GeoRoBERTa: A Transformer-based Approach for Semantic Address Matching

Yassine Guermazi[1], Sana Sellami[1] and Omar Boucelma[1]

[1]Aix Marseille Univ, CNRS, LIS, Marseille, France

Abstract

In this paper, we describe a solution for a specific *Entity Matching* problem, where entities contain (postal) address information. The matching process is very challenging as addresses are often prone to (data) quality issues such as typos, missing or redundant information. Besides, they do not always comply with a standardized (address) schema and may contain polysemous elements. Recent address matching approaches combine static word embedding models with machine learning algorithms. While the solutions provided in this setting partially solve data quality issues, neither they handle polysemy, nor they leverage of geolocation information. In this paper, we propose *GeoRoBERTa*, a semantic address matching approach based on RoBERTa, a Transformer-based model, enhanced by geographical knowledge. We validate the approach in conducting experiments on two different real datasets and demonstrate its effectiveness in comparison to baseline methods.

1. Introduction

Entity Matching (EM) is the problem of identifying data instances that refer to the same real-world entities [1, 2]. In this paper, we address a specific EM problem where entities consist of postal addresses. More precisely, *Given two postal addresses A and B, do those addresses refer to the same real world (address) entity ?* We coin this problem as *Address Matching* although that terminology may also refer to either work based on *Geocoding* [3], or to software tools such as *PlaceKey*[1]. Address matching is a crucial task for various location-based businesses as one may lose clients or prospects in case of delivery failure. It is a challenging one, especially in absence of a standard address model.

Formally, the address matching task may be considered as a binary classification problem [4, 3, 5, 6, 7] where the predicted class is either *Match* or *No Match.* However, given two companies with the same name, it is important to identify addresses that are partially similar, such as those having the same city and the same road but differ in the house number or in the case where both addresses are correct but one of them corresponds to a former address company, in order to complete addresses with up-to-date information. As a result, we consider the problem as a multiclass classification one in adding a $PartialMatch$ class. Table 1 shows examples of address matching. Given two Senegalese addresses A and B, the first pair illustrates the case where there is no similarity between address elements apart from the City Dakar (*NoMatch* label). The second address pair has a *PartialMatch* label as there is a similarity between at least one of its elements (Road: Avenue Lamine Gueye), apart from the similarity between the City Dakar. The last row represents an example of a *Match* between addresses as all their elements are similar (except the missing PoBox in address A).

Former address matching approaches [6, 7] are based on similarity measures and matching rules. However, these methods perform a structural comparison of addresses and are unable to identify some relationship between two addresses when they have few literal overlaps [3]. In such cases, semantic address matching is required for identifying/exhibiting semantic similarities between addresses that have the same location with different representations [8].

Recently, semantic address matching solutions have been proposed [4, 3, 5], based mainly on word embedding models combined with classical Machine Learning (ML) or Deep Learning (DL). Nevertheless, these solutions may be impacted by the presence of polysemous words since they are based on static word embedding models. Polysemy cases may occur in an address when it contains a place name that refers to different places in a country or worldwide as illustrated in Table 2. Identifying and resolving polysemic situations is mandatory to avoid matching distortion. This has led to the advent of transformer-based solutions [9] which have shown promising results on general Entity Matching [10, 11] thanks to their highly contextualized embedding.

This motivated us to explore the effectiveness of Transformers in address matching by proposing an approach based on RoBERTa [12], a pre-trained Transformer language model, for address matching in the context of French-speaking countries. Nevertheless, since these models produce address embedding mainly from linguistic contexts, they may miss some (domain) knowledge,

Published in the Workshop Proceedings of the EDBT/ICDT 2023 Joint Conference (March 28-March 31, 2023, Ioannina, Greece)

yassine.guermazi@lis-lab.fr (Y. Guermazi); sana.sellami@lis-lab.fr (S. Sellami); omar.boucelma@lis-lab.fr (O. Boucelma)

[1] https://www.placekey.io/

Table 1
Examples of address pairs with their corresponding matching label

Address A	Address B	Label
Medina 39 X 18 Dakar	16 Rue Parchappe Dakar	NoMatch
25 Avenue Lamine Gueye Dakar	59 Avenue Lamine Gueye X Galandou Diouf Dakar	PartialMatch
4373 Sicap Amitie 3 Dakar	Sicap Amitié 3 Numero 4373 BP 3110 Dakar	Match

Table 2
Examples of Address Matching Challenges

Challenge	Address A	Address B	Description
Semantic similarity	Immeuble Azur Dakar Senegal	12 Boulevard Djily Mbaye Dakar Senegal	*Immeuble Azur* and *12 Boulevard Djily Mbaye* refer to the same geographic location, in Dakar
Polysemy	Les Dunes 9002 Rue Des **Garennes** France	Les **Garennes** 78130 Les Mureaux France	There is no match between A and B although they contain the same place name "Garennes" which is a polysemous word as it refers to two different places: a Road and an industrial zone (in two different cities).

which is difficult to learn from raw texts. Therefore, we propose to enhance the contextual address embedding of RoBERTa by two types of geographical knowledge, obtained from address tag embedding and address geographic coordinates.

