Salogni at GeoLingIt: Geolocalization by Fine-tuning BERT

Ilaria Salogni[1]

[1]*Università di Pisa, Italy*

Abstract

The recent growing interest in low-resource languages has been significantly bolstered by transformer-based models. By fine-tuning three such models, two based on BERT and the other on RoBERTa, I aim at geolocating sequences exhibiting non-standard language varieties relying solely on linguistic content. I find that, given that the information contained in the embeddings is all we need to carry out this complex task, a model architecture with less task-specific layers leads to better results. Furthermore, models pre-trained on miscellaneous corpora generalize better than those trained exclusively on tweets. The work also shows that the greater availability of resources of a certain regional variety positively affects the capacity of the model.

1. Introduction

Recognizing varieties and forming an opinion about where the speaker comes from is something so ingrained in our experience as speakers that it seems innate, and even a little magical. The question that drives this work is: can Large Language Models (LLMs) do what we do and if so, how well can they do it? do they do that in a way that is operationally similar to ours? The Italian scenario is a good testing ground as despite his limited geographical extent, it is one of the most linguistically-diverse in Europe. In their work, Ramponi and Casula say that *current transformer-based models are rather limited for modeling language variation over space in highly multilingual areas such as Italy* [1]. I don't agree completely, not only because of the encouraging results of the application of LLMs in a always growing number of tasks, but also because what we can explain on how they work does not highlight anything which may prevent good performance. Furthermore, the work of Lutsai and Lampert [2] reaches the astonishing result of a median error of 30km worldwide level, and fewer than 15 km on the US-level datasets for the models trained and evaluated on text features of tweets' content and meta data context, using a BERT model [3]. The fact that Twitter language identifier classifies with the label designed for standard Italian language also contents both partially and fully written in language varieties of Italy, as observed again by Ramponi and Casula [1] may suggest that the LLMs already have in their pre-training dataset the knowledge that they need to carry out a geolocalization task.

This document describes the model I submitted to the EVALITA 2023 evaluation campaign [4] for the task GeoLingIT [5].

1.1. Task

The goal of this project is to predict its location in terms of longitude and latitude coordinates (fine-grain geolocation) of tweets exhibiting non-standard language, based solely on linguistic content. This is a (double) regression task. In contrast to previous geolocation shared tasks on other areas ([6]; [7]; [8]), GeoLingIt is focused on Italy.

1.2. Dataset

GeoLingIT task data comprises 15K geotagged tweets that exhibit non-standard Italian language use (the content may be fully written in local language varieties or exhibiting code-switching with standard italian), and that have been collected in the corpus DiatopIt [1]. The data is annotated with latitude and longitude. After removing the emojis and tags, all the labeled data provided by the organizers were merged and then split into train-eval-test sets. Several crossvalidations were performed with 3-folds or 2-folds split, using train-eval sets. Target and output coordinate data were normalized using Min-Max scaling, as this understandably improved the quality of model prediction.

2. System description

Knowing that representations learned by transformer-based models achieve strong performance across many tasks with various datasets ([9], *inter alia*), I first decided to perform the fine-tuning of three different monolingual BERT-based [3] or RoBERTa-based [10] models, pre-trained on Italian texts. After picking the best performing model, I cross-validated it on a diverse set of hyperparameter configurations (e.g., number and size of hidden layers, activation functions) to pick the best task-specific architecture. All the runs were performed on Colab using high-RAM Nvidia A100 GPUs.

EVALITA 2023: 8th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Sep 7 – 8, Parma, IT

i.salogni@studenti.unipi.it (I. Salogni)

© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

Table 1
averaged MAE on the eval set for each model; batch size 50-100-150; 10 epochs; in bold the best result

	umberto-commoncrawl-cased-v1				bert-base-italian-cased				AlBERTo-it			
	fold1	fold2	fold3	avg	fold1	fold2	fold3	avg	fold1	fold2	fold3	avg
50	0.0147	0.0136	0.0144	**0.0142**	0.0184	0.0179	0.0184	0.0182	0.0216	0.0209	0.021	0.0211
100	0.0159	0.018	0.0176	0.0171	0.0176	0.0206	0.0191	0.0191	0.0232	0.0228	0.0235	0.0231
150	0.0168	0.0163	0.0173	0.0168	0.0181	0.019	0.0202	0.0191	0.0248	0.0239	0.0247	0.0244

