Are All Languages Equal? Curriculum Learning over **Different Languages**

Giulia Pucci¹, Leonardo Ranaldi^{1,2} and Fabio Massimo Zanzotto¹

¹University of Rome Tor Vergata

²Idiap Research Institute, Switzerland

Abstract

Curriculum Learning (CL) is emerging as a relevant technique to reduce the cost of pre-training Large Language Models. The idea, tested for the English language, is to train LLMs by organizing training examples from the simplest to the most complex. Complexity measures may depend on the specific language. Hence, this paper aims to investigate whether CL and the complexity measure can be easily exported to other languages. For this reason, we present a set of linguistically motivated measures to determine the complexity of examples, which has been used in English: these measures are based on text length, rarity, and comprehensibility. We then test the approach to two Romance languages: Italian and French. Our results show that the technique can be easily exported to languages other than English without adaptation.

Keywords

Efficient Pre-training, Multilingual LLMs, Natural Language Processing,

1. Introduction

Transformers-based models have disrupted natural language understanding methods outperforming previous methods and sometimes even humans in many tasks [1, 2, 3, 4]. Unsupervised learning on huge corpora, no matter the domain, seems to be the way to increase performance; however, besides the onerous costs, there are difficulties with the data.

Therefore, this results in a significant carbon footprint [5], contrary to global sustainability goals. There are many approaches to address the AI carbon footprint problem, ranging from using more carbon-efficient energy sources to applying efficient AI models and training algorithms. Indeed, Transformers seem to be only huge memories [6, 7] and, thus, better ways to train these models are necessary. Bengio et al. [8] in Curriculum Learning (CL) proposes a specific class of efficient training strategies for deep learning models.

The naïve approach for training Large Language Models involves feeding textual batches randomly sampled from the training corpora is re-visited in the CL, where the model is refined with a sequence of progressively more challenging examples [9]. This is motivated by and emulates how humans learn, starting with more straightforward concepts and gradually building up more complex ones. Soviany et al. [10] show that CL helps the model to perform better and converge faster.

CLiC-it 2023: 9th Italian Conference on Computational Linguistics, Nov 30 - Dec 02, 2023, Venice, Italy name.username@uniroma2.it (G. Pucci);

name.username@idiap.ch (L. Ranaldi); name.username@uniroma2.it (F. M. Zanzotto)





In this paper, we deeply analyze the learning divergencies training from scratch with BERT [11] and GPT2 [12] on the same corpus in multiple languages. Furthermore, following our CL-LRC metrics [13] based on length, rarity, and comprehensibility, computational costs are reduced, and the divergences are filled.

Hence, using the same small corpus in three different languages, English (original), Italian, and French (translated), experimental results show that loss values during the training vary in the different languages. Moreover, this difference seems to be softened in terms of perplexity scores when the pre-training block-sizes increase incrementally.

2. Background

Optimizing the use of computational resources to increase the learning capabilities of Large Language Models (LLMs) is a widely studied problem. The main approaches are based on architecture, learning, and, finally, data. Although current optimization methods at the architectural level have demonstrated extensive functionality on further fine-tuning, there still needs to be gaps at the pre-training level.

Clark et al. [14] propose a method for reducing computational costs by modifying the Masked Language Models with a discriminator, but it may have limitations in tasks that require a deep understanding of long-term dependencies or complex relationships between words. Sanh et al. [15] proposed parameter reduction techniques and obtained a lightweight version of BERT that is less compelling than the original in adapting parameters on specific tasks.

Finally, the last approach in vogue concerns the ef-



Table 1

Curriculum Learning and LRC pre-training overview.

ficient adjustment of parameters. Parameter-Efficient Tuning (PEFT) is an efficient technique for tuning a small portion of model parameters and freezing others. Standard techniques for PEFT: LoRA [16], Prefix Tuning [17], P-Tuning [18] reduce computational and storage and maintain the performance. However, these PEFT methods are applied to fine-tuning a model for a specific task and not to pre-training from scratch. While these topics have been extensively studied, the data-level approach has yet to be explored.

Many studies have found that the multi-headed selfattention mechanism requires tremendous computational effort. Since each head of this mechanