International Journal of Geo-Information

MDPI

Article

A Novel Deep Learning Approach Using Contextual Embeddings for Toponym Resolution

Ana Bárbara Cardoso [1], Bruno Martins [1] and Jacinto Estima [2,*]

[1] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, 1649-004 Lisboa, Portugal; barbara.inacio@tecnico.ulisboa.pt (A.B.C.); bruno.g.martins@tecnico.ulisboa.pt (B.M.)
[2] INESC-ID, Faculdade de Design, Tecnologia e Comunicação (IADE), Universidade Europeia, 1200-649 Lisboa, Portugal
* Correspondence: jacinto.estima@universidadeeuropeia.pt

Abstract: This article describes a novel approach for toponym resolution with deep neural networks. The proposed approach does not involve matching references in the text against entries in a gazetteer, instead directly predicting geo-spatial coordinates. Multiple inputs are considered in the neural network architecture (e.g., the surrounding words are considered in combination with the toponym to disambiguate), using pre-trained contextual word embeddings (i.e., ELMo or BERT) as well as bi-directional Long Short-Term Memory units, which are both regularly used for modeling textual data. The intermediate representations are then used to predict a probability distribution over possible geo-spatial regions, and finally to predict the coordinates for the input toponym. The proposed model was tested on three datasets used on previous toponym resolution studies, specifically the (i) *War of the Rebellion*, (ii) *Local–Global Lexicon*, and (iii) *SpatialML* corpora. Moreover, we evaluated the effect of using (i) geophysical terrain properties as external information, including information on elevation or terrain development, among others, and (ii) additional data collected from Wikipedia articles, to further help with the training of the model. The obtained results show improvements using the proposed method, when compared to previous approaches, and specifically when BERT embeddings and additional data are involved.

Keywords: geographical text analysis; resolving toponyms in textual documents; deep learning for NLP; contextual word embeddings; machine learning with neural networks

check for updates

Citation: Cardoso, A.B.; Martins, B.; Estima, J. A Novel Deep Learning Approach Using Contextual Embeddings for Toponym Resolution. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 28. https://doi.org/10.3390/ijgi11010028

Academic Editor: Wolfgang Kainz

Received: 12 November 2021
Accepted: 27 December 2021
Published: 31 December 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Toponym resolution concerns the disambiguation of place names, in some cases considering also other references to places such as adjectival and demonymic forms, given in textual documents. Place names are first recognized through a Named Entity Recognition (NER) model, and the disambiguation is then achieved by associating a unique position on the surface of the Earth to each of the place references, e.g., by assigning geographic coordinates. As place references are highly ambiguous, antonym resolution tasks are notably challenging. When resolving toponyms in textual documents, three specific types of ambiguity need to be addressed [1,2]:

1. Geo/geo ambiguity happens when the same place name is shared by different locations. For instance, the name *Kent* can be related to either *Kent County*, *Delaware*, *United States*, or *New Kent Count*, *Virginia*, *United States*;
2. Geo/non-geo ambiguity happens when common language words are used to identify places names, i.e., when the same name is shared by a location and also by a non-location. For instance, the word *Charlotte* can refer to the specific location of *Charlotte County*, *Virginia*, *United States* or to a person name. The word *Manhattan* can also refer to a cocktail beverage, or to the specific location of *Manhattan*, *New York*, *United States*;
3. Reference ambiguity, which arises when the same place can be referred by multiple names. For instance, *Motor City* is a commonly used name to refer to *Detroit*, *Michigan*, *United States*.

Problem (2) should be addressed by the NER model that identifies place references in the textual documents, while problems (1) and (3) should be tackled when attempting to associate, unambiguously, physical locations to the references recognized in the text (e.g., using geo-spatial coordinates in the form of latitude and longitude).

Several applications can use the results from toponym resolution methods. These include the organization and visualization of documents according to spatial criteria, for instance by grouping documents into relevant clusters and/or mapping textually encoded information [2]. Another example relates to improving the presentation of results within search engines, for instance through geographic indexing, ranking, and/or clustering of search results [3–6]. Yet another possible application relates to supporting studies in areas such as the computational social sciences or the digital humanities [7], for instance analyzing and processing geographic data extracted from collections of textual documents. Furthermore, the resolution of place references can be used as an auxiliary component for the geo-location of complete documents [8], given that toponyms found in textual documents can provide clues on the general geographic area under discussion.

Most of the previously developed systems for toponym resolution in text are based on heuristics (e.g., prefer places with high population density, or minimize the geo-spatial distance between the different locations referenced within a single textual unit), also relying on external knowledge bases (i.e., gazetteers) to decide the location that is more suitable to correlate to each reference. The place references encountered in the text are first compared with analogous entries available in a gazetteer [9,10] and, for instance, the matching entries corresponding to highly populated places can then be favored, since they are prone to be used in textual descriptions [11,12]. Methods using multiple heuristics as features integrated into standard supervised machine learning techniques are also considered in some studies [13–16], while newer studies considered the application of language modeling methods [17,18]. More recently, the use of deep learning techniques yielded state-of-the-art results for the toponym resolution task [19–22].

This article introduces a novel method for toponym resolution that uses deep learning to model the textual elements, by combining bidirectional Long Short-Term Memory (LSTM) units with pre-trained contextual word embeddings (i.e., static features extracted using either the Embeddings from Language Models (ELMo) [23] or the Bidirectional Encoder Representations from Transformers (BERT) [24] methods). We focused on the disambiguation of previously recognized place references, noting that there are many NER approaches that can easily be employed to accurately identify place names in text (e.g., software packages such as spacy.io feature robust models for entity recognition).

The model that we are proposing combines multiple textual inputs, more specifically the place name reference, the sentence where the reference occurs, and the corresponding paragraph. The model also considers multiple outputs that are connected together, particularly a primary output corresponding to geographic coordinates, together with an alternate classification output related to coarse regions on the earth's surface (i.e., the model combines a multi-class classification objective, associated with predicting regions, together with a regression objective associated with predicting latitude and longitude coordinates). The classification regions are established using a Hierarchical Equal Area isoLatitude Pixelisation (HEALPix) [25] method that generates equal area cells, obtained by partitioning a spherical surface representing the earth's surface recursively. The network output corresponding to geo-spatial coordinates, which derives its values from the probability distribution associated with the classification output, is connected to a loss function corresponding to the great circle distance. This value is combined with a cross-entropy loss for the multi-class classification output, jointly optimizing both parts to hopefully improve the prediction of the geographic coordinates for each place reference. Additionally, we experimented with the use of information corresponding to geophysical properties (i.e., the elevation, the development of the terrain, the minimum distance from water zones, and the percentage of vegetation), in an attempt to further improve performance. This additional information is derived from external datasets and integrated in the model, as additional

regression outputs, to guide the prediction of the geographic coordinates with basis on the descriptions given in the surrounding context.

The proposed model was tested using three well-known datasets extensively used in previous studies, specifically the *Local–Global Lexicon*, the *War of the Rebellion*, and the *SpatialML* corpora. The source code supporting the experiments was also made publicly available on a Github repository (http://github.com/barbarainacioc/toponym-resolution (accessed on 10 November 2021)). In sum, our experimental results show that the proposed approach can surpass the results obtained in previous studies over the same datasets. Moreover, we tested different variants of the proposed approach (e.g., comparing ELMo and BERT embeddings) and, to understand the impact of the size of training data on the results, we selected also a scenario in which a larger data sample is used for model training. The new instances added to the original corpora were extracted from a random sample of English Wikipedia articles, taking advantage of the Wikipedia link structure to collect spans of text corresponding to place references (i.e., considering spans of text linking to Wikipedia pages associated with geo-spatial coordinates). The results show that increasing the training data only marginally improved results, although using BERT embeddings lead to significantly lower errors.

The remaining of the article is organized as follows: Section 2 describes related work, and presents the corpora used in this project. Section 3 details the proposed model. Section 4 depicts the followed experimental evaluation, including the evaluation methodology as well as the obtained results. Finally, Section 5 exposes our conclusions and draws ideas for future work.

2. Related Work

This section presents relevant previous studies addressing toponym resolution with different techniques, more specifically methods based on heuristics (Section 2.1), methods combining heuristics with supervised learning (Section 2.2), methods using geodesic grids together with language models (Section 2.3), and finally methods leveraging deep learning (Section 2.4). Lastly, Section 2.5 describes the corpora that was used in the experiments described in this work.

2.1. Heuristic Methods for Toponym Resolution

The majority of the systems previously developed for toponym resolution are based on heuristics, using external knowledge sources to access a collection of data about locations on the surface of the Earth (e.g., place types, alternative names, geo-spatial footprints, or population density, among others). Systems based on heuristics usually employ the aforementioned types of data to decide which of the possible locations, within a set of candidates retrieved from a gazetteer, corresponds to the place name provided in the text [11,12,14,26,27].