The contributions of this paper can be summarized as follows:

- We defined *GeoRoBERTa*, a semantic address matching approach, which relies on RoBERTa, a transformer-based model.
- We injected two types of geographical knowledge into RoBERTa: address tag embedding and geographic coordinates encoding. This enables better handling of polysemy and better identification of semantic similarity between addresses.
- We conducted an extensive experimental study where *GeoRoBERTa* is compared to baseline methods. Real (unstructured and structured) data, consisting of French postal addresses, has been used.

The rest of the paper is organized as follows: Related work on address matching is reviewed in Section 2. Section 3 presents a formalization of the problem. We describe our solution in Section 4, and present experimental results in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

Address matching pipeline [13, 14] is generally composed by three steps: (1) address parsing, i.e., decomposition of an address into its different components (e.g. street name, zip code), (2) generation of an embedded address vector by means of word embedding models and (3) application of a ML or a DL model resulting in a binary address classification (Match, No Match). Word embedding techniques have gained momentum for solving the semantic address matching problem. They are integrated in address matching pipeline. For example, several studies [13, 14, 3, 5] adopted the same pipeline with different used techniques in the three steps: CRF model, Trie syntax tree algorithm, jieba library[2] or rule based method [14] as address parser, Word2vec [15] or fastText [16] as word embedding models and several ML (e.g. SVM, XGBoost) and DL models (e.g. enhanced sequential inference model, Bi-LSTM, CNN) as classifiers. These works have shown the effectiveness of their proposed approaches compared to baseline methods (non word embedding-based methods) thanks to their capacity to detect semantic similarity between address attributes.

However, these approaches may present two weaknesses. The first one is related to the management of polysemous cases. In fact, these approaches are based on static word embedding models, which cannot handle polysemy as they generate static vector representations of words. Contrariwise, contextual word embedding models, among which the transformer-based ones, resolve this problem thanks to their highly contextualized embedding as demonstrated in entity matching works [11, 10]. The second weakness is related to the leveraging of geographic information. Indeed, these approaches are designed without geographic location information, which ignores the geographic features when performing address matching. And yet, addresses that belong to the same geographic area should have intuitively similar geospatial characteristics. However, these assumptions may fail as existing methods rely only on address text which can contain vernacular content or place synonyms and does not follow a standard structure making them inherently ambiguous. Thus, modelling the problem from linguistic perspective alone is not enough.

In this context, former approaches have specifically

[2] https://github.com/fxsjy/jieba

used geocoding in the address standardization process to obtain the geolocation followed by a reverse geocoding, which generates a complete and proper address before performing the matching. This strategy has been applied for example in [17]. Recently, some works [18, 19] focused on the enrichment of Point Of Interest (POI) embedding using geographic information. The most popular form of this information is the encoding string of the geographic coordinates, obtained by the Geohash geocode system [3]. In [19], authors proposed a POI-Transformers framework to generate POI embeddings in order to perform POI Matching. A POI is defined as an entity composed by four attributes: name, category, address and geographic coordinates. The proposed matching approach consists firstly in generating an embedding vector for each POI by fusing the text embedding of the first three attributes using BERT [20], a Transformer-based model, and the geographic location embedding of the last attribute. Then, the similarity between each pair of POI's embedding is computed using two techniques: cosine similarity and SentEval toolkit [21]. The proposed approach achieves results comparable in terms of performance with those of existing DL-based methods (e.g. DeepER [22], DeepMatcher [23]) on general Entity Matching benchmark datasets but it outperforms them on POI Entity Matching datasets.

In summary, Transformer-based models have proven their effectiveness in general entity matching but they are less explored in the address matching task. Their application on these domain-specific data should also take into account geographic information in addition to the linguistic context. From this perspective, some works start introducing geographical knowledge (geohash encoding) in Transformer-based model to perform, especially, POI matching, but they may miss additional domain information to effectively deal with polysemy. Therefore, in this work, we propose *GeoRoBERTa* a semantic address matching approach based on a pre-trained transformer-based language model (RoBERTa) which incorporates two types of geographical knowledge: address tag embedding and geohash encoding in order to better deal with polysemous cases and to improve the identification of semantic similarity between addresses.

3. Problem Statement

As discussed in Section 1, due to the heterogeneity in address representations, we need to extract some « intuitive/hidden » semantic relationships between addresses. Prior to that, we first present the address model that we adopted in this paper. Then, we provide a definition of *semantic address matching*, along the lines of the one provided in Xu et al [8].

3.1. Address model

Definition 1 (Address Schema). *Given a set of (entity) attributes $\{a_1, .., a_N\}$, an address $A = list\,\{a_1, .., a_n\}$ where a_i is the i-th address token (word) and n is the address length, with $n \leq N$, and $list()$ is a "list" constructor.*

More formally, to cope with different address representations (e.g., France and Senegal in this paper), we distinguish between two types of addresses:

1. A *Simple Address* is a sequence of attributes (Table 3) which are defined by the address model proposed in [24].
2. A *Complex Address* is a composition of (at least) two simple addresses by means of a spatial operator. Table 4 below illustrates the proximity operator op_{prox} and the intersection operator op_{int}, while Table 5 shows two complex Senegalese addresses.

