UNIVERSIDAD
DE GRANADA

FACULTAD DE FILOSOFÍA Y LETRAS
DEPARTAMENTO DE FILOLOGÍAS INGLESA Y ALEMANA
PROGRAMA DE DOCTORADO EN LENGUAS, TEXTOS Y CONTEXTOS

DOCTORAL THESIS

A LINGUISTICALLY-AWARE COMPUTATIONAL APPROACH TO MICROTEXT LOCATION DETECTION

Nicolás José Fernández Martínez

Supervised by
Dr. Ángel Miguel Felices Lago (University of Granada)
Dr. Carlos Periñán Pascual (Universitat Politècnica de València)

Granada, 2020

Editor: Universidad de Granada. Tesis Doctorales
Autor: Nicolás José Fernández Martínez
ISBN: 978-84-1306-680-6
URI: http://hdl.handle.net/10481/64577

ACKNOWLEDGMENTS

Although the present thesis is the result of my own effort and dedication, I would like to express my sincerest and deepest gratitude for the time, support, help, and patience of the following people, without whom I would not have been able to undertake this path:

- My supervisor Dr. Ángel Miguel Felices Lago, for trusting me and my abilities from the very beginning, and providing me with the opportunity to work in the fields of Natural Language Processing and Computational Linguistics. He has acted as a godfather, offering me unflagging support in each one of the steps taken in the present thesis, immediate availability whenever I needed help of any sort, and deep affection, professionalism and encouraging words, which have definitely led me to where I am today.
- My co-supervisor Dr. Carlos Periñán Pascual, who has also trusted me and my abilities, providing me with extensive knowledge in the language-related computational fields that I have been working on, and inestimable, immediate and unconditional support and help in the technical aspects of the present thesis. He has shown me that linguists can play a prominent role in today's digital and technological world. He has taught me everything that has helped develop the computational aspects of the thesis, and shape the researcher that I am today. I am deeply grateful about his wise comments, time, and relentless dedication.
- My whole family, for their unconditional love, encouragement and support, and for believing in me, and helping shape the person I am today.
- My friends and colleagues, who have always believed in me and been always there in difficult times and whenever I needed help of any sort.
- The lecturers that I have met at the University of Jaén and the University of Granada, who have greatly contributed to instill their linguistic knowledge, their passion for linguistics, and their humanitarian values in me, with special mention to Dr. Alfonso Rizo Rodríguez from the University of Jaén and Dr. Pamela Faber Benítez from the University of Granada.

TABLE OF CONTENTS

List of Tables ..... **vii**
List of Figures ..... **x**
List of Acronyms and Abbreviations ..... **xii**
Abstract ..... **xiv**
Resumen ..... **xv**

1. INTRODUCTION **1**

2. BACKGROUND **5**
2.1. Computational Linguistics and Natural Language Processing: definition, uses, and role of linguists 6
2.2. Location detection and Geographic Information Retrieval 8
2.3. Practical applications of tweet-based geolocation systems in social sensing settings 9
2.4. Social sensors and emergency situation awareness 10
2.5. The representation of spatial knowledge in natural languages: a linguistic-based approach 11
2.5.1. The structural, syntactic, conceptual and pragmatic features of spatial expressions 11
2.5.2. Named entities of places: toponyms and geographical names 13
2.6. Microtext genres: the tweet subgenre 15
2.7. Techniques and main frameworks used in location detection 16
2.7.1. Feature-based NER 17
2.7.2. Neural NER 18
2.7.2.1. Neuronal networks: layers, neurons and hyperparameters 19
2.7.3. Rule-based NER 23
2.7.4. Named Entity Matching 24
2.8. A typology of Twitter-based geolocation models 25
2.8.1. Classification of Twitter-based geolocation systems in terms of target 26
2.8.1.1. User location 26
2.8.1.2. Tweet location 26
2.8.1.3. Locative reference extraction 27
2.8.2. Classification of Twitter-based geolocation systems in terms of method 34
2.8.3. Classification of Twitter-based geolocation systems in terms of data and resources 35

3. RESEARCH CHALLENGE 35
3.1. Research issues and limitations 35
3.2. Research questions 36
3.3. Research hypothesis and justification 37

4. OBJECTIVES 37

5. MODEL AND METHODOLOGY 38
5.1. Formal, semantic and structural boundaries of locative references 39
5.2. Corpus compilation phase 43
5.3. LOcative Reference Extractor (LORE) 52
5.3.1. Development phase 52
5.3.1.1. A typology of linguistic regex-based rules 54
5.3.1.1.1. Rules for n-gram combinations of locative references using a geodatabase 54
5.3.1.1.2. Rules that exploit locative prepositions 55
5.3.1.1.3. Rules that exploit location-indicative nouns 57
5.3.1.1.4. Rules that exploit locative markers 62
5.3.1.1.5. Safe-checking rules 64
5.3.2. Language-specific lexical datasets 65
5.3.2.1. POS-tag and locative-preposition dataset 65
5.3.2.2. Place-name dataset 66
5.3.2.3. Location-indicative noun dataset 66
5.3.2.4. Locative-marker dataset 68
5.3.2.5. Stopword dataset 69
5.3.3. The pipeline of LORE 70
5.3.3.1. Pre-processing 72
5.3.3.2. Tokenization and POS tagging 72
5.3.3.3. Place-name search 74
5.3.3.4. Linguistic processing 75
5.3.3.4.1. For locative references introduced by locative prepositions 75
5.3.3.4.2. For locative references introduced by location-indicative nouns 75
5.3.3.4.3. For locative markers 76
5.4. Neuronal LORE (nLORE) 76
5.4.1. Deep Learning: RNN, CRF, and word embeddings 77
5.4.1.1. Bidirectional RNN with LSTM and CRF on top 77
5.4.1.2. CRF: output layer structure 80
5.4.1.3. Vector semantics: dense, static word embeddings 81

5.4.2. Training phase 84
5.4.3. Linguistic-based feature engineering 85
5.4.3.1. Template features 86
5.4.3.2. Context template features 87
5.4.3.3. Word embedding features 88
5.4.4. Hyperparameterization: parameter tuning and settings 89

6. IMPLEMENTATION 89
6.1. LORE 90
6.1.1. Internal files in LORE 92
6.1.1.1. The configuration file 93
6.1.1.2. The place-name dataset files 94
6.1.1.3. The location-indicative noun dataset files 95
6.1.1.4. Place abbreviation list and locative marker dataset files 96
6.1.1.5. The stopword dataset and frequency dictionary files 96
6.2. nLORE 96
6.2.1. Internal files in nLORE 98
6.2.1.1. Neuronal network configuration file 98
6.2.1.2. Tags file 99
6.2.1.3. Template features file 99
6.2.1.4. Word embeddings file 100
6.3. Evaluation tool 101

7. EVALUATION 102
7.1. Results 103
7.1.1. Experiment I 103
7.1.1.1. Multilingual LORE 103
7.1.1.2. LORE vs off-the-shelf NER tools 106
7.1.2. Experiment II 110
7.1.2.1. nLORE 111
7.1.2.2. LORE with the English eval corpus II 113
7.1.2.3. LORE vs nLORE 113
7.2. Discussion 115
7.2.1. Experiment I 115
7.2.1.1. Error analysis 119
7.2.1.1.1. Errors of omission 119
7.2.1.1.2. Errors of commission 123

7.2.2. Experiment II 125
7.2.2.1. nLORE 125
7.2.2.2. LORE with the English eval corpus II 131
7.2.2.3. LORE vs nLORE 133
7.2.3. Issues, limitations, and areas of improvement 135

8. CONCLUSIONS 137

9. BIBLIOGRAPHY 139

APPENDIX 156
Flowchart 1. Regex-based rules for n-gram combinations of locative references using a geodatabase 156
Flowchart 2. Regex-based rules that exploit locative prepositions 156
Flowchart 3. Regex-based rules that exploit location-indicative nouns for the English language 157
Flowchart 4. Regex-based rules that exploit location-indicative nouns for the Spanish and French languages 157
Flowchart 5. Regex-based rules that exploit highways and road-naming conventions 158
Flowchart 6. Regex-based rules that exploit locative markers 158
Flowchart 7. The pipeline of LORE 159

LIST OF TABLES

Table 1. Corpus types and their usages 45
Table 2. Corpora's features 45
Table 3. Sample of the English dev corpus 47
Table 4. Sample of the English eval corpus II 47
Table 5. Sample of the gold standard of the English dev corpus 48
Table 6. Sample of the gold standard of the Spanish dev corpus 48
Table 7. Distribution of locative references in terms of n-gram size in the corpora 49
Table 8. Corpora's statistics 49
Table 9. Most frequent locative references in the English dev corpus 50
Table 10. Most frequent locative references in the English eval corpus I 50
Table 11. Most frequent locative references in the English train corpus 50
Table 12. Most frequent locative references in the English valid corpus 51
Table 13. Most frequent locative references in the English eval corpus II 51
Table 14. Most frequent locative references in the Spanish dev corpus 51
Table 15. Most frequent locative references in the Spanish eval corpus 52
Table 16. Most frequent locative references in the French eval corpus 52
Table 17. Language-specific resources in LORE 53
Table 18. Locative prepositions in LORE 65
Table 19. Sample of the English location-indicative noun dataset 67
Table 20. Sample of the Spanish location-indicative noun dataset 67
Table 21. Sample of the French location-indicative noun dataset 68
Table 22. A sample of the English locative-marker dataset 68
Table 23. A sample of the Spanish locative-marker dataset 68
Table 24. A sample of the French locative-marker dataset 69
Table 25. English stopword dataset 69
Table 26. Spanish stopword dataset 70
Table 27. French stopword dataset 70
Table 28. The matrix representation of token objects 73
Table 29. The matrix representation of a list of tweet objects 74
Table 30. Co-occurrence matrix in sparse vectorial representations 81
Table 31. Trained nLORE models 85
Table 32. Template features for the extended nLORE model 86
Table 33. Template features for the basic nLORE model 87
Table 34. Features generated for the example in the extended nLORE model 87
Table 35. Word embeddings settings 88

Table 36. Semantic similarity of *city* with respect to other words in the word embeddings 88
Table 37. Parameter settings in trained nLORE models 89
Table 38. Input and output files in LORE 91
Table 39. Internal files required by LORE, their function and data type or structure 92
Table 40. English config file 93
Table 41. Spanish config file 93
Table 42. French config file 94
Table 43. Input and output files in nLORE 97
Table 44. Internal files required by nLORE, their function, and context of usage 98
Table 45. Settings, functions and possible options in the neuronal network configuration file .. 99
Table 46. Word embedding settings in Txt2Vec, functions, and parameters 100
Table 47. Evaluation with the English eval corpus I 103
Table 48. Evaluation with the Spanish eval corpus 103
Table 49. Evaluation with the French eval corpus 104
Table 50. Evaluation with the English eval corpus I in terms of n-gram size for the place-name search + linguistic processing modules 104
Table 51. Evaluation with the Spanish eval corpus in terms of n-gram size for the place-name search + linguistic processing modules 105
Table 52. Evaluation with the French eval corpus in terms of n-gram size for the place-name search + linguistic processing modules 105
Table 53. Processing speed for the English eval corpus I 108
Table 54. Processing speed for the Spanish eval corpus 108
Table 55. Processing speed for the French eval corpus 108
Table 56. Evaluation metrics for each English location-detection model 109
Table 57. Evaluation metrics for each Spanish location-detection model 109
Table 58. Evaluation metrics for each French location-detection model 109
Table 59. Processing speed for each of the nLORE models 112
Table 60. Evaluation of nLORE with the English eval corpus II 113
Table 61. Evaluation of the extended 7k nLORE model with the English eval corpus II in terms of n-gram size 113
Table 62. Evaluation of LORE with the English eval corpus II 113
Table 63. Evaluation of LORE with the English eval corpus II in terms of n-gram size 113
Table 64. Evaluation of LORE vs nLORE with the English eval corpus II 114
Table 65. Processing speed for each of the LORE and nLORE models 114
Table 66. Example of the basic 7k nLORE model 126
Table 67. Example of the extended 7k nLORE model 127
Table 68. Example of the basic 7k nLORE model 127

Table 69. Example of the extended 7k nLORE model 127
Table 70. Example of the basic 7k nLORE model 128
Table 71. Example of the extended 7k nLORE model 128
Table 72. Example of the basic 7k nLORE model 129
Table 73. Example of the extended 7k nLORE model 129
Table 74. Example of the basic 7k nLORE model 129
Table 75. Example of the extended 7k nLORE model 129
Table 76. Example of the basic 7k nLORE model 130
Table 77. Example of the extended 7k nLORE model 130
Table 78. Example of the basic 7k nLORE model 130
Table 79. Example of the extended 7k nLORE model 130
Table 80. Example of the basic 7k nLORE model 131
Table 81. Example of the extended 7k nLORE model 131
Table 82. Example of LORE 133
Table 83. Example of the extended 7k nLORE model 133
Table 84. Example of LORE 133
Table 85. Example of the extended 7k nLORE model 134
Table 86. Example of LORE 134
Table 87. Example of the extended 7k nLORE model 134
Table 88. Example of LORE 135
Table 89. Example of the extended 7k nLORE model 135

LIST OF FIGURES

Figure 1. Tweet about an accident. 15
Figure 2. Tabular-based representation of tokens in NER 17
Figure 3. Layers and neurons in a simple neuronal network 19
Figure 4. The computation of the activation function in a neuron 20
Figure 5. Forward propagation and backward propagation (Negrov et al., 2015) 21
Figure 6. The interaction of the loss function, accuracy and iterations. 22
Figure 7. Overfitting 23
Figure 8. The phrasal structure of locative references 39
Figure 9. The FireAnt app for tweet collection 43
Figure 10. N-gram-based matching 74
Figure 11. A graphical representation of a simple RNN 77
Figure 12. A bidirectional RNN (Ogawa & Hori, 2015) 78
Figure 13. A bidirectional RNN structure with LSTM as hidden layers 80
Figure 14. A graphical representation of a CRF 80
Figure 15. Skip-gram algorithm 81
Figure 16. Continuous bag of words algorithm 82
Figure 17. A reduced three-dimensional space for word embeddings 83
Figure 18. A glimpse of the coding of LORE 90
Figure 19. UI of LORE 90
Figure 20. Our tweet corpus pre-processor command-line tool 91
Figure 21. A screenshot of the English place-names file retrieved from the GeoNames database 94
Figure 22. A screenshot of the EuroWordNet hyponym extractor 95
Figure 23. UI of nLORE 97
Figure 24. UI of LORE with the implemented NER tools 101
Figure 25. A screenshot of our evaluation tool 101
Figure 26. F1 of English LORE in terms of n-gram size 105
Figure 27. F1 of Spanish LORE in terms of n-gram size 105
Figure 28. F1 of French LORE in terms of n-gram size 106
Figure 29. Bar chart for the evaluation metrics for each English location-detection model 109
Figure 30. Bar chart for the evaluation metrics for each Spanish location-detection model 110
Figure 31. Bar chart for the evaluation metrics for each French location-detection model 110
Figure 32. Precision of nLORE 111
Figure 33. Recall of nLORE 111
Figure 34. F1 of nLORE 112
Figure 35. Precision of LORE vs nLORE 114

Figure 36. Recall of LORE vs nLORE.....115
Figure 37. F1 of LORE vs nLORE 115

LIST OF ACRONYMS AND ABBREVIATIONS

AI: Artificial Intelligence

ANN: Artificial Neuronal Network

AP: adverbial phrase

API: Application Program Interface

BERT: Bidirectional Encoder Representations from Transformers

biLSTM: bidirectional Long Short-Term Memory

BMESO: Beginning-Medial-End-Single-Outside

CASPER: CAtegory and Sentiment-based Problem FindER

CL: Computational Linguistics

CNN: Convolutional Neuronal Network

COCA: Corpus of Contemporary American English

CRF: Conditional Random Field

dev corpus: development corpus

DL: Deep Learning

eval corpus: evaluation corpus

FN: false negative

FP: false positive

GCNL: Google Cloud Natural Language

GCS: Geographic Coordinate System

GeoAI: Geospatial Artificial Intelligence

GIR: Geographic Information Retrieval

GIS: Geographic Information Science

GPE: geopolitical entity

HLT: Human Language Technologies

IOB: Inside-Outside-Beginning

LE: Language Engineering

LORE: LOcative Reference Extraction

ML: Machine Learning

NEM: Named Entity Matching

NER: Named Entity Recognition

nLORE: neuronal LOcative Reference Extraction

NLP: Natural Language Processing

NLTK: Natural Language Toolkit

NLU: Natural Language Understanding

NP: noun phrase

P: precision

POI: point of interest

POS: part of speech

PP: prepositional phrase

R: recall

regex: regular expression

RNN: Recurrent Neuronal Network

TP: true positive

train corpus: training corpus

UI: User Interface

valid corpus: validation corpus

VP: verb phrase

ABSTRACT

Extracting geospatially rich knowledge from microtexts such as tweets is of utmost importance for location-based systems in emergency services to raise situational awareness about a given emergency (i.e. natural or man-made disasters), such as earthquakes, floods, pandemics, car accidents, terrorist attacks, shooting attacks, etc. (Vieweg et al., 2010; Crooks et al., 2013; Imran et al., 2014; Jongman et al., 2015; Martínez-Rojas et al., 2018; C. Zhang et al., 2019; Siriaraya et al., 2019). In other words, emergency responders and competent authorities need to understand where the incident happened, where people are in need of help, and/or which areas were affected, with the aim of coordinating effective and immediate aid and allocating resources in the affected areas and/or to the affected persons. Such systems could potentially help save lives and/or prevent further damage to environmental or urban areas in emergency- and crisis-related contexts.

The problem is that the wide majority of tweets are not geotagged (Middleton et al., 2014), so we need to resort to the messages in the search of geospatial evidence (Wallgrün et al., 2018). In this context, we present LORE, a multilingual, rule-based location-detection system for English, Spanish, and French tweets that leverages lexical datasets of place names and location-indicative words together with linguistic knowledge through Natural Language Processing and computational techniques. We also present nLORE, a Deep Learning model that feeds off the linguistic knowledge provided by LORE. One of the main contributions of our models is to capture fine-grained complex locative references, ranging from geopolitical entities (e.g. towns, cities, regions, countries, etc.) and natural landforms (e.g. mountains, rivers, lakes, hills, valleys, etc.) to points of interest (e.g. squares, cathedrals, universities, residences, restaurants, museums, etc.) and traffic ways (e.g. streets, avenues, roads, highways, etc.). LORE outperforms well-known, general-purpose, off-the-shelf entity-recognizer systems typically used in benchmarking (Schmitt et al., 2019): Stanford NER, spaCy, NLTK, OpenNLP, Google Natural Language Cloud, and Stanza. LORE achieves an unprecedented trade-off between precision and recall, while showing similar performance when applied to other corpora. nLORE outperforms LORE by a slight margin, and confirms the usefulness of linguistic-based feature engineering in Artificial Intelligence (Linzen, 2019). Therefore, our models provide not only a quantitative advantage over other well-known entity-recognizer systems in terms of performance and accuracy but also a qualitative advantage in terms of the diversity and semantic granularity of the locative references extracted from the tweets.

Keywords: location detection, location extraction, geolocation, named-entity recognition, natural language processing, deep learning, emergencies, disasters

RESUMEN

La extracción de información geoespacial rica de microtextos como los tweets es sumamente importante para sistemas geolocalizadores en servicios de emergencias para contribuir a la conciencia situacional sobre una emergencia como desastres naturales o producidos por el hombre, ya sean terremotos, inundaciones, pandemias, accidentes de tráfico, ataques terroristas, tiroteos, etc. (Vieweg et al., 2010; Crooks et al., 2013; Imran et al., 2014; Jongman et al., 2015; Martínez-Rojas et al., 2018; C. Zhang et al., 2019; Siriaraya et al., 2019). Dicho de otra manera, los servicios de emergencias y autoridades competentes necesitan comprender dónde ha ocurrido el incidente, dónde necesita la gente ayuda y/o qué lugares han sido afectados con el objetivo de proporcionar asistencia inmediata y destinar recursos en aquellas áreas o a aquellas personas afectadas. Estos sistemas podrían servir para salvar vidas y prevenir futuros daños a zonas urbanas o áreas medioambientales en contextos de crisis o emergencias.

El problema reside en la escasez de tweets geoetiquetados (Middleton et al., 2014); por tanto, ha de recurrirse a los mensajes de texto en búsqueda de esa evidencia geoespacial (Wallgrün et al., 2018). En este contexto, presentamos LORE, un sistema multilingüístico de detección de localizaciones en tweets en inglés, español y francés basado en reglas que integra recursos léxicos de nombres de lugar y de palabras que indican localización junto con conocimiento lingüístico proporcionado por diversas técnicas computacionales de Procesamiento de Lenguaje Natural. También introducimos nLORE, un modelo basado en *Deep Learning* que se nutre del conocimiento lingüístico proporcionado por LORE. Una de las contribuciones más notables de nuestros modelos tiene que ver con la granularidad semántica de los tipos de localizaciones extraídas, desde entidades geopolíticas (e.g. pueblos, ciudades, regiones, países, etc.) y accidentes geográficos (e.g. montañas, ríos, lagos, colinas, valles, etc.) hasta puntos de interés (e.g. plazas, catedrales, universidades, residencias, restaurantes, museos, etc.) y vías de tráfico (e.g. calles, avenidas, carreteras, autovías, etc.). LORE supera a sistemas conocidos de dominio general de reconocimiento de entidades nombradas que se utilizan con frecuencia en sistemas de evaluación (Schmitt et al., 2019) como Stanford NER, spaCy, NLTK, OpenNLP, Google Natural Language Cloud y Stanza, alcanzando unas puntuaciones récord de evaluación en términos de precisión y cobertura, a la vez que muestra un rendimiento similar cuando se aplica a otros corpora. nLORE llega a superar LORE por un margen estrecho y confirma la utilidad de la implementación de características lingüísticas en la Inteligencia Artificial (Linzen, 2019). En este sentido, nuestros modelos proporcionan, no solo un salto cuantitativo respecto a la competencia en términos de rendimiento y precisión, sino también un salto cualitativo dada la diversidad y granularidad semántica de las referencias locativas que se pueden extraer de los tweets.

Palabras clave: detección de localizaciones, extracción de localizaciones, geolocalización, reconocimiento de entidades nombradas, procesamiento del lenguaje natural, deep learning, emergencias, desastres

1. INTRODUCTION

The microtext genre permeates the Internet, especially in the form of micro-blogging and social media services. Twitter, in particular, is one of the most widely used and popular micro-blogging sites, and the tweet subgenre, a prototypical example of microtext, is one of the most investigated in event detection, sentiment analysis and/or tweet geolocation, among many other Natural Language Processing (NLP) and Artificial Intelligence (AI) research areas (Murthy, 2018; Stock, 2018). The large volume of user-generated content on Twitter can be exploited in social sensing settings where disaster and crisis management and tracking become of utmost importance for disaster and crisis relief operations (Vieweg et al., 2010; Crooks et al., 2013; Imran et al., 2014). In this respect, Twitter can function as a real-time social sensor system that provides multidirectional channels of communication in emergency and crisis events between the affected persons and disaster management agencies (Aggarwal, 2013; Martínez-Rojas et al., 2018; C. Zhang et al., 2019). Having a rapid understanding about these events can help handle human and economic resources effectively through immediate and timely decisions and actions taken by aid organizations and competent authorities. Emergency responders can then coordinate effective aid and help allocate resources in the affected areas and/or to the affected persons. Obtaining geographic information from tweets proves to be a difficult task, considering that geotagged metadata (i.e. coordinates) attached to tweets represent around 1% of tweets only (Middleton et al., 2014), which hinders any further geographical-based application. Moreover, Twitter has restricted sharing precise geotagged metadata in June 2019. Given the low volume of geotagged tweets, its recent sharing restrictions, and thus the sparse geographic metadata, it becomes necessary to turn to other geospatial evidence, such as that found in tweet text materialized by the presence of locative references. In fact, locative references in tweet text are usually much more frequent than geotagged data (Wallgrün et al., 2018), and therefore can be a very valuable piece of information for emergency responders and other competent authorities in the absence of other geospatial cues. Twitter has in fact been exploited in many geolocation systems that handle real-life scenarios, ranging from natural or human-made disaster detection and tracking in floods, earthquakes, storms, civil unrest, war, crime, etc. (Vieweg et al., 2010; Crooks et al., 2013; Imran et al., 2014; Jongman et al., 2015; Martínez-Rojas et al., 2018; C. Zhang et al., 2019; Siriaraya et al., 2019), health surveillance and disease tracking (Eke, 2011; Dredze et al., 2013), e.g. the current COVID-19 pandemic (Singh et al., 2020), to marketing and advertising purposes (Mourad et al., 2019), or traffic incident detection, road traffic control and/or traffic congestion (Ahmed et al., 2019; Das & Purves, 2019; Gonzalez-Paule et al., 2019; Khodabandeh-Shahraki et al., 2019). All these studies highlight that the extraction of fine-grained geospatial information from Twitter is a key component in intelligent systems for crisis management services. In this sense, the location dimension proves to be critical for raising

situation awareness of crisis-related events and understanding their impact; in other words, understanding where the incident happened, where people are in need of help, and/or which areas were affected. Such systems could, as a last resort, potentially help save lives and/or prevent further damage to environmental or urban areas in emergency- and crisis-related contexts.

Given the importance of geolocation systems in real-life scenarios and the sparsity of geospatial data, the present thesis aims to exploit tweet text in search of geospatial evidence in the form of locative references. The thesis is split into two main parts. The first part focuses on LORE (LOcative Reference Extractor), a multilingual, linguistically-aware, fine-grained location-detection model for tweets that leverages linguistic knowledge through NLP techniques such as tokenization, part-of-speech (POS) tagging, n-gram detection, regular expressions (regex) for linguistic-based rules and patterns (Jurafsky & Martin, 2018b), location-indicative noun datasets retrieved from EuroWordNet (Miller, 1995; Fellbaum, 1998), a place-name dataset obtained from the geographic database GeoNames (Ahlers, 2013), and other lexical datasets for locative markers or place abbreviations. LORE can capture any type of simple or complex locative reference found in tweets written in English, Spanish, or French: geopolitical entities (e.g. towns, cities, provinces, states, regions, countries, neighborhoods, districts, etc.), natural landforms (e.g. lakes, rivers, mountains, parks, ridges, valleys, beaches, shores, seas, etc.), points of interest (POIs) (e.g. schools, churches, cinemas, casinos, bus stations, airports, gardens, taverns, museums, commercial centers, police stations, etc.), and traffic ways (e.g. street, st, avenue, av, boulevard, blvd, turnpike, tpike, tpk, highway, hwy, freeway, fwy, route X, I-X, M-X, etc. where X represents a given number). Our model can target any kind of crisis-, environment- or disaster-related event, local or global, from a given corpus of tweets. To the best of our knowledge, our rule-based model is the first that implements a multilingual system leveraging language-specific, lexically-rich datasets of place names together with location-indicative nouns from EuroWordNet and semi-automatic methods for the language-specific inventories of lexico-syntactic rules for a fine-grained location-detection system. This contrasts with traditional and current geolocation models that focus on coarse-grained location types such as geopolitical entities and natural forms, leaving aside geospatially-rich information such as traffic ways and POIs (Wang & Hu, 2019). Moreover, most research on tweet location detection does not propose linguistically-rich, rule-based methods such as ours (Stock, 2018). The present research originates with the purpose of implementing this microtext geo-extraction application into CASPER (CAtegory and Sentiment-based Problem FindER) (Periñán-Pascual & Arcas-Túnez, 2017, 2018, 2019), a multi-domain problem detection system that deals with environment-related issues in tweets. Besides English, LORE provides support for other languages such as Spanish and French by means of semi-automatic methods, making it ideal in multilingual contexts.

The second part focuses on neuronal LORE (nLORE), a proof-of-concept Deep Learning (DL) computational model for English tweets. By means of an implemented bi-directional Recurrent Neural Network algorithm, nLORE automatically learns the linguistic features provided by LORE in the training phase to extract locative references in the evaluation corpus with even greater accuracy than LORE, thus outperforming LORE and achieving state-of-the-art performance in the task of locative reference extraction.

We provide a definition of a locative reference, setting its formal, structural, and semantic boundaries, and establish a typology of locative references; then, we describe the building process of the corpora and the creation of a gold standard of locative references for each of the languages and models, all of which became the cornerstone that underlie the development and training stages of LORE and nLORE. Accordingly, the evaluation stage was split into two experiments. In the first experiment, we developed and assessed the performance of LORE with development and evaluation corpora for English, Spanish, and French. We also compared the performance of our model in terms of processing speed and the evaluation metrics precision, recall, and F1 scores obtained by our model against well-known, open-source, state-of-the-art Named Entity Recognition (NER) tools. In the second experiment, we used training, validation, and evaluation corpora of only English tweets to train and assess our probabilistic-based model nLORE. The aims of the second experiment were trifold: i) we wished to elucidate whether nLORE could outperform LORE, ii) whether enriching the probabilistic-based model with linguistic features could improve the performance of nLORE, and ii) whether we could overcome the conundrum of finding and labeling a large training dataset by using these built-in linguistic features, thus alleviating the computational cost, time and resources typical of probabilistic-based approaches. Other objective was to check whether the performance of LORE would remain similar with other evaluation corpora, since in the NLP community concerns are constantly being voiced about the necessity to train models that can be generalized and applied with the same success to new, unseen collections of data.

The corpus compilation process was carried out using text-based data retrieval techniques (Hu, 2018a) that considered the prototypical emergency and crisis-related keywords ‘earthquake’, ‘flood’, ‘flooding’, ‘car accident’, ‘bombing attack’, ‘shooting attack’, ‘terrorist attack’, and ‘incident’, and their near-equivalents in Spanish and French to mine tweets of issues of different nature. In the first experiment, for the development and evaluation stages of LORE for English tweets, we compiled a development corpus of 500 tweets and an evaluation corpus of 800 tweets. In the case of Spanish, we built a development corpus of 100 tweets and an evaluation corpus of 500 tweets containing the keywords ‘terremoto’, ‘inundaciones’, ‘accidente de coche’, ‘ataque terrorista’, ‘bombardeo’, ‘tiroteo’, and ‘incidente’. For French tweets, we only compiled an evaluation corpus of 391 tweets using keywords such as ‘séisme’, ‘tremblement de terre’, ‘inondations’, ‘coups de feu’, ‘attentat terroriste’, ‘attentat à la bombe’,

‘fusillade’, ‘accident de voiture’, ‘accident de la route’, ‘incident’, since we departed from the assumption that the rules, which were already developed for Spanish, could be automatically extended to French due to their similarities as Romance languages. Afterwards, we explain the building process of the lexical resources used in LORE, together with a typology of the linguistic-based rules which underlie LORE. After that, we offer a thorough explanation of the modular architecture of LORE. All this is followed by a technical explanation for the training of our probabilistic-model nLORE with regards to the implemented neuronal networks and word embeddings. In the first experiment, for the English, Spanish and French evaluation corpora, LORE achieved an F1 score of 0.81, 0.68, and 0.62, respectively. For all languages, those scores would be considered state-of-the-art performance. We also show that, under the same conditions, the evaluation measures of our model outperformed other well-known, open-source named-entity recognizers such as Stanford NER, spaCy, NLTK, Google Cloud Natural Language, OpenNLP, and Stanza, which rely on probabilistic-based algorithms (Schmitt et al., 2019). Only in the case of English was the processing speed of LORE slightly superior to the others. In the second experiment, we show that nLORE, leveraging the linguistic features provided by the rule-based model, achieves greater performance than LORE. Overall, the present results suggest that our rule-based approach achieves, in comparison with other NER tools, the highest scores in location extraction without the high computational cost, time and resources characteristic of probabilistic-based approaches grounded on Machine Learning (ML) or DL algorithms. This involves a quantitative advantage in terms of the performance achieved and a qualitative advantage in terms of the diversity, variety and semantic granularity of the location types. At the same time, nLORE, exploiting the linguistic power of LORE and trained on a relatively small corpus of tweets, achieves even greater results, suggesting that linguistic-based feature engineering in such probabilistic-based approaches may still provide a much-valued added benefit –though slight–, which could pave the way for more linguistic-oriented computational work in the field of NER. This research goes in line with recent calls in the linguistic and computational communities, requesting a greater interaction between linguistics and AI (Linzen, 2019).

The present thesis is structured as follows. Section 2 introduces the theoretical background that underlies the development of LORE and nLORE. In Section 2.1., we explain what Computational Linguistics and Natural Language Processing are about, the undervalued yet prominent role of linguists in practical applications derived from those fields, and how the present thesis attempts to lay the basis for future work of linguists in these linguistic-related computational disciplines. Section 2.2. introduces the task of location detection and how it works in relation to the field of Geographic Information Retrieval in combination with Natural Language Processing and Computational Linguistics. Section 2.3. introduces the practical applications of tweet-based geolocation systems in social-sensing settings, where extracting

geospatial evidence from tweets becomes essential to come to the rescue of persons and to the affected areas in times of natural or man-made disasters, helping save lives and/or prevent further damage to urban and environmental areas. Section 2.4. explains the concept of Twitter as a social sensor, and how tweets and 'voluntweeters' provide sensorial information that contributes to raising emergency situation awareness, leveraged by competent authorities to take immediate action in emergency-related situations. Section 2.5. provides the linguistic basis for how spatial knowledge is represented in natural languages, in terms of the structure, function, and conceptualization of space in human languages. Section 2.6. presents the techniques and main frameworks used in location detection in symbolic- and probabilistic-based models. Section 2.7. introduces a comprehensive typology of Twitter-based geolocation models where we present previous work on location extraction with different techniques and objectives, with special emphasis on works dealing with the extraction of locative references. Section 3 presents an overview of the research challenges offered by the present thesis, in terms of research issues and limitations (Section 3.1.), research questions (Section 3.2.), and research hypothesis and justification (Section 3.3.). Section 4 describes the objectives which guided our research. Section 5 explains the model and methodology, with a focus on a characterization of what we mean by locative references in terms of their form, semantics and structure (Section 5.1.), the compilation phase of the corpora used in the development and training of our models (Section 5.2.), the development of linguistic resources in LORE (Section 5.3.), and the neuronal networks and other deep-learning techniques that were used in nLORE and how nLORE was trained (Section 5.4.). Section 6 gives details about the computational implementation of the models, i.e. the under-the-hood work on the development of the models and their corresponding tools and resources, using programming languages such as C# or Python. Section 6.1. and Section 6.2. explain the apps of and files required by LORE and nLORE, respectively, to operate. Section 6.3. introduces the evaluation tool developed to assess the performance of the models. Section 7 presents how the evaluation was carried out for each of the models and which evaluation measures were taken into account. Section 7.1. gives the results of the evaluations performed for LORE and nLORE, and Section 7.2. provides an account of the evaluation numbers presented in the tables and figures with many examples explaining the strengths and weaknesses of the models presented, concluding with an account of the limitations and future lines of improvement. Section 8 presents the conclusions with final remarks, highlighting the pros and cons of our models. Section 9 gives the bibliography, and the Appendix section provides flowcharts of the internal rules that operate in LORE.

2. BACKGROUND

2.1. Computational Linguistics and Natural Language Processing: definition, uses, and role of linguists

Computational Linguistics (CL) or Language Engineering (LE) are umbrella terms that encompass NLP and other sub-disciplines such as Natural Language Understanding (NLU). Despite that, CL and NLP are often used interchangeably (Llisterri, 2003). CL is a very hot discipline in today's digital world. It integrates insights and knowledge from Linguistics and Computer Science. In this sense, CL requires the expertise of the humanities in combination with engineering skills. CL can be defined as the scientifically motivated study of human languages from a computational view (Periñán-Pascual, 2012). Computational linguists develop models, tools, and algorithms with the aim of resolving a particular linguistic phenomenon, or for technological purposes in commercial or research contexts that require handling text in human-machine interaction (e.g. automatic speech recognition, conversational agents, text-based geolocation systems, etc.). These computational models can be developed using symbolic-based or rule-based techniques through heuristics (i.e. hand-crafted rules that exploit morphological or lexico-syntactic knowledge), lexical resources, and/or ontologies typically grounded on linguistic knowledge. They can also be developed using probabilistic-based techniques that rely on statistical and mathematical models for the inference of patterns and rules. The latter approaches generally rely on ML or DL algorithms, which may or not need some degree of linguistic feature engineering, i.e. built-in linguistic knowledge and expertise (Periñán-Pascual, 2012). Nowadays, we are still far off from a human-like computational model capable of understanding language with all its intricacies and complexities. Still, substantial progress is being made in many sub-areas such as Information Extraction and Retrieval, Opinion Mining, Machine Translation, etc.

NLP is in many ways related to the practical applications of CL. In other words, NLP can be defined as sub-discipline that deals with the understanding, analysis and interpretation of human languages through computational tasks which may have direct real-world applications or be part of a larger computational system (Cambria & White, 2014). In this sense, the aim of NLP is not so much about understanding how languages internally work or why they work in the ways they do, or how speakers use language in actual discourse, or resolving a particular linguistic phenomenon, but about implementing practical solutions to real-world problems in commercial or research settings that demand the processing of natural-language texts (Periñán-Pascual, 2012). Some of the commonest NLP tasks are:

- Tokenization: splitting sentences into tokens where each token usually corresponds to a word or a punctuation mark.

- Lemmatization or stemming: the removal of inflections of words to obtain their roots or stems.
- POS tagging: the assignation of grammatical categories to tokens.
- Syntactic parsing: the extraction and delimitation of phrases and clauses contained in sentences, and analysis of constituency or dependency relations among syntactic elements.
- Terminology Extraction: the retrieval of relevant words from a given corpus.
- NER: the identification and classification of named entities such as person, organization and/or location entities found in text.
- Relation Extraction: extracting relevant connections between different named entities.
- Sentiment Analysis: inferring the sentiment of people's opinions about a particular product, idea or topic on the basis of the use of positively and negatively oriented words and depending on the linguistic context and the speaker's intention.
- Topic Analysis or Topic Detection: given a sentence, text, or collection of texts, determining its theme or what is talked about, or clustering them on the basis of thematic similarity.
- Text Summarization: providing a summary from a piece of text, using relevant phrases and sentences or rephrasing the main ideas.
- Machine Translation: the automatic translation of texts from one language to another.
- Question-Answering and Conversational Agents: engaging in dialogs that require machine-human interaction by means of questions and answers or taking turns.

Linguists can play a prominent role in Human Language Technologies (HLT) that require expertise in CL and NLP (Llisterri, 2003). Usually, their role has involved compiling corpora and linguistic resources for the implementation of computational models, or devising rules in symbolic-based systems (Periñán-Pascual, 2012). Their role as developers or engineers in the development and implementation of computational models has been non-existent or very limited at best. In this sense, we could claim that their role has been underappreciated and undervalued, if not outright ignored, both in Humanities and Computer Science. Though many NLP tasks successfully perform with probabilistic-based models without built-in linguistic knowledge, symbolic- and rule-based approaches and linguistic-based feature engineering still can provide immense value to any computational system insofar as these systems deal with text and human languages. This consideration, that linguists can play a key role in HLT, drove our interest to develop a rule-based location-detection model that relies on NLP techniques and language-specific resources, as well as a probabilistic-based model that feeds off linguistic knowledge, for their implementation in and application with microtexts to extract locative

references. In this respect, the present PhD project is grounded on an interdisciplinary approach because of its heavy focus on Applied and Theoretical Linguistics, Natural Language Processing, and Human Language Technologies, with the aim of developing original location-detection models based on linguistically-aware NLP techniques and linguistic-based features to tackle real-life issues such as helping competent authorities find the location of a given emergency, incident or crisis event and thus coordinate effective aid and resources to the affected areas and/or persons.

2.2. Location detection and Geographic Information Retrieval

A location refers to a named space that can be computed by means of geographic coordinates or polygons, among other geographic representations (Purves & Derungs, 2015). These named entities of places have been coined 'place names' or 'toponyms' in the linguistic and geographic literature (Levinson, 2003). Location detection or extraction is an Information Extraction task that focuses on the identification and retrieval of locative references from natural language texts (Middleton et al., 2018; Purves et al., 2018). This task corresponds to the area of Geographic Information Science (GIS) or, more concretely, Geographic Information Retrieval (GIR) (Jones & Purves, 2008), a very hot topic that interconnects CL and NLP with Geospatial Artificial Intelligence (GeoAI) (Janowicz et al., 2019). All this reinforces the great relevance and momentum of the field of Digital Humanities in its quest to merge computational methods with the humanistic research, especially in GIS and GIR (Murrieta-Flores & Martins, 2019). All these research areas deal with unstructured text data and the geospatial information contained therein, which is particularly plentiful in the World Wide Web, and one of the most frequently asked queries in web-search engines (i.e. *where*-questions) (Jones & Purves, 2008; Purves et al., 2018; Yingjie Hu, 2018b; Hamzei et al., 2019; Yingjie Hu & Adams, 2020).

As of today, most online social-network sites deliver location-based services, and many social-media microtexts are brimming with locative mentions which could be further utilized with these services (Yingjie Hu & Adams, 2020), all of which reinforcing the importance and relevance of GIS and GIR systems in today's digital world (Sui & Goodchild, 2011). Indeed, handling online unstructured text through geolocation systems becomes of utmost importance for many up-to-date location-based services such as web-search queries, recommendation-based services, sentiment analysis, or emergency-based services (Purves et al., 2018; Yingjie Hu & Adams, 2020). Natural language ambiguity characteristic of unstructured text constitutes in itself a great challenge for GIR systems in the retrieval of locative references, the extraction of spatial relationships, and the location disambiguation process of the extracted spatial knowledge (Frank & Mark, 1991; Al-Olimat et al., 2019). Natural language ambiguity is further exacerbated by the noisy, informal and abbreviated nature of the microtext genre (Baldwin et al.,

2013; Eisenstein, 2013). In this regard, expertise in Theoretical and Applied Linguistics and Computational Linguistics may prove to be essential for the extraction and representation of locative references and spatial expressions in a structured, digitalized format (Stock et al., 2019).

Location detection from text receives other names in the literature, such as toponym recognition (Middleton et al., 2018), geoparsing (Leidner & Lieberman, 2011; F. Liu et al., 2014), geotagging (Gritta et al., 2018, 2019), georeferencing (Purves et al., 2018), or location extraction (Dutt et al., 2018). It should not be confused with geocoding (Middleton et al., 2018), which deals with the assignation of spatial coordinates to location mentions after going through a location disambiguation phase (Gritta et al., 2018). Geocoding is also typically used interchangeably with geotagging. On some occasions, geoparsing or georeferencing is said to be composed of two phases, those of location detection and location disambiguation or geocoding respectively (Gelernter & Balaji, 2013; Purves et al., 2018; Wallgrün et al., 2018). Due to the terminological confusion, we will use and retain henceforth the expressions location detection, location extraction, and location recognition interchangeably to refer to the identification and extraction of locative references from unstructured text. Also, any system that deals with the extraction of location information from either text or other sources will be termed a 'geolocation system'.

2.3. Practical applications of tweet-based geolocation systems in social-sensing settings

Geolocation systems deal with the extraction, disambiguation and/or visualization of geospatial information from text, images and other resources (Hu, 2018a; Janowicz et al., 2019). The focus of the present thesis lies in those geolocation systems that process textual data from Twitter. Most of these text-based geolocation systems incorporate, in their geoparsing modules, a location extractor that is in charge of the detection of locative references found in the texts. Geolocation systems play a key role in social sensing settings, that is, in diverse real-life scenarios where geospatial information proves vital to allocate resources and services to affected areas and/or persons in times of crisis and emergencies (Martínez-Rojas et al., 2018; C. Zhang et al., 2019; Dutt et al., 2019). For instance, in health-related scenarios such as health surveillance or disease tracking, geospatial information obtained from tweets can be exploited by public health and medical officials for tracking or prevention measures in disease propagation (Eke, 2011; Dredze et al., 2013) such as tracking the location of people infected with the influenza virus (Santillana et al., 2015; Vilain et al., 2019) or infected with the current COVID-19 (Singh et al., 2020), or to perform opinion mining together with geolocation about a controversial medical issue such as vaccination to know about the sentiment expressed by people depending on their location (Luo et al., 2019). With regards to the current COVID-19 outbreak, Singh et al. (2020) highlighted the importance of geolocation systems in the extraction

of location mentions in tweets for disease forecasting and prevention purposes. In this regard, they claimed that a greater incidence of confirmed cases of people infected with COVID-19 in particular locations highly correlated with a greater number of tweets that dealt with the coronavirus pandemic mentioning those locations.

Not only can these applications be derived from tweets, but also from biomedical texts (Magge et al., 2018), which can be further utilized for medical research. Other uses in other types of text genres involve drawing on cultural, historical and/or literary texts to study heritage data on the basis of the locations mentioned in such texts (Gregory et al., 2015; Kew et al., 2019; McDonough et al., 2019).

Many emergency-based services employ natural and/or human-made disaster detection and tracking systems with a geolocation module for tweets in the case of floods, earthquakes, storms, civil unrest, war, crime, etc. (Vieweg et al., 2010; Crooks et al., 2013; Imran et al., 2014; Jongman et al., 2015; Martínez-Rojas et al., 2018; C. Zhang et al., 2019; Siriaraya et al., 2019). Tweet-based geolocation systems can also be vital for real-time traffic-incident detection, road-traffic control and/or traffic congestion (Ahmed et al., 2019; C. Zhang et al., 2019; Gonzalez-Paule et al., 2019; Khodabandeh-Shahraki et al., 2019), where user-generated content, either in the form of attached GPS coordinates or through microtexts mentioning locative references, can help track the location of vehicle accidents on roads, highways, streets, avenues, etc. and thus send this valuable information to competent authorities to coordinate further action. Another practical application derived from tweet geolocation systems is that of marketing and advertising, where the locations mentioned by Twitter users can be exploited to suggest potential places for these users to visit or attend to or local products to buy (Li & Sun, 2014).

2.4. Social sensors and emergency situation awareness

Sensors are devices that detect an input signal –typically, from the physical environment– for its subsequent processing. For instance, a thermometer is a sensor that receives temperature as input and processes it to display such data in centigrade and/or Fahrenheit degrees. Likewise, Twitter can act as a real-time social sensor for crisis events, whereby each Twitter user is seen as a social sensor and their tweets as sensory information used for the reading of a particular crisis-related event (Aggarwal, 2013). In times of crisis events, people, or the so-called ‘voluntweeters’ (Starbird et al., 2011), increasingly turn to Twitter to report and inform about the occurrence and ongoing circumstances of disaster-related emergencies (Potts et al., 2011), e.g. the epicenter and trajectory of those disaster-related events, especially when there is potential damage to places or to people around them (Martínez-Rojas et al., 2018). In other words, when a car accident happens, for instance, witnesses as Twitter users (i.e. social sensors) may post about it (i.e. input signals) to report the incident to competent authorities for them to

come to the rescue of the injured persons, or to advise other drivers to take another route. Emergency responders react to these signals by coordinating effective aid and relief efforts and allocating resources in the affected areas and to the affected persons (Cameron et al., 2012). Tweets containing location information, either geotagged or mentioned by means of locative references, in critical emergency-related situations are more likely to circulate, which means that Twitter users are aware of the great importance of attaching geospatial information in raising emergency situation awareness (Imran et al., 2014).

2.5. The representation of spatial knowledge in natural languages: a linguistic-based approach

According to the linguistic literature (Herskovits, 1985; Landau & Jackendoff, 1993; Talmy, 2000; Kracht, 2002; Levinson, 2003; Coventry & Garrod, 2004; Bennett & Agarwal, 2007; Radke et al., 2019; Stock et al., 2019), spatial knowledge, typically represented by spatial prepositions in analytical languages, indicates a spatial relationship held by different entity types or arguments, formally expressed as S(*x*, *y*), where S determines the kind of spatial relationship held by *x* and *y*, *x* refers to what is spatially defined, and *y* represents the region of space occupied by *x*.

2.5.1. The structural, syntactic, conceptual and pragmatic features of spatial expressions

From a structural standpoint, in Western European languages such as English, Spanish, or French, a spatial expression is generally composed of a 'subject' (i.e. what is located) and a prepositional phrase (PP) made up of a preposition and an 'object' (i.e. where is located). This PP can modify a noun (e.g. *the glass on the table*), or predicate something about a noun phrase (NP) (e.g. *John is at school*) or a clause (e.g. *He is buying groceries at the market*) (Geis, 1975; Herskovits, 1985; Creary et al., 1989). The object or 'place' refers to a physical location, real or imaginary, which describes the position, direction/path, or distance of a given entity (Kracht, 2002; Coventry & Garrod, 2004; Bennett & Agarwal, 2007; Cinque & Rizzi, 2010). Whereas position indicates a spatial relationship of location among objects, path specifies a trajectory understood in terms of source and goal, and distance provides a measure of space among two or more entities.

In natural languages, places are typically encoded as nouns, which can be proper if used to identify a specific and unambiguous spatial region or portion (e.g. *Granada, Valencia, Spain, France*), receiving the name of 'toponym' or 'place name' (Levinson, 2003; Stock et al., 2019), or common when they are used in a generic sense, often representing a semantic type of different granularity (e.g. *neighborhood, city, country, beach, canyon, street, road*). They can

also be formally represented by means of complex NPs (e.g. *the black chair next to the table standing in the corner*), which can recursively become very intricate, especially if multiple reference frames are mentioned (Stock et al., 2019). Places can be formally represented in ontologies where subsumption relationships specify the type of place or other conceptual relationships held by toponyms (e.g. Madrid-IsA-city, California-IsPartOf-United States, etc.) (Bateman et al., 2010; Hu, 2018b).

