studies [8] proposes an unsupervised method to segment and standardize POIs' addresses, based on a GRU neural network combined with the spatial correlation between address elements for the automatic segmentation task, and a tree-based fuzzy matching of address elements for the standardization task, with experimental results pointing to a relatively high accuracy.

2.2. Research Gaps

Within the more recently published papers considered in the present literature review, the most relevant opportunities for further work can be summarized as follows: the use of representative and large enough datasets [46]; the inclusion of duplicate place names, in order to enable the application of the proposed methodology to a national address database [9]; to improve accuracy, different weights might be assigned to the address-element vectors depending on their hierarchy [9]; the need to fine tune the weight ratio of fused features, such as coordinates and the semantic representation of addresses, alongside the improvement of the underlying concatenation method and measurement metrics [46]; the adoption of systematic approaches for tuning hyper-parameters and experimenting with different architectures [40]; the need to involve more complex spatial objects and relations [2][8]. Some of the limitations highlighted in less recent studies, however, should also be taken in consideration in the application of the most recent methods, like the need to tackle privacy and confidentiality

issues [17] when using personal quasi-identifiers such as addresses (especially, residential ones). Another concern that should be addressed and which was tackled in some of the earlier studies [22][26][28] is related to the minimization of human labelling when generating both training and test data. Lastly, no references have been found about the use of genetic programming (GP) [48] in the field of semantic address matching. GP has several advantages over other machine learning methods, including the ability to provide results that can be easily interpreted, based on programs, rules, or functions, as well as the ability to easily incorporate specific knowledge about a problem, despite its efficiency issues, which are primarily due to a time-consuming fitness function computation [49].

References

1. Javidaneh, A.; Karimipour, F.; Alinaghi, N. How Much Do We Learn from Addresses? On the Syntax, Semantics and Pragmatics of Addressing Systems. *ISPRS Int. J. Geo-Inf.* 2020, 9, 317.
2. Cheng, R.; Liao, J.; Chen, J. Quickly Locating POIs in Large Datasets from Descriptions Based on Improved Address Matching and Compact Qualitative Representations. *Trans. GIS* 2021, 1–26.
3. Comber, S.; Arribas-Bel, D. Machine Learning Innovations in Address Matching: A Practical Comparison of Word2vec and CRFs. *Trans. GIS* 2019, 23, 334–348.
4. Sun, Y.; Ji, M.; Jin, F.; Wang, H. Public Responses to Air Pollution in Shandong Province Using the Online Complaint Data. *ISPRS Int. J. Geo-Inf.* 2021, 10, 126.
5. Lee, K.; Claridades, A.R.C.; Lee, J. Improving a Street-Based Geocoding Algorithm Using Machine Learning Techniques. *Appl. Sci.* 2020, 10, 5628.
6. Churches, T.; Christen, P.; Lim, K.; Zhu, J.X. Preparation of Name and Address Data for Record Linkage Using Hidden Markov Models. *BMC Med. Inform. Decis. Mak.* 2002, 2, 9.
7. Shan, S.; Li, Z.; Yang, Q.; Liu, A.; Zhao, L.; Liu, G.; Chen, Z. Geographical Address Representation Learning for Address Matching. *World Wide Web.* 2020, 23, 2005–2022.
8. Luo, A.; Liu, J.; Li, P.; Wang, Y.; Xu, S. Chinese Address Standardisation of POIs Based on GRU and Spatial Correlation and Applied in Multi-Source Emergency Events Fusion. *Int. J. Image Data Fusion* 2021, 12, 319–334.
9. Lin, Y.; Kang, M.; Wu, Y.; Du, Q.; Liu, T. A Deep Learning Architecture for Semantic Address Matching. *Int. J. Geogr. Inf. Sci.* 2019, 34, 559–576.
10. Santos, R.; Murrieta-Flores, P.; Martins, B. Learning to Combine Multiple String Similarity Metrics for Effective Toponym Matching. *Int. J. Digit. Earth* 2017, 11, 913–938.

