A RoBERTa Based Approach for Address Validation

Yassine Guermazi, Sana Sellami, Omar Boucelma

Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
{yassine.guermazi, sana.sellami, omar.boucelma}@lis-lab.fr

Abstract. Address verification is becoming more and more mandatory for businesses involved in parcel or mail delivery. Situations where shipments are returned or delivered to the wrong person (or legal entity) are harmful and may incur several costs to the stakeholders. Indeed, addresses often carry incorrect information that need to be corrected prior to any shipment process. In this paper we propose a 2-step address validation approach consisting of (1) standardization and (2) classification, both steps are based on RoBERTa, a pre-trained language model. Experiments have been conducted on real datasets and demonstrate the effectiveness of the approach in comparison to other methods.

Keywords: Data Quality · Address Cleansing · Address Classification · Natural Language Processing · Deep Learning · Transformers · RoBERTa.

1 Introduction

Bad address data severely impacts many industries such as postal services, e-commerce or transportation businesses. *"Shipment Returned because of Bad address"* or *"Package Delivered to Wrong Address"* are some of the situations that often happen in the general context of parcel (or surface mail) delivery.

In this paper we describe a solution for the *address verification* problem: we propose a cleansing and address validation process, in providing (1) a categorization of dirt (e.g. typos, misspelling, geographic inconsistencies), (2) an address standardization method , and finally (3) an address classification method.

Although the problem is a pretty old one, it recently received a lot of attention, both from industry and academia. In the industry, software vendors such as Experian [1] or Informatica [2] are expanding their businesses in providing address verification/validation solutions. In the academia, some research has focused on address cleansing solutions, including preprocessing and parsing, especially for structured addresses [1–3]. Address validation was usually performed based on geocoding solutions such as geocoding APIs (e.g. Google, Bing). However, these tools are not able to manage unstructured and dirty addresses (e.g. missing attributes, geographic inconsistencies) which are frequent in developing countries.

[1] https://www.edq.com/demos/address-verification/

[2] https://www.informatica.com/products/data-quality/data-as-a-service/address-verification.html

In particular, geographic inconsistencies occur when at least two address attributes do not coexist in the same geographical area. The most frequent are those related to address elements (*city*, *district*, *road*) as illustrated in Table 1.

Table 1: Examples of *Invalid* addresses in Senegal

Inconsistency's type	**Addresses**	**Description**
City inconsistency (CI)	Sicap Amitie 2 Villa Numero 4030 **Louga**	the address does not really exist in **Louga** city
District inconsistency (DI)	**Sicap Amitie III** Vdn Numero 9982 Pres Auchan Dakar Senegal	**Vdn** road does not exist in **Sicap amitie III** district
Road inconsistency (RI)	**Route De Ngor** X Avenue Birago Diop Dakar Senegal	there is no intersection (X) between Route (i.e. Road) and Avenue

Table 2: An example of polysemy in Senegalese addresses

Polysemous Word	**Referring Place**	**Example**
Diourbel	Road Name	Rue Saint Louis **Diourbel** Point E BP 116 Dakar Senegal
	City	Route De La Gare Face Pharmacie Baol **Diourbel** Senegal

To come up with the solution described in this paper, we consider the problem as a text classification one [6] after a standardization phase has been performed. However, in doing so, we had to face polysemous difficulties, e.g. place names that may refer to different places as illustrated in Table 2. Identifying and resolving polysemous situations is mandatory in order to avoid classification distortion.

The rest of the paper is organized as follows. Section 2 reviews some address parsing and classification work. In Section 3, we detail our solution while, in Section 4 we describe our experimental results. Finally, Section 5 concludes this paper.

2 Related Work

We describe in this section related works on address parsing and classification.

2.1 Address Parsing

In the field of NLP, parsing is considered as a sequence labeling task [3]. As far as parsing models, we looked at Hidden Markov models (HMM) [11] and Conditional Random Field (CRF) based models [12]. HMM does not take into account all possible address patterns, in particular those with low probabilities. CRFs perform better than HMM because they use a conditional probability instead of

the independence assumption made in HMM. However, their performance is affected by the presence of non-standardized addresses and also polysemous words. Recently, deep learning models including Transformers, have been proposed for address parsing. In [13], authors proposed a BERT+CRF approach for parsing Chinese addresses: BERT is applied first for generating address contextual representation, then a CRF model is applied for predicting tags. The evaluation results performed on Chinese addresses show that the F1-score is better than approaches that combine Word2vec, BiLSTM and CRF.

Promising results of BERT applications in sequence labeling, particularly in address parsing [13] motivate us to apply RoBERTa [10] in parsing. Compared to BERT, RoBERTa allows to get rid of the next sentence prediction objective in model's pre-training which improve performance in some downstream tasks.

2.2 Address Classification

Recently, static or contextual word embedding models have been used to perform address classification. Seng *et al.* [4] proposed an approach that classifies addresses to its property type. It consists in applying Long Short-Term Memory Neural Networks (LSTM) on Word2Vec [5] address representations. However, static word embedding, such as Word2vec, cannot handle polysemy. To address this problem, contextual word embedding among which Pre-trained Language Models (PLM), such as BERT [7], have made it possible to strengthen the contextual modeling of texts. In the address classification context, Mangalgi *et al.* [6] propose a RoBERTa-based approach to classify Indian addresses according to the sub-regions to which they belong. A comparison with Word2vec and Bi-LSTM approaches shows that the RoBERTa approach outperforms the other ones in terms of accuracy. Indeed, Word2vec loses the sequential information by averaging the word vectors. RoBERTa better captures the context than Bi-LSTM. However, PLM, rarely consider incorporating structured semantic information which can provide rich semantics for language representation.

For better language understanding, some works have investigated the grounding of PLM with high quality (domain) knowledge, which are difficult to learn from raw texts. Indeed, incorporating external knowledge into PLM has proven effective in various NLP tasks [8, 9].

Our address classification approach draws inspiration from these recent works. It consists on injecting knowledge in the form of address tag embedding into a PLM. These tags result from the address parsing step.

3 Address Validation Approach

In this section, we describe our RoBERTa-based approach for address validation which consists of two main steps: (1) address standardization in order to clean data and to obtain the different address tags and (2) a binary (*valid, invalid*) address classification.

3.1 Address Standardization

Address standardization refers to the transformation of an address into a normalized standard format. It involves two tasks: preprocessing and parsing.

Preprocessing The purpose of this step is to normalize entities and to clean addresses in removing special characters and correcting different spelling errors. For that, we adopt a dictionary-based approach which provides the keywords that may be used to define the address components (city, road, etc.) as well as common abbreviations of these words. In addition, we use a spell checker *pyspellchecker* [3]/ [4] in order to correct address keywords.

Parsing Given an address $A = \{a_1, .., a_n\}$ where a_i is the i-th word and n represents the length of the address, the parsing of A aims to assign a label l to each word a_i of A among the corresponding list of address tags Y; $Y = \{IB, EB, P, Z, HN, RN, D, RS, PB, ZC, C, CO\}$. These tags are defined following the address model depicted in Fig. 1.

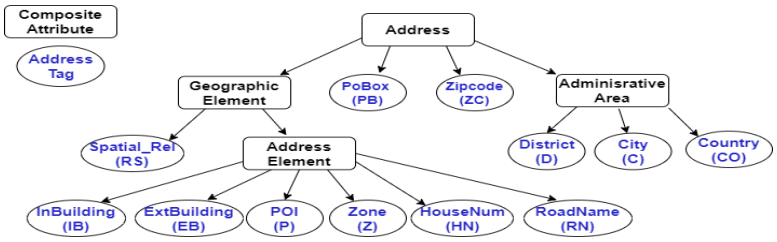

Fig. 1: Address model

We propose a parsing method (Fig. 2) which consists firstly in generating a contextual representation of an address A using pre-training RoBERTa model on a corpus of addresses (section 4.1) by following these two sub-steps:

- RoBERTa calculates the input representations of A by summing over the token, position, and segment embedding. Token embedding for each token is generated using byte-level BPE tokenizer. Position embedding includes the positional information of each token in the address. Segment embedding provides the same label to the tokens that belong to the address.
- 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 noted as H.

Then, the resulting representation is passed to a tagging layer to obtain address tags, using the IOB tagging scheme. A linear layer takes as input the

[3] https://readthedocs.org/projects/pyspellchecker/downloads/pdf/latest/
[4] https://norvig.com/spell-correct.html

last hidden state of the sequence $H = \{h_1, .., h_n\}$ and provides as result the prediction of the tags T.