As far as syntax is concerned, in the clause spatial expressions can go either at the beginning (e.g. *In Tokyo the earthquake caused great damage*) or at the end (e.g. *Floodings were reported in New Jersey*). As mentioned above, they specify different semantics, such as position (e.g. *John lives in New York*), direction (e.g. *An ambulance is heading to Glenwood Avenue*), or distance (e.g. *Mary drove for 35 miles southwest of London*) (Quirk et al., 1985: ch. 8). Their referent can be the subject (e.g. *Paul flew to Los Angeles*), the direct object (e.g. *I parked the car at Nevada Shopping*) or even both (e.g. *I met Anna at the National Museum*). Spatial expressions typically perform the adverbial function in the clause (Geis, 1975), although they can also act as postmodifiers of a noun in an NP when formally realized as PP (e.g. *The man outside the bus station is waiting for his friends*) (Quirk et al., 1985). According to Quirk et al. (1985), spatial expressions performing the adverbial syntactic function of space adjuncts can be either obligatory (e.g. **John lives*) or optional (e.g. *We bought groceries* (*at Tesco*)) (Geis, 1975). When obligatory, these syntactic units additionally perform the function of postmodifiers with verbs of stative meaning (e.g. *be, live, stand, lie*…). The formal realization of these phrases as space adjuncts can be NP (e.g. *John walked five miles*), PP (e.g. *Mary was a teacher in Newcastle*), Adverbial Phrase (AP) (e.g. *The warriors died there*) or subordinate clauses of distinct complexity (e.g. *The missing boy was found where the police could have not ever imagined*). PP is the most typical phrasal realization and the most connected with spatial expressions (Quirk et al., 1985: Ch. 9). Also, since our interest is in place names, and these are nouns, we only take into account NPs and PPs that introduce these NPs.

Spatial prepositions act as linkers to encode spatial relations between objects or between an object and a region/place (Landau & Jackendoff, 1993). A distinction should be made between those spatial prepositions indicating location (i.e. locative prepositions *in, at, near, en, à, dans, sur*, etc.) and spatial prepositions indicating direction (i.e. directional prepositions such as *to*/*from, hacia, vers*) (Coventry & Garrod, 2004). Locative prepositions can be further divided into topological terms that express topological relations among entities (e.g. *in, at, on, near, en, dans, sur, à*, etc.) and projective terms that need a frame of reference (e.g. *in front of, above, to the right, arriba de, à droite de*, etc.). The prepositions *in* and *at* are prototypical items of locative prepositions in English (Levinson, 2003), *en* in Spanish, and *dans*, *en* and *à* in French. In English, these prototypical locative prepositions obey different patterns for their usage in discourse: *in* is usually reserved for large geopolitical entities such as districts, regions, cities,

countries, continents, etc., or to refer to the dimensional side of buildings (e.g. *John works in a record company*), whereas *at* is rather used with small geopolitical entities (e.g. *Mary lives at Stratford-upon-Avon*) and buildings in the institutional and functional sense (e.g. *John works at a record company*) (Quirk et al., 1985; Vasardani et al., 2013). In Spanish, *en* goes with practically any location type, from geopolitical entities or POIs to traffic ways, etc. (e.g. *en España, en Murcia, en el museo Reina Sofía, en la carretera Montejícar*). In French, *à* accompanies towns (e.g. *à Paris*), some POIs (e.g. *au restaurant Le Ciel*), and countries in the masculine gender (e.g. *aux Etats-Unis*), whereas *en* is reserved for countries in the feminine gender (e.g. *en France*) and *dans* for many POIs, especially in the physical sense (e.g. *dans la cathédrale Notre Dame*), and traffic ways (e.g. *dans la rue Bellevue*).

From a conceptual standpoint, a spatial relationship is said to hold between a figure (object) and ground (reference object) in a spatial scene (e.g. *the car near the house*) (Talmy, 2000). Spatial relations can be binary consisting of one figure and one ground (e.g. *x in y*, *x across y*, etc.) or *n*-ary consisting of one figure and several grounds (e.g. *x between y and z, x in front of y and z,* etc.) (Landau & Jackendoff, 1993). Each of these spatial arguments is conceptually defined in different terms. In the case of the figure, its spatial properties are unknown, it is more relevant to the speakers and is thus the topic to be informed about to the addressee. In the case of the ground, it acts as a reference entity and background for the figure with less relevance and permanently fixed in space. Large and fixed objects are usually used as ground because of their salience (e.g. *Your wallet is in the car*). However, this condition is less strict and more volatile with certain locative prepositions such as *next to* or *behind*, where it may deviate to some extent (e.g. *pick the glass next to John*). In this sense, spatial meaning underlying some prepositions can be described in terms of systematicity and idiosyncrasy (Herskovits, 1985). In other words, pragmatic principles such as relevance, salience, tolerance, and typicality play a key role in the choice and interpretation in the use of locative expressions. An ideal or prototypical-like meaning characterizes each preposition. For instance, *in* involves a relation of containment (e.g. *clothes in the closet*), which occasionally may slightly shift (e.g. *the bird in the tree*). Between the speaker's representation of the physical world and the use of spatial expressions lies geometric conceptualizations, in that the use of distinct spatial prepositions with respect to their context signal different geometric descriptions, relations, or schemas. For example, *to* and *from* indicate a geometric line between the ground and the figure (e.g. *from the airport to the hotel*); *in* signals a three-dimensional geometric space (e.g. *the crack in the glass*); other prepositions signal a frame of reference (e.g. *in front of, behind, to the left, to the right*…), etc. From a pragmatic point of view, the purpose of spatial expressions is to tell the addressee about the location of a given figure or to identify it (Talmy, 2000).

2.5.2. Named entities of places: toponyms and geographical names

On the one hand, toponyms or place names can be defined as named place specifications which by themselves do not provide a precise frame of reference, typical of quantitative methods involving coordinate systems (Levinson, 2003). They can be accordingly casted into a generic semantic class (e.g. London:city) (Bennett & Agarwal, 2007). Ascribing a place name to a particular location is a special type of topological relation whereby the place name acts as the ground location of a given figure (e.g. *John lives in London*) (Levinson, 2003). In this sense, as Levinson (2003: 69) claims, toponyms offer an "underlying mental map of locations" which speakers can have access to and more or less place on a map.

Geographical names include place names in their lexical scope with the addition of location-indicative nouns, also called "descriptors" with an "appositive function" (Quirk et al., 1985: 1317): e.g. *Mount Everest, New York State, Sunset Boulevard*, etc. In the English language, the 'name-first construction' is especially common, where location-indicative nouns typically follow toponyms (e.g. *Nile Valley, Quebec Province*). It is not rare, however, to find examples of location-indicative nouns preceding place names (e.g. *River Thames*). At times, both can be reversed (e.g. *Cork County* or *County Cork*). At other times, location-indicative nouns and place names can be linked by the preposition *of* as in the *State of Missouri*, the *Island of Cyprus*, or the *coast of New Zealand*. In Spanish and French, location-indicative nouns precede place names (e.g. c*alle Vicente Montuno, residencia La Inmaculada, École Thérèse D'avila, la ville de Lyon*), and can also be linked by the preposition *de* in both languages and/or definite determiners such as *el*, *la*, or *le* (e.g. *ciudad de Granada, barrio del Albaicín, la ville de Lyon, Hôtel La Residence Du Vieux Port*).

Toponyms and geographical names alike are often preceded by spatial prepositions (Al-Olimat et al., 2019), though they do not necessarily need to be accompanied by them (e.g. *Madrid is the capital of Spain*). Also, toponyms do not always act as grounds but may act as figures in certain spatial configurations (e.g. *Granada se encuentra entre las ciudades de Jaén, Almería, Málaga, and Córdoba*).

Overall, spatial knowledge as lexicalized by means of toponyms and geographical names in natural languages is fuzzy and ambiguous, needing further contextual clues and approximation for computing systems (Al-Olimat et al., 2019). This is especially problematic when locative markers (e.g. *south of, 30 min away from, 45 miles SW from, 35 kms al noroeste de, 10 kilomètres au sud de,* etc.) precede named place nouns, which pose a greater problem for quantitative geographic positioning (e.g. latitude/longitude, the commonest Geographic Coordinate System (GCS)), since locative markers do not usually provide a precise, specific geographic landmark that can be easily mapped to coordinates. Recent proposals have been put forward for a more precise delimitation of toponyms in geocoding systems involving the use of polygons, instead of coordinates (Al-Olimat et al., 2019).

The present research focuses on topological relations involving both toponyms and geographical names. This means that our models have been developed taking into account named entities of places together with the most relevant and also the least ambiguous spatial prepositions of the languages supported, mainly of locative type, as one of the contextual clues used in the retrieval of locative references. Further details about the formal and semantic criteria followed in the delimitation and classification of named entities of places, henceforth called 'locative references', are discussed in Section 5.

2.6. Microtext genres: the tweet subgenre

Among social media and, in particular, among microblogging services, Twitter stands out as one of the most popular worldwide microblogging platforms for information sharing and communication purposes (Murthy, 2018; Stock, 2018). Moreover, since its application program interface (API) is research-friendly (Gelernter & Balaji, 2013), it is the development platform of preference for researchers who can have almost unlimited free access to vast amounts of data that can be handled for many NLP tasks. In Twitter, users can post microtexts, called tweets, which are brief, character-limited (280 characters max.) messages that typically express the users' thoughts, activities, and opinions about their daily lives or about a given topic (Yuheng Hu et al., 2013).

Figure 1. Tweet about an accident.

Microtexts are usually informal, noisy and abbreviated. This means that language conventions generally deviate from the linguistic norm through informal language devices such as abbreviations (e.g. *pls* instead of *please*), acronyms (e.g. *FYI* instead of the phrase *for your information*), misspellings (e.g. *madrizz* instead of *Madrid*), lack of capitalization (*united kingdom* instead of *United Kingdom*), ungrammatical forms (e.g. *you was* instead of *you were*), ellipsis and truncated sentences (e.g. *incident in Newcastle* instead of *There was an incident in Newcastle*) (Baldwin et al., 2013; Eisenstein, 2013). In this regard, one particular challenge in the identification of locative references in tweets is related to the linguistic peculiarities of the microtext genre. Most NLP systems, which have historically been trained on formal genres such as the news genre, face problems when applied to tweets and, as a result, their performance is usually much degraded (Hoang & Mothe, 2018). This occurs because these systems rely on

proper spelling, capitalization and grammatical patterns for different NLP tasks (i.e. POS tagging, chunking, etc.) and, in the absence of that, their predictive power decreases. Several strategies have been proposed to overcome the present linguistic difficulties in NLP systems applied to Twitter, such as the normalization of the tweet text (Liu et al., 2012), and/or the adaptation of NLP tools to social media genres and their linguistic idiosyncrasies (Eisenstein, 2013). However, despite the widely-believed claim that tweets are noisy and informal, Yuheng Hu et al. (2013) disagree as to the apparent degree of informality of the tweet genre, arguing that tweets, surprisingly enough, are not as informal as other microtext genres (e.g. SMS). In fact, according to the authors, tweets can be considered as a projection of other formal textual genres onto a size-restricted format.

Tweets may also encode metadata or additional information about author profile, time of posting, and spatial coordinates when they are geotagged, which can also be useful for geographic applications. Given the informal and noisy character of tweets and the little amount of geotagged tweets, it comes as no surprise that the scientific literature that has dealt with location extraction models and techniques has struggled to provide high performance tools to address those issues (Gelernter & Balaji, 2013; de Bruijn et al., 2018; Hoang & Mothe, 2018; Middleton et al., 2018, among many others).

2.7. Techniques and main frameworks used in location detection

Location detection involves recognizing and extracting locative references in unstructured text through probabilistic-based methods grounded on ML or DL frameworks and/or symbolic- or rule-based methods that exploit linguistic evidence with hand-crafted rules and lexical resources (Yingjie Hu, 2018a). Some of the NLP techniques used in geolocation models are tokenization, POS tagging, n-grams, and syntactic chunking. The commonest approach to location detection is NER, which is a line of research in NLP in the fields of Information Extraction and Information Retrieval that deals with the identification and classification of named entities, not only location names but person names and organization names, *inter alia*, extracted from a corpus of texts (Barrière, 2016; Goyal et al., 2018).

Most existing NER approaches to location extraction in text perform reasonably well (Karimzadeh et al., 2019), especially those developed with and implemented for formal genres (de Bruijn et al., 2018). To tackle the noisy and informal nature of tweets, Twitter-specific NER-based tools for microblogging services have been implemented (Karimzadeh et al., 2019). As a rule of thumb, NER systems usually experience performance drops when dealing with non-standard spelling, typographical, and grammatical characteristics of social-media microtexts. Another issue is related to the coarse semantic granularity of standard NER tools with locative references, since the location types matched by standard NER tools are not clearly delimited

(Van et al., 2013). In this regard, most NER systems, including those specifically targeting tweets, can detect geopolitical entities, a few natural landforms, and a few POIs and addresses, at the most.

According to Jurafsky & Martin (2018a) and Li et al. (2018), there are mainly three types of NER models:

- feature-based NER, which employs ML algorithms such as Conditional Random Fields (CRF) (Finkel et al., 2005; Han, Jimeno-Yepes, et al., 2014) or Hidden Markov Models (Sarkar, 2015),
- neural NER, which uses DL techniques such as bidirectional Long Short Term Memory (biLSTM) (Gerguis et al., 2016; Limsopatham & Collier, 2016), usually in combination with Convolutional Neural Networks (CNN) (Dugas & Nichols, 2016; Aguilar et al., 2018), and
- rule-based NER, which is based on hand-crafted lexico-syntactic rules typically using regexes and lexical resources (Malmasi & Dras, 2016; Dutt et al., 2018; Yang-Lim et al., 2019).

2.7.1. Feature-based NER

In feature-based NER, sentences are tokenized, where each token or word is taken as a vector with a set of attributes or linguistic features, typically returning string or Boolean values, i.e. either true or false (Nadeau & Sekine, 2007; Leidner & Lieberman, 2011; Middleton et al., 2018). Some of these linguistic features are capitalization, POS tags, affixes, gazetteer inclusion, or shallow syntactic features (i.e. chunk labels such as NPs, VPs, etc.). Two steps are essential to build any ML model: a training phase and a testing phase. First, we train an ML algorithm with a training corpus, which contains manually-tagged data in the form of the different linguistic features, represented in a tabular-based format. In the case of NER, named entities are delimited in terms of their boundaries with different tagging schemes such as IOB (Inside-Outside-Beginning) or BMESO (Beginning-Medial-End-Single-Outside), together with their POS tag (Figure 2).

Figure 2. Tabular-based representation of tokens in NER.

Token	POS tag	Label
John	NNP	O
lives	VBZ	O
in	IN	O
New	NNP	B-LOC
York	NNP	E-LOC

After this phase, the model is applied to a test corpus, i.e. unseen data, to measure the performance of the model; in other words, we evaluate how well the model was trained with the training corpus by making some predictions on the test corpus. For example, the Stanford NER tool uses a CRF algorithm, achieving very high performance in the news genre (Finkel et al., 2005), but that performance considerably degrades in the Twitter medium. However, its performance increases when the model is retrained with tweet data (Lingad et al., 2013; Hoang & Mothe, 2018). In this regard, Ritter et al. (2011) implemented a well-known feature-based Twitter-specific NER tool which can detect named entities such as locations, person names and organization names. Le et al. (2016) used a CRF model in their Twitter-based NER system, using a great variety of spelling, lexical, and syntactic features fed into the training stage of their model.

2.7.2. Neural NER

Neural NER models achieve very good performance in many NER tasks (Espinosa et al., 2016; Yadav & Bethard, 2019), despite generally not needing any manual feature engineering of linguistic features or lexica. DL models can automatically discover features in the training process. The architecture of DL-based NER is characterized by three different components: distributed representations for input, context encoder, and tag decoder (Li et al., 2018). The distribution representation component may incorporate linguistic-based features such as capitalization, spelling, POS tags, chunk tags, gazetteer inclusion, and word- and/or character-embeddings, thus adopting a tabular-based format as in feature-based NER. The context encoder component captures contextual information from text using CNN, Recurrent Neural Networks (RNN) and/or biLSTM or, more recently, using Transformers and language models (Devlin et al., 2018). Recently, the later has gained traction, especially thanks to Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018), a language model based on bidirectional encoder representations using Transformers that yield promising results in any kind of NLP task and in NER tasks in particular. The tag decoder component is in charge of predicting the final output or labels from a text sequence by means of, for instance, a CRF or

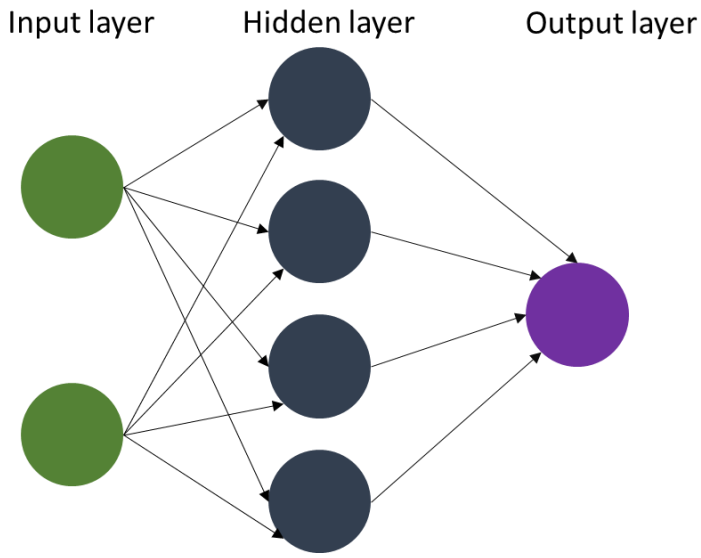
softmax layer. With regard to Twitter-specific NER tasks, biLSTM and/or CNN have been successfully applied (Dugas & Nichols, 2016; Espinosa et al., 2016; Aguilar et al., 2018). The most successful and popular neural NER systems consist of biLSTM networks with an added CRF layer on top, resulting in state-of-the-art sequence labeling performance (Limsopatham & Collier, 2016). Current research emphasizes that neural NER models should still consider linguistic-based feature engineering to achieve superior performance (Yadav & Bethard, 2019), especially in the case of user-generated content such as tweets (Li et al., 2018). In this regard, emerging research is corroborating such claims (Aguilar et al., 2019).

Since neural NER is considered a state-of-the-art approach and is also the framework used in the development of our DL-based model, we present a brief introduction on the theoretical principles behind neuronal networks to understand their working mechanisms.

2.7.2.1. Neuronal networks: layers, neurons and hyperparameters

Neural NER models rely on Artificial Neuronal Networks (ANN) which, imitating the functioning of biological neurons, receive data as input to learn and infer patterns that are then put to the test to see whether they can generalize with good performance (Gurney, 1997; Cole, 2018). An ANN consists of the following parts: an input layer, a hidden layer, and an output layer (Figure 3). These layers are nodes that transform real-world data into numerical values and process them to obtain an output that is then learned by the algorithm. The hidden layer, placed between the input and the output layers, receives weighted inputs and produces an output by means of an activation function. Usually, neuronal networks contain multiple hidden layers, in which case they receive the name of ‘deep neuronal networks’. The number of neurons in an input layer depends on the number of properties or features, where each neuron represents a given feature. In the case of the output layer(s), the number of output layers depends on the nature of the algorithm. In neural NER, since NER is a multi-class classification task, the output layer consists of as many layers as instances need to be identified. For instance, in the task of location extraction, we could use a scheme such as IOB whereby the classes to be identified are B-LOC, I-LOC, and O. The number of hidden layers and neurons in them depends on the nature of the algorithm and the task, too. Usually, 1-5 layers are employed with 1-300 neurons each.

Figure 3. Layers and neurons in a simple neuronal network.



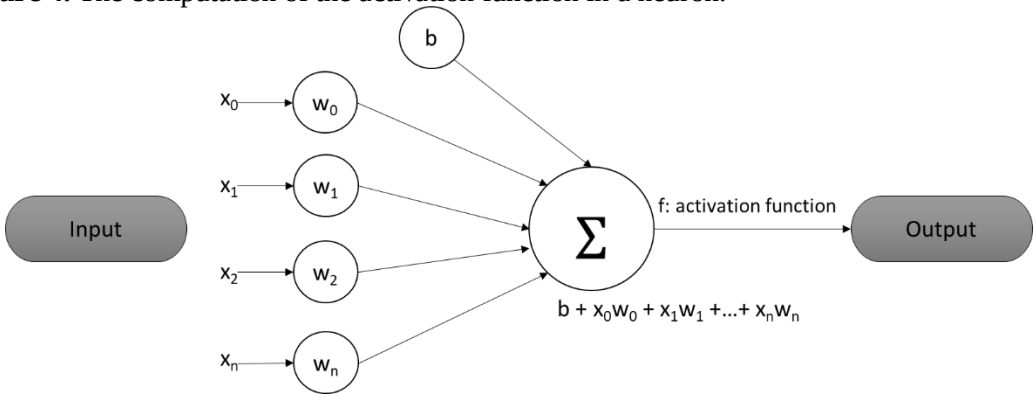
Each layer is made up of neurons defined in terms of weights and biases. When a neuron processes input data, the neuron computes the network hyperparameters, often set randomly, through a linear function, where W represents the weight, X the input data, and b the biases (Equation (1)).

$$Z = W \cdot X + b \tag{1}$$

Z represents a linear regression. However, since a neuronal network must also learn non-linear patterns, another different function, the activation function (Equation (2)), must be computed after calculating Z (Figure 4).

$$A = g(Z) = g(W \cdot X + b) \tag{2}$$

Figure 4. The computation of the activation function in a neuron.

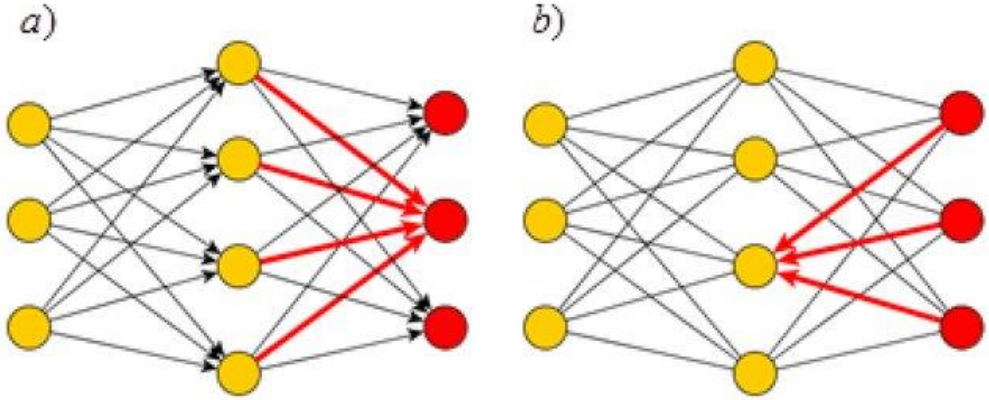


The role of the activation function is to process a node's input signal to convert it into an output signal, which is, in turn, used as input for the next layer. A neuron fires or not depending on the

activation function. Activation functions can be of different types, depending on the nature of the neural network: sigmoid, hyperbolic tangent, rectified linear unit, etc.

In the training phase of a neuronal network, the layers of neurons learn the values of the parameters by going forward and backward in an iterative process, usually called epoch or iteration (see Figure 5 below). Such movements are termed 'forward propagation' and 'backpropagation'. First, training data pass through all the layers and their neurons, which apply the activation function with the set parameter values, and reach the final layer with an output label predicted on the basis of the previous computations. By means of a loss function, the algorithm estimates and measures the degree of success in determining the output label by comparing the predicted label with the correct label: the closer to zero, the less divergence there is between the output label and the correct label. During the training process, the weight values are gradually adjusted to obtain better predictions so that our neuronal networks learn optimal parameters. While our algorithm computes the loss function, it propagates this information backwards through backpropagation. In other words, from the output later, the hidden layers receive the contribution of the loss function relative to each neuron.

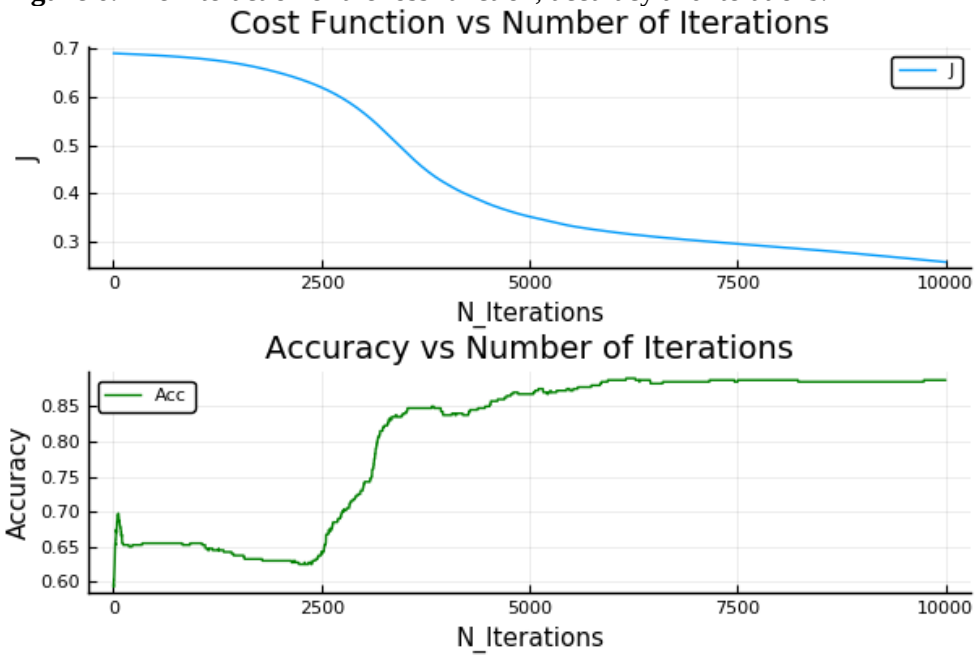
Figure 5. Forward propagation and backward propagation (Negrov et al., 2015).



On the basis of this process, weights are also adjusted to achieve a better prediction. In order to minimize the loss function, gradient descent, a very useful technique in neuronal networks, is introduced by slightly modifying the weights with small increments in each iteration through the computation of the derivative of the loss function multiplied by the learning rate. The learning rate is a hyperparameter which affects the learning speed by determining how quick the weights shift in each training epoch. The learning rate value usually ranges from 0.001 to 1. The lower the learning rate, the more accurate the estimation because of smaller increments in weights, but the more time it takes to train the network. It thus remains a matter of finding the most adequate value to balance accuracy and computational cost and time. The end result in the training phase

of the neuronal network must be such that the loss function is gradually minimized in each iteration which, in turn, helps increase the accuracy of the trained neuronal network (Figure 6).

Figure 6. The interaction of the loss function, accuracy and iterations.

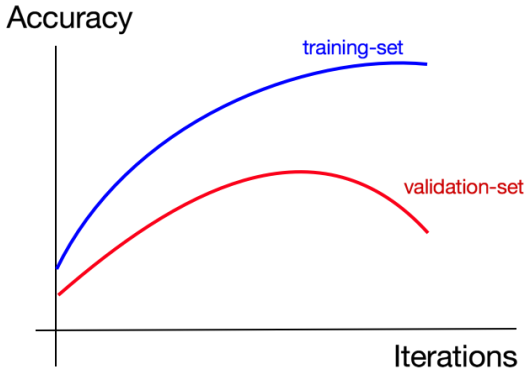


Besides the hyperparameters above mentioned, there are others which can positively impact the training performance of the neuronal network. One of such hyperparameters is called regularization. Regularization prevents overfitting, which refers to the phenomenon of achieving great accuracy with the training dataset at the expense of poorer performance with unseen data. Regularization works by decreasing the weights' increments so that they become more regular. This is done by adding a cost to the loss function. One of the commonest regularization techniques is called dropout, whereby a number of random output values are set to zero during training. Dropout values typically range from 0.2 to 0.5, representing a fraction of output values. Another important hyperparameter is called batch size, which selects a random subset of training data (i.e. a mini-batch) to be used in each iteration, usually ranging from 10 to 1000. Optimal performance in terms of speed and accuracy has been observed with minibatch sizes of 2 up to 32. The process of finding optimal hyperparameter values is termed hyperparameterization or hyperparameter tuning or optimizing, traditionally performed with manual tuning or trial and error.

Focusing on improving accuracy alone in the training phase can lead to overfitting or underfitting problems. In other words, the resulting neuronal network is a highly accurate one, but only when applied to the training dataset. This means that it might not generalize well when presented with new data. The main objective of training a neuronal network is to obtain a neuronal network capable of making accurate predictions with unknown data, that is, to be

generalizable. To avoid overfitting and underfitting, the training phase is complemented with a validation process. In probabilistic-based frameworks, validation means that the dataset used for training is split into two datasets: a training set of, usually, 70-90% of the samples, and a validation set of 10-30% of the samples. The validation process is performed in each iteration to test the performance of the trained neuronal network. When training accuracy increases but validation accuracy decreases, this leads to overfitting (Figure 7).

Figure 7. Overfitting.



This phenomenon can be avoided by using more training data or, in the case in which the amount of data is not the source of the problem, reducing the complexity of the neuronal network by changing the number of layers and/or neurons, or with regularization techniques. On the other hand, when both training accuracy and validation accuracy decrease, we may be presented with underfitting. This phenomenon results from low complexity in the model, and can be solved by adding more layers or neurons.

2.7.3. Rule-based NER

Both feature-based NER and neural NER are based on probabilistic models, whose performance when applied to unseen corpora of texts largely depends on the coverage, quantity, and quality of the training data (Purves et al., 2018). In general, these NER models require large annotated corpora, whose annotation and preparation process is time-consuming and labor-intensive (Li et al., 2018). In contrast, rule-based NER is based on a symbolic model, which makes use of hand-crafted lexico-syntactic rules and lexical resources that help extract locative references from the text. These rules can usually take the form of regexes, which are formal notations that follow a specific syntax for finding patterns in text strings. They are used in rule-based NER to capture linguistic patterns in text strings from the knowledge provided by NLP tasks such as tokenization and POS tagging. For example, the presence of locative prepositions followed by proper nouns is usually taken as a strong linguistic cue that can be exploited to extract locative

references (Hoang & Mothe, 2018). Other cues that can be exploited involve the combination of different phrasal patterns, such as verbs of movement followed by prepositional phrases (Béchet et al., 2011; Van et al., 2013; Moncla et al., 2014). One of the most well-known rule-based NER system is GATE (Cunningham et al., 2002), which leverages regexes, linguistic knowledge, and resources to detect named entities with great precision. Overall, rule-based NER alone can achieve very high precision but low recall (J. Li et al., 2018; Jurafsky & Martin, 2018a; Yadav & Bethard, 2019). In the NLP research community, rule-based approaches are seen as a dead-end technology, whereas DL and ML-based approaches heavily dominate the NLP landscape (Chiticariu et al., 2013). According to Chiticariu et al. (2013), this disregard of rule-based approaches in academia contrasts with the preference of companies and industries which still opt for rule-based approaches in the business world. As noted by the authors, the reason for this preference in business settings is due to the domain specificity of the rules and the lack of need of large amounts of data for their development, which translates to better suited practical applications, and the runtime efficiency of the rules. Moreover, the reusability and scalability of a rule-based model facilitate debugging and adapting the rules to new scenarios, whereas probabilistic-based models require gathering large and extensive labeled data and a more demanding fine-tuning process in the case of retraining phases with heavier computational costs, time, and resources. On the other hand, NLP researchers in academia consider that manually devising rules is a labor-intensive, time-consuming task that requires domain-specific knowledge and expertise in areas such as linguistics, steering away from their mathematical or computational background. Be that as it may, in biomedical NER, rule-based systems still show superior performance to those based in ML or DL (Gorinski et al., 2019).

2.7.4. Named Entity Matching

On the other hand, another frequently used approach to location detection is Named Entity Matching (NEM) (Leidner & Lieberman, 2011; Middleton et al., 2018). NEM consists in the use of digitalized lexical lists or gazetteers of named entities of places retrieved from geographic databases (Laurini & Kazar, 2016; Yingjie Hu, 2018b) such as GeoNames[1] (Ahlers, 2013) or OpenStreetMaps[2] (Acheson et al., 2017) for their application in the identification of locative references in text through a lookup, typically by exploiting n-grams (Middleton et al., 2014; Malmasi & Dras, 2016; de Bruijn et al., 2018). In computational linguistics, n-grams are described as linear sequences or combinations of *n* words in a given sample of text. Thus, in the sentence *The quick brown fox jumps over the lazy dog*, we can find unigrams or n-grams of size $n = 1$ (e.g. {the}, {quick}, {brown}, {fox}...), bigrams or n-grams of size $n = 2$ (e.g. {the quick},

[1] www.geonames.org
[2] https://www.openstreetmap.org/

{quick brown}, {brown fox}…), trigrams or n-grams of size n = 3 (e.g. {the quick brown}, {quick brown fox}…), and so on.

The GeoNames database is one of the most widely used in location-detection systems. It is a very large and comprehensive database of geographic references containing around 25 million places. Some of the location types stored are, using the GeoNames terminology, administrative places (i.e. geopolitical entities), natural features (i.e. natural landforms), and some urban features (i.e. POIs and traffic ways). Besides named entities of places, it contains other geographic-related data such as population size and latitude-longitude coordinates, which can be very helpful for location disambiguation purposes and map geovisualization (Purves et al., 2018). However, despite including a high number of places, GeoNames lacks location subtypes such as addresses, roads, buildings, etc. (Ahlers, 2013; Dutt et al., 2018). NEM systems for location detection in tweets seem to achieve greater performance than NER systems (Middleton et al., 2014). Sometimes, both are used jointly (Stock, 2018). However, NEM presents several drawbacks. First, geographic databases are finite, so they might not capture the full range of existing places (Purves et al., 2018). Second, these models cannot disambiguate proper nouns of named entities of places from proper nouns of person names, e.g. the city of *Paris* from *Paris Hilton* (Gritta et al., 2019): this phenomenon is known in the literature as geo/non-geo ambiguity (Amitay et al., 2004).

Finally, Middleton et al. (2018) suggested that a hybrid approach, based on the combination of NER with NEM, can greatly reduce the number of errors. Although most location-detection models perform relatively well with a few or even without linguistic features, it is our contention that they fail to fully exploit the linguistic knowledge that permeates natural-language texts, e.g. locative prepositions (e.g. *in, at, near*, etc.), location-indicative nouns (e.g. *avenue, city, province, road, school, street,* etc.), or locative markers (e.g. *south of, XX kms away from,* etc.) that signal the presence of named entities of place (Hoang & Mothe, 2018). This view emphasizes the need to develop rule-based location-detection systems that could improve state-of-the-art performance without requiring the significant amount of processing time and computational resources involved in ML and DL techniques (Gelernter & Balaji, 2013; Malmasi & Dras, 2016; Dutt et al., 2018; Middleton et al., 2018). In this context, one of the main contributions of our research lies in the heavy linguistic focus and the fine granularity of the extracted locative references, unlike previous NER and/or NEM models.

2.8. A typology of Twitter-based geolocation models

There exists a great range of location-based systems for Twitter that take into account different variables and data according to the targets, aims and methods (Ikawa et al., 2016; Stock, 2018; Zheng et al., 2018). We present a typology of geolocation systems for Twitter that considers, on

the one hand, the targets and aims, on the other hand, the methods employed and, finally, the data used. For each classification, we cite the most relevant and important pieces of research and models and a brief description of some of their main contributions.

2.8.1. Classification of Twitter-based geolocation systems in terms of target

2.8.1.1. User location

Some approaches are aimed at extracting the user location on the basis of the user's profile information, tweet metadata, and/or the user's tweet history (Cheng et al., 2010; Kinsella et al., 2011; W. Li et al., 2011; Han, Cook, et al., 2014; Alex et al., 2016; Miyazaki et al., 2018; Mourad et al., 2019). For instance, W. Li et al. (2011) studied the linguistic and social implications attached to POIs by tweet users using tweet geotagged metadata. In the case of Kinsella et al. (2011), they created language models of locations on the basis of geotagged tweet data for advertising content and nearby places to Twitter users. Han et al. (2014) proposed a geolocation prediction system for user location at city level through location-indicative words in tweets and user profile information and examined the influence of non-geotagged tweets, language variation, and metadata for geolocation inference. Leveraging non-geotagged tweets significantly improved location accuracy from 12.6% to 28% on a benchmark dataset. Dredze et al. (2013) introduced CARMEN, a geolocation algorithm that detects toponyms of different administrative levels (country, state, county, city) of Twitter users through their tagged coordinates and profile information aimed at public health and medical applications such as disease surveillance and propagation. Miyazaki et al. (2018) devised a knowledge-based neural network framework for Twitter user geolocation that exploits the user's tweet history with semantic relations (e.g. isLocatedIn, livesIn, happenedIn…).

2.8.1.2. Tweet location

Not be confused with the extraction of locative references from tweets, approaches that target tweet location focus on where the tweet was posted using, typically, tweet geotagged metadata and/or user profile information (Sakaki et al., 2010; Priedhorsky et al., 2014; Chong & Lim, 2018; Gao & Li, 2019; Gonzalez-Paule et al., 2019; Khodabandeh-Shahraki et al., 2019). For example, Gonzalez-Paule et al. (2019) proposed a geolocation model that targets non-geotagged tweets by exploiting similarity content of geotagged tweets for traffic incident detection purposes. Sakaki et al. (2010) presented a spatiotemporal algorithm that locates disaster-related events from tweets using tweet geotagged data and user profile information. Khodabandeh-Shahraki et al. (2019) proposed a model for event geolocation that takes into account multiple variables such as tweet text, user profile location, geotagged data, and posting time to estimate the location of a particular event. They noted that location references in tweet text are not

always a reliable variable to predict the location of an event. Gao & Li (2019) introduced a geolocation model for English microtexts such as tweets that estimates the location of the microtext using a weight probability model, where words or phrases that express location are given a higher weight than others. Izbicki et al. (2019) presented a multilingual model based on a neural network that exploits character-level features in tweet text to estimate tweet location without, theoretically, drawing on linguistic-based features (i.e. tokenization, POS tagging, etc.), and achieving great results. The authors suggest that their model can transfer the knowledge learnt in language to at least more than 100 languages, and that their training dataset contained at least 900 million tweets. Unfortunately, we are not explicitly provided with textual evidence of its capabilities, or an in-depth explanation of how location is captured across many languages, and what types of location can be extracted. Elad et al. (2020) devised a system that estimates the location of a tweet on the basis of an ad-hoc list of location types using linear classifiers and neuronal networks to asses which of these probabilistic-based frameworks performs best.

2.8.1.3. Locative reference extraction

The focus of the present thesis deals with the identification and extraction of locative references that are mentioned in the tweet text. Usually, tweet texts related to emergency-related situations contain geospatial information more relevant to the locus of the event than to user location in such scenarios (Arthur et al., 2018). Because of that, tweet text is a very valuable source of information in emergency-tracking systems to locate emergency-related events. A comprehensive case-by-case analysis is given for each location-detection model in chronological order, with an emphasis on the most impactful works in this area.

Gonzalez et al. (2012) introduced TweoLocator, a framework that focuses mainly on location mentions in tweets to infer user location patterns. Given a tweet text, the model applies a set of heuristics that take into account n-grams, different gazetteers (i.e., Wikipedia and GeoNames), and rule-based techniques to detect locative references, as well as different syntactic combinations to estimate user location. The location types considered are geopolitical entities and POIs. However, it presents some limitations, such as the restriction of n-gram size to trigrams, or considering only one location mention in the tweet text to infer user location, ignoring many other potential location mentions. Given their focus on user location and geocoding, no evaluation metrics are offered to test their location extractor.

Gelernter & Balaji (2013) proposed a microtext location-detection model using regex-based rules, the Open Calais NER software, ML techniques for abbreviation disambiguation, and NEM with a National Geospatial Intelligence Agency gazetteer for the identification of places in New Zealand and Australia at and within city level such as administrative places, buildings, and streets. It was, to the best of our knowledge, the first research work in the literature of locative reference extraction from tweets that made explicit use of linguistic knowledge for its

regex-based module. The authors created a training dataset with tweets from the 2011 Christchurch earthquake and two evaluation datasets, one about this event, and another for the 2011 Texas wildfire in the US. In the first evaluation dataset, the algorithm achieved an F1 score of 0.85 for streets, 0.86 for buildings, 0.96 for administrative places, and 0.88 for place abbreviations, giving an average of 0.9. In the second dataset, it obtained an F1 score of 0.71. The reason why this algorithm achieved a very high F1 score for the first dataset is explained by the fact that the training dataset and test dataset shared the same emergency event and also the gazetteer was preloaded with locative references from New Zealand. In other words, the model may have been particularly skewed for the location types mentioned in the Christchurch earthquake. The F1 score of the Texas fire evidenced that in other events the algorithm may suffer from poorer performance. On that note, it has been reported that case studies of particular disaster events with well delimited spatial boundaries usually yield higher evaluation results (Karimzadeh et al., 2019). It would thus remain to be seen whether such greatly high performance could be replicated with global-scale events or local events other than those that may occur in New Zealand. Another issue has to do with the lack of explicit information about the type of evaluation conducted: were precision, recall and F1 scores obtained in terms of entity-based matching or token-based matching?[3] If the latter, evaluation numbers tend to correlate higher. Another downside of this study has to do with the building and street identification module. Although the authors constructed a rich list of building types from the Wikipedia[4], they decided to create an ad-hoc list of traffic ways with few references: *st, street, ln, lane, dr, drive, boulevard, blvd, road, rd, avenue, ave, pl, way, wy*. Despite being a valid method, its scientific rigor might be debatable.

Lingad et al. (2013) aimed at investigating the effectiveness and the accuracy of existing NER tools in recognizing location mentions of type geopolitical entities, natural landforms, and POIs from tweet text using the pre-defined NER categories LOCATION and ORGANIZATION. They compared the performance of these out-of-the-box NER tools against their retrained counterparts, and found that existing NER tools such as Stanford NER, once retrained with tweet data, can yield great overall NER performance, outperforming even Twitter-based NER tools (e.g. Twitter NLP). For that purpose, they compiled and annotated a gold standard dataset of 3203 disaster-related tweets from 2010 to 2012. This training dataset comprised tweets from the 2012 flooding in Queensland (Australia), the 2011 earthquake in Christchurch (New Zealand), the 2011 England riots, the 2012 flooding in York (England), and the 2012 Hurricane Sandy (US). Evaluation measures were provided on a per-token basis, rather than on a per-entity

[3] A more detailed explanation about different types of NER-specific evaluation criteria is given in the Section 7.
[4] https://en.wikipedia.org/wiki/List_of_building_types

basis, for out-of-the-box NER tools, and their retrained counterparts considering or not hashtags. The best F1 score, 0.902, was achieved by the retrained Stanford NER tool.

Daly et al. (2013) focused on the study and detection of traffic-related incidents from tweets and SMS messages. In their model, a georeferencing module was developed to extract locative references and assign them coordinates. For the extraction of locative references, they used the OpenStreetMaps geodatabase for an n-gram-based lookup of location names. Despite a lack of an explicit characterization of the location types, their focus was on POIs as well as traffic ways, the latter extracted by means of rule-based methods. No evaluation metrics were provided for the location-extraction module but for the accuracy of the assigned coordinates.

Ghahremanlou et al. (2014) devised OzCT Geotagger, a geoparsing algorithm that targeted toponym recognition and toponym resolution. It can automatically find location mentions at a fine-grained level (streets, suburbs…) from tweet text using NEM (gazetteer-matching with a lexicon and GeoNames, and Google Maps API querying) and the Stanford NER tool before linking them to geographic coordinates. Each tweet was classified into definite, ambiguous or no-loc for unambiguous geospatial information, ambiguous geospatial information and non-existent geospatial information, respectively. The authors collected tweets for their training and test datasets from 2012 natural disaster events in Australia, obtaining an F1 score of 0.804 for the detection of definite locations.

Han et al. (2014) focused on the detection and classification of location mentions (countries, cities and POIs) in tweet data by training their own CRF classifier and re-engineered a supervised ML model on the basis of previous feature-engineered tools such as Stanford NER, TwitterNER and Washington NER, achieving an F1 score of 0.7261.

Malmasi & Dras (2016) proposed a linguistic-based unsupervised location-detection model based on linguistic techniques and rules such as NP extraction and n-gram matching techniques using regex-based rules and GeoNames. It targeted geopolitical entities, POIs, addresses, and surrounding distance and direction markers, giving an F1 score of 0.792. This research work provided a more linguistic-based focus for the task of location detection. However, there are a number of drawbacks that need to be discussed: first, the debatable rigor in the authors' decision to create ad-hoc lists of location-indicative words (addresses, POIS…) and second, a loose evaluation metric standard performed on a per-token basis, rather than on a per-location entity basis, both of which might have contributed to a higher F1 score.

Inkpen et al. (2017) presented an ML-based model with a CRF classifier for the detection of US and Canada geopolitical entities (cities, provinces/states, and countries) in tweet text with the help of NEM using GeoNames for business and marketing purposes. For the training phase of their algorithm, they employed linguistic features such as POS tags, contextual grammatical information, and GeoNames membership. They reported evaluation metrics on a per-token and

per-entity basis for each type of geopolitical entity. The best F1 score for the entity-based evaluation at city level was 0.81, at state and province levels 0.86, and at country level 0.90.

Middleton et al. (2018) developed a rule-based location-detection model for English tweets using the OpenStreetMaps database. The location-detection model was based on previous work (Middleton et al., 2014) with additional improvements. It has an important linguistic component in that it used NLP techniques such as the NLTK sentence tokenizer[5], entity-based matching with OpenStreetMaps using an n-gram-based module, their own corpus of building and street types, and the NLTK stopword list enriched with a list of names. They focused on geopolitical entities, buildings, and streets. The evaluation stage was carried out for separate corpora of tweets about different incidents (i.e. blackout, earthquake, and hurricane) in different geographic areas (i.e. Christchurch, Milan, New York, and Turkey) for which the geodatabase was preloaded with locations for those specific areas for the evaluation of each corpus. Precision numbers were impressive, ranging from 0.93 to 0.99, and F1 scores ranged from 0.90 to 0.97, except for the Turkey earthquake dataset where it achieved an F1 score of 0.28. They also compared the performance of other microtext location-detection models, such as that of Gelernter & Balaji (2013) or one that used Stanford NER with regex-based patterns for matching sequences of nouns. The first one achieved an F1 score of 0.27, 0.67, 0.55, and 0.66 for the Turkey earthquake corpus, the New York hurricane corpus, the Milan blackout corpus, and the Christchurch earthquake corpus, respectively. The second one achieved lower F1 scores, the best one being 0.52 in the Milan blackout corpus. As the authors, noted, a disadvantage of their model is its very slow processing speed, since it has to preload many locations in memory before deploying the location-extraction module, lasting many minutes. Overall, the authors highlighted the importance of implementing linguistic knowledge and using geodatabases in location extraction from tweets to achieve great results. It would be interesting to see whether the application of their model to global-scale corpora of tweets about different issues and targeting more location types delivers the same results, and how processing speed becomes affected.

de Bruijn et al. (2018) built TAGGS, a model specifically designed for toponym recognition and resolution purposes, which exploits metadata and contextual geospatial information from disaster-related clusters of tweets, instead of individual tweets only. TAGGS was used to research flood locations at coarse-grained levels (i.e. geopolitical entities such as towns, cities, countries, regions, etc.) from 55.1 million flood-related tweets in twelve languages extracted in real time from the Twitter API using flood-related keywords. TAGGS first performs toponym recognition with NEM using n-grams and GeoNames, and then performs toponym resolution

[5] NLTK is a Python library for NLP tasks. Further information about NLTK is given in Section 7, in which we used the NER module in NLTK to benchmark its performance against LORE.

thanks to Twitter metadata, such as GPS coordinates, and geospatial clues from individual tweets and clusters of tweets. TAGGS achieved an F1 score of 0.865.

Hoang & Mothe (2018) explored how the combination of different models and methods developed in NLP to perform location extraction on tweets (e.g. Ritter's NER tool, Gate NLP and Stanford NER) could help improve recall and precision. Then they used DBpedia[6], a knowledge base, to filter locative references. They also proposed a location-prediction system which involved selecting only location-specific tweets for improved location-extraction performance. This system used the third-party NER tools Ritter NER, Gate NLP, and Stanford NER together with NEM using Gate NLP framework's gazetteer and linguistic knowledge through leveraging locative prepositions and location-indicative words. The best combination consisting of Ritter NER, Stanford NER, and DBpedia yielded an F1 score of 0.85 in the Ritter test dataset (Ritter et al., 2011).

Al-Olimat et al. (2018) proposed an unsupervised location-detection model for tweet text drawing on NEM (GeoNames) with gazetteer augmentation and filtering and an n-gram model complemented by collocational information. It was applied on three tweet datasets corresponding to three local flood events in Chennai, Louisiana and Houston respectively, achieving an F1 score of 0.81 on a per-token evaluation basis. However good the results are, we are not provided with an explanation about the location types extracted by their model.

Dutt et al. (2018) developed an unsupervised location-detection model for tweets based on regex-based rules, ad-hoc lists of location-indicative words, syntactic chunking and dependency parsing, the Spacy NER tagger[7], and GeoNames that achieved an F1 score of 0.81 on a per-entity-based evaluation. It was applied to a large test corpus of tweets (239,256 tweets) collected using the keywords *dengue* and *flood* for emergency-related events of those types located in India. The methodology followed is linguistic-based, since they made use of linguistic knowledge and NLP techniques for NER and NEM. The authors did not present information about the location types extracted by their model.

Avvenuti et al. (2018) devised GPS, a geoparsing and geotagging tool drawing on ML techniques that uses semantic annotation and entity linking with knowledge-based resources (i.e. RDF-based resources). It can detect location mentions at a fine-grained level and then disambiguate them with geographic coordinates, obtaining an F1 score of around 0.738 for English tweets and 0.885 for Italian tweets.

Karimzadeh et al. (2019) presented GeoTxT, a scalable geoparsing tool that detects and disambiguates global-scale location mentions in unstructured text, with the help of six implemented third-party NER tools. For the toponym-resolution phase, they leveraged the

[6] https://wiki.dbpedia.org/

[7] SpaCy is a Python library for NLP tasks. Further information about this library and its NER module is presented in Section 7.