Besides external information, it also possible to consider linguistic aspects in the formation of toponym disambiguation heuristics [12,27]. In seminal work in the area, Leidner [12] used both linguistic heuristics (i.e., inferring rules and patterns from the textual content) and extra-linguistic heuristics (i.e., using an external source of knowledge). For instance, one of the heuristics considered by Leidner is based on a *contained-in* qualifier, that recognizes patterns such as $toponym_1$ *(*$toponym_2$*)* or $toponym_1$ *in* $toponym_2$, and evaluates the geo-spatial containment for the possible candidate locations (i.e., locations with the same name as the one being resolved) regarding both toponym mentions, defining the geographic coordinates based on the geo-spatial containment (e.g., assigning to *London* the coordinates of England's capital in case the recognized pattern relates to *London (UK)*, or the coordinates of the city in Ontario if the recognized pattern refers to *London*, *Ontario*, *Canada*). Leidner also used extra-linguistic heuristics such as the one based on assigning, to the toponym mention being disambiguated, the candidate location with a higher population density. Another example is a one sense per discourse heuristic that considers that if a given toponym occurs multiple times in the text without additional contextual clues, and if one candidate location is highly prominent (e.g., it corresponds to a capital city), then there

is a higher probability that the other same toponym mentions refer to that candidate (e.g., if *Paris* occurs in the text, always consider the coordinates of Paris, the capital of France, and do not assign other locations named Paris).

Leidner [12] also reported on hybrid combinations of both types of heuristics, e.g., considering textual-spatial correlation and assuming that textual similarity in the occurrence contexts should be strongly correlated with spatial proximity. For example, if the mentions *Paris* and *Versailles* are present within a small text span (i.e., in close textual proximity), then the mention *Paris* is likely to be associated with Paris, the capital city of France.

A recent heuristic method developed in the context of the GeoTxt geocoder (i.e., a flexible application programming interface used for extracting and disambiguating toponyms in small textual documents), was reported by Karimzadeh et al. [14]. This system uses existing resources to recognize toponyms in the text, with an exclusive focus on disambiguating the recognized place references. During toponym resolution, for each mention, the system uses a custom index over the GeoNames gazetteer to retrieve a set of candidate locations, assigning to each of them a specific score. This score results from a combination of multiple features, which include boosts for population density or particular place type indicators (e.g., administrative divisions, regions, continents, independent political entities, populated places, or establishment types (e.g., stadium, train station, college), among others). Moreover, GeoTxt also considers additional disambiguation heuristics based on the co-occurrence of toponyms in the text, leveraging the idea of *spatial minimality* previously also reported by Leidner [12]. Two of such heuristics are related to hierarchical relationships between toponyms (i.e., if a containment relationship towards the same geographic space is shared by two toponyms, either connected to immediately consecutive place names (e.g., a combination of state, city) or related to toponyms separately showed in the text). A third heuristic is related to spatial proximity, and aims to minimize the average distance between the predicted location of a toponym and the location of other toponyms co-occurring in the text.

2.2. Combining Heuristics through Supervised Learning

Several previous studies have used supervised approaches that consider heuristics, such as those referenced in the previous section, as features within standard machine learning techniques [13,15,16].

For instance, taking inspiration on previous general entity linking studies, Santos et al. [16] explored the combination of multiple features, such as place prominence, similarities among potential candidate locations and the context related to the place reference (e.g., comparing descriptions for the candidates against the text surrounding the mention), and similarities towards candidates to other toponyms present in the text. A learning to rank model is trained from a set of instances associated with the correct disambiguation and, when processing place references in previously unseen texts, the model is used to assign a rank to each candidate location, according to the likelihood of it corresponding to the correct disambiguation. The location with the highest rank is finally associated with the place mention being resolved.

Methods such as those from Lieberman and Samet [15] or Santos et al. [16] can naturally incorporate many different types of features, avoiding the need for manual parameter tuning. Features can also be combined with representation learning methods based on deep learning (e.g., Canwen et al. have recently described the use of LSTM networks to represent location mentions together with their left and right contexts, combining these representation with other features within an entity linking method for associating location mentions in tweets to entries within a points-of-interest database [28]). However, as in the case of the studies surveyed in the previous section, these methods depend on the availability of external resources such as gazetteers [9,10]. One of the downsides of using gazetteers is that these resources are usually incomplete and outdated, thus having a direct impact on the results of systems that use them, and causing them to be unable to deal with new and vernacular place names.

2.3. Methods Combining Geodesic Grids and Language Models

Some previous studies have proposed toponym resolution methods that avoid the need of external information in the form of gazetteers, instead using language models associated with different regions in order to predict which region is more likely to correspond to the place reference under analysis [7,17,18]. Most of these methods leverage geodesic grids to partition the geographic space into multiple regions, and they assume that even common language words (and not just place names) can often be geographically indicative.

A set of textual utterances, for which the corresponding geo-spatial coordinates are known, is associated with each of the regions in a geodesic grid. These textual utterances are used for training a language model for each of the regions (e.g., a generative n-gram language model, or instead a discriminative model based on logistic regression or some other machine learning method). The resolution of the grid is usually also adapted according to the number of available textual utterances, and multi-resolution grids can also be used. Given a place reference and its surrounding context, a prediction can be made by choosing the region whose language model better matches the text of the reference plus the context, and then taking the region's centroid coordinates.

In a seminal study following the aforementioned methodology, Wing and Baldridge [29] explored the use of geodesic grids for geo-locating entire textual documents. The authors discretized the Earth's surface into 1×1 degree grid cells, assigning Kullback–Liebler divergences to each cell given a document, based on uni-gram language models learned for each cell from geo-located Wikipedia articles. Speriosu and Baldridge [18] further refined this idea, proposing a text-driven toponym resolution method that uses, as input to the language model, context windows composed of twenty words from each side of each toponym occurrence in the text. Continuing this line of research, Wing [7] reported on experiments using multi-resolution geodesic grids (i.e., based on a k-d tree partitioning), together with discriminative logistic regression models instead of generative uni-gram language models, for both document geo-coding and toponym resolution.

More recently, DeLozier et al. [17,30] described the *TopoCluster* system, which uses language models in a slightly different way. Specifically, the authors used spatial statistics [31] to derive for each word in the vocabulary a smoothed geographic likelihood. Toponym resolution was then made by finding the points of strongest overlap for a toponym and the words surrounding it. In more detail, given a reference collection of geo-located documents, an unsmoothed local language model is used to measure the association of each word to each document. The method can directly use the locations of the reference documents or, optionally, these locations can be aggregated into geodesic grid cells. The Local Getis–Ord Gi* statistic [31], together with an Epanichnikov kernel that penalizes large distances between pairs of locations, is used to measure the strength of the association between words and geographic regions, resulting in a matrix of statistics with grid cells as columns and each word as a row vector (i.e., the Gi* statistic can be seen as the geographically aggregated and smoothed likelihood of seeing each word at specific points in the geographic space). To disambiguate a toponym, the authors separate the toponym words from non-toponym words in a surrounding context window (i.e., 15 words on each side, filtering out function words), and finally compute a weighted sum of all the Gi* values for all the words. The centroid from the cell with the highest value can be returned as the disambiguation for the input toponym.

The aforementioned studies showed that text-driven approaches can obtain superior results without recurring to gazetteers, reporting on good results over corpora consisting of international news articles and historical texts.

2.4. Deep Learning Techniques for Toponym Resolution

Recent studies have advanced deep learning methods for toponym resolution, either formulating the task as a particular entity linking problem [28], or extending methods based on geodesic grids, such as those referenced in Section 2.3, in the direction of replacing simpler language models with approaches based on deep neural networks [19,20,22].

For instance, Adams and McKenzie [19] described a character-level approach based on convolutional neural networks for geo-coding multilingual text. The model receives as input a sequence of UTF-8 characters, each encoded as a one-hot vector, and a series of temporal convolution and max-pooling operations are applied to it. These operations result in a vector representation for the input text, to which multiple transformations are then applied. Finally, the output layer predicts a region classification, e.g., based on a coarse geodesic grid. The use of character-level convolutional neural networks allows the authors avoid the need for language-specific tokenization rules. Still, experiments showed that the character-based model did not always achieve the best results (i.e., better results could often be achieved by SVM models using *n*-gram features). The authors concluded that individual words can sometimes be good geographical indicators.

In another recent study, Gritta et al. [20] presented the *CamCoder* system, that tries to disambiguate place references by detecting lexical clues resorting the context words surrounding the mention. A sparse vector representation is proposed by the authors, titled *MapVec*, that encodes a distribution for prior geographic probabilities associated with locations (i.e., based on location coordinates and population counts). Specifically, external geo-spatial data is projected onto a geodesic grid with a resolution of 1×1 degree (i.e., for each location mentioned in a context window, and for each of its ambiguous candidates in a gazetteer, the authors use population counts to estimate a prior probability, adding it to the corresponding grid cell), that is next reshaped into a 1D feature vector (i.e., the *MapVec*). The *CamCoder* system combines lexical and geographic information, corresponding to the following inputs: the entity targeted to disambiguate, any mentions to locations (excluding the context words), any context words (excluding mentions to the location), and the *MapVec* feature vector. The textual inputs (i.e., the first three inputs from the previous enumeration) are fed into separate layers that combine convolutions with global max-pooling operations, to detect words indicative of particular locations. The *MapVec* feature vector is, in turn, provided to a fully-connected layer. Afterwards, the four resulting vectors are passed individually into another dense layer, followed by a concatenation of their results. This representation is finally delivered to an output layer, with which the model predicts a location based on the classification into regions defined by a geodesic grid. The authors reported on an extensive set of tests with multiple corpora, showing that the complete approach leveraging CNNs+*MapVec* achieved state-of-the-art results at the time. The *MapVec* features remained effective when used within other machine learning approaches (e.g., Random Forests), and they also improved performance when combined with models based on recurrent neural networks.