Table 2
averaged MAE on the eval set; umberto-commoncrawl-cased-v1 in each architecture and configuration; batch size 50; 10 epochs; in bold the 4 best results, in red the 4 worse results

	No Hidden Layer				1 Hidden Layer (5)				1 Hidden Layer (300)		
	fold1	fold2	fold3	avg	fold1	fold2	fold3	avg	fold1	fold2	avg
Identity	0.0147	0.0136	0.0144	**0.0142**	0.0485	0.0179	0.0282	0.0315	0.0150	0.0145	**0.0147**
Sigmoid	/	/	/	/	0.0174	0.0138	0.0142	**0.0151**	0.0187	0.0151	0.0169
ReLU	/	/	/	/	0.2957	0.0143	0.1462	0.1520	0.0147	0.0152	**0.0149**
	3 Hidden Layers (5, 5, 10)				3 Hidden Layers (10, 5, 5)				3 Hidden Layers (300, 100, 100)		
	fold1	fold2	fold3	avg	fold1	fold2	fold3	avg	fold1	fold2	avg
Identity	0.0434	0.0517	0.0761	0.0570	0.024	0.0204	0.0412	0.0285	0.0228	0.0207	0.0217
Sigmoid	0.0347	0.0275	0.0395	0.0339	0.0229	0.0351	0.0272	0.0284	0.0144	0.0171	0.0157
ReLU	0.1819	0.0303	0.034	0.0820	0.3067	0.1171	0.1822	0.202	0.0784	0.1479	0.1131

2.1. Comparing different models

To assess how different pre-trained monolingual models perform on the given dataset, I fine-tuned the RoBERTa-based umberto-commoncrawl-cased-v1 [11] model, and the BERT-based models bert-base-italian-cased [12] and AlBERTo-it [13], adding a single linear layer with two output neurons to the pooling layer of each model, without activation function. I tested this "minimal" task-specific architecture on 3 batch values (50, 100, 150) for 10 epochs, dividing the train-dev set into 3 folds.

2.2. Adding a single hidden layer

To explore the potential benefits of introducing additional complexity to the model, I designed a new task-specific architecture adding a single hidden layer right after the pooling layer, testing different sizes (5 neurons and then 300 neurons) followed by an activation function (Identity, Sigmoid or ReLU), and finally, a two-neuron output layer. To reduce the computational cost, only umberto-commoncrawl-cased-v1 was tested using this and the next architectures. For the same reason batch size 50 was maintained.

2.3. Adding more hidden layers

Still with the rationale of knowing whether adding further complexity would enhance the model's learning capacity, I tested a task-specific setting with 3 hidden layers with neurons in combination (5, 5, 10) (10, 5, 5) and (300, 100, 100) on various activation functions (Identity, Sigmoid, ReLU) added before the two-neuron output layer.

3. Results

The umberto-commoncrawl-cased-v1 model on the "minimal" task specific architecture yielded the best Mean Absolut Error (MAE) results in 3-fold cross-validation using the provided labelled data, and achieved 128.19 km of avg distance in km on the blind test set provided by the challenge organizer. Although AlBERTo-it and bert-base-italian-cased were outperformed, their achieved results are not too distant, as shown in Table 1.

The second best MAE results were achieved using the 300 neurons single hidden layer task-specific architecture, and then followed by the 5 neurons single hidden layer architecture, as shown in Table 2. This can be explained by thinking that adding a small hidden layer after the pooling layer leads to an initial drastic reduction in the size of the model output.

The worst results, on the other hand, were all obtained with the 3 hidden layers architecture and ReLU as activation function. The accuracy dropped possibly because of excessive feature compression: when several hidden layers are stacked, this reduction is followed by another further reduction of the size of the input vector, and the linear activation function was of no use in this case. Therefore, further complicating the architecture requires an additional regularization effort, which the results achieved with only one hidden layer or even

without hidden layers show to be useless.

4. Discussion

From the coordinates of the target-output pairs obtained with the best-performing model (umberto-commoncrawl-cased-v1, "minimal" task-specific architecture, batch 50, 10 epochs), I calculated the error in km using the Haversine distance. From the error histogram we see how the average error (117.21 km) is not at all generalizable to all examples. In fact, 25% of the inputs were geolocated more than 154km away from the target.