GeoNames database. Apparently, they targeted geopolitical entities, natural landforms, and some POIs (i.e. buildings), which are the location types typically recognized by NER tools. The NER system CogComp alone achieved the best F1 score of 0.7854 on a per-entity-based evaluation.

Kumar & Singh (2019) tackled the issue of tweet-location extraction in several earthquake events by means of a supervised DL-based approach using a CNN algorithm without linguistic-based feature engineering. The per-token-based evaluation achieved an F1 score of 0.96. The authors did not provide a sound theoretical basis of what location types were targeted by their model.

Hernandez-Suarez et al. (2019) proposed a NER system for detecting and geocoding toponyms (i.e. street, avenue, country, region, building) in Spanish tweets from the 2017 Mexico City earthquake through a DL-based model based on a bi-LSTM neural network with a CRF top layer, and pre-trained word embeddings using the corpus of Spanish tweets as training data. Their model achieved an F1 score of 0.80.

Di Rocco et al. (2019) introduced a knowledge-driven model for the location detection and geocoding of sub-city level locative references using LinkedGeoData, a semantically enriched version of OpenStreetMaps, and openStreetMaps Facet Ontology. The algorithm performs entity-based matching with the geographic databases and then extract the geospatial coordinates attached to the location names extracted. The evaluation, centered on the accuracy of the geocoding part, was performed on two datasets, GeoText (Priedhorsky et al., 2014) and FollowTheHashtag (Yuan et al., 2015), using only tweets geolocated in New York City (US) and London (UK).

Y. Zhang et al. (2019) presented a probabilistic-based framework based on coarse-grained syntactic knowledge that automatically learns syntactic patterns to discover locative references in abnormal traffic events. They evaluated their model using tweet datasets containing traffic incidents in New York and Los Angeles, and compared their results against Google Named Entity Detection[8], Stanford Core NLP, and spaCy. The F1 scores ranged from 0.6 to 0.7 using distinct evaluation procedures, outperforming the previously mentioned NER tools.

Yang-Lim et al. (2019) introduced TEXT, a rule-based location-detection model that only focused on the extraction of traffic ways from 1500 English tweets, comparing their system against other generic NER systems such as Stanford NER or NLTK. TEXT outperformed all of them by a large margin, achieving an F1 score of 0.9128 for the task of extracting traffic ways. To the best of our knowledge, this is the only piece of work in the literature that addressed this specific location type.

[8] It is an API web service that makes use of Google Natural Language Cloud capabilities. Further information about it and its NER capabilities is provided in Section 7.

Xu et al. (2019) devised a DL pipeline with a bidirectional LSTM–CRF network for location detection and disambiguation of locative references in tweets, achieving an F1 score of 0.80. Although there is no explicit mention of using linguistic features for the training phase of the model, feature engineering was performed with character-level information, pre-trained embeddings, and with a gazetteer of POIs to label location mention candidates.

Das & Purves (2019) presented a hybrid location-detection system consisting of a supervised-learning algorithm, using OpenNLP and Stanford NER with retrained data, and a rule-based module for the detection of traffic-related locations in Greater Mumbai (India) in the context of traffic-event location detection. The rule-based module infers locative references on the basis of the presence of locative prepositions, obtained from a database, and location-indicative words. No information is provided as to which criteria were used in the process of constructing a location-indicative word dataset. They retrained each of the NER systems with their own data and tried several combinations to obtain the best performing model. As the authors explained, they achieved a fairly poor precision score (0.53) but high recall (0.79) which is due to a high number of false positives because of the functioning of the rules together with the location-indicative noun dataset.

Singh et al. (2020) provided an in-depth study of the current coronavirus COVID-19 pandemic in which they also focused on locative references mentioned in tweets dealing with the COVID-19 outbreak. They used NEM with Wikipedia and Statoids[9], two major databases to extract geopolitical entities such as countries, states, provinces, and cities. With this geospatial information, they analyzed the correlation between confirmed number of cases in different regions of the world and number of location mentions in the tweets, finding a high correlation between both: the more confirmed cases of coronavirus in a given area, the more that area appeared mentioned in the tweets. Singh et al. (2020) underlined the importance of location extraction techniques to study the evolution and spread of pandemics and for disease forecasting.

Wang et al. (2020) built a location extractor called NeuroTPR, which used a bidirectional RNN with LSTM enriched with linguistic-based features for the task of location extraction from tweets. The process of feature engineering was carried out using character embeddings, word embeddings, and linguistic-based features such as POS tags and deep-contextualized word embeddings. The tagging scheme adopted in the datasets was the IOB, which is the typical of NER tools. For the training phase, they employed 599 tweets from a dataset called WNUT 2017, together with automatically annotated location-related chunks from the Wikipedia that were split into 140-character chunks and purposefully introducing misspellings to make them resemble tweets. For the evaluation of their tool, they used different corpora. They built a tweet corpus from the 2017 Hurricane Harvey dataset; moreover, they used GeoCorpora (Wallgrün et

[9] http://www.statoids.com/

al., 2018), which is also made up of tweets and another dataset called Ju2016, with chunks from the Web. For the compilation of the *Harvey2017* corpus, they created a regex-based rule with 70 location-indicative nouns and abbreviations to mine 1000 location-rich tweets containing at least one locative reference. The location types were geopolitical entities, natural landforms, POIs, and a few traffic ways. They did not consider, in their definition of location, demonyms, metonymical references, and vague and unspecific location mentions. In the evaluation stage, they compared their model against standard, off-the-shelf NER tools such as Stanford NER, using the standard and caseless models, and a retrained model with the same training data used for NeuroTPR, spaCy NER, a basic biLSTM-CRF model and another DL model from the 2019 SemEval geoparsing competition, both using also the same training data as NeuroTPR. The evaluation phase was carried out using exact matching of location references. The best NeuroTPR model, consisting of 3000 Wikipedia articles and 599 tweets, achieved a precision score of 0.787, a recall score of 0.678, and an F1 score of 0.728 on the *Harvey2017* corpus. They tried to expand their training dataset by adding 50 tweets from the original 2017 Hurricane Harvey dataset to their training corpus, and, although evaluation numbers improved (i.e. precision score of 0.832, recall score of 0.843, and F1 score of 0.837), they pointed to the fact that the model might have suffered from overfitting. They noticed that, for training NeuroTPR, using a different number of Wikipedia chunks which resembled tweets through the introduction of misspellings performed worse (i.e. F1 scores lower than 0.5) than using those chunks untouched. They also noticed that adding more of these raw Wikipedia chunks did not improve the performance of the model but actually worsened it. When comparing the best NeuroTPR with the other models and tools, they noticed that the standard Stanford NER was not successful in extracting fine-grained location types such as POIs and traffic ways, although it showed great precision numbers (0.828), and much better evaluation numbers than the caseless or the retrained models. SpaCy obtained much worse numbers (i.e. F1 score of 0.366), whereas the basic biLSTM-CRF model and the 2019 SemEval geoparsing model achieved evaluation numbers closely resembling those of NeuroTRP (i.e. F1 scores of 0.649 and 0.703, respectively). With GeoCorpora, NeuroTPR achieved a precision score of 0.8, a recall score of 0.761, and an F1 score of 0.78. In the discussion of the errors committed by their tool, they pointed out that highway names, especially US interstates (e.g. *I-45*) were not captured by their tool, or that concatenated location names at the end of tweets were not properly delimited and captured. For the first type of error, they suggested using regexes. Overall, many of the location types targeted by this model and encountered issues were already addressed by LORE and nLORE since the inception of the present research project. Wang et al., (2020) showed that the topic of location detection from tweets is a very active line of research gaining momentum in the last few years and months.

2.8.2. Classification of Twitter-based geolocation systems in terms of method

According to the type of model, location-detection systems can be based on probabilistic models, such as ML or DL (Cheng et al., 2010; Sakaki et al., 2010; Lingad et al., 2013; Yin et al., 2014; Ghahremanlou et al., 2014; Han, Cook, et al., 2014; Han, Jimeno-Yepes, et al., 2014; Inkpen et al., 2017; Avvenuti et al., 2018; Miyazaki et al., 2018; Chong & Lim, 2018; Xu et al., 2019; Gonzalez-Paule, 2019; Hernandez-Suarez et al., 2019; Wang et al., 2020), symbolic-based or rule-based models (Gelernter & Balaji, 2013; Malmasi & Dras, 2016; Al-Olimat et al., 2018; Dutt et al., 2018; Middleton et al., 2018; Yang-Lim et al., 2019) or a combination of both (Hoang & Mothe, 2018; Das & Purves, 2019; Y. Zhang et al., 2019).

2.8.3. Classification of Twitter-based geolocation systems in terms of data and resources

Another criterion is related to which type of Twitter data is used for a geolocation model. For instance, many rely on tweet text (Lingad et al., 2013; C. Li & Sun, 2014; Ghahremanlou et al., 2014; Han, Jimeno-Yepes, et al., 2014; Malmasi & Dras, 2016; Ikawa et al., 2016; Inkpen et al., 2017; Avvenuti et al., 2018; Middleton et al., 2018; Miyazaki et al., 2018; Dutt et al., 2018; Y. Zhang et al., 2019; Das & Purves, 2019; Hernandez-Suarez et al., 2019; Karimzadeh et al., 2019; Wang et al., 2020), whereas others rely on tweet geotagged metadata only (W. Li et al., 2011), user profile information and the user's tweet history (Cheng et al., 2010; Alex et al., 2016; Chong & Lim, 2018; Mourad et al., 2019) or a combination of the previous data (Sakaki et al., 2010; Kinsella et al., 2011; Dredze et al., 2013; Han, Cook, et al., 2014; Yin et al., 2014; Gonzalez-Paule, 2019).

3. RESEARCH CHALLENGE

3.1. Research issues and limitations

There are some research issues and limitations that should be taken into consideration:

i) Lack of a heavy linguistic basis: although there is some NLP-based research in location-detection models that makes use of tokenization, POS tagging, n-grams, some basic regex-based rules and different lexical lists and gazetteers, few, if any, provide a sound, linguistic-focused theoretical basis of what a locative reference is, how it can surface in the clause, and which linguistic-based rules, on the basis of linguistic evidence, can be devised to enable their identification and avoid the extraction of wrong instances. Most models follow a result-oriented approach that potentially ignores the underlying linguistic knowledge encapsulated in texts.

ii) Lack of sufficient semantic granularity: as of now, most location-detection systems lack a sufficiently fine-grained semantic coverage of location types. In this sense, many still rely on coarse-grained location types such as geopolitical entities and a few natural landforms (Wang & Hu, 2019), or a few roads and POIs (Gelernter & Balaji, 2013).

iii) Geo/non-geo ambiguity in text (Amitay et al., 2004) harms location-detection models, especially those that make use of NEM approaches. The granularity level in those models have a direct impact on the level of ambiguity that might surface. In this regard, country-level toponyms are easier to identify correctly than other geopolitical entities, which, in turn, are easier to identify than other location types.

iv) Most research has focused on case studies of particular disaster-related events with well-delimited spatial boundaries. This facilitates the training and testing phases of their models, since the scope of locations is greatly reduced and much restricted. Those approaches using NEM frameworks especially benefit from local-scale events. It thus remains to be seen whether performance on different global-scale crisis and emergency-related scenarios remains equally good.

v) There are no clear, strict guidelines in the evaluation phase of location-detection models: some elaborate on the per-token-based and per-entity-based statistics, others only offer the evaluation numbers without a critical assessment of what is measured and how. This potentially compromises the validity and interpretability of the evaluation statistics offered.

vi) Real-time application of these tools is a chimera in most cases, since the processing speed of most location-detection models is overall slow, especially when they rely on huge geographic databases such as OpenStreetMaps.

vii) Challenges and research directions in corpus construction (Wallgrün et al., 2018): there is an ongoing body of research that studies the relationship between the degree of success of a given location-detection model and corpus size and type. In this respect, probabilistic-based models require a tremendous amount of corpus data to train their algorithms.

viii) Linguistic-based feature engineering in ML and DL approaches for location detection mostly relies on token, POS tag and embedding features at best and at worst such linguistic features are not even used in these models. Whether extended linguistic-based feature engineering provides a real benefit or not in a DL-based location-detection framework remains to be addressed.

3.2. Research questions

These are the questions that guided our research:

i) Can we build a location-detection model for tweets with a heavy linguistic basis that can exploit linguistic knowledge using NLP-based techniques only? If so, can it detect any fine-grained location types? Can it provide almost instant, real-time results? Can it be applied to languages other than English so that it can be adapted to multilingual contexts? Can this multilingual adaptation be facilitated with semi-automatic methods?
ii) Can we use our model with any type of crisis-related or emergency event on a global-scale and can it perform well on a regular basis? Can we ensure that our model can generalize well with new, unseen corpora of tweets?
iii) Can we train a successful probabilistic-based model that relies on a DL algorithm using a relatively small corpus? If so, can we prove that linguistic-based feature engineering can still play a decisive role in cutting-edge computational NLP approaches? Can this probabilistic-based model be more intelligent and thus have greater performance in the task of location extraction than its rule-based counterpart?

3.3. Research hypothesis and justification

Our initial hypothesis is that, by exploiting the explicit linguistic and contextual knowledge in microtexts, we can build a location-detection model which (a) can detect any fine-grained location type in English, Spanish and other languages almost instantly, (b) which can perform on a global scale in any kind of emergency or crisis scenario, and (c) which achieves state-of-the-art performance without the high computational cost, time and resources characteristic of ML and DL frameworks. Another hypothesis that resulted from our foray into AI and Computational Linguistics is whether we can prove the importance of linguistic-based feature engineering in cutting-edge NLP computational approaches, by developing a DL model that feeds off the linguistic knowledge provided by our rule-based method. In relation to this, another research hypothesis revolves around the question whether our DL model can outperform our rule-based system. In this regard, the present research project meets the current needs of research development in digital humanities, interdisciplinarity among linguistic-related computational disciplines and real-world practical applications that have a direct impact on society in the resolution of logistical problems in any kind of crisis-related or emergency situation.

4. OBJECTIVES

The goals that we intend to reach in this research are as follows:

i) Our primary aim is to develop an original and innovative multilingual, linguistically-aware, fine-grained location-detection model that can capture any kind of location type from tweets in English, Spanish and other languages through rich linguistic knowledge for its subsequent implementation into CASPER, a multi-domain problem detection system for tweets that targets environment-related issues. This system could ultimately be of great help for emergency-based services and responders for the detection of locations in not only environment-related problems but also any kind of real-world incidents and issues, such as car accidents, pandemics, or terrorist attacks.

ii) Another primary aim is the development of a DL model that exploits the linguistic features provided by the previous model to automatically learn and infer linguistic patterns in the extraction of locative references from English tweets. This is a proof-of-concept implementation which could pave the way for future computational work in the field of NER, while showing the potential capabilities of linguistic knowledge in cutting-edge computational approaches.

iii) One specific objective, in relation to the previous ones, is the compilation of different and representative enough corpora of tweets in different languages for the development, building, training, and evaluation phases of the models.

iv) Another specific objective is to test the models in the evaluation stages with the previously compiled evaluation corpora that represent real-life scenarios of crisis-related and emergency events. In this light, we wanted to check (a) whether LORE excelled in the identification of locative references with respect to other state-of-the-art, off-the-shelf NER tools, (b) whether the performance of LORE can remain stable and unchanged with other evaluation corpora, and (c) whether nLORE, the DL counterpart of LORE, can achieve greater performance than LORE.

5. MODEL AND METHODOLOGY

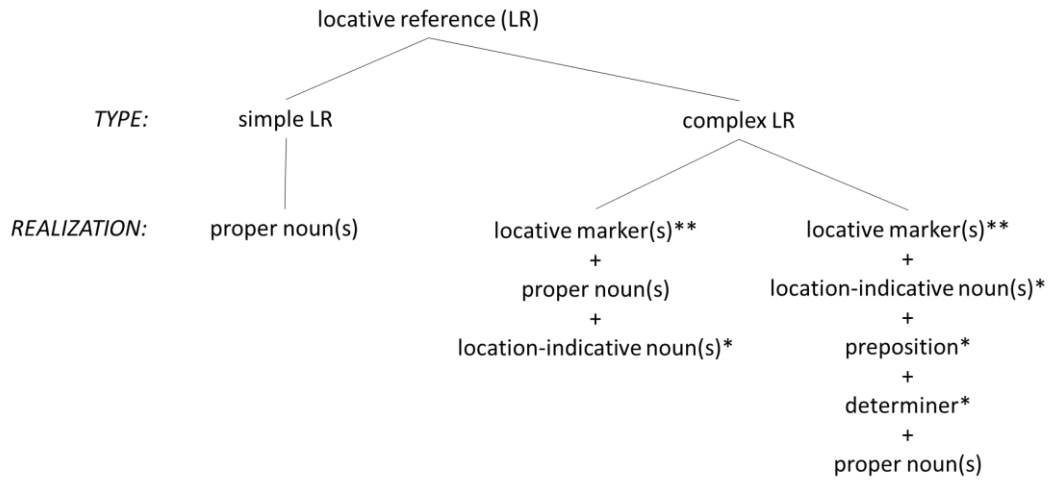
In this section, we present the methodology used in the development and deployment of multilingual LORE and English-based nLORE, and the datasets used in these processes, that is, the development corpus (dev corpus), the test or evaluation corpus (eval corpus), training corpus (train corpus), and validation corpus (valid corpus), which became essential components in the building and evaluation phases of the location-detection models. Before doing so, it becomes necessary to provide a thorough definition of what we mean by locative references, and their formal and semantic boundaries. Afterwards, we present the corpus compilation phase, accompanied by several tables and figures that offer some statistics regarding the nature of the

corpora compiled. Then, we explain the steps carried out in the development of multilingual LORE, the lexical datasets used, and the modular architecture of LORE. Afterwards, we introduce the DL implementation of LORE for English, nLORE, from the basics of neuronal networks to the details of the specific type of neuronal networks and word embeddings used and the linguistic-based feature-engineering process.

5.1. Formal, semantic and structural boundaries of locative references

A locative reference, location entity or location mention is a subtype of named entity that designates a specific, unambiguous, and precise physically-locatable geographic reference, i.e. one that can be typically rendered into geographic coordinates or other geospatial measurements and thus pinpointed on a map (Leidner & Lieberman, 2011; Gelernter & Balaji, 2013; F. Liu et al., 2014; Gritta et al., 2018; Hoang & Mothe, 2018). In linguistic terms, locative references are typically proper nouns that designate named entities of place (i.e. toponyms or geographical names). Attending to their morphology, locative references can be realized as full words (e.g. *Bruxelles, city of London, costa del Levante*), abbreviations (e.g. *FR, bcn*), acronyms (e.g. *UK, US*), alphanumeric codes (e.g. *M-40*), or also as a combination of them (e.g. *I-90 SW, NE of Budapest*). As for their semantics, we establish five main locative categories: geopolitical entities (e.g. *New York*), natural landforms (e.g. *Mount Everest, río Pisuerga*), POIs (e.g. *Victoria Coach Station, musée du Louvre*), and traffic ways (e.g. *110 Croydon Road, I-290, Rue Adolphe-Thiers*). From a structural point of view, we distinguish between simple and complex locative references according to the number and complexity of lexical units that make up one locative reference. In this respect, a simple locative reference is composed of one or several proper nouns (e.g. toponyms such as *Granada, United Kingdom*), whereas a complex locative reference offers a rich lexical network by means of the juxtaposition of location-indicative nouns and locative markers to the proper noun: e.g. *Lake Michigan, Meseta Central, Quartier du Marais,* or *25 miles NW of London*. Taking into account the surrounding lexical elements that comprise locative references in the form of location-indicative nouns and/or locative markers offers more detailed geospatial information (Van et al., 2013), which could ultimately be more useful for competent authorities to trace the location of a given emergency or persons affected by emergency-related scenarios. To illustrate the complexity of locative references, Figure 8 presents the phrasal structure of locative references, where the asterisk is used to mark optionality, and double asterisk refers to the optional presence of locative markers either at the beginning or at the end of the locative reference.

Figure 8. The phrasal structure of locative references.



A fine-grained taxonomy is provided to show the semantic richness and variety of locative references. In this taxonomy, we offer a few examples of location-indicative words that may accompany proper nouns in complex locative references:

1. Geopolitical entities: country, state, region, province, city, town, kingdom, villa, ciudad, provincia, estado, región, pueblo, ville, pays, département…
2. Natural landforms: mountain, mount, ridge, volcano, valley, lake, river, shore, beach, park, canyon, meseta, río, golfo, cabo, islote, playa, cordillera, île, montaigne, fleuve, plaine…
3. POIs: building, museum, school, station, stadium, garden, café, tavern, hospital, court, theater, residence, zoo, casino, square, catedral, universidad, tienda, museo, teatro, hôtel, bâtiment, supermarché, gare, église…
4. Traffic ways (addresses, roads, highways): street, st, boulevard, blvd, avenue, av, alley, road, rd, highway, hwy, freeway, fwy, turnpike, tpk, calle, c., carretera, avenida, avda, callejón, ruta, rue, route, voie, I(-)n, M(-)n, SR-n (where n represents a given number), etc.

The wide majority of location-detection systems target coarse-grained locations of type (1) and (2) (Wang & Hu, 2019), though a few models have begun to consider (3) and (4) (Gritta et al., 2018). However, to the best of our knowledge, location types (3) and (4) have not been thoroughly addressed in any location-detection model yet. With regards to location type (4), location-detection models such as those in Gelernter & Balaji (2013), Malmasi & Dras (2016), or Middleton et al. (2018) mostly focus on prototypical traffic ways (e.g. streets and roads), and mostly talk about their ubiquitous presence, without providing ways or rules to extract them. Yang-Lim et al. (2019) seems to be the only study that focuses on the extraction of "location and traffic state" from tweets, but we could not find what they meant by these. Overall, a

comprehensive study of highways and other roads has been neglected in all models. Moreover, most works that focus on POIs, such as Gelernter & Balaji (2013), Malmasi & Dras (2016), or Zou et al. (2019), depart from ad-hoc lists of location-indicative nouns instead of retrieving those items from reliable and comprehensive lexical resources. We also provide a typology of locative markers according to their semantics:

- Distance marker: *4 Kms from Narok Town, 5miles from Dublin, 20 kilómetros hacia la ciudad de Atenas, 45 kilomètres de Paris*, etc.
- Directional markers: *East Coast of Honshu, east of Exit 55*, sur *de Portugal, ouest de la France, 20 km NW of Durrës,* etc.
- Movement markers: *southbound I-91, northbound J19, eb J19*, etc.
- Temporal markers: *1h away from London, 25min out of Melbourne, 5 minutos para la Gran Vía, 10 mins du parc de Bagatelle*, etc.

The following examples illustrate the taxonomy presented in Figure 8, and the possible combinations of proper nouns, location-indicative words, and/or locative markers which represent actual locative references:

- China, New York, Buenos Aires (proper noun(s))
- Sur de Madrid, 66km NW of Kota Ternate (locative marker [directional] + proper noun(s))
- 35 kilomètres de Bordeaux (locative marker [distance] + proper noun(s))
- 1h away from London, 25min out of Melbourne (locative marker [temporal] + proper noun(s))
- Hotel Park Villa, Sierra Nevada (location-indicative noun + proper noun(s))
- Province of Ontario (location-indicative noun(s) + preposition + proper noun(s))
- Costa del Sol, restaurant du Pelvet (location-indicative noun(s) + preposition + determiner + proper noun(s))
- Dyckman Street Station, Fox Valley Animal Referral Center (proper noun(s) + location-indicative noun(s))
- 5 minutos de la calle Mesones (locative marker [temporal] + location-indicative noun(s) + proper noun(s))
- I 95 NB (proper noun(s) + locative marker [movement])
- Francis Scott Key Brg SB (proper noun(s) + location-indicative noun(s) + locative marker [movement])

- 4kms from Narok Town (locative marker[distance] + proper noun(s) + location-indicative noun(s))

There is some controversy regarding the semantic nature of some locative references when they represent the people of that place (e.g. *US officials*), organizations (e.g. *New Orleans Police Department*), government units (e.g. *London Councils*), or events (e.g. *New Zealand mass shooting*) (F. Liu et al., 2014; Gritta et al., 2018). These are, according to Gritta et al. (2019), 'embedded literal toponyms' and 'embedded associative toponyms' that are nested within larger NPs. The difference between both of them lies in the fact that the later takes part in organization and government names. In these cases of attribute usage (Wolf et al., 2014), the locative reference accompanied by other nouns of events or organizations might not always correspond to the location of the action or event described in the tweet. Metonymic instances of locative references still possess, though very loosely, a locative meaning (e.g. *The US and Iran appear to have stepped away from the brink of full-blown conflict*). They usually refer to government units or sports teams (e.g. *Madrid played against Barcelona*). We only considered the former, i.e. government units, as cases of locative references, because government units may, though loosely, refer to actual locative events. In the case of sports team, locative meaning is almost non-existent. In other words, sports teams do not tell much about the location of the event (i.e. a match) but about the origin of the sports team; hence, we have excluded them in the annotation of our corpora, and count as false positives if matched by the model.

Geography experts consider all these instances as borderline cases of locative references (Wallgrün et al., 2018). Some location-detection models filter them out (F. Liu et al., 2014; Gritta et al., 2018), whereas others extract the locative reference from all these instances (Malmasi & Dras, 2016). We could argue that, though not explicitly referring to physical locations, the majority is to be fundamentally understood in terms of the locative reference alluded to. This means that we should not be categorical when dealing with these non-standard uses of locative expressions, since geospatial meaning still underlies these uses.

(1) *California parties trash. The DJ just said make some noise if u got earthquake insurance*

For instance, in Example (1) 'California parties' is an instance of 'embedded literal toponym' that represents an event, i.e. parties thrown in California. The location meaning might not hold for the entire clause but for the alluded metonymic instance. Thus, our location-detection model would mine the locative reference *California.* The same goes for parts of organization names, as long as they reveal rich geospatial information (Example (2)).

(2) *C. Sulawesi earthquake sends shocks across Makassar Strait - The Jakarta Post - Jakarta Post*

The Jakarta Post is an 'embedded associative toponym' that refers to an Indonesian newspaper organization whose headquarters is in Jakarta. In this regard, it makes sense to extract the locative reference *Jakarta* from that instance.

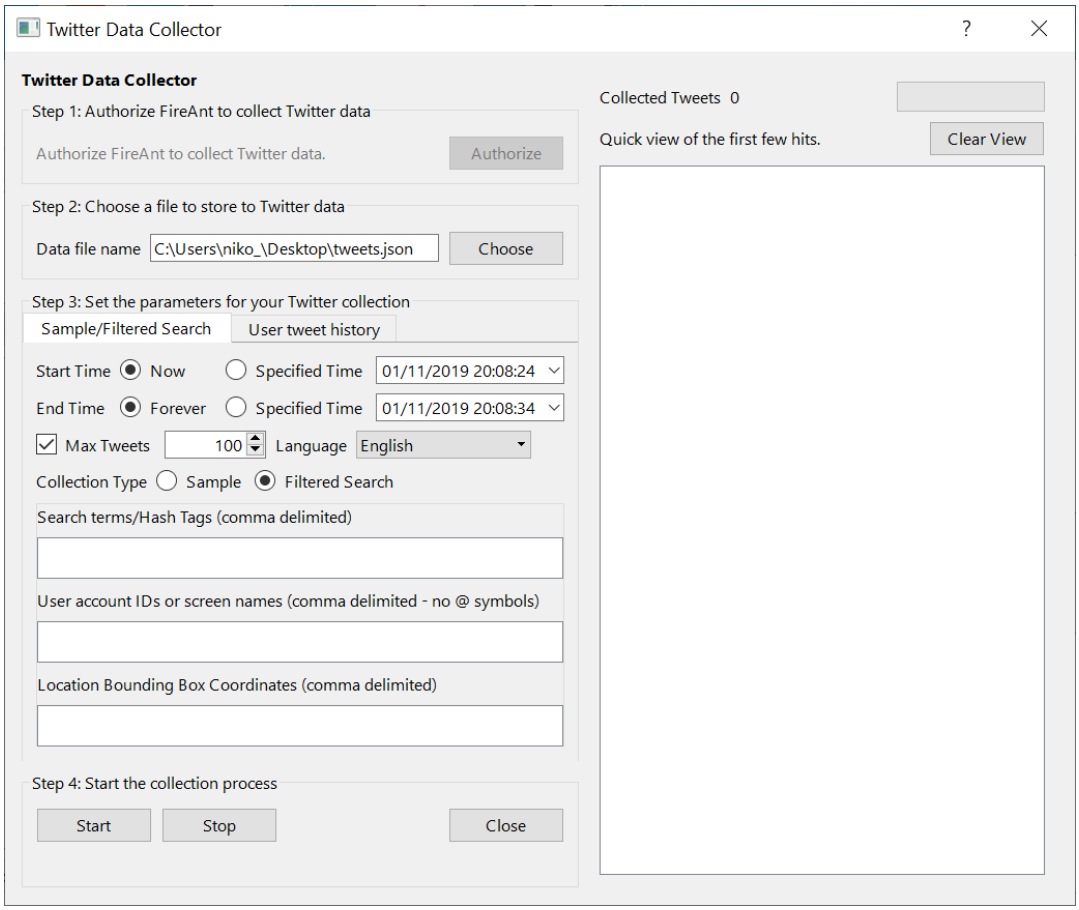
We also explain what we do not mean by locative references to ensure clear and crisp semantic and formal boundaries. To this end, we need to refer to commonplace or informal locative expressions (Herskovits, 1985; F. Liu et al., 2014). These are phrasal chunks in the clause that contain vague, ambiguous and unspecific geospatial information that usually appear in noun phrases containing common noun words or pronouns (e.g. *at home, in the garden, in front of you, on the street*), or in adverb phrases with co-referential adverbs (e.g. *here, there*). Since they are too unspecific and vague for crisis and emergency-related events and because they cannot be pinpointed on a map without any further contextual clue, we left them out. Other ambiguous cases such as demonyms (e.g. *Spanish citizen*) or adjectival modifiers (e.g. *Spanish olive oil*) were likewise discarded, except when these adjectival modifiers are followed by location-indicative words (e.g. *Iranian power plant*).

5.2. Corpus compilation phase

For the corpus compilation phase, we used the FireAnt app[10], a tweet collector application, to perform an automatic extraction of the corpora, using text-based data retrieval techniques (Yingjie Hu, 2018a).

Figure 9. The FireAnt app for tweet collection.

[10] Documentation details can be found on https://www.laurenceanthony.net/software/fireant/



In other words, for our automatic search of tweets we employed up to seven keywords related to crisis and emergency events which were *earthquake*, *flood*, *car accident*, *bombing attack*, *shooting attack*, *terrorist attack*, and *incident*, so that we could extract tweets mentioning issues of different nature. We used their near-equivalents *terremoto*, *inundaciones*, *accidente de coche*, *ataque terrorista*, *bombardeo*, *tiroteo*, and *incidente* for the corpus construction stage of the Spanish corpora. For French tweets, we used the keywords *séisme*, *tremblement de terre*, *inondations*, *coups de feu*, *attentat terroriste*, *attentat à la bombe*, *fusillade*, *accident de voiture*, *accident de la route*, and *incident*. Moreover, we strictly followed corpus linguistic principles in the compilation phase to comply with representativeness (Reppen, 2010). In this sense, our corpora are representative because they contain sufficiently enough tweets dealing with incidents, crises, and emergencies. In other words, since our interest is in studying and discovering locative references in microtexts, we expect the corpora to be composed of microtexts that have a sufficiently high number of locative references and that talk about different issues.

Since our interest was in raw tweet text only, we discarded tweet metadata. While the great majority of tweets contained one of these crisis-related keywords, it was the case that some tweets were repeated on multiple occasions, split into different lines, or empty. In this respect, we performed a pre-processing or pruning step for tweet text data which consisted in:

i) grouping multi-line tweets into a single line where each line represented one tweet by means of a regex that takes into account line breaks,

ii) removing retweets by means a regex that finds those retweets and discards it, and

iii) removing duplicates and very similar tweets through fuzzy matching algorithms.

Fuzzy matching algorithms calculate the degree of similarity among text strings. The ones that we used were the Levenshtein distance (Navarro, 2001) and the cosine distance using n-grams. The Levenshtein distance computes the minimum number of characters that have been added, deleted or replaced in a text string with respect to another, whereas the cosine distance using n-grams takes into account different combinations of n-grams in two strings to determine their degree of similarity. Table 2 below specifies which of these algorithms were used for each the different corpus. Further details about the computational implementation of this pre-processing step are provided in Section 6.1.

Our goal was to ensure that the majority of tweets were fairly unique and thus obtain a very representative dataset of unique tweets. The output of this pruning step resulted in the compilation of the corpora used for the models. Table 1 summarizes the main usages of the corpora employed in the present thesis.

Table 1. Corpus types and their usages.

Corpus type	**Usage**
Dev	Used to build a rule-based model
Train	Used to train a ML or DL model
Valid	Used to test the performance of the ML or DL model in the training process
Eval	Used to test the performance of the model

Table 2 presents the different compiled corpora in terms of their language, type, size, date of extraction, whether they were used in LORE and/or nLORE, and in which experiment they were used.

Table 2. Corpora's features.

Language	**Type**	**Size (tweets)**	**Date of extraction**	**Pre-processing algorithm**	**Model**	**Experiment**
English	Dev corpus	500	8 April 2019	Levenshtein distance	LORE	I
English	Eval corpus I	800	11 April 2019	Levenshtein distance	LORE	I
English	Train corpus	7000	17 November 2019, 30 November 2019,	Cosine distance with n-	nLORE	II

			1 December 2019, 2 and 9 January 2020	grams		
English	Valid corpus	1063	17 November 2019, 30 November 2019, 1 December 2019, 2 and 9 January 2020	Cosine distance with n-grams	nLORE	II
English	Eval corpus II	1372	5 January 2020	Cosine distance with n-grams	LORE and nLORE	II
Spanish	Dev corpus	100	28 May 2019	Levenshtein distance	LORE	I
Spanish	Eval corpus	500	27 August 2019	Levenshtein distance	LORE	I
French	Eval corpus	391	1 October 2019	Cosine distance with n-grams	LORE	I

Here, the reader may have noticed the distinct use of the labels ‘dev corpus’ and ‘train corpus’ in each of the models. Such distinction is grounded on the fact that, for our rule-based model LORE, we did not use a train corpus per se because LORE does not rely on data to train an ML or DL algorithm. The dev corpus was thus manually investigated for the study of locative references and for the extraction of linguistic patterns and cues that may help in the task of locative extraction. However, as opposed to LORE, for the training phase of nLORE we fed the neuronal network with the English train corpus. As can be seen, the English train and valid corpora are heterogeneous, with many incidents of different nature retrieved in different dates to capture incidents from as many places as possible. The reason why the Spanish dev corpus is five times smaller than its English counterpart is due to the fact that most of the modular blocks and stages of the models had already been developed with the English dev corpus. Moreover, we did not employ a dev corpus for French, since the capabilities of LORE only needed to be extended to French with semi-automatic methods.

For the evaluation stage, we conducted two experiments: in the first one, LORE is assessed with the English eval corpus I, the Spanish eval corpus and the French eval corpus against its competitors and, in the second experiment, LORE and nLORE are pitted against each other using the English eval corpus II. To avoid overfitting when training nLORE, the English eval corpus II was obtained on a different date, meaning it may have locative references different from the ones included in the English train and valid corpora.

Also, the format adopted by the corpora was subject to the nature of each experiment. In the first experiment, the corpora used for LORE followed a layout such that each tweet was

represented by a tweet ID indicating the number of the tweet, followed by the tweet text in the next column. A sample of the English dev corpus format is provided in Table 3.

Table 3. Sample of the English dev corpus.

Tweet ID	Tweet
01	Cleared: Incident on #GardenStateParkway NB at North of New Gretna Toll Plaza
05	RT @naqvi1966: Another incident of police harassment at street 13 of off-sunset boulevard Karachi. Reportedly the squad of AIG Karachi stop.
09	RT @califortia: California parties trash. The DJ just said make some noise if u got earthquake insurance
14	#M4 : Westbound : J33 Capel Llanilltern to J34 Miskin : Incident : Accident : Lanes closed : Delays #TrafficWalesAlert

The corpora used in Experiment II, the English train corpus, valid corpus, and eval corpus II followed a token-based tabular representation, where each row represents a given token, and each column indicates a given feature (i.e. original token, POS tag, presence in the place-name dataset, presence in the location-indicative noun dataset, locative marker) and label, following the BMESO tagging scheme. With BMESO, one-token locative references are labeled as S_LOCATION, two-token locative references are labeled as B_LOCATION and E_LOCATION, and those with more than two tokens are labeled as B_LOCATION, M_LOCATION, and E_LOCATION. If a token does not take part in a locative reference, the label used is O. Except the token and POS tag features, the other features were represented by means of Boolean values, that is, either 0 or 1: 0 when the feature is absent (e.g. not present in the place-name dataset) and 1 if the feature exists. To delimit tweets, an empty row is introduced to separate them. Such layout is the convention used in training probabilistic-based NER models. A sample of the English eval corpus II is provided in Table 4.

Table 4. Sample of the English eval corpus II.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Tag
Cleared	VBN	0	0	0	O
:	:	0	0	0	O
Incident	NN	0	0	0	O
on	IN	0	0	0	O
Frank	NNP	1	0	0	B_LOCATION
Lind	NNP	1	0	0	M_LOCATION
Roosevelt	NNP	1	0	0	M_LOCATION
Drive	VB	0	1	0	M_LOCATION
NB	NN	0	0	0	E_LOCATION
at	IN	0	0	0	O
58th	NNP	0	0	0	B_LOCATION
Street	NNP	1	1	0	E_LOCATION

One key part in the testing phases of our models was the annotation or labeling of locative references to have a ground truth or gold standard with which we can realistically assess the performance of our models. For the first experiment, we created a dataset of manually-annotated locative references related to their tweet ID which were retrieved from each of the dev and eval corpora. This dataset constituted our ground truth or gold standard to test the results generated by LORE and by the other NER tools. A sample of the gold standards of the English and Spanish dev corpora is shown in Table 5 and Table 6.

Table 5. Sample of the gold standard of the English dev corpus.

Tweet ID	Locative Reference
01	North of New Gretna Toll Plaza
01	Garden State parkway NB
05	off-sunset boulevard
05	street 13
05	Karachi
05	Karachi
09	California
14	M4 Westbound
14	J33
14	Capel Llanilltern
14	J34
14	Miskin
14	Wales

Table 6. Sample of the gold standard of the Spanish dev corpus.

Tweet ID	Locative Reference
39	Arrendario
39	Altares
39	Hermosillo
43	PANAMERICANA SUR 4
47	Perú
48	Yabucoa
48	San Lorenzo
50	BRASIL
53	zona 2 de Santiago Sacatepéquez
55	bodega de Haro
58	Puerto Rico
60	Nueva Jersey
61	Paraje el cerro del mono

As we can observe, for tweets without any locative reference, we did not include those. Also, when a given tweet contained two or more locative references, each of these was represented with the same ID number but in different rows. The reason behind using this layout lies in the fact this is also the format adopted by LORE in the output.

For the second experiment, the English eval corpus II (as shown above in Table 4), as well as the other corpora, was first semi-automatically labeled with the tags generated by LORE, and then manually inspected to correct any deficiency, that is, any missing or wrongly-delimited locative reference. In this way, we obtained our ground truth of locative references, but following the token-based tabular format. In both experiments for both models, we strictly followed the aforementioned definition and typology of locative references.

Table 7 presents the number of locative references for each corpus, respectively, in terms of n-grams, and Table 8 offers some statistics about the ratios of locative references per tweet in each of the corpora for each of the languages, where all corpora are overall very rich in locative references.

Table 7. Distribution of locative references in terms of n-gram size in the corpora.

	Experiment I					**Experiment II**		
	English dev corpus	**English eval corpus I**	**Spanish dev corpus**	**Spanish eval corpus**	**French eval corpus**	**English train corpus**	**English valid corpus**	**English eval corpus II**
No of unigrams	213	264	37	197	119	2702	178	376
No of bigrams	109	190	16	59	55	1430	101	176
No of trigrams	48	60	8	46	37	411	27	63
No of n-grams where n ≥ 4	13	23	3	19	13	259	16	29
Total	**383**	**537**	**64**	**321**	**224**	**4802**	**322**	**644**

Table 8. Corpora's statistics.

Corpus type	**No of locative refs.**	**No of tweets with locative refs.**	**Average of locative refs. per locative-rich tweet**	**Average of locative refs. per tweet**
English dev corpus	383	199	1.92	0.77
English eval corpus I	537	259	2.07	0.67
English train corpus	4802	2877	1.67	0.67
English valid corpus	322	188	1.71	0.3
English eval corpus II	644	351	1.83	0.47
Spanish dev corpus	64	40	1.6	0.64
Spanish eval corpus	321	215	1.49	0.64
French eval	224	143	1.57	0.57

Finally, Table 9, Table 10, Table 11, Table 12, Table 13, Table 14, Table 15 and Table 16 provide, whenever possible, a list for the 13 most frequent locative references found in the corpora.

Table 9. Most frequent locative references in the English dev corpus.

Locative reference	Category	Occurrences #
Iran	Geopolitical entity (country)	41
Edison station	POI (station)	4
Indonesia	Geopolitical entity (country)	4
Trenton station	POI (station)	4
Auburn	Geopolitical entity (city)	3
EB I-84	Traffic way (highway)	3
Fort Lauderdale	Geopolitical entity (city)	3
Garda	POI (headquarters)	3
Halifax	Geopolitical entity (city)	3
Halifax library	POI (library)	3
I 5 NB	Traffic way (highway)	3
New Zealand	Geopolitical entity (country)	3
Syria	Geopolitical entity (country)	3

Table 10. Most frequent locative references in the English eval corpus I.

Locative reference	Category	Occurrences #
Iran	Geopolitical entity (country)	20
India	Geopolitical entity (country)	11
Pulwama	Geopolitical entity (city)	6
San Bernadino	Geopolitical entity (city)	5
J18	Traffic way (highway)	5
Japan	Geopolitical entity (country)	5
M74	Traffic way (highway)	5
New Zealand	Geopolitical entity (country)	4
Grapevine	Geopolitical entity (city)	4
Pakistan	Geopolitical entity (country)	4
Sr4 E	Traffic way (highway)	4
Kingston	Geopolitical entity (district)	4
Balakot	Geopolitical entity (town)	4

Table 11. Most frequent locative references in the English train corpus.

Locative reference	Category	Occurrences #
Iran	Geopolitical entity (country)	91
Puerto Rico	Geopolitical entity (country)	74
US	Geopolitical entity (country)	59
Jakarta	Geopolitical entity (city)	54
India	Geopolitical entity (country)	42
America	Geopolitical entity (country)	41
Indonesia	Geopolitical entity (country)	39
Australia	Geopolitical entity (country)	39

Hong Kong	Geopolitical entity (region)	34
California	Geopolitical entity (state)	29
Venice	Geopolitical entity (city)	29
Israel	Geopolitical entity (country)	28
UK	Geopolitical entity (country)	27

Table 12. Most frequent locative references in the English valid corpus.

Locative reference	Category	Occurrences #
Texas	Geopolitical entity (state)	6
Japan	Geopolitical entity (country)	5
California	Geopolitical entity (state)	4
NY	Geopolitical entity (city)	4
America	Geopolitical entity (country)	3
Venice	Geopolitical entity (city)	3
US	Geopolitical entity (country)	3
London Bridge	POI (bridge)	3
Nnewi	Geopolitical entity (town)	2
Toronto	Geopolitical entity (city)	2
ENGLEWOOD	Geopolitical entity (town)	2
Mississipi	Geopolitical entity (state)	2
Morocco	Geopolitical entity (country)	2

Table 13. Most frequent locative references in the English eval corpus II.

Locative reference	Category	Occurrences #
Iran	Geopolitical entity (country)	40
America	Geopolitical entity (country)	15
Indonesia	Geopolitical entity (country)	12
Iraq	Geopolitical entity (country)	9
US	Geopolitical entity (country)	9
Hong Kong	Geopolitical entity (region)	9
Jakarta	Geopolitical entity (city)	8
LA	Geopolitical entity (city)	6
Us	Geopolitical entity (country)	6
USA	Geopolitical entity (country)	5
IN	Geopolitical entity (state)	5
New Zealand	Geopolitical entity (country)	4
Oklahoma	Geopolitical entity (state)	4

Table 14. Most frequent locative references in the Spanish dev corpus.

Locative reference	Category	Occurrences #
Puebla	Geopolitical entity (state)	6
Puerto Rico	Geopolitical entity (country)	4
San Lorenzo	Geopolitical entity (town)	3
Acatzingo	Geopolitical entity (town)	2
Cuota	Traffic way (highway)	2
MEX-150D	Traffic way (highway)	2
Perú	Geopolitical entity (country)	2
Autopista Puebla-Acatzingo	Traffic way (highway)	2

Table 15. Most frequent locative references in the Spanish eval corpus.

Locative reference	Category	Occurrences #
Madrid	Geopolitical entity (city)	42
Arganda del Rey	Geopolitical entity (town)	7
Arganda	Geopolitical entity (town)	7
M-40	Traffic way (highway)	6
Comunidad de Madrid	Geopolitical entity (region)	5
Valladolid	Geopolitical entity (city)	5
EEUU	Geopolitical entity (country)	4
Monterrey	Geopolitical entity (city)	4
Valdemoro	Geopolitical entity (town)	4
Borox	Geopolitical entity (town)	4
Fuenlabrada	Geopolitical entity (town)	3
Rivas	Geopolitical entity (town)	3
Nuevo León	Geopolitical entity (state)	3

Table 16. Most frequent locative references in the French eval corpus.

Locative reference	Category	Occurrences #
Rouen	Geopolitical entity (city)	11
France	Geopolitical entity (country)	5
Seveso	POI (factory)	4
Paris	Geopolitical entity (city)	3
Ligne H	POI (train station)	3
Inde	Geopolitical entity (country)	3
Vallée de la Marne	Natural landform (valley)	3
Finlande	Geopolitical entity (country)	3
usine Seveso	POI (factory)	3

Those locative references with a higher number of occurrences may be a more informative and credible source of information about the locus of an event (Middleton et al., 2014), which could in turn provide emergency-based services and responders with vital information for the deployment of effective aid and resources in the relevant areas. In other words, their frequency was, more than likely, indicative of emergency-related scenarios in those places.

5.3. LOcative Reference Extractor (LORE)

5.3.1. Development phase

In the development phase of LORE, we did not use a train corpus per se because our model does not rely on data to train an ML or DL algorithm. Instead, the reason why we departed from a dev corpus was to study and extract rich linguistic patterns. The extraction of rich linguistic patterns was carried out by paying special attention to the linguistic idiosyncrasies of the geospatial features of natural languages and the microtext genre, as discussed in Section 2.5 and Section 2.6, respectively. In other words, we thoroughly analyzed the different combination of

n-grams and the part-of-speech of tokens, the presence of locative-contextual clues such as locative prepositions, location-indicative words, and locative markers, which usually signal the presence of locative references. All this knowledge was materialized in the formulation of regexes that took into account the aforementioned linguistic variables. Through engaging in continuous evaluations in an 'iterative refinement process' of our rule-based approach (Barrière, 2016), the regexes had to be tweaked and fine-tuned to tackle natural language ambiguity and the noisy nature of tweets, up to their current high-performance state. This involved looking at, not individual errors or missed locative references, which could potentially lead to overfitting and ad-hoc decisions, but different error-prone patterns derived from poorly-defined regex-based rules. Our goal was to anticipate and prevent erratic behavior of the model when applied to any other corpus of tweets. Obviously, this process led to more restrictive rules than those elaborated at the initial stages. The extracted linguistic patterns and rules, though each language expresses locative relations in slightly different ways, could theoretically be applied to languages other than those supported by LORE. By means of semi-automatic methods, most language-specific resources (i.e. the stopword, place-name and location-indicative noun datasets) are retrieved from the Web, except the locative-marker dataset, which needs to be manually supplied. Table 17 provides a summary of the language-specific resources used in LORE. Further details about these language-specific resources are given below in Section 5.3.2.

Table 17. Language-specific resources in LORE.

Type	**Definition**	**Extraction and compilation process**	**Examples**
Stopword dataset	List of the most frequent words that appear in a given language. Sometimes it can also include a list of person names and surnames.	Semi-automatic	car, boy, the, table, John, Mary
Place-name dataset	List of place names, mostly toponyms, taken from a geographic database.	Semi-automatic	London, Vienna, Italy, New York City
Location-indicative noun dataset	List of location-indicative nouns that typically accompany place names, taken from knowledge bases such as WordNet.	Semi-automatic	beach, road, pub, school, avenue, museum
Locative-marker dataset	List of words that indicate direction, distance or time in phrasal locative patterns.	Manual	south, northwest, kms, miles, hours

Thus, the multilingual adaptation of the model to other languages did not start from scratch, and only required a few tweaks in the regex-based rules together with semi-automatic methods for the retrieval of lexical resources. These tweaks and modifications involved taking into account the linguistic peculiarities of Romance languages, which express spatial relations in different ways. For instance, Spanish geographical names start with the location-indicative noun(s) and

may incorporate different combinations of prepositions and determiners before arriving at the toponymic part (e.g. *Avenida de la Constitución*). Also, Spanish locative-marker constructions are different from the English ones. For instance, complex locative-marker constructions have a different lexico-grammatical profile (e.g. *XX mins away from ____ vs XX mins hasta/de ___*). Since these phrasal structures are also found in French and are grammatically encoded in the same way as in Spanish, multilingual support was extended to French using the same built-in regex-based rules, thus not needing a specific development phase for French.

5.3.1.1. A typology of the linguistic regex-based rules

We present a typology of the regex-based rules that exploit linguistic knowledge and contextual evidence for their application on any of the languages supported in LORE. We tested these rules with the English, Spanish, and French eval corpora, showing their effectiveness in the task of extracting locative references from tweets. We provide examples from the eval corpora to illustrate the strengths and weaknesses of our rules. Whenever the rules failed in the extraction of locative references, we provide an explanation to account for their faulty behavior, and suggest potential solutions for a future refinement process.