Table 3
Simple Address Attributes

Address Attribute	Tag
Country	*CO*
State	*S*
City	*C*
District	*D*
Zone	*Z*
Road	*RN*
HouseNum	*HN*
POI	*P*
ExtBuilding	*EB*
InBuilding	*IB*
Zipcode	*ZC*
PoBox	*PB*

Table 4
Spatial Operators

Spatial Operator	Tag
op_{int}	*IN*
op_{prox}	*PR*

Table 5
Spatial operators in Senegalese addresses

Examples of Spatial Operators	Examples of Addresses
Intersection operator **X (Intersect)**	2 Avenue Ballaz **X** Avenue De L'Administration Dakar Senegal
Proximity operator **Face (In front of)**	Route De La Gare **Face** Pharmacie Baol Dakar

3.2. Semantic Address Matching

Definition 2 (Semantic Address Matching). *Given two address datasets: $D_1 = \{A_1, .., A_l\}$ and $D_2 = \{B_1, .., B_{l'}\}$, where l and l' are the size of D_1 and D_2 (respectively), the Semantic Address Matching aims to find each address pair (A_i, B_j), satisfying $A_i = B_j$ or $A_i \approx B_j$, where A_i and B_j are simple or complex addresses such as $A_i \in D_1$ and $B_j \in D_2$, $=$ and $\approx$ represent the equality and the approximation operator,*

[3] http://geohash.org/

respectively. The addresses on either side of the equality operator refer to the same real-world object with the same geographic location (coincide with relationship). Whereas, the addresses on either side of the approximation operator are semantically related: there is a specific relationship located in between their attributes (i.e. an address A is located in an address B or vice versa). In this work, the address pairs labels are defined as follows:

- *Match: it is attributed to an address pair, between which there is the relationship coincide with*
- *PartialMatch: it is attributed in two scenarios: (1) there is a relationship located in between an address pair or (2) there is a relationship coincide with between a partial part of an address pair.*
- *NoMatch: otherwise*

4. Proposed Approach

In this section we describe *GeoRoBERTa* (Figure 1), a RoBERTa-based approach and model for semantic address matching.

GeoRoBERTa consists of three main tasks: (1) Data Preprocessing in order to clean data, (2) Geographical Knowledge Generation and (3) Address Matching which is based on a pre-trained RoBERTa model enhanced by the geographical knowledge in order to classify each address pair as either *Match*, *PartialMatch* or *NoMatch.*

4.1. Data Preprocessing

The purpose of this step is to normalize and clean addresses with removing special characters and expanding abbreviations. For that, we adopt a dictionary-based approach which provides the keywords that may be used to define the components of addresses as well as common abbreviations of these words. As we are interested in addresses belonging to French-speaking countries, we extract French keywords from official sources, in France, such as the Post Office, the INSEE [4] service and unofficial sources which generally have common abbreviations, such as the list of abbreviations recognized by the OpenStreetMap [5] query tools. In addition, all addresses are normalized with expanding abbreviations to their corresponding words in the created dictionary which contains a set of keywords that are likely to be used to define address's components (avenue, road, building, etc.) and their abbreviations.

4.2. Geographical Knowledge Generation

4.2.1. Geographic Coordinates Encoding

We augment each address by a geographical knowledge derived from the encoding of geographical location represented as a latitude (lat) and a longitude (long) pair. First, we used Google Geocoding API [6] to convert each address into geographic coordinates (lat and long). Then, we translate the two-dimensional location into geographically meaningful embeddings using Geohash [25] which is a geocoding system that encodes the geographic location of a place into a short string of letters and digits. An important property of geohash is that two places with a long common geohash prefix are close to each other [26].

We append address texts with geohashes to provide the geospatial context to the RoBERTa model. Figure 2 shows an example of geographic coordinates encoding of a French address.

4.2.2. Generation of Address Tag Embedding

It consists of two steps: address parsing and address tag embedding.

(1) Address Parsing: The parsing of an address $A = \{a_1, .., a_n\}$ aims to assign a label l to each word a_i of A among the corresponding list of address tags $Y = \{IB, EB, P, Z, HN, RN, D, IN, PR, PB, ZC, C, S, CO\}$

These tags (Table 3 and 4) are defined following the address model described in section 3.1.

We applied the address parsing method (Figure 3) proposed in [24], thanks to its effectiveness compared to several baseline methods, especially in identifying polysemous address elements. The parsing is based on the use of a RoBERTa model, which generates firstly a contextual representation of an input address A, following these two sub-steps:

- RoBERTa calculates the input representations of A by summing over the token, position, and segment embedding.
- Input address representation goes through 12 transformer encoders which capture the contextual information for each token by self-attention and produces a sequence of contextual embeddings.

The resulted representation is then provided to a tagging layer (a Fully Connected Layer) to obtain address tags, using the IOB (Inside–outside–beginning) tagging scheme [27], where a token is labeled as B-tag if it is at the beginning of the address element, or I-tag if inside the address element but not first, otherwise O-tag.