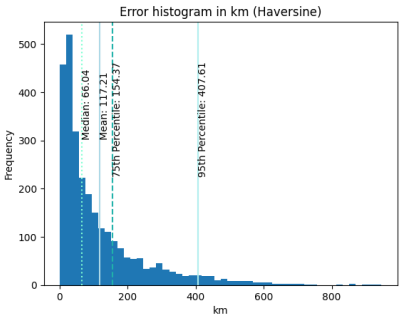

Figure 1: Error distribution of tweets; umberto-commoncrawl-cased-v1 "minimal" architecture; 50 bins

There are no specific areas where the inputs have a larger error. In contrast, inputs from areas in Piedmont-Lombardy-Veneto and Lazio-Campania have lower error than the others. In fact, two or three marked clusters can be observed in the scatterplots of the outputs (Figure 2), depending on the model configuration, the most persistent of which is between Lazio and Campania, then a cluster that follows the Alpine arc and finally less frequently by a cluster on Sicily. Excluding that this can be attributed to an imbalance in our fine-tuning dataset, this result must come from the representation of the embeddings of each model. Ramponi and Casula [1] argue on the fact that the pre-training material that had been used by those models may include content in language varieties of Italy, and they attribute it to the over-prediction of Italian of current language identifiers, observing that content both partially and fully written in language varieties of Italy is typically classified as standard italian by the Twitter language identifier. I can further hypothesize that the varieties from the areas with the smallest error are also quantitatively more present in the pre-training dataset of each model, as these are also the ones from the most densely populated areas in Italy.

However, it is very complex to reconnect these observations to one or more linguistic facts concerning the Italian regional varieties. The question then is how did we get this results, or even *"Does BERT make any sense?"* [14]. We defer the answer to later work.

Comparing the outputs' scatterplots (Figure 2) of the models umberto-commoncrawl-cased-v1, bert-base-italian-cased and AlBERTo-it fine-tuned on the provided dataset, we hope to probe in some way the embedding space generated by each model, and hopefully to gain insights into the quality and characteristics of the learned embeddings.

In the "minimal" architecture, AlBERTo-it shows a main cluster in the middle, from which all the other outputs radiate, while the other two models identify about two or three main clusters, like we said before, around the Alpine arc, between Lazio and Campania and the last, least marked, approximately on Sicily. Comparing the specifications of each model, we see that the most important differences are in the training dataset. While umberto-commoncrawl-cased-v1[1] and bert-base-italian-cased were pre-trained on a miscellaneous corpus, respectively OSCAR [15] Italian subcorpus and Wikipedia and OPUS corpus [16], AlBERTo-it was pre-trained on tweets. In this work, the models pre-trained on a miscellaneous corpus (umberto-commoncrawl-cased-v1 and bert-base-italian-cased) provided embeddings that performed better on tweets than a model pre-trained on a corpus specifically of the same genre. It is also worth noting that in Ramponi and Casula's work [1], the best performing model was AlBERTo-it, that has a vocabulary size of 128k, while in the current work the best performing ones are the 32k umberto-commoncrawl-cased-v1 and the 31k bert-base-italian-cased.

Adding a single large hidden layer makes fewer outliers compared to the models with the minimal architecture, and seems to be able to overcome at least partially the cluster of the Alpine arc and Sicily, while keeping the central one very compact. However it is necessary to remember that only umberto-commoncrawl-cased-v1 has been tested with a large hidden layer, and further experiments could be carried out.

5. Conclusions

The behavior shown by our models (need for regularization in the presence of numerous layers, better results with a single bigger hidden layer) is what we can expect from a simple neural network. However, it is astonishing that such a simple architecture manages to obtain non-disastrous results in an complex NLP task. The success of this regression task is undoubtedly attributable

[1]The OSCAR [15] subcorpus also has some subsets in other italian language varieties (such Piedmontese), but the official umberto-commoncrawl-cased-v1 model card says that it was pre-trained only on the Italian subcorpus, deduplicated.