5.3.1.1.1. Rules for n-gram combinations of locative references using a geodatabase

These rules are language-independent and apply to n-grams. In particular, the rules deal with bigrams and unigrams when matching the tokens in the tweets against the tokens found in our place-name dataset built from GeoNames (see Flowchart 1 in Appendix):

i) For bigrams, if (a) the first token is not a noun, and (b) the second token is not a proper noun, or the second token is a directional marker (e.g. *South*, *sur*), it is very likely that the n-gram is not a locative reference. Examples of bigrams taken from the corpora that were found in the place-name dataset but are not actual locative references according to the linguistic context are *the country, beautiful isle, nice airport, the South, el tiroteo* (‘the shooting’), *la bomba* (‘the bomb’), *buenas tardes* (‘good evening’), *de armas* (‘of weapons’), *el Sur* (‘the South’) *la femme* (‘the woman’), *le signal* (‘the signal’), *la piscine* (‘the swimming pool’), etc.

ii) For unigrams, (a) if the unigram is not a proper noun, or (b) if the unigram is in the stopword dataset, the location-indicative noun dataset or the locative marker dataset, it is very likely that it is not a locative reference. Examples of unigrams taken from the corpora that were found in the place-name dataset but are not actual locative references according to the linguistic context are *police, going, Ashley, accident, Clinton, Gracias* (‘thanks’), *terremoto* (‘earthquake’), *camping*, *compartir* (‘share’), *López*, *amor* (‘love’), *coche* (‘car’), *grand* (‘big thing/person’), etc.

At times, these rules were bypassed by certain n-grams that were captured in place-name dataset but that were unfortunately not filtered by the stopword dataset, especially in the case of person names (e.g. *Jam*, *Yao*, *Robles*, *Nemo*, *Obama*, etc.). The rules and datasets thus play a preventive role which might not always avoid the extraction of wrong instances, since these person names can also be actual locations and only the linguistic context can disambiguate them. Therefore, we searched for a trade-off between precision and recall when using these rules and datasets.

5.3.1.1.2. Rules that exploit locative prepositions

The following rules are language-independent. If a token is a locative preposition, then it is very likely that any succeeding combination of proper nouns is a locative reference, except when (a) the proper noun is a date, or (b) the proper noun is a person name or any other type of named entity, ruled out by the stopword dataset (see Flowchart 2 in Appendix). The following examples of actual tweets from the corpora illustrate the extracted locative references. Example (3) shows the extracted unigram *Palakkad*, thanks to the presence of the locative preposition *at*.

(3) *Visited home of Mr. Shobha Aboobacker Sahib at Palakkad who passed away today morning in an accident*

in and *across* are other locative prepositions that can signal a locative reference, illustrated by Example (4), Example (5), Example (6), and Example (7).

(4) *When you're doing your show in San Bernardino...and you need a listener to tell you about a 3.5 earthquake*
(5) *Golestan province N Iran Three weeks after the floods, the houses are still surrounded by floods in Aqqala.*
(6) *Floods in #Iran - Villages in #Khuzestan surrounded by floods, no sign of state relief. #IranFloods #IranRegimeChange.*
(7) *#GhassemSoleymani very clearly doesn't care about #flood and its victims across Iran.*

In Spanish, *en* is the most prototypical locative preposition:

(8) *La #Tormenta en MADRID pone de manifiesto, otra vez, el lamentable estado de las infraestructuras*
'The storm in Madrid exposes, once again, the lame conditions of the infrastructures'

(9) *Vuelve a caer más fuerte que antes en Valdemoro, ahora con aparato eléctrico.*
'It rains more heavily than before in Valdemoro, now with thunder and lightning'

(10) *Inundaciones en Arturo Soria. Garajes inundados, ahora cae piedra #Madrid @112cmadrid @E112Andalucia.*
'Floods in Arturo Soria. Flooded garages, dropping stones now #Madrid @112cmadrid @E112Andalucia'

(11) *#NuevoLeon Fuertes lluvias en Nuevo León dejan dos muertos e inundaciones*
'#NuevoLeon Heavy rains in Nuevo León kill two people and cause floods'

(12) *Agresiones al ejército en Michoacán - Severos daños por lluvias en Sinaloa*
'Assaults on the army in Michoacán – Serious damage caused by rains in Sinaloa'

In French, *dans*, *en*, and *à* introduce many locative references:

(13) *Au moins un mort et de nombreux blessés après une #attaque dans un lycée professionnel en #Finlande*
'At least one killed and many wounded after #attack in technical college in #Finland'

(14) *Nouvel incident dans une usine " #Seveso seuil haut" à #Rouen*
'New incident in a factory " #Severo high threshold" in #Rouen'

(15) *j'me promenais tranquillement dans Madrid quand soudain j'ai entendu des coups de feu !!*
'I was calmly walking in Madrid when suddenly I heard gunshots !!'

Now we present other tweets in which our rules and datasets did not manage to detect the locative references. In Example (16), *Indinapuram* was missed because *between* was not considered a locative preposition in the English lexical dataset due to its ambiguity in some contexts and its less-than-prototypical spatial nature[11]. In this regard, we also excluded the directional prepositions *to* and *from*, considering the cost-benefit ratio of their ambiguous nature, since they appear with ditransitive constructions (e.g. *give, obtain, receive,* etc.) typically followed by person names.

(16) *Pls consider asking the #NHAI to close the central verge on #NH24 between #Indirapuram and…*

[11] Ambiguity from an NLP perspective refers to the inability of machines to disambiguate more than one meaning. This phenomenon is more commonly known as 'polysemy' in Theoretical Linguistics.

Rules were constructed with respect to the languages supported by LORE. Therefore, only proper nouns that follow locative prepositions are considered, so the rules cannot for now handle the combinations of proper nouns with words of different grammatical categories, e.g. determiners, prepositions, etc., as shown in Example (17) and Example (18).

(17) *Se desborda rio en Los Reyes*
'Overflowed river in Los Reyes'

(18) *Pollution à l'arsenic dans l'Aude*
'Arsenic contamination in l'Aude'

At other times, n-gram combinations were wrongly detected as locative references. In Example (19), *Mandarin* was extracted as a locative reference because, according to the POS tagger, its grammatical category is proper noun. Since it was preceded by the preposition *en*, and the stopword dataset could not filter it out, it was wrongly retrieved as a locative reference.

(19) *Si claro como no...ahora digame el chiste en ~~Mandarin~~ por favor!!*
'Yeah, yeah, of course…now tell me the joke in Mandarin, please!!'

5.3.1.1.3. Rules that exploit location-indicative nouns

These rules are language-dependent. On the one hand, in the case of English, there are several cases in which a combination of tokens including a location-indicative noun refers to a locative reference (see Flowchart 3 in Appendix). For example:

i) when location-indicative nouns are preceded by one or a combination of proper nouns,

(20) *Pattonville Fire Protection District is currently responding to an emergency incident for a(n) 13 Diabetic Problems QD*

(21) *Westville Public Schools is having a mock accident today at 10 am. Please do not be alarmed at all of the EMS*

(22) *Incident on #LLine Both directions from Myrtle Avenue Station to Rockaway Parkway-Canarsie Station*

(23) *Rising Seas May Mean Tampa Bay Floods Even During Sunny Days*

ii) when one of the preceding tokens is an Arabic numeral, since it is very likely that the locative reference is an address,

(24) *South LA 13219 S Penrose Ave **Hit and Run No Injuries***

iii) when one of the preceding tokens is a directional marker,

(25) *Accident cleared in #Edmond on NW 178th St at N Pennsylvania Ave #OKCtraffic*

iv) when they are followed by one or a combination of proper nouns, including numbers or directional or movement markers (e.g. *Mount Everest*, *River Thames*),

v) if they are followed by the preposition *of*, and while they are followed by one or a combination of proper nouns, then it is very likely that they refer to a locative reference,

(26) *I'm from an upper middle class suburb of Boston.*

No examples of missed locative references were found in relation to the functioning of the rules themselves. It is true, however, that a few were missed because the POS tagger assigned grammatical categories other than nouns for a few location-indicative words in some contexts. In Example (27), *ST* was assigned the adjective POS tag.

(27) *Motor Vehicle Accident - WATERBURY #RT8 South at Exit 34 (WEST MAIN ST #1) at 4/11/2019 10:58:08 AM #cttraffic*

There were a few cases of wrongly retrieved instances, as those in Example (28) and Example (29). In Example (28), *Dr*. was wrongly taken as the abbreviation for the location-indicative noun *drive*, and since the tokens that preceded it were all proper nouns, the whole set of tokens were wrongly considered within the boundaries of a false locative instance. Again, context and a deep-semantic system could have proven essential in disambiguating this type of cases.

(28) *#~~RoadSafetyInitiativeByDSS Saint Dr.~~ MSG has come up with the initiative to tie reflector belts on the stray animals*

In Example (29), *1st church* and *2nd church* were wrongly extracted by means of the rules that searched for Arabic numerals, which may sometimes be ordinal numbers (e.g. *101th street*).

(29) *@TalbertSwan The ~~1st church~~ burned, everyone thought it could have been an accident. After the ~~2nd church~~ burned, deacons…*

On the other hand, in the case of Spanish or French, a combination of tokens refers to a locative reference when location-indicative nouns are followed by one or a combination of proper nouns, sometimes introduced by (a) a preposition, (b) a determiner, or (c) a preposition + determiner, or followed by one number (see Flowchart #4 in Appendix). The following examples illustrate the locative references extracted on the basis of this rule:

(30) *Incidente vial entre bus ?? ?y un ciclista ????? en la Av. Boyacá con Calle 12, sentido norte- sur. Unidad de ?? @TransitoBta y ?? asignada.*
'Road incident between bus and a cyclist in the Boyacá Ave with 12 Street, northbound-southbound. @TransitoBta unit assigned.'

(31) *#26Ago Accidente vial de camionetas del Sebin en la carretera Higuerote-Curiepe dejó un fallecido.*
'#26Aug Road incident between Sebin vans in the Curie-Higuerote road kills one person.'

(32) *INUNDACIONES EN LA M-40. Imagen de la cámara de la M-40 en el barrio de La Fortuna, en el kilómetro 30.*
'FLOODS IN M-40. Picture from the M-40 camera in the La Fortuna neighborhood, in kilometer 30.'

(33) *La peor parada: inundaciones en Baños de Río Tobía por las tormentas*
'Worst off: floods in Tobía River Baths caused by storms'

(34) *Patrulla de vialidad permanente y campaña concientización, después de accidente en carretera a Boquilla*
'Ongoing road management patrol and awareness campaign after accident in the road to Boquilla'

(35) *Un appel à témoins a été lancé par le commissariat d'Ivry suite à un accident de la route*
'A call for witnesses was launched by the Yvri police station following a road accident'

However, there were a few examples of missed locative references. In Example (36), only *provincias de Ávila* could be extracted, because the regex-based rules could not capture the coordinated items in the NP. Since the number of coordinating items is subject to variation, it is hard to formalize a general pattern without finding exceptions to the rule.

(36) *Inundaciones en las provincias de Ávila, Segovia y Valladolid*
'Floods in the provinces of Ávila, Segovia, and Valladolid'

In Example (37), *Calzada* is not in the location-indicative noun dataset, because it was not subsumed by any of the synsets extracted from EuroWordNet, so the rules could not detect the locative reference.

(37) *Vecinos de #Naucalpan se manifiestan sobre Calzada San Agustín para exigir reforzamiento de muros del Río Hondo*
'#Naucalpan residents protest over San Agustín road to demand the reinforcement of walls in Hondo river'

Moreover, symbols such as the dash, which might occur within the boundaries of locative references, as in Example (38), are not currently dealt with by the rules because these could appear in any position, making the formalization of patterns very hard.

(38) *Alrededor de las 9:10 de esta mañana, una volcadura en la carretera Navojoa - Los Mochis dejó sin vida a una persona.*
'Around 9:10 this morning, rollover in Navojoa – Los Mochis road killed one person.'

In Example (39), the reason why this instance was extracted is due to the fact that *cámara* is a location-indicative noun in the Spanish location-indicative dataset. Since there is not a word-sense disambiguation system in LORE, it is for now impossible to avoid matching ambiguous items whose meaning is different from the location-based one.

(39) *INUNDACIONES EN LA M-40. Imagen de la ~~cámara de la M-40~~ en el barrio de La Fortuna, en el kilómetro 30.*
'FLOODS IN M-40. Picture from the M-40 camera in the La Fortuna neighborhood, in kilometer 30.'

In Example (40), the regex-based rules could not capture the locative reference because of the complexity of the NP, in which a determiner appeared between two proper nouns.

(40) *Plusieurs riverains ont composé le « 17 » ce dimanche soir à Montpellier, dans le quartier de la Croix d'Argent*
'Several local residents composed the "17" this Sunday evening in Montpellier, in the Croix d'Argent district.'

We developed a language-independent rule on the basis of road and highway naming conventions used in English-speaking, Spanish-speaking countries, and French-speaking countries obtained from Wikipedia[12]. For example, the regex-based rule can extract highways and roads such as the following, where round brackets indicate the optionality of the dash, and where *n* indicates a given number: A(-)n, B(-)n, J(-)n, I(-)n, H(-)n, N(-)n, TX(-)n, US(-)n, SR(-)n, CR(-)n, RT(-)n, RTE(-)n, HWY(-)n, NH(-)n, MD(-)n, etc. The rule states that, if a token includes one or two letters, accompanied or not by the dash symbol, and then followed by a number between 0 and 9999 and an optional letter at the end, then it is very likely that it is the locative reference of a traffic way (i.e. highway or road) (see Flowchart 5 in Appendix):

(41) *Cortadas por inundación tras la tormenta la M-506, la M-40 y al menos 6 líneas de Metro*
'M-506 and M-40 and at least 6 underground lines blocked because of floods after storm'

(42) *Gracias a la #Tormenta llevamos dos horas parados en la A-42 por inundaciones y sin previsiones de movernos. Genial oye.*
'Thanks to the #Storm we have been kept for two hours in A-43 because of floods, and not expecting to move. Great, huh.'

In English, directional or movement markers may precede or follow highways, which are also captured within the boundaries of the extracted locative references. In Spanish and French, directional markers may follow highways. Moreover, by means of another rule, we account for whitespaces between characters (e.g. *I 84*):

(43) *Update - #M5 northbound J19 #Gordano towards J18 #Avonmouth. Our traffic officers have driven through the area*

(44) *accident:NorthWest Pkwy (TX-114 alt) eastbound TX-26 Grapevine various Lns blocked*

(45) *Incident on #I278 EB from 3rd Avenue to Exit 26 - Hamilton Avenue*

(46) *Motor Vehicle Accident - WATERBURY #RT8 South at Exit 34 (WEST MAIN ST #1) at 4/11/2019 10:58:08 AM #cttraffic*

(47) *One person was killed in an accident on southbound I-91 in New Haven on Thursday morning.*

(48) *#INCIDENT: #A40 est*
'#INCIDENT: #A40 east'

[12] For instance, see https://en.wikipedia.org/wiki/List_of_motorways_in_the_United_Kingdom or https://en.wikipedia.org/wiki/Highways_in_Spain, or https://en.wikipedia.org/wiki/Autoroutes_of_France

Example (49) and Example (50) contain locative references missed by these rules.

(49) *Accident on 35W NB @ County Road 96*

In Example (50), the slash symbol, which is not captured by the rules, hampers a successful extraction of the whole locative reference.

(50) **UPDATE* 15:20?? #M8 E/B J22 Plantation - J18 Charing Cross remains ?CLOSED? due to a police incident on the Kingston.*

Other instances, such as Example (51), Example (52), and Example (53), were wrongly taken as locative references.

(51) *Today #Afghan Army helicopter (~~MD-530~~) crashed dawn due to technical issues while returning from a training operation*

(52) *I have done this by accident and printed tickets ~~A2~~..*

(53) *Terremoto ~~M5.0~~ - Ryukyu Islands, Japan*
'M5.0 earthquake - Ryukyu Islands, Japan'

5.3.1.1.4. Rules that exploit locative markers

This type of rules can be divided into two main groups: (a) rules that apply to directional markers, and (b) rules that apply to distance and temporal markers (see Flowchart 6 in Appendix). On the one hand, a combination of tokens containing a directional marker is very likely to refer to a locative reference:

i) when the tokens following the directional marker are proper nouns or locative references previously retrieved, which could be preceded by a preposition (e.g. *de, of*); this rule is language-independent.

(54) *South LA 13219 S Penrose Ave **Hit and Run No Injuries***

(55) *Cleared: Incident on #US9 SB from South of CR 522/Throckmorton St to Exit 26 - Hamilton Avenue*

(56) *Incident on #I78 WB at East of Exit 55 - CR 602/Lyons Ave*

(57) *#VIDEO Este fin de semana, se registraron severas inundaciones al norte de #LosMochis´*

'#FOOTAGE This weekend several floods were recorded in the north of #LosMochis'

(58) *Inondations meurtrières dans le nord de l'Inde*
'Deadly floods in northern India'

ii) when the tokens following the directional marker are proper nouns or locative references previously retrieved preceded by a preposition (e.g. *of*), and if the preceding token is a distance marker (e.g. *km, miles*) preceded by a number; this rule is English-specific.

(59) *A 3.5 magnitude earthquake occurred 1.86mi SW of San Bernardino, CA.*

(60) *#Earthquake (#tërmet) M2.7 strikes 20 km NW of #Durrës (#Albania) 42 min ago*

iii) when the tokens following the directional marker are proper nouns or locative references previously retrieved preceded by a preposition (e.g. *de*), and the preceding tokens are a number followed by a distance marker (e.g. *kms, millas*) followed by a preposition + determiner (e.g. *al, del, au, du*), e.g. *20 kilómetros al sur de Granada* or *100 kms au sud de Paris*; this rule is specific of Spanish and French.

On the other hand, a combination of tokens containing a distance marker (e.g. *km, mile, metro, kilomètre*) or temporal marker (e.g. *horas, hrs, mins, heures*) is very likely to refer to a locative reference:

i) when these markers are preceded by a number and followed by an optional adverb + preposition + optional definite determiner (e.g. *away from, out of, from the, of*) and the following tokens are proper nouns or locative references previously retrieved; this rule is English-specific.

(61) *18:03 Very bad accident just 4 Kms from Narok town*

(62) *Cleared: Motor Vehicle Accident - HARTFORD #I84 West 0.02 miles before Exit 51 (I-91NB) at 4/11/2019 10:56:03 AM*

ii) when these markers are preceded by a number and followed by a preposition + optional definite determiner (e.g. *de, de la, du, hacia el, ver le*) and the following

tokens are proper nouns or locative references previously retrieved; this rule is Spanish- and French-specific.

(63) *Se desploma helicóptero matrícula XB-GIL a 6 kilómetros de Tuxtepec, Oaxaca, en la Finca Nuevo Mundo*
'XB-GIL helicopter crashes 6 kilometers away from Tuxtepec, Oaxaca, at Finca Nuevo Mundo'

(64) *A las 22:37 horas, un terremoto de 7.3 grados Richter, con epicentro a 111 km de puerto El Triunfo*
'At 22:37 hours, an earthquake of 7.3 degrees Richter, with epicenter 111 km away from Puerto El Triunfo'

(65) *Tremblement de terre mag 4.5 à 23,36km de Ak-Chaganak*
'Earthquake mag 4.5 at 23.36km away from Ak-Chaganak'

At times, the rules missed or wrongly retrieved locative references. That was the case of Example (66), where *N Iran* was not extracted but only *Iran*, due to the fact that the rules belonging to the location-indicative word module previously extracted *Golestan province N*.

(66) *April 11 -Aqqala, Golestan province N Iran Three weeks after the floods, the houses are still surrounded by floods in Aqqala.*

In Example (67), the coordinated items could not be captured by the rules due to the lack of a formalized pattern for coordination.

(67) *Preocupación por las inundaciones en las zonas este y sur de Madrid, tras la tormenta*
'Worries over floods in the eastern and southern areas of Madrid after storm'

In Example (68), the rules could not capture the locative reference because they did not account for a directional marker following a proper noun.

(68) *#Duplessis nord, hauteur boulevard du Versant-Nord, VD bloquée*
'Duplessis north, to the height of Versant-Nord boulevard, VD blocked'

5.3.1.1.5. Safe-checking rules

The successful application of the linguistic-based rules must be accompanied by safe-checking rules to ensure that (i) the same extracted locative reference is not repeated, (ii) that boundaries

between locative references do not overlap, and (iii) that the boundaries of locative references are well delimited.

In particular, when delimiting the boundaries of locative references, if a detected proper-noun token takes part in another locative reference, either (a) discard the proper-noun token and leave the previously detected locative reference intact, or (b) remove the locative reference, probably wrongly delimited, and add it again with decreased or expanded boundaries. Case (a) applies in all the linguistic processing modules as the last safe-checking rule before adding a potential locative reference that might have already been extracted. For instance, if proper nouns follow a locative preposition, and the first of those was contained in an already-extracted locative reference from the place-name search in the geodatabase, the safe-checking rule discards those proper nouns. Case (b) is specific to how the linguistic processing module handles location-indicative nouns by expanding the boundaries of previously detected locative references (e.g. *Athens → city of Athens, M-30 → autovía M-30, Versant-Nord → boulevard du Versant-Nord*), and also applies to the addition of locative markers to previously detected locative references by expanding their boundaries with these markers (e.g. *Silicon Valley → 40miles SW of Silicon Valley*, *calle Menéndez Pelayo → 15 minutos de la calle Menéndez Pelayo*, *Paris → 30kms au sud de Paris*).

5.3.2. Language-specific lexical datasets

Our location-detection model exploits five language-specific lexical resources: a POS-tag and locative-preposition dataset, a place-name dataset, a location-indicative noun dataset, a locative-marker dataset, and a stopword dataset.

5.3.2.1. POS-tag and locative-preposition dataset

This dataset defines the language-specific grammatical categories and locative prepositions fed into the system for multilingual location detection. The relevant POS tags considered were common nouns, proper nouns, prepositions, determiners, and definite determiners. The choice of the locative prepositions was rigorously considered on the basis of the manual linguistic analysis performed on the corpora, together with the results derived from preliminary studies on the use of locative prepositions in microtexts (Vasardani et al., 2013; Dittrich et al., 2014; Radke et al., 2019). All these were materialized with regexes and configuration files. Table 18 shows the locative prepositions considered for each of the languages supported.

Table 18. Locative prepositions in LORE.

Language	Locative prepositions
English	at, @, in, near, along, across
Spanish	en, hacia, hasta

French	à, au, aux, en, vers

Further details about the layout of the file and regexes used for this dataset are given in Section 6.1.1.1.

5.3.2.2. Place-name dataset

The place-name dataset was automatically retrieved from the geographic database GeoNames for each of the languages. Each place-name dataset contains location types such as geopolitical entities, some natural landforms, POIs, and traffic ways. We carried out an automatic filtering and pre-processing process that consisted of the following two consecutive tasks:

(i) Retrieve place names only, whose population size is greater than 100 inhabitants. The population size filter was needed to avoid the retrieval of place names that corresponded to very common words, and whose presence resulted in the retrieval of many false positives.

(ii) Remove names of historical places that no longer exist, which are marked by the tag "historical" (e.g. ancient Roman provinces).

All this greatly contributed to speeding up the performance of our model. Indeed, the population size filter served to dramatically decrease the rate of false positives, although it slightly increased the number of false negatives. The English place-name dataset is the largest, containing 792,060 entries, whereas the Spanish and French ones store 217,900 and 98,989, respectively. This lower number of place names in the Spanish and French datasets had an impact on the lower recall scores achieved by LORE in those languages. Further details about the computational methods used for the processing and compilation of the place-name datasets are given in Section 6.1.1.2.

5.3.2.3. Location-indicative noun dataset

The location-indicative noun dataset was built from the EuroWordNet lexicon (Miller, 1995; Fellbaum, 1998). We automatically extracted all the hyponyms as lexicalized in each of the languages subsumed by the synsets that had a locative meaning: "road.n.01", "building.n.01", "facility.n.01", "junction.n.01", "district.n.01", "area.n.01", "geological_formation.n.01", "body_of_water.n.01", "tract.n.01", "way.n.06", and "beach.n.01". In the filtering process, duplicates, more-than-two-word items, and multi-word lexical units containing place names (e.g. *Roman Empire, Baltic state,* etc.) were automatically removed. Then we manually discarded items that are not typically accompanied by proper nouns, and that are not included in our

definition of locative references (e.g. *bed, melting pot, scene of action, junk pile, parts,* etc.) in the English, Spanish, and French datasets.

In the end, the resulting English dataset, containing 1217 lexical items, was expanded with a list of traffic-way and other place abbreviations obtained from the US postal service database[13], with a sum total of 1766 items. Table 19 provides a sample of some of English location-indicative nouns organized in terms of the proposed typology of locative references.

Table 19. Sample of the English location-indicative noun dataset.

Geopolitical Entities	Natural landforms	POIs	Traffic ways
barrio	beach	art school	alley
caliphate	canyon	bus station	avenue
city	gulf	café	boulevard
country	hill	castle	driveway
county	lake	cathedral	freeway
jurisdiction	mountain	embassy	highway
province	ridge	hospital	street
region	river	hospital	parkway
state	valley	hotel	road
town	volcano	university	street

Likewise, the Spanish dataset, after the pre-processing step, contained 644 items and was subsequently expanded with additional items of abbreviated location-indicative nouns retrieved from the Web[14]. Table 20 offers a sample of some of the Spanish location-indicative nouns following the same proposed typology of locative references.

Table 20. Sample of the Spanish location-indicative noun dataset.

Geopolitical Entities	Natural landforms	POIs	Traffic ways
barrio	afluente	academia	acceso
ciudad	cima	albergue	autovía
condado	cuenca fluvial	biblioteca	avenida
distrito	desierto	centro médico	calle
dominio	isla	cine	camino
localidad	lago	escuela	carretera
municipio	litoral	hospital	carril
país	llanura	museo	intersección
provincia	montaña	restaurante	parada
urbanización	río	teatro	vía

In the case of French, we only obtained location-indicative nouns from EuroWordNet with automatic methods using the synsets for words of location meaning in this language. Table 21

[13] http://cool.conservation-us.org/lex/abbr_suf.html

[14] http://www.wikilengua.org/index.php/Lista_de_abreviaturas_de_v%C3%ADas and https://www.abreviaciones.es/edificios-lugares-y-negocios/

shows a sample of some of the French location-indicative nouns following the same proposed typology of locative references.

Table 21. Sample of the French location-indicative noun dataset.

Geopolitical Entities	Natural landforms	POIs	Traffic ways
arrondisement	aquifère	aérodrome	allée
califat	bassin	ambassade	arrêt
capitale	canal	boulangerie	autoroute
département	canyon	caféteria	avenue
etat	désert	église	chémin
ghetto	île	galerie	route
municipalité	littoral	gym	rue
pays	mer	hôtel	ruelle
province	plateau	palais	sortie
ville	prairie	station	voie

Further details about how these datasets were compiled are given in Section 6.1.1.3. and Section 6.1.1.4.

5.3.2.4. Locative-marker dataset

The locative-marker dataset was, on the other hand, manually constructed. The proposed typology for locative markers comprises directional markers, movement markers, distance markers, and temporal markers. These act as optional phrases in complex locative references. Table 22 illustrates a few examples of these markers for the English language, whereas Table 23 and Table 24 present a few of those locative markers used for the Spanish and French languages, respectively. In sum, the English locative-marker dataset contains 71 items, while the Spanish and French datasets stores 45 items each. Further information about the compilation of these datasets are provided in Section 6.1.1.4.

Table 22. A sample of the English locative-marker dataset.

Directional markers	Movement markers	Distance markers	Temporal markers
North, N	Northbound, NB	kilometre(s), kilometer(s), km(s)	hour(s), hr(s), h(s)
Southwest, sw	Southbound, SW	metre(s), meter(s), m(s)	minute(s), min(s)
East-North-East, ENE	Eastbound, EB	mile(s), mi(s)	
North, N	Westbound, WB	yard(s), yd(s)	
South, S			
Western			
South-east, SE			

Table 23. A sample of the Spanish locative-marker dataset.

Directional	Distance markers	Temporal markers

markers		
norte, n	kilómetro(s), km(s)	hora(s), hr(s), h(s)
sur, s	metro(s), m(s)	minuto(s), min(s)
este, e	milla(s), mi(s)	
oeste, s		
noreste, ne		
suroeste, so		

Table 24. A sample of the French locative-marker dataset.

Directional markers	Distance markers	Temporal markers
nord, n	kilomètre(s), km(s)	heure(s), hr(s), h(s)
soud, s	mètre(s), m(s)	minute(s), min(s)
est, e	mile(s), mi(s)	
ouest, s		
nord-est, ne		
sud-ouest, se		

5.3.2.5. Stopword dataset

The stopword dataset was automatically built and processed from different sources for each of the languages. The goal of this module is to filter and discard very frequent words in the place-name search or in the linguistic processing modules which, in most cases, do not correspond to actual place names, and may thus compromise the precision of the model. In the case of English, the English stopword dataset contains the 5000 most frequent English words downloaded from the *Corpus of Contemporary American English* (COCA)[15] together with 5541 common names and surnames[16], plus the words for week days and months. For Spanish, the Spanish stopword dataset containing 1989 lexical items was obtained from the *Corpus del Español*[17], enriched with a list of Spanish names and surnames of 558 items from a GitHub repository[18]. For French, the frequent word-lists and name lists were automatically retrieved and processed from GitHub repositories[19]. Table 25, Table 26, and Table 27 display samples of the different stopword datasets used in each of the languages, alphabetically organized.

Table 25. English stopword dataset.

Common	Names

[15] The 5000 most frequent English words were retrieved from COCA on https://www.wordfrequency.info/

[16] The names and surnames were compiled from https://names.mongabay.com/ and https://surname.sofeminine.co.uk/w/surnames/most-common-surnames-in-great-britain.html, and then those that matched the names of cities and countries were filtered out (e.g. *Nevada*, *Verona*, *Milan*, *Paris*, *Kenya*, *Valencia*, etc.).

[17] The 20k most frequent Spanish words were retrieved from the *Corpus del Español* on https://www.wordfrequency.info/files/spanish/spanish_lemmas20k.txt

[18] The names and surnames datasets can be found on https://github.com/olea/lemarios

[19] The repository for frequency word lists can be found on https://github.com/hermitdave/FrequencyWords/tree/master/content/2018 and the repository for lists of names on https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/packages/corpora/names.zip

words	
a	Aaron
abandon	Abbey
ability	Abbie
able	Abby
abortion	Abdul
about	Abe
above	Abel
abroad	Abigail
absence	Abraham
absolute	Abram

Table 26. Spanish stopword dataset.

Common words	Names
abdicar	Aarón
abeja	Abdón
abolengo	Abel
abolir	Abelardo
abonado	Abrahán
abrasar	Absalón
abrazo	Acacio
absurdamente	Adalberto
absurdo	Adán
abundancia	Adela

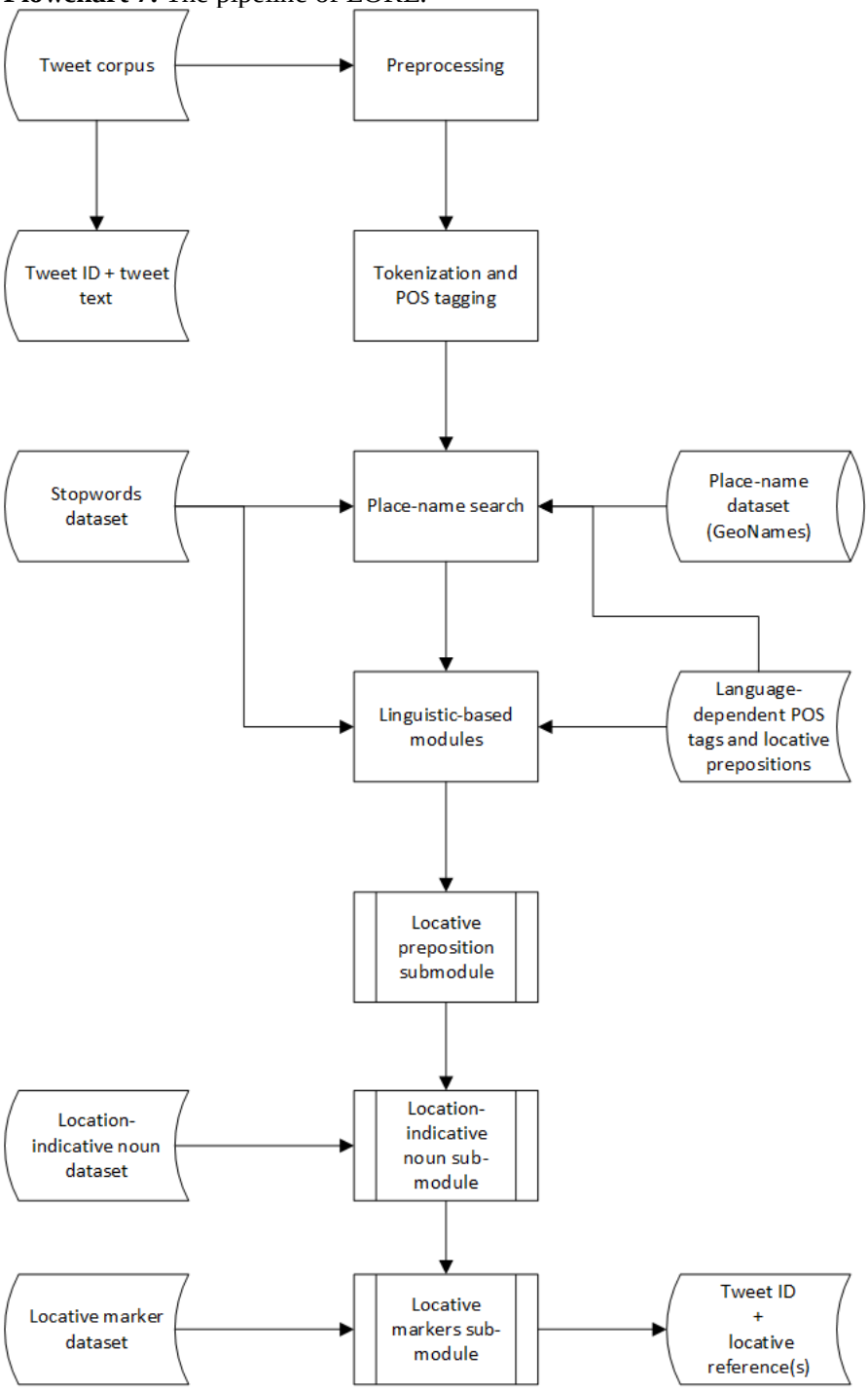
Table 27. French stopword dataset.

Common words	Names
abaisser	Aamir
abandon	Aaron
abdiquer	Abbey
abeille	Abbie
abolir	Abbot
aborder	Abbott
aboutir	Abby
aboyer	Abdel
abrasif	Abdul
abreuver	Abdulkarim

5.3.3. The pipeline of LORE

The pipeline of our rule-based location-detection model consists of four main blocks or modules that run sequentially, the fourth of which is further divided into three distinct sub-modules. The model makes use of multiple NLP libraries and techniques for tokenization, POS tagging, and an n-gram algorithm for the extraction of locative references from tweets. The functioning of the location-detection pipeline of LORE is depicted through the visual workflow of Flowchart 7 from the Appendix section, here reproduced, which shows the process of how the model processes a tweet corpus up to the final output of locative references.

Flowchart 7. The pipeline of LORE.



As we can observe, the location-detection pipeline runs sequentially. First, tweets are preprocessed before being tokenized and labeled with POS tags. Then, the place-name search module makes use of the place-name dataset constructed from the GeoNames database to extract locative references. The linguistic-based modules leverage the linguistic context to extract more locative references through linguistic cues such as locative prepositions, location-indicative nouns, and locative markers. Each of the submodules makes use of the

aforementioned language-specific datasets and regex-based rules. The pipeline is sequential because (a) the processing speed of each module is already fast enough and concurrent functioning does not add much of a speed improvement, and (b) the sequential character of the modules is paramount for the proper identification and delimitation of locative references. In this regard, the locative-marker submodule must come last to ensure that these locative markers are added to previously mined location references (e.g. *Cambie rd →south of Cambie rd*). Through the different safe-checking rules explained in Section 5.3.1.1.5., any location-detection module and submodule secure, to the best of their capabilities, that the same location reference is not added twice, and if added but wrongly delimited, the modules take care of removing the wrongly delimited one and adding the well-delimited instance. In the following sections, we introduce a more detailed explanation of the modular architecture of LORE and its sequential steps.

5.3.3.1. Pre-processing

At this stage, tweet text is pre-processed and cleaned for the later stages of the model. The consecutive tasks performed by this block are as follows:

i) Replace user mentions and URLs by the tokens "user" and "url" respectively.
ii) Remove emojis and other special characters and leave punctuation marks and other commonly used characters (/, @, |…).
iii) Remove genitive marker since it may interfere with the matching module.
iv) Remove extra white spaces.
v) Segment words in hashtags for possible location references contained therein.

All these tasks were performed by means of regexes. The model employs a third-party word segmentation algorithm[20] to split up words embedded in hashtags, since these usually contain otherwise missed location references. These steps are performed on any of the languages supported, except the word segmentation algorithm, which is language-specific. Example (69) and Example (70) show how the pre-processing stage is applied to a given English tweet.

(69) *Accident with injury in #EastBatonRouge on Airline SB at I 12 #traffic https://t.co/1WyWN3ulTM*
(70) *Accident with injury in East Baton Rouge on Airline SB at I 12 Traffic url*

5.3.3.2. Tokenization and POS tagging

[20] WordSegmentationTM: https://github.com/wolfgarbe/WordSegmentationTM

In this step, the Stanford POS tagger through the DAMIEN web service API (Periñán-Pascual, 2017) performs tokenization and POS tagging.

Tokenization involves splitting the tweets into unique tokens, such as words, punctuation marks, symbols, etc. Tokenizing a sentence of a given language presupposes knowledge about how that language, in textual form, is represented, and which element(s) determines the boundaries of each token. This knowledge can be provided by linguist experts, and materialized by means of rules or probabilistic-based algorithms. In the particular case of English (i.e. a Germanic language) or Spanish (i.e. a Romance language), tokens usually align to words, split by spaces and punctuation marks. Other languages such as Chinese pose a much greater challenge, since what we know as words appear together, and splitting by whitespace is thus not a good predictor of word boundary. On top of that, the irregular idiosyncrasy of the microtext genre provides at times greater difficulty in determining word boundaries due to the abundance of misspellings, hashtags, etc.

POS tagging assigns grammatical categories to each of these tokens. Assigning these implicitly assume knowledge about the morphology of the split tokens (e.g. inflections) and, above all, about the syntactic function that these tokens perform in the clause. In NLP, each language has its own set of part-of-speech labels, derived from their naming conventions. The Twitter medium makes it harder for POS taggers to perform reasonably well, due to grammatical errors, disjointed items, irregular spelling conventions, etc. As will be discussed below in Section 7.2., such error-prone behavior of the POS tagger with the tweets had an impact on the retrieval of locative references and on the discarding of false positives.

The Stanford POS tagger returns the split tokens from the tweets with their respective POS tags (Example (71) and Example (72)), which, together with their original form and their position in the tweet, are stored as attributes for each token object (Table 28).

(71) *Police have also closed the Kingston Bridge on the M8 because of this incident. https://t.co/L6MBWJz3lw*

(72) Police/NNS/1 have/VBP/2 also/RB/3 closed/VBN/4 the/DT/5 Kingston/NNP/6 Bridge/NNP/7 on/IN/8 the/DT/9 M8/NN/10 because/IN/11 of/IN/12 this/DT/13 incident/NN/14 ././15 url/NN/16

Table 28. The matrix representation of token objects.

wf_i (string value)	pt_i (string value)	p_i (integer value)
Police	NNS	1
have	VBP	2
also	RB	3
closed	VBN	4
the	DT	5

Kingston	NNP	6
Bridge	NNP	7
on	IN	8
the	DT	9
M8	NN	10
because	IN	11
of	IN	12
this	DT	13
incident	NN	14
.	.	15
url	NN	16

In other words, for every token object t_i, the model stores an attribute original word form $wf_{i,}$ an attribute POS tag pt_i, and an attribute position $p_{i..}$ Tweets are then stored as an object list $\{tw_{j,}$ tw_{j+1}, $tw_{j+2}\ldots\}$, each of which comprises a list of associated token objects $\{t_i, t_{i+1}, t_{i+2}\ldots\}$ and their corresponding tweet ID number id_k (Table 29).

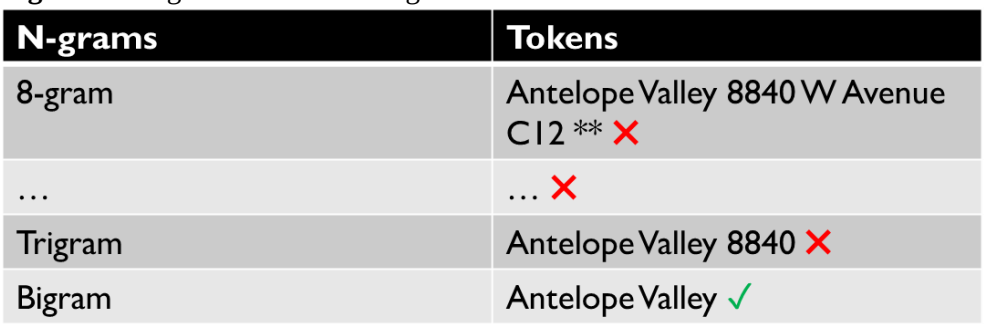
Table 29. The matrix representation of a list of tweet objects.

id_k	tw_j
j	$\{t_i, t_{i+1}, t_{i+2}\ldots\}$
j+1	$\{t_i, t_{i+1}, t_{i+2}\ldots\}$
j+2	$\{t_i, t_{i+1}, t_{i+2}\ldots\}$
j+n	$\{t_i, t_{i+1}, t_{i+2}\ldots\}$

5.3.3.3. Place-name search

The place-name search module aims to match n-grams of different size from the tweets with the place names included in the place-name dataset. The n-gram-based matching is grounded on a depth-search algorithm. It works in a decreasing fashion, starting from the highest n-gram of *n* tokens and iterating through its embedded n-grams until a match takes place or unigrams are reached. If no match is found, it iterates to the following tokens and starts all over again. Figure 10 shows a graphical example of the functioning of the n-gram-based matching.

Figure 10. N-gram-based matching.



As explained in Section 5.3.1.1.1., regex-based rules apply to discard different overmatching-prone n-gram combinations which are found in the place-name dataset and produce false

positives. These rules are indistinctively applied to all the languages supported, since, as we have attested in the development process of the model, the rules provide a sufficiently good benefit-cost ratio. By this ratio, we mean that the rules do not compromise the performance of the model or overload the system with long-winded regexes.

5.3.3.4. Linguistic processing

This module does not use the place-name dataset to mine locative references. Instead, it exploits language-specific linguistic knowledge and contextual clues found in the tweets to expand previously-detected locative references or discovering new locative references whose presence is signaled by locative prepositions, location-indicative nouns, and/or any kind of locative marker. The linguistic processing module is the result of the comprehensive linguistic analysis performed on the dev corpus and the extracted linguistic regex-based patterns and rules.

5.3.3.4.1. For locative references introduced by locative prepositions

The first task searches for proper nouns followed by locative prepositions that were not captured by the place-name dataset, in an n-gram window size of *n* words. Thus, following the regex-based rules in Section 5.3.1.1.2., if the current token t_i is a locative preposition and when the following tokens (i.e. t_{i+1}, t_{i+2}, t_{i+n}) are proper nouns, there is a high chance that these take part in a locative reference. The locative prepositions can be provided by the end user who specifies the language-specific locative prepositions of interest. For instance, for English, the prepositions considered were *in, at, @, near, across, along*. The reason why other English prepositions that signal location and direction (e.g. *on, to, from*) were not included was because these also have non-spatial senses in many ubiquitous contexts (Radke et al., 2019). In other words, those prepositions tend to overproduce many false positives when, for instance, they function as indirect objects (e.g. in giving verbs as *in John gave a present to Mary*) or oblique objects (e.g. *I received a present from John*). Only the stopword and location-indicative noun datasets are used to discard unlikely locative references. Acronyms and abbreviations of place names that went unnoticed in the place-name search module can now be retrieved thanks to the presence of locative prepositions (e.g. *in TX*).

5.3.3.4.2. For locative references introduced by location-indicative nouns

This second task expands locative references already retrieved by the previous modules or find new ones signaled by the presence of location-indicative nouns. This sub-module first matches location-indicative nouns, either unigrams or bigrams (i.e. t_i, or t_i and t_{i+1}), found in the location-indicative noun dataset and then considers a range of n-grams of size *n* to the left (e.g. t_{i-1}, t_{i-2}, t_{i-n}) or to the right (e.g. t_{i+1}, t_{i+2}, t_{i+n}) in search of proper nouns, resulting in the extraction of complex locative references. In other cases, the preposition *of* can be found between the

location-indicative noun(s) and the proper noun(s) in English locative references (e.g. *district of___*, *city of___*, or *province of___*, *coast of___*, etc.), or prepositions plus optional definite determiners in the case of Spanish and French (e.g. *barrio de ____*, *montaña del ____*, *region du ____*, *place de la ____*). This sub-module is also in charge of extracting traffic ways such as streets, highways, and roads. The regex-based rules that apply in this submodule as well as examples supporting their use are detailed in Section 5.3.1.1.3.

By means of the safe-checking rules explained in Section 5.3.1.1.5., this submodule also checks whether locative references found by the place-name search stage were wrongly delimited or, in other words, when locative references were shorter than expected or truncated: *High School* instead of *Batavian High School*, *Sichuan* instead of *South Sichuan Basin*, or *Glenwood* instead of *Glenwood Ave*, to name but a few. Boundaries would then be accordingly expanded in each of the locative references.

5.3.3.4.3. For locative markers

By means of the regex-based rules explained in Section 5.3.1.1.4. and the locative-marker dataset, locative references that were found in the previous modules can be expanded with any of the locative markers contained in the locative marker datasets, with the purpose of capturing the full scope of complex locative references (e.g. *Milan → 25kms SE of Milan*). Moreover, besides expanding already-retrieved locative references, this sub-module could in practice leverage these rules and the locative markers as contextual clues to identify new complex locative references missed by previous modules by looking at proper nouns that follow these markers in an n-gram window size of *n* words.

5.4. Neuronal LORE (nLORE)

Departing from LORE and the state-of-the-art approaches in NER, we decided to implement a neuronal network of type deep bidirectional RNN with LSTM as hidden layer structure and a CRF layer on top exploiting linguistic-based feature engineering and semantic information contained in the vectorial representations of tokens (i.e. word embeddings). Our aim was tri-fold. First, we wanted to assess whether the addition of linguistic-based features in a neuronal network can provide further benefits than using token form and POS tags as the only features. Second, we wanted to check whether linguistic-based feature engineering can somehow overcome the conundrum of finding and labeling a large train corpus, thus alleviating the computational cost, time, and resources typical of probabilistic-based approaches. Thirdly, we wished to elucidate whether a DL-based location-detection model can outperform our rule-based approach.

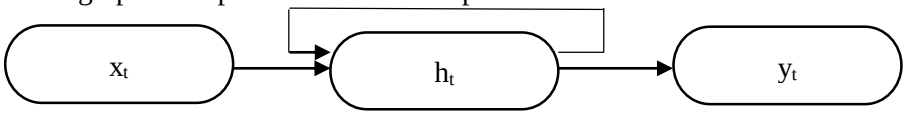
This section is organized as follows. First, we provide an explanation of the functioning of the neuronal network and word embeddings used. Then, we introduce the implementation of the neuronal network together with the training phase, followed by a description of the process of linguistic-based feature engineering and the hyperparameterization process undertaken in the training phase of the algorithm.

5.4.1. Deep Learning: RNN, CRF and word embeddings

5.4.1.1. Bidirectional RNN with LSTM and CRF on top

The underlying idea in the use of RNNs in NLP scenarios is that language must be treated as a temporal phenomenon, that is, language as a sequence of tokens that are combined one after another, where the prediction of a given word is dependent on earlier words (Jurafsky & Martin, 2019a). In a simple RNN network, also called Elman network (Elman, 1990), the output value is calculated on the basis of the input unit, multiplied by a weight matrix, which is then used in an activation function to calculate an activation value for the layers of hidden units. On the basis of these hidden units, an output value is obtained. The neuronal network is recurrent in the sense that the activation value of the hidden layer depends on the current input value and also on the output value of the previous hidden layer via backpropagation (Figure 11).

Figure 11. A graphical representation of a simple RNN.



In mathematical terms, this would be expressed as follows, where the value for a given hidden layer h_t at a given time t results from the activation function g that takes into account the sum of two values, the multiplication of a weight matrix U by the value of the previous hidden layer h_{t-1} and the multiplication of a weight matrix W by the input unit x_t (Equation (3)). For the output value y_t, we apply another activation function resulting from the multiplication of the hidden layer h_t by a weight matrix V, as seen in Equation (4).

$$h_t = g(Uh_{t-1} + W\,x_t) \quad (3)$$

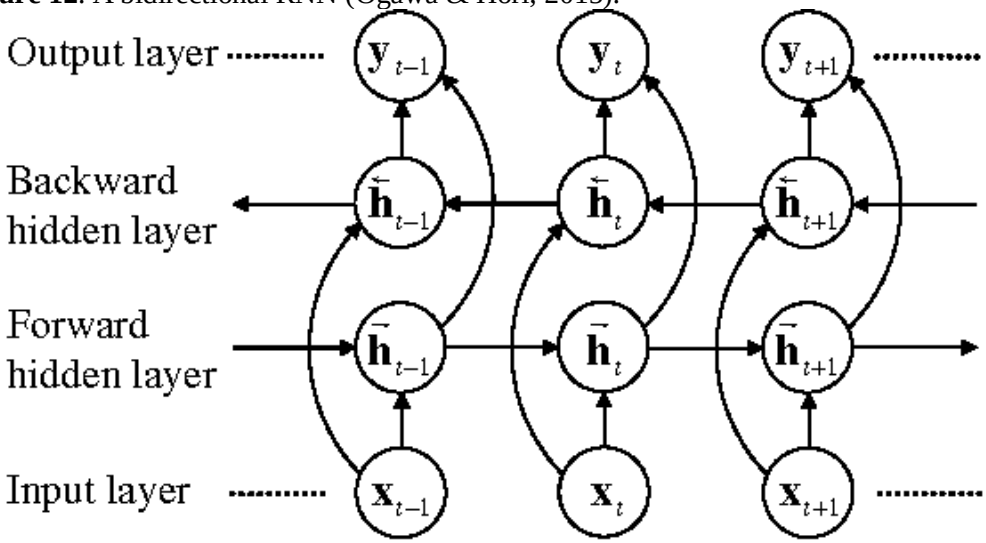
$$y_t = f(Vh_t) \quad (4)$$

In this sense, the previous hidden layer acts as contextual information or memory that captures earlier information for its later processing.