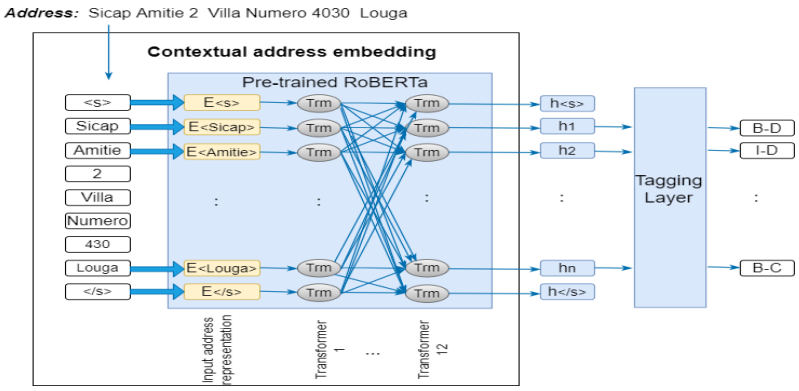

Fig. 2: Address parsing method

3.2 Address Classification

We propose a RoBERTa based classification method (Fig. 3) to classify addresses to *Valid* or *Invalid*. It consists of two steps : 1) generating a fusion of two vector representations which are the contextual vector representation of the address and the vector representation of the address tags, and 2) a classification of addresses according to resulted vectors.

Vectors fusion We use a concatenation function to fuse two embedding vectors as follows:

1. Contextual address embedding: we retrieve the contextual vector representations H of the address A, generated by the pre-trained RoBERTa model, in the address parsing step (see section 3.1).
2. Address tags embedding: the output of the address parsing step is n tags denoted by $T = \{t_1, .., t_n\}$. Since these tags are at the word level, their length is equal to the length n of an address A. We use a look-up table to map these tags to $\{id_1, .., id_n\}$ and feed a linear layer (fully connected layer) in order to obtain the tags embedding, denoted as $W = \{w_1, .., w_n\}$, of A.

Address Classification It is performed using a linear layer (fully connected layer). First, this layer takes as input the embedding fusion vector and generates as output the class logits (probabilities), knowing that the objective function of the training is the CrossEntropy. Then, the Argmax function is applied to these probabilities to obtain the predicted class.

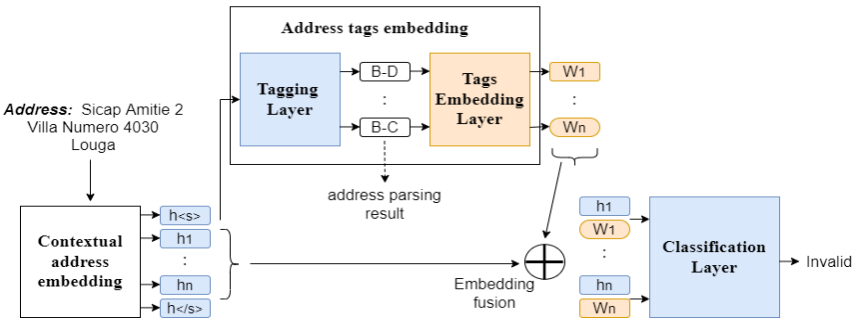

Fig. 3: Address classification method

4 Evaluation

In this section, we describe the experiments carried out in order to evaluate our approach.

4.1 Experimental Setting

Dataset Description Parsing and classification have been performed in using two real-world datasets: (1) a French dataset J_f, which represents 10000 structured addresses extracted from the French Sirene directory [5] and (2) Senegalese dataset J_s, which contains 500 unstructured addresses collected from a Senegalese companies directory [6], characterized by the presence of spatial operators and often the absence of keywords allowing the identification of address elements.

Evaluation setup GeLU activation is used in RoBERTa with the ADAM Optimizer. The dropout and learning rate are set respectively to 0.1 and 3e-5, in such a way 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.

The pretraining of RoBERTa is performed through the Pytorch framework [7]. We generated two pretrained RoBERTa models corresponding to each of the following corpora: (1) French corpora composed of 1,048,575 addresses [8] and (2)

[5] https://www.data.gouv.fr/fr/datasets/base-sirene-des-entreprises-et-de-leurs-etablissements-siren-siret
[6] https://www.goafricaonline.com/
[7] https://pytorch.org/
[8] https://www.data.gouv.fr/fr/datasets/base-sirene-des-entreprises-et-de-leurs-etablissements-siren-siret

Senegalese corpora composed of 31893 addresses collected from Web business directories [9]/[10]/[11]/[12].

