Proceedings of the Second Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA 2022), pages 68–72
Language Resources and Evaluation Conference (LREC 2022), Marseille, 25 June 2022

BERToldo, the Historical BERT for Italian

Alessio Palmero Aprosio, Stefano Menini, Sara Tonelli
Fondazione Bruno Kessler
Trento, Italy
{aprosio,menini,satonelli}@fbk.eu

Abstract

Recent works in historical language processing have shown that transformer-based models can be successfully created using historical corpora, and that using them for analysing and classifying data from the past can be beneficial compared to standard transformer models. This has led to the creation of BERT-like models for different languages trained with digital repositories from the past. In this work we introduce the Italian version of historical BERT, which we call BERToldo. We evaluate the model on the task of PoS-tagging Dante Alighieri's works, considering not only the tagger performance but also the model size and the time needed to train it. We also address the problem of duplicated data, which is rather common for languages with a limited availability of historical corpora. We show that deduplication reduces training time without affecting performance. The model and its smaller versions are all made available to the research community.

Keywords: transformer models, Italian, historical data

1. Introduction

Recent advances in language modelling have shown that fine-tuning transformer-based models (Devlin et al., 2019) represent the state-of-the-art approach for several NLP tasks. As a consequence, specific BERT-like models have been created basically for any language for which enough training data are available. More recently, transformer models have been created also starting from historical corpora, showing that their adoption can benefit classification of historical texts in different tasks such as NER, word sense disambiguation and PoS-tagging (Manjavacas and Fonteyn, 2022). Historical BERTs have been developed first for English (Manjavacas and Fonteyn, 2021; Beelen et al., 2021), being a language with a large availability of historical corpora, but have included in the last year also other languages such as Dutch and French (Gabay et al., 2022).

As for Italian, there are no historical transformer models available. For this reason, we present in this work BERToldo, an Italian BERT trained on documents extracted from different freely-available repositories of historical corpora and covering a time period between 1200 and 1900. To evaluate the model, we fine-tune and test it on a PoS-tagged dataset containing texts written by the Italian poet Dante Alighieri (1265 – 1321), in order to measure its adaptation capabilities compared to standard BERT. We also split the training data into time periods and create smaller versions of BERToldo to assess what is the impact of training size and of the temporal dimension on the accuracy of the PoS-tagger on Dante's works. All versions of BERToldo are made available to the research community at `https://github.com/dhfbk/historical-bert`.

2. Related Work

The development of BERT-like models trained on historical data has been investigated only recently and has concerned so far few languages. The most represented one is English, for which some historical transformer models have been created following different approaches. A first strategy has been to further train a standard BERT model using historical data (Hosseini et al., 2021; Beelen et al., 2021). A second approach, instead, relies on training BERT from scratch using only historical data, which has led to the development of MacBERTh (Manjavacas and Fonteyn, 2021). The same authors have shown that this latter approach works better on a number of NLP tasks rather than fine-tuning standard BERT (Manjavacas and Fonteyn, 2022). Beside English, a historical version of BERT has been created also for French (Gabay et al., 2022), following the same approach as MacBERTh, i.e. training a RoBERTa-like model from scratch. Since no similar model has been developed for Italian, we create BERToldo in different versions, covering different time spans and using the two different training approaches described above.

3. Corpus Collection and Cleaning

BERToldo has been collected starting from two available digital repositories containing Italian texts belonging to various centuries.

- **Liberliber.it**[1] is a collection of more than 4,000 Italian books with different types of copyright. Most of them, whose authors have died more than 70 years ago, are released under the public do-

[1] `https://www.liberliber.it/benvenuto/`

main license (CC0).[2] In total, Liberliber.it contains around 230M tokens.

- **Wikisource**[3] is an online digital library operated by Wikimedia. The project contains works that are either in the public domain or freely licensed. Wikisource exists in 72 languages, and the Italian version includes more than 170,000 pages of content,[4] for a total of around 140M tokens.

While Wikisource is downloadable for free from the Wikimedia Downloads website,[5] Liberliber.it is freely available for navigation, but one needs to make a donation of 9.99 euros to download the entire resource.

Within BERToldo, we want to create models of Italian texts covering different periods in the history of language. To this purpose, after downloading the texts, we need to identify at least the century in which each document was written. This can be done easily in datasets where such information is structured and included as metadata. In both Liberliber.it and Wikisource, however, any additional information (such as publication year, author, translator, etc.) is only included along with the document itself, therefore all the documents need a pre-processing phase where the additional data are collected and removed from the file that is then used to train the BERT model.