[4] https://www.sirene.fr/sirene/public/variable/typvoie
[5] https://wiki.openstreetmap.org/wiki/Name_finder:Abbreviations
[6] https://developers.google.com/maps/documentation/geocoding

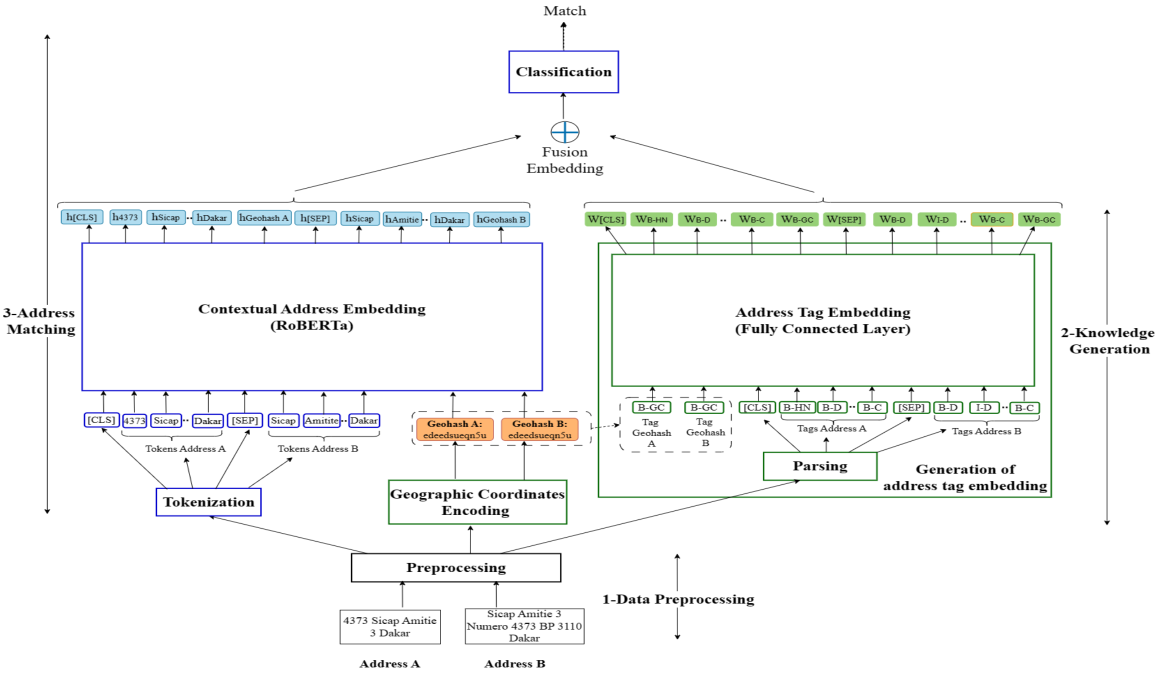

Figure 1: GeoRoBERTa Architecture

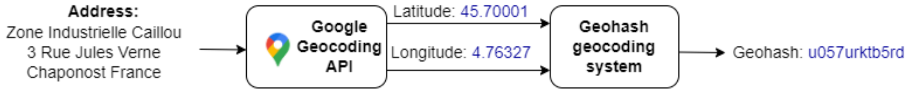

Figure 2: Geographic Coordinates Encoding

The tagging layer takes as input the last hidden state of the obtained sequence of contextual embeddings and provides as result the prediction of the tags.

(2) Address Tag Embedding: The output of the parsing step of the address A (respectively address B) is n tags (respectively m tags). Since these tags are at the word level, their length is equal to the length n of A (respectively the length m of B). We augment these tags by another tag (*B-GC*) which represents the corresponding geohash of each address. Then, we use a look-up table to map these tags to identifiers and feed a linear layer to obtain the representations of the tags of the address pair.

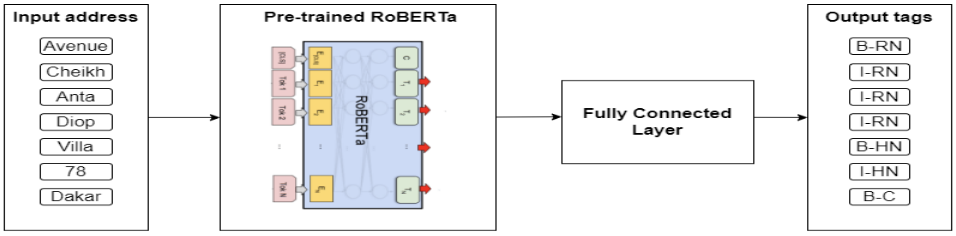

Figure 3: Address Parsing Method

4.3. Address Matching

It consists of two steps (Figure 1): (1) generating a fusion of two vector representations which are the contextual vector representation of the address pair and the vector representation of the address pair tags, and (2) a classification of each pair according to resulted vectors.

4.3.1. Vectors Fusion

We fuse two embedding vectors as follows:

1. Contextual embedding of address pair: The byte pair encoding (BPE) tokenizer [7] of RoBERTa was used to encode the input addresses into tokens. These tokens and the two geohashes, representing the address pair (A, B), form the input to the pre-trained RoBERTa model. Then, this model generates the contextual vectors representations of the address pair (A, B).
2. Tags embedding of address pair: They are generated from the previous step (as described in Section 4.2.2).

The fusion of vectors is performed by a concatenation function which is the most popular feature-level fusion methodology [28, 29].

4.3.2. Address Pair Classification

It is performed using a fully connected layer (a linear layer), which is the classifier layer by default in RoBERTa packages. This layer takes as input the resulting embedding fusion vector and generates as output the class logits (probabilities), knowing that the objective of the training is the CrossEntropy. Then, the Argmax function is applied to these probabilities to get the predicted class.

5. Evaluation

In this section, we describe the experiments carried out in order to evaluate our address matching approach. Source code is available at the following Git repository: https://github.com/MatchSystem/GeoRoBERTa.

