Resolving Top-of-Hierarchy Locations First Improves Generate-and-Rank Toponym Disambiguation

Anonymous ACL submission

Abstract

Geocoding is the task of converting location mentions in text into structured geospatial data. We propose a new two-stage approach to geocoding that first resolves countries, states, and counties, and then uses these as document-level context to disambiguate the remaining location mentions. We apply this approach to two state-of-the-art geocoding models, CamCoder and SSPART. Our proposed two-stage approach to toponym resolution applied to SSPART yields state-of-the-art performance on multiple datasets. Our analysis shows that SSPART's direct incorporation of geographic database entries is key to its success over CamCoder in leveraging document context. Code and models are available at `https://<anonymized>`.

1 Introduction

Geocoding, also called toponym resolution or toponym disambiguation, is the task of linking place names in text (known as *toponyms*) to geospatial databases. It is a fundamental building block for natural language processing applications such as geographical document classification and retrieval (Bhargava et al., 2017), historical event analysis (Tateosian et al., 2017), tracking the evolution and emergence of infectious diseases (Hay et al., 2013), and disaster response mechanisms (Ashktorab et al., 2014; de Bruijn et al., 2018).

The goal of geocoding is, given a textual mention of a location, to choose the corresponding geospatial coordinates, geospatial polygon, or entry in a geospatial database. There are two kinds of challenges in geocoding: first, different geographical locations can be referred to by the same place name (e.g., *Edmonton* in Alberta, Canada vs. *Edmonton* in Queensland, Australia); second, different place names can refer to the same geographical location (e.g., *Tibet* and *Xizang* are two names for the same place in China).

Most existing geocoding systems utilize a variety of hand-engineered heuristics including lexical features (e.g., mention name, candidate entry name, and context window) and geographical features (e.g., population or type of place) (Speriosu and Baldridge, 2013; Zhang and Gelernter, 2014; DeLozier et al., 2015; Kamalloo and Rafiei, 2018; Wang et al., 2019). Recent deep learning based geocoding systems have yielded large improvements since neural networks can better extract contextual information with less feature engineering (Gritta et al., 2018; Cardoso et al., 2019; Kulkarni et al., 2021). However, deep learning systems have rarely used the *spatial minimality* feature common to prior work, which takes advantage of the fact that different toponyms in a document tend refer to spatially near locations. Incorporating this feature can be complex, since until toponym resolution is complete, we do not know the database entries for the locations and therefore do not know their coordinates to measure spatial distances.

Dataset	Models	Precision			
		Country	State	County	Other
LGL	CamCoder	0.943	0.898	0.529	0.477
	SSPART	0.968	0.806	0.829	0.745
GWN	CamCoder	1.000	0.565	0.156	0.302
	SSPART	1.000	0.765	0.778	0.752
TR-News	CamCoder	1.000	1.000	0.000	0.837
	SSPART	1.000	1.000	0.000	0.830

Table 1: Precision of two state-of-the-art geocoding systems on three geocoding development sets. See appendix A.3 for recall.

We propose a solution to this problem that takes advantage of the fact that current geocoding systems have good precision on locations at the top of the geographic hierarchy: countries, states, and counties (see Table 1 and Table 3). We therefore propose a new two-step architecture, shown in Figure 1, where these top-of-hierarchy locations are