2.5. Corpora from Previous Studies Employed in Our Work

This section describes the previously proposed datasets that were also used for the evaluation of our method, specifically the *War of the Rebellion* [30], the *Local–Global Lexicon* [32], and the *SpatialML* [33] corpora. These three distinct datasets have been largely used in the context of previous studies in the area [11,16,17,20,30,34], covering distinct domains.

Lieberman et al. [32] presented the *Local–Global Lexicon* (LGL) corpus, composed of 588 articles extracted from newspapers that are small and geographically distributed. This corpus was deliberately built to challenge toponym resolution systems, since it is composed of local news articles, particularly from small cities with ambiguous names. For instance, *Paris* is a highly ambiguous toponym and this specific dataset contains articles extracted from local newspapers such as The Paris Post-Intelligencer (*Paris*, *Tennessee*), The Paris News (*Paris, Texas*), and The Paris Beacon-News (*Paris*, *Illinois*). The corpus is now available (http:

//raw.githubusercontent.com/geoai-lab/EUPEG/master/corpora/lgl.xml (accessed on 10 November 2021)) within the EUPEG [35,36] benchmark platform for toponym resolution.

In turn, the *War of the Rebellion* (WOTR) corpus (http://github.com/utcompling/WarOfTheRebellion (accessed on 10 November 2021)) contains 1644 historical texts extracted from military archives connected to the American Civil war, where reports, military orders, and government correspondence are predominant. The annotation process related to these historical documents is described by DeLozier et al. [30], who also presented an evaluation of other existing toponym resolution systems on the corpus, assessing additionally the same methods over other corpora to compare the results. The authors concluded that the WOTR corresponds to the most challenging corpus that was surveyed, e.g., with systems achieving lower overall performance results than with the *Local–Global Lexicon* corpus, which was considered the most challenging one until then.

Finally, the *SpatialML* corpus (http://catalog.ldc.upenn.edu/LDC2011T02 (accessed on 10 November 2021)) is available from the Linguistic Data Consortium, composed of 428 English documents from the ACE 2005 evaluation campaign, including magazine news, broadcast news, broadcast conversations, web blog entries, and newsgroup posts. Besides the corpus, SpatialML also refers to the XML-based scheme that was employed in the annotation of the data, where location references appearing in the text are combined with a tag `PLACE` together with an attribute `LATLONG` encompassing the geographical coordinates of latitude and longitude.

The majority of the methods surveyed in the previous sections used at least one of these three datasets for evaluation. Section 4 of the present article provides a statistical characterization for the different datasets (e.g., see Table 1), also contrasting our method against previously reported results.

Table 1. Statistical characterization for the different corpora used in the experiments.

Statistic	WOTR	LGL	SpatialML
Number of documents	1644	588	428
Number of toponyms	10,377	4462	4606
Average number of toponyms per document	6.3	7.6	10.8
Average number of word tokens per document	246	325	497
Average number of sentences per document	12.7	16.1	30.7
Vocabulary size	13,386	16,518	14,489
Number of HEALPix classes/regions	999	761	461

3. The Proposed Toponym Resolution Method

The following sections give a detailed description of the techniques employed in the proposed method. Section 3.1 overviews recurrent neural networks, namely the Long Short-Term Memory (LSTM) units used in our model, and also supporting ELMo word embeddings. Section 3.2 consider specific approaches for representing text through word embeddings, namely ELMo and BERT. Lastly, Section 3.3 describes our toponym resolution approach, which models the problem as a prediction task that is addressed by a deep neural network, whose architecture is explained in detail.

3.1. Recurrent Neural Networks

A Recurrent Neural Network (RNN) supports the modeling of sequences, thus naturally applying to Natural Language Processing (NLP) tasks that involve sequences of characters or words [37].

The representation of input sequences with discretionary length (e.g., sequences of vectors encoding textual utterances) is allowed by a RNN, for instance transforming them into a fixed-size vector representation that maintains the properties of the input sequence. Generally, an RNN can be recursively defined through a function $\mathrm{R}(\cdot)$ that accepts a state vector s_{j-1} as input (i.e., a vector corresponding to the previous state), together with an input vector for the current state x_j, returning a new state vector s_j. The state vector s_j is

next mapped by a function $\mathrm{O}(\cdot)$ into a vector that resembles to the output vector of the present state y_j. The aforementioned structure contemplates the set of the history related to all previous states $(x_1, x_2, \ldots, x_j)$. After processing the last input vector, an encompassing representation for the entire input sequence can be obtained from the final output vector, or by a pooling operation (e.g., max-pooling) applied to the sequence of output vectors.

A bi-directional RNN can also be defined, following the ideas mentioned before but in this case connecting (e.g., concatenating the output vectors) two RNN units processing the inputs in opposite directions (i.e., from left-to-right and from right-to-left). Therefore, the output of each position can encode information combining together past (backward) and future (forward) states. Multiple RNN or bi-directional RNN units can also be stacked together in deep neural networks, using the outputs produced by one RNN as the inputs to another subsequent RNN layer.

The aforementioned general ideas can be implemented in practice through different architectures, in many cases relying on gating mechanisms that facilitate supervised learning by regulating gradient updates. Long Short-Term Memory (LSTM) units are perhaps the most common example of a concrete RNN architecture [38]. In this case, when processing each input vector, a gating mechanism decides how much of the new input a memory cell should receive, and how much of the current memory cell content the memory cell should forget. Following the notation of Goldberg [37], LSTM units can be formally defined as shown in Equation (1).

$$s_j = \mathrm{R}_{\mathrm{LSTM}}(s_{j-1}, x_j) = [c_j; h_j] \tag{1a}$$

where,

$$c_j = f \odot c_{j-1} + i \odot z \tag{1b}$$

$$h_j = o \odot \tanh(c_j) \tag{1c}$$

$$i = \sigma(x_j \cdot W^{xi} + h_{j-1} \cdot W^{hi}) \tag{1d}$$

$$f = \sigma(x_j \cdot W^{xf} + h_{j-1} \cdot W^{hf}) \tag{1e}$$

$$o = \sigma(x_j \cdot W^{xo} + h_{j-1} \cdot W^{ho}) \tag{1f}$$

$$z = \tanh(x_j \cdot W^{xz} + h_{j-1} \cdot W^{hz}) \tag{1g}$$

$$y_j = \mathrm{O}_{\mathrm{LSTM}}(s_j) = h_j \tag{1h}$$

When considering each input position j, the state s_j corresponds to the concatenation of two vectors c_j and h_j, respectively a memory component and a hidden state component (Equation (1a)). Three gating components are responsible for regulating the information flow, particularly the input, the forget, and the output gates, respectively, represented by the variables i, f, and o (Equation (1d–f), where the different W parameters correspond to learnable weight matrices). The gate values are collected from linear combinations of the current input x_j and the previous state h_{j-1}, and passed through a sigmoid activation function. A linear combination of x_j and h_{j-1} determines an update candidate z, to which a hyperbolic tangent activation function (Equation (1g)) is applied. Next, the memory component c_j is updated considering the forget gate, that controls the amount of the previous memory that should be preserved, and the input gate, that controls how much of the proposed update should be kept (Equation (1b)). Finally, the value of h_j, concerning the output y_j (Equation (1h)), is calculated by considering the memory content c_j, passed through a hyperbolic tangent function, and controlled by the output gate (Equation (1c), where the symbol $\odot$ denotes an element-wise product).

LSTMs are nowadays a core building block of many different models for NLP, and the reader is referred to the tutorial from Goldberg [37] for a more detailed explanation.

3.2. Representing Text through Contextual Word Embeddings

Representing words and longer pieces of text is an essential aspect when applying machine learning to NLP tasks. A common approach involves the use of word embedding

methods to represent textual elements, in a way that captures linguistic/semantic information. The reader is referred to Smith [39] for an introduction to these methods, and to Liu et al. [40] or Qiu et al. [41] for more in-depth explanations.

Most word embedding approaches (e.g., the algorithms within the popular word2vec package [42]) use simple neural networks to map words into dense real number vectors, where each vocabulary word is represented by a vector and words that appear in similar contexts are likely to have similar vectors (i.e., the distance between the word embeddings is related to their semantic similarity). Contextual word embeddings go beyond the aforementioned idea of mapping vocabulary entries to dense vectors, considering representations that depend on the surrounding context (i.e., dynamic word representations, in opposition to static mappings of words to vectors), and thus can better handle polysemic words. In our study, we used two of these methods, specifically Embeddings from Language Models (ELMo) and Bidirectional Encoder Representations from Transformers (BERT).