The extraction of metadata information from the two repositories is error prone, since the years are often uncertain or written in a way that a machine could not easily understand (for example, "XVI secolo circa", *around XVI century*, or "tra il 1628 e il 1650", *between 1628 and 1650*). To deal with these problems, we build a list of common patterns in order to convert as many possible date expressions as possible, leaving only some tens of remaining cases to a manual check. During the conversion, we compute the date in the middle in case of time periods. For example, "XVI secolo circa" (*around XVI century*), becomes 1550, and "tra il 1628 e il 1650" (*between 1628 and 1650*), becomes 1639.

Sometimes, no year is found or the date associated with the document corresponds to the year of publication, that is often centuries after the actual date of the composition. To deal with these cases, we searched the author's name into WikiData[6] and get their biographical information.[7] If the year extracted from the document is not compatible with the lifespan of the author, we discard it and set a new one as the average between 20 years after the author's birth date and 5 years before their death.

3.1. Removing duplicates

As a last step, we deal with a relevant problem that to our knowledge was not addressed in other existing historical transformer models, i.e. duplicated content. Indeed, the amount of digital documents that can be used to create historical BERTs is limited compared to crawled content largely used for standard, contemporary transformers. For some languages, including Italian, the availability of digitised historical documents is so limited that the few digital repositories available tend to contain a large amount of overlapping documents. In fact, the *Project Gutenberg*[8] data, which are the third large repository of digital documents in Italian, were not included in BERToldo because the majority of the documents were already present in Wikisource or Liberliber.

Even between these two repositories (and also between documents in the same collection) we observed some overlaps, which are difficult to remove given that no curated list of metadata is released with the texts. We therefore have to first identify automatically existing duplicates and then remove them using the FuzzyWuzzy string matching Java library.[9] In particular, we use the token set ratio, performing a set operation that takes out the common tokens. Extra or same repeated words do not matter. Therefore, we obtain a very high value when one document is completely included in the other one (this happens often in our task, since sometimes one dataset contains a single poem, while the other one contains a collection of poems, including the first work considered).

Since fuzzy matching algorithms are often very slow, expecially on long texts, making a comparison all-versus-all is not feasible in a reasonable time. Removing duplicates makes sense since one can save time during training the language model without drop in performance (see Section 4), but if this filtering takes too much time, it becomes useless for the purpose.

For this reason, we first cluster the documents applying the fuzzy algorithm to authors' names. Documents where the author is not known or where our tool cannot extract it correctly are merged together in a single cluster. The clusterization is managed using JGraphT.[10]

[2] https://www.liberliber.it/online/opere/libri/licenze/

[3] https://it.wikisource.org/

[4] Usually longer works span on more than one page; for instance, "La Divina Commedia", written by Dante Alighieri, uses around 120 wiki pages.

[5] https://dumps.wikimedia.org/

[6] https://www.wikidata.org/

[7] In Wikisource, the author is often already linked to WikiData, resulting in a perfect match; in Liberliber.it, we searched the name into WikiData and use the information when it is not ambiguous, under the assumption that if a text is present in Liberliber.it then its author is famous enough to be present in WikiData.

[8] https://www.gutenberg.org/browse/languages/it

[9] https://github.com/xdrop/fuzzywuzzy

[10] https://jgrapht.org/

For each cluster, a complexity value is calculated, multiplying the number of documents considered for each dataset (Wikisource or Liberliber.it). When such complexity is very high, a sub-clustering is performed using the titles of the works.

In addition to fuzzy matching, authors and titles are also searched through WikiData, so to merge names expressed using different spellings (for example "Francesco Bacone", "Francis Bacon", and "Franciscus Baco", all referring to the same person).

The fuzzy matching between texts is then applied in an all-versus-all paradigm inside the single cluster, reducing drastically the execution time of the entire process (from tens of hours to tens of minutes).

We use the `TokenSetRatio` fuzzy algorithm, because it gives a high percentage matching on subsets: in fact, it often happens that a collection of works by a single author is considered as a single work in a dataset and a set of works in another (i.e., collection of poems). Since we want our computation to be fast, in case of very long works we restrict the comparison to the first 10,000 characters.

We consider two texts as overlapping when the similarity score between them is higher than 0.9 (1 meaning perfect match). We retain the longest text, just to avoid discarding useful data for our training.

After the filtering operation, the resulting dataset contains 304M tokens (starting from 410M, obtained after merging the two above-described resources).