11. Levenshtein, V.I. Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. *Sov. Phys. Dokl.* 1966, 10, 707–710.
12. Jaro, M.A. Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. *J. Am. Stat. Assoc.* 1989, 84, 414–420.
13. Winkler, W.E. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. *Proc. Sect. Surv. Res. Am. Stat. Assoc.* 1990, 354–359.
14. Forney, G.D. The Viterbi Algorithm. *Proc. IEEE* 1973, 61, 268–278.
15. McCallum, A.; Freitag, D.; Pereira, F. Maximum Entropy Markov Models for Information Extraction and Segmentation. In *Proceedings of the 17th International Conference on Machine Learning*, 2000, San Francisco, CA, USA, 29 June–2 July 2000.
16. Wang, M.; Haberland, V.; Yeo, A.; Martin, A.; Howroyd, J.; Bishop, J.M. A Probabilistic Address Parser Using Conditional Random Fields and Stochastic Regular Grammar. In *Proceedings of the 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, Barcelona, Spain, 12–15 December 2016.
17. Lafferty, J.; McCallum, A.; Pereira, F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data Abstract. In *Proceedings of the 18th International Conference on Machine Learning 2001*, San Francisco, CA, USA, 28 June–1 July 2001; pp. 282–289.
18. Christen, P.; Willmore, A.; Churches, T. A Probabilistic Geocoding System Utilising a Parcel Based Address File. In *Data Mining*; Springer: Berlin/Heidelberg, Germany, 2006; Volume 3755, pp. 130–145.
19. Li, X.; Kardes, H.; Wang, X.; Sun, A. HMM-Based Address Parsing with Massive Synthetic Training Data Generation. *Int. Conf. Inf. Knowl. Manag. Proc.* 2014, 33–36.
20. Fu, Q.; Ding, X.Q.; Liu, C.S.; Jiang, Y. A Hidden Markov Model Based Segmentation and Recognition Algorithm for Chinese Handwritten Address Character Strings. *Proc. Int. Conf. Doc. Anal. Recognit. ICDAR 2005*, 2005, 590–594.
21. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* 2013, arXiv:1301.3781.
22. Comber, S. Demonstrating the Utility of Machine Learning Innovations in Address Matching to Spatial Socio-Economic Applications. *Region* 2019, 6, 17–37.
23. Dani, M.N.; Faruque, T.A.; Garg, R.; Kothari, G.; Mohania, M.K.; Prasad, K.H.; Subramaniam, L.V.; Swamy, V.N. A Knowledge Acquisition Method for Improving Data Quality in Services Engagements. In *Proceedings of the 2010 IEEE International Conference on Services Computing*, Miami, FL, USA, 5–10 July 2010; pp. 346–353.

24. Tang, X.; Chen, X.; Zhang, X. Research on Toponym Resolution in Chinese Text. *Wuhan Daxue Xuebao (Xinxi Kexue Ban)/Geomat. Inf. Sci. Wuhan Univ.* 2010, 35, 930–935.
25. Weinman, J. Geographic and Style Models for Historical Map Alignment and Toponym Recognition. In *Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, Kyoto, Japan, 9–15 November 2017; pp. 957–964.
26. Liu, J.; Wang, J.; Zhang, C.; Yang, X.; Deng, J.; Zhu, R.; Nan, X.; Chen, Q. *Chinese Address Similarity Calculation Based on Auto Geological Level Tagging Jing*; Springer International Publishing: Cham, Switzerland, 2019; Volume 2.
27. Kothari, G.; Faruque, T.A.; Subramaniam, L.V.; Prasad, K.H.; Mohania, M.K. Transfer of Supervision for Improved Address Standardization. In *Proceedings of the 2010 20th International Conference on Pattern Recognition*, Istanbul, Turkey, 23–26 August 2010; pp. 2178–2181.
28. Teh, Y.W.; Jordan, M.I.; Beal, M.J.; Blei, D.M. Hierarchical Dirichlet Processes. *J. Am. Stat. Assoc.* 2006, 101, 1566–1581.
29. Guo, H.; Zhu, H.; Guo, Z.; Zhang, X.X.; Su, Z. Address Standardization with Latent Semantic Association. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Paris, France, 28 June–1 July 2009; pp. 1155–1163.
30. Jiang, Y.; Ding, X.; Ren, Z. A Suffix Tree Based Handwritten Chinese Address Recognition System. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Curitiba, Brazil, 23–26 September 2007; Volume 1, pp. 292–296.
31. Wei, X.; Lu, S.; Wen, Y.; Lu, Y. Recognition of Handwritten Chinese Address with Writing Variations. *Pattern Recognit. Lett.* 2016, 73, 68–75.
32. Tian, Q.; Ren, F.; Hu, T.; Liu, J.; Li, R.; Du, Q. Using an Optimized Chinese Address Matching Method to Develop a Geocoding Service: A Case Study of Shenzhen, China. *ISPRS Int. J. Geo-Inf.* 2016, 5, 65.
33. Liu, Q.; Wang, D.; Lu, H.; Li, C. *Handwritten Chinese Character Recognition Based on Domain-Specific Knowledge*; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; Volume 2, pp. 221–231.
34. Li, H.; Lu, W.; Xie, P.; Li, L. Neural Chinese Address Parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, MN, USA, 2–7 June 2019; Volume 1, pp. 3421–3431.
35. Koumarelas, I.; Kroschk, A.; Mosley, C.; Naumann, F. Experience: Enhancing Address Matching with Geocoding and Similarity Measure Selection. *J. Data Inf. Qual.* 2018, 10, 1–16.