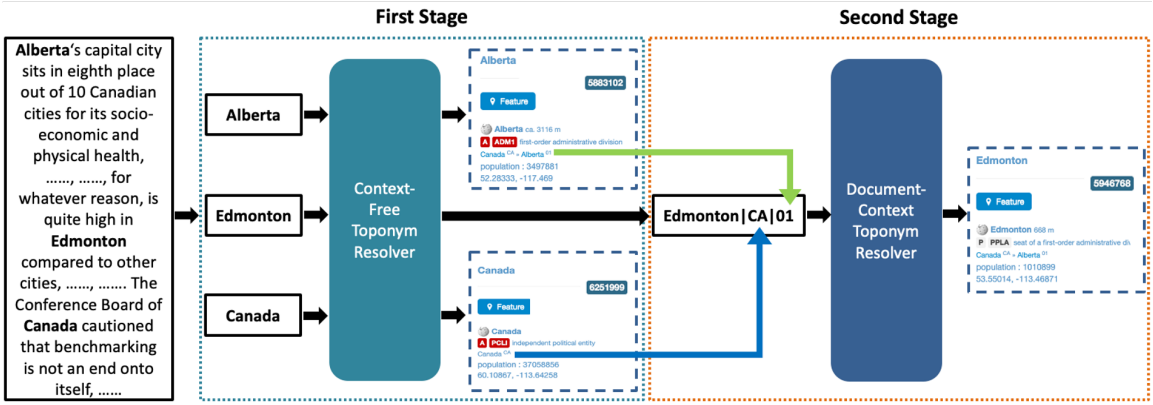

Figure 1: The architecture of our two-stage approach to toponym resolution.

resolved first and then used as context when resolving the remaining location names. Our work makes the following contributions:

- Our proposed architecture for geocoding achieves new state-of-the-art performance on multiple datasets.
- Our approach is the first neural architecture to incorporate document-level context for geocoding.
- We apply our approach to two different state-of-the-art geocoders and our analysis shows that SSPART's direct incorporation of geographic database entries is key to success.

2 Related Work

Our work focuses on mention-level geocoding in which the objective is to match phrases within a text to their corresponding locations. We do not address the separate named entity recognition task of geotagging, which typically precedes mention-level geocoding.

Many systems for geocoding used hand-crafted rules and heuristics to predict geospatial labels for place names. Examples include the Edinburgh geoparser (Grover et al., 2010), Tobin et al. (2010), Lieberman et al. (2010), Lieberman and Samet (2011), CLAVIN (Berico Technologies, 2012), GeoTxt (Karimzadeh et al., 2013), and Laparra and Bethard (2020). The most common features and heuristics were based on string matching, population count, and type of place (city, country, etc.).

As more shared tasks and annotated datasets were proposed, geocoding systems began to take the heuristics of rule-based systems and use them as features in supervised machine learning models, including logistic regression (WISTR, Speriosu and Baldridge, 2013), support vector machines (Martins et al., 2010; Zhang and Gelernter, 2014), random forests (MG, Freire et al., 2011; Lieberman and Samet, 2012), stacked LightGBMs (DM_NLP, Wang et al., 2019) and other statistical learning methods (Topocluster, DeLozier et al., 2015; CBH, SHS, Kamalloo and Rafiei, 2018).

Recently, deep learning methods have been introduced for toponym resolution (CamCoder, Gritta et al., 2018; Cardoso et al., 2019; MLG, Kulkarni et al., 2021). Each system has a unique neural architecture for combining inputs to make predictions based on convolutional neural networks (CNNs: CamCoder, Gritta et al., 2018; MLG, Kulkarni et al., 2020), recurrent neural networks (RNNs: Cardoso et al., 2019), vector-space models (Ardanuy et al., 2020), or pre-trained transformers (Anonymous, 2022).

Our proposed approach allows these deep learning systems to take advantage of document-level features, while respecting their limits on input size (e.g., 512 word-pieces). Our approach is broadly related to multi-stage approaches to Wikipedia entity linking (Guo and Barbosa, 2014; Xue et al., 2019; Yang et al., 2019), though these models assume an in-context example of every entry in the database, something that is available in Wikipedia, but unavailable in geographic databases like GeoNames.

3 Proposed Methods

We define the task of toponym resolution as follows. We are given an ontology or knowledge base with a set of entries $E = \{e_1, e_2, ..., e_{|E|}\}$. Each input is a text made up of sentences $T = \{t_1, t_2, \ldots, t_{|T|}\}$ and a list of location mentions $M = \{m_1, m_2, ..., m_{|M|}\}$ in the text. The goal is to find a mapping function $f(m_i, E) \rightarrow e_j$ that

Algorithm 1: Two-stage toponym resolution using document-level context.

Input: location mentions, M
GeoNames ontology, E
geocoding system, $f(m, c, E) \rightarrow e$
m is a location mention
c is a context string
$e \in E$ is the predicted entry
Output: mapping of mentions to entries, $\hat{R}$

1 $\hat{R} \leftarrow \{\}$
2 $C \leftarrow \emptyset$
3 **for** $m \in M$ **do**
4 $\quad e \leftarrow f(m, \texttt{""}, E)$
5 $\quad$ **if** TYPE$(e) \in \{\texttt{adm1}, \texttt{adm2}, \texttt{adm3}\}$ **then**
6 $\quad\quad \hat{R}[m] \leftarrow e$
7 $\quad\quad C \leftarrow C \cup \{\text{CODE}(e)\}$
8 **for** $m \in M$ **do**
9 $\quad$ **if** $m \notin \hat{R}$ **then**
10 $\quad\quad \hat{R}[m] \leftarrow f(m, \texttt{"|".join}(C), E)$
11 **return** $\hat{R}$

maps each location mention in the text to its corresponding entry in the ontology.

We propose to model $f(m_i, E)$ with Algorithm 1. Lines 1-7 are the context-free stage, where an existing geocoding system is first applied to all location mentions. If the feature type of a predicted entry, $\texttt{type}(e)$, is an administrative district 1-3 (i.e., the top of the geographic hierarchy: countries, states, or counties), then the prediction is accepted. Such predictions are also converted to their administrative codes (e.g., *United States* $\rightarrow$ `US`) and added to the context. Lines 8-10 are the second stage, where the geocoding system is applied to all remaining location mentions but this time incorporating the collected context.

4 Experiments

4.1 Datasets

We use the same three toponym resolution datasets and training/dev/testing splitting method as in previous work. Below we briefly describe each dataset and refer readers to their paper for details.

Local Global Lexicon (LGL; Lieberman et al., 2010) was constructed from 588 news articles from local and small U.S. news sources.

GeoWebNews (GWN; Gritta et al., 2019) was constructed from 200 articles from 200 globally distributed news sites.

TR-News (Kamalloo and Rafiei, 2018) was constructed from 118 articles from various global and local news sources.

4.2 Geospatial Database

Following previous work, we use **GeoNames** as our database. GeoNames is a crowdsourced database of geospatial locations. GeoNames contains almost 7 million entries and each entry contains a variety of geographical information such as coordinates (latitude and longitude), alternative names, feature type (country, city, river, mountain, etc.), population, elevation, and positions within a political geographic hierarchy. Three entry examples for *Alberta*, *Edmonton* and *Canada* from GeoNames are shown in fig. 1.

4.3 Evaluation Metrics

To evaluate the toponym resolution systems comprehensively, we adopt both database entry level metrics (more strict) and coordinate level metrics (less strict):

Accuracy measures the fraction of location mentions predicted with the correct database entry ID.

Accuracy@161km measures the fraction of predicted coordinates that were less than 161 km away from the gold coordinates.

Mean error distance calculates the mean over all distances between each predicted and gold coordinates.

Area Under the Curve (AUC) calculates the area under the curve of the distribution of geocoding error distances.

4.4 Systems

We compare two geocoding systems that allow programmatic manipulation of the context they consider. Both models incorporate local context, but their neural architectures do not have the capacity to operate over long, full document contexts.

CamCoder Gritta et al. (2018) combines a convolutional neural network over the target mention and its context with a population vector derived from location mentions in the context and populations from GeoNames. CamCoder considers only 400 tokens of context around the target mention, and predicts one of 7823 tiles of the earth's surface.

To apply our proposed two-stage resolution algorithm to CamCoder, for the first stage we run the most accurate model from table 1 (SSPART) without textual context. For the second stage, we collect any predicted countries, states or counties, and both

Model				LGL (test)				GeoWebNews (test)				TR-News (test)			
Name	Local Context?	Two-Stage?	Entry-Side-Codes?	Accuracy	Accuracy@161km	Mean Error	AUC	Accuracy	Accuracy@161km	Mean Error	AUC	Accuracy	Accuracy@161km	Mean Error	AUC
CamCoder	✓			.580	.651	82	.288	.572	.665	155	.290	.660	.778	89	.196
CamCoder (ours-partial)	✓	✓		.562	.638	83	.297	.553	.644	183	.307	.656	.774	88	.198
SSPART	✓			.759	.783	67	.166	.782	.832	60	.131	.777	.798	92	.166
SSPART (ours)		✓	✓	**.807**	**.824**	**46**	**.135**	**.828**	**.862**	**55**	**.114**	**.918**	**.933**	**34**	**.057**
SSPART (ours-partial)		✓		.723	.756	79	.193	.795	.834	56	.130	.848	.858	66	.114
SSPART				.760	.785	59	.167	.788	.834	61	.131	.798	.816	89	.154

Table 2: Performance on the test sets. Higher is better for Accuracy and Accuracy@161km. Lower is better for Mean Error and AUC.

concatenate them to the mention name (where CamCoder inserts textual context) and include them when building the MapVector. Though it would be useful to allow CamCoder to directly match the countries, states, and counties from the global context to the corresponding information of a candidate entry in the database, CamCoder's surface-tile formulation means it does not look at database entries at inference time. (See Appendix A.4 for implementation details.)

SSPART Anonymous (2022) uses Lucene search to generate candidate entries from the GeoNames database, sorts those candidates by population, and feeds the top candidates to a transformer-based reranker. The reranker considers the mention, its context, and the candidate entry. SSPART considers only three sentences of context around the target mention, and predicts one of GeoNames's 7 million database entries.

To apply our proposed two-stage resolution algorithm to SSPART, for the first stage we run the most accurate model from table 1 (SSPART) without textual context. For the second stage, we collect any predicted country, state, or county codes, and concatenate them to the mention name (where SSPART inserts textual context). Because SSPART compares mentions directly to candidate entries from the database, we also concatenate each candidate entry with its country, state, and county codes.

5 Results

We use the original code from the various authors and evaluate the CamCoder and SSPART models, both with and without our proposed two-stage resolution algorithm, on three public toponym resolution datasets. Table 2 shows that SSPART with our two-stage approach, "SSPART (ours)", achieves new state-of-the-art across all three datasets.

CamCoder, on the other hand, does not benefit from the two-stage approach. As noted above, CamCoder can incorporate the global country, state, and county context of the mention, but because it does not compare mentions to database entries at inference time, it cannot see the corresponding information for database entries. For this reason, we mark it as "ours-partial" rather than "ours" in table 2. To determine if this lack of country, state, and county information from the database entries is the reason for CamCoder's failure to benefit from global context, we ablated that information from SSPART. That is, we removed the country, state, and county codes from the candidate entry (while retaining the global context for the mention). The result is the "SSPART (ours-partial)" row in table 2. Similar to CamCoder, when the country, state, and county information is present in the global context but missing from the database entry, SSPART does not benefit from the two-stage approach.

This analysis suggests a clear benefit for geocoders that compare mentions to database entries over those that predict surface tiles: they can more easily take advantage of document-level context.

6 Conclusion

We propose a new two-stage toponym resolution architecture that first resolves locations at the top of the geographical hierarchy (countries, states, and counties) and uses those as context when resolving the other locations in the document. Our experiments show that applying this algorithm to the current best geocoder, SSPART, achieves new state-of-the-art performance on all our geocoding datasets.

References

Anonymous. 2022. Reranking with transformers improves toponym resolution. In *OpenReview*.

Mariona Coll Ardanuy, Kasra Hosseini, Katherine McDonough, Amrey Krause, Daniel van Strien, and Federico Nanni. 2020. A deep learning approach to geographical candidate selection through toponym matching. In *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, pages 385–388.

Zahra Ashktorab, Christopher Brown, Manojit Nandi, and Aron Culotta. 2014. Tweedr: Mining twitter to inform disaster response. In *ISCRAM*, pages 269–272.

Berico Technologies. 2012. Cartographic location and vicinity indexer (clavin).

Preeti Bhargava, Nemanja Spasojevic, and Guoning Hu. 2017. Lithium NLP: A system for rich information extraction from noisy user generated text on social media. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 131–139, Copenhagen, Denmark. Association for Computational Linguistics.

Ana Bárbara Cardoso, Bruno Martins, and Jacinto Estima. 2019. Using recurrent neural networks for toponym resolution in text. In *EPIA Conference on Artificial Intelligence*, pages 769–780. Springer.

Jens A de Bruijn, Hans de Moel, Brenden Jongman, Jurjen Wagemaker, and Jeroen CJH Aerts. 2018. Taggs: grouping tweets to improve global geoparsing for disaster response. *Journal of Geovisualization and Spatial Analysis*, 2(1):2.