In brief, Peters et al. [23] presented Embeddings from Language Models (ELMo) as an approach for generating pre-trained contextual word embeddings. To contextualize the word representations, ELMo examines entire sentences using a neural language model (i.e., a model capable of assigning probability distributions to word sequences) that is trained to predict the most likely next word given a sequence of words. The language model used by ELMo starts by representing words based on the characters that compose them (i.e., through a simple convolutional neural network), and it then relies on a multi-layer stack of bi-directional LSTMs, as previously described in Section 3.1, to produce the contextual word representations. Thus, when generating word embeddings, ELMo uses both the previous and the following words as contextual information. To obtain a representation for each word, ELMo computes the hidden state of each bi-directional LSTM layer in the stack, and it then computes a weighted sum of these vectors.

Taking inspiration on ELMo, Devlin et al. [24] proposed Bidirectional Encoder Representations from Transformers (BERT), using a language model based on the Transformer neural architecture [43] instead of relying on LSTMs (i.e., modeling text through Transformer encoders, which use attention layers rather than sequential recurrence). In opposition to directional RNN models as explained in Section 3.1, which read the text input sequentially, Transformer encoders read the entire sequence of words at once. Therefore, these models can be considered bidirectional, though it would be more accurate to say that they are non-directional.

The fundamental building block of the Transformer model used in BERT consists of two sub-layers, namely a self-attention layer and a dense feed-forward layer. These blocks are stacked together in a deep architecture. The model receives as input a sequence of word pieces (i.e., BERT considers a vocabulary consisting of some complete words together with sub-word pieces, that can be used in the representation of rare words), and each of them is represented as a vector, denoting the word piece together with its position in the sequence. Each encoder layer applies a self-attention mechanism over the input (i.e., enabling the encoder to consider other words available in the input sequence, when encoding a specific word), processes the result through a feed-forward layer, and passes the output to the next encoder layer. The model is trained through a masked language modeling objective, in which a mask is applied to 15% of the input tokens, after which the output of the masked words position is used to predict the corresponding words. This training objective is complemented with a next sentence prediction task, where the model is given a pair of sentences and is trained to identify when the second one follows the first. This second task is meant to capture more long-term or pragmatic information.

In practice, ELMo and BERT are slightly more elaborate, and the reader is referred to the original publications for a more in-depth explanation [23,24]. In our experiments, we used publicly available pre-trained English models, namely (i) the orginal ELMo model considering LSTM hidden states with a dimensionality of 4096 and an output size of 512, and (ii) the cased version of the original BERT_{base}.

3.3. The Neural Architecture for Toponym Resolution

Our toponym recognition model builds a representation from the toponym that is to be disambiguated (i.e., it processes each individual place reference previously recognized in a text document) together with its surrounding context, using it to make a region classification based on a geodesic grid. The probability distribution from the multi-class categorization output is then used to obtain geographical coordinates (i.e., latitude and longitude) for the toponym.

Three different sequences of words are provided as input to the model, specifically (i) the place mention itself, (ii) the set of words surrounding the mention (i.e., a fixed window, to the left and right sides of the span of text, comprising 50 word tokens and that often suffices to capture sentences and the most relevant context), and (iii) a paragraph text, defined also by a fixed window of 512 tokens (i.e., the maximum sequence size considered in BERT). Both the sentence and the paragraph inputs consider the context surrounding the mention in the right and left directions. Each of the three inputs is first converted into a sequence of vectors through a contextual word embedding mechanism (i.e., using ELMo or BERT), and these vectors are then further processed by a bi-directional LSTM. Notice that the components that generate ELMo or BERT embeddings are pre-trained on very large datasets and kept fixed during the training of our neural model, whereas the bi-directional LSTM units that process these representations are trained by us from a random initialization.

We are taking into account the general and the closest context around the entity by, respectively, delivering to the neural network the paragraph of the mention (i.e., providing the general context of the document), and a smaller textual window containing the mention (i.e., a span of text that approximately corresponds to a sentence). In both cases, the context might provide clues about the location of the mention, e.g., through other toponyms. Even common language words available in the surrounding text might portray characteristics of specific geographic regions.

The geodesic grid used to support the multi-class categorization objective is built through the Hierarchical Equal Area isoLatitude Pixelization (HEALPix (http://healpix.sourceforge.io (accessed on 10 November 2021))) scheme, proposed by Górski et al. [25] and used in previous studies related to document geo-coding [44]. In brief, the HEALPix algorithm partitions a spherical representation of the Earth's surface, generating cells of equal area related to distinct regions. The partitions are collected hierarchically from recursive divisions, and the number of recursive divisions to execute (i.e., the desired resolution) can be defined by the user. The partitioning scheme is demonstrated in Figure 1, which shows the grid resulting from divisions related to multiple resolution parameters, with differences in the number of cells that are generated. The number of generated regions (i.e., cells in the geodesic grid) is defined according to Equation (2), where the N_{side} is related to the aimed resolution.

$$N_{pix} = 12 \times N_{side}^2 \quad (2)$$

In the context of this work, the resolution parameter was fixed to $N_{side} = 256$, which corresponds to considering a maximum of 786,432 regions (N_{pix}). This resolution was selected in a way that the region size is sufficiently large to accommodate enough examples of toponyms inside some regions. In practice, given that most regions might not be associated with any instance in the training set, the number of classes will be much smaller.

An overview on the proposed method is given on Figure 2. The three textual inputs are first represented through either ELMo or BERT contextual embeddings (i.e., we tested each of these alternatives on separate experiments), and the sequences of vectors are processed by custom bi-directional LSTM units, with a dimensionality of 512 in their hidden state vectors (i.e., 1024 values, given the use of bi-directional LSTMs) and which use penalized hyperbolic tangent activation functions, instead of logistic sigmoid or regular hyperbolic tangent functions (i.e., see Equation (3)). The use of the penalized hyperbolic tangent, as

shown in Equation (3), was first suggested by Eger et al. [45], which observed improved results throughout a diversity of natural language processing tasks.

$$f(x) = \begin{cases} \tanh(x), & \text{if } x > 0 \\ 0.25 \times \tanh(x), & \text{otherwise} \end{cases} \quad (3)$$

The sequences of states produced by each bi-directional LSTM are summarized through a max-pooling operation, and the resulting three vectors are then concatenated to form an encompassing representation for the inputs. This representation is then processed by a fully-connected layer, which predicts HEALPix regions (i.e., it produces a probability distribution over the possible HEALPix regions) through a soft-max activation function. This HEALPix class probability vector is one of the outputs of the model (i.e., a categorical cross-entropy loss function is computed over this result), and simultaneously it is also used to estimate the corresponding geographic coordinates. Specifically, the probability values are raised to the third power and re-normalized (i.e., a more peaked distribution is built from the soft-max results, emphasizing the most likely region), and the results are then used as weights by an interpolation scheme that considers a centroid coordinates matrix (i.e., a fixed matrix that contains the coordinates of the centroid of each HEALPix class, where each row corresponds to a distinct class). In practical terms, we compute the product between the re-adjusted probability vector and the centroid coordinates matrix to estimate the coordinates, and the obtained result is then connected to a second loss function that computes the great circle distance between the predicted and the ground-truth coordinates. Model training thus involves minimizing the combined loss functions associated with each of the outputs, each mutually guiding the learning process and hopefully contributing to better overall results.

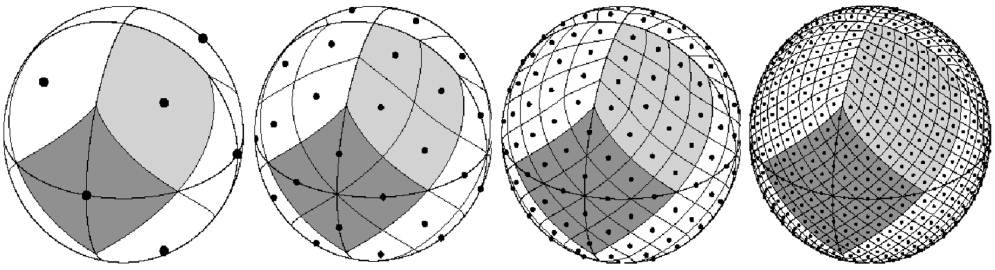


Figure 1. Orthographic views for the results produced by the HEALPix partitioning scheme. The illustration is adapted from a representation originally available from the HEALPix website.

Besides the HEALPix regions and the geo-spatial coordinates, we tried to estimate geophysical terrain properties associated with the predicted HEALPix regions, hoping to further guide the model towards correct location predictions. Similarly to what was made for predicting geo-spatial coordinates, we used the adjusted class probability values as interpolation weights, together with a matrix encoding the geophysical terrain properties at the centroid coordinates of each HEALPix cell. A set of column vectors was created, one for each property, with numerical values for each of the different HEALPix regions. The predicted geophysical properties were then compared against the ground-truth property values associated with the true location of the toponym, using additional loss functions corresponding to the absolute difference.