The network structure that we used is named 'bidirectional RNN'. A bidirectional RNN network consists of two simple RNNs stacked on top of each other, where, not only the previous

values are taken into account, but also the following ones, by means of backpropagation and forward-propagation (Schuster & Paliwal, 1997). In other words, if we had a sentence such as *The cat is on the mat*, if the current token is *on*, the network takes into account the previous word *is* and the following word *the*. Figure 12, taken from Ogawa & Hori (2015), illustrates the functioning of a bidirectional RNN.

Figure 12. A bidirectional RNN (Ogawa & Hori, 2015).



Mathematically expressed, a bidirectional RNN takes into account the preceding context by backpropagation and the following context by forward propagation, and then both outputs are concatenated by their addition or multiplication, as observed in Equation (5), Equation (6), and Equation (7).

$$h_t^f = RNN_{forward}(x_1^t) \quad (5)$$

$$h_t^b = RNN_{forward}(x_t^n) \quad (6)$$

$$h_t = h_t^f \cdot h_t^b \quad (7)$$

For the hidden layer structure, we used LSTM units because they keep in memory distant contextual information (Hochreiter & Schmidhuber, 1997), which is appropriate given the nature of language, where nearby contextual information is not enough to predict sequences of words. Consider, for instance, the sentence *The man who wore a hat on the top of his head and his wife were crossing the street*. A language model based on a bidirectional RNN would have missed subject-verb agreement because of the long nature of this complex NP containing two NPs, since in the local context, before the auxiliary form *were*, we find *wife*, which is attached by means of coordination to the previous NP. A simple bidirectional RNN would have not been

able to take into account the whole complex NP and would have assigned fewer probabilities to *were*, opting for the singular form *was* instead. LSTM networks provide an additional context layer, besides the recurrent hidden layer, and also neural units that, by operating on the input units, previous hidden layer and context layers, employ 'gates' or 'gating mechanisms' which are in charge of controlling the flow of information in and out of these neural units. A LSTM network receives a context layer as input, the previous hidden state, and the current input vector to produce updated context and hidden vectors as output. A forget gate erases unnecessary contextual information by calculating the weighted sum of the previous state's hidden layer and the current input, which is then computed by a sigmoid function and whose result is multiplied by a context vector.

Equation (8) and Equation (9), where f_t stands for forget gate, σ represents the sigmoid function and c_{t-1} the context vector, exemplify the functioning of a LSTM network.

$$f_t = \sigma\left(U_f h_{t-1} + W_f x_t\right) \tag{8}$$

$$k_t = c_{t-1} \cdot f_t \tag{9}$$

Afterwards, we need to compute the current information from the previous hidden state and current inputs, where *tanh* represents a hyperbolic function in Equation (10).

$$g_t = tanh\left(U_g h_{t-1} + W_g x_t\right) \tag{10}$$

Then, with the add gate, we provide the information relevant to the current context, as observed in Equation (11) and Equation (12).

$$i_t = \sigma(U_i h_{t-1} + W_i x_t) \tag{11}$$

$$j_t = g_t \cdot i_t \tag{12}$$

The context vector gets updated in Equation (13).

$$c_t = j_t + k_t \tag{13}$$

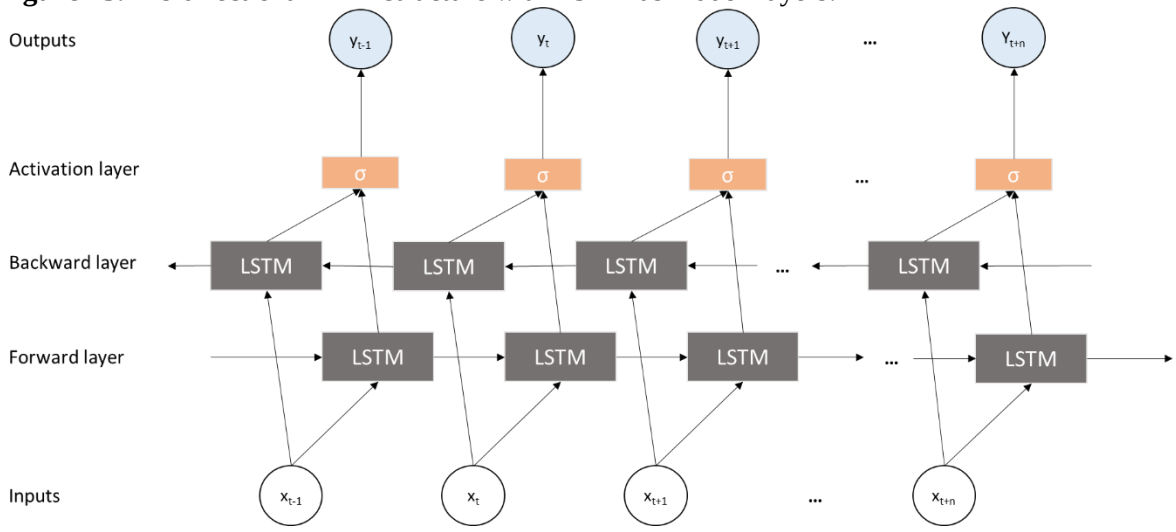
Another gate (i.e. the output gate) selects the information relevant for the current hidden state, as seen in Equation (14) and Equation (15).

$$o_t = \sigma(U_o h_{t-1} + W_o x_t) \tag{14}$$

$$h_t = o_t \cdot tanh(c_t) \tag{15}$$

Figure 13 summarizes in a graphical way the functioning of LSTM networks as hidden layers in a bidirectional RNN structure.

Figure 13. A bidirectional RNN structure with LSTM as hidden layers.



For the output layer structure, we used a CRF layer on top of the bidirectional RNN with LSTM, which can more accurately predict the tokens' labels.

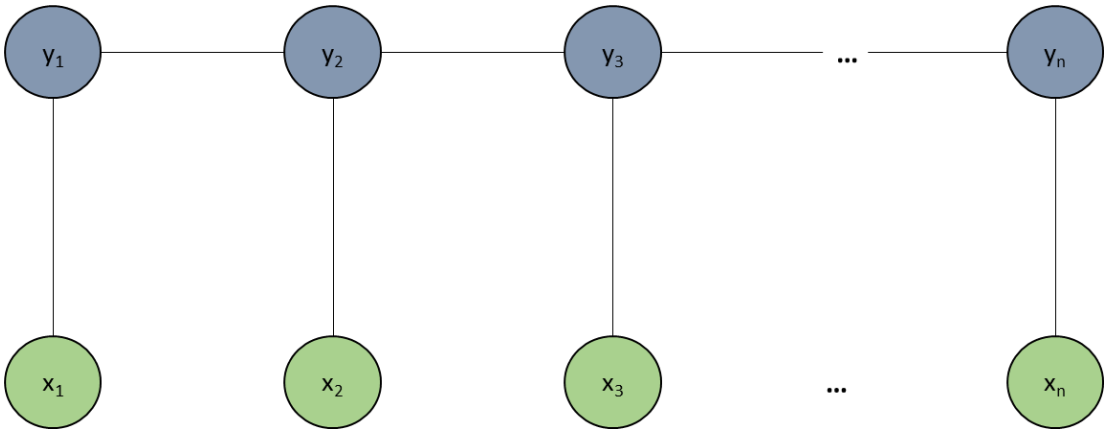
5.4.1.2. CRF: output layer structure

For the output layer structure, we used a CRF layer on top of the bidirectional RNN with LSTM. CRF is a discriminative model for sequential data that is able to predict a given label on the basis of contextual information (Lafferty et al., 2001). This is commonly expressed in Equation (16) as the probability of a given tag or label *y* on the basis of an input *x*, measured by the exponential of a weight parameter *w* multiplied by a feature function *ψ*, divided by the sum of all the exponentials. In mathematical notation, this is expressed as follows:

$$P\ (y|x) = \frac{\exp(w \cdot \psi(x,y))}{\sum_y \exp(w \cdot \psi(x,y))} \qquad (16)$$

The feature function determines whether a given feature exists or not. Figure 14 represents the functioning of a CRF algorithm, where each feature function is dependent on the estimation of the previous feature function.

Figure 14. A graphical representation of a CRF.



5.4.1.3. Vector semantics: dense, static word embeddings

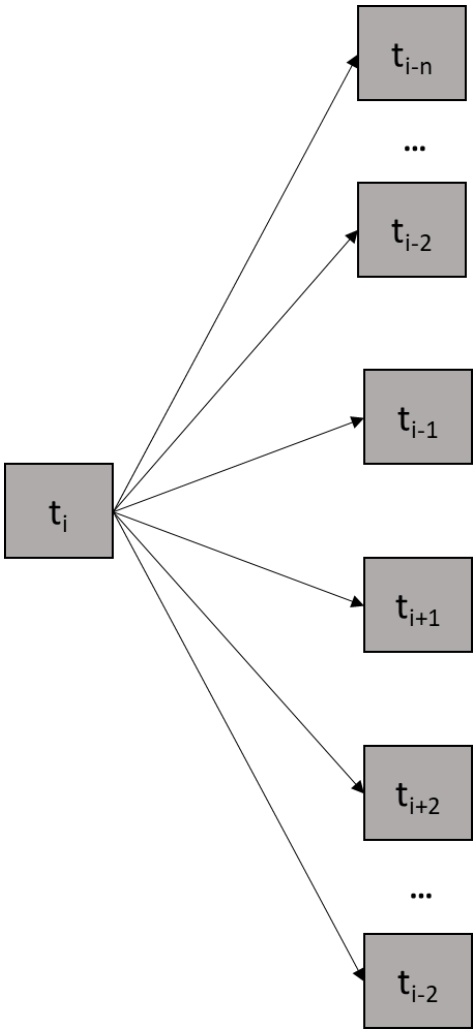
Vector semantics deals with the vectorial representation of words in an *n*-dimensional spaces using sparse or dense vector semantic models to capture meaning from text (Jurafsky & Martin, 2019b). Vector semantics depart from the all-time linguistic assumption that the meaning of a word lies in its neighboring company (Firth, 1957). Sparse vectorial representations (e.g. co-occurrence vectors) usually focus on word co-occurrence to determine their vectorial space. For instance, given the sentences *I like apples* and *I like reading*, we can compute sparse vectorial representations by means of a co-occurrence matrix (Table 30).

Table 30. Co-occurrence matrix in sparse vectorial representations.

	I	like	apples	reading
I	0	2	0	0
like	2	0	1	1
apples	0	1	0	0
reading	0	1	0	0

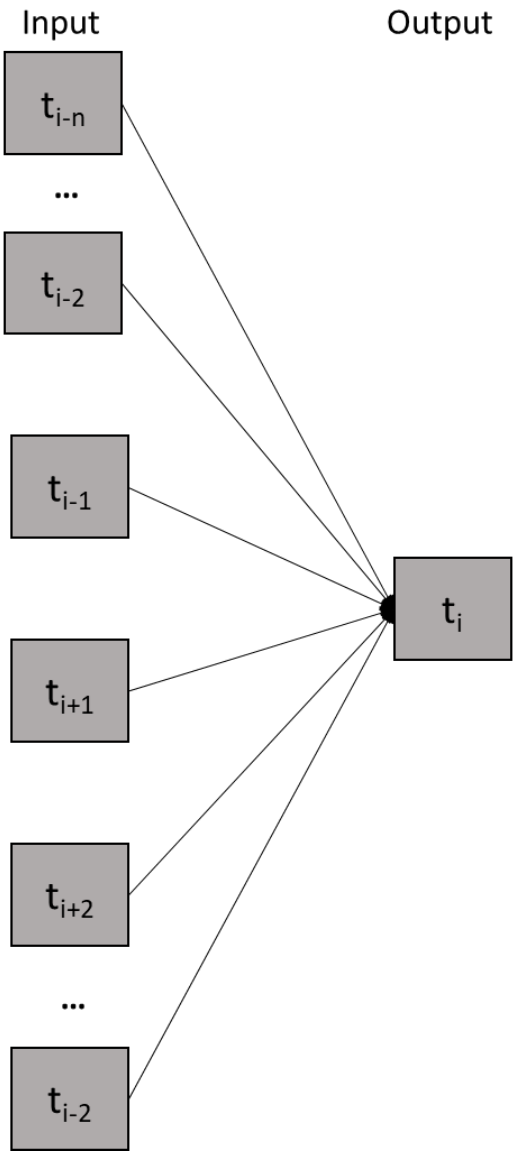
Thus, the vectors for the words are as follows: *I* [0, 2, 0, 0], *like* [2,0,1,1], *apples* [0,1,0,0], *reading* [0,1,0,0]. The number of vectorial spaces is determined by the vocabulary size of a given corpus which, if large, can produce vectorial spaces of thousands of dimensions, with many zero values. This can be very costly in computational terms. To solve this issue, dense, static word embeddings such as Word2Vec can store fewer dimensional spaces of vectors and dense vectors (i.e. where the value is not zero) by means of the skip-gram algorithm (Mikolov et al., 2013). The skip-gram algorithm, instead of counting words that co-occur together, performs a binary classification task that computes the probability of one word appearing next to another, that is, it predicts the context of words on the basis of a given word (Figure 15).

Figure 15. Skip-gram algorithm.



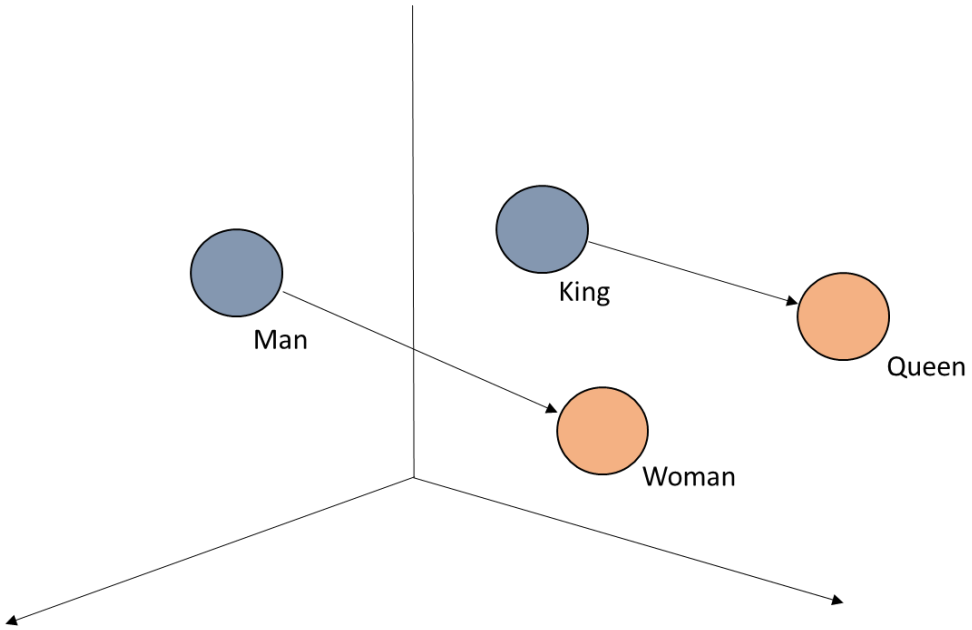
In the sentence __ ___ *sat* ___ ___ ___, the skip-gram algorithm computes the probability of the tokens in the syntagmatic axis that appear next to *sat*, selecting, for instance, *The cat* in subject position and *on the mat* in the prepositional object position as the most probable tokens. Another similar algorithm, the continuous bag of words, predicts a word given its context (Figure 16).

Figure 16. Continuous bag of words algorithm.



Thus, in the sentence *The cat ___ on the mat*, given the context, thanks to the continuous bag of words algorithm we could predict the most probable tokens for that gap in the paradigmatic axis: *sat*, *lay*, *slept*, etc. Thanks to both models, we can quantify or categorize semantic similarity among words (Figure 17).

Figure 17. A reduced three-dimensional space for word embeddings.



As illustrated by Figure 17, we can capture semantic similarity with word embeddings in terms of, for instance, gender relationships, besides synonymy and other semantic relationships (e.g. tense forms, hyponymy). In mathematical notation, semantic similarity between two words is computed by calculating the cosine similarity between the word vector of *a* and the word vector of *b*, where θ is the angle between the two vectors, and whose range will cover values between 0 and 1 (i.e. from no similarity to full similarity), as shown in Equation (17).

$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{||\vec{a}||\cdot||\vec{b}||} \tag{17}$$

5.4.2. Training phase

We trained up to eight models, according to the variables corpus size and linguistic-based features. With respect to the first variable, using a different corpus size responded to our aim to determine the effect of additional linguistic-based features in the performance of our model. In other words, our goal was to check whether linguistic-based feature engineering can overcome the problem of sparse corpus data in a probabilistic-based framework. Considering that one of our purposes was to examine whether the addition of extended linguistic-based features other than the basic ones, i.e. token form and POS tag, could enhance the performance of our model, we differentiate between a basic and an extended model: a basic one with token form and POS tag as main features, and an extended one with token form, POS tag, presence in the place-name dataset, presence in the location-indicative noun dataset, and locative marker. Further details

about the computational implementation of nLORE can be found in Section 6.2. and its subsections. Table 31 displays the characteristics of the eight trained models.

Table 31. Trained nLORE models.

Model	Corpus size (tweets)	Linguistic-based feature engineering
1	1000	Token + POS tag
2	1000	Token + POS tag + place-name dataset + location-indicative noun dataset + locative marker
3	3000	Token + POS tag
4	3000	Token + POS tag + place-name dataset + location-indicative noun dataset + locative marker
5	5000	Token + POS tag
6	5000	Token + POS tag + place-name dataset + location-indicative noun dataset + locative marker
7	7000	Token + POS tag
8	7000	Token + POS tag + place-name dataset + location-indicative noun dataset + locative marker

5.4.3. Linguistic-based feature engineering

It is widely accepted in the AI and NLP communities that, for the task of achieving optimal performance, constructing a set of meaningful features proves to be an effective strategy. For the training phase, we employed template features, context template features, and word embedding features to leverage linguistic-based features. As already sketched in the corpus compilation phase (Section 5.2.), the corpora used in nLORE follow a token-based tabular representation, where each column indicates a given feature, and the last column represents the token label. The token-form and POS-tag features are the same as those used in LORE; however, the membership of a given token or a combination of tokens in the place-name or location-indicative noun datasets are indicated by Boolean values, regardless of any of the rules used in the modules of LORE. In other words, no pre-processing or filtering step was applied, such as the stopword filtering step or the discarding of words that are not tagged as proper nouns. The only criterion is that, as long as a given word or combination of words is captured by the place-name dataset or the location-indicative noun dataset, they get the Boolean value 1; otherwise, their Boolean value is 0. The reason for this lies in our belief that the neuronal network used can automatically learn and infer, implicitly, the rules that we manually devised for the place-name search and linguistic processing modules without any manual input on our part. For the locative-marker feature, we did make use of the rules in the annotation process to find locative markers and label each locative-marker item or combination of items as 1. Also, the Boolean value 1 was assigned to tokens that are locative markers through a lookup in the locative-marker dataset, in case the rules missed any locative marker. The following subsections explain the implementation of the different feature types used in nLORE.

5.4.3.1. Template features

Also known as sparse features, these features exploit the CRF top layer to capture contextual information through feature sets that take into account tokens' position, POS tags, presence in the place-name dataset, presence in the location-indicative noun dataset, and locative markers. We took into account the current token form t_i, its previous token form t_{i-1} and the following token form t_{i+1}; the POS tag of the current token pt_i, the POS tag of the previous token pt_{i-1} and the POS tag of the following token pt_{i+1}; the combination of these token forms with their POS tags; the presence in the place-name dataset pn_i of the current token in combination with its POS tag, the presence in the place-name dataset pn_{i-1} of the previous token in combination with its POS tag, and the presence in the place-name dataset pn_{i+1} of the following token in combination with its POS tag; the presence in the location-indicative noun dataset li_i of the current token together with its token form and the presence in the location-indicative noun dataset li_{i+1} of the following token together with its token form; and whether the current token is a locative marker lm_i, whether the previous token is a locative marker lm_{i-1}, and whether the following token is a locative marker lm_{i+1}. All these rules provided the best results considering the cost-benefit ratio, since adding rules is costly in terms of computational resources and time in the training and evaluation phases. Table 32 provides the configuration of template features. An explanation of the notation used can be found in Section 6.2.1.3.

Table 32. Template features for the extended nLORE model.

Prefix	Feature type	Rule string features
U01:%x	Unigram token	[-1,0]
U02:%x	Unigram token	[0,0]
U03:%x	Unigram token	[1,0]
U04:%x	Unigram POS tag	[-1,1]
U05:%x	Unigram POS tag	[0,1]
U06:%x	Unigram POS tag	[1,1]
U07:%x	Unigram POS tag	[-1,0]/%x[-1,1]
U08:%x	Unigram POS tag	[0,0]/%x[0,1]
U09:%x	Unigram POS tag	[1,0]/%x[1,1]
U10:%x	Unigram place-name	[-1,2]/%x[-1,1]
U11:%x	Unigram place name	[0,2]/%x[0,1]
U12:%x	Unigram place name	[1,2]/%x[1,1]
U13:%x	Unigram location-indicative noun	[0,3]/%x[0,0]
U14:%x	Unigram location-indicative noun	[1,3]/%x[1,0]
U15:%x	Unigram locative marker	[-1,4]
U16:%x	Unigram locative marker	[0,4]
U17:%x	Unigram locative marker	[1,4]

Since one of our aims was to assess the impact on performance of additional linguistic-based features other than token form and POS tag, we also trained our model with only these two features, which meant deleting the other ones (Table 33).

Table 33. Template features for the basic nLORE model.

Prefix	Feature type	Rule string features
U01:%x	Unigram token	[-1,0]
U02:%x	Unigram token	[0,0]
U03:%x	Unigram token	[1,0]
U04:%x	Unigram POS tag	[-1,1]
U05:%x	Unigram POS tag	[0,1]
U06:%x	Unigram POS tag	[1,1]
U07:%x	Unigram POS tag	[-1,0]/%x[-1,1]
U08:%x	Unigram POS tag	[0,0]/%x[0,1]
U09:%x	Unigram POS tag	[1,0]/%x[1,1]

To understand how template features work, let us consider the example *John lives in the south of New York*. If the current token is *south*, the features generated would correspond to the following:

Table 34. Features generated for the example in the extended nLORE model.

Prefix	Features generated
U01:	the
U02:	south
U03:	of
U04:	DT
U05:	NNS
U06:	IN
U07:	the/DT
U08:	south/NNS
U09:	of/IN
U10:	0/DT
U11:	1/NNS
U12:	0/IN
U13:	0/south
U14:	0/of
U15:	0
U16:	1
U17:	1

Since *south of* is a locative marker, the locative-marker phrase was assigned the Boolean value 1. *South* was also a word found in the place-name dataset, hence its Boolean value 1 in the place-name dataset feature column.

5.4.3.2. Context template features

Here we explicitly indicate whether the template features can also apply to tokens other than the current one. The context window takes into account the preceding token, the current token, and the following token, so that the previous features can be combined for the previous and following tokens, too. Considering the example from the previous section, features would be

generated, not only to *south*, but also to the previous token *the* and the following token *of*. This feature is computationally heavy and costly, too, since the wider the contextual window, the more time and computational resources the training phase consume. Considering the cost-benefit ratio, widening the contextual window was discarded.

5.4.3.3. Word embedding features

These features, also known as dense features, provide rich semantic information in the training phase. We used a different corpus for the training of the word embeddings, one which was unlabeled and much larger, having 3,844,612 English tweets, used in and collected by Cheng et al. (2010)[21]. The settings used were the following:

Table 35. Word embeddings settings.

Vector dimensional size	Context window (no of words)	Minimum frequency	Occurrence threshold	Continuous bag of words	No of threads	Save step	Negative examples	Iterations
200	5	5	1e-4	No (skipgram model)	1	100M	15	5

These dense features were fed into the training phase of nLORE. Table 36 shows an example of the semantic and syntactic information captured by the word embeddings for the token *city*.

Table 36. Semantic similarity of *city* with respect to other words in the word embeddings.

Words	Cosine distance
state	0.709802897326614
valley	0.697769563451627
region	0.68459698768071
village	0.681373935435019
area	0.681025145287523
town	0.675501859846169
neighborhood	0.675385347272232
county	0.673755104766063
lobby	0.644855317278645
desert	0.642095013376314
western	0.639060831540725
country	0.636974015778523
border	0.633752032202236
river	0.632661068591006
forest	0.622150292139476

Those words with higher cosine distance values were words which had a similar syntactic behavior in the corpus and thus a certain degree of semantic similarity. This could be helpful in the training process of nLORE to attach more importance to, for instance, words that are related

[21] Available on the following link: https://archive.org/details/twitter_cikm_2010

to location-indicative words, even when, for any reason, they were not captured by the location-indicative noun dataset, thus helping in the task of location extraction. Also, since we selected the skip-gram model, location-indicative nouns could also signal the high probability of neighboring tokens that are part of or that predict a given a locative reference, such as prepositions, proper nouns, etc.

We specified a context window for word embeddings that takes into account the current token, the past two tokens, and the following two tokens. This context window was found optimal for our purposes.

5.4.4. Hyperparameterization: parameter tuning and settings

One key step in training a model is the process of hyperparameterization where parameters are adjusted to achieve optimal performance. The model structure of the deep bidirectional RNN with LSTM and CRF was specified in the training process. Table 37 shows the parameters used in each of the trained models.

Table 37. Parameter settings in trained nLORE models.

Paramaters	**Values**
Network type	Bidirectional
Dropout	0.5
Hidden layer type, numer, and size	LSTM, 1, 200 neurons
Output layer type	Simple, CRF
Learning rate	0.1
Minibatch size	16
Save step	200K
Maximum iterations	0 (unlimited)

Models were saved when they achieved the best result in the valid corpus. Iterations ranged from 20 to 25: the larger the corpus, the greater number of feature types that were taken into account, the greater time it took to train each model, from a few hours to more than one day. We also used the English valid corpus which was used to validate the results obtained in each iteration for the automatic tuning of the model in the following iterations.

6. IMPLEMENTATION

In this section we describe the computational implementation of our models and different tools used for different purposes. We describe the LORE tool in Section 6.1. and the nLORE tool in

Section 6.2., explaining the files and other apps required for them to operate. In Section 6.3. we present the evaluation tool used for the evaluation stage of the models.

6.1. LORE

The pipeline of LORE was computationally implemented from scratch in the C# programming language using the .NET framework[22]. This involved learning the basics of programming, such as variables, data types and structures, the syntax that work with these (e.g. conditionals, loops) and methods, and applying all this knowledge in the development of the applications. The functionalities of our models were encapsulated in a static class, where we specified the different fields, properties, internal classes for tokens and tweets, and methods for different functions such as reading files, determining the language of the tweets, pre-processing the tweet text, tokenizing and POS tagging the tweets, loading the language-specific datasets, performing the different sub-modules, saving the locative references in a file, etc. Figure 18 shows a glimpse of the code from the method in the LORE static class that performs the task of locative reference extraction from a given corpus of tweets.

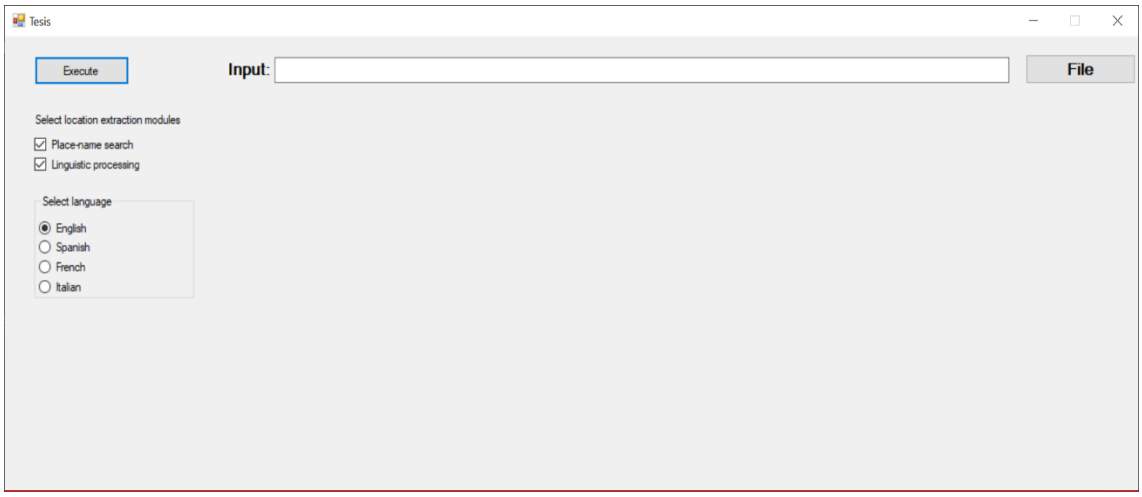
Figure 18. A glimpse of the coding of LORE.

```
//Location-detection tasks: i) Preprocessing, ii) Tokenization and POS tagging, iii) place-name search, and iv) linguistic processing
3 references
public static void LOcativeReferenceExtractor(string filepath, string language, bool placeNameSearchModule, bool linguisticProcessingModule)
{
    if (language == "English" || language == "Unknown") { ISOlanguage = "en"; }
    else if (language == "Spanish") { ISOlanguage = "es"; }
    else if (language == "French") { ISOlanguage = "fr"; }

    string originalTweets = ReadFile(filepath);
    string[] splitTweets = originalTweets.Split(new string[] { Environment.NewLine }, StringSplitOptions.RemoveEmptyEntries);
    CSVtweets(splitTweets); //tweet corpus converted into a more readable format: a csv file with fields ID | tweet
    language = CheckTweetLanguage(splitTweets, language);
    LoadRulesandGazetteers(language);

    //Module 1: Pre-processing stage: tweet cleaning + word segmentation algorithm
    string processedTweets = MicrotextPreprocessing(splitTweets);

    //Module 2: Tokenization and POS tagging
    string postaggedTweets = TokenizerPOSTagger(processedTweets, language);

    //Store tweets object list and their properties: original word form, pos tag, and word position
    StoreTweetObjects(postaggedTweets);
    tokenObjectList = CreateTokenObjectList(postaggedTweets);
    locRefs = new List<LocRefs>();

    //Module 3: place-name search: i) load preprocessed GeoNames file/download GeoNames file and pre-process it and ii) perform n-gram-based matching with n-gram algorithm and GeoNames database. This module works for
    if (placeNameSearchModule)
    {
        LoadGeoNamesDatabase();
        EntityBasedMatching();
    }
    //Module 4: Linguistic processing (i.e. language-specific)
    if (linguisticProcessingModule)
    {
        LocPrepsPlaceNames();
        LocIndWords(language);
        LocativeMarkers(language);
    }
    //Output
    locRefs = locRefs.OrderBy(q => int.Parse(q.ID)).ThenBy(i => i.StartPos).ToList();
    using (CsvWriter csv = new CsvWriter(new StreamWriter(dataDir + @"\output\" + "LORE_output" + DateTime.Now.Ticks + ".csv", false, Encoding.GetEncoding("iso-8859-1"))))
    {
        csv.Configuration.HasHeaderRecord = true;
        csv.Configuration.Delimiter = "|";
        csv.WriteRecords(locRefs);
    }
}
```

The application's User Interface (UI) of LORE is presented in Figure 19.

Figure 19. UI of LORE.

[22] LORE is freely accessible from the FunGramKB website (http://www.fungramkb.com/nlp.aspx).



Pressing the *File* button lets us browse our files in search of a TXT file (i.e. the pre-processed tweet corpus), which is then loaded as the input file. Afterwards, we can select which location extraction modules apply and the language of the tweets. The *Execute* button is in charge of processing the corpus up to the extraction of locative references. Table 38 shows the files that are externally received (i.e. input files) and generated (i.e. output files) in LORE.

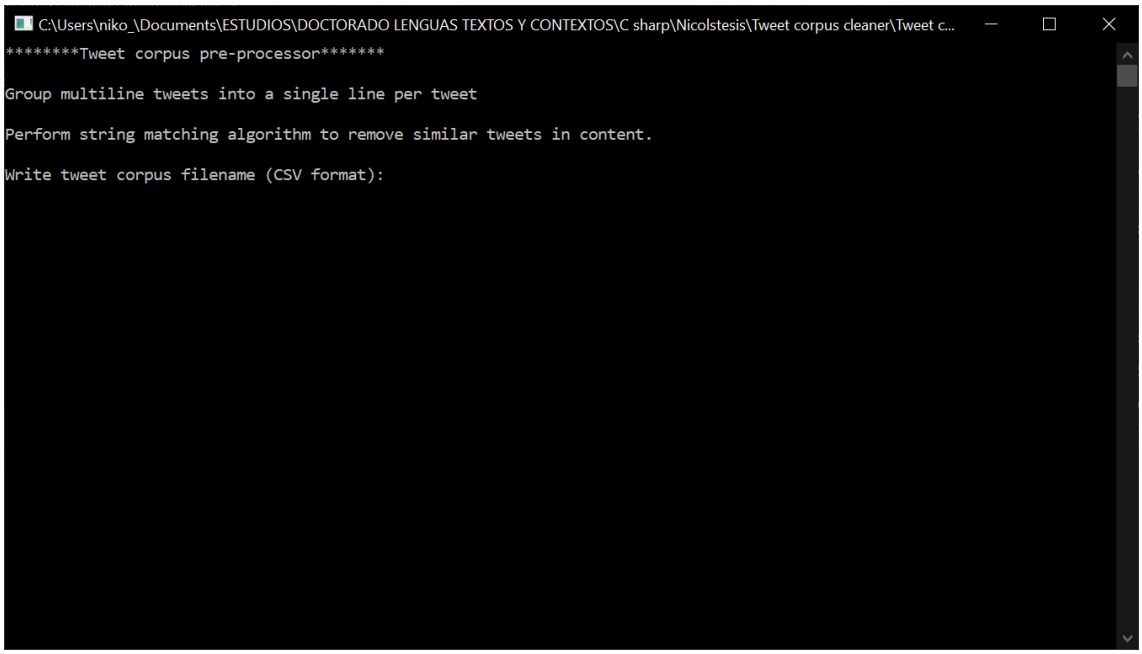
Table 38. Input and output files in LORE.

File type	Function	Input/Output
Preprocessed tweet corpus (TXT file)	File loaded by the end user with tweet texts, where each line represents a given tweet.	Input
Tweet corpus (CSV file)	Each tweet is marked by its ID number and its text in each line.	Output
Locative references (CSV file)	Locative references are extracted and organized according to their tweet ID.	Output

The preprocessed tweet corpus was obtained out of the raw tweet corpus by means of our tweet corpus pre-processor tool. Also developed in C#, our command-line tool (Figure 20) makes use of the capabilities offered by the StringSimilarity.NET library[23], in particular the Levenshtein distance or the cosine distance using n-grams. The tool loads as input the CSV file delivered by the FireAnt app and obtains as output the preprocessed tweet corpus in TXT format for its processing in LORE or nLORE.

Figure 20. Our tweet corpus pre-processor command-line tool.

[23] https://github.com/feature23/StringSimilarity.NET



6.1.1. Internal files in LORE

LORE requires several files to work properly. Table 39 provides a summary of the internal files required by LORE to operate.

Table 39. Internal files required by LORE, their function, and data type or structure.

File type	Function	Data type/structure
ISO language codes (TXT file)	Facilitates the automatic retrieval of the language-specific datasets in case these are missing and prepares LORE for expanding its multilingual adaptation in the future.	String
Configuration file (TXT file)	Provides regexes with language-specific POS tags and locative prepositions chosen for each of the languages supported.	IEnumerable
Place-name dataset (CSV file)	Stores place names, obtained and pre-processed from GeoNames.	Hashset
Location-indicative noun dataset (CSV file)	Stores location-indicative nouns, obtained and pre-processed from EuroWordNet using our EuroWordNet hyponym extractor tool.	Hashset
Place abbreviation list (CSV file)	Stores place abbreviations for nouns contained in the location-indicative noun dataset.	Hashset
Locative marker dataset (CSV file)	Stores the locative markers manually provided by the end user.	Hashset
Stopword dataset (CSV files)	Composed of two files: common words CSV file and names and surnames CSV file(s). They are used to filter false positives.	Hashset

Frequency dictionary (TXT file)	Used for the word segmentation algorithm.	String

All are language-dependent except the ISO language codes file. Also, in all the files each line represents a given entry of an item. If the language-dependent datasets are empty or absent, the program automatically retrieves and generates those language-independent resources, except the place abbreviation list and the locative marker dataset that need to be manually supplied. Those files with the largest number of entries are loaded in the program as hashsets because hashsets can perform faster lookups than lists, arrays, and other data structures. The following subsections explain the nature of the files for the datasets used.

6.1.1.1. The configuration file

Language-specific divergences can be handled by the end user in a configuration file by providing the language-specific POS tags and locative prepositions, which are subsequently fed into the built-in regex-based rules. For this, an external file is available for the end user to provide or modify the corresponding POS tags of their language, and those most relevant locative prepositions used in the linguistic processing task of the model

The layout of the configuration files follows the conventions used in regexes, since this grammatical and lexical information is directly fed into the system's rules. ^ and $ indicate the beginning and end of a string, respectively. The asterisk * is a quantifier whose function is to match the preceding symbol or character zero or more times. The dot . matches any character except line breaks. Round brackets are used to capture a multiple group of strings where the symbol | acts as the logical operator OR. For instance, in Table 40, Table 41, and Table 42 we can observe the specified POS tags and locative prepositions chosen for English, Spanish, and French in their configuration files.

Table 40. English config file.

POS tags and locative prepositions	**Regex**
Common noun POS tag	^NNS?$
Proper noun POS	^NNPS?$
Preposition POS tag	^IN$
Determiner POS tag	^DT$
Definite determiner POS tag	^DT$
Locative prepositions	^(at|in|near|along|across|@)$

Table 41. Spanish config file.

POS tags and locative prepositions	**Regex**
Common noun POS tag	^nc.*$
Proper noun POS	^np.*$
Preposition POS tag	^sp.*$
Determiner POS tag	^d.*$

Definite determiner POS tag	^da$
Locative prepositions	^(en|hacia|hasta)$

Table 42. French config file.

POS tags and locative prepositions	Regex
Common noun POS tag	^(NC|N)$
Proper noun POS	^NPP$
Preposition POS tag	^P$
Determiner POS tag	^DET$
Definite determiner POS tag	^DET$
Locative prepositions	^(à|au|aux|en|vers)$

6.1.1.2. The place-name dataset files

For the extraction of place names, we used the third-party C# library NGeoNames[24] integrated in the static methods in LORE. The language parameter was automatically set, according to the language of the tweet corpus, to obtain place names of that language. For instance, for English, it downloaded a very large file that contained English place names for all countries and another file for alternate names (both weighing around 1.92 GB in total). Not only did it include place names, but it also contained information in relation to the type of location, geographic coordinates, etc. The pre-processing and filtering steps were automatically performed using LINQ queries and regexes due to the extremely large size of the files and the possible performance drops derived from using those files in memory, and also due to unnecessary data that these files contained. This pre-processing step generated a much smaller file of only 12.4 MB for English place names, 3.62 MB for Spanish place names, and 1.45 MB for French place names. If our program did not detect these files, it would automatically retrieve them and perform the pre-processing step. Also, thanks to the ISO language codes, we could automatically compile place-name datasets for many other languages. Figure 21 shows a screenshot of the original GeoNames files.

Figure 21. A screenshot of the English place-names file retrieved from the GeoNames database.

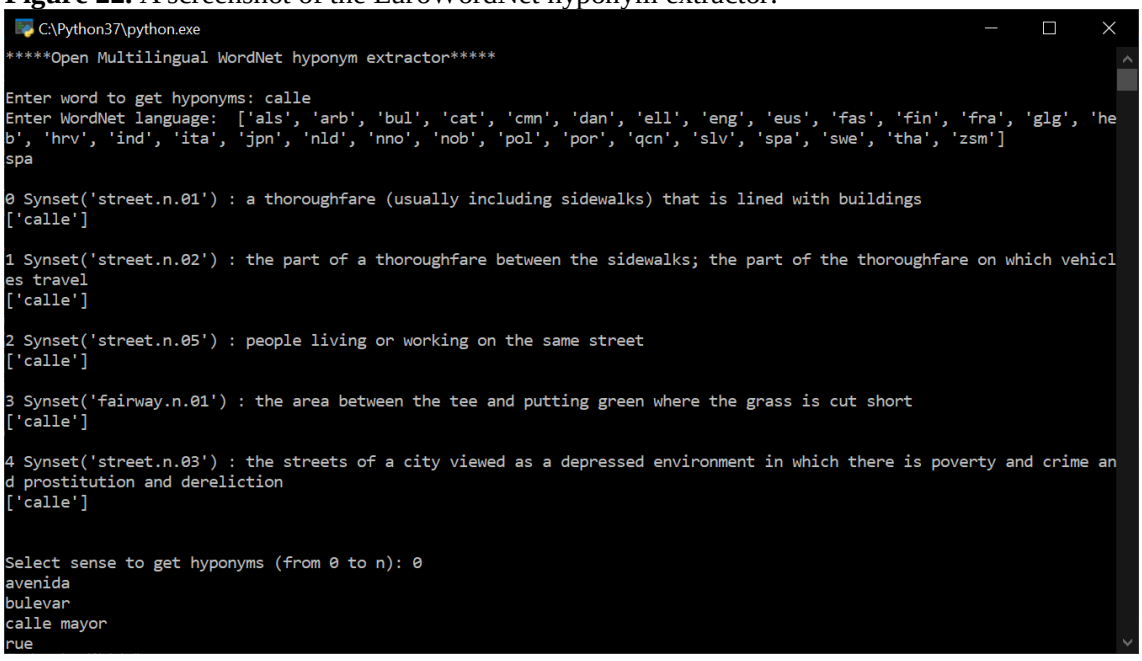
[24] NGeoNames: https://github.com/RobThree/NGeoNames

3349070	Cupa	Cupa		-15.6	17.3	P	PPL	AO		4			0	1280	Africa/Luandi	31/01/1994
3349071	Cunzumbia	Cunzumbia	Cunzumbia,K	-15.61667	19.85	H	STM	AO	AO	4			0	1191	Africa/Luandi	17/01/2012
3349072	Cunze	Cunze	Cunze,Rio Cu	-13.33333	18.51667	H	STM	AO	AO	14			0	1308	Africa/Luandi	17/01/2012
3349073	Cunjo	Cunjo	Condjo,Cund	-16.1	14.05	P	PPL	AO	AO	9			0	1219	Africa/Luandi	17/01/2012
3349074	Cunjo	Cunjo	Cundjo,Cunjo	-11.15782	14.56367	P	PPL	AO		6			0	1177	Africa/Luandi	08/10/2018
3349075	Cunjo	Cunjo	Cunjo,Ecungo	-12.15	14.36667	T	MT	AO	AO	1			0	1489	Africa/Luandi	17/01/2012
3349076	Cunje	Cunje	Candeje,Cun	-11.71667	17.58333	H	STM	AO	AO	2			0	1255	Africa/Luandi	17/01/2012
3349077	Cunini	Cunini	Cunini,Rio Cu	-14.46575	19.64284	H	STM	AO		4			0	1156	Africa/Luandi	17/01/2012
3349078	Cunhinga	Cunhinga	Cunhinga,Mu	-11.97695	16.81817	A	ADM2	AO		2	3349078		0	1755	Africa/Luandi	06/08/2015
3349079	Cunhinga	Cunhinga		-10.70209	16.68576	H	STM	AO		2			0	1059	Africa/Luandi	09/07/2011
3349080	Cunhinga	Cunhinga	Cunhinga,Do	-12.23333	16.78333	P	PPL	AO	AO	2			0	1772	Africa/Luandi	17/01/2012
3349081	Cunhangamu	Cunhangamu	Cunhangamu	-13.3	15.7	H	STM	AO	AO	8			0	1655	Africa/Luandi	17/01/2012
3349082	Cunhangama	Cunhangama	Cunhangama	-11.75	15.4	H	STM	AO	AO	8			0	1322	Africa/Luandi	17/01/2012
3349083	Cungungo	Cungungo	Bango,Cungu	-12.11667	15.65	P	PPL	AO	AO	8			0	1434	Africa/Luandi	17/01/2012
3349084	Cunguigi	Cunguigi	Cunguige,Cur	-10.28638	15.86275	H	STM	AO		6			0	1115	Africa/Luandi	17/01/2012
3349085	Cunguene	Cunguene	Cunguene,Ri	-12.38333	19.51667	H	STM	AO	AO	14			0	1228	Africa/Luandi	17/01/2012
3349086	Cungo	Cungo		-12.7	14.03333	P	PPL	AO		1			0	797	Africa/Luandi	31/01/1994
3349087	Cungo	Cungo		-12.7	14.01667	P	PPL	AO		1			0	692	Africa/Luandi	31/01/1994
3349088	Cungo	Cungo		-12.7	14	P	PPL	AO		1			0	661	Africa/Luandi	31/01/1994
3349089	Cungo	Cungo		-10.36667	16.71667	P	PPL	AO		12			0	1063	Africa/Luandi	31/01/1994
3349090	Cunga	Cunga	Bamba-Gung	-14.7	15.36667	P	PPL	AO	AO	9			0	1317	Africa/Luandi	17/01/2012
3349091	Cunga	Cunga		-11.35	14.21667	P	PPL	AO		6			0	1048	Africa/Luandi	31/01/1994
3349092	Cunga	Cunga		-11.53333	14.26667	T	MT	AO		6			0	996	Africa/Luandi	31/01/1994
3349093	Cunenga	Cunenga		-11.5	14.15	H	STM	AO		6			0	314	Africa/Luandi	31/01/1994
3349094	Cunenga	Cunenga	Cunenga,Cun	-11.48115	15.10041	H	STM	AO		6			0	1278	Africa/Luandi	17/01/2012
3349095	Cunenga	Cunenga		-11.58333	14.25	P	PPL	AO		6			0	896	Africa/Luandi	31/01/1994

6.1.1.3. The location-indicative noun dataset files

The extraction of location-indicative nouns in each of the languages supported was carried out through a EuroWordNet hyponym extractor command-line tool developed in Python using the NLTK library. First, we enter a given word to get its hyponyms. Usually, this would be a general word of locative meaning. Then, we enter the language using any of the ISO codes. Afterwards, we select the synset in which we are interested. We are offered a list of hyponyms and then we can save them into a CSV file. In Figure 22 we can see all these steps to obtain the hyponyms of the synset that refers to the Spanish word *calle*.

Figure 22. A screenshot of the EuroWordNet hyponym extractor.



If any of location-indicative noun dataset files were missing, we integrated in the static class of LORE a method that would perform the same actions taken by our command-line tool, using the synsets described in Section 5.3.2.3. to generate and process the dataset files in an automatic manner.

6.1.1.4. Place abbreviation list and locative marker dataset files

These are the only files that need to be manually supplied in LORE. If the place abbreviation list file were missing, LORE would only take into account the location-indicative noun dataset. If the locative marker dataset file were absent, the submodule in charge of processing locative markers would not perform its task.

6.1.1.5. The stopword dataset and frequency dictionary files

They were manually obtained for the sources cited in Section 5.3.2.5. If the stopword dataset files were missing, the files would be automatically generated from (a) the frequency dictionary, which is in turn automatically retrieved from a GitHub repository[25], and (b) from a list of person names from another GitHub repository[26].

6.2. nLORE

nLORE was also computationally implemented in C#. Departing from LORE, we used a different functioning under the hood and different elements in the UI. In this case, since we had to train and test our DL models, we embedded the functionalities of the RNNSharp library[27] in our code for the training and testing phases. With RNNSharp we can employ neuronal networks suited for sequence-labeling tasks such as NER. Specifically, we made use of its functionalities to implement a deep bidirectional RNN-CRF network with LSTM for the hidden layer structure. It supports different feature types, which we leveraged for linguistic-based feature engineering. On the other hand, Text2Vect provides friendly ready-made functionalities to build a word-embeddings model on the basis of an unlabeled corpus, generating vectors for the words and phrases contained in that corpus. It can also display the cosine similarity among words captured by the generated model in its command-line application. The generated word-embeddings model was used as dense features in nLORE.

Figure 23 shows the UI of nLORE, where we select (a) *Automatic labeling* if we want to format our tweet corpus in a token-based tabular representation using the capabilities of LORE to represent the linguistic-based features selected and using the BMESO tagging scheme for the

[25] https://raw.githubusercontent.com/hermitdave/FrequencyWords/master/content/2018/
[26] https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/packages/corpora/names.zip
[27] https://github.com/zhongkaifu/RNNSharp

labels of tokens, (b) *Training* if we want to train our location-detection model, and (c) *Testing* if we want to test our model.

Figure 23. UI of nLORE.