4. Training BERToldo

The creation of the pre-trained model using the corpora presented in Section 3 follows as much as possible the BERT (uncased) architecture. Tokenization is performed by segmenting the input text data into subword units using the `BertWordPieceTokenizer`. As vocabulary size we keep 30,522 subword tokens. Pre-training is done with a maximum sequence length of 128 and a batch size of 64, while for the other parameters we keep the default values, following when possible the configuration adopted in MacBERTh (Manjavacas and Fonteyn, 2021).

We create eight versions of BERToldo. BERToldo$_{all}$ is the biggest one and was trained from scratch using all the documents from Wikisource and Liberliber after duplicate removal. BERToldo$_{till1500}$ was trained only on documents issued before 1500, BERToldo$_{1500\text{-}1700}$ was trained with documents from 1500 to 1700 and BERToldo$_{1700\text{-}1900}$ was created using documents pubblished between 1700 and 1900. For each of the splits, the model is trained for 10 epochs.

Following the approach presented in (Gururangan et al., 2020) and adopted also for the creation of historical BERT for English presented in (Hosseini et al., 2021), four additional models are created starting from standard Italian BERT-base uncased and further training it with additional historical data. BERT-base is available from Hugging Face and was created starting from a recent Wikipedia dump and various texts from the OPUS and OSCAR corpora collections.[11] The additional training process is carried out considering the three time slices independently, and then all together. Also in this case, the model is trained for 10 epochs. We summarise the different configurations in Table 1.

5. Evaluation

Similar to previous works (Manjavacas and Fonteyn, 2022; Gabay et al., 2022), we evaluate our BERToldo models on a task where historical language needs to be processed. By comparing the performance obtained with a standard BERT and a historical one, we can assess to what extent historical models can be beneficial. Unfortunately, the availability of historical datasets in Italian with some kind of manual annotation is very limited. Two exceptions are the corpus of Alcide De Gasperi's documents (Tonelli et al., 2019), a subset of which has been enriched with manual annotation of named entities and events, and the D(h)ante corpus (Basile and Sangati, 2016), containing annotated PoS-tags in CoNLL-like format. Since De Gasperi's documents were issued between 1920 and 1950, their language is not much different from contemporary Italian and it is not particularly necessary to use historical transformer models. We therefore perform our evaluation on Dante's texts.

We fine-tune the different BERT versions in Table 1 using the same test, development and train split of Dante's corpus used in (Basile and Sangati, 2016). For fine-tuning we use MaChAmp,[12] an extension of AllenNLP library that supports out-of-the-box a variety of standard NLP tasks (van der Goot et al., 2021).

The classification results of the different PoS-tagging models in terms of accuracy are reported in Table 2. We include in the last row also the classification result reported in (Basile and Sangati, 2016) and obtained using the same data splits with the Max-Ent Stanford Tagger included in Stanford CoreNLP version 3.5.2 (Toutanova et al., 2003).

These results provide interesting insights into the effectiveness of transformer models trained on historical data. First, (historical) BERTs are all better than the maximum entropy tagger. Among the transformers, further training BERT-base with historical data yields (slightly) better results than training BERT from scratch. This is in contrast with the findings in (Manjavacas and Fonteyn, 2022), showing the opposite on PoS-tagging English data. We will investigate in the future possible reasons behind this difference. Our results show also that BERToldo$_{till1500}$ performs worse than the models trained with more historical data, even if they were published centuries later than Dante's works. Overall, using BERT trained with a large amount of

[11] `https://huggingface.co/dbmdz/bert-base-italian-xxl-uncased`

[12] `https://github.com/machamp-nlp/machamp`

Dataset	Continue	Time (h)	Data	Tokens
BERToldo$_{till1500}$	No	42	127 MB	1,595,768
BERToldo$_{1500\text{-}1700}$	No	48	143 MB	1,784,656
BERToldo$_{1700\text{-}1900}$	No	214	700 MB	7,176,328
BERToldo$_{all}$	No	328	970 MB	10,556,752
Hugging Face BERT			81 GB	
ContBERToldo$_{till1500}$	Yes	+36	81 GB + 127 MB	+1,595,768
ContBERToldo$_{1500\text{-}1700}$	Yes	+40	81 GB + 143 MB	+1,784,656
ContBERToldo$_{1700\text{-}1900}$	Yes	+213	81 GB + 700 MB	+7,176,328
ContBERToldo$_{all}$	Yes	+300	81 GB + 970 MB	+10,556,752

Table 1: Training and size information for each BERToldo version. The models in the upper part of the table were trained from scratch on historical data, while those below (ContBERToldo) are the outcome of continuous training starting from Hugging Face BERT-base uncased. For this reason, the training time and the amount of documents for the ContBERToldo should be added to the ones needed to train BERT-base.