We used the Adam optimization algorithm to train the model through back-propagation, with a cyclical learning rate policy that adjusts the learning rate along the training [46], based on a cycle between a lower bound of 0.00001 and an upper bound of 0.0001. Given the fact that the categorical cross-entropy, the great circle distance, and the absolute error loss functions produce values in different ranges, we weighted the contribution of each function in the combined loss (i.e., a weight of 100 to one was given to the categorical cross-entropy in relation to the other values, through an initial set of tests that assessed

the impact of this parameter, and which also confirmed better results when combining the cross-entropy and the great circle distance loss functions). An early stopping strategy (i.e., a way of regularization used to overcome over-fitting, where the training process is stopped once the model performance is not improving) was also applied, forcing the training to stop when the combined loss on the training data was not improving for five consecutive epochs.

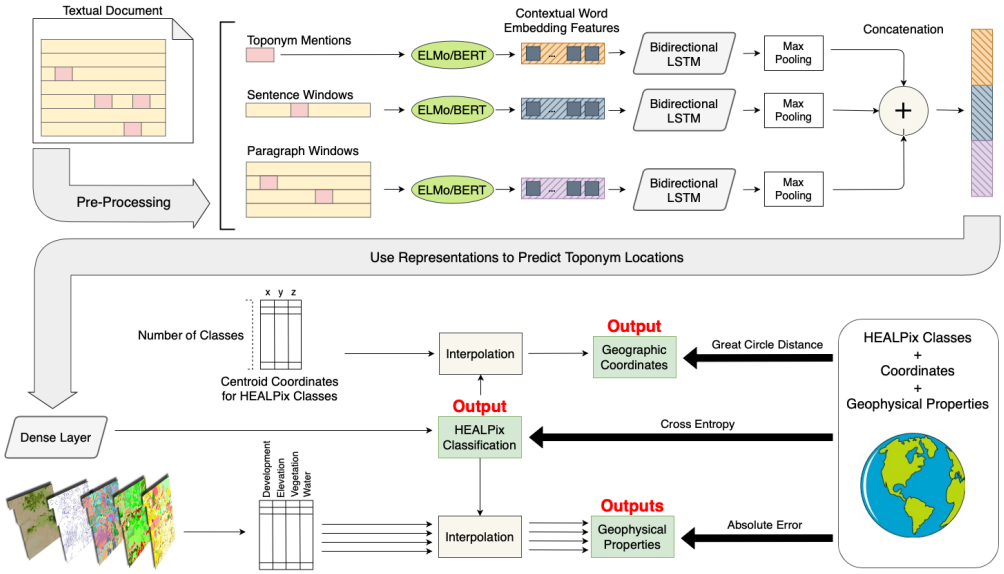


Figure 2. Overall workflow for the proposed toponym resolution method.

4. Experimental Evaluation

This section reports on the experimental evaluation of the model proposed in this work. Section 4.1 details both the evaluation methodology and the complete set of experiments that were performed. In turn, Section 4.2 presents the obtained results and their analysis, while Section 4.3 presents a summary discussion on key findings and main limitations.

4.1. Experimental Evaluation Methodology

Using the neural neural architecture and general methodology described in Section 3.3, also represented in Figure 2, we performed experiments with several alternative approaches, namely involving the use of (i) ELMo, (ii) BERT, (ii) Wikipedia data to augment the training instances, and (iii) external information corresponding to geophysical properties. A short description for each of these options is given next.

- **ELMo models**—This corresponds to our base approach, leveraging recurrent neural networks as described on Section 3.2, and using the ELMo method for generating contextual embeddings when representing the textual inputs.
- **BERT models**—To observe the impact of using different text representation methods, we replaced ELMo by BERT contextual word embeddings. This particular approach, based on the Transformer neural architecture and also described on Section 3.2, has been previously shown to provide superior results across a range of NLP tasks.
- **Wikipedia models**—To understand the impact of the size of the training dataset, we built a new corpus containing random articles collected from the English Wikipedia. We identified existing hyperlinks towards pages associated with geographic coordinates, and collected the source article text, the hyperlink text (i.e., the automatically generated place reference), and the target geographic coordinates. These data were used to create additional training instances, which were then filtered to match with the HEALPix regions present in the original corpora. Thus, Wikipedia was used to augment the available training instances, without modifying the region classification space of each corpus. Experiments were performed with either ELMo or BERT em-

beddings, in the setting that involved Wikipedia data. A total of 15,000 new instances were added to each of the three training datasets.

- **Models integrating geophysical properties**—We also experimented with the use of additional information corresponding to geophysical terrain properties associated with each of the HEALPix regions, namely terrain development (i.e., a quantification on the amount of impervious/developed versus natural terrain, inferred from historical land coverage datasets in the case of experiments with the WOTR corpus, and from modern sources in the remaining cases), percentage of vegetation, terrain elevation, and minimum distance from water zones. We collected this information from public raster datasets, incorporating it into the model using a similar technique to that associated with the interpolation of geographic coordinates. Specifically, we encoded each of the four geophysical properties as real values, and then created column vectors with values corresponding to the measurements associated with the centroid coordinates of each HEALPix class. We then computed a dot product between each of the column vectors and the adjusted HEALPix class probability vector, resulting in estimates for the geophysical properties. Additional loss functions were incorporated into the model, corresponding to the absolute difference between the predicted and the ground-truth values. The main intuition behind this set of experiments relates to seeing if the geophysical properties of the terrain, which are perhaps described in the text surrounding the place references, can guide the task of predicting the geographic coordinates. As in the previous case, experiments under this setting were performed with either ELMo or BERT contextual word embeddings.

Three well-known datasets, previously described on Section 2.5, were used to support the comparative evaluation of the different modeling alternatives, specifically (i) the *Local–Global Lexicon* (LGL) corpus [32], (ii) the *SpatialML* corpus [33], and (iii) the *War of the Rebellion* (WOTR) corpus [30]. The documents within these corpora have different sources (i.e., historical reports, news articles from local newspapers, and international news), naturally also corresponding to slightly different document characteristics. For example, SpatialML is mostly based on international news articles, which are likely more extensive and with more general toponym references than the other datasets. Table 1 presents a statistical characterization of the different datasets, including aspects such as the average document lengths or the number of toponyms per document.

We tried simulating the experimental conditions of previous studies, which enabled to compare model performance and results. The exact same data splits that were shared by the authors were used in the experiments with the WOTR corpus (i.e., the same division of the documents into training and testing datasets). Concerning the LGL and the SpatialML corpora, the data was randomly split considering 90% of the documents for training, and the extra 10% for testing (i.e., the training and testing datasets only contained place references given in different documents). The results measured over the LGL and SpatialML datasets are not directly comparable to those reported on previous studies, although large differences should nonetheless be indicative.

To calculate the HEALPix regions over the surface of the Earth, as described on Section 3.3, and for converting between latitude/longitude coordinates and HEALPix regions, we used the healpy Python library (http://pypi.org/project/healpy/ (accessed on 10 November 2021)). The representation of text contents through contextual word embeddings relied on pre-trained ELMo (http://allennlp.org/elmo (accessed on 10 November 2021)) and BERT (http://github.com/google-research/bert (accessed on 10 November 2021)) models. The proposed deep learning model was, in turn, implemented through the keras Python library (http://keras.io (accessed on 10 November 2021)).

To assess the prediction performance of geographic coordinates related to each toponym, the distance between the predicted and the ground-truth geo-spatial coordinates was calculated, using Vincenty's geodesic formulae [47] (i.e., a well-known method to calculate shortest geographic distances among pairs of points over the surface of the earth, achieving an accuracy within 0.5 mm). From the individual error measurements (i.e., from

the distances between the estimates and the ground-truth), the mean and median distances in kilometers were calculated, as well as the accuracy (i.e., percentage of correct results) taking a threshold on the distance values corresponding to 161 km. All the aforementioned metrics have been commonly used in previous studies related to document geo-coding or toponym resolution.

4.2. The Obtained Results

Table 2 summarizes the results obtained by our base model (i.e., the model using ELMo), comparing them against the results reported on previous publications that have used the same datasets and the same evaluation metrics (i.e., previous studies such as that from Lieberman et al. [32] have also used the LGL and SpatialML datasets, but these authors have instead measured performance through precision and recall metrics, in terms of finding correct matches to gazetteer entries). The proposed model achieved quite interesting results, in most cases outperforming the previous state of the art. We specifically measured very low mean error distances in the WOTR and LGL datasets, with a difference of minus 281 and 463 km, respectively, when compared to the previous best results. On the SpatialML corpus, the learning to rank system from Santos et al. still records the best mean distance error, although our model reached a much better median distance error. It is important to notice that resources such as Wikipedia, or global-coverage gazetteers such as GeoNames (http://www.geonames.org (accessed on 10 November 2021)), have been used in the annotation of toponym resolution corpora. Hence, the geo-spatial coordinates given in the ground-truth often match exactly with the coordinates associated with particular entries within these resources. Systems that rely on gazetteer matching, such as the learning to rank system from Santos et al. can somewhat benefit from the experimental setting that was considered, measuring distances towards ground-truth geo-spatial coordinates. Still, without involving gazetteer matching, our approach can, on average, achieve very accurate results.