For obvious reasons, depending on the option that we choose, we would need to browse and select the files required for such option. For automatic labeling, we browse and select our raw corpus, and select the linguistic-based features that are to be represented in that newly formatted corpus. After clicking on *Execute*, the modules of LORE would be in charge of representing the linguistic-based features and the labels for each token, generating a TXT file with the newly formatted corpus. Afterwards, since LORE does not have a 100% accuracy, the end user would have to manually revise the labels to check for erroneous labels and assigning the right ones. For training, we browse and select a train corpus and a valid corpus with the token-based tabular representation, and enter the training hyperparameters. After the training phase, we would obtain as output the location-detection model in a BIN format together with the template features generated by the train corpus in DART and TEMPLATE formats. For testing, we need the eval corpus without the last column of labels, since the aim of the testing phase is to load the trained model and predict the correct labels for that corpus. The trained model file and template files would be used as input here. After the testing phase, a newly formatted corpus would be generated with the predicted labels for each of the tokens. Table 43 summarizes the input and output files of nLORE.

Table 43. Input and output files in nLORE.

File type	Function	Input/Output
Raw corpus (TXT file)	File loaded by the end user with tweet texts, where each line represents a given tweet, to generate a token-based tabular representation of the tweets and assigning the linguistic-based features and labels provided by the modules of LORE.	Input

Train corpus (TXT file)	Formatted corpus of tweets with features and correct labels for the training phase of the model.	Input
Valid corpus (TXT file)	Formatted corpus of tweets with features and correct labels for the validation phase of the model in its training stage.	Input
Eval corpus (TXT file)	Formatted corpus of tweets with features but without the last column of labels for the testing phase of the model.	Input
Trained model (BIN file)	The output file after the training phase. It acts as input in the testing phase, since it will be used for predicting the labels. It is usually large, taking a few gigabytes.	Input and output
Template features (DART and TEMPLATE files)	The output files after the training phase. They act as input in the testing phase, since they will be used for predicting the labels.	Input and output
New eval corpus (TXT)	Formatted corpus of tweets with features and predicted labels after the testing phase of the model.	Output

6.2.1. Internal files in nLORE

Besides the internal files of LORE (Table 38), nLORE needs a few internal files to operate, as summarized in Table 44, and explained in the following subsections.

Table 44. Internal files required by nLORE, their function, and context of usage.

File type	Function	Context of usage
Neuronal network configuration (TXT file)	Contains essential info for the training and testing phases of the models, such as model type, neuronal network type, and other hyperparameters.	Training and testing
Tags (TXT file)	Provides the labels used in the tagging scheme of tokens (e.g. BMESO, IOB).	Training and testing
Template features (TXT)	Specifies contextual features based on the linguistic-based features of the corpora used.	Training
Word embeddings (BIN file)	Generated by the Txt2Vec command-line tool, it is used as dense features in the training phase.	Training

6.2.1.1. Neuronal network configuration file

This file, as used in RNNSharp, encapsulates the nature of the neuronal network used, other hyperparameters used for the training phase, and the specification of dense features and other

types of features in the model. Table 45 provides the different parameters that can be specified in the configuration file for the training and testing of the neuronal network.

Table 45. Settings, functions and possible options in the neuronal network configuration file.

Specification	Function	Parameters
Model type	The type of model used.	Sequence labeling (e.g. used in NER)/sequence to sequence (e.g. used in automatic translation)
Network type	The type of neuronal network used.	Forward RNN/Bidirectional RNN/Forward seq2seq
Model file path	The file path of the trained model for the testing stage.	-
Dropout	Hyperparameter used for the training stage.	Default value: 0.5
LSTM and hidden layer settings	Number of LSTM layers used and size.	As many as desired with as many neurons as desired. Default value is one LSTM layer of size 200.
Output layer settings	Defines nature of the output layer.	Simple/softmax/sampled softmax
CRF layer	Specifies whether we use a CRF layer on top or not.	True/false
Template features	Filename, contextual window of tokens, and weight type used in the template features for the training and testing stages.	Default value contextual window: 0,1,2 Weight type: binary/frequency
Pre-trained features type	Used as dense features in the training and testing stages of the model.	Embedding/autoencoder
Word embeddings file path, context window and column	The file path of the pre-trained word embeddings, the contextual window of tokens and the column in the corpus where they are applied.	Default value contextual window: -1,0,1 Default column: 0
Runtime features	Takes into account tokes in training sequence to sequence models, does not apply to sequence to label tasks.	Default value: -1

6.2.1.2. Tags file

We indicate the tagging scheme used in the corpora and necessary to predict the labels in the testing stage, by writing the initial letters of each label, and the nature of the labels (i.e. B_LOCATION, M_LOCATION, E_LOCATION, S_LOCATION, O).

6.2.1.3. Template features file

Known as sparse features, they are specified in the file following the notation of N*n*:%x[*i*,*j*], where N represents the prefix for n-grams (e.g. unigram, bigram), *n* the ID number, and %x[*i*,*j*] the rule-string feature where *i* and *j* represent a specific row (e.g. preceding token, current token, following token) and column (e.g. POS tag, locative marker) of features in the corpora, respectively. We can also combine rule-string features by adding a slash (e.g. U04:%x[-

1,0]/%x[0,0]). In this way, we can provide different feature combinations to be exploited by the CRF layer for predicting the right labels.

6.2.1.4. Word embeddings file

The pre-trained word embeddings, used as dense features, can be obtained using the Txt2Vec library[28], based on Word2Vec. Using the command-line tool of that library and a corpus, we can extract those word embeddings, generating a BIN file that is fed into the training stage to capture the semantic and syntactic information of tokens. Table 46 presents the settings that can be adjusted to train a word embeddings model with Txt2Vec.

Table 46. Word embedding settings in Txt2Vec, functions, and parameters.

Setting	Function	Parameters
Vector dimensional size	Sets the number of dimensions of word vectors.	Default value: 200
Context window	Sets the maximum number of words skipped in a context window.	Default value: 5
Minimum frequency	Sets the minimum number of words to be taken into account.	Default value: 5
Occurrence threshold	Minimizes or downsamples those words which appear much more frequently, such as grammatical words.	Default value: 0 (off)
Continuous bag of words	Sets the continuous bag of words model or skip-gram model.	skip-gram model/continuous bag of words model
Number of threads	Sets the number of threads.	Default value: 1
Save step	Saves the model after a number of words have been processed.	Default value: 10^8
Negative examples	Sets the number of negative examples.	Default value: 5
Iterations	Sets the number of training iterations.	Default value: 5
Train file	Specifies file path of corpus for training word embeddings.	-
Model file	Specifies name and file path of generated word embeddings.	-
Pre-trained model file	Specifies file path of pre-trained word embeddings	-
Update corpus words	Updates corpus words or all the words.	0 (update all the words)/1 (update corpus words only)

6.3. Evaluation tool

To perform our evaluation stage, two main steps were carried out: (a) we implemented the functionalities of well-known, general-purpose, off-the-shelf NER tools in the UI of LORE for a comparison of the performance of LORE against these tools and (b) we developed a command-

[28] https://github.com/zhongkaifu/Txt2Vec

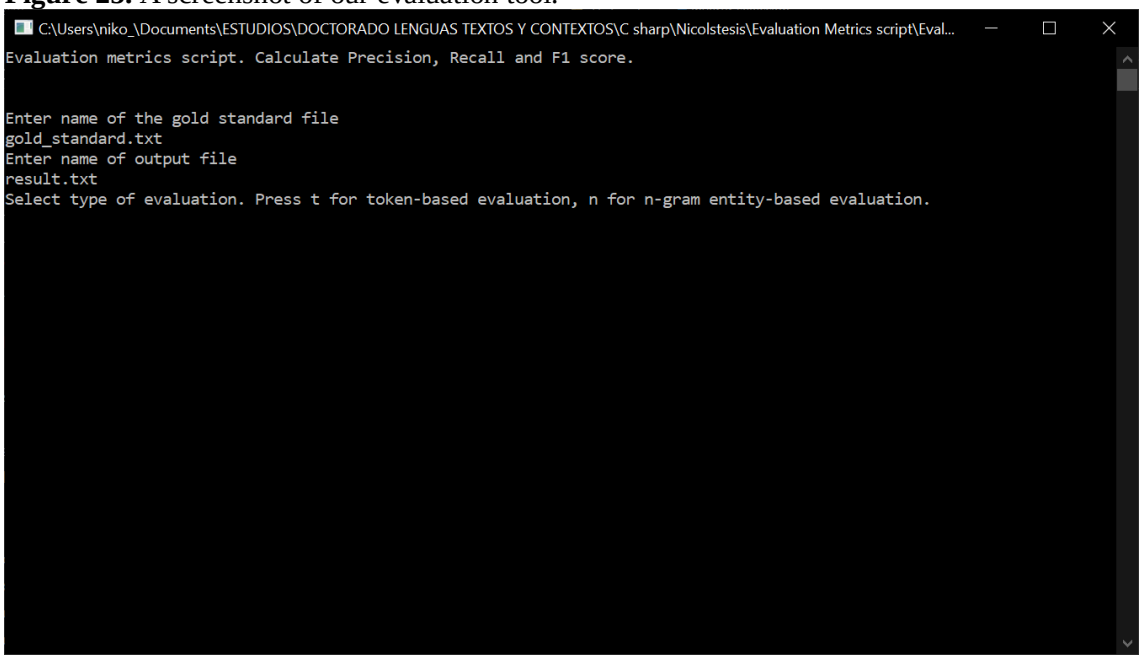
line tool that provides the evaluation numbers achieved by every model or tool. Figure 24 shows the implementation of these NER tools in the UI of LORE.

Figure 24. UI of LORE with the implemented NER tools.

The same eval corpus fed into LORE was fed into these NER systems with the same pre-processing techniques applied to the tweet corpus. For obvious reasons, considering that each NER software includes their own tagging and output format, their output had to be automatically standardized and accommodated to the format adopted by our model (see Table 5 in Section 5.2.), so that comparison could be effected under the same circumstances. Further details about the comparison stage are given in Section 7.1.1.2.

Figure 25 presents a screenshot of the command-line tool developed for our evaluation purposes.

Figure 25. A screenshot of our evaluation tool.



Our evaluation tool receives as input the gold standard of locative references of the tweet corpus (i.e. the format used for LORE) or a corpus of tweets with their correctly labeled locative references (i.e. the format used for nLORE). Then, we also provide as input the files of the output delivered by LORE, nLORE, or the other NER tools, which are processed in the appropriate format. Next, we select the type of evaluation that we wish to conduct, after which we are presented with the evaluation numbers, which can be saved as an output file. Further details about the different types of evaluations and the evaluation results are given in Section 7.

7. EVALUATION

We present the evaluation metrics for the evaluation stage of our model. We followed the evaluation measures that are most widely used in Information Extraction for NER: precision (P), recall (R), and the F1 measure, which is the harmonic mean of precision and recall (Grossman & Frieder, 2004; Jurafsky & Martin, 2018a):

$$Precision = \frac{TP}{TP+FP} \quad (18)$$

$$Recall = \frac{TP}{TP+FN} \quad (19)$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad (20)$$

True Positive (TP) refers to a correctly identified locative reference. False Positive (FP) refers to instances that have been wrongly identified as locative references. False Negative (FN) is the label used for those locative references that were missed. Departing from these variables and depending on where the focus is placed, we arrive at different evaluation metrics, i.e. precision, recall and F1. All these measures range from 0 to 1.

The evaluation process was performed on the English, Spanish, and French eval corpora of tweets following the metrics presented above. The evaluation stage was carried out with two different evaluation methods: entity-based evaluation and token-based evaluation (Gritta et al., 2018), also called complete and fuzzy matching, respectively (Das & Purves, 2019). In both methods, exact matches count as TP. However, they differ in the treatment of cases of partial or inexact matches. In entity-based evaluation, partial matches are penalized, since they count either as FP when the boundaries of the extracted instance exceed the boundaries of the locative reference (e.g. *Off East Coast of Honsu* instead of *East Coast of Honsu*) or as FN when the boundaries of the extracted instance fall short (e.g. *Camino* instead of *Camino Pablo*). In token-based evaluation, partial matches of the type commented above, besides counting as FP or FN, also count as TP. Thus, entity-based evaluation is the strictest evaluation method, and also the

commonest for benchmarking NER systems (Jurafsky & Martin, 2018a). On the other hand, token-based evaluation works more leniently, yielding higher numbers. In general, achieving an F1 score of 0.9 across multiple domains is the ultimate goal of location-detection models for near-human-level competence in location detection (Gritta et al., 2018).

7.1. Results

In Experiment I in Section 7.1.1., we present the results of LORE in the different languages supported, as well as in comparison with well-known, general-domain, off-the-shelf NER tools. In Experiment II in Section 7.1.2., we introduce the results achieved by the trained nLORE models, by LORE with the English eval corpus II, and we benchmark LORE against the trained nLORE models.

7.1.1. Experiment I

Experiment I involved testing LORE, our rule-based approach, with the English eval corpus I, Spanish eval corpus, and French eval corpus, and against other well-known, general-purpose, off-the-shelf NER tools.

7.1.1.1. Multilingual LORE

Table 47, Table 48, and Table 49 below show the P, R, and F1 scores of the evaluation phase performed on the English eval corpus I, the Spanish eval corpus, and the French eval corpus, respectively, following a per-token basis and a per-entity basis. For each of these corpora, we also provide the evaluation measures for the individual working modules, the place-name search module and the linguistic processing module, and as a whole, to observe the contributions made by each of these modules and the substantial improvement in the F1 scores resulting from their combination. The best results for each type of evaluation are highlighted in bold.

Table 47. Evaluation with the English eval corpus I.

	Token-based evaluation			**Entity-based evaluation**		
	P	**R**	**F1**	**P**	**R**	**F1**
Only with place-name search	0.84	0.48	0.61	0.59	0.40	0.47
Only with linguistic processing	**0.90**	0.56	0.69	**0.86**	0.52	0.65
Place-name search + linguistic processing	0.85	**0.83**	**0.84**	0.81	**0.81**	**0.81**

Table 48. Evaluation with the Spanish eval corpus.

	Token-based evaluation	**Entity-based evaluation**

	P	R	F1	P	R	F1
Only with place-name search	0.74	0.56	0.64	0.55	0.49	0.52
Only with linguistic processing	**0.82**	0.41	0.54	**0.70**	0.34	0.46
Place-name search + linguistic processing	0.73	**0.74**	**0.74**	0.64	**0.72**	**0.67**

Table 49. Evaluation with the French eval corpus.

	Token-based evaluation			Entity-based evaluation		
	P	R	F1	P	R	F1
Only with place-name search	**0.95**	0.38	0.54	0.77	0.33	0.46
Only with linguistic processing	0.86	0.34	0.48	**0.82**	0.30	0.44
Place-name search + linguistic processing	0.90	**0.55**	**0.68**	0.81	**0.51**	**0.62**

Table 50, Table 51, and Table 52 provide entity-based evaluation numbers in terms of n-gram size for each of the languages.

Table 50. Evaluation with the English eval corpus I in terms of n-gram size for the place-name search + linguistic processing modules.

N-gram size	P	R	F1
Unigrams	0.78	0.81	0.79
Bigrams	0.85	0.86	0.85
Trigrams	0.88	0.72	0.79
Fourgrams	0.67	0.63	0.65
Fivegrams	0.67	0.8	0.73
Sixgrams	1	1	1
Sevengrams	N/A	0	N/A
Eightgrams	N/A	N/A	N/A

Table 51. Evaluation with the Spanish eval corpus in terms of n-gram size for the place-name search + linguistic processing modules.

N-gram size	P	R	F1
Unigrams	0.67	0.74	0.70
Bigrams	0.52	0.77	0.62
Trigrams	0.74	0.67	0.70
Fourgrams	0.60	0.50	0.55
Fivegrams	0.50	0.33	0.4
Sixgrams	1	0.25	0.4
Sevengrams	N/A	N/A	N/A
Eightgrams	N/A	N/A	N/A

Table 52. Evaluation with the French eval corpus in terms of n-gram size for the place-name search + linguistic processing modules.

N-gram size	P	R	F1
Unigrams	0.83	0.68	0.75
Bigrams	0.79	0.35	0.48

Trigrams	0.71	0.32	0.44
Fourgrams	1	0.25	0.40
Fivegrams	N/A	0	N/A
Sixgrams	N/A	N/A	N/A
Sevengrams	N/A	N/A	N/A
Eightgrams	N/A	N/A	N/A

Figure 26, Figure 27, and Figure 28 provide a graphical representation of the F1 scores in terms of n-grams for each of the languages.

Figure 26. F1 of English LORE in terms of n-gram size.

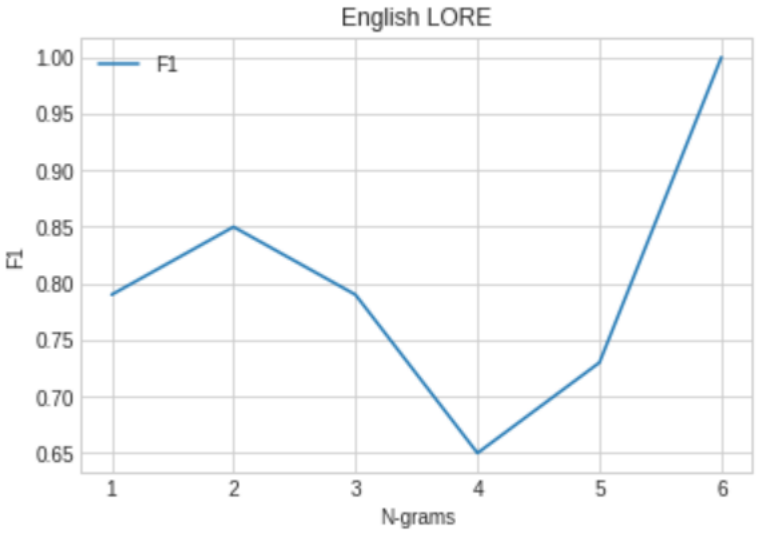


Figure 27. F1 of Spanish LORE in terms of n-gram size.

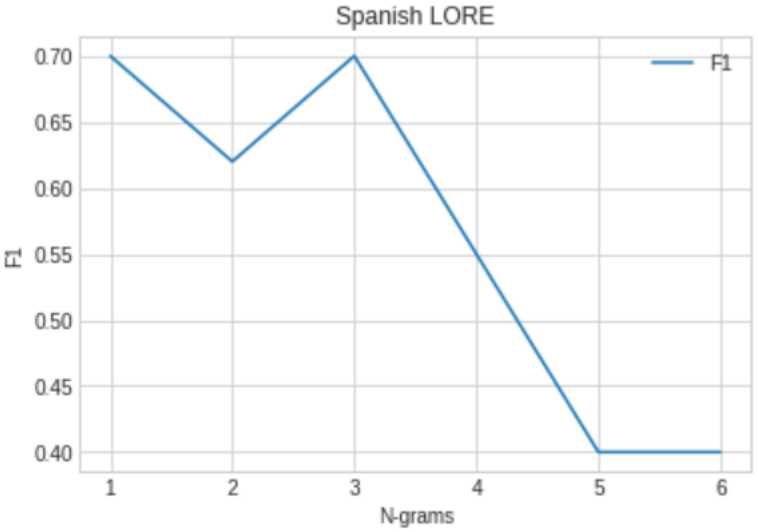
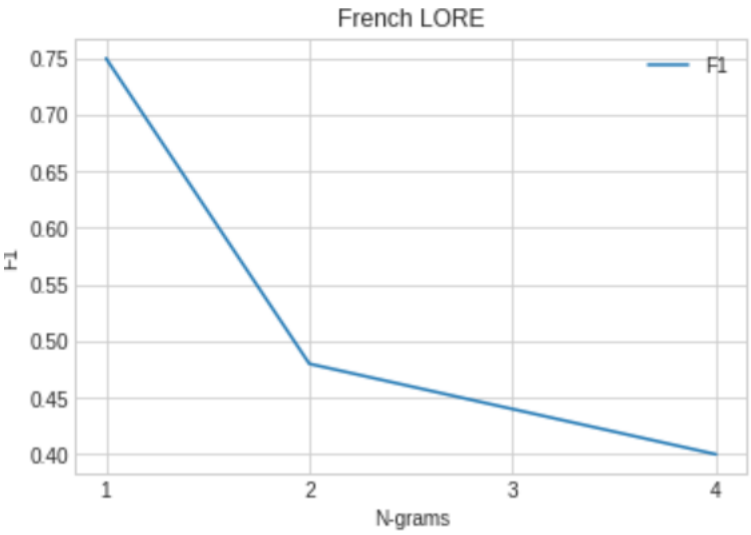


Figure 28. F1 of French LORE in terms of n-gram size.



7.1.1.2. LORE vs off-the-shelf NER tools

We also benchmarked the performance of our model against other well-know, general domain, off-the-shelf NER systems that are commonly employed in benchmarking (Schmitt et al., 2019). This evaluation and comparison stage was performed under the same conditions for an unbiased analysis and study of their performance. In other words, this stage was carried out under the same computing environment (i.e. i5-6200U @ 2.30 GHz with 2 cores and 8GB RAM). Some of these tools did not offer extended support to languages other than English. Before presenting the evaluation numbers of the NER systems, a brief, technical description of them is provided to contextualize the nature of the performance tests:

- Stanford NER: Originally written in Java, we used the Stanford.NLP.NET port that implements a probabilistic algorithm based on a CRF linear classifier for NER in unstructured text (Finkel et al., 2005)[29]. The model makes use of three labels PERSON, LOCATION, ORGANIZATION for named entities. Stanford NER for English is trained on news corpora from CoNLL 2003, MUC 6 and MUC 7, ACE 2002 and additional data. We only considered the entity type LOCATION for the extraction of locative references.
- Natural Language Toolkit (NLTK) 3.4.4 is a Python library for a wide variety of NLP tasks such as tokenization, lemmatization, POS tagging, chunking, NER, semantic tagging, parsing, among others (Bird, 2006)[30]. Its NER module is based on a Maximum Entropy algorithm trained on the ACE corpus (http://catalog.ldc.upenn.edu/LDC2005T09), and uses the entity classes ORGANIZATION, PERSON, LOCATION, DATE, TIME, MONEY, PERCENT,

[29] https://sergey-tihon.github.io/Stanford.NLP.NET/
[30] http://nltk.org/

FACILITY, and GPE. Since our focus is in locative references, the only relevant categories were FACILITY (for POIs), GPE (for geopolitical entities), and LOCATION for any other location type.

- spaCy 2.1.6. is another increasingly popular and widely used library for Python used for many NLP and NLU tasks and applications[31]. The Entity Recognizer component for the English language is based on a deep-learning CNN algorithm trained on OntoNotes 5.0 data (https://catalog.ldc.upenn.edu/LDC2013T19). spaCy can recognize many entity types, of which we selected GPE (for geopolitical entities), FAC (for POIs), and LOC for the remaining location types.
- The Google Cloud Natural Language (GCNL) API is a state-of-the-art commercial platform that provides a free trial for resources and functionalities in NLP and NLU tasks and applications through an accessible Web Service[32]. Some of the functionalities offered are Sentiment Analysis, Syntactic Analysis, Entity Analysis, or Content Classification. Unfortunately, we could not find documentation details about the algorithm implementation of the Entity Analysis functionality. The Entity Analysis component detect many entity types among which we only considered LOCATION and ADDRESS and, from these, those locative references that had the metadata property, since the label LOCATION was also attached to informal, vague and unspecific locative expressions realized as common nouns.
- C# OpenNLP is a C# port of a Java-based NLP tool for basic NLP tasks such as sentence splitting, tokenization, POS tagging, chunking, and NER (Ingersoll et al., 2013)[33]. Its NER system is based on a Maximum Entropy model trained on a variety of corpora such as MUC6, MUC7, ACE, CONLL 2002 and CONLL 2003. The built-in location types are DATE, LOCATION, MONEY, ORGANIZATION, PERCENTAGE, PERSON, and TIME, of which LOCATION was only considered.
- Stanza is a novel Python library for NLP with pretrained neural networks for 66 human languages (Qi et al., 2020). One of the main functionalities in its NLP pipeline is that of NER. The algorithm used for NER is a biLSTM with a CRF layer on top. The English version of NER was trained on the OntoNotes corpus and provides tags for many entities, of which our interest is in LOC, FAC, and GPE, standing for location, facilities and geopolitical entities, respectively. For Spanish and French, Stanza was trained on the CoNLL02 and WikiNER datasets, respectively.

[31] https://spacy.io/
[32] https://cloud.google.com/natural-language/docs/
[33] https://github.com/AlexPoint/OpenNlp

Tables 53, Table 54, and Table 55 offer the performance tests in terms of the processing speed according to each NER system for each of the languages supported, with the best speed numbers highlighted in bold.

Table 53. Processing speed for the English eval corpus I.

English location-detection model	Processing speed (min:sec.cs)
LORE	**00:08.69**
Stanford NER	00:09.82
NLTK	00:10.88
spaCy	00:12.15
GCNL	02:50.90
OpenNLP	03:35.10
Stanza	08:26.63

Table 54. Processing speed for the Spanish eval corpus.

Spanish location-detection model	Processing speed (min:sec.cs)
LORE	00:56.75
Stanford NER	00:06.41
NLTK	**00:06.37**
spaCy	00:31.16
GCNL	02:48.82
OpenNLP	N/A
Stanza	05:00.97

Table 55. Processing speed for the French eval corpus.

French location-detection model	Processing speed (min:sec.cs)
LORE	00:06.36
Stanford NER	N/A
NLTK	**00:05.61**
spaCy	00:11.81
GCNL	01:38.14
OpenNLP	N/A
Stanza	04:05.77

Table 56, Table 57, and Table 58 provide the performance tests in terms of the evaluation numbers achieved by each NER system for each of the languages supported. The best numbers are highlighted in bold.

Table 56. Evaluation metrics for each English location-detection model.

English location-detection model	Token-based evaluation			Entity-based evaluation		
	P	R	F1	P	R	F1
LORE	0.85	**0.83**	**0.84**	**0.81**	**0.81**	**0.81**
Stanford NER	**0.89**	0.42	0.57	0.79	0.37	0.50
NLTK	0.55	0.29	0.38	0.43	0.24	0.31
spaCy	0.75	0.33	0.46	0.66	0.28	0.39

GCNL	0.85	0.43	0.57	0.74	0.38	0.51
OpenNLP	0.73	0.27	0.40	0.56	0.21	0.30
Stanza	0.84	0.52	0.64	0.72	0.43	0.53

Table 57. Evaluation metrics for each Spanish location-detection model.

Spanish location-detection model	**Token-based evaluation**			**Entity-based evaluation**		
	P	**R**	**F1**	**P**	**R**	**F1**
LORE	0.73	**0.74**	**0.74**	0.64	**0.72**	**0.67**
Stanford NER	0.87	0.49	0.63	0.62	0.37	0.48
NLTK	0.33	0.27	0.30	0.23	0.21	0.22
spaCy	0.71	0.62	0.66	0.58	0.55	0.57
GCNL	**0.96**	0.56	0.71	**0.84**	0.53	0.65
OpenNLP	N/A	N/A	N/A	N/A	N/A	N/A
Stanza	0.88	0.64	**0.74**	0.73	0.59	0.65

Table 58. Evaluation metrics for each French location-detection model.

French location-detection model	**Token-based evaluation**			**Entity-based evaluation**		
	P	**R**	**F1**	**P**	**R**	**F1**
LORE	0.90	0.55	**0.68**	**0.81**	**0.51**	**0.62**
Stanford NER	N/A	N/A	N/A	N/A	N/A	N/A
NLTK	0.18	0.13	0.15	0.14	0.11	0.12
spaCy	0.64	0.50	0.56	0.49	0.40	0.44
GCNL	**0.95**	0.41	0.57	**0.81**	0.36	0.50
OpenNLP	N/A	N/A	N/A	N/A	N/A	N/A
Stanza	0.64	**0.59**	0.62	0.45	0.47	0.46

Figure 29, Figure 30, and Figure 31 represent, by means of a bar chart, the information from the evaluation metrics.

Figure 29. Bar chart for the evaluation metrics for each English location-detection model.

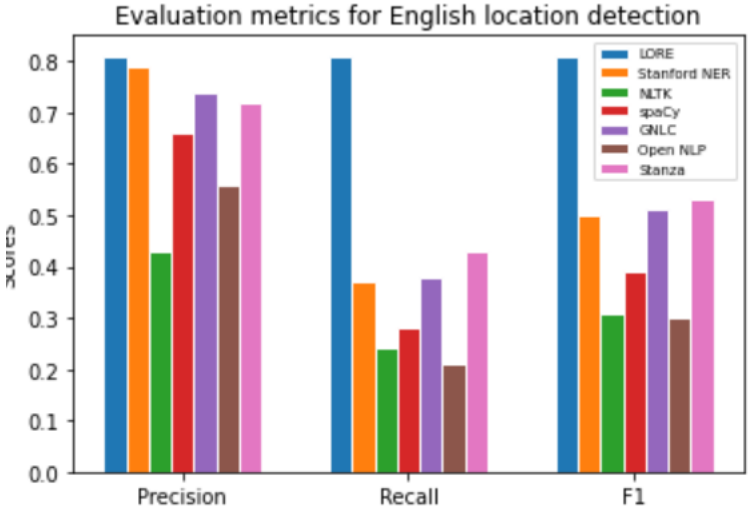


Figure 30. Bar chart for the evaluation metrics for each Spanish location-detection model.

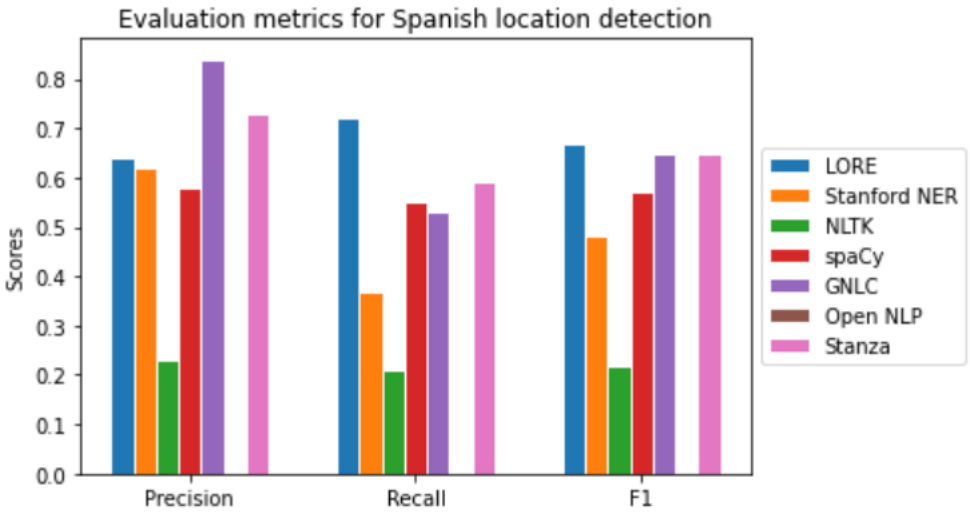
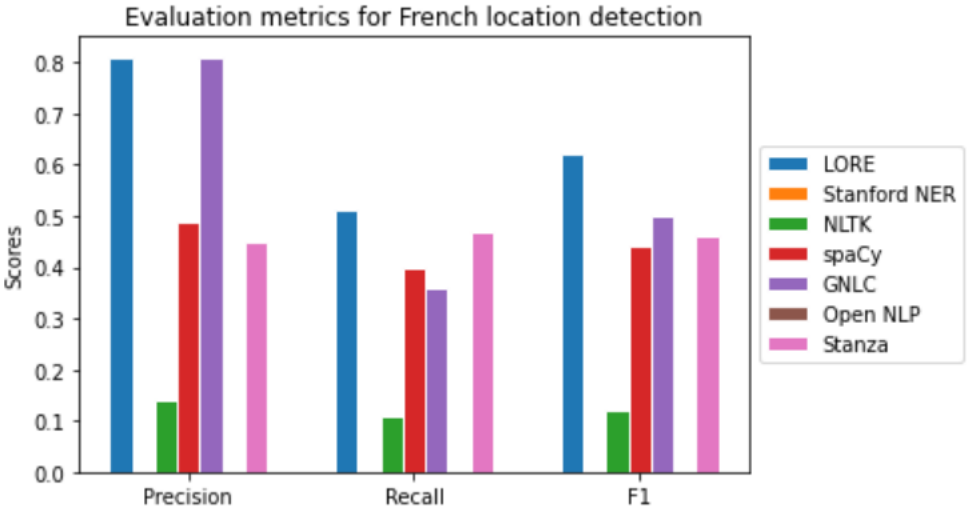


Figure 31. Bar chart for the evaluation metrics for each French location-detection model.



7.1.2. Experiment II

In the second experiment, our goal was trifold: we wanted to (a) examine which of the eight trained models for nLORE works best and whether the implementation of linguistic-based feature engineering provides superior performance, (b) check whether the performance of our first model, LORE, could remain stable and regular with another eval corpus (i.e. generalizability), and (c) compare which model, LORE or nLORE, performs best. For this, we used the newer eval dataset of English tweets composed of 1372 tweets, the English eval corpus II.

7.1.2.1. nLORE

In this experiment, our interest is in the assessment of the impact of linguistic-based feature engineering on the performance of nLORE. For that purpose, as explained in section 5.4.2, we trained eight different models with different corpus size and different linguistic-based features

(i.e. either basic or extended). Figure 32, Figure 33, and Figure 34 provide a graphical summary, by means of line graphs, of the precision, recall, and F1 scores of nLORE in its eight trained environments, testing the basic and extended models with 1000, 3000, 5000, and 7000 tweets.

Figure 32. Precision of nLORE.

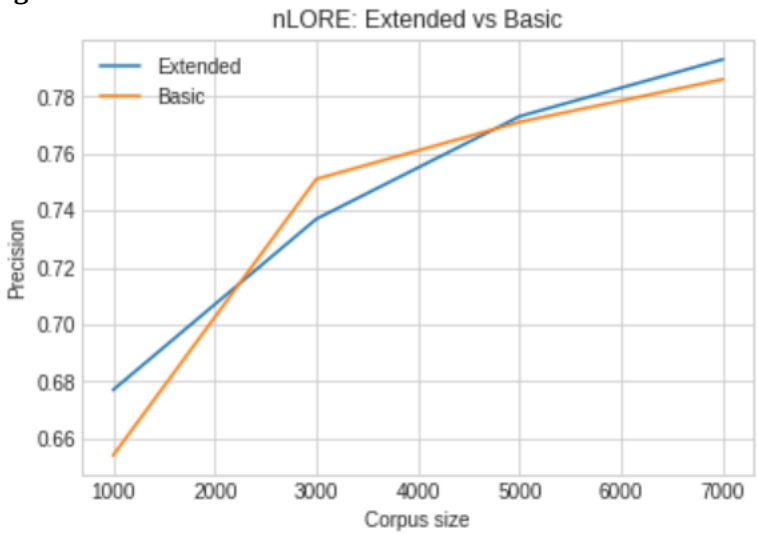


Figure 33. Recall of nLORE.

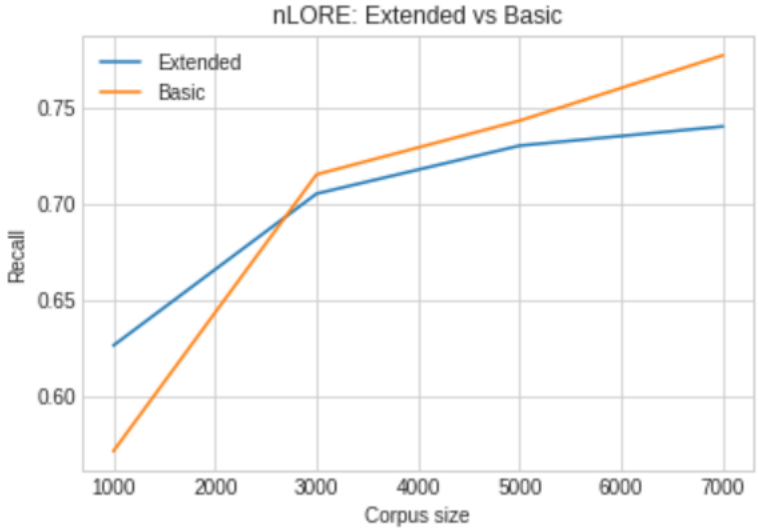


Figure 34. F1 of nLORE.

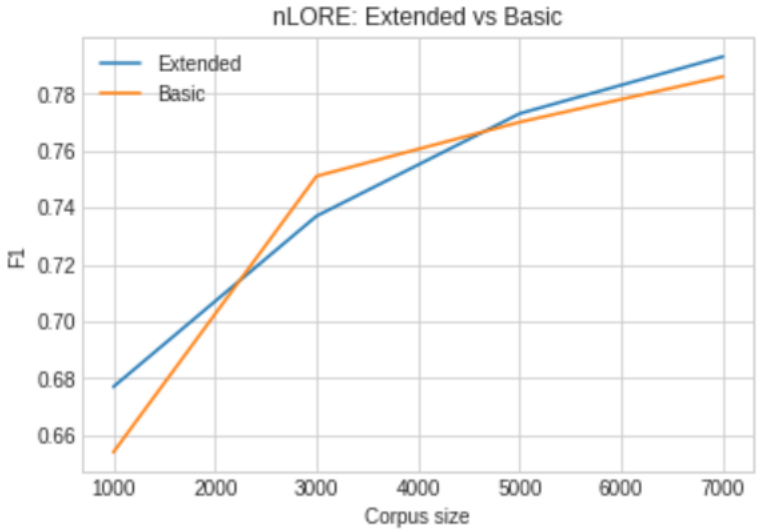


Table 59 offers the performance tests in terms of the processing speed of each of the trained models (in bold the quickest model).

Table 59. Processing speed for each of the nLORE models.

nLORE model	Processing speed (min:sec.cs)
1k with basic features	**00:53.27**
1k with extended features	01:20.17
3k with basic features	01:07.04
3k with extended features	01:39.52
5k with basic features	01:14.97
5k with extended features	01:50.45
7k with basic features	01:17.16
7k with extended features	01:58.39

Table 60 shows the evaluation numbers of the nLORE models with the English eval corpus II in terms of each evaluation method with the best results highlighted in bold.

Table 60. Evaluation of nLORE with the English eval corpus II.

Model	Token-based evaluation			Entity-based evaluation		
	P	R	F1	P	R	F1
nLORE (7k with extended features)	**0.91**	0.79	**0.85**	**0.85**	0.74	**0.79**
nLORE (7k with basic features)	0.87	**0.81**	0.84	0.79	**0.78**	**0.79**
nLORE (5k with extended features)	0.90	0.77	0.83	0.82	0.73	0.77
nLORE (5k with basic features)	0.88	0.78	0.83	0.80	0.74	0.77
nLORE (3k with extended features)	0.85	0.76	0.80	0.77	0.70	0.74
nLORE (3k with basic features)	0.87	0.76	0.81	0.79	0.71	0.75
nLORE (1k with extended features)	0.82	0.70	0.75	0.74	0.63	0.68
nLORE (1k with basic features)	0.85	0.64	0.73	0.76	0.57	0.65

The F1 scores of the basic and extended 7k models were 0.7858 and 0.7926 respectively, though in the table they appear approximated. On the basis of the precision, recall and F1 scores obtained, we picked the extended 7k model, despite the fact that it took more time than the

others to process the tweets. Table 61 shows the evaluation numbers for the best nLORE model in terms of n-gram size.

Table 61. Evaluation of the extended 7k nLORE model with the English eval corpus II in terms of n-gram size.

N-gram size	P	R	F1
Unigrams	0.88	0.74	0.80
Bigrams	0.87	0.78	0.83
Trigrams	0.78	0.60	0.68
Fourgrams	0.64	0.88	0.74
Fivegrams	0.69	0.75	0.72
Sixgrams	N/A	0	N/A
Sevengrams	N/A	N/A	N/A
Eightgrams	N/A	N/A	N/A

7.1.2.2. LORE with the English eval corpus II

We endeavored to test the performance of LORE with a second eval corpus, the English eval corpus II, the one that we used to test nLORE. Table 62 and Table 63 show the evaluation numbers of LORE with the English eval corpus II. Processing the corpus and extracting the locative references took 12.15 secs.

Table 62. Evaluation of LORE with the English eval corpus II.

Token-based evaluation			Entity-based evaluation		
P	**R**	**F1**	**P**	**R**	**F1**
0.79	0.83	0.81	0.73	0.79	0.76

Table 63. Evaluation of LORE with the English eval corpus II in terms of n-gram size.

N-gram size	P	R	F1
Unigrams	0.73	0.83	0.77
Bigrams	0.77	0.78	0.78
Trigrams	0.74	0.59	0.65
Fourgrams	0.60	0.81	0.68
Fivegrams	0.50	0.75	0.60
Sixgrams	0	0	N/A
Sevengrams	N/A	N/A	N/A
Eightgrams	N/A	N/A	N/A

7.1.2.3. LORE vs nLORE

In Table 64, we provide the evaluation numbers for the trained nLORE models against LORE using the English eval corpus II, with the best numbers highlighted in bold.

Table 64. Evaluation of LORE vs nLORE with the English eval corpus II.

Model	Token-based evaluation			Entity-based evaluation		
	P	**R**	**F1**	**P**	**R**	**F1**
LORE	0.79	**0.83**	0.81	0.73	**0.79**	0.76

nLORE (7k with extended features)	**0.91**	0.79	**0.85**	**0.85**	0.74	**0.79**
nLORE (7k with basic features)	0.87	0.81	0.84	0.79	0.78	**0.79**
nLORE (5k with extended features)	0.90	0.77	0.83	0.82	0.73	0.77
nLORE (5k with basic features)	0.88	0.78	0.83	0.80	0.74	0.77
nLORE (3k with extended features)	0.85	0.76	0.80	0.77	0.70	0.74
nLORE (3k with basic features)	0.87	0.76	0.81	0.79	0.71	0.75
nLORE (1k with extended features)	0.82	0.70	0.75	0.74	0.63	0.68
nLORE (1k with basic features)	0.85	0.64	0.73	0.76	0.57	0.65

Table 65 shows the processing speed of each model, where the fastest model appears indicated in bold.

Table 65. Processing speed for each of the LORE and nLORE models.

Model	**Processing speed (min:sec.cs)**
LORE	**00:12.15**
nLORE (1k with basic features)	00:53.27
nLORE (1k with extended features)	01:20.17
nLORE (3k with basic features)	01:07.04
nLORE (3k with extended features)	01:39.52
nLORE (5k with basic features)	01:14.97
nLORE (5k with extended features)	01:50.45
nLORE (7k with basic features)	01:17.16
nLORE (7k with extended features)	01:58.39

Figure 35, Figure 36, and Figure 37 present, by means of a line graph, a depiction of the performance of both LORE and the best nLORE model, the extended 7k model, in terms of n-grams (X axis) and precision, recall and F1 scores (Y axis), respectively.

Figure 35. Precision of LORE vs nLORE.

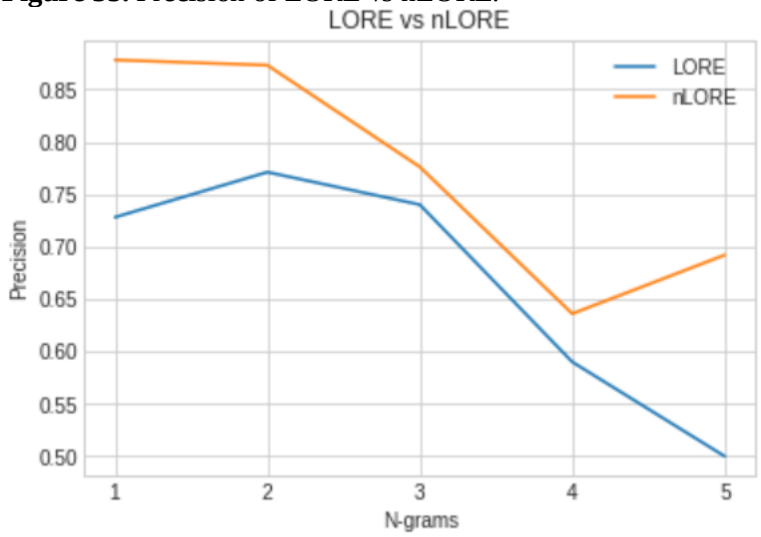


Figure 36. Recall of LORE vs nLORE.

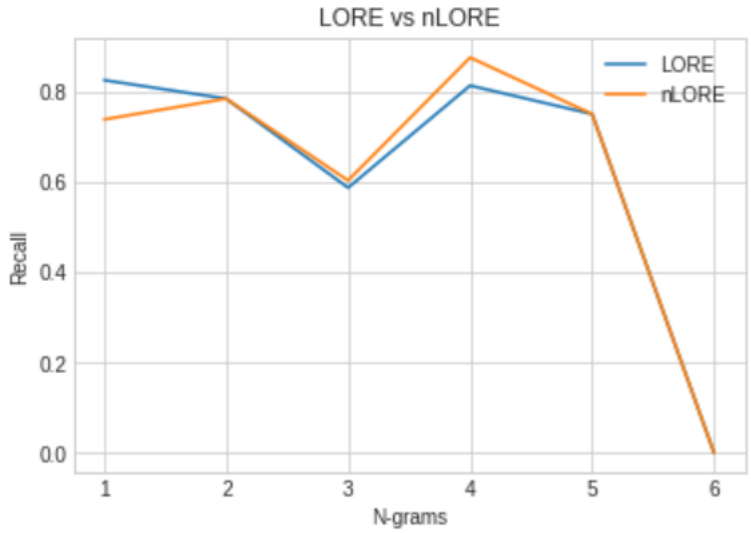
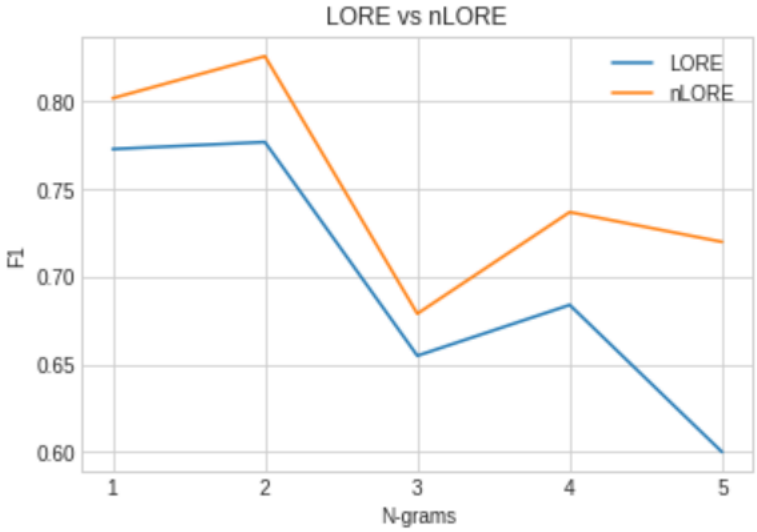


Figure 37. F1 of LORE vs nLORE.



7.2. Discussion

In Experiment I in Section 7.2.1., we discuss the performance of LORE by providing examples supporting its strengths and weaknesses, and comment on the results achieved by LORE against the competition. In Experiment II in Section 7.2.2., we explain the results obtained by nLORE by confronting the best two nLORE models, the basic and extended 7k models, providing evidence of their performance. We also comment on the performance of LORE with the English eval corpus II and the issue of generalizability of results in NLP, and then we comment on the performance of LORE against the best nLORE model providing examples of the output of both.

7.2.1. Experiment I

As can be observed in Table 47, Table 48, and Table 49 in Section 7.1.1.1., the combination of the working modules of LORE provides more balanced scores in terms of precision, recall, and F1. Their combination outperforms any of the individual modules, except for precision in the linguistic processing module, which delivers greater precision scores overall. It is true, however, that using the place-name search module on top of the linguistic processing module helped improve recall, presumably because the place-name search module found locative references that the rule-based patterns could not capture due to a lack of enough contextual clues. At the same time, using the place-name search module also slightly reduced overall precision, since this module tended to overmatch instances that were not actual locative references, despite our attempts to minimize this overmatching-prone behavior. With regards to using the place-name search module only, it is also noteworthy to highlight the enormous score differences between the token-based and entity-based evaluation methods. This can be explained by how these evaluation methods handle partial or inexact matches, which, in the case of entity-based evaluation, had a negative impact on the evaluation numbers. Put differently, the place-name search module mostly matches proper nouns as found in simple locative references and of type geopolitical entity and natural geographic references, which means that this module is generally not able to extract the full scope of complex locative references, nor can it extract traffic ways and most POIs. Only our regex-based rules and patterns can recognize most POIs and traffic ways, as illustrated by the following examples:

(73) *Cleared: Motor Vehicle Accident - HARTFORD #I84 West 0.02 miles before Exit 51 (I-91NB) at 4/11/2019 10:56:03 AM*

(74) *Extremist Pleads Guilty to Planning Mass Shooting Attack at Texas Mall*

(75) *South LA 13219 S Penrose Ave **Hit and Run No Injuries***

(76) *Vecinos de #Naucalpan se manifiestan sobre Calzada San Agustín para exigir reforzamiento de muros del Río Hondo*

'#Naucalpan neighbors demonstrate on Calzada San Agustin to demand reinforcement of Rio Hondo walls'

(77) *SanMiguel // Vehículo volcado tras accidente de tránsito sobre la carretera hacia San Jorge, en El Tránsito*

'SanMiguel // Vehicle overturned after a traffic accident on the road to San Jorge, in El Tránsito'

(78) *INCIDENTE #NoticiasPSK Precaución Vial por percance vehicular en la Autopista México - Puebla a la altura de calle José Alfredo Jiménez.*

'INCIDENT #NoticiasPSK Road safety due to car accident on the Mexico - Puebla highway at José Alfredo Jiménez street.'