36. Walford, N.S. Bringing Historical British Population Census Records into the 21st Century: A Method for Geocoding Households and Individuals at Their Early-20th-Century Addresses. *Popul. Space Place* 2019, 25, e2227.
37. Cayo, M.R.; Talbot, T.O. Positional Error in Automated Geocoding of Residential Addresses. *Int. J. Health Geogr.* 2003, 2, 1–12.
38. Cortes, T.R.; da Silveira, I.H.; Junger, W.L. Improving Geocoding Matching Rates of Structured Addresses in Rio de Janeiro, Brazil. *Cad. Saude Publica* 2021, 37, e00039321.
39. Shah, T.I.; Bell, S.; Wilson, K. Geocoding for Public Health Research: Empirical Comparison of Two Geocoding Services Applied to Canadian Cities. *Can. Geogr.* 2014, 58, 400–417.
40. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv* 2014, arXiv:1412.3555.
41. Santos, R.; Murrieta-Flores, P.; Calado, P.; Martins, B. Toponym Matching through Deep Neural Networks. *Int. J. Geogr. Inf. Sci.* 2018, 32, 324–348.
42. Shan, S.; Li, Z.; Qiang, Y.; Liu, A.; Xu, J. DeepAM: Deep Semantic Address Representation for Address Matching; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; Volume 3.
43. Li, P.; Luo, A.; Liu, J.; Wang, Y.; Zhu, J.; Deng, Y.; Zhang, J. Bidirectional Gated Recurrent Unit Neural Network for Chinese Address Element Segmentation. *ISPRS Int. J. Geo-Inf.* 2020, 9, 635.
44. Chen, J.; Chen, J.; She, X.; Mao, J.; Chen, G. Deep Contrast Learning Approach for Address Semantic Matching. *Appl. Sci.* 2021, 11, 7608.
45. Chen, Q.; Ling, Z.; Jiang, H.; Zhu, X.; Wei, S.; Inkpen, D. Enhanced LSTM for Natural Language Inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, BC, Canada, 30 July–4 August 2017; Volume 1, pp. 1657–1668.
46. Parikh, A.P.; Täckström, O.; Das, D.; Uszkoreit, J. A Decomposable Attention Model for Natural Language Inference. *arXiv* 2016, arXiv:1606.01933.
47. Xu, L.; Du, Z.; Mao, R.; Zhang, F.; Liu, R. GSAM: A Deep Neural Network Model for Extracting Computational Representations of Chinese Addresses Fused with Geospatial Feature. *Comput. Environ. Urban Syst.* 2020, 81, 101473.
48. Zhang, H.; Ren, F.; Li, H.; Yang, R.; Zhang, S.; Du, Q. Recognition Method of New Address Elements in Chinese Address Matching Based on Deep Learning. *ISPRS Int. J. Geo-Inf.* 2020, 9, 745.
49. Koza, J.R. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*; MIT Press: Cambridge, MA, USA, 1992.

50. Araujo, L. Genetic Programming for Natural Language Processing. *Genet. Program. Evolvable Mach.* 2020, 21, 11–32.
-

Retrieved from <https://encyclopedia.pub/entry/history/show/42383>