# Toponym Resolution in Text with Neural Language Models

**Diogo Viegas**

Instituto Superior Técnico

## Abstract

Toponym resolution concerns the task of mapping names for places (toponyms), previously detected in a textual document, into the corresponding locations on Earth (e.g., through geographical coordinates). Complexity arises when a single place name refer to many real locations (e.g., *Paris* can refer to more than 50 places in over 20 countries around the world), making disambiguation a non-trivial procedure. We present a novel toponym resolution method based on deep learning, taking inspiration from recent methods achieving state-of-the-art results. The proposed neural network architecture is based on a pre-trained language model, sharing parameters for the processing of different inputs (e.g., the toponym to disambiguate along with the surrounding words). We use the HEALPix method to model toponym resolution as a classification task. Subsequently, the result of the classification is used to inform the prediction of geographic coordinates for each place name reference, through a separate layer that directly applies the great circle distance as a loss function. Additionally, we also test the use of external geophysical information, by using an additional term in the cost function for each geophysical property considered, thus allowing the model to obtain more information about the locations when making predictions. The proposed model was tested on collections of documents used and developed in previous studies. The obtained results show that the proposed model can significantly outperform previous approaches.

## 1 Introduction

Toponym resolution, also known as geo-parsing, geo-grounding, or place name resolution, aims to assign unambiguous locations (e.g., geographic coordinates) to location names mentioned within textual documents. The task is usually performed in two independent steps. The first step concerns toponym detection, where the spans of text corresponding to place names are identified. In the second step, toponym disambiguation or geocoding, each of the discovered place names can be mapped to latitude and longitude coordinates, corresponding to the centroid of its physical location. This work focuses exclusively on the toponym disambiguation task, intending to assign an unambiguous position over the surface of the Earth to each place name reference in a textual document.



The names of locations are usually ambiguous expressions, since these names can correspond to distinct geographical referents (e.g., the place name *Paris* is associated to several geographic locations, besides the capital city of *France*), and the same location can often be associated to several alternative names (e.g., the names *New York* and *Big Apple* can both be used as references to *New York City* in the *United States*).

Toponym resolution is important in many real-life applications, for instance supporting geospatial text analysis within digital humanities, computational social sciences (Wing et al. 2015), and other scientific domains.

This article proposes a novel method using a Transformer-based pre-trained language model, that builds a representation of the toponym to be disambiguated together with its surrounding context. Other toponyms, or even regular words appearing in the surrounding context, can be characteristic of a certain region, which can provide clues about the locaftion of the mention (e.g., the words *Louvre* or *Seine* are usually associated with the toponym *Paris*).

We use the Hierarchical Equal Area isoLatitude Pixelization (HEALPix) (Gorski et al. 2005) scheme to partition the geographic study region into a set of cells, in order to approach the task of toponym resolution as a classfication task, by associating each place name reference with a region on the surface of the Earth. Furthermore, a regression loss based on the great circle distance (i.e., the shortest distance between two points on the surface of a sphere) is used for the output corresponding to the geographical coordinates of the place name reference.

Additionally, we test the use of external information corresponding to geophysical properties (e.g., terrain properties, natural resources, etc.) in order to turn the system more robust and improve performance. This information was extracted from external raster datasets and incorporated in the proposed model, through an additional term in the cost function for each geophysical property considered. Our goal with this experiment is to enable the model to obtain more information about the locations when making predictions.

The rest of this article is organized as follows. Section 2 describes previously developed studies in the field of toponym resolution. Section 3 details the proposed model, while Section 4 presents the corpora used in our experiments, together with the evaluation methodology. This sec-

tion also presents the obtained results. Finally, Section 5 summarizes our conclusions and describes ideas for future work, within the field of toponym resolution.

## 2 Background

Many previous methods for toponym resolution have explored the use of heuristics, relying on an external source of knowledge (e.g., a gazetteer) to assign textual references to corresponding locations (Leidner 2007; Leidner, Sinclair, and Webber 2003). More recently, several studies have considered supervised learning approaches that take heuristics as features in machine learning models (Santos, Anastácio, and Martins 2015; Lieberman and Samet 2012; Wang et al. 2019), while later studies have explored the use of the text of a document within the surrounding context, through statistical language models (Wing et al. 2015; DeLozier, Baldridge, and London 2015).

Nowadays, the task of toponym resolution can also be addressed through the use of state-of-the-art deep learning methods for NLP, e.g. related to the use of contextual embedding models such as BERT or RoBERTa (Liu, Kusner, and Blunsom 2020). These neural methods offer several advantages over existing rule-based techniques for toponym resolution, namely the ability to naturally leverage contextual clues to improve predictions and disambiguate location names. However, previous studies have shown that the performance of these methods varies greatly when applied to corpora of different genres and domains.

One of these approaches is the *CamCoder* system (Gritta, Pilehvar, and Collier 2018), which combines a sparse vector representation that generates geographic features from text that go beyond lexical features (i.e., geographic representation of location mentions), and a system that uses representations based on lexical features. The system combines lexical and geographic information, considering four inputs: the target entity (i.e., the toponym mention to disambiguate), the context location mentions (i.e., other places mentioned in the same context), the context words (excluding location mentions), and the feature vector named *MapVec*. The first three inputs are fed into convolutional layers with global maximum pooling, while the fourth input is fed into a fully dense layer. The resulting vectors are then combined and passed into a final layer that predicts the location prediction region based on a classification into regions. This system achieved state-of-the-art results when compared to its competitors, thus showing that lexical clues improve the performance of toponym resolution (Gritta, Pilehvar, and Collier 2018).