5.1. Experimental Settings

5.1.1. Dataset Description

Our experiments are conducted on two real-world datasets representing addresses from two French-speaking countries: (1) France and (2) Senegal. Unlike several known data contest competitions (e.g. Kaggle, SIGIR), there is no such real dataset for these countries in the context of address matching. Therefore, we proceed with the following steps in order to create our own labeled dataset:

1. **Step 1: Address collection**: The French dataset has been collected (on July 12, 2022) from the Legal Entity Identifier (LEI) database [8](the French company's addresses) and contains 40000 addresses, whereas the Senegalese dataset is generated from Senegalese company directories [9] and contains 5000 addresses.
2. **Step 2: Address pairs creation:** For each dataset, we create a labeled set, composed by address pairs and their corresponding label using different strategies based on [3]:
   - For *Match* address pair: the creation of these pairs are performed using 3 strategies: (1) a simple clone of the address, (2) attribute removal, to create semantic similar elements such as removing either the street address (*HouseNum+Road*) or the *ExtBuilding* or *POI* if they both exist, and (3) token removal, by a deletion of a randomly sampled span of tokens.
   - For *PartialMatch* address pair: we use mainly the attribute removal strategy to create the address pairs, while ensuring that City elements are similar and there is at least a similarity between another element of the address pair.
   - For *NoMatch* address pair: for each address from the dataset (French or Senegalese), three strategies are used to choose the second address of the pair: (1) Random selection of an address from the dataset (the two datasets do not contain duplicate address), (2) Selection of an address with the same city and, (3) Selection of an address with literal overlap.

   The frequency of the classes (labels) of address pairs for the two datasets (French dataset denoted J_F and Senegalese dataset denoted J_S) is given by the Table 7. Besides, J_F and J_S are split into the training, validation, and test sets using the ratio of 3:1:1. Table 6 shows a sample of the training set of J_S, on which RoBERTa is trained.

5.1.2. Compared Methods

We compare *GeoRoBERTa* with baseline methods used in some address matching related works [13, 14]. We

[7] https://huggingface.co/docs/transformers/tokenizer_summary

[8] https://www.gleif.org/en/lei-data/gleif-golden-copy/download-the-golden-copy#/

[9] https://www.goafricaonline.com/sn

Table 6
J_S training set (extract)

$Address(A)$+$Geohash_A$	$Tags_A$	$Address(B)$+$Geohash_B$	$Tags_B$	Class
[’Medina’, ’39’, ’X’, ’18’, ’Dakar’, ’edeedbud527v’]	[’B-D’,’B-RN’,’B-IN’, ’B-RN’, ’B-C’,’B-GC’]	[’16’, ’Rue’, ’Parchappe’, ’Dakar’, ’edee7q7yjdmb’]	[’B-HN’, ’B-RN’, ’I-RN’, ’B-C’, ’B-GC’]	NoMatch
[’25’, ’Avenue’, ’Lamine’, ’Gueye’, ’Dakar’, ’edee7pwrhn7s’]	[’B-HN’,’B-RN’,’I-RN’, ’I-RN’, ’B-C’, ’B-GC’]	[’59’, ’Avenue’, ’Lamine’, ’Gueye’, ’X’, ’Galandou’,’Diouf’, ’Dakar’, ’edee7pn94qpv’]	[’B-HN’, ’B-RN’,’I-RN’, ’I-RN’, ’B-IN’,’B-RN’, ’I-RN’, ’B-C’, ’B-GC’]	PartialMatch
[’4373’, ’Sicap’, ’Amitie’, ’3’, ’Bp’, ’3110’, ’Dakar’, ’edeedsueqn5u’]	[’B-HN’, ’B-D’, ’I-D’,’ I-D’, ’B-PB’, ’I-PB’, ’B-C’, ’B-GC’]	[’Sicap’, ’Amitié’, ’3’,’Numero’, ’4373’, ’BP’, ’3110’, ’Dakar’, ’edeedsueqn5u’]	[’B-D’,’I-D’,’I-D’,’B-HN’ ,’I-HN’,’B-PB’,’I-PB’, ’B-C’, ’B-GC’]	Match

Table 7
Frequency of labels in French and Senegalese datasets

Label	J_F	J_S
NoMatch	20000	2500
PartialMatch	10000	1250
Match	10000	1250

also compare with variants of *GeoRoBERTa* without the Geographic Tag embedding (GT) and/or the Geohash encoding (GH) Knowledge to evaluate the effectiveness of the model after the injection of each knowledge type. We summarize these approaches below.

- *Word2vec + XGBoost* [13]: in adopting this approach, we trained a Word2vec model over an address corpus (section 5.2.2) using Gensim [10] library with vectors of dimension 100, a window size of 15. Then, the model is used to generate word embedding of each address of the training dataset. We obtain the embedding of each address attribute by averaging all their words embedding. The cosine similarity between the embedding of the same type of address attributes is used as features in a XGBoost classifier implemented using scikit-learn [11].
- *fastText + SVM* [14]: fastText model is firstly used to obtain address embedding. It is trained over an address corpus (section 5.2.2) using Gensim library with vectors of dimension 100, a window size of 15. Then, features are obtained by applying cosine similarity between embedding of the same and of the different type of address attributes. These features serve as input to a SVM classifer.
- *RoBERTa*: This base form of *GeoRoBERTa* corresponds to fine-tuning the pre-trained RoBERTa on address matching. We did not inject any geographical knowledge. This variant is similar to the entity matching approach proposed in [11].
- *GeoRoBERTa(GT)*: In this version, only geographic tags embedding knowledge has been added to RoBERTa, i.e., we removed the *Geographic Coordinated Encoding* block in Figure 1.
- *GeoRoBERTa(GH)*: This version includes the geohash encoding knowledge only, i.e., we removed the *Generation of address tag embedding* block in Figure 1.