Table 2. Experimental results obtained with the base ELMo models, without using Wikipedia data to complement the training instances, or the information on geophysical terrain properties.

Dataset	Mean Dist. (km)	Median Dist. (km)	Accuracy@161 km (%)
WOTR corpus			
TopoCluster [30]	604		57.0
TopoClusterGaz [30]	468	-	72.0
GeoSem [11]	445	-	68.0
Our Neural Model	**164**	11.48	**81.5**
LGL corpus			
GeoTxt [20]	1400	-	68.0
CamCoder [20]	700	-	76.0
TopoCluster [17]	1029	28.00	69.0
TopoClusterGaz [30]	1228	**0.00**	71.4
Learning to Rank [16]	742	2.79	-
Our Neural Model	**237**	12.24	**86.1**
SpatialML corpus			
Learning to Rank [16]	**140**	28.71	-
Our Neural Model	395	**9.08**	87.4

Table 3 presents the results obtained in a second set of experiments, in which we tested different modeling alternatives (e.g., using BERT instead of ELMo, and considering additional training data or geophysical properties). The results denote that the textual representation method has a serious impact on the results, with BERT contextual embeddings achieving better results across all the datasets (i.e., on average, an improvement of 41 km for the mean error, of 0.3 km for the median error, and an increment of 3.9% to the accuracy@161 measure), except in the median error over the SpatialML dataset (i.e., Table 3

shows that all the different alternatives achieved the same median error distance over the SpatialML dataset, when considering a numerical precision of two decimal digits).

Table 3. Experimental results obtained with different modeling alternatives.

Model and Dataset	Mean Distance (km)	Median Distance (km)	Accuracy@161 km (%)
WOTR corpus			
ELMo	164	11.48	81.5
ELMo + Wikipedia	158	11.28	82.4
ELMo + Geophysical	166	11.35	81.9
BERT	117	**10.99**	**87.3**
BERT + Wikipedia	122	11.04	86.4
BERT + Geophysical	**114**	**10.99**	**87.3**
LGL corpus			
ELMo	237	12.24	86.1
ELMo + Wikipedia	304	12.16	87.4
ELMo + Geophysical	282	12.24	87.7
BERT	**193**	11.81	90.1
BERT + Wikipedia	226	**11.51**	**90.6**
BERT + Geophysical	216	12.24	87.9
SpatialML corpus			
ELMo	395	**9.08**	87.4
ELMo + Wikipedia	364	**9.08**	88.5
ELMo + Geophysical	387	**9.08**	87.4
BERT	363	**9.08**	89.2
BERT + Wikipedia	**205**	**9.08**	**92.4**
BERT + Geophysical	339	**9.08**	89.4

Increasing the size of the training data with more training instances collected from Wikipedia, although not consistently, often resulted in a slight improvement on the results. For example, in the LGL corpus and both with ELMo or BERT embeddings, an increase in the mean error, an increase in the accuracy@161 measure, and a slight decrease in the median error, were observed. It should be noted that Wikipedia data has very different characteristics from the documents associated with the different corpora (e.g., historical reports or news articles), perhaps hindering the results. Future experiments can perhaps consider model pre-training with Wikipedia data followed by fine-tuning on the domain-specific corpora, instead of augmenting the set of training instances.

On what regards the experiments involving geophysical information, both with ELMo or BERT embeddings, we recorded only slight and non-consistent improvements over the results. On the WOTR and SpatialML datasets, the model seems to benefit from the addition of geophysical information, whereas in the LGL corpus the model integrating geophysical information performed worse. In future work, we plan to do a more in-depth assessment on the contribution of different geophysical properties, improving also the assignment of the ground-truth geophysical properties to HEALPix regions.

Besides high level assessments in terms of overall distance errors, we also attempted to analyze specific cases in which the models performed correctly or incorrectly, in an attempt to identify patterns in the results. Table 4 present place name references, taken from each of the datasets, for which the base model (i.e., the model leveraging ELMo embeddings, without Wikipedia training data and not using the physical terrain properties) produced either the lowest or the highest distance errors. The locations having low prediction error included demonyms (e.g., *English* in the SpatialML corpus, resolved to the *England*, *UK* location with a small error of 2.44 km), or small places specified through vernacular names (e.g., the case of *Owen's Big Lake* in the WOTR corpus). These would likely be incorrectly disambiguated in approaches relying on gazetteer matching. Toponym references with a large distance error include highly ambiguous names (e.g., *Capital* in the SpatialML corpus), or references to large areas (e.g., *North America* in the LGL corpus). Although Table 4 only shows examples for the base model, similar results would also be obtained with the other

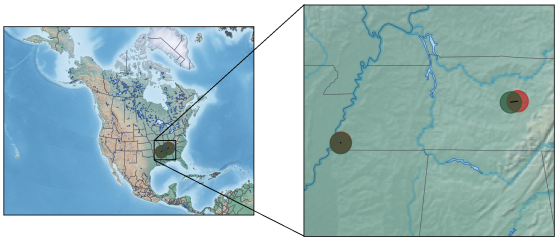
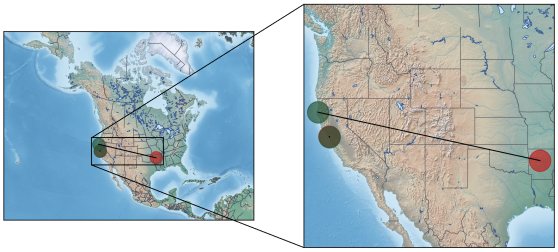
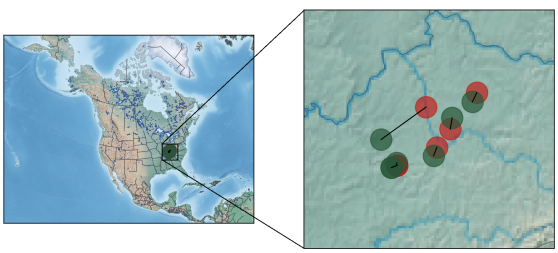
modeling alternatives (i.e., many of the same place names would still be seen on the lists featuring the lowers/highest errors).

Table 4. Examples of toponyms, taken from the different corpora, with low/high prediction errors as assigned by the base model leveraging ELMo embeddings.

Corpus	Lowest Distance Errors (km)	Highest Distance Errors (km)
WOTR	(0.63) Mexico (1.00) Resaca (1.09) Owen's Big Lake	(3104.59) Fort Welles (3141.29) Washington (3682.01) Astoria
LGL	(1.21) W.Va. (1.36) Butler County (1.51) Manchester	(8854.04) Ohioans (9225.86) North America (9596.54) Nigeria
SpatialML	(0.45) Tokyo (2.38) Lusaka (2.44) English	(9687.43) Capital (10,818.50) Omaha (13,140.64) Atlantic City

Table 5 further illustrates the results obtained with the base model, showing examples of textual utterances containing place references, together with the geo-spatial locations corresponding to the predictions and to the ground-truth. In each example, the toponyms are highlighted, and the related map shows the real location (green points) and the predicted locations (red points), including the correspondences between these points pictured by black lines. Distinct cases where the error among the predicted and the real points is either small or considerably big are shown. Note that some of the examples include toponyms co-occurring in close proximity (e.g., *Memphis*, *Tenn.* in the first example), which can give clues about the locations. In the third example, all the toponyms have locations with small errors assigned with an average distance of nearly 16.6 km, among which a reference to a small location named *Paris* is shown (i.e., an usual ambiguous place name that the model resolves well using the help of the surrounding context).

Table 5. Examples from the WOTR corpus, showing predictions vs. ground truth coordinates.

Predicted vs. Ground-Truth Coordinates	Textual Contents
	[Indorsement.] HDQRS. DETACHMENT SIXTEENTH ARMY CORPS, **Memphis**, **Tenn.**, 12 June 1864. Respectfully referred to Colonel David Moore, commanding THIRD DIVISION, SIXTEENTH Army Corps, who will send the THIRD Brigade of his command, substituting some regiment for the Forty-ninth Illinois that is not entitled to veteran furlough, making the number as near as possible to 2000 men. They will be equipped as within directed, and will move to the railroad depot as soon as ready. You will notify these headquarters as soon as the troops are at the depot. By order of Brigadier General A. J. Smith: J. HOUGH, Assistant Adjutant-General.
	**HYDESVILLE**, 21 October 1862 SIR: I started from this place this morning, 7.30 o'clock, en route for **Fort Baker**. The express having started an hour before, I had no escort. About two miles from Simmons' ranch I was attacked by a party of Indians. As soon as they fired they tried to surround me. I returned their fire and retreated down the hill. A portion of them cut me off and fired again. I returned their fire and killed one of them. They did not follow any farther. I will start this evening for my post as I think it will be safer to pass this portin of the country in the night. Those Indians were lurking about of rthe purpose of robbing Cooper's Mills. They could have no othe robject, and I think it would be well to have eight or ten men stationed at that place, as it will serve as an outpost for the settlement, as well as a guard for the mills. The expressmen disobeyed my orders by starting without me this morning. I have the honor to be, very respectfully, your obedient servant, H. FLYNN, Captain, Second Infantry California Volunteers. First Lieutenant JOHN HANNA, Jr., Acting Assistant Adjutant-General, Humboldt Military District.
	**LEXINGTON**, **KY.**, 11 June 1864–11 p.m. Colonel J. W. WEATHERFORD, **Lebanon**, **Ky.** Have just received dispatch from General Burbridge at **Paris**. He says direct Colonel Weatherford to closely watch in the direction of **Bardstown** and **Danville**, and if any part of the enemy's force appears in that region to attack and destroy it. J. BATES DICKSON, Captain and Assistant Adjutant-General.