Another recent deep learning approach based on contextual embeddings was developed by Cardoso, Martins, and Estima (2019). This system relies on contextual embeddings model such as ELMo or BERT to transform the input text, feeding these representations into a neural network in order to make a region classification based on a geodesic grid. This method achieved state-of-the-art results on two different dataset: WOTR and LGL, when compared with other systems such as *CamCoder*. Slightly better results were also achieved when using BERT instead of ELMo for embedding the input, although the author did not attempt to fine-tune BERT, instead relying on LSTMs to generate representations from BERT embeddings

More recently, Radford (2021) presented yet another neural network method named ELECTRO-map. This end-to-end probabilistic model for toponym resolution relies on the fine-tuning of a transformer language model, namely DistillRoBERTa, to minimize the negative log-likelihood of a five component mixture of von-Mises Fisher (vMF) distributions. More particularly, the vMF distribution generalizes the von Mises distributon beyond two dimensionto the surfaces of sphres or hyperspheres. For every input text, the model predicts parameters for five parameters vMF distributions, as well as a set of mixing probabilities describing the weights given to each of the five components. By using five components, the model can then fit a more flexible distributional shape than it would be able to with a single vMF component. The authors also proposed several solutions for aggregating results to a single latitude/longitude prediction per observation, namely choosing the single highest probability prediction and choosing the best prediction from the mixture, given apriori knowledge. The ELECTRO-map system achieved state-of-the-art results while choosing the first solution to aggregate results, when compared to Mordecai (i.e., a full-text geoparsing system that extracts place names from text, resolves them to their correct entries in a gazetteer, and returns structured geographic information for the resolved place name).

One more neural system was developed by Kulkarni et al. (2021), named the Multi-Level Geocoder (MLG). This approach, unlike *CamCoder*, does not rely on gazetteer metadata and population signals, therefore avoiding biased predictions towards locations with large populations. MLG learns spatial language representations by mapping toponyms from text to coordinates on the Earth's surface. In particular, The system uses multi-level S2 [1] cells as the output of a multi-headed feature encoding model. The model defines losses at several levels of granularity (L5, L6, L7) and minimizes them jointly. This method was evaluated in the same datasets used to evaluate *CamCoder*, and achieved better results than its competitors, displaying that the architecture is effective on text geocoding. Moreover, results also showed that it is possible and even preferable to solely rely on lexical clues present in the text. Additionally, the authors also verified inconsistencies in the true coordinates of the different datasets, whereby decided to unify the true coordinates corresponding to the same target entity, thus creating a consistent evaluation.

## 3 Proposed Model

The end-to-end toponym resolution model (1) detailed in this article relies on the fine-tuning of a pre-trained language model, and builds a representation from the toponym that is to be disambiguated, as well as its surrounding context. This section is organized as follows: Section 3.1 describes the architecture of the model, while Section 3.2 details the use of geophysical properties as an additional prediction in our

[1] https://s2geometry.io/

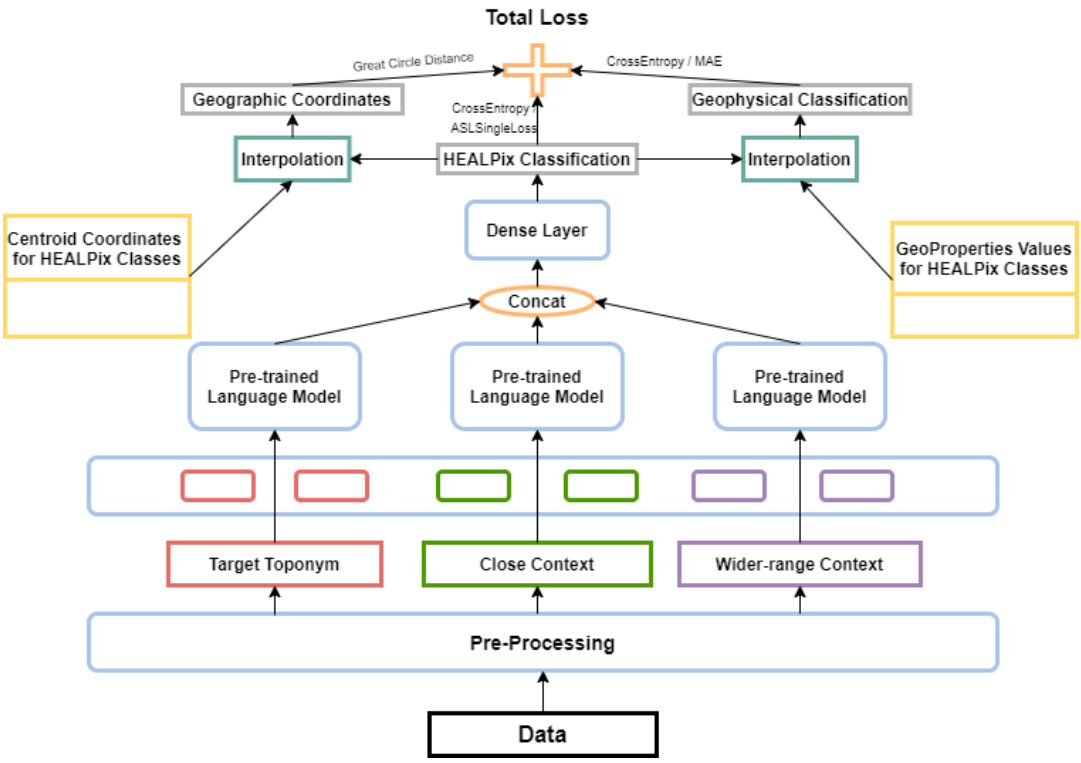



Figure 1: Proposed model architecture

model. Finally, Section 3.3 illustrates the training hyperparameters used in our experiments.

### 3.1 Model Architecture

The proposed model considers multiple textual inputs, following the input partition proposed by Cardoso, Martins, and Estima (2019). Firstly, the textual document needs to be tokenized, using the language model tokenizer. Then, the tokenized text is divided into three sequences of tokens, namely: the toponym mention to be disambiguated (target toponym), the close context around the toponym mention (i.e., a fixed window, to the left and right sides of the target toponym, totalling 50 tokens), and a wider range context around the toponym mention (i.e., a fixed window, to the left and right sides of the target toponym, totalling 510 tokens). It is worth mentioning that most language models have a maximum length limit for the input of 512 tokens, and knowing that each input vector needs to be fed with a [CLS] and a [SEP] token (at the beginning and ending of the vector, respectively), the bare maximum of actual textual input tokens has to be 510.