As illustrated in Section 4, *GeoRoBERTa* takes as input the whole address pairs augmented with the corresponding geohash, to the contrary of the two baseline approaches [13, 14] where the input is the set of attributes of each address pair. For a fair comparison, we added two attributes to each address pair corresponding to its geohash strings.

5.2. Evaluation Setup

5.2.1. Hardware

The experiments were carried out on a Dell PC with the following characteristics:

- **Processor:** Intel® Core 8th (4 core), HT, 1.9Ghz, 8Mo, 15W / UHD 620
- **Hard disk:** SSD 512Go M.2 SATA
- **RAM:** 16Go 2400MHz DDR4 (2x8Go)
- **Operating system:** Microsoft Windows 10 Pro, 64 bits

The compared approaches are executed on "NVIDIA Tesla K80" GPU using Google Colab (with 12 GB of RAM).

5.2.2. RoBERTa pre-training and fine-tuning

RoBERTa-base architecture (12-layer, 768-hidden, 12-heads, 125M parameters) is used for pre-training and fine-tuning. The model is pre-trained to optimize the Masked Language Modeling objective. RoBERTa pre-training was performed with the Pytorch framework [12] and Transformers library [13] with a vocabulary size of 30000 tokens. We generated two pre-trained RoBERTa models corresponding to each of the following corpora:

[10] https://pypi.org/project/gensim/
[11] https://scikit-learn.org/stable/
[12] https://pytorch.org/
[13] https://huggingface.co/docs/transformers/index

(1) French corpora composed of 1,048,575 addresses [14] and (2) Senegalese corpora composed of 31893 addresses collected from Web business directories [15]/[16]/[17]/[18]. These datasets have been processed according to the steps described in Section 4.1.

5.2.3. Hyperparameters Tuning

GeLU activation is used in RoBERTa with the ADAM Optimizer. For both tasks (parsing and matching), the dropout and learning rates are set respectively to 0.1 and 3e-5 in such a way as to maximize the accuracy in the validation set. To avoid overfitting, we use the early stop technique based on loss validation by setting a maximum number of training epochs (=12) and a batch size of 32.

5.2.4. Evaluation Metric

To evaluate the performance of our model and all the baselines, we use the F-measure, which is the harmonic mean of the precision, the rate of correct predictions, and the recall, the fraction of correct classes being predicted.

$$F - measure = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (1)$$

Table 8
F-measure of Address Matching Approaches.

Approach Name	J_F	J_S
Word2vec + XGBoost	0.917	0.9
fastText + SVM	0.931	0.916
GeoRoBERTa	0.949	0.94

5.3. Results

5.3.1. Comparison with baselines

First, *GeoRoBERTa* is compared to baseline approaches, using the same address parsing method (RoBERTa). The evaluation results, illustrated in Table 8, show that *GeoRoBERTa* outperforms the other approaches on the two datasets thanks to the highly contextualized vector representations of RoBERTa compared to fastText and Word2vec. Besides, the fastText-based approach outperforms the Word2vec-based one due to the richness of the extracted features in the former approach compared to the second one. These features represent the cosine similarity between attributes from different types (e.g., Road vs District) and those from the same type (e.g., Road vs Road). Overall, the performance of the different models, in terms of F-measure, is higher in the case of French addresses (vs. Senegalese ones) due to their structured nature.

Impact of the parsing method To evaluate the impact of the parsing on matching results, we consider three address baseline parsing methods: rules-based [14], CRF-based, and RoBERTa. Parsing evaluation results (Table 9) show that RoBERTa is more accurate than the other methods because it handles polysemous words. Table 10 illustrates the impact of the parsing method on the performance of the address-matching approaches. We note that all the matching approaches combined with a parsing method based on RoBERTa perform better than the approaches combined with CRF or those based on Rules. Besides, the impact of the parsing method is more important with Senegalese data since it contains more polysemous cases.

Table 9
F-measure of Address Parsing Methods.

Method Name	J_F	J_S
Rule-based	0.962	0.905
CRF	0.981	0.943
RoBERTa	0.989	0.959

Table 10
Impact of the Address Parsing Method on the Matching Results.

Matching Approach	**Parsing Method**	J_F	J_S
Word2vec + XGBoost	Rule-based	0.894	0.87
	CRF	0.908	0.888
	RoBERTa	0.917	0.9
fastText + SVM	Rule-based	0.913	0.89
	CRF	0.924	0.906
	RoBERTa	0.931	0.916
GeoRoBERTa	Rule-based	0.938	0.925
	CRF	0.944	0.934
	RoBERTa	0.949	0.94

Table 11
Computation time (sec.) of Address Matching Approaches.