Dataset	Accuracy
BERToldo$_{till1500}$	0.939
BERToldo$_{1500\text{-}1700}$	0.937
BERToldo$_{1700\text{-}1900}$	0.951
BERToldo$_{all}$	0.955
Hugging Face BERT	0.952
ContBERToldo$_{till1500}$	0.960
ContBERToldo$_{1500\text{-}1700}$	0.958
ContBERToldo$_{1700\text{-}1900}$	0.958
ContBERToldo$_{all}$	**0.961**
Stanford CoreNLP	0.92

Table 2: BERToldo evaluation of Part-of-Speech tagging task on D(h)ante.

historical data performs better than using less data which are specific to the time period of the training and test sets. Continuous training with all historical data yields the best result, but if no standard BERT is available, comparable results can be obtained by training BERT from scratch with less than 1 GB of historical data.

As a comparison, we run the same experiments training BERToldo$_{till1500}$ and BERToldo$_{1500\text{-}1700}$ without duplicate removal described in Section 3.1. While the training time to build the BERToldo models increases by 19%, the accuracy of the Part-of-Speech tagger after fine tuning remains exactly the same. This demonstrates that removing duplicates makes BERT training less computationally expensive without a performance drop. The effects of duplication removal have been recently analysed also on large language models trained on contemporary corpora, confirming that deduplication should be generally encouraged (Lee et al., 2022).

6. Dataset and Models Release

We release all the BERT models and the source code on Github.[13]. We also release the data used to train BERToldo in an aggregated format divided into time periods. The dataset contaning the documents is distributed under the CC0 (public domain) license. The models are released under the Creative Commons Attribution 4.0 International (CC BY 4.0).[14] Finally, the source code (written in Java and Python) is free to use under the Apache License 2.0.

7. Conclusions

In this work we present BERToldo the first historical BERT for Italian. The model has been trained using freely-available documents published between 1200 and 1900. We plan to improve this first version of BERToldo by adding new historical documents as soon as they are available online. Another possible improvement could be performing language identification before creating the models, since we noticed that the split of documents published before 1500 contains some texts in Latin. We also plan to make our evaluation more robust by adding new tasks. This would be possible if diverse types of annotated data are created for Italian covering different time periods.

8. Acknowledgements

This research has been supported by the European Union's Horizon 2020 program project ODEUROPA under grant agreement number 101004469.

[13] https://github.com/dhfbk/historical-bert

[14] https://creativecommons.org/licenses/by/4.0/

9. References

Beelen, K., Nanni, F., Coll Ardanuy, M., Hosseini, K., Tolfo, G., and McGillivray, B. (2021). When time makes sense: A historically-aware approach to targeted sense disambiguation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2751–2761, Online, August. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). 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)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Gabay, S., Suarez, P. O., Bartz, A., Chagué, A., Bawden, R., Gambette, P., and Sagot, B. (2022). From freem to d'alembert: a large corpus and a language model for early modern french.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online, July. Association for Computational Linguistics.

Hosseini, K., Beelen, K., Colavizza, G., and Ardanuy, M. C. (2021). Neural language models for nineteenth-century english. *Journal of Open Humanities Data*, 7.

Lee, K., Ippolito, D., Nystrom, A., Zhang, C., Eck, D., Callison-Burch, C., and Carlini, N. (2022). Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland, May. Association for Computational Linguistics.

Manjavacas, E. and Fonteyn, L. (2021). Macberth: Development and evaluation of a historically pre-trained language model for english (1450-1950). In *Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH)*.

Manjavacas, E. and Fonteyn, L. (2022). Adapting vs Pre-training Language Models for Historical Languages. working paper or preprint, April.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

van der Goot, R., Üstün, A., Ramponi, A., Sharaf, I., and Plank, B. (2021). Massive choice, ample tasks (MaChAmp): A toolkit for multi-task learning in NLP. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 176–197, Online, April. Association for Computational Linguistics.

10. Language Resource References

Basile, Angelo and Sangati, Federico. (2016). *D(H)ante: A New Set of Tools for XIII Century Italian*. European Language Resources Association (ELRA).

Tonelli, S., Sprugnoli, R., and Moretti, G. (2019). Prendo la parola in questo consesso mondiale: A multi-genre 20th century corpus in the political domain. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019)*.