4.3. Discussion on the Overall Results

Overall, the obtained results corroborate the superiority of the proposed method over earlier approaches, significantly outperforming the previous state of the art.

The use of contextual word embeddings has previously been shown to be beneficial across a range of NLP tasks, specifically when involving relatively small quantities of annotated training data. Our results further attest this observation, and in particular we observed better results with BERT embeddings, compared to our base model leveraging ELMo embeddings. Increasing the training datasets with out-of-domain examples collected from Wikipedia, or the addition of information on geophysical terrain properties, only marginally improved the results. Still, additional experiments should be considered to further assess the contribution of both these ideas. For instance, much larger samples of Wikipedia instances could be considered for extending the training datasets, or we could instead consider the use of Wikipedia data for initial pre-training followed by fine-tuning the models using only in-domain data.

The usage of external information capturing terrain properties is a particularly interesting research direction, which we would like to pursue and assess in more detail. We can for instance improve the assignment of ground-truth measurements to HEALPix cells, using areal statistics instead of just collecting data for the centroid coordinates of each cell. The four different properties considered in our initial experiments are also perhaps not all equally informative , and additional sources of information (e.g., rasters encoding the human population density, which is used as a prior in many toponym resolution systems, or information on land usage derived from the OpenStreetMap) could be also considered. Instead of high-level products derived from remote sensing, we can perhaps also consider multi-modal approaches that explore the direct usage of satellite imagery as the source of external information (e.g., a convolutional neural network can be used to derive vector representations for images corresponding to the HEALPix regions, and these representations could then be similarly incorporated into our toponym resolution model).

All our experiments were executed on relatively modest hardware (e.g., standard PCs with Titan Xp GPUs, featuring 12 GB of memory), also taking a short amount of time for model training and evaluation (e.g., training takes just a couple of hours, in each of the considered datasets). This is due to the fact that training involves updating a relatively small number of parameters, given that the models that compute ELMo or BERT embeddings are kept fixed (i.e., only our bidirectional LSTM layers, and the feed-forward layer associated with the class predictions, are adjusted). However, for future work, it would be interesting to experiment with the end-to-end fine-tuning of a BERT model for toponym resolution (i.e., use BERT directly, instead of bidirectional LSTM layers processing embeddings that are computed separately). One such approach can perhaps offer better results, although at the cost of much higher computational requirements.

5. Conclusions and Future Work

This article addressed the toponym resolution problem, proposing a novel neural network architecture specifically designed for the task. The network considers multiple textual inputs, corresponding to the toponym to be disambiguated plus the associated contextual information, and uses pre-trained contextual word embeddings (ELMo or BERT) to represent the text. Moreover, the neural network also considers multiple outputs, predicting a probability distribution over coarse-grained geo-spatial regions, and then using this probability distribution to guide the prediction of geo-spatial coordinates of latitude and longitude matching the input toponym.

We conducted evaluation experiments with three datasets extensively used in previous studies, and evaluated the impact of different modeling alternatives (e.g., using BERT versus ELMo embeddings, using additional training data collected from Wikipedia, or using external information on geophysical terrain properties to guide model training). Overall, the experimental results show that the proposed approach can clearly surpass previously reported methods over the same datasets.

For future work, it may be interesting to explore multi-lingual or cross-lingual embeddings (e.g., the multi-lingual BERT models released by Google), to support the idea of using existing data in a given (set of) language(s) and design an approach capable of working on texts from distinct languages. Besides ELMo and BERT, there are numerous other contextual word embedding models that could be explored [40,41,48,49], pre-trained with larger datasets and/or considering additional language modeling objectives. One example is RoBERTa [50], an optimized version of BERT (i.e., trained with larger mini-batches, more data and for a larger amount of time, considering a different policy for adjusting the learning rate, and eliminating the pre-training objective concerning the prediction of the next sentence) that has been shown to be more effective, producing state-of-the-art results across a wide range of tasks. Other examples include models such as LUKE [51] or ERICA [52], pre-trained to better capture entities and relations between entities in text, and hence perhaps also performing better in tasks that involve dealing with toponyms. Perhaps even more interestingly, instead of using pre-trained contextual embedding models as feature extractors (i.e., in the generation of fixed representations that are then provided as input to bi-directional LSTMs), we can consider directly fine-tuning models such as BERT to the toponym disambiguation task. This particular alternative would be more demanding computationally, although perhaps it can also lead to better results.

It would also be interesting to further extend the experimental validation, e.g., using other corpora from different sources (e.g., scientific documents, as used in the competition on toponym resolution in the SemEval-2019 [21] challenge), and comparing the performance of the proposed approach against a larger set of previous systems. One possibility would involve integrating the proposed approach within EUPEG [35,36], a recent benchmark platform developed for evaluating toponym recognition and disambiguation systems, that integrates a wide range of document collections and systems. Still, in its present version, EUPEG does not support model training with information from the corpora that are integrated into the platform.

Author Contributions: Conceptualization, Ana Bárbara Cardoso, Bruno Martins, and Jacinto Estima; Methodology, Ana Bárbara Cardoso, Bruno Martins, and Jacinto Estima; Software, Ana Bárbara Cardoso; Validation, Ana Bárbara Cardoso; Formal Analysis, Ana Bárbara Cardoso, Bruno Martins, and Jacinto Estima; Investigation, Ana Bárbara Cardoso, Bruno Martins, and Jacinto Estima; Resources, Ana Bárbara Cardoso and Bruno Martins; Data Curation, Ana Bárbara Cardoso and Bruno Martins; Writing—Original Draft Preparation, Ana Bárbara Cardoso; Writing—Review and Editing, Bruno Martins, and Jacinto Estima; Visualization, Ana Bárbara Cardoso; Supervision, Bruno Martins, and Jacinto Estima. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported through the European Union's Horizon 2020 research and innovation programme under grant agreement No 874850 (MOOD), as well as through the Fundação para a Ciência e Tecnologia (FCT) through the project grants with references PTDC/EEI-SCR/1743/2014 (Saturn), T-AP HJ-253525 (DigCH), PTDC/CCI-CIF/32607/2017 (MIMU), and POCI/01/0145/FEDER/031460 (DARGMINTS), and through the INESC-ID multi-annual funding from the PIDDAC programme (UIDB/50021/2020).

Data Availability Statement: Most of the data that supported our experiments are available from the following public domain repositories, created by third parties: WOTR Corpus (http://github.com/utcompling/WarOfTheRebellion (accessed on 10 November 2021)); LGL Corpus (http://raw.githubusercontent.com/geoai-lab/EUPEG/master/corpora/lgl.xml (accessed on 10 November 2021)); BERT Embeddings and Code (http://github.com/google-research/bert (accessed on 10 November 2021)); ELMo Embeddings and Code (http://allennlp.org/elmo (accessed on 10 November 2021)). Section 2.5 describes the corpora in the previous enumeration, while Section 3.2 describes the BERT/ELMo word embedding models. The SpatialML corpus, also described in Section 2.5, is available from the Linguistic Data Consortium (http://catalog.ldc.upenn.edu/LDC2011T02 (accessed on 10 November 2021)), although restrictions apply to its availability. The authors of the SpatialML corpus are not involved with the research reported here. All the source code that supported our tests, together with pre-processed versions of the datasets that are publicly available, is available in a public GitHub repository (http://github.com/barbarainacioc/toponym-resolution (accessed on 10 November 2021)).

Acknowledgments: We gratefully acknowledge the support of NVIDIA Corporation, with the donation of the two Titan Xp GPUs used in our experiments.