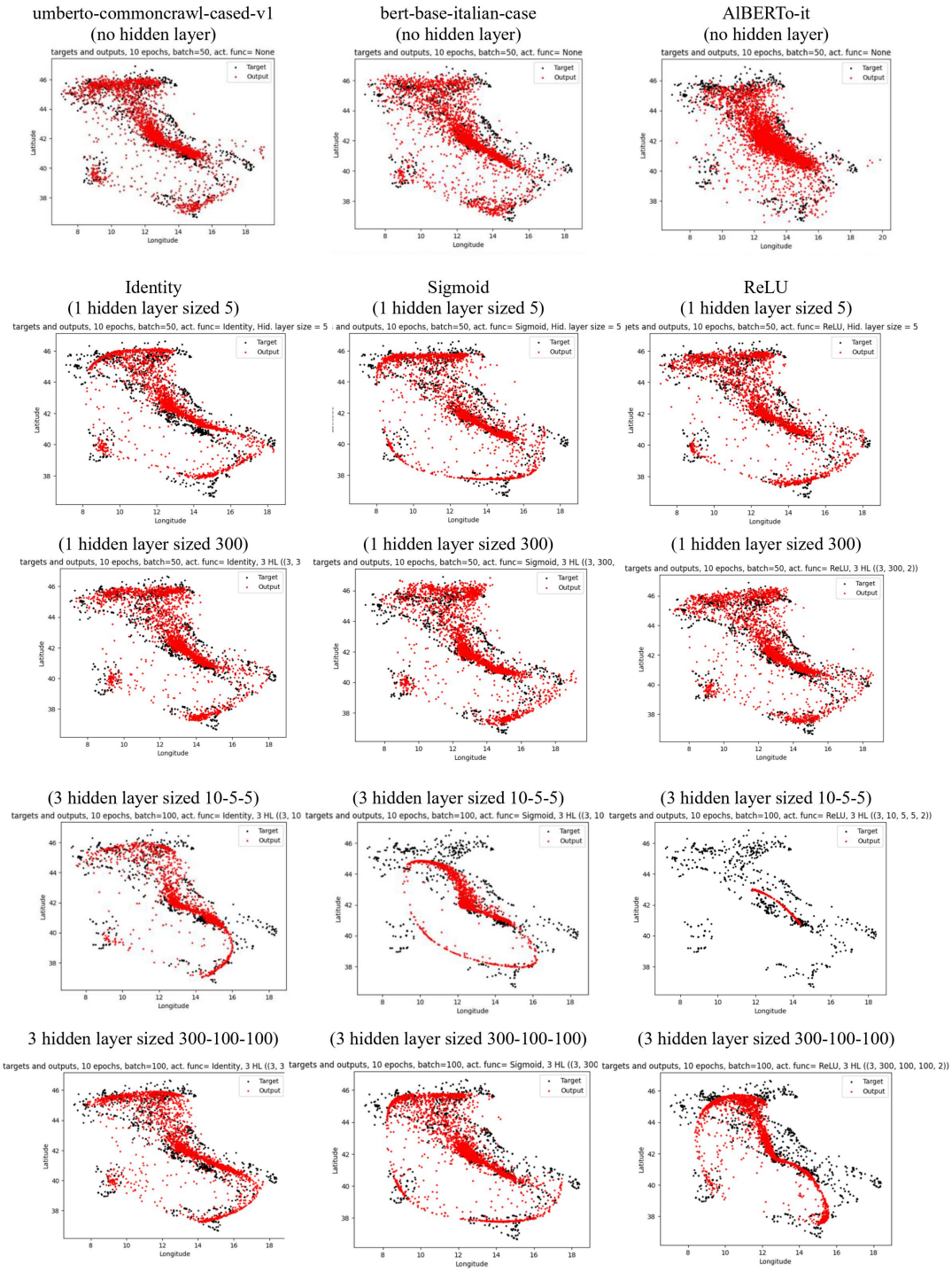

Figure 2: Scatterplots of the target (in black) and output (in red) coordinates for each configuration

to the high-level representations of the input data, together with BERT's ability to understand the linguistic context. Therefore, less is more: a simple setup, using even just two output neurons, seems to work better than a more complex one for BERT fine-tuned models on this task. Furthermore, in this work the models pre-trained on a miscellaneous corpus provides embeddings that performed better on tweets than a corpus specifically of the same genre. In conclusion, it is difficult to say how close we came to the goal, if the goal was to adequately map the diatopic variation of contemporary Italian, trying to automatically extract regional and dialectal patterns. Even if in this work we were unable to further probe the linguistic information used to carry out our task, the studies converge in believing that *BERT's structure is, however, linguistically founded, although perhaps in a way that is more nuanced than can be explained by layers alone* [17].