Approach Name	J_F		J_S	
	Training	Evaluation	Training	Evaluation
Word2vec + XGBoost	1381	32	96	10
fastText + SVM	1870	38	114	12
GeoRoBERTa	7843	127	971	36

Runtime We evaluate the different address matching models on their training and evaluation in the test set. Results (Table 11) show that the training time of *GeoRoBERTa* is costly due to the deep transformer based

[14] https://www.data.gouv.fr/fr/datasets/base-sirene-des-entreprises-et-de-leurs-etablissements-siren-siret/#description
[15] https://creationdentreprise.sn/
[16] http://pagesjaunesdusenegal.com/
[17] https://www.goafricaonline.com/
[18] https://www.yelu.sn/

architecture of RoBERTa. On the other hand, evaluation time of *GeoRoBERTa* takes just few seconds (127s for J_F and 36s for J_S).

Table 12
F-measure of Ablation Analysis.

Approach Name	J_F	J_S
RoBERTa	0.935	0.926
GeoRoBERTa(GT)	0.94	0.937
GeoRoBERTa(GH)	0.946	0.93
GeoRoBERTa	0.949	0.94

5.3.2. Ablation Study

We analyze the contribution of each type of geographic knowledge by comparing *GeoRoBERTa* with its variants (described in section 5.1.2). The experimental results are shown in Table 12. We first focus on comparing *GeoRoBERTa(GT)* and *GeoRoBERTa(GH)* to *RoBERTa*. The obtained results show that the injection of geographical knowledge (regardless of their types) slightly improves the performance as we note an increase of F-measure in *GeoRoBERTa(GT)* and *GeoRoBERTa(GH)* compared to *RoBERTa* on the two datasets. In fact, these models are more robust when dealing with semantic similarities and polysemy cases.

Next, we note that the precision results of *GeoRoBERTa(GT)* and *GeoRoBERTa(GH)* are close to each other. Moreover, unlike *GeoRoBERTa(GT)*, *GeoRoBERTa(GH)* can detect semantic similarities between unseen addresses during the pre-training or the training steps, thanks to the geohash, as illustrated in Table 13 (first row's example): *There is a Semantic similarity between Zone Industrielle Les Blanchisseries and Rue Louis Leprince Ringuet: The road exists in the zone area (Similar geohash between the two addresses).*

On the other hand, *GeoRoBERTa(GT)* is more efficient when dealing with polysemy cases thanks to the semantic labels embedding. Indeed, polysemy cases can represent examples of ambiguous addresses that are difficult to geocode as illustrated in Table 13 (second row's example): *Rufisque is a polysemous element which may refer to a Road or a District in Senegal and can be found in different geographical areas. GeoRoBERTa(GH)* did not consider this polysemy case as the two generated geohash are similar, while *GeoRoBERTa(GT)* captures the polysemy and predicts the correct label of the address pair. Furthermore, the quality of geographic coordinates can influence the performance of *GeoRoBERTa(GH)*. In such cases, we note that this model is almost competitive with *RoBERTa* for the Senegalese dataset due to the low accuracy of Google Geocoding API, which is 64 % (Table 14). On the other side, *GeoRoBERTa(GH)* outperforms *GeoRoBERTa(GT)* when dealing with the French dataset for which the geocoding accuracy is better (89%).

Overall, we can note that *GeoRoBERTa* outperforms all its variants against the two datasets as it leverages the two types of incorporated knowledge. The incorporation of geohash encoding allowed us to have a more efficient model able to improve the identification of semantically similar address pairs, mainly when they are not used in the training of RoBERTa. Incorporating address tag embeddings allowed *GeoRoBERTa* to better deal with polysemous cases, (e.g., Senegal).

6. Conclusion

In this paper, we described GeoRoBERTa, a transformer-based address-matching solution that relies on RoBERTa, a pre-trained transformer language model, leveraging two types of geographical knowledge during the matching phase. Extensive experimental evaluations on two real-world datasets show that our solution is effective and outperforms baseline models. Besides, the ablation study demonstrated the positive impact of geographical knowledge injection in improving the matching phase, especially in semantic similarities and polysemy cases.

In the future, we intend to extend this work in two directions: (1) evaluating the impact of the geocoding in the matching result by testing other geocoding solutions, and (2) studying the performance of GeoRoBERTa on dirty address datasets (by injecting spelling errors).

References

[1] A. K. Elmagarmid, P. G. Ipeirotis, V. S. Verykios, Duplicate record detection: A survey, IEEE Trans. Knowl. Data Eng. 19 (2007) 1–16.

[2] P. Christen, Data Matching - Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection, Data-Centric Systems and Applications, Springer, 2012.

[3] Y. Lin, M. Kang, Y. Wu, Q. Du, T. Liu, A deep learning architecture for semantic address matching, Int. J. Geogr. Inf. Sci. 34 (2020) 559–576.

[4] S. Shan, Z. Li, Q. Yang, A. Liu, L. Zhao, G. Liu, Z. Chen, Geographical address representation learning for address matching, World Wide Web 23 (2020) 2005–2022.

[5] J. Chen, J. Chen, X. She, J. Mao, G. Chen, Deep contrast learning approach for address semantic matching, Applied Sciences 11 (2021) 7608.

[6] D. K. Matci, U. Avdan, Address standardization using the natural language process for improving geocoding results, Comput. Environ. Urban Syst. 70 (2018) 1–8.