Each of these sequences is then fed into a pre-trained language model, sharing parameters for the processing of the three sequences. Since we are dealing with a classification task, only the first output vector associated with the special token [CLS] is extracted from each. The resulting three vectors are then concatenated to form a representation of the whole input data. This vector is then processed by an output layer that produces a probability distribution over the regions that were defined by the Earth partitioning algorithm.

Probably the most widely known pre-trained language model is the BERT model. In brief, the BERT model architecture corresponds to a multi-layer bidirectional Transformer encoder based on the original implementation. It is designed to train deep bidirectional representations by jointly conditioning on the left and right context in all layers, resulting in the capacity to be fine-tuned with the addition of an output layer to create models for downstream tasks (i.e., supervised-learning tasks that use a pre-trained model) such as classifying input texts (Devlin et al. 2018).

As previously mentioned, we choose to tackle the toponym resolution task as a prediction problem, where each place name reference is associated with a certain region of the Earth through a geodesic grid. We then use the classification probability distribution to obtain geographical coordinates (i.e., latitude and longitude) of each recognized place name, through a regression loss.

The geodesic grid used to support the classification is built through the Hierarchical Equal Area isoLatitude Pixelization (HEALPix) scheme, proposed by Gorski et al. (2005). In brief, the HEALPix algorithm partitions a spherical representation for the Earth's surface, generating equal are cells corresponding to different regions. Throughout the experiments reported on this article, we fixed the resolution parameter to $N_{side}$=256, which is equivalent to considering a maximum of 786,432 regions. However, in reality, the number of classes will be much smaller, given that most regions will not be associated to any instance in the data.

The predicted class probability over the partition schema is one of the outputs of the system, so a loss function is computed over it. A natural choice is to compute the cross-entropy loss between the predicted probabilities and the one-hot true vector class. Alternatively, and taking advantage of studies performed in the field of Computer Vision, we also test the use of an asymmetric loss function. Asymmetric loss functions were first introduced by Zhou et al. (2021), and later explored by Ben-Baruch et al. (2020), and are robust to learning with noisy labels for various types of noise.

The predicted class probability is then passed through a softmax layer. The softmax probability distribution vector is then raised to the third power and re-normalized (i.e., a more peaked distribution is built from the softmax results, emphasizing the most likely region), and a final interpolation between this peaked probability distribution vector and a vector representation of the cell centroids (i.e., a matrix that contains the centroid coordinates of each HEALPix class) is performed to obtain the predicted geospatial coordinates of the toponym to be disambiguated, similarly to what was done in the work of Cardoso, Martins, and Estima (2019). The obtained vector is then connected to a new loss function that computes the great circle distance between the predicted and the ground-truth coordinates.

### 3.2 Geophysical properties

Additionally, and besides HEALPix regions and geospatial coordinates, other outputs can be considered. We estimate geophysical properties, such as land cover, elevation, and percentage of vegetation, associated with the predicted HEALPix cell, in order to guide the model towards correct location predictions.

This geophysical information is extracted from datasets in the raster format (i.e., a matrix of cells organized into rows and columns where each cell contains a value representing information), and incorporated into the model with the same interpolation technique used to estimate the prediction of geospatial coordinates, previously described. More specifically, each of the geophysical properties is encoded into real values, and then vectors corresponding to the measurements associated to each HEALPix class are created. In order to obtain these values, a polygon associated with the HEALPix class is used to crop the original rasters, thus obtaining a representative value of the region, and not only of the HEALPix cell centroid. The dot product between each vector and the previously calculated peaked probability distribution vector is computed, and the results are connected to new loss functions that correspond to the absolute difference between the predicted and the real values (in the case of the elevation and vegetation geophysical properties), and the cross-entropy between the predicted and label vectors (in the case of the land coverage geophysical property).

The raster datasets are obtained from the "Global Map data archives" project, and the "International Satellite Land-Surface Climatology Project" initiative. The following geophysical properties are considered: (1) the land coverage classification (i.e., amount of developed versus natural terrain), inferred from an historical source[2] (in the case of the WOTR corpus), and from modern sources[3] (in the remaining datasets); (2) the percentage of vegetation [4] (i.e., the ratio of the area covered with branches and leaves of trees); (3) the terrain elevation[5] (i.e., the elevation data at 1m interval covering the whole world) .

### 3.3 Training

Our model is trained using PyTorch (Paszke et al. 2019), by using the pre-trained language model architectures provided by the transformers library [6]. The model is trained for 6 epochs, using a optimal learning rate of 0.0003, and AdamW as optimizer (Loshchilov and Hutter 2017). The learning rate scheduler used creates a schedule with a learning rate that decreases linearly from the initial learning rate set in the optimizer to 0. Each training batch has a total of 8 examples. Given the fact that some versions of these pre-trained language models are usually extremely big and require major computational resources to fine-tune and even produce inferences, we consider a smaller batch size for the large versions of the language models (this is explained later on the document). In particular, a gradient accumulation technique (i.e., running a configured number of steps without updating the model variables while accumulating the gradients of those steps and then using the accumulated gradients to compute the variable updates) is used to simulate the original batch size of 8. Additionally, we use the Healpy python library[7], based on the HEALPix scheme, to generate the regions over the Earth's surface.

The cost function of our model can be defined as a combination of the following loss functions:

- (1) HEALPix region classification. One can either consider the cross-entropy loss or the asymmetric loss between the predicted class probabilities and the one-hot true vector class (i.e., labels).

$$\begin{aligned} &\text{regionClassification}(outputs, labels) = \\ &\quad \text{crossEntropyLoss}(outputs, labels) \\ &\quad\quad\quad\quad\quad\quad\quad\quad \text{or} \\ &\text{regionClassification}(outputs, labels) = \\ &\quad \text{ASLSingleLabelLoss}(outputs, labels) \end{aligned} \tag{1}$$

- (2) Coordinates prediction. Great circle distance computed between the predicted coordinates and the ground-truth ones.

$$\begin{aligned} \text{greatCircleDistance}(\phi_1, \lambda_1, \phi_2, \lambda_2) = \sin\left(\frac{\phi_2 - \phi_1}{2}\right)^2 \\ + \cos\phi_1 \cos\phi_2 \sin\left(\frac{\lambda_2 - \lambda_1}{2}\right)^2 \end{aligned} \tag{2}$$

[2] https://thredds.daac.ornl.gov/thredds/catalog/ornldaac/967/historic_landcover_hdeg/catalog.html?dataset=967/historic_landcover_hdeg/historic_landcover_hd_1850.nc4

[3] https://globalmaps.github.io/glcnmo.html

[4] https://globalmaps.github.io/ptc.html

[5] https://globalmaps.github.io/el.html

[6] https://huggingface.co/transformers/

[7] https://pypi.org/project/healpy/

| Statistic | SpatialML | LGL | WOTR | SemEval-19 Task-12 Corpus | GeoVirus |
|---|---|---|---|---|---|
| Number of Docs | 428 | 588 | 1644 | 105 | 229 |
| Number of Toponyms | 4783 | 4462 | 10377 | 4659 | 2167 |
| Number of Distinct Toponyms | 825 | 1201 | 1970 | 950 | 685 |
| Average Number of Toponyms per Document | 11.18 | 7.59 | 6.31 | 44.37 | 9.46 |
| Number of Distinct HEALPix classes | 461 | 758 | 999 | 801 | 558 |

Table 1: Statistical characterization of the used corpora

Where $\phi_1$ and $\lambda_1$ represent the latitude and longitude values of the predicted coordinates, and $\phi_2$ and $\lambda_2$ the latitude and longitude values of the ground-truth coordinates.

- (3) Geophysical properties predictions. The mean absolute error is computed between the predicted values and the ground-truth values for the elevation and vegetation geophysical properties. Regarding the land coverage property, one can consider the cross-entropy loss between the predicted class probability and the one-hot true vector class.

$$\text{elevationLoss}(ePred, eLabel) = \text{MAE}(ePred, eLabel) \tag{3}$$

$$\text{vegetationLoss}(vPred, vLabel) = \text{MAE}(vPred, vLabel) \tag{4}$$

$$\text{landCoverageLoss}(lcPred, lcLabel) = \text{crossEntropyLoss}(lcPred, lcLabel) \tag{5}$$

It is worth mentioning that given the fact that each of the previously described loss functions produce different range values, the contribution of each loss function to the final combined loss is weighted (i.e., different weights to each loss function were tested, and the ones achieving the best results were kept). The following equation resumes the cost function of our model where $\gamma_n$ represent the result of the nth previously described equation..

$$\text{costFunction} = 1 * \gamma_1 + 0.005 * \gamma_2 + 0.1 * \gamma_3 + 0.1 * \gamma_4 + 0.01 * \gamma_5 \tag{6}$$

Model training therefore involves minimizing the combined loss functions associated to each of the outputs.

## 4 Datasets and Evaluation Methodology

This section describes the overview of the experiments conducted. Section 4.1 characterizes the corpora used throughout the experiments, and the evaluation methodology. Finally, Section 4.2 illustrates the experiments, and provides the results.

### 4.1 Experimental Methodology

We use four well-known public datasets: the War of the Rebellion (WOTR) (DeLozier et al. 2016) [8], the Local-Global Lexicon (LGL) (Lieberman, Samet, and Sankaranarayanan 2010)[9], the SpatialML (Mani et al. 2010) and the GeoVirus (Gritta, Pilehvar, and Collier 2018)[10]. These corpus have been target of intense study over the years in the area (Cardoso, Martins, and Estima 2019; Gritta, Pilehvar, and Collier 2018; DeLozier, Baldridge, and London 2015; Santos, Anastácio, and Martins 2015; DeLozier et al. 2016). Additionally, the training corpus used in the context of the SemEval-2019 Task-12 Challenge (Weissenbacher et al. 2019) [11] is also used in our experiments.

The SpatialML corpus is a subset of the ACE 2005 English SpatialML Annotations (Mani et al. 2010), available from the Linguistic Data Consortium. It contains documents that represent a variety of data sources, among which are broadcast news, magazine news, and web blogs. Each document is annotated using an XML-based language also called SpatialML, that allows the association of toponyms in the text with their respective locations and other geographically-relevant attributes. It should nonetheless be noted that,the SpatialML corpus is limited to the purpose of evaluating local toponyms, since news are usually directed to a much more geographically distributed audience. For that reason, Lieberman, Samet, and Sankaranarayanan (2010) presented the Local-Global Lexicon (LGL) corpus, composed of articles from a variety of smaller distributed geographic newspapers. This corpus was specifically created for challenging toponym resolution systems, as it contains highly ambiguous names, including small cities and local mentions.

DeLozier et al. (2016) proposed another corpus, in this case composed of annotated documents from a set of American Civil War archives, known as War of the Rebellion (WOTR). Some of these archives contained military reports and orders, and others contained historical correspondence. It was concluded that WOTR was the most challenging corpus at the time (i.e., end-to-end toponym resolution systems achieved lower performance than in other previous developed corpora, such as LGL and SpatialML), as it contained place names not in gazetteers and had a respectable size (i.e., it had roughly twice the number of toponyms than in previously developed corpora).

Aside from proposing a new method for toponym resolution, Gritta, Pilehvar, and Collier (2018) introduced GeoVirus, a dataset for the evaluation of geoparsing of news events covering global disease outbreaks and epidemics (Gritta, Pilehvar, and Collier 2018). Place names are manually tagged and assigned Wikipedia page URLs along with their global coordinates.

More recently, yet another corpus was introduced in the

[8] https://github.com/utcompling/WarOfTheRebellion

[9] https://github.com/milangritta/Pragmatic-Guide-to-Geoparsing-Evaluation

[10] https://github.com/milangritta/Geocoding-with-Map-Vector/tree/master/data

[11] https://competitions.codalab.org/competitions/19948

| Dataset | Mean dist. (km) ↓ | Median dist. (km) ↓ | Accuracy@161 km (%) ↑ |
|---|---|---|---|
| **SpatialML Corpus** | | | |
| Learning to Rank (Santos, Anastácio, and Martins 2015) | **140** | 28.71 | - |
| **Our model** | 206 | **9.08** | **90.9** |
| **LGL Corpus** | | | |
| MLG w/o Gaz (Kulkarni et al. 2021) | 1407 | - | 53.0 |
| MLG with Gaz (Kulkarni et al. 2021) | 620 | - | 73.0 |
| **Our model** | **193** | **12.52** | **85.0** |
| **WOTR Corpus** | | | |
| TopoCluster (DeLozier et al. 2016) | 604 | - | 57.0 |
| TopoClusterGaz (DeLozier et al. 2016) | 468 | - | 72.0 |
| Cardoso, Martins, and Estima (2019) model | 164 | 11.48 | 81.5 |
| **Our model** | **99** | **11.11** | **87.1** |
| **SemEval-2019 Task-12 Challenge corpus** | | | |
| **Our model** | **146** | **11.71** | **85.4** |
| **GeoVirus** | | | |
| MLG w/o Gaz (Kulkarni et al. 2021) | 1690 | - | 49 |
| MLG with Gaz (Kulkarni et al. 2021) | **276** | - | **85** |
| **Our model** | 720 | 180.80 | 49.8 |

Table 2: Experimental results with the base model

context of the SemEval-2019 Task-12 Challenge, where toponym resolution systems were evaluated on scientific articles. Weissenbacher et al. (2019) summarized the corpus used within this challenge. The corpus is composed of documents linked with different viruses, namely the Influenza A, B, and C virus, and the West Nile river virus. Additionally, some other documents associated with biomedical research articles were added to the corpus. The result was a dataset with fine-level toponyms that can be used to resolve name places in other scientific domains, particularly related to the domain of epidemiology.

We decided to simulate the conditions of the experiments developed by previous methods, thus making the comparisons as trustworthy as possible. In particular, we decided to keep the changes performed by Kulkarni et al. (2021) in the LGL and the GeoVirus corpora. As previously explained in Section 2, the authors decided to unify the true coordinates corresponding to the same target entity. Regarding the WOTR corpus, we chose to use the same data split presented by the authors (DeLozier et al. 2016). As far as the SpatialML is concerned, we decided to split the data in the following proportions: 90% of the instances for train and 10% for test. Finally, and regarding the corpus used in the SemEval-2019 Task-12 Challenge, we determined to merge the train and validation splits into one (Weissenbacher et al. 2019). Since the test data was not made publicly available, we chose to split the merged data in the following proportions: 90% of the instances for train and 10% for test.

Table 1 shows a statistical characterization of the main corpora used in the development of this project. Note that this statistical description was performed without a train-test data split, so these statistics refer to the whole data for each corpus. Additionally, the number of distinct HEALPix classes considered for each dataset is also presented.

To assess the performance of the developed system across the multiple datasets, the distance between the predicted coordinates and the ground-truth coordinates is computed, using Vincenty's formula (i.e., a well-known method for calculating geodesic distances between a pair of latitude/longitude points on an ellipsoidal model of the Earth). Having these values, some evaluation metrics can be computed, such as the mean and median error distances, and an accuracy (i.e., percentage of correct decisions) based on a given threshold over the distance. Previous studies have used 161 kilometers as the threshold distance value for the accuracy, so that value was also taken into consideration in this project.

## 4.2 Obtained Results

The architecture presented in Section 3.1 is the one used in our experiments. We decided to use two different pre-trained language models, namely the BERT (Devlin et al. 2018) and the RoBERTa (Liu et al. 2019). As briefly explained in Section 3.1, BERT is probably the most widely known Transformer-based language model. However, several new language models based on the BERT architecture, namely the RoBERTa model, have been producing prosperous results in several NLP tasks. Additionally, and besides testing two different models, we also test different versions of those models (i.e., a single pre-trained language model can have multiple versions that share the same general architecture but differ in the number of parameters and in the corpora used to train the model). Particularly, the versions used in our experiments are the BERT-base and the RoBERTa-large. In brief, the BERT-base model has 12 encoder layers stacked on top of each other whereas RoBERTa-large has 24 layers of encoders stacked on top of each other. As the number of layers is increased so does the number of parameters and the number of attention heads. BERT-base has a total of 12 attention heads and 110 million parameters. On the other hand, RoBERTa-large has 16 attention heads with 355 million parameters. BERT-base has 768 hidden layers while RoBERTa-large has 1024 hidden layers.

We test both the BERT-base and the RoBERTa-large with and without integrating geophysical properties, originating

| Model and Dataset | Mean dist. (km) ↓ | Median dist. (km) ↓ | Accuracy@161 km (%) ↑ |
|---|---|---|---|
| **SpatialML Corpus** | | | |
| BERT-base W/GeoProperties | 206 | **9.08** | 90.9 |
| BERT-base w/o GeoProperties | 205 | **9.08** | 90.6 |
| RoBERTa-large W/GeoProperties | **182** | **9.08** | **92.1** |
| RoBERTa-large w/o GeoProperties | **182** | **9.08** | 91.9 |
| **LGL Corpus** | | | |
| BERT-base W/GeoProperties | 193 | 12.52 | 85.0 |
| BERT-base w/o GeoProperties | 192 | 12.57 | 84.5 |
| RoBERTa-large W/GeoProperties | **171** | 12.26 | 87.8 |
| RoBERTa-large w/o GeoProperties | 172 | **12.24** | **88.9** |
| **WOTR Corpus** | | | |
| BERT-base W/GeoProperties | 99 | 11.11 | 87.1 |
| BERT-base w/o GeoProperties | 105 | 11.26 | 85.7 |
| RoBERTa-large W/GeoProperties | **74** | **10.99** | **88.5** |
| RoBERTa-large w/o GeoProperties | 81 | **10.99** | 88.0 |
| **SemEval-2019 Task-12 Challenge corpus** | | | |
| BERT-base W/GeoProperties | 146 | 11.71 | 85.4 |
| BERT-base w/o GeoProperties | 152 | 11.90 | 84.3 |
| RoBERTa-large W/GeoProperties | **110** | **11.12** | **89.1** |
| RoBERTa-large w/o GeoProperties | 116 | 11.15 | 87.8 |
| **GeoVirus** | | | |
| BERT-base W/GeoProperties | 720 | 180.80 | 49.8 |
| BERT-base w/o GeoProperties | 652 | 138.65 | 51.6 |
| RoBERTa-large W/GeoProperties | 621 | 32.21 | 59.9 |
| RoBERTa-large w/o GeoProperties | **583** | **29.52** | **61.1** |

Table 3: Experimental results with different modelling approaches

the following models: (1) BERT-base W/GeoProperties (this particular model will be addressed as the baseline) and BERT-base w/o GeoProperties, and (2) RoBERTa-large W/-GeoProperties and RoBERTa-large w/o GeoProperties.

Table 2 summarizes the results obtained by our base model (i.e., the BERT-base model considering GeoProperties), comparing them against the results reported on previous publications that have used the same datasets and the same evaluation metrics. We can verify that our base model achieves results that outperform those of previous methods achieving state-of-the-art results in several metrics. In particular, our model achieves best results on both the WOTR and the LGL datasets, outperforming the previous best results. Regarding the SpatialML corpus, the learning to rank system from Santos, Anastácio, and Martins (2015) achieves the smallest mean distance error, even though our model obtains a much smaller median distance error. As to the corpus used in the SemEval-2019 Task-12 Challenge we couldn't compare the results achieved by our model with the ones achieved by the teams competing in the contest, given the fact that we didn't have access to the test data. Finally, and regarding the GeoVirus dataset, we achieve worst results than the system that relies on gazetteer information proposed by Kulkarni et al. (2021). However, when compared to the system that does not rely on gazetteer information proposed by the same authors, our model obtain a much smaller mean distance error and slightly better accuracy. One possible explanation for this results is the fact that the GeoVirus dataset is mainly composed by large populated toponyms (i.e., most of the documents are concerned with locations with large populations), thus a system that relies on gazetteer information can somewhat benefit from this, by leaning to predict more common toponyms (i.e., when dealing with toponym disambiguation, a gazetteer-based system will likely opt to choose the larger toponym).

Additionally, it is also worth mentioning that the loss function computed over the class probability differs from corpus to corpus. In the case of the WOTR and the SemEval-19 Task-12 Corpus we use the asymmetric loss function, and in the rest of the corpora we decided to stick with the cross entropy loss function. This decision is based on the fact that both the WOTR and the corpus from the SemEval-19 Task-12 Corpus have the highest number of distinct HEALPix classes (see Table 1), thus could benefit from the use of the asymmetric loss function.

Table 3 describes the results obtained by our different modelling approaches. The results show that the large version of the RoBERTa language model obtains better results across all datasets (i.e., on average, less 42 kilometers of mean distance error, less 50 kilometers in the median error, and an increase of 3.74% on the accuracy@161) . Although the large version can lead to significantly better results, it also exponentially increases the training time, and the amount of computational resources needed to fine-tune the model. Thus, using or not a larger version of a pre-trained language model is a trade-off between better performance overall and less computational efficiency.

With respect to the experiments with geophysical infor-

| Corpus | Lowest distance error (km) | Highest distance error (km) |
|---|---|---|
| **WOTR** | Mexico (0.62)<br>Spring River (0.70)<br>Owen's Big Lake (1.08) | Shelter Cove (4016.65)<br>Fort Sheward (3956.01)<br>Gravely Ford (3179.59) |
| **LGL** | Iowans (1.16)<br>Pa. (2.30)<br>Pennsylvania (2.30) | Nigeria (9304.27)<br>Philippines (8344.62)<br>Vietnam (7674.42) |
| **SpatialML** | Tokyo (0.44)<br>Lusaka (2.38)<br>Basra (2.76) | Dunblane (15062.31)<br>Cayman Islands (12900.50)<br>British Colombia (7361.93) |
| **SemEval-2019 Task-12 Challenge corpus** | Dominican Republic (0.42)<br>Japan (0.63)<br>Stratford (0.983) | Topografov River (4584.17)<br>Antananarivo (3122.57)<br>United Arab Emirates (2399.63) |
| **GeoVirus** | United States (4.53)<br>China (6.07)<br>California (10.23) | New Zealand (16610.02)<br>Wisconsin (12859.72)<br>North America (8098.64) |

Table 4: Toponyms with the lowest and highest distance errors

mation, we record some inconsistencies in the results. We achieve worst results in the GeoVirus dataset, and slightly better results among the remaining corpora. In particular, we notice a bigger gap in the results on the WOTR dataset. This can probably be explained by the fact that the WOTR dataset is mainly composed of annotated documents from a set of American Civil War archives that contain a lot of geophysical information, such as terrain properties, water coverage, and resources, hence useful features when predicting the actual values of the geophysical properties. We can then conclude that the model indeed benefits from the addition of geophysical information, though not consistently.

Table 4 presents the place names with the lowest and highest distance error predictions for all corpora. There are a few conclusion one might retrieve from this statistical characterization, namely: (1) the model can identify different textual names as the same real location (e.g., Pa. and Pennsylvania both refer to the state in the United States), (2) the model can identify demonyms (i.e., a noun used to denote the natives or inhabitants of a particular country, state, city, etc.), such as Iowans (i.e., a native or inhabitant of the US state of Iowa), and predict the locations of the place itself, (3) there are a few small places that are among the locations with the lowest mean distance error, such as Owen's Big Lake or Spring River. These results show language model-based systems can effectively predict the geographical coordinates of some toponyms that would be poorly predicted by gazetteer-based approaches.

Finally, Table 5 presents some illustrative examples of toponym resolution together with the corresponding textual reference. Each of the examples has the document text with the place names references highlighted in red, and the image that show the real location of the place name (in red), and the corresponding predicted location (in blue). Additionally, the distance between the real and predicted locations is represented through a black line. In the examples shown, we illustrate two different behaviours of our model. In the first example, the distance between the predicted and the real locations is significant. In particular, the average distance error is 375 kilometers. Also, the first example has quite a few interesting toponyms such as *Peidmont* and *Peidmont Valley*, that correspond to the same real location and are predicted differently by our model. In the second example, the average distance error is much smaller (i.e., 18 kilometers), probably due to the presence of toponym co-ocurrence consecutively (i.e., *Fort Magruder, Va.*), which can give clues about both toponym locations.

## 5 Conclusions and Future Work

This article proposed a novel method using a Transformer-based pre-trained language model for the toponym resolution task, by considering multiple textual inputs, producing multiple outputs for classification and regression tasks. More specifically, the neural network predicts a probability distribution over HEALPix regions, and then uses this probability distribution to calculate the corresponding geographical coordinates of the toponym to be disambiguated. Additionally, we conducted several experiments, including the use of large versions of pre-trained language models, and adding geophysical properties retrieved from a raster dataset to guide the prediction of geographical coordinates.

The proposed method was tested on the WOTR, the LGL, and the SpatialML. Additionally, we also tested the model on scientific corpora such as the corpora used in the SemEval-2019 Task-12 Challenge, and the GeoVirus corpus. The results obtained confirm the superiority of the proposed model over previous methods that produced state-of-the-art results. The use of the large version of the RoBERTa pre-trained model produces better results than the non-large version of the model, even though it introduces a trade-off between performance and efficiency, given that the large version requires major computational resources. The incorpora-

tion of external geographical information into the model had a generally beneficial impact on the results, even though it also produced worst results in some experiments.

For future work, we would like to add a term to the cost function based on a contrastive function. The idea behind it is to preserve neighborhood relationships between data instances. More specifically, since the training will be done in mini-batches, the distance between the predicted coordinates among each instance in the batch will be compared to the distance between the ground-truth coordinates among each instance in the batch.

Additionally, it would be interesting to explore other techniques to partition the Earth into a set of discrete cells. Some approaches that were first introduced in the context of image geolocation that we could possibly analyze are:

- **Combinatorial Partitioning**: Hongsuck Seo et al. (2018) proposed an algorithm to tackle the trade-off between accuracy and overfitting by generating a large number of fine-grained classes by intersecting multiple coarse-grained cells, allowing a model to predict locations at a fine-scale while keeping several training examples per class.
- **Hierarchical Partitioning**: An algorithm that exploits hierarchical knowledge of multiple partitions was proposed by Muller-Budack, Pustu-Iren, and Ewerth (2018). This approach also uses the S2 library to generate sets of geoclasses, and applies an adaptive hierarchical subdivision, where each cell is the node of a quad-tree (i.e., a tree data structure in which each internal node has exactly four children).
- **Using an MvMF Output Layer**: Izbicki, Papalexakis, and Tsotras (2019) introduced a geolocation method that exploits the Earth's spherical geometry based on the *von Mises-Fisher* (vMF) distribution, which is one of the standard distributions in the field of directional statistics and that can be seen as the spherical analog of the Gaussian distribution.

## References


Ben-Baruch, E.; Ridnik, T.; Zamir, N.; Noy, A.; Friedman, I.; Protter, M.; and Zelnik-Manor, L. 2020. Asymmetric Loss for Multi-label Classification. *arXiv preprint arXiv:2009.14119*.

Cardoso, A. B.; Martins, B.; and Estima, J. 2019. Using Recurrent Neural Networks for Toponym Resolution in Text. In *Proceedings of the EPIA Conference on Artificial Intelligence*.

DeLozier, G.; Baldridge, J.; and London, L. 2015. Gazetteer-Independent Toponym Resolution using Geographic Word Profiles. In *Proceedings of the AAAI conference on Artificial Intelligence*.

DeLozier, G.; Wing, B.; Baldridge, J.; and Nesbit, S. 2016. Creating a Novel Geolocation Corpus from Historical Texts. In *Proceedings of the Linguistic Annotation Workshop held in conjunction with Association for Computational Linguistics*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.

Gorski, K. M.; Hivon, E.; Banday, A. J.; Wandelt, B. D.; Hansen, F. K.; Reinecke, M.; and Bartelmann, M. 2005. HEALPix: A Framework for High-resolution Discretization and Fast Analysis of Data Distributed on the Sphere. *The Astrophysical Journal, 622(2)*.

Gritta, M.; Pilehvar, M.; and Collier, N. 2018. Which Melbourne? Augmenting Geocoding with Maps. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Hongsuck Seo, P.; Weyand, T.; Sim, J.; and Han, B. 2018. CPlaNet: Enhancing Image Geolocalization by Combinatorial Partitioning of Maps. In *Proceedings of the European Conference on Computer Vision*.

Izbicki, M.; Papalexakis, E. E.; and Tsotras, V. J. 2019. Exploiting the Earth's Spherical Geometry to Geolocate Images. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*.

Kulkarni, S.; Jain, S.; Hosseini, M. J.; Baldridge, J.; Ie, E.; and Zhang, L. 2021. Multi-Level Gazetteer-Free Geocoding. In *Proceedings of Second International Combined Workshop on Spatial Language Understanding and Grounded Communication for Robotics*.

Leidner, J. L. 2007. Toponym Resolution in Text: Annotation, Evaluation and Applications of Spatial Grounding. In *ACM SIGIR Forum*.

Leidner, J. L.; Sinclair, G.; and Webber, B. 2003. Grounding spatial named entities for information extraction and question answering. In *Proceedings of the HLT-NAACL Workshop on Analysis of Geographic References*.

Lieberman, M. D.; and Samet, H. 2012. Adaptive Context Features for Toponym Resolution in Streaming News. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Lieberman, M. D.; Samet, H.; and Sankaranarayanan, J. 2010. Geotagging with Local Lexicons to Build Indexes for Textually-Specified Spatial Data. In *Proceedings of the IEEE International Conference on Data Engineering*.

Liu, Q.; Kusner, M. J.; and Blunsom, P. 2020. A Survey on Contextual Embeddings. *arXiv preprint arXiv:2003.07278*.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Loshchilov, I.; and Hutter, F. 2017. Decoupled Weight Decay Regularization. *arXiv preprint arXiv:1711.05101*.

Mani, I.; Doran, C.; Harris, D.; Hitzeman, J.; Quimby, R.; Richer, J.; Wellner, B.; Mardis, S.; and Clancy, S. 2010. SpatialML: annotation scheme, resources, and evaluation. *Language Resources and Evaluation, 44(3)*.

Muller-Budack, E.; Pustu-Iren, K.; and Ewerth, R. 2018. Geolocation Estimation of Photos using a Hierarchical Model and Scene Classification. In *Proceedings of the European Conference on Computer Vision*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in neural information processing systems*.

Radford, B. J. 2021. Regressing Location on Text for Probabilistic Geocoding. *arXiv preprint arXiv:2107.00080*.

Santos, J.; Anastácio, I.; and Martins, B. 2015. Using machine learning methods for disambiguating place references in textual documents. *GeoJournal, 80(3)*.

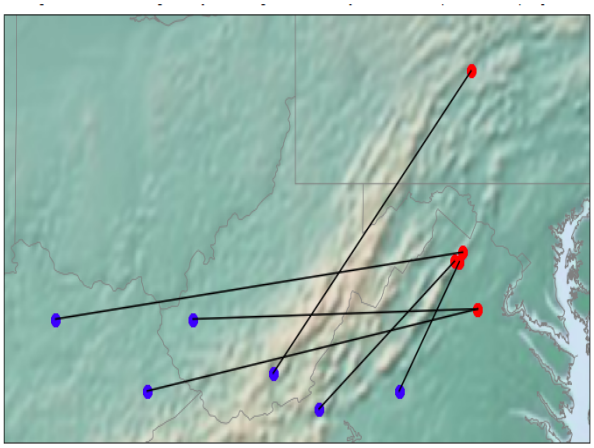
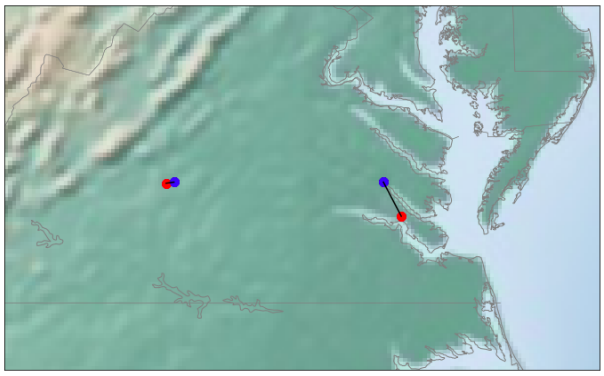
Wang, X.; Ma, C.; Zheng, H.; Liu, C.; Xie, P.; Li, L.; and Si, L. 2019. DM_NLP at SemEval-2019 Task 12: A Pipeline System for Toponym Resolution. In *Proceedings of the International Workshop on Semantic Evaluation*.

Weissenbacher, D.; Magge, A.; O'Connor, K.; Scotch, M.; and Gonzalez-Hernandez, G. 2019. SemEval-2019 Task 12: Toponym Resolution in Scientific Papers. In *Proceedings of the International Workshop on Semantic Evaluation*.

Wing, B. P.; et al. 2015. *Text-Based Document Geolocation and its Application to the Digital Humanities*. Ph.D. thesis, The University of Texas at Austin.

Zhou, X.; Liu, X.; Jiang, J.; Gao, X.; and Ji, X. 2021. Asymmetric Loss Functions for Learning with Noisy Labels. *arXiv preprint arXiv:2106.03110*.

Table 5: Illustrative examples

| | |
|---|---|
|  | I took the main column on the Peidmont. At that point I sent Captain Hart with 150 men of the First New Jersey Cavalry to pass through Peidmont Valley and stop at Paris until I arrived. With 100 men of the First Pennsylvania, under Captain McGregor, and 50 men of the Third Pennsylvania, under Captain Wetherill, I marched to Markham Station in Manassas Gap. From that point I crossed the mountains by a bypath, and joined the other parties at Paris at 12 o'clock on the day of the 18th. The column under Lieutenant Bradbury lost their way and came into Paris without passing through Upperville, and captured some horses and arms without seeing any of the enemy. The column under Captain Hart passed through Piedmont Valley, and surprised and captured 15 of Mosby's guerrillas and furloughed soldiers, and a quantity of arms, equipments, and horses. The other column with myself passed into Manassas Gap to Markham, and furloughed soldiers, and a quantity of arms, equipments, horses, and some medical stores. The latter we destroyed. As we came near Paris about 40 guerrillas charged on my rear guard. I sent a squadron and charged, scattering them. No casualties on our side. |
|  | GENERAL: The following dispatch, in cipher, just received from General Kilpatrick, dated Fort Magruder, Va., March 3, 1864: HEADQUARTERS CAVALRY EXPEDITION, March 3, 1864-9 p. m. Major General A. PLEASONTON, Commanding Cavalry Corps: I have reached General Butler's lines with my command in good order. HEADQUARTERS CAVALRY CORPS, March 4, 1864. GENERAL: The following dispatch, in cipher, just received from General Kilpatrick, dated Fort Magruder, Va., March 3, 1864: HEADQUARTERS CAVALRY EXPEDITION, March 3, 1864-9 p. m. Major General A. PLEASONTON, Commanding Cavalry Corps: I have reached General Butler's lines with my command in good order. I have failed to accomplish the great object of the expedition, but have destroyed the enemy's communications at various points on the Virginia Central Railroad; also the canal and mills along the James River, and much other valuable property. Drove the enemy into and through his fortifications to the suburbs of Richmond. |