(79) *Terminé*** ?? #Duplessis nord, hauteur boulevard du Versant-Nord, VD bloquée – Incident*
'Done***? #Duplessis north, at boulevard du Versant-Nord, VD blocked - Incident'

In this sense, the linguistic processing module alone achieved very high precision thanks to its fine-grained semantic coverage, but low recall, in line with the results typically obtained by rule-based NER system, because it cannot retrieve locative references if not signaled by locative prepositions, location-indicative nouns, or locative markers. This lower false-positive rate of rule-based approaches could be particularly useful for emergency situations that place greater importance on precision over recall (Middleton et al., 2014). Geopolitical entities embedded in hashtags lacked sufficient linguistic or contextual clues for the linguistic processing module to recognize them, as seen in Example (80) and Example (81),

(80) *#Incident #Ottawa #HWY417 WB at Metcalfe St (IC 119A)*
(81) *CLEAR - #BCHwy10 EB vehicle incident at #BCHwy91 overpass. #DeltaBC*

unless when followed, for instance, by locative prepositions (Example (82)).

(82) *Accident with injury in #EastBatonRouge on Airline SB at I 12 #traffic*

Likewise, geopolitical entities that appeared as NPs in the subject position (Example (83)) or object position (Example (84)) could not be inferred with regex-based rules, since there was not a distinguishing linguistic clue in that position that made place names different from person names or other proper nouns.

(83) *New Zealand's new gun laws going into effect less than a month after mosque shootings*
(84) *#Estados | Lluvias azotan #Monterrey Nuevo León, donde este fin de semana se registraron severas inundaciones.*
'#States | Rains hit #Monterrey Nuevo Leon, where severe flooding was recorded this weekend.'

Thus, in the absence of any contextual clue, only the place-name search module can detect them, as observed in the following examples:

(85) *#Nacional | ¡Fuertes lluvias e inundaciones cobran vidas humanas! - - #Inundaciones #Monterrey*

'#National | Heavy rains and floods take lives! #Floods #Monterrey'

(86) *Cortadas casi todas las líneas del metro por inundaciones. En fin, las cosas de Madrid.*

'Almost all the subway lines were cut off due to flooding. In short, the things of Madrid.'

(87) *#Montpellier : coups de feu à la #Kalachnikov, la police judiciaire est saisie.*

'#Montpellier: shots fired using a Kalashnikov, the judicial police is seized.'

(88) *Conséquences de l'accident de personne : #TER839122 Châlons-en-Champagne 10:17*

'Consequences of the person's accident: #TER839122 Châlons-en- Champagne 10:17'

In short, to compensate for the pitfalls presented in each of these modules, the integration of both modules provided the best of both worlds without compromising precision or recall.

The performance benchmarks of the models in Section 7.1.1.2. show that the processing speed is fast with similar timing among all models except for GCNL, OpenNLP and Stanza, which are the slowest of all (Table 53, Table 54, and Table 55). However, LORE was almost 10x times slower when applied to the Spanish eval corpus (Table 54), the reason being that the POS tagger took most of the time to tag the tokens. It would probably take fewer than 10 seconds if another POS tagger were used. The enormous delay by GCNL with all the eval corpora can probably be explained by the time-consuming nature of the API requests to the Web Service. Only our model excelled at processing speed by only a few seconds in the English eval corpus I, the Stanford NER tool closely following on second position (Table 53). In the case of the French eval corpus, our model came very close to the first position, only closely surpassed by NLTK (Table 55). With regard to the evaluation numbers, all NER models, except LORE, achieved low or very low recall (Table 56, Table 57, Table 58, Figure 29, Figure 30, and Figure 31), except the Stanza tool in the token-based evaluation of the French eval corpus (Table 58, Figure 31), where it narrowly outperformed LORE. A reason for these low recall scores can be attributed to the lack of sufficient granularity in their NER systems, which could not address the full semantics of locative references. In other words, although they performed well in the identification of location types such as geopolitical entities (mainly towns, cities, and countries) and a few POIs and addresses, many suffered to recognize natural landforms, most POIs and addresses, or traffic ways, let alone the locative markers or location-indicative nouns that accompanied proper nouns. Despite that, their precision was fairly high, especially in the case of Stanford NER, GCNL, and Stanza. These evaluation numbers support what recent studies have

found regarding the performance of Stanford NER, which is known to perform best in different NER-based evaluation tasks and scenarios (Schmitt et al., 2019). All in all, given that their recall was low, their F1 score was not very high. In this regard, we argue that their coarse-grained semantics was the main reason behind their low evaluation numbers. LORE achieved the best evaluation results in the evaluation metrics and had excellent processing speed on most occasions, outperforming the state-of-the-art NER systems used in the performance tests.

7.2.1.1. Error analysis

At this stage, we analyze the commonest sources of errors committed by LORE (i.e. place-name search + linguistic processing) and offer an explanation of their occurrence, as well as provide some possible solutions and alternatives that we leave for future research. We distinguish between errors of omission and errors of commission. By errors of omission, we mean those instances of locative references that were missed by the model (i.e. FNs), while errors of commission took place when instances that were not actual locative references were wrongly extracted (i.e. FPs).

7.2.1.1.1. Errors of omission

The population-size filtering of the place-name dataset, despite helping increase precision, affected recall because of some cases of FNs, particularly of geopolitical entities. These FNs, at times, could not be mitigated by the regex-based rules in the linguistic processing module either. In Example (89), *Indinapuram* is a locative reference of geopolitical-entity type which were missed exactly for the reasons mentioned above.

(89) *@MORTHRoadSafety Pls consider asking the #NHAI to close the central verge on #NH24 between #Indirapuram and […]*

Obviously, retrieving the full GeoNames database could have avoided many FNs and therefore increased recall, but at the expense of many more cases of FPs, dramatically affecting precision, and a slower performance. Considering the benefit-cost ratio, we opted for the population size restriction.

At other times, the GeoNames database, for languages other than English, lacked a sufficiently large coverage of locative references, as evidenced in the omission of a few locative references that were not affected by the population filtering step:

(90) *Que barbaridad!!! ????? #ArgandaDelRey #Madrid*
'Outrageous!!! ????? #ArgandaDelRey #Madrid'

(91) *Como se ve relampagear y tronar desde Lizarra-Estella por la zona de la Sierra de Urbasa y Lokiz*

'What a sight of lightning and thunder from Lizarra-Estella in the area of the Urbasa and Lokiz mountains'

(92) *Gambetta (incident technique) Le départ d'Epron prévu à 12h19 ne sera pas effectué*

'Gambetta (technical incident) The departure from Epron scheduled at 12.19pm will not be carried out.'

(93) *Orage au péage de #Villefranche #Limas sur #A6 ?@VoyageAPRR?*

'Storm at the Villefranche tollbooth, Limas on A6 ?@VoyageAPRR?'

This could have possibly been avoided if we had enriched the place-name dataset with other geographic databases such as OpenStreetMaps. However, that could have had a negative impact on the processing speed of the model, and on the precision of the model, since the number of FPs would have augmented.

Another source of omission errors has to do with bad spelling or lack of proper capitalization, as observed in the following examples:

(94) […] *a video in 2016 at the east west highway*

(95) *00:36 Magpie Swamp Rd/mingbool Rd, Pleasant Park - Tree Down going (one appliance, CFS region 5)*

(96) *Y por si fuera poco con las tormentas e inundaciones, tornado en campillos @AEMET_SINOBAS @AEMET_Esp.*

'And as if that wasn't enough with the storms and floods, tornadoes in campillos @AEMET_SINOBAS @AEMET_Esp.'

(97) *en eeuu se sabe que tu hijo ha madurado despues de su primer tiroteo, q rapido crecen??????*

'in the u.s., your son is known to have matured after his first shooting, how fast they grow?????'

The Stanford POS tagger mislabeled the underlined instances as common nouns instead of proper nouns because of lack of proper capitalization in the initial letter. Since one key linguistic clue in all modules is that locative references must contain proper nouns, and the modules highly depend on the performance of the third-party POS tagger, the model could not avoid these cases of FNs. This affected the performance of both modules, the place-name search and the linguistic processing. To mitigate this type of errors, we

implemented a third-party library for text normalization[34] which could a *priori* help in the performance of the POS tagger. Although it helped in this regard, it however delayed the performance of the model up to 3x times slower. Considering the cost-benefit ratio, we preferred not to perform text normalization. Perhaps by using a Twitter-specific POS tagger the model could reduce the number of FNs without compromising the quick processing speed, but this remains an issue for future research. However, not always did the POS tagger fail to recognize proper nouns when they lacked proper capitalization (Example (98)).

(98) *Jajajajksks fui al oftalmólogo y me dijo ""sos algo del zavalia radical de santiago del estero*
'hahhhahah I went to the ophthalmologist and he told me ""you are something from the radical zavalia of santiago del estero'

Another important source of errors of omission relates to the difficulties in handling abbreviations and acronyms (Example (99) and Example (100)), which often go unnoticed in location-detection systems. In these cases, if these were neither contained in the place-name dataset nor recognized by regex-based rules, they would simply be missed.

(99) *Just passed a terrible car accident on jfk ?? seeing that got my stomach hurting right now*

(100) *We had an earthquake in the IE?*

We are also aware of the existence of complex locative expressions other than those containing locative markers. By this we mean, for instance, coordinated place names (e.g. *in the US and the UK*, *between Madrid and Barcelona*, etc.) or other more complex locative formulas (e.g. *close to London but not far away from Croydon*). A few instances of these were found in the Spanish and French eval corpora, as shown in Example (101) and Example (102).

(101) *Preocupación por las inundaciones en las zonas este y sur de Madrid, tras la tormenta*
'Concern over flooding in eastern and southern Madrid following the storm'

(102) *ADSL / SDSL:: Incident départements 31 et 33.: Nous rencontrons actuellement une dégration de service*
'ADSL / SDSL:: Incident in departments 31 and 33: We are currently experiencing a service degradation'

[34] SymSpell library: https://github.com/wolfgarbe/SymSpell

These are quite challenging and offer problems to current models, since the linguistic patterns that underlie them are unpredictable and obscure and remain elusive to formalize in regex-based rules. Perhaps using a syntactic parser could help delimit locative references within their phrasal boundaries. In preliminary versions of our model, we used a syntactic chunker for this, but it dramatically slowed down performance and did not offer much improvement. For now, locative references found in such complex locative expressions can usually be identified individually, and our model has been able to detect some complex locative patterns, especially those involving locative markers.

In the languages supported other than English, there are quite a few complex locative patterns involving the use and combination of location-indicative nouns and determiners, prepositions and/or punctuation marks plus proper nouns that could only be partially captured by our current rules, or not captured at all. Examples of these are as follows:

(103) *A mi me tomó 8 meses para que me instalarán aba aquí en San Antonio del tachira*

'It took me 8 months to get aba installed here in San Antonio del tachira'

(104) *El lamentable accidente ocurrió en la carretera libre Aguascalientes-Encarnación de Díaz*

'The unfortunate accident occurred on the free highway Aguascalientes-Encarnación de Díaz'

(105) *@AlertesRER ??? - INCIDENT AFFECTANT LA VOIE (2) Ligne E : Paris Chelles ralenti ? Entre Chelles Gournay et Paris.*

'@AlertesRER ??? - INCIDENT AFFECTING THE ROAD (2) Line E: Paris Chelles slowed down ? Between Chelles Gournay and Paris.'

(106) *Plusieurs riverains ont composé le « 17 » ce dimanche soir à Montpellier, dans le quartier de la Croix d'Argent*

'Several local residents composed the "17" this Sunday evening in Montpellier, in the Croix d'Argent district.'

7.2.1.1.2. Errors of commission

The overmatching-prone behavior of the place-name dataset generates some FPs, as in Example (107) where the animal name *Nemo* was wrongly identified as a locative reference, or as in Example (108) where the person name *Chaparro* was not filtered by the stopword dataset, or also in Example (109) where another person name, *Julien*, was not filtered by the stopword dataset either.

(107) *Happy #NationalPetDay and I really miss my ~~Nemo~~ (cat) passed away in car accident.*

(108) *Se puede ser más falsa y sobreactuada que Carme ~~Chaparro~~ ?? Sobran sus caras absurdas , sus aspavientos*

'Can you be more false and over-acting than Carme Chaparro? No more absurd faces, no more fuss'

(109) *@Julien_Db Bonjour ~~Julien~~. Navré de cet incident. Je vous invite à échanger en DM pour que nous puissions régler le problème.*

'@Julien_Db Hello Julien. Sorry about this incident. I invite you to exchange DM so that we can solve the problem.'

All of them exist as locative references in the place-name dataset. Contextual evidence suggests, however, that they do not refer to actual locative references. Indeed, they were correctly identified as proper nouns by the POS tagger, but then bypassed the safe-checking rules because they were not captured by the stopword dataset.

Despite our conscious efforts to mitigate FPs by leveraging safe-checking rules, we blame some cases of FPs on the performance of the Stanford POS tagger, which sometimes considered common nouns and other parts of speech to be proper nouns because of wrong capitalization patterns, as shown in Example (110), Example (111), and Example (112).

(110) *y'ALL THIS AU IS SPOT ON!!! IT'S SO BEAUTIFULLY ~~MADE~~ I CRIED OMGGGGGGGGGGGG*

(111) *@MckarloFernan Honestly go awf fam ???? if the put cheese in by accident next time call me and I'll eat it for you ~~LOL~~*

(112) *Por favor señores todo o lo que pasa ahora es ~~el Presidente~~ y usted si en realidad quisiera a su vástago…*

'Please gentlemen, everything that happens now is due to the President and you, if you really want your offspring…'

MADE, *LOL*, and *el Presidente* were wrongly retrieved as locative references. Surprisingly, *Made* is a Dutch and also an Indonesian village[35], *Lol* refers to a South Sudan state[36], and *El Presidente* is a mountain in Mexico[37]. These erroneous instances of locative references can be explained by the confluence of different factors: their POS tag was wrongly assigned as proper noun, they were captured in the place-name dataset, and finally they bypassed the safe-checking

[35] https://www.geonames.org/2751272/made.html and https://www.geonames.org/6407244/made.html
[36] https://www.geonames.org/11550548/lol.html
[37] https://www.geonames.org/3521135/el-presidente.html

rules involving the use of the stopword dataset. In this respect, the place-name search module does not have a corrective capacity, but a preventive one only, which occasionally might fail as in this case.

On the other hand, the location-indicative noun matching submodule also led to the extraction of wrong locative references, as shown in the following examples:

(113) *@BrianBLevinson I like how they list ~~Lief Green~~ next to Jim Greenleaf. No way that was by accident.*

(114) *@manishinsha93: #Road~~SafetyInitiativeByDSS Saint Dr~~. MSG has come up with the initiative to tie reflector belts on the stray animals*

(115) *~~CLEARED~~ HUDSON VALLEY: Slow traffic* […]

(116) *Mauranne et Laura, deux jeunes étudiantes cousines, se feront égorgées par un terroriste sur le parvis de la Gare S. ~~url~~*
'Mauranne and Laura, two young cousin students, will have their throats cut by a terrorist on the square in front of the S. Station url'

(117) *Cortadas por inundación tras la tormenta la M-506, la M-40 y al menos 6 ~~líneas de Metro~~*
'M-506, M-40 and at least 6 subway lines cut by flooding after the storm'

In Example (113), *green*, a location-indicative noun used to denote an area of land covered with grass, mismatched G*reen* in the tweet and, since the model found that the previous word was a proper noun, *Lief Green* was mined as a locative reference. In Example (114), *dr* mismatched the abbreviation of the location-indicative noun *drive*, which coincides with the abbreviation for the word *doctor*, and took the preceding proper nouns as part of a locative reference. In Example (115), the location boundary was wrongly delimited because *CLEARED* was tagged as a proper noun, again a case of a malfunctioning POS tagger. In Example (116), the locative reference *Gare S. url* was partially matched, since *url* does not take part in it, but was nevertheless taken into account for being mislabeled as proper noun by the Stanford POS tagger. In Example (117), *líneas de Metro* was mistaken as a locative reference because *línea* is a location-indicative noun followed by a preposition and a proper noun, which is one of the formalized regex-based rules in the Spanish and French location-indicative noun matching submodule.

Finally, another source of error lies in the regex-based rules for capturing address numbers in the English location-indicative noun matching submodule, as shown in Example (118).

(118) *The ~~1st church~~ burned, everyone thought it could have been an accident. After the ~~2nd church~~ burned, deacons.*

7.2.2. Experiment II

Since in Experiment I we already focused on the capabilities and performance offered by LORE, in Experiment II we compare the basic and extended nLORE models in Section 7.2.2.1., then we check the generalizability of LORE with the new eval corpus in Section 7.2.2.2., and last we benchmark the best nLORE model against LORE in Section 7.2.2.3.

7.2.2.1. nLORE

nLORE was trained departing from a series of basic or core linguistic-based features, the token form and POS tag, extended with those that were applied in its rule-based counterpart, the presence in the place-name dataset, the presence in the location-indicative noun dataset, and being part of locative marker or not, and using different train datasets with varying corpus size with the aim of testing the impact of the different combinations of features and corpora. The addition of extra linguistic-based features, other than token and POS tag, in the process known as linguistic-based feature engineering, was carried out with the objective of endowing nLORE with the capacity to learn and infer linguistic patterns in the process of locative reference extraction.

In this sense, having examined Figure 32, Figure 33, Figure 34, and Table 60 in Section 7.1.2.1., we can appreciate that the precision scores, which were higher in the basic 1k and 3k models, grow exponentially larger in favor of the extended model when the train dataset is larger, outnumbering the precision scores of the basic 5k and 7k models, as observed in the extended 5k and 7k models. In the case of recall scores, the opposite is true: while recall scores were better for the extended 1k model, they are eventually outnumbered by the basic 3k, 5k and 7k models, with the divide exponentially growing larger and larger. Thus, the contribution of the extended linguistic-based features seems to be, at best, poorly significant and, at worst, only incidental. Though a trend can be noted, especially with the smallest corpus, results remain certainly inconclusive. If we were to sketch the main reasons in support of the use of the extended linguistic-based features, these would be the slightly better performance, especially when corpus size is fairly limited. However, things tend to level out as soon as corpus size becomes increasingly larger, and even in this scenario providing extra linguistic features might seem a bit counterproductive, since they worsen performance very slightly, as can be concluded from the recall scores, while also affecting the processing speed, despite improving precision scores.

In this sense, our intuition is that the extended features may help avoiding the extraction of wrong instances that cannot be properly discarded by the token and POS tag features alone, hence the higher precision scores in the extended 5k and 7k models. However, adding such

features might have a negative impact –though very slight– on the identification of right locative references, as spotted in the recall scores. Having those extra linguistic-based features may have incidentally made the process of locative reference extraction stricter, which explains the fewer number of FPs and the greater number of FNs.

The following examples from the English eval corpus II serve to illustrate the powerful capacity for prediction of the basic and extended 7k models when identifying locative references. Given that corpus data format follows a token-based tabular representation and that the assignation of the correct label was important for the evaluation stage, we provide such format for the examples in the tables below. We picked the two best nLORE models, i.e. the basic and extended 7k models. In red, we highlight the source of errors, if any.

As previously stated, both models had state-of-the-art performance, despite being confronted with many different location types in many different emergency-related tweets (Table 66 and Table 67).

Table 66. Example of the basic 7k nLORE model.

Token	POS tag	Label
Incident	NN	O
on	IN	O
I684	NN	B_LOCATION
NB	NN	E_LOCATION
at	IN	O
Exit	NN	B_LOCATION
6A	NN	E_LOCATION
-	:	O
NY	NNP	B_LOCATION
22	CD	E_LOCATION
to	TO	O
NY	NNP	S_LOCATION
138	CD	E_LOCATION
-	:	O
Goldens	NNP	B_LOCATION
Bridge	NNP	E_LOCATION
(	(	O
Northbound	NNP	B_LOCATION
Exit	NN	M_LOCATION
Ramp	NN	E_LOCATION
)	)	O

Table 67. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
Incident	NN	0	0	0	O
on	IN	0	0	0	O
I684	NN	0	0	0	B_LOCATION
NB	NN	0	0	0	E_LOCATION
at	IN	0	0	0	O
Exit	NN	0	1	0	B_LOCATION

6A	NN	0	0	0	E_LOCATION
-	:	0	0	0	O
NY	NNP	1	0	0	B_LOCATION
22	CD	0	0	0	E_LOCATION
to	TO	0	0	0	O
NY	NNP	1	0	0	B_LOCATION
138	CD	0	0	0	E_LOCATION
-	:	0	0	0	O
Goldens	NNP	1	0	0	B_LOCATION
Bridge	NNP	1	1	0	E_LOCATION
(	(	0	0	0	O
Northbound	NNP	0	0	0	B_LOCATION
Exit	NN	0	1	0	M_LOCATION
Ramp	NN	0	1	0	E_LOCATION
)	)	0	0	0	O

Despite this great performance, there were instances of partially or fully missed locative references. For instance, let us have a look at how the basic nLORE model could extract the whole locative reference (Table 68) whereas the extended model could only capture it partially (Table 69), having a negative impact on recall.

Table 68. Example of the basic 7k nLORE model.

Token	POS tag	Label
FAAN	NNP	O
shuts	VBZ	O
down	RP	O
Port	NNP	B_LOCATION
Harcourt	NNP	M_LOCATION
airport	NN	E_LOCATION
temporarily	RB	O
over	IN	O
bush	JJ	O
fire	NN	O
incident	NN	O

Table 69. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
FAAN	NNP	0	0	0	O
shuts	VBZ	0	0	0	O
down	RP	1	0	0	O
Port	NNP	1	1	0	B_LOCATION
Harcourt	NNP	1	0	0	E_LOCATION
airport	NN	0	1	0	O
temporarily	RB	0	0	0	O
over	IN	1	0	0	O
bush	JJ	1	0	0	O
fire	NN	0	0	0	O
incident	NN	0	0	0	O

Even though the location-indicative noun dataset feature was supposed to help in the identification of location-indicative nouns that take part in locative references, in Table 69 it simply failed to leverage such clue. This was not the case in many other instances, where this feature, together with the place-name dataset feature, may have helped to a great extent, as shown in Table 71, as opposed to the basic model which could only capture part of the locative reference (Table 70).

Table 70. Example of the basic 7k nLORE model.

Token	POS tag	Label
and	CC	O
your	PRP$	O
Bronx	NNP	B_LOCATION
River	NNP	E_LOCATION
Neighborhood	NNP	O
Coordination	NN	O
Officers	NNS	O

Table 71. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
and	CC	0	0	0	O
your	PRP$	0	0	0	O
Bronx	NNP	1	0	0	B_LOCATION
River	NNP	1	1	0	M_LOCATION
Neighborhood	NNP	1	0	0	E_LOCATION
Coordination	NN	0	0	0	O
Officers	NNS	0	0	0	O

In Table 72, we can observe how the basic nLORE model failed to capture the two locative references and instead extracted a whole one, assigning wrong labels, whereas the extended Nlore model could effectively extract the two locative references (Table 73), probably thanks to the linguistic-based features. All this had an impact on precision.

Table 72. Example of the basic 7k nLORE model.

Token	POS tag	Label
Two	CD	O
vehicle	NN	O
incident	NN	O
,	,	O
48	CD	B_LOCATION
St	NNP	E_LOCATION
and	CC	M_LOCATION
32	CD	M_LOCATION
Ave	NN	M_LOCATION
NE	NNS	E_LOCATION

Table 73. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
Two	CD	0	0	0	O
vehicle	NN	0	0	0	O
incident	NN	0	0	0	O
,	,	0	0	0	O
48	CD	0	0	0	B_LOCATION
St	NNP	0	1	0	E_LOCATION
and	CC	0	0	0	O
32	CD	0	0	0	B_LOCATION
Ave	NN	1	1	0	M_LOCATION
NE	NNS	1	0	1	E_LOCATION

In Table 75, we believe that the linguistic-based features, especially the locative-marker feature, helped correctly delimit the scope of the locative reference in the extended nLORE model, as opposed to the basic nLORE model that missed the locative marker phrase (Table 74).

Table 74. Example of the basic 7k nLORE model.

Token	POS tag	Label
earthquake	NN	O
995	CD	O
km	NN	O
north-east	NN	M_LOCATION
of	IN	M_LOCATION
Whangarei	NNP	E_LOCATION

Table 75. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
earthquake	NN	0	0	0	O
995	CD	0	0	1	B_LOCATION
km	NN	0	0	1	M_LOCATION
north-east	NN	1	0	1	M_LOCATION
of	IN	1	0	1	M_LOCATION
Whangarei	NNP	1	0	0	E_LOCATION

Sometimes, both models failed to capture locative references properly, such as the locative items *Tel Aviv Elevator* (Table 76 and Table 77) or *American Iraq airbase* (Table 78 and Table 79), by assigning wrong labels and not capturing the full locative reference, affecting recall.

Table 76. Example of the basic 7k nLORE model.

Token	POS tag	Label
Killing	NNP	O
2	CD	O
in	IN	O
Tel	NNP	S_LOCATION
Aviv	NNP	S_LOCATION
Elevator	NNP	O

Table 77. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
Killing	NNP	0	0	0	O
2	CD	0	0	0	O
in	IN	0	0	0	O
Tel	NNP	1	0	0	S_LOCATION
Aviv	NNP	1	0	0	E_LOCATION
Elevator	NNP	0	0	0	O

Table 78. Example of the basic 7k nLORE model.

Token	POS tag	Label
Anti-War	JJ	O
Protests	NNS	O
Flood	NNP	O
U.S.	NNP	S_LOCATION
Streets	NNPS	O
As	IN	O
American	NNP	O
Iraq	NNP	S_LOCATION
Airbase	NNP	O
Attacked	VBD	O

Table 79. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
Anti-War	JJ	0	0	0	O
Protests	NNS	0	0	0	O
Flood	NNP	1	0	0	O
U.S.	NNP	0	0	0	S_LOCATION
Streets	NNPS	0	1	0	O
As	IN	1	0	0	O
American	NNP	0	0	0	O
Iraq	NNP	1	0	0	S_LOCATION
Airbase	NNP	0	0	0	O
Attacked	VBD	0	0	0	O

At other times, both models were unable to identify locative references, especially a few roads (Table 80 and Table 81), impacting recall.

Table 80. Example of the basic 7k nLORE model.

Token	POS tag	Label
Road	NNP	O
closed	VBD	O
and	CC	O
queueing	VBG	O
traffic	NN	O
due	JJ	O
to	TO	O

accident	NN	O
on	IN	O
M32	NN	O

Table 81. Example of the extended 7k nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
Road	NNP	0	1	0	O
closed	VBD	0	0	0	O
and	CC	0	0	0	O
queueing	VBG	0	0	0	O
traffic	NN	1	0	0	O
due	JJ	1	0	0	O
to	TO	0	0	0	O
accident	NN	1	0	0	O
on	IN	0	0	0	O
M32	NN	0	0	0	O

Overall, having analyzed the extracted locative references, although the models struggled with a few traffic ways and POIs, it seems that both models were quite successful in extracting many location types such as geopolitical entities or natural landforms.

7.2.2.2. LORE with the English eval corpus II

One of the many critical issues in NLP is that of the generalizability of results. By this, NLP practitioners refer to the more than desirable outcome of any computational model of being able to derive the same performance to new, unseen collections of data. Generalizability can thus be tested by using more than one eval corpus and comparing performance. If performance remains the same or similar, we could conclude that the model is indeed successful in the task for which it was trained or developed. Having this in mind, we can conclude, on the basis of the evaluation results which show a more or less regular and homogeneous performance (Table 62 and Table 63 in Section 7.1.2.2.), that the methodological foundations of LORE have been successfully laid out. Though performance has degraded only by a very little margin, it can be revealing to sketch out the main reasons behind this. Since the performance of LORE has already been thoroughly analyzed in Experiment I of Section 7.2.1., we do not intend to offer a detailed explanation here, but a few examples. For instance, let us consider Example (119).

(119) *Cleared: Incident on #FranklinDRooseveltDrive NB at 58th Street*

In this tweet, only a part of the locative reference contained within the hashtag was retrieved. The reason why it was not fully extracted was due to the fact that (a) the hashtag was wrongly segmented into *Frank Lind Roosevelt Drive* and (b) *Drive* was tagged as a verb. *Lind* is an actual place found in the place-name dataset, hence its extraction. Besides errors of omission

like these, errors of commission were also frequent for the same reasons that were sketched in Experiment I, as shown in Example (120) and Example (121).

(120) ~~*ALERT*~~*: Two vehicle incident, 48 St and 32 Ave NE, blocking the right lane. #yyctraffic #yycroads*

(121) *Many Feared Dead As Gas Explosion* ~~*Rocks*~~ *Kaduna*

In Example (120), *48 St* and *32 Ave NE* were not extracted because of a regex-based rule that avoids the extraction of any location-indicative noun preceded by an Arabic numeral, since they commonly denote quantity (e.g. *5 churches, 2 restaurants*). It is not the case, however, with ordinal numbers (e.g. *58th street*), as seen in Example (119). *Yyc Traffic Yyc Roads* was extracted as a locative item when the hashtag was split into tokens because *Roads* was taken as a location-indicative noun preceded by proper nouns. Another source of errors, as previously analyzed, had to do with the functioning of the POS tagger because it mislabeled some parts of speech as proper nouns which coincidentally matched with the items contained in the place-name dataset, as was the case with *Rocks* in Example (121).

Overall, it thus becomes evident that our rule-based model brings with itself the same erratic behavior shown with the English eval corpus I, while also retaining its powerful capabilities when confronted with other corpora. The slight drop in performance does not indicate anything beyond the more than expected irregularity and randomness that comes with new, unseen collections of data. In other words, LORE may perform the same, slightly better or worse with new corpora for no apparent reason other than the unpredictability associated with new data. Unlike an intelligent DL-based system which might be able to infer from the linguistic context when a given token is or is not a location-indicative noun or a locative reference, LORE is not able to make such distinction because of its pattern-based matching behavior. All in all, it is both the more or less stable performance and fast processing speed that make LORE a very effective and powerful rule-based model for location extraction, despite its known limitations.

7.2.2.3. LORE vs nLORE

The extended nLORE models feed off the linguistic knowledge provided by LORE. In other words, they leveraged the labels provided by LORE in the train corpus, later manually revised, to target the same semantic location types. At the same time, with the addition of the extended linguistic-based features and word embeddings as dense features we hoped to endow nLORE with symbolic-based knowledge on the basis of which it could more intelligently infer and predict locative patterns in the tweets. Our aim was, in a sense, to be able to improve the extraction of locative references by alleviating some of the commonest source of errors caused by, for instance, mislabeled POS tags and mismatched locative items from the place-name

dataset. This was confirmed by the results shown in Table 64, Figure 35, Figure 36, and Figure 37 in Section 7.1.2.3. In this section, we provide examples of LORE against the best nLORE model, the 7k extended model, following the token-based tabular format, where errors are highlighted in red. Such examples provide evidence that supports the success of our aims. Let us consider Table 82, Table 83, Table 84, and Table 85.

Table 82. Example of LORE.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
I	PRP	0	0	0	O
've	VBP	0	0	0	O
read	VBN	0	0	0	O
all	DT	0	0	0	O
of	IN	1	0	0	O
Clancy	NNP	1	0	0	S_LOCATION
novels	NNS	0	0	0	O

Table 83. Example of the 7k extended nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
I	PRP	0	0	0	O
've	VBP	0	0	0	O
read	VBN	0	0	0	O
all	DT	0	0	0	O
of	IN	1	0	0	O
Clancy	NNP	1	0	0	O
novels	NNS	0	0	0	O

Table 84. Example of LORE.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
WTH	NNP	0	0	0	O
DID	NNP	1	0	0	S_LOCATION
YOU	PRP	0	0	0	O
EXPECT	VBP	0	0	0	O

Table 85. Example of the 7k extended nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
WTH	NNP	0	0	0	O
DID	NNP	1	0	0	O
YOU	PRP	0	0	0	O
EXPECT	VBP	0	0	0	O

nLORE was, indeed, capable of avoiding these errors of commission since it was able to learn, by the linguistic context, that these instances were not actual locative references, but for different reasons. In Table 82 and Table 83, *Clancy*, a proper noun, coincides with a locative

item in the place-name dataset. LORE (Table 82), based on rules, automatically extracts it, without taking into account the linguistic context: *Clancy* is the author of a novel, not a location. nLORE was somehow able to disambiguate the context and did not extract it (Table 83). In Table 85, we can observe how a mislabeled token, *DID*, which is a verb and not a proper noun, was appropriately discarded by nLORE, because it realized that *DID* was actually a verb, not a noun, and definitely not a location (Table 84). The reason why nLORE could infer probably lies in the contextual window of tokens to the left and to the right and the semantic information provided by the word embeddings.

At other times, LORE could extract locative items thanks to the patterns formalized by the regex-based rules (Table 86 and Table 88), helping recall, which nLORE could not (Table 89) or only partially (Table 87).

Table 86. Example of LORE.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
FAAN	NNP	0	0	0	O
shuts	VBZ	0	0	0	O
down	RP	1	0	0	O
Port	NNP	1	1	0	B_LOCATION
Harcourt	NNP	1	0	0	M_LOCATION
airport	NN	0	1	0	E_LOCATION
temporarily	RB	0	0	0	O

Table 87. Example of the 7k extended nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
FAAN	NNP	0	0	0	O
shuts	VBZ	0	0	0	O
down	RP	1	0	0	O
Port	NNP	1	1	0	B_LOCATION
Harcourt	NNP	1	0	0	E_LOCATION
airport	NN	0	1	0	O
temporarily	RB	0	0	0	O

Table 88. Example of LORE.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label
Police	NN	1	0	0	O
activity	NN	0	0	0	O
on	IN	0	0	0	O
I-39NB	NN	0	0	0	S_LOCATION

Table 89. Example of the 7k extended nLORE model.

Token	POS tag	Place-name dataset	Location-indicative noun dataset	Locative marker	Label

Police	NN	1	0	0	O
activity	NN	0	0	0	O
on	IN	0	0	0	O
I-39NB	NN	0	0	0	O

All these examples support with evidence the evaluation metrics presented in Section 7.1.2.3., and reinforce the claim that LORE helps extract otherwise missed locative references, which translated to improved recall, whereas nLORE helps avoid the extraction of wrong items, which means better precision scores. Also, in Table 65 from that section, we can observe that LORE clearly outperformed the extended 7k nLORE model, being almost 10x times quicker than its probabilistic-based counterpart. This supports the claim that rule-based models show better runtime efficiency than probabilistic-based models (Chiticariu et al., 2013).

7.2.3. Issues, limitations, and areas of improvement

Overall, despite the optimal performance of our rule-based and probabilistic-based models, we attested some issues and limitations. We address these and suggest some areas of improvement that could help our model be less error-prone.

In this regard, we would like to emphasize that using a Twitter-specific POS tagger that handles with greater effectivity the noise-prone character of tweets could probably help improve the recognition of many locative references that go unnoticed due to wrongly assigned POS tags. Since our rule-based model heavily relies on the detection of proper nouns captured by the place-name dataset or signaled by location-indicative nouns and locative markers to capture and delimit locative references, this could very much improve the performance of LORE. By the same token, nLORE could largely benefit from a POS tagger like the one suggested to learn and infer predictions with greater accuracy, although it has shown greater capabilities in handling mislabeled POS tags in the task of location detection.

A robust and fast abbreviation and disambiguation system together with spell checking for fast text normalization might also improve the performance of our models. However, we have not found enough grounded reasons to perform spell checking, since tweet texts were, as observed in the development stage of LORE, mostly well-written and not as informal as widely believed in the literature. Moreover, any spell checking process could severely affect timing with a poor cost-benefit ratio, as demonstrated in the use of a spell checker in the initial stages of the present thesis. Due to the very slight margin in performance improvements, we opted not to use one, though we do not discard this possibility in future research.

In addition, our models might benefit from knowing the affected locations beforehand, since, in that case, we could retrieve from geographic ontologies and databases the full scope of

specific geographic areas where emergency-related events take place and thus effectively perform entity-based matching with even increased granularity and accuracy. In this regard, detecting locative references at a local level has, in fact, been shown to be a much easier task than global-scale location detection (Wallgrün et al., 2018). However, as our focus lay in the reusability and generalizability of our models with any emergency or crisis-related event, past or future and local or global, as represented in tweet corpora, we did not focus on retrieving particular geographic areas from the GeoNames database, but the full GeoNames database for our place-name dataset.

Furthermore, many cases of natural language ambiguity, especially geo and non-geo ambiguity, may not be solved with current NER and NEM techniques because these methods cannot as yet parse the underlying semantics of sentences. In other words, relying on linguistic clues alone cannot disambiguate all locative references in sentences of the type *Paris (person name) said that Parisian (demonym) artists don't have to live in Paris (place name)* without an adequate deep-semantics and syntactic framework (Gritta et al., 2019). However, in the discussion of nLORE, we provided examples that showed the potential of neural networks in understanding the syntactic context, and thus solving this ambiguity. In other words, probabilistic-based methods that rely on neural networks and recent AI methods may be able to capture the linguistic context with greater depth and accuracy than rule-based methods. Another way of approaching this conundrum, from a linguistic point of view, would involve developing a system based on deep semantics and syntactic parsing that could tackle, with great solvency, the challenges posed by natural languages. If we had a semantic and syntactic representation of each tweet, we could unveil the conceptual and syntactic relationships among the events and participants expressed in the tweets, and derive logical conclusions from these relationships. In this sense, we could exploit FunGramKB capabilities, a lexico-conceptual knowledge base that integrates rich semantic and syntactic knowledge (Periñán Pascual & Arcas Túnez, 2007, 2010). We thus leave open a very interesting future line of research.

With reference to the use of train corpora in nLORE, we wonder whether using larger corpora would have translated to a linear improvement in terms of performance, and how to uncover, if possible, the mathematical rule that governs such linear growth of performance, or whether improvement eventually reaches a plateau. Other issues or potential areas for research relate to the use of different novel approaches in DL based on Transformers or language models such as BERT. This could pave the way for future research, improving the performance of nLORE.

8. CONCLUSIONS

We have presented LORE, a multilingual, linguistically-aware model based on a rule-based approach that exploits rich linguistic knowledge and different NLP techniques, achieving state-of-the-art performance and outperforming well-known, state-of-the-art NER tools in tweet location detection without the high computational cost, time, and resources characteristic of probabilistic-based frameworks. Our rule-based approach is novel and innovative, in that we entirely rely on linguistic knowledge and lexical resources, achieving great results in the task at hand. The integration of lexically-rich datasets with NLP techniques such as tokenization, POS tagging, n-grams and regular expressions helped recognize coarse-grained location types (i.e. geopolitical entities and natural geographic references) and fine-grained location types (i.e. POIs, addresses, and different traffic ways). To the best of our knowledge, our model is the first that makes use of EuroWordNet-based datasets of location-indicative nouns for the identification of locative references. Moreover, LORE can retrieve complex locative references consisting of any location-indicative word and/or locative marker accompanying a given place name. This semantic granularity constitutes in itself a great qualitative advantage over other location-detection models, as shown by the highest recall achieved in all of the languages supported in the evaluation metrics. Although most researchers agree that street and building extraction performs worse than geopolitical entity extraction (Middleton et al., 2018), the linguistic-processing module has shown that this is no longer the case for its fine-grained detection capabilities for those location types. Also, our model has shown a great quantitative advantage in that it has outperformed state-of-the-art NER systems not only in precision but also, and by a large margin, in recall, due to the diversity and variety of the location types extracted. Both these quantitative and qualitative advantages are particularly useful to avoid missing locative references that could greatly contribute to raising emergency-situation awareness in real-life crisis scenarios, a key component for emergency-based services.

Our rule-based model is also multi-faceted, scalable, versatile, and reusable in that:

(i) the modular architecture of the model, consisting of two primary modules (i.e. the place-name search and the linguistic processing), synergistically work in the location-detection task, making up for any performance loss that may occur due to the noisy nature of tweets and the challenges offered by natural language ambiguity;

(ii) since it is not particularly suited to any local event or domain, similar performance can be expected on any collection of tweets about any emergency-related setting, as shown with the English eval corpus II, and

(iii) the modular architecture of linguistic knowledge facilitates the adaptation of the model's functionalities to languages other than English by means of semi-automatic methods that allow the end user to modify or update the language-dependent lexical resources and regex-based rules, which makes the model ideal in multilingual contexts.

We have also presented nLORE, the DL counterpart of LORE, feeding off the linguistic knowledge provided by LORE and trained on a relatively small corpus of English tweets, to infer and extract locative references with greater accuracy than LORE. In many respects, nLORE overcame some of the limitations presented by LORE, supported by the improved evaluation scores. At the same time, it has also shown that linguistic-based feature engineering in probabilistic-based approaches may still provide a much-valued added benefit –though slight–, which could pave the way for more linguistic-oriented computational work in the field of NER. This sentiment goes in line with recent calls in the linguistic and computational communities, requesting a greater interaction between linguistics and AI (Linzen, 2019).

We propose the following lines of future research on the basis of the issues and limitations found in our research enterprise to enhance the performance of our models and extend their functionalities:

- Since our model heavily relies on the performance of a POS tagger for the detection of proper nouns in locative references, a Twitter-trained POS tagger could in this regard recognize otherwise missed locative references and avoid retrieving wrong items, all due to mislabeled POS tags.
- Tweaking the rules can be a feasible outcome after the comprehensive error analysis stage carried out in the evaluation phase of the models and the rules. This could thus improve precision and recall.
- The system could also enrich location semantics by providing, besides the locative reference and its ID, a semantically-rich characterization of the location type (i.e. city, POI, road, etc.), if possible.
- We consider extending nLORE to languages other than English, such as Spanish or French, as its rule-based counterpart, by compiling and annotating tweet corpora in those languages, and comparing their performance.
- Also, other state-of-the-art methods and techniques in NLP and AI involving Transformers or language models such as BERT may be used to improve the capabilities and performance of nLORE.
- At the same time, and with a view to endowing the system with a heavier linguistic focus, the rule-based system could exploit rich semantic and syntactic knowledge in the tweets by means of FunGramKB deep-semantic and syntactic parsing for intelligent location detection.
- Implementing a geocoding module could help in the disambiguation phase of the extracted locative references.

- Deploying the model in a full-blown application for crisis or emergency tracking to explore the capacity of LORE or nLORE in real-time location detection. This would make our model be even more useful for competent authorities and emergency responders interested in using our models for real-life crisis-related scenarios that demand instant, accessible and accurate geospatial information.

9. BIBLIOGRAPHY

Acheson, E., De Sabbata, S., & Purves, R. S. (2017). A quantitative analysis of global gazetteers: Patterns of coverage for common feature types. *Computers, Environment and Urban Systems*, *64*, 309–320. https://doi.org/10.1016/j.compenvurbsys.2017.03.007

Aggarwal, C. C. (2013). *Managing and Mining Sensor Data*. Springer US.

Aguilar, G., Maharjan, S., López Monroy, A. P., & Solorio, T. (2018). A Multi-task Approach for Named Entity Recognition in Social Media Data. In *Proceedings of the 3rd Workshop on Noisy User-generated Text* (pp. 148–153). Copenhagen, Denmark: Association for Computational Linguistics. https://doi.org/10.18653/v1/W17-4419

Aguilar, G., Maharjan, S., López Monroy, A. P., & Solorio, T. (2019). *A Multi-task Approach for Named Entity Recognition in Social Media Data*. https://doi.org/10.18653/v1/w17-4419

Ahlers, D. (2013). Assessment of the accuracy of GeoNames gazetteer data. In *Proceedings of the 7th Workshop on Geographic Information Retrieval - GIR '13* (pp. 74–81). https://doi.org/10.1145/2533888.2533938

Ahmed, M. F., Vanajakshi, L., & Suriyanarayanan, R. (2019). Real-Time Traffic Congestion Information from Tweets Using Supervised and Unsupervised Machine Learning Techniques. *Transportation in Developing Economies*, *5*(2). https://doi.org/10.1007/s40890-019-0088-2

Al-Olimat, H. S., Shalin, V. L., Thirunarayan, K., & Sain, J. P. (2019). Towards Geocoding Spatial Expressions (Vision Paper). In *SIGSPATIAL '19* (pp. 75–78). Chicago, Illinois, USA. https://doi.org/10.1145/3347146.3359356

Al-Olimat, H. S., Thirunarayan, K., Shalin, V., & Sheth, A. (2018). Location Name Extraction from Targeted Text Streams using Gazetteer-based Statistical Language Models. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 1986–1997). Association for Computational Linguistics. Retrieved from

http://arxiv.org/abs/1708.03105

Alex, B., Llewellyn, C., Grover, C., Oberlander, J., & Tobin, R. (2016). Homing in on Twitter users: Evaluating an enhanced geoparser for user profile locations. In *LREC 2016* (pp. 3936–3944). Retrieved from http://homepages.inf.ed.ac.uk/balex/publications/LREC2016.pdf

Amitay, E., Har'El, N., Sivan, R., & Soffer, A. (2004). Web-a-Where: Geotagging Web Content. *Proceedings of SIGIR '04 Conference on Research and Development in Information Retrieval*, 273–280. https://doi.org/10.1145/1008992.1009040

Arthur, R., Boulton, C. A., Shotton, H., & Williams, H. T. P. (2018). Social sensing of floods in the UK. *PLoS ONE*, *13*(1), 1–18. https://doi.org/10.1371/journal.pone.0189327

Avvenuti, M., Cresci, S., Nizzoli, L., & Tesconi, M. (2018). Geoparsing and Geotagging with Machine Learning on top of Linked Data. In *Extended Semantic Web Conference (ESWC)* (pp. 1–15). https://doi.org/10.1007/978-3-319-93417-4_2

Baldwin, T., Cook, P., Lui, M., MacKinlay, A., & Wang, L. (2013). How Noisy Social Media Text, How Diffrnt Social Media Sources? *International Joint Conference on Natural Language Processing*, (October), 356–364. Retrieved from http://www.aclweb.org/anthology/I13-1041

Barrière, C. (2016). Searching for Named Entities. In *Natural Language Understanding in a Semantic Web Context* (pp. 23–38). Switzerland: Springer International Publishing. https://doi.org/10.1007/978-3-319-41337-2

Bateman, J. A., Hois, J., Ross, R., & Tenbrink, T. (2010). A linguistic ontology of space for natural language processing. *Artificial Intelligence*, *174*(14), 1027–1071. https://doi.org/10.1016/j.artint.2010.05.008

Béchet, F., Sagot, B., Stern, R., Université, A. M., Luminy, D., Cedex, L. C., … Bourse, D. (2011). Coopération de méthodes statistiques et symboliques pour l'adaptation non-supervisée d'un système d'étiquetage en entités nommées. In *TALN 2011* (pp. 1–6). Montpellier.

Bennett, B., & Agarwal, P. (2007). Semantic Categories Underlying the Meaning of 'Place.' In S. Winter, M. Duckham, L. Kulik, & B. Kuipers (Eds.), *Spatial Information Theory* (Vol. 4736, pp. 78–95). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-74788-8_6

Bird, S. (2006). NLTK: The Natural Language Toolkit. In *Proceedings of COLING/ACL* (pp. 69–72). Sidney, Australia: Association for Computational Linguistics.

https://doi.org/10.3115/1225403.1225421

Cambria, E., & White, B. (2014). Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine*, (May), 48–57.

Cameron, M. A., Power, R., Robinson, B., & Yin, J. (2012). Emergency situation awareness from twitter for crisis management. *Proceedings of the 21st International Conference Companion on World Wide Web - WWW '12 Companion*, (September 2014), 695. https://doi.org/10.1145/2187980.2188183

Cheng, Z., Caverlee, J., & Lee, K. (2010). You are where you tweet: A content-based approach to geo-locating Twitter users. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (pp. 759–768). https://doi.org/10.1145/1871437.1871535

Chiticariu, L., Li, Y., & Reiss, F. R. (2013). Rule-based information extraction is dead! Long live rule-based information extraction systems! In *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference* (pp. 827–832).

Chong, W.-H., & Lim, E.-P. (2018). Exploiting User and Venue Characteristics for Fine-Grained Tweet Geolocation. *ACM Transactions on Information Systems*, *36*(3), 1–34. https://doi.org/10.1145/3156667

Cinque, G., & Rizzi, L. (Eds.). (2010). *Mapping Spatial PPs*. New York: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780195393675.001.0001

Cole, M. R. (2018). *Hands-On Neural Network Programming with C#*. Birmingham, UK: Packt Publishing.

Coventry, K. R., & Garrod, S. C. (2004). *Saying, Seeing and Acting: The Psychological Semantics of Spatial Prepositions*. Taylor & Francis Routledge. https://doi.org/10.4324/9780203641521

Creary, L. G., Gawron, J. M., & Nerbonne, J. (1989). Reference to locations. In *ACL '89 Proceedings of the 27th annual meeting on Association for Computational Linguistics* (pp. 42–50). https://doi.org/10.3115/981623.981629

Crooks, A., Croitoru, A., Stefanidis, A., & Radzikowski, J. (2013). #Earthquake: Twitter as a Distributed Sensor System. *Transactions in GIS*, *17*(1), 124–147. https://doi.org/10.1111/j.1467-9671.2012.01359.x

Cunningham, H., Maynard, D., Bontcheva, K., & Tablan, V. (2002). GATE: A Framework and

Graphical Development Enviroment for Robust NLP Tools and Applications. *Proc. of the 40th Anniversary Meeting of the Association for Computational Linguistics, ACL '02*, (July).