References

[1] A. Ramponi, C. Casula, DiatopIt: A corpus of social media posts for the study of diatopic language variation in Italy, in: Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023), Association for Computational Linguistics, Dubrovnik, Croatia, 2023, pp. 187–199. URL: https://aclanthology.org/2023.vardial-1.19.

[2] K. Lutsai, C. H. Lampert, Geolocation predicting of tweets using bert-based models, 2023. arXiv:2303.07865.

[3] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423. doi:10.18653/v1/N19-1423.

[4] M. Lai, S. Menini, M. Polignano, V. Russo, R. Sprugnoli, G. Venturi, Evalita 2023: Overview of the 8th evaluation campaign of natural language processing and speech tools for italian, in: Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR.org, Parma, Italy, 2023.

[5] A. Ramponi, C. Casula, GeoLingIt at EVALITA 2023: Overview of the geolocation of linguistic variation in Italy task, in: Proceedings of the Eighth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2023), CEUR.org, Parma, Italy, 2023.

[6] B. Han, A. Rahimi, L. Derczynski, T. Baldwin, Twitter geolocation prediction shared task of the 2016 workshop on noisy user-generated text, in: Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT), The COLING 2016 Organizing Committee, Osaka, Japan, 2016, pp. 213–217. URL: https://aclanthology.org/W16-3928.

[7] M. Gaman, D. Hovy, R. T. Ionescu, H. Jauhiainen, T. Jauhiainen, K. Lindén, N. Ljubešić, N. Partanen, C. Purschke, Y. Scherrer, M. Zampieri, A report on the VarDial evaluation campaign 2020, in: Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects, International Committee on Computational Linguistics (ICCL), Barcelona, Spain (Online), 2020, pp. 1–14. URL: https://aclanthology.org/2020.vardial-1.1.

[8] B. R. Chakravarthi, G. Mihaela, R. T. Ionescu, H. Jauhiainen, T. Jauhiainen, K. Lindén, N. Ljubešić, N. Partanen, R. Priyadharshini, C. Purschke, E. Rajagopal, Y. Scherrer, M. Zampieri, Findings of the VarDial evaluation campaign 2021, in: Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects, Association for Computational Linguistics, Kiyv, Ukraine, 2021, pp. 1–11. URL: https://aclanthology.org/2021.vardial-1.1.

[9] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, GLUE: A multi-task benchmark and analysis platform for natural language understanding, in: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 353–355. URL: https://aclanthology.org/W18-5446. doi:10.18653/v1/W18-5446.

[10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, 2019. arXiv:1907.11692.

[11] L. Parisi, S. Francia, P. Magnani, Umberto: an italian language model trained with whole word masking, https://github.com/musixmatchresearch/umberto, 2020.

[12] S. Schweter, Italian bert and electra models, 2020. URL: https://doi.org/10.5281/zenodo.4263142. doi:10.5281/zenodo.4263142.

[13] M. Polignano, P. Basile, M. de Gemmis, G. Semeraro, V. Basile, AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets, in: Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019), volume 2481, CEUR, 2019. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85074851349&partnerID=40&md5=7abed946e06f76b3825ae5e294ffac14.

[14] G. Wiedemann, S. Remus, A. Chawla, C. Biemann, Does BERT make any sense? interpretable word sense disambiguation with contextualized embeddings, CoRR abs/1909.10430 (2019). URL: http://arxiv.org/abs/1909.10430. arXiv:1909.10430.

[15] J. Abadji, P. Ortiz Suarez, L. Romary, B. Sagot, Towards a Cleaner Document-Oriented Multilingual Crawled Corpus, arXiv e-prints (2022) arXiv:2201.06642. arXiv:2201.06642.

[16] J. Tiedemann, S. Thottingal, OPUS-MT – building open translation services for the world, in: Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, European Association for Machine Translation, Lisboa, Portugal, 2020, pp. 479–480. URL: https://aclanthology.org/2020.eamt-1.61.

[17] J. Niu, W. Lu, G. Penn, Does BERT rediscover a classical NLP pipeline?, in: Proceedings of the 29th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 2022, pp. 3143–3153. URL: https://aclanthology.org/2022.coling-1.278.