Table 13
Comparison of Address Matching Approaches

Address A	Address B	True label	RoBERTa	Geo RoBERTa(GT)	Geo RoBERTa(GH)	Geo RoBERTa
Zone Industrielle Les Blanchisseries Voiran France	Rue Louis Leprince Ringuet Voiron France	**Partial Match**	NoMatch	NoMatch	Partial Match	Partial Match
Avenue Ousmane Soce Diop Rufisque Dakar Senegal	Hann Belair Rufisque Dakar Senegal	**NoMatch**	Partial Match	NoMatch	Partial Match	NoMatch

Table 14
Accuracy of the Geocoding System on the French and Senegalese Datasets.

Geocoding system	J_F	J_S
Google Geocoding API	89%	64%

[7] T. Gschwind, C. Miksovic, J. Minder, K. Mirylenka, P. Scotton, Fast record linkage for company entities, in: 2019 IEEE International Conference on Big Data (IEEE BigData), Los Angeles, CA, USA, December 9-12, 2019, IEEE, 2019, pp. 623–630.

[8] L. Xu, R. Mao, C. Zhang, Y. Wang, X. Zheng, X. Xue, F. Xia, Deep transfer learning model for semantic address matching, Applied Sciences 12 (2022) 10110.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 5998–6008.

[10] Y. Li, J. Li, Y. Suhara, A. Doan, W. Tan, Deep entity matching with pre-trained language models, Proc. VLDB Endow. 14 (2020) 50–60.

[11] U. Brunner, K. Stockinger, Entity matching with transformer architectures - A step forward in data integration, in: Proceedings of the 23rd International Conference on Extending Database Technology, EDBT 2020, Copenhagen, Denmark, March 30 - April 02, 2020, OpenProceedings.org, 2020, pp. 463–473.

[12] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019).

[13] S. Comber, D. Arribas-Bel, Machine learning innovations in address matching: A practical comparison of word2vec and crfs, Trans. GIS 23 (2019) 334–348.

[14] Y. Guermazi, S. Sellami, O. Boucelma, Address validation in transportation and logistics: A machine learning based entity matching approach, in: ECML PKDD 2020 Workshops - Workshops of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2020) Ghent, Belgium, September 14-18, 2020, Proceedings, volume 1323 of *Communications in Computer and Information Science*, Springer, 2020, pp. 320–334.

[15] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings, 2013.

[16] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Trans. Assoc. Comput. Linguistics 5 (2017) 135–146.

[17] I. K. Koumarelas, A. Kroschk, C. Mosley, F. Naumann, Experience: Enhancing address matching with geocoding and similarity measure selection, ACM J. Data Inf. Qual. 10 (2018) 8:1–8:16.

[18] J. Jin, Z. Xiao, Q. Qiu, J. Fang, A geohash based place2vec model, in: 2019 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2019, Yokohama, Japan, July 28 - August 2, 2019, IEEE, 2019, pp. 3344–3347.

[19] J. Zhang, C. Zhang, X. Liu, X. Li, W. Liao, P. Liu, Y. Yao, J. Zhang, Poi-transformers: Poi entity matching through poi embeddings by incorporating semantic and geographic information (2021).

[20] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018).

[21] A. Conneau, D. Kiela, Senteval: An evaluation toolkit for universal sentence representations, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018, European Language Resources Association (ELRA), 2018.

[22] M. Ebraheem, S. Thirumuruganathan, S. R. Joty, M. Ouzzani, N. Tang, Distributed representations of tuples for entity resolution, Proc. VLDB Endow. 11 (2018) 1454–1467.

[23] S. Mudgal, H. Li, T. Rekatsinas, A. Doan, Y. Park, G. Krishnan, R. Deep, E. Arcaute, V. Raghavendra, Deep learning for entity matching: A design space exploration, in: Proceedings of the 2018 International Conference on Management of Data, SIG-

MOD Conference 2018, Houston, TX, USA, June 10-15, 2018, ACM, 2018, pp. 19–34.

[24] Y. Guermazi, S. Sellami, O. Boucelma, A roberta based approach for address validation, in: New Trends in Database and Information Systems - ADBIS 2022 Short Papers, Doctoral Consortium and Workshops: DOING, K-GALS, MADEISD, MegaData, SWODCH, Turin, Italy, September 5-8, 2022, Proceedings, volume 1652 of *Communications in Computer and Information Science*, Springer, 2022, pp. 157–166.

[25] Z. Balkic, D. Sostaric, G. Horvat, Geohash and UUID identifier for multi-agent systems, in: Agent and Multi-Agent Systems. Technologies and Applications - 6th KES International Conference, KES-AMSTA 2012,Dubrovnik, Croatia, June 25-27, 2012. Proceedings, volume 7327 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 290–298.

[26] K. Lee, R. K. Ganti, M. Srivatsa, L. Liu, Efficient spatial query processing for big data, in: Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Dallas/Fort Worth, TX, USA, November 4-7, 2014, ACM, 2014, pp. 469–472.

[27] L. A. Ramshaw, M. Marcus, Text chunking using transformation-based learning, in: Third Workshop on Very Large Corpora, VLC@ACL 1995, Cambridge, Massachusetts, USA, June 30, 1995, 1995.

[28] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers), Association for Computational Linguistics, 2018, pp. 2227–2237.

[29] Q. Zheng, J. Zhu, Z. Li, S. Pang, J. Wang, Y. Li, Feature concatenation multi-view subspace clustering, Neurocomputing 379 (2020) 89–102.