Conflicts of Interest: The authors declare no conflict of interest. The funders also had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Amitay, E.; Har'El, N.; Sivan, R.; Soffer, A. Web-a-where: Geotagging web content. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, 25–29 July 2004; pp. 273–280.
2. Monteiro, B.; Davis, C.; Fonseca, F. A survey on the geographic scope of textual documents. *Comput. Geosci.* **2016**, *96*, 23–34. [CrossRef]
3. Cardoso, N.; Martins, B.; Chaves, M.; Andrade, L.; Silva, M.J. The XLDB group at GeoCLEF 2005. In *Workshop of the Cross-Language Evaluation Forum for European Languages, Proceedings of the 6th Workshop of the Cross-Language Evalution Forum, CLEF 2005, Vienna, Austria, 21–23 September 2005*; Springer: Berlin/Heidelberg, Germany, 2005.
4. Martins, B.; Calado, P. Learning to rank for geographic information retrieval. In Proceedings of the ACM SIGSPATIAL Workshop on Geographic Information Retrieval, Zurich, Switzerland, 18–19 February 2010.
5. Coelho, J.A.; Magalhães, J.A.; Martins, B. Improving Neural Models for the Retrieval of Relevant Passages to Geographical Queries. In Proceedings of the ACM SIGSPATIAL Conference on Advances in Geographic Information Systems, Beijing, China, 2–5 November 2021.
6. Purves, R.S.; Clough, P.; Jones, C.B.; Hall, M.H.; Murdock, V. Geographic Information Retrieval: Progress and Challenges in Spatial Search of Text. *Found. Trends Inf. Retr.* **2018**, *12*, 164–318. [CrossRef]
7. Wing, B. Text-Based Document Geolocation and Its Application to the Digital Humanities. Ph.D. Thesis, University of Texas at Austin, Austin, TX, USA, 2015.
8. Melo, F.; Martins, B. Automated geocoding of textual documents: A survey of current approaches. *Trans. GIS* **2017**, *21*, 3–38. [CrossRef]
9. Berman, M.; Mostern, R.; Southall, H. *Placing Names: Enriching and Integrating Gazetteers*; Indiana University Press: Bloomington, IN, USA, 2016.
10. Manguinhas, H.; Martins, B.; Borbinha, J.; Siabato, W. The DIGMAP geo-temporal web gazetteer service. *E-Perimetron* **2009**, *4*, 9–24.
11. Ardanuy, M.; Sporleder, C. Toponym disambiguation in historical documents using semantic and geographic features. In Proceedings of the Conference on Digital Access to Textual Cultural Heritage, Göttingen, Germany, 1–2 June 2017; pp. 175–180.
12. Leidner, J. Toponym Resolution in Text. Ph.D. Thesis, University of Edinburgh, Edinburgh, UK, 2007.
13. Freire, N.; Borbinha, J.; Calado, P.; Martins, B. A metadata geoparsing system for place name recognition and resolution in metadata records. In Proceedings of the Annual International ACM/IEEE Joint Conference on Digital Libraries, Ottawa, ON, USA, 13–17 June 2011; pp. 339–348.
14. Karimzadeh, M.; Pezanowski, S.; MacEachren, A.; Wallgrün, J. GeoTxt: A scalable geoparsing system for unstructured text geolocation. *Trans. GIS* **2019**, *23*, 118–136. [CrossRef]
15. Lieberman, M.; Samet, H. Adaptive context features for toponym resolution in streaming news. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, Portland, OR, USA, 12–16 August 2012; pp. 731–740.
16. Santos, J.; Anastácio, I.; Martins, B. Using machine learning methods for disambiguating place references in textual documents. *GeoJournal* **2015**, *80*, 375–392. [CrossRef]
17. DeLozier, G.; Baldridge, J.; London, L. Gazetteer-independent toponym resolution using geographic word profiles. In Proceedings of the AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 2382–2388.
18. Speriosu, M.; Baldridge, J. Text-driven toponym resolution using indirect supervision. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Sofia, Bulgaria, 4–9 August 2013; Volume 1, pp. 1466–1476.
19. Adams, B.; McKenzie, G. Crowdsourcing the character of a place: Character-level convolutional networks for multilingual geographic text classification. *Trans. GIS* **2018**, *22*, 394–408. [CrossRef]
20. Gritta, M.; Pilehvar, M.; Collier, N. Which Melbourne? Augmenting geocoding with maps. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 1285–1296.
21. Weissenbacher, D.; Magge, A.; O'Connor, K.; Scotch, M.; Gonzalez-Hernandez, G. SemEval-2019 task 12: Toponym resolution in scientific papers. In Proceedings of the Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 907–916.
22. Yan, Z.; Yang, C.; Hu, L.; Zhao, J.; Jiang, L.; Gong, J. The Integration of Linguistic and Geospatial Features Using Global Context Embedding for Automated Text Geocoding. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 572. [CrossRef]
23. Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep contextualized word representations. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; Volume 1, pp. 2227–2237.
24. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 6–7 June 2019; Volume 1, pp. 4171–4186.

25. Górski, K.; Hivon, E.; Banday, A.; Wandelt, B.; Hansen, F.; Reinecke, M.; Bartelman, M. HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *Astrophys. J.* **2005**, *622*, 759–771. [CrossRef]
26. Bensalem, I.; Kholladi, M.K. Toponym disambiguation by arborescent relationships. *J. Comput. Sci.* **2010**, *6*, 653. [CrossRef]
27. Moncla, L. Automatic Reconstruction of Itineraries from Descriptive Texts. Ph.D. Thesis, University of Pau and Pays de l'Adour, Pau, France, 2015.
28. Xu, C.; Li, J.; Luo, X.; Pei, J.; Li, C.; Ji, D. DLocRL: A deep learning pipeline for fine-grained location recognition and linking in tweets. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019.
29. Wing, B.; Baldridge, J. Simple supervised document geolocation with geodesic grids. In Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, USA, 19–24 June 2011; pp. 955–964.
30. DeLozier, G.; Wing, B.; Baldridge, J.; Nesbit, S. Creating a novel geolocation corpus from historical texts. In Proceedings of the ACL Linguistic Annotation Workshop, Berlin, Germany, 11 August 2016; pp. 188–198.
31. Ord, J.K.; Getis, A. Local spatial autocorrelation statistics: Distributional issues and an application. *Geogr. Anal.* **1995**, *27*, 286–306. [CrossRef]
32. Lieberman, M.; Samet, H.; Sankaranarayanan, J. Geotagging with local lexicons to build indexes for textually-specified spatial data. In Proceedings of the IEEE Conference on Data Engineering, Long Beach, CA, USA, 1–6 March 2010; pp. 201–212.
33. Mani, I.; Doran, C.; Harris, D.; Hitzeman, J.; Quimby, R.; Richer, J.; Wellner, B.; Mardis, S.; Clancy, S. SpatialML: Annotation scheme, resources, and evaluation. *Lang. Resour. Eval.* **2010**, *44*, 263–280. [CrossRef]
34. Gritta, M.; Pilehvar, M.; Limsopatham, N.; Collier, N. What's missing in geographical parsing? *Lang. Resour. Eval.* **2018**, *52*, 603–623. [CrossRef] [PubMed]
35. Wang, J.; Hu, Y. Enhancing spatial and textual analysis with EUPEG: An extensible and unified platform for evaluating geoparsers. *Trans. GIS* **2019**, *23*, 1393–1419. [CrossRef]
36. Wang, J.; Hu, Y. Are we there yet? Evaluating state-of-the-art neural networkbased geoparsers using EUPEG as a benchmarking platform. In Proceedings of the ACM SIGSPATIAL Workshop on Geospatial Humanities, Chicago, IL, USA, 5 November 2019; pp. 1–6.
37. Goldberg, Y. *Neural Network Methods in Natural Language Processing*; Morgan & Claypool: San Rafael, CA, USA, 2017.
38. Schmidhuber, J.; Hochreiter, S. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
39. Smith, N.A. Contextual word representations: A contextual introduction. *arXiv* **2019**, arXiv:1902.06006.
40. Liu, Q.; Kusner, M.J.; Blunsom, P. A Survey on Contextual Embeddings. *arXiv* **2020**, arXiv:2003.07278.
41. Qiu, X.; Sun, T.; Xu, Y.; Shao, Y.; Dai, N.; Huang, X. Pre-trained Models for Natural Language Processing: A Survey. *arXiv* **2020**, arXiv:2003.08271.
42. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. In Proceedings of the International Conference on Learning Representations, Scottsdale, AZ, USA, 2–4 May 2013.
43. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
44. Melo, F.; Martins, B. Geocoding textual documents through the usage of hierarchical classifiers. In Proceedings of the Workshop on Geographic Information Retrieval, Paris, France, 26–27 November 2015; pp. 1–9.
45. Eger, S.; Youssef, P.; Gurevych, I. Is it time to swish? Comparing deep learning activation functions across NLP tasks. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 4415–4424.
46. Smith, L. Cyclical learning rates for training neural networks. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Santa Rosa, CA, USA, 24–31 March 2017; pp. 464–472.
47. Vincenty, T. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Surv. Rev.* **1975**, *23*, 88–93. [CrossRef]
48. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Online, 16–20 November 2020.
49. Rogers, A.; Kovaleva, O.; Rumshisky, A. A Primer in BERTology: What we know about how BERT works. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 842–866. [CrossRef]
50. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
51. Yamada, I.; Asai, A.; Shindo, H.; Takeda, H.; Matsumoto, Y. Luke: Deep contextualized entity representations with entity-aware self-attention. *arXiv* **2020**, arXiv:2010.01057.
52. Qin, Y.; Lin, Y.; Takanobu, R.; Liu, Z.; Li, P.; Ji, H.; Huang, M.; Sun, M.; Zhou, J. ERICA: Improving entity and relation understanding for pre-trained language models via contrastive learning. *arXiv* **2020**, arXiv:2012.15022.