Daly, E. M., Lecue, F., & Bicer, V. (2013). Westland row why so slow? Fusing social media and linked data sources for understanding real-time traffic conditions. *International Conference on Intelligent User Interfaces, Proceedings IUI*, 203–212. https://doi.org/10.1145/2449396.2449423

Das, R. D., & Purves, R. S. (2019). Exploring the Potential of Twitter to Understand Traffic Events and Their Locations in Greater Mumbai, India. *IEEE Transactions on Intelligent Transportation Systems*, 1–10. https://doi.org/10.1109/TITS.2019.2950782

de Bruijn, J. A., de Moel, H., Jongman, B., Wagemaker, J., & Aerts, J. C. J. H. (2018). TAGGS: Grouping Tweets to Improve Global Geoparsing for Disaster Response. *Journal of Geovisualization and Spatial Analysis*, *2*(2), 1–14. https://doi.org/10.1007/s41651-017-0010-6

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Retrieved from http://arxiv.org/abs/1810.04805

Di Rocco, L., Buscaldi, D., Bertolotto, M., Catania, B., & Guerrini, G. (2019). The role of geographic knowledge in sub-city level geolocation. In *34th ACM/SIGAPP Symposyum On Applied Computing* (pp. 1–7). Limassol, Cyprus. https://doi.org/10.1145/3297280.3297557

Dittrich, A., Richter, D., & Lucas, C. (2014). Analysing the Usage of Spatial Prepositions in Short Messages. In G. Gartner & H. Huang (Eds.), *Progress in Location-Based Services* (pp. 153–171). https://doi.org/10.1007/978-3-319-11879-6

Dredze, M., Paul, M. J., Bergsma, S., & Tran, H. (2013). Carmen: A twitter geolocation system with applications to public health. In *Expanding the Boundaries of Health Informatics Using Artificial Intelligence: Papers from the AAAI 2013 Workshop* (pp. 20–24). https://doi.org/10.2218/ijdc.v9i1.318

Dugas, F., & Nichols, E. (2016). DeepNNNER : Applying BLSTM-CNNs and Extended Lexicons to Named Entity Recognition in Tweets. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)* (pp. 178–187). Osaka, Japan. Retrieved from https://www.aclweb.org/anthology/W16-3924

Dutt, R., Basu, M., Ghosh, K., & Ghosh, S. (2019). Utilizing microblogs for assisting post-

disaster relief operations via matching resource needs and availabilities. *Information Processing and Management*, *56*(5), 1680–1697. https://doi.org/10.1016/j.ipm.2019.05.010

Dutt, R., Hiware, K., Ghosh, A., & Bhaskaran, R. (2018). SAVITR: A System for Real-time Location Extraction from Microblogs during Emergencies. In *WWW'18 Companion: The 2018 Web Conference Companion* (pp. 1643–1649). Lyon, France. https://doi.org/10.1145/3184558.3191623

Eisenstein, J. (2013). What to do about bad language on the internet. *Proceedings of NAACL-HLT 2013*, (June), 359–369. https://doi.org/10.1109/GEOINFORMATICS.2010.5567952

Eke, P. I. (2011). Using Social Media for Research and Public Health Surveillance. *Journal of Dental Research*, *90*(9), 1045–1046. https://doi.org/10.1177/0022034511415277

Elad, K., Benny, K., Yaron, K., & Roi, R. (2020). Automatic Location Type Classification From Social-Media Posts. In *Proceedings of ACM Conference (Conference'17)* (pp. 1–10). New York, NY, USA.

Elman, J. L. (1990). Structure in Time. *Cognitive Science*, *14*, 179–211.

Espinosa, K. J., Batista-Navarro, R., & Ananiadou, S. (2016). Learning to recognise named entities in tweets by exploiting weakly labelled data. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)* (pp. 153–163). Osaka, Japan: The COLING 2016 Organizing Committee.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. *International Journal of Lexicography*. Cambridge, MA: MIT Press.

Finkel, J. R., Grenager, T., & Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics - ACL '05*, (1995), 363–370. https://doi.org/10.3115/1219840.1219885

Firth, J. R. (1957). *A synopsis of linguistic theory 1930-1955*. Oxford: Oxford University Press.

Frank, A. U., & Mark, D. M. (1991). Language Issues for Geographical Information Systems. In D. J. Maguire, M. F. Goodchild, & D. W. Rhind (Eds.), *Geographical Information Systems: Principles and Applications* (pp. 147–163). London: Longman Publishers.

Gao, C., & Li, Y. (2019). Fine-Grained Geolocalization of User-Generated Short Text Based on a Weight Probability Model. *IEEE Access*, *7*, 153579–153591. https://doi.org/10.1109/ACCESS.2019.2948355

Geis, M. L. (1975). English Time and Place Adverbials. *Ohio State University Working Papers in Linguistics, 18*, 1–11. Retrieved from https://kb.osu.edu/bitstream/handle/1811/81364/WPL_18_June_1975_001.pdf

Gelernter, J., & Balaji, S. (2013). An algorithm for local geoparsing of microtext. *GeoInformatica, 17*(4), 635–667. https://doi.org/10.1007/s10707-012-0173-8

Gerguis, M. N., Salama, C., & Watheq El-Kharashi, M. (2016). ASU: An Experimental Study on Applying Deep Learning in Twitter Named Entity Recognition. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)* (pp. 188–196). Osaka, Japan. Retrieved from https://www.aclweb.org/anthology/W16-3925

Ghahremanlou, L., Sherchan, W., & Thom, J. A. (2014). Geotagging twitter messages in crisis management. *Computer Journal*, *58*(9), 1937–1954. https://doi.org/10.1093/comjnl/bxu034

Gonzalez-Paule, J. D. (2019). *Inferring the Geolocation of Tweets at a Fine-Grained Level*. University of Glasgow.

Gonzalez-Paule, J. D., Sun, Y., & Moshfeghi, Y. (2019). On fine-grained geolocalisation of tweets and real-time traffic incident detection. *Information Processing and Management*, *56*(3), 1–14. https://doi.org/10.1016/j.ipm.2018.03.011

Gonzalez, R., Figueroa, G., & Chen, Y. S. (2012). TweoLocator: A non-intrusive geographical locator system for Twitter. In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on Location-Based Social Networks, LBSN 2012 - Held in Conjunction with ACM SIGSPATIAL GIS 2012* (pp. 24–31). https://doi.org/10.1145/2442796.2442804

Gorinski, P. J., Wu, H., Grover, C., Tobin, R., Talbot, C., Whalley, H., … Alex, B. (2019). *Named Entity Recognition for Electronic Health Records: A Comparison of Rule-based and Machine Learning Approaches*. Retrieved from http://arxiv.org/abs/1903.03985

Goyal, A., Gupta, V., & Kumar, M. (2018). Recent Named Entity Recognition and Classification techniques: A systematic review. *Computer Science Review, 29*, 21–43. https://doi.org/10.1016/j.cosrev.2018.06.001

Gregory, I., Donaldson, C., Murieta-Flores, P., & Rayson, P. (2015). Geoparsing, GIS, and Textual Analysis: Current Developments in Spatial Humanities Research. *International Journal of Humanities and Arts Computing*, *11*(1), 1–14. https://doi.org/10.3366/ijhac.2015.0135

Gritta, M., Pilehvar, M. T., & Collier, N. (2019). *A Pragmatic Guide to Geoparsing Evaluation*. Retrieved from http://arxiv.org/abs/1810.12368

Gritta, M., Pilehvar, M. T., Limsopatham, N., & Collier, N. (2018). What's missing in geographical parsing? *Language Resources and Evaluation*, *52*(2), 603–623. https://doi.org/10.1007/s10579-017-9385-8

Grossman, D., & Frieder, O. (2004). *Information Retrieval: Algorithms and Heuristics* (2nd ed.). Springer Netherlands.

Gurney, K. (1997). *An introduction to neural networks*. London, UK: UCL Press. https://doi.org/10.1016/S0140-6736(95)91746-2

Hamzei, E., Winter, S., & Tomko, M. (2019). Initial Analysis of Simple Where-Questions and Human-Generated Answers. In S. Timpf, C. Schlieder, M. Kattenbeck, B. Ludwig, & K. Stewart (Eds.), *14th International Conference on Spatial Information Theory (COSIT 2019)* (pp. 1–8). Dagstuhl, Germany: Schloss Dagstuhl: Leibniz-Zentrum fuer Informatik. https://doi.org/10.4230/LIPIcs.COSIT.2019.12

Han, B., Cook, P., & Baldwin, T. (2014). Text-based twitter user geolocation prediction. *Journal of Artificial Intelligence Research*, *49*, 451–500. https://doi.org/10.1613/jair.4200

Han, B., Jimeno-Yepes, A., Mackinlay, A., & Chen, Q. (2014). Identifying Twitter Location Mentions. In *Proceedings of the Australasian Language Technology Association Workshop 2014* (pp. 157–162). Melbourne, Australia.

Hernandez-Suarez, A., Sanchez-Perez, G., Toscano-Medina, K., Perez-Meana, H., Portillo-Portillo, J., and Luis, V. S., & Javier García Villalba, L. (2019). Using Twitter Data to Monitor Natural Disaster Social Dynamics: A Recurrent Neural Network Approach with Word Embeddings and Kernel Density Estimation. *Sensors*, *19*(7), 1–22. https://doi.org/10.3390/s19071746

Herskovits, A. (1985). Semantics and pragmatics of locative expressions. *Cognitive Science*, *9*(3), 341–378. https://doi.org/10.1016/S0364-0213(85)80003-3

Hoang, T. B. N., & Mothe, J. (2018). Location extraction from tweets. *Information Processing and Management*, *54*(2). https://doi.org/10.1016/j.ipm.2017.11.001

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hu, Yingjie. (2018a). Geo-text data and data-driven geospatial semantics. *Geography Compass*, *12*(11), 1–19. https://doi.org/10.1111/gec3.12404

Hu, Yingjie. (2018b). Geospatial Semantics. In B. Huang, T. J. Covas, & M.-H. Tsou (Eds.), *Comprehensive Geographic Information Systems* (pp. 80–94). Oxford, UK: Elsevier.

https://doi.org/10.1016/B978-0-12-409548-9.09597-X

Hu, Yingjie, & Adams, B. (2020). Harvesting Big Geospatial Data from Natural Language Texts, 1–23.

Hu, Yuheng, Talamadupula, K., & Kambhampati, S. (2013). Dude, srsly?: The Surprisingly Formal Nature of Twitter's Language. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media - ICWSM '13* (pp. 244–253). https://doi.org/10.1.1.297.589

Ikawa, Y., Vukovic, M., Rogstadius, J., & Murakami, A. (2016). Location-based insights from the social web. In *WWW '13 Companion Proceedings of the 22nd International Conference on World Wide Web* (pp. 1013–1016). Rio de Janeiro, Brazil. https://doi.org/10.1145/2487788.2488107

Imran, M., Castillo, C., Diaz, F., & Vieweg, S. (2014). *Processing Social Media Messages in Mass Emergency: A Survey*. *ACM Computing Surveys* (Vol. 47). https://doi.org/10.1145/3184558.3186242

Ingersoll, G. S., Morton, T. S., & Farris, A. L. (2013). *Taming Text*. Manning Publications.

Inkpen, D., Liu, J., Farzindar, A., Kazemi, F., & Ghazi, D. (2017). Location detection and disambiguation from twitter messages. *Journal of Intelligent Information Systems*, *49*(2), 237–253. https://doi.org/10.1007/s10844-017-0458-3

Izbicki, M., Papalexakis, V., & Tsotras, V. (2019). Geolocating Tweets in any Language at any Location. In *CIKM '19 Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 89–98). Beijing, China. https://doi.org/10.1145/3357384.3357926

Janowicz, K., Gao, S., Mckenzie, G., & Hu, Y. (2019). GeoAI : spatially explicit artificial intelligence techniques for geographic knowledge discovery and beyond. *International Journal of Geographical Information Science*, 1–12. https://doi.org/10.1080/13658816.2019.1684500

Jones, C. B., & Purves, R. S. (2008). Geographical information retrieval. *International Journal of Geographical Information Science*, *22*(3), 219–228. https://doi.org/10.1080/13658810701626343

Jongman, B., Wagemaker, J., Romero, B., & de Perez, E. (2015). Early Flood Detection for Rapid Humanitarian Response: Harnessing Near Real-Time Satellite and Twitter Signals. *ISPRS International Journal of Geo-Information*, *4*(4), 2246–2266. https://doi.org/10.3390/ijgi4042246

Jurafsky, D., & Martin, J. H. (2018a). Information Extraction. In *Speech and Language Processing* (pp. 1–31). Book in preparation.

Jurafsky, D., & Martin, J. H. (2018b). Regular Expressions, Text Normalization, Edit Distance. In *Speech and Language Processing* (pp. 1–28). Book in preparation.

Jurafsky, D., & Martin, J. H. (2019a). Sequence Processing with Recurrent Networks. In *Speech and Language Processing* (pp. 1–23). Book in preparation.

Jurafsky, D., & Martin, J. H. (2019b). Vector Semantics and Embeddings. In *Speech and Language Processing*. Book in preparation.

Karimzadeh, M., Pezanowski, S., MacEachren, A. M., & Wallgrün, J. O. (2019). GeoTxt: A scalable geoparsing system for unstructured text geolocation. *Transactions in GIS*, *23*(1), 118–136. https://doi.org/10.1111/tgis.12510

Kew, T., Shaitarova, A., Meraner, I., Goldzycher, J., Clematide, S., & Volk, M. (2019). Geotagging a Diachronic Corpus of Alpine Texts: Comparing Distinct Approaches to Toponym Recognition. In *Recent Advances in Natural Language Processing (RANLP)* (pp. 11–19).

Khodabandeh-Shahraki, Z., Fatemi, A., & Tabatabaee Malazi, H. (2019). Evidential fine-grained event localization using Twitter. *Information Processing and Management*, *56*(6), 102045. https://doi.org/10.1016/j.ipm.2019.05.006

Kinsella, S., Murdock, V., & O'Hare, N. (2011). "I'm eating a sandwich in Glasgow." In *Proceedings of the 3rd international workshop on Search and mining user-generated contents - SMUC '11* (p. 61). https://doi.org/10.1145/2065023.2065039

Kracht, M. (2002). On the Semantics of Locatives. *Linguistics and Philosophy*, *25*(2), 157–232.

Kumar, A., & Singh, J. P. (2019). Location reference identification from tweets during emergencies: A deep learning approach. *International Journal of Disaster Risk Reduction*, *33*, 365–375. https://doi.org/10.1016/j.ijdrr.2018.10.021

Lafferty, J., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning* (Vol. 8, pp. 282–289). https://doi.org/10.1038/nprot.2006.61

Landau, B., & Jackendoff, R. (1993). "What" and "where" in spatial language and spatial cognition. *Behavioral and Brain Sciences*, *16*(2), 217–265. https://doi.org/10.1017/s0140525x00029733

Laurini, R., & Kazar, O. (2016). Geographic Ontologies : Survey and Challenges. *Journal for Theoretical Cartography*, *9*(March), 1–13.

Le, N. T., Mallek, F., & Sadat, F. (2016). UQAM-NTL : Named entity recognition in Twitter messages Computer Science Faculty. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)* (pp. 197–202).

Leidner, J., & Lieberman, M. (2011). Detecting geographical references in the form of place names and associated spatial natural language. *SIGSPATIAL Special*, *3*(2), 5–11. https://doi.org/10.1145/2047296.2047298

Levinson, S. C. (2003). *Space in Language and Cognition: Explorations in Cognitive Diversity*. Cambridge, UK: Cambridge University Press.

Li, C., & Sun, A. (2014). Fine-grained location extraction from tweets with temporal awareness. In *SIGIR 2014* (pp. 43–52). https://doi.org/10.1145/2600428.2609582

Li, J., Sun, A., Han, J., & Li, C. (2018). *A Survey on Deep Learning for Named Entity Recognition*. Retrieved from http://arxiv.org/abs/1812.09449

Li, W., Serdyukov, P., de Vries, A. P., Eickhoff, C., & Larson, M. (2011). The where in the tweet. In *Proceedings of the 20th ACM international conference on Information and knowledge management - CIKM '11* (p. 2473). https://doi.org/10.1145/2063576.2063995

Limsopatham, N., & Collier, N. (2016). Bidirectional LSTM for Named Entity Recognition in Twitter Messages. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)* (pp. 145–152). Osaka, Japan. Retrieved from https://www.aclweb.org/anthology/W16-3920

Lingad, J., Karimi, S., & Yin, J. (2013). Location extraction from disaster-related microblogs. In *Proceedings of the 22nd International Conference on World Wide Web - WWW '13 Companion* (pp. 1017–1020). https://doi.org/10.1159/000152140

Linzen, T. A. L. (2019). What can linguistics and deep learning contribute to each other? Response to pater. *Language*, *95*(1), 99–108. https://doi.org/10.1353/lan.2019.0015

Liu, F., Vasardani, M., & Baldwin, T. (2014). Automatic Identification of Locative Expressions from Social Media Text. In *Proceedings of the 4th International Workshop on Location and the Web* (pp. 9–16). Shangai. https://doi.org/10.1145/2663713.2664426

Liu, X., Zhou, M., Wei, F., Fu, Z., & Zhou, X. (2012). Joint inference of named entity recognition and normalization for tweets. In *50th Annual Meeting of the Association for Computational Linguistics, ACL 2012 - Proceedings of the Conference* (Vol. 1, pp. 526–

535).

Llisterri, J. (2003). Lingüística y tecnologías del lenguaje. *Lynx*, *2*, 9–71.

Luo, X., Zimet, G., & Shah, S. (2019). A natural language processing framework to analyse the opinions on HPV vaccination reflected in twitter over 10 years (2008 - 2017). *Human Vaccines & Immunotherapeutics*, *15*(7–8), 1496–1504. https://doi.org/10.1080/21645515.2019.1627821

Magge, A., Weissenbacher, D., Sarker, A., Scotch, M., & Gonzalez-Hernandez, G. (2018). Deep neural networks and distant supervision for geographic location mention extraction. *Bioinformatics*, *34*(13), i565–i573. https://doi.org/10.1093/bioinformatics/bty273

Malmasi, S., & Dras, M. (2016). Location mention detection in tweets and microblogs. In K. Hasida & A. Purwarianti (Eds.), *Communications in Computer and Information Science* (Vol. 593, pp. 123–134). Singapore: Springer Singapore. https://doi.org/10.1007/978-981-10-0515-2_9

Martínez-Rojas, M., Pardo-Ferreira, M. del C., & Rubio-Romero, J. C. (2018). Twitter as a tool for the management and analysis of emergency situations: A systematic literature review. *International Journal of Information Management*, *43*(April), 196–208. https://doi.org/10.1016/j.ijinfomgt.2018.07.008

McDonough, K., Moncla, L., & van de Camp, M. (2019). Named entity recognition goes to old regime France: geographic text analysis for early modern French corpora. *International Journal of Geographical Information Science*, *00*(00), 1–25. https://doi.org/10.1080/13658816.2019.1620235

Middleton, S. E., Kordopatis-Zilos, G., Papadopoulos, S., & Kompatsiaris, Y. (2018). Location Extraction from Social Media. *ACM Transactions on Information Systems*, *36*(4), 1–27. https://doi.org/10.1145/3202662

Middleton, S. E., Middleton, L., & Modafferi, S. (2014). Real-time crisis mapping of natural disasters using social media. *IEEE Intelligent Systems*, *29*(2), 9–17. https://doi.org/10.1109/MIS.2013.126

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the International Conference on Learning Representations (ICLR 2013)* (pp. 1–12). Retrieved from http://arxiv.org/abs/1301.3781

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, *38*(11), 39–41. https://doi.org/10.1145/219717.219748

Miyazaki, T., Cohn, T., & Baldwin, T. (2018). Twitter Geolocation using Knowledge-Based Methods. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text* (pp. 7–16). https://doi.org/http://dx.doi.org/10.18653/v1/W18-6102

Moncla, L., Renteria-Agualimpia, W., Nogueras-Iso, J., & Gaio, M. (2014). Geocoding for texts with fine-grain toponyms: An experiment on a geoparsed hiking descriptions corpus. In *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems* (pp. 183–192). https://doi.org/10.1145/2666310.2666386

Mourad, A., Scholer, F., Magdy, W., & Sanderson, M. (2019). A Practical Guide for the Effective Evaluation of Twitter User Geolocation. *ACM*, 1–23. https://doi.org/10.1145/3352572

Murrieta-Flores, P., & Martins, B. (2019). The geospatial humanities: past, present and future. *International Journal of Geographical Information Science*, *33*(12), 2424–2429. https://doi.org/10.1080/13658816.2019.1645336

Murthy, D. (2018). *Twitter: Social Communication in the Twitter Age* (2nd ed.). Malden, MA: Polity Press.

Nadeau, D., & Sekine, S. (2007). A Survey of Named Entity Recognition and Classification. *Lingvisticae Investigationes*, *30*(1), 3–26. https://doi.org/10.1162/COLI_a_00178

Navarro, G. (2001). A guided tour to approximate string matching. *ACM Computing Surveys*, *33*(1), 31–88. https://doi.org/10.1145/375360.375365

Negrov, D. V., Karandashev, I. M., Shakirov, V. V., Matveyev, Y. A., Dunin-Barkowski, W. L., & Zenkevich, A. V. (2015). An Approximate Backpropagation Learning Rule for Memristor Based Neural Networks Using Synaptic Plasticity, (November). Retrieved from http://arxiv.org/abs/1511.07076

Ogawa, A., & Hori, T. (2015). ASR error detection and recognition rate estimation using deep bidirectional recurrent neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4370–4374).

Periñán-Pascual, C. (2012). En defensa del procesamiento del lenguaje natural fundamentado en la lingüística teórica. *Onomázein*, *26*, 13–48. Retrieved from http://www.fungramkb.com/resources/papers/022.pdf

Periñán-Pascual, C. (2017). Bridging the gap within text-data analytics : a computer environment for data analysis in linguistic research. *Revista de Lenguas Para Fines Específicos*, *23*(2), 111–132.

Periñán-Pascual, C., & Arcas-Túnez, F. (2017). A Knowledge-Based Approach to Social Sensors for Environmentally-Related Problems. In C. Analide & P. Kim (Eds.), *Intelligent Environments 2017* (Vol. 22, pp. 49–58). Amsterdam: IOS Press. https://doi.org/10.3233/978-1-61499-796-2-49

Periñán-Pascual, C., & Arcas-Túnez, F. (2018). The Analysis of Tweets to Detect Natural Hazards. In I. Chatzigiannakis, Y. Tobe, P. Novais, & O. Amft (Eds.), *Intelligent Environments 2018* (pp. 87–96). Amsterdam: IOS Press. https://doi.org/10.3233/978-1-61499-874-7-87

Periñán-Pascual, C., & Arcas-Túnez, F. (2019). Detecting environmentally-related problems on Twitter. *Biosystems Engineering*, *177*, 31–48. https://doi.org/10.1016/j.biosystemseng.2018.10.001

Periñán Pascual, C., & Arcas Túnez, F. (2007). Deep semantics in an NLP knowledge base. *12th Conference of the Spanish Association for Artificial Intelligence*, 279–288.

Periñán Pascual, C., & Arcas Túnez, F. (2010). Ontological commitments in FunGramKB. *Procesamiento Del Lenguaje Natural*, *44*, 27–34.

Potts, L., Seitzinger, J., Jones, D., & Harrison, A. (2011). Tweeting disaster: Hashtag Constructions and Collisions. *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 235–240. https://doi.org/10.1145/2038476.2038522

Priedhorsky, R., Culotta, A., & Del Valle, S. Y. (2014). Inferring the Origin Locations of Tweets with Quantitative Confidence. In *CSCW '14 Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing* (pp. 1523–1536). https://doi.org/10.1016/j.tsf.2010.06.052

Purves, R. S., Clough, P., Jones, C. B., Hall, M. H., & Murdock, V. (2018). Geographic Information Retrieval: Progress and Challenges in Spatial Search of Text. In *Foundations and Trends in Information Retrieval* (Vol. 12, pp. 164–318). https://doi.org/10.1561/1500000034

Purves, R. S., & Derungs, C. (2015). From Space to Place: Place-Based Explorations of Text. *International Journal of Humanities and Arts Computing*, *9*(1), 74–94. https://doi.org/10.3366/ijhac.2015.0139

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python Natural Language Processing Toolkit for Many Human Languages.

Quirk, R., Greenbaum, S., Leech, G., & Svartvik, J. (1985). *A Comprehensive Grammar of the*

English Language. New York, USA: Longman Publishers.

Radke, M., Stock, K., & Jones, C. B. (2019). Detecting the Geospatialness of Prepositions from Natural Language Text. In S. Timpf, C. Schlieder, M. Kattenbeck, B. Ludwig, & K. Stewart (Eds.), *14th International Conference on Spatial Information Theory (COSIT 2019)* (pp. 1–8). Dagstuhl, Germany: Schloss Dagstuhl: Leibniz-Zentrum fuer Informatik. https://doi.org/10.4230/LIPIcs.COSIT.2019.11

Reppen, R. (2010). Building a corpus. In *The Routledge Handbook of Corpus Linguistics* (pp. 31–37). Routledge. https://doi.org/10.4324/9780203856949.ch3

Ritter, A., Clark, S., Etizioni, M., & Etzioni, O. (2011). Named entity recognition in tweets: An experimental study. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 1524–1534. https://doi.org/10.1075/li.30.1.03nad

Ritter, A., Clark, S., Mausam, & Etzioni, O. (2011). Named Entity Recognition in Tweets : An Experimental Study. In *EMNLP '11 Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1524–1534). Edinburgh, United Kingdom.

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors Takeshi. In *WWW '10 Proceedings of the 19th international conference on World wide web* (pp. 851–860). https://doi.org/10.1145/1772690.1772777

Santillana, M., Nguyen, A. T., Dredze, M., Paul, M. J., Nsoesie, E. O., & Brownstein, J. S. (2015). Combining Search, Social Media, and Traditional Data Sources to Improve Influenza Surveillance. *PLoS Computational Biology*, *11*(10), 1–15. https://doi.org/10.1371/journal.pcbi.1004513

Sarkar, K. (2015). A hidden Markov model based system for entity extraction from social media english text at FIRE 2015. In *CEUR Workshop Proceedings* (Vol. 1587, pp. 89–95).

Schmitt, X., Kubler, S., Robert, J., Papadakis, M., & Letraon, Y. (2019). A Replicable Comparison Study of NER Software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)* (pp. 338–343). IEEE.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. In *IEEE Transactions on Signal Processing* (Vol. 45, pp. 2673–2681). https://doi.org/10.1109/78.650093

Singh, L., Bansal, S., Bode, L., Budak, C., Chi, G., Kawintiranon, K., … Wang, Y. (2020). A first look at COVID-19 information and misinformation sharing on Twitter. Retrieved

from http://arxiv.org/abs/2003.13907

Siriaraya, P., Zhang, Y., Wang, Y., Kawai, Y., Mittal, M., Jeszenszky, P., & Jatowt, A. (2019). Witnessing Crime through Tweets. In *SIGSPATIAL '19* (pp. 568–571). Chicago, Illinois, USA. https://doi.org/10.1145/3347146.3359082

Starbird, Kate & Palen, L. (2011). "Voluntweeters": self-organizing by digital volunteers in times of crisis. In *CHI '11 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1071–1080).

Stock, K. (2018). Mining location from social media: A systematic review. *Computers, Environment and Urban Systems*, *71*(May), 209–240. https://doi.org/10.1016/j.compenvurbsys.2018.05.007

Stock, K., Jones, C. B., & Tenbrink, T. (2019). Speaking of Location: Communicating about Space with Geospatial Natural Language. In K. Stock, C. B. Jones, & T. Tenbrink (Eds.), *Proceedings of the Workshop on Speaking of Location 2019: Communicating about Space co-located with 14th International Conference on Spatial Information Theory (COSIT 2019)* (pp. 1–7). Regensburg, Germany. Retrieved from http://ceur-ws.org/Vol-2455/paper1.pdf

Sui, D., & Goodchild, M. (2011). The convergence of GIS and social media: Challenges for GIScience. *International Journal of Geographical Information Science*, *25*(11), 1737–1748. https://doi.org/10.1080/13658816.2011.604636

Talmy, L. (2000). How Language Structures Space. In *Toward a Cognitive Semantics* (Vol. 1, pp. 177–254). MIT Press.

Van, T. N., Gaio, M., & Moncla, L. (2013). Topographic subtyping of place named entities: a linguistic approach. In *AGILE 2013* (pp. 1–5). Leuven, Belgium.

Vasardani, M., Winter, S., Richter, K.-F., Stirling, L., & Richter, D. (2013). Spatial interpretations of preposition "at," 46. https://doi.org/10.1145/2442952.2442961

Vieweg, S., Hughes, A. L., Starbird, K., & Palen, L. (2010). Microblogging during two natural hazards events. *Proceedings of the 28th International Conference on Human Factors in Computing Systems - CHI '10*, (May 2014), 1079. https://doi.org/10.1145/1753326.1753486

Vilain, P., Menudier, L., & Filleul, L. (2019). Twitter: a complementary tool to monitor seasonal influenza epidemic in France? *Online Journal of Public Health Informatics*, *11*(1), 2017–2020. https://doi.org/10.5210/ojphi.v11i1.9724

Wallgrün, J. O., Karimzadeh, M., MacEachren, A. M., & Pezanowski, S. (2018). GeoCorpora: building a corpus to test and train microblog geoparsers. *International Journal of Geographical Information Science*, *32*(1), 1–29. https://doi.org/10.1080/13658816.2017.1368523

Wang, J., & Hu, Y. (2019). Are We There Yet ? Evaluating State-of-the-Art Neural Network based Geoparsers Using EUPEG as a Benchmarking Platform. In *GeoHumanities '19 Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Geospatial Humanities* (pp. 1–6). Chicago, Illinois. https://doi.org/10.1145/3356991.3365470

Wang, J., Hu, Y., & Joseph, K. (2020). NeuroTPR : A Neuro-net ToPonym Recognition Model for Extracting Locations from Social Media Messages. *Transactions in GIS*, 1–22.

Wolf, S. J., Henrich, A., & Blank, D. (2014). Characterization of Toponym Usages in Texts Categories and Subject Descriptors. In *GIR '14: Proceedings of the 8th Workshop on Geographic Information Retrieval* (pp. 1–8). https://doi.org/10.1145/2675354.2675703

Xu, C., Pei, J., Li, J., Li, C., Luo, X., & Ji, D. (2019). DLocRL: A deep learning pipeline for fine-grained location recognition and linking in tweets. *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019*, 3391–3397. https://doi.org/10.1145/3308558.3313491

Yadav, V., & Bethard, S. (2019). *A Survey on Recent Advances in Named Entity Recognition from Deep Learning models*. Retrieved from http://arxiv.org/abs/1910.11470

Yang-Lim, C., Tan, I. K. T., & Selvaretnam, B. (2019). Domain-General Versus Domain-Specific Named Entity Recognition: A Case Study Using TEXT. In R. Chamchong & K. W. Wong (Eds.), *Lecture Notes in Artificial Intelligence* (pp. 328–246). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-030-33709-4

Yin, J., Karimi, S., & Lingad, J. (2014). Pinpointing Locational Focus in Microblogs. In *ADCS* (pp. 66–72). https://doi.org/10.1145/2682862.2682868

Yuan, Q., Cong, G., Zhao, K., Ma, Z., & Sun, A. (2015). Who, Where, When, and What. *ACM Transactions on Information Systems*, *33*(1), 1–33. https://doi.org/10.1145/2699667

Zhang, C., Fan, C., Yao, W., Hu, X., & Mostafavi, A. (2019). Social media for intelligent public information and warning in disasters: An interdisciplinary review. *International Journal of Information Management*, *49*(April), 190–207. https://doi.org/10.1016/j.ijinfomgt.2019.04.004

Zhang, Y., Dong, X., Zhang, D., & Wang, D. (2019). A Syntax-based Learning Approach to Geo-locating Abnormal Traffic Events using Social Sensing. In *2019 IEEE/ACM*

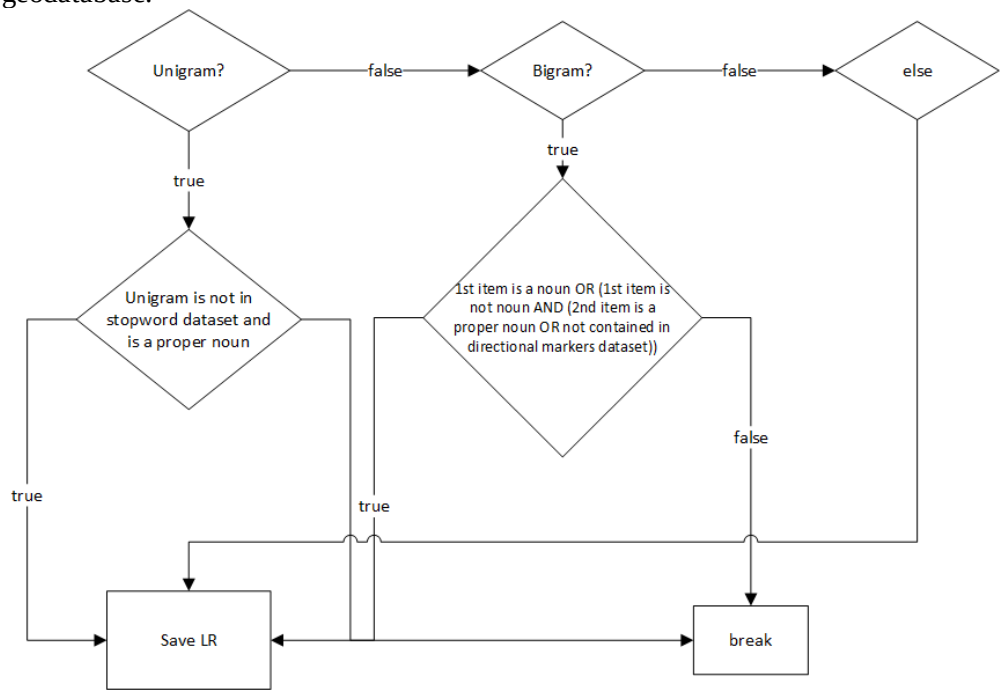
International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 663–670). Vancouver, Canada.

Zheng, X., Han, J., & Sun, A. (2018). A Survey of Location Prediction on Twitter. *IEEE Transactions on Knowledge and Data Engineering*, *4347*(c), 1–20. https://doi.org/10.1109/TKDE.2018.2807840

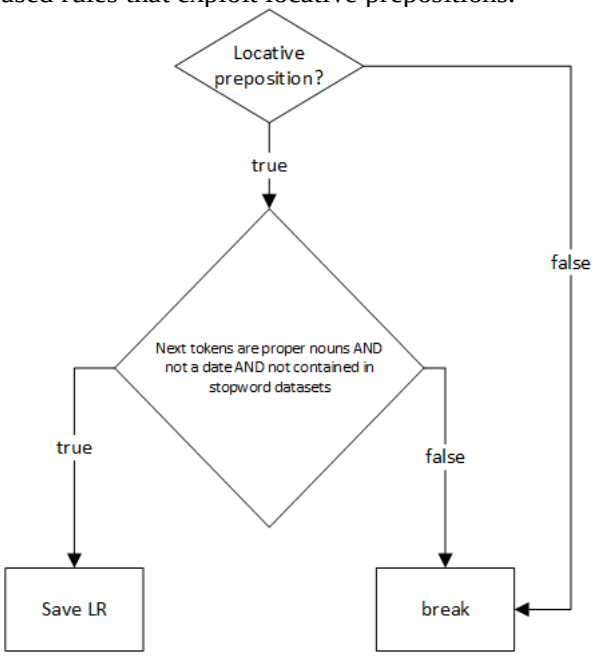
Zou, Z., He, X., & Zhu, A. (2019). An Automatic Annotation Method for Discovering Semantic Information of Geographical Locations from Location-Based Social Networks. *International Journal of Geo-Information*, *8*(487), 1–18. https://doi.org/10.3390/ijgi8110487

APPENDIX

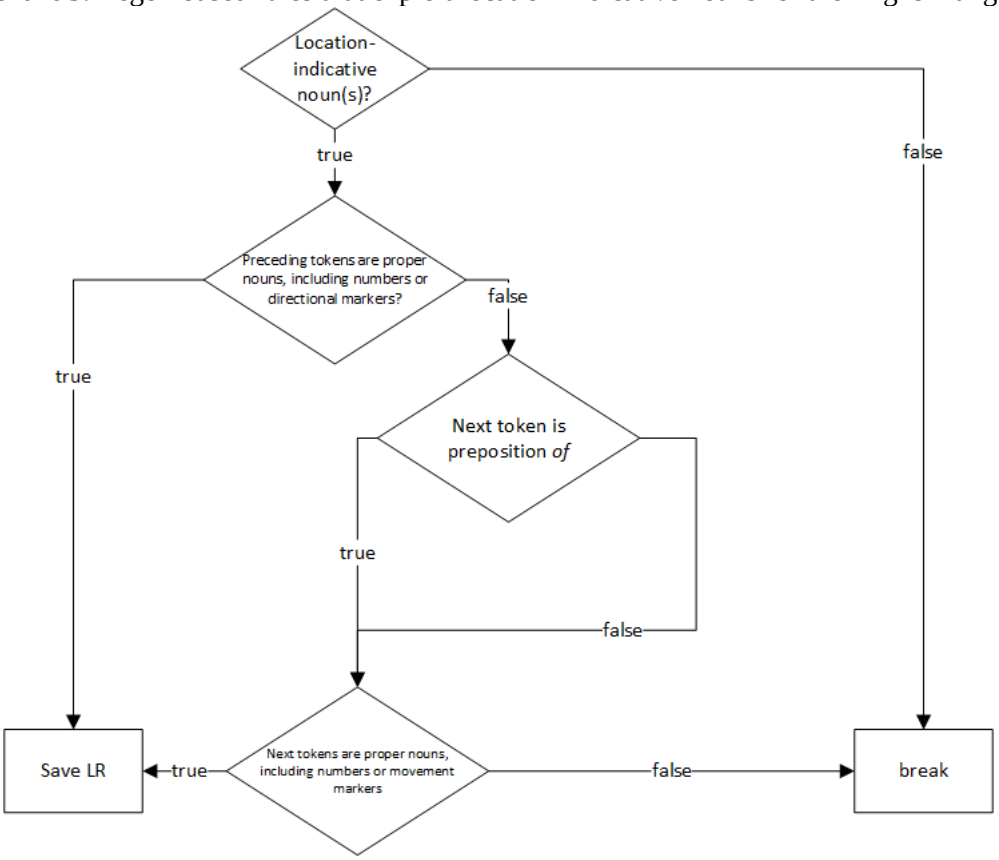
Flowchart 1. Regex-based rules for n-gram combinations of locative references using a geodatabase.



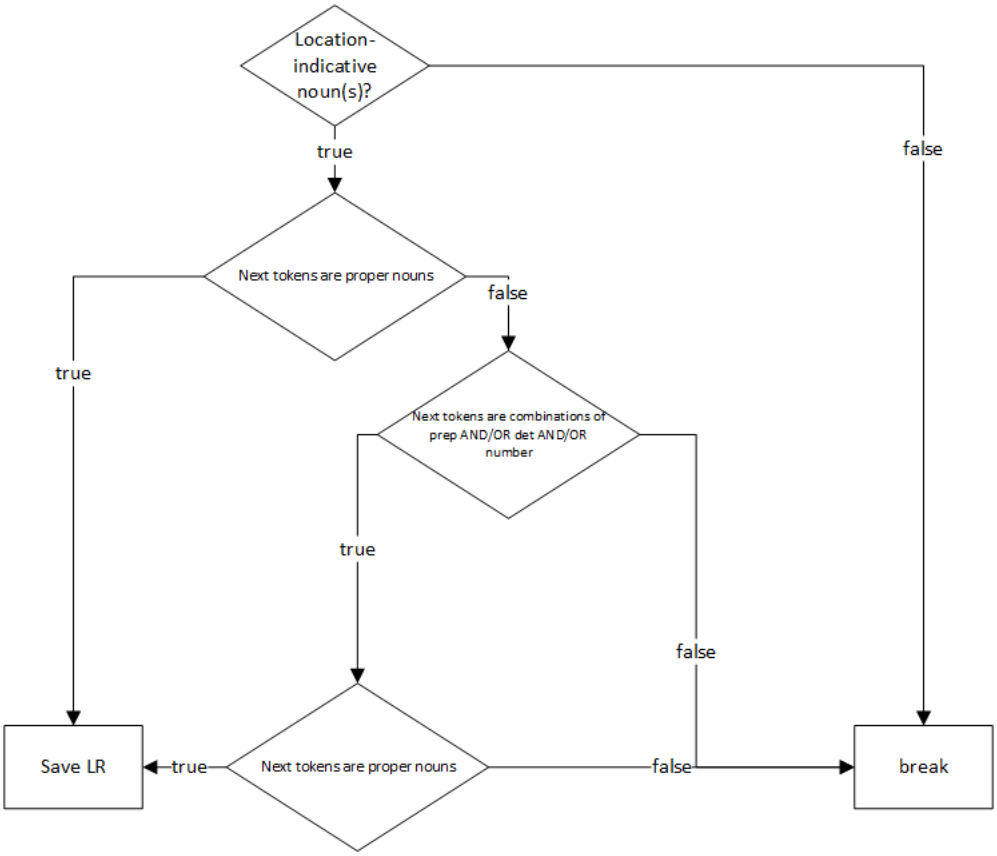
Flowchart 2. Regex-based rules that exploit locative prepositions.



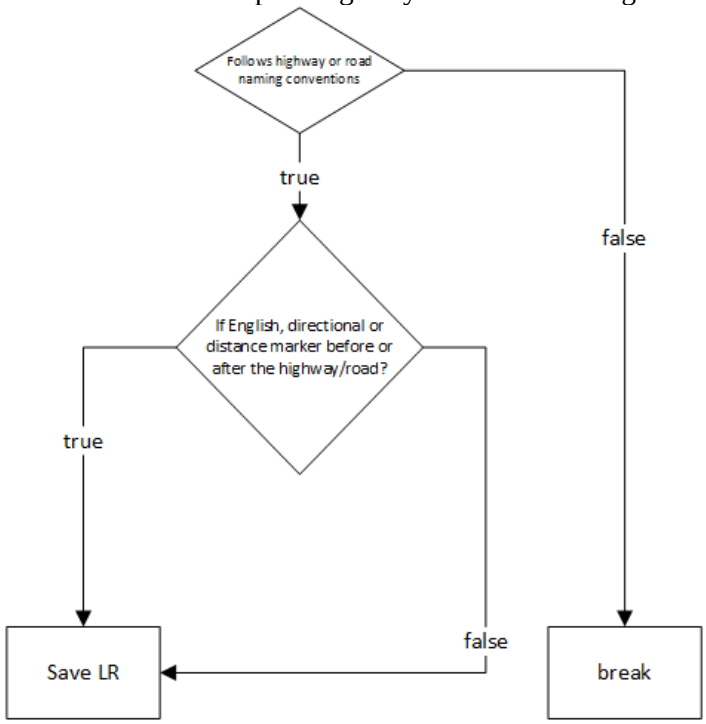
Flowchart 3. Regex-based rules that exploit location-indicative nouns for the English language.



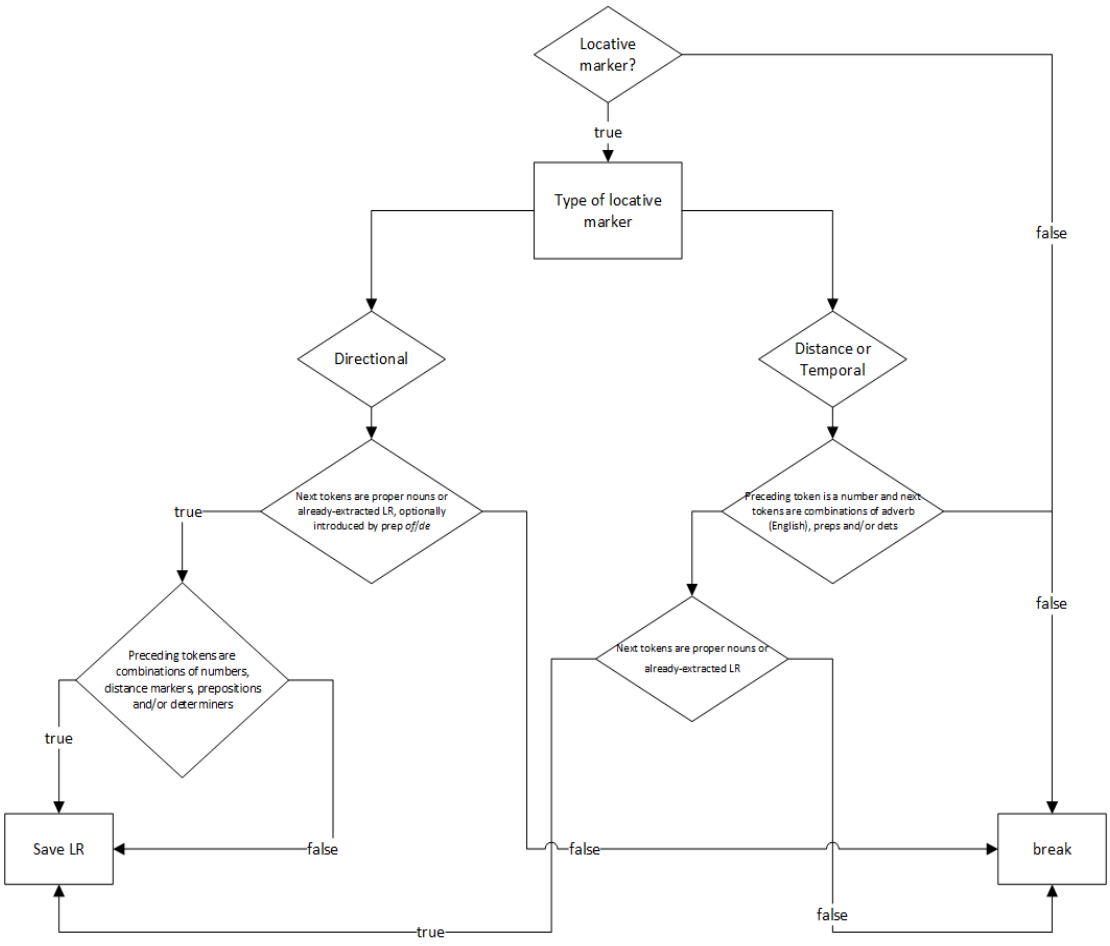
Flowchart 4. Regex-based rules that exploit location-indicative nouns for the Spanish and French languages.



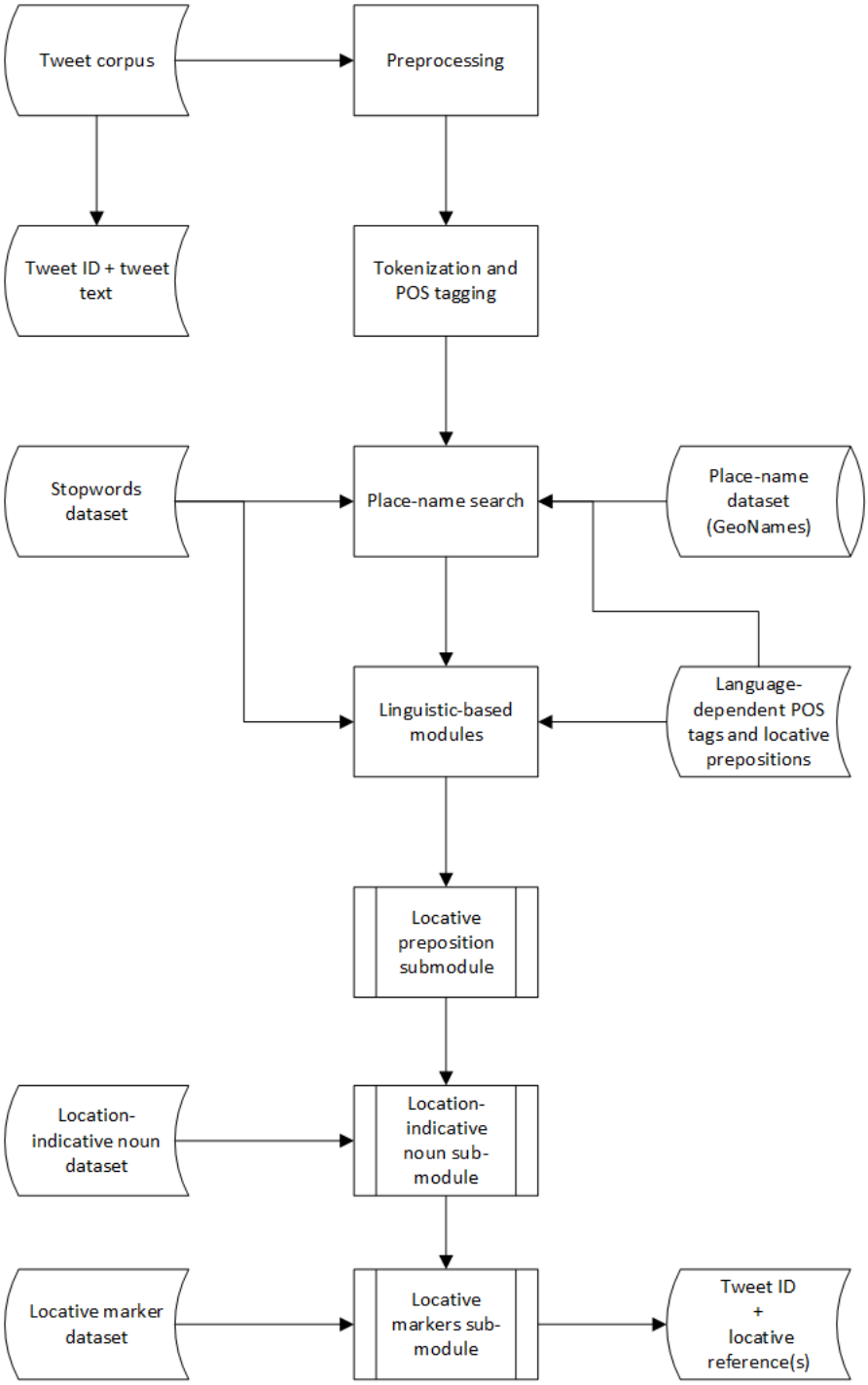
Flowchart 5. Regex-based rules that exploit highways and road-naming conventions.



Flowchart 6. Regex-based rules that exploit locative markers.



Flowchart 7. The pipeline of LORE.


Tweet corpus
Preprocessing
Tweet ID + tweet text
Tokenization and POS tagging
Stopwords dataset
Place-name search
Place-name dataset (GeoNames)
Linguistic-based modules
Language-dependent POS tags and locative prepositions
Locative preposition submodule
Location-indicative noun dataset
Location-indicative noun sub-module
Locative marker dataset
Locative markers sub-module
Tweet ID + locative reference(s)