Grant DeLozier, Jason Baldridge, and Loretta London. 2015. Gazetteer-independent toponym resolution using geographic word profiles. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2382–2388. AAAI Press.

Nuno Freire, José Borbinha, Pável Calado, and Bruno Martins. 2011. A metadata geoparsing system for place name recognition and resolution in metadata records. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, pages 339–348.

Milan Gritta, Mohammad Taher Pilehvar, and Nigel Collier. 2018. Which Melbourne? augmenting geocoding with maps. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1285–1296, Melbourne, Australia. Association for Computational Linguistics.

Milan Gritta, Mohammad Taher Pilehvar, and Nigel Collier. 2019. A pragmatic guide to geoparsing evaluation. *Language Resources and Evaluation*, pages 1–30.

Claire Grover, Richard Tobin, Kate Byrne, Matthew Woollard, James Reid, Stuart Dunn, and Julian Ball. 2010. Use of the edinburgh geoparser for georeferencing digitized historical collections. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 368(1925):3875–3889.

Zhaochen Guo and Denilson Barbosa. 2014. Robust entity linking via random walks. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '14, page 499–508, New York, NY, USA. Association for Computing Machinery.

Simon I Hay, Katherine E Battle, David M Pigott, David L Smith, Catherine L Moyes, Samir Bhatt, John S Brownstein, Nigel Collier, Monica F Myers, Dylan B George, et al. 2013. Global mapping of infectious disease. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1614):20120250.

Ehsan Kamalloo and Davood Rafiei. 2018. A coherent unsupervised model for toponym resolution. In *Proceedings of the 2018 World Wide Web Conference*, pages 1287–1296.

Morteza Karimzadeh, Wenyi Huang, Siddhartha Banerjee, Jan Oliver Wallgrün, Frank Hardisty, Scott Pezanowski, Prasenjit Mitra, and Alan M MacEachren. 2013. Geotxt: a web api to leverage place references in text. In *Proceedings of the 7th workshop on geographic information retrieval*, pages 72–73.

Sayali Kulkarni, Shailee Jain, Mohammad Javad Hosseini, Jason Baldridge, Eugene Ie, and Li Zhang. 2020. Spatial language representation with multi-level geocoding. *arXiv preprint arXiv:2008.09236*.

Sayali Kulkarni, Shailee Jain, Mohammad Javad Hosseini, Jason Baldridge, Eugene Ie, and Li Zhang. 2021. Multi-level gazetteer-free geocoding. In *Proceedings of Second International Combined Workshop on Spatial Language Understanding and Grounded Communication for Robotics*, pages 79–88, Online. Association for Computational Linguistics.

Egoitz Laparra and Steven Bethard. 2020. A dataset and evaluation framework for complex geographical description parsing. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 936–948, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Michael D Lieberman and Hanan Samet. 2011. Multifaceted toponym recognition for streaming news. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 843–852.

Michael D Lieberman and Hanan Samet. 2012. Adaptive context features for toponym resolution in streaming news. In *Proceedings of the 35th international*

ACM SIGIR conference on Research and development in information retrieval, pages 731–740.

Michael D Lieberman, Hanan Samet, and Jagan Sankaranarayanan. 2010. Geotagging with local lexicons to build indexes for textually-specified spatial data. In *2010 IEEE 26th international conference on data engineering (ICDE 2010)*, pages 201–212. IEEE.

Bruno Martins, Ivo Anastácio, and Pável Calado. 2010. A machine learning approach for resolving place references in text. In *Geospatial thinking*, pages 221–236. Springer.

Michael Speriosu and Jason Baldridge. 2013. Text-driven toponym resolution using indirect supervision. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1466–1476.

Laura Tateosian, Rachael Guenter, Yi-Peng Yang, and Jean Ristaino. 2017. Tracking 19th century late blight from archival documents using text analytics and geoparsing. In *Free and open source software for geospatial (FOSS4G) conference proceedings*, volume 17, page 17.

Richard Tobin, Claire Grover, Kate Byrne, James Reid, and Jo Walsh. 2010. Evaluation of georeferencing. In *proceedings of the 6th workshop on geographic information retrieval*, pages 1–8.