4.2 Address Parsing Evaluation

We describe in this section the experiments carried out for the parsing of French J_f and Senegalese J_s addresses. Each dataset is split into a training, validation, and test sets using the ratio of 3:1:1. Moreover, we perform a manual labeling of the datasets.

Baseline Methods We compare our method with two known address parsing sequence models: (1) HMM, implemented with Febrl [13] and (2) CRF, implemented with python-crfsuite library [14].

Results Table 3 illustrates our results in terms of F-measure. First, it is worth noticing that RoBERTa outperforms the two other methods for all datasets. Second, HMM and CRF seem more accurate in case of J_f, the French dataset, given the structured nature of addresses, but less accurate in the case of J_s, the Senegalese dataset. Indeed, J_s addresses contains more polysemous words and are less structured than J_f ones. Finally, we note that RoBERTa better handles polysemous words as illustrated in Table 4 but fails in parsing addresses that lacks for some address elements and/or keywords: those addresses can be characterized as poorly contextualized addresses. Besides, the low frequency of some address elements in the pre-training corpus prevent RoBERTa from an efficient learning context.

Table 3: F-measure of address parsing methods

Method	J_f	J_s
HMM	0.973	0.931
CRF	0.984	0.947
RoBERTa	**0.988**	**0.956**

Table 4: Percentage of polysemous resolution

Method	J_f	J_s
HMM	71.1%	46.6%
CRF	82.1%	68.8%
RoBERTa	**91.2%**	**86.6%**

[9] https://creationdentreprise.sn/
[10] http://pagesjaunesdusenegal.com/
[11] https://www.goafricaonline.com/
[12] https://www.yelu.sn/
[13] http://users.cecs.anu.edu.au/P̃eter.Christen/Febrl/febrl-0.3/febrldoc-0.3/node24.html
[14] https://github.com/scrapinghub/python-crfsuite

4.3 Address Classification Evaluation

We evaluate classification approach on J_f and J_s dataset. We assume that the addresses belonging to these 2 datasets are of class *Valid* because they come from 2 reliable data sources: (1) French government database and (2) an African company directory. To resolve the imbalance class problem in J_f and J_s, we propose a data augmentation technique that allows the generation of synthetic addresses labelled as *Invalid* by applying transformations to collected *Valid* addresses.

Data augmentation: The proposed method is based on address attributes replacement and consists in: (1) Creating two subsets G_f and G_s, from the French and the Senegalese corpora, which represent a hierarchical arrangement of address elements per country and (2) Applying a sequence of attribute replacement, for each dataset J_f and J_s, using G_f and G_s, to create different types of geographic inconsistency (see examples in Table 1).

For each dataset $J \in \{J_f, J_s\}$, we have divided J into two subsets: J_v (70% of addresses) and J'_v (30% of addresses). The generation of *Invalid* addresses is performed on the J'_v dataset. For French addresses, we have injected two types of invalidity related to the most frequent address elements which are: city (CI) and road (RI). For the Senegalese dataset, we have also injected inconsistency for the district (DI) which is more frequent for this dataset than for French addresses. We denote the invalid dataset as J'_{inv}. The classification dataset, denoted as Jc (i.e. J_fc or J_sc), is thus composed by J_v, representing *Valid* addresses, and J'_{inv} representing *Invalid* addresses such as $size(J'_{inv}) = size(J_v)$.

Baseline Models We compare our approach "AllRoBERTa" with the models used in address classification works which are based on (1) static word embedding (Word2vec) plus a SVM classifier and (2) RoBERTa with no knowledge injection. The idea here is to compare the effectiveness of static versus contextual word embedding and to outline the importance of knowledge injection in the proposed approach.

Results Table 5 illustrates the classification results obtained with the different approaches. We notice that whatever the type of invalidity or the country, "AllRoBERTa" is more efficient. We note also that RoBERTa-based models are more efficient than a Word2vec one for both datasets. This can be explained by the highly contextualized representations offered by RoBERTa. Moreover, pretraining RoBERTa on a large corpus of business addresses allows the model to learn several geographical facts related to the context of each address element. Classification results show that our "AllRoBERTa" is a promising solution which can be useful, mainly when geographic databases are missing in some countries such as Senegal.

We evaluated the percentage of polysemy in the misclassified Senegalese addresses. As illustrated in table 6, for all the tested approaches, more than 50% of the misclassified addresses are polysemous. This ratio can even reach 72.5%

Table 5: F-measure of different address classification approaches

Approach	J_fc		J_sc		
	CI	RI	CI	DI	RI
Without parsing					
Word2vec+ SVM	0.911	0.862	0.9	0.869	0.848
RoBERTa	0.949	0.928	0.942	0.919	0.912
With parsing					
AllRoBERTa	**0.981**	**0.957**	**0.971**	**0.948**	**0.938**

Table 6: Impact of polysemy in Senegalese addresses classification

Approach	Polysemy percentage in missclassified addresses
Word2vec + SVM	72.5%
RoBERTa	69.2%
AllRoBERTa	51.2%

Table 7: Impact of a "perfect" parsing on addresses classification

Approach	J_sc		
	CI	DI	RI
AllRoBERTa	0.971	0.948	0.938
Parsing "Ground Truth" + RoBERTa	**0.985**	**0.965**	**0.958**

in the case of a classification based on "Word2vec + SVM". For AllRoBERTa, classification errors come from cases of unresolved polysemous situations during parsing, with a percentage greater than 83%. We conclude that polysemous elements badly impacts the address classification process.

Finally, we analyzed the impact of the introduction of address tags in the classification. To this end, we first performed manual parsing (J_s) in order to perfectly identify addresses tags, then we carried out the classification (J_sc) with RoBERTa. The obtained results compared to AllRoBERTa (table 7) show that the quality of parsing has an impact on the classification results.

5 Conclusion

In this paper, we described an address validation approach based on RoBERTa, a pre-trained transformer-based language model. Usage of RoBERTa is motivated by its ability to manage polysemy. We inject semantic address tags into the pre-trained RoBERTa model in order to improve semantic understanding of domain-specific data. Experimental evaluations, carried out on two real-world datasets involving French and Senegalese addresses, show the effectiveness of our solution. In the future, we intend to extend this work in at two directions: (1) explore an active learning method to minimize the efforts of manually labelling data sets and (2) make the approach usable through an address validation API.

References

1. D. K. Matci and U. Avdan.: Address standardization using the natural language process for improving geocoding results. Computers, environment and urban systems, vol. 70, pp. 1–8 (2018)
2. X.-f. Xi, L. Wang, E. Zou, C. Zeng, and B. Fu.: Joint learning for non-standard chinese building address standardization. In 2018 IEEE International Smart Cities Conference (ISC2), pp. 1–8. IEEE (2018)
3. N. Abid, A. ul Hasan, and F. Shafait.: Deepparse: A trainable postal address parser. In 2018 Digital Image Computing: Techniques and Applications (DICTA), pp. 1–8. IEEE (2018)
4. L. Seng.: A Two-Stage Text-based Approach to Postal Delivery Address Classification using Long Short-Term Memory Neural Networks. (2019)
5. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean.: Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (NIPS), pp. 3111–3119 (2013)
6. S. Mangalgi, L. Kumar, and R. B. Tallamraju.: Deep contextual embeddings for address classification in e-commerce. arXiv preprint arXiv:2007.03020 (2020)
7. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
8. Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, and Q. Liu.: Ernie: Enhanced language representation with informative entities. In ACL, pp. 1441–1451 (2019)
9. Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, and X. Zhou.: Semantics-aware bert for language understanding. In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, pp. 9628–9635 (2020)
10. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
11. P. Christen and D. Belacic.: Automated probabilistic address standardisation and verification. In Australasian Data Mining Conference (AusDM'05), pages 53–67, Sydney (2005)
12. M. Wang, V. Haberland, A. Yeo, A. Martin, J. Howroyd, and J. M. Bishop.: A probabilistic address parser using conditional random fields and stochastic regular grammar. In 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), pp. 225–232, IEEE (2016)
13. H. Zhang, F. Ren, H. Li, R. Yang, S. Zhang, and Q. Du.: Recognition method of new address elements in chinese address matching based on deep learning. ISPRS International Journal of Geo-Information, vol. 9, no. 12, p. 745 (2020)