Xiaobin Wang, Chunping Ma, Huafei Zheng, Chu Liu, Pengjun Xie, Linlin Li, and Luo Si. 2019. Dm_nlp at semeval-2018 task 12: A pipeline system for toponym resolution. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 917–923.

Davy Weissenbacher, Arjun Magge, Karen O'Connor, Matthew Scotch, and Graciela Gonzalez-Hernandez. 2019. SemEval-2019 task 12: Toponym resolution in scientific papers. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 907–916, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Mengge Xue, Weiming Cai, Jinsong Su, Linfeng Song, Yubin Ge, Yubao Liu, and Bin Wang. 2019. Neural collective entity linking based on recurrent random walk network learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5327–5333. International Joint Conferences on Artificial Intelligence Organization.

Xiyuan Yang, Xiaotao Gu, Sheng Lin, Siliang Tang, Yueting Zhuang, Fei Wu, Zhigang Chen, Guoping Hu, and Xiang Ren. 2019. Learning dynamic context augmentation for global entity linking. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 271–281, Hong Kong, China. Association for Computational Linguistics.

Wei Zhang and Judith Gelernter. 2014. Geocoding location expressions in twitter messages: A preference learning method. *Journal of Spatial Information Science*, 2014(9):37–70.

A Appendix

A.1 Artifact intended use and coverage

The intended use of CamCoder and SSPART is matching English place names in text to the GeoNames ontology. We have used them for that purpose. The intended use of our two-step method is also matching English place names in text to the GeoNames ontology.

Though GeoNames covers millions of place names, our evaluation corpora cover only English news articles, and thus the performance we report is only predictive of performance in that domain.

A.2 Limitations

Our experiments are limited by the availability of models. Though we aimed to apply our two-stage method to several geocoding models, most published geocoding models have not released their code. We have thus applied our two-stage method to the two models that accept context as input and where code was available, CamCoder and SSPART.

Our experiments are also limited by the availability of datasets. Though we have attempted to collect a variety of geocoding datasets, some datasets, such as the SemEval-2019 Task 12 data (Weissenbacher et al., 2019), have not released test sets, making comparison to prior work difficult. We have thus applied our method to the three datasets where we were able to obtain the complete data: LGL, GeoWebNews, and TR-News.

Our two-step method has the same limitations as CamCoder and SSPART: their training and evaluation data covers only thousands of English toponyms from news articles, while there are many millions of toponyms across the world. It is likely that there are regional differences in our model's accuracy.

A.3 Recall of geocoding systems

Our proposed approach depends on high precision predictions for country, state, and county. But high precision with very low recall would also be problematic. Table 3 shows that this is fortunately not a problem for CamCoder or SSPART.

Dataset	Models	Recall			
		Country	State	County	Other
LGL	CamCoder	0.485	0.898	0.783	0.540
	SSPART	0.893	0.915	0.739	0.763
GWN	CamCoder	0.571	0.591	1.000	0.371
	SSPART	0.966	0.591	1.000	0.810
TR-News	CamCoder	0.800	1.000	0.000	0.766
	SSPART	1.000	1.000	0.000	0.830

Table 3: Recall of two state-of-the-art geocoding systems on three geocoding development sets.

A.4 CamCoder details

The original CamCoder code, when querying GeoNames to construct its input population vector from location mentions in the context, assumes it has been given canonical names for those locations. Since canonical names are not known before locations have been resolved to entries in the ontology, we have CamCoder use mention strings instead of canonical names for querying GeoNames.

We follow the hyperparameter settings in the original paper when training CamCoder: Keras 2.2.0, Tensorflow 1.8, Python 2.7, RMSprop optimizer, a learning rate of 1e-3, a batch size of 64, the context length of 200 and a number of epochs of 250. The total number of parameters in CamCoder is 178M and the training time is about 3 hours.

A.5 SSPART details

We follow the hyperparameter settings in the original paper when training SSPART: Adam optimizer, a learning rate of 1e-5, a maximum sequence length of 128 tokens, and a number of epochs of 30. When training without context, we use one Tesla V100 GPU with 32GB memory and a batch size of 8. When training with context, we use four Tesla V100 GPU with 32GB memory and a batch size of 32. The total number of parameters in SSPART is 168M and the training time is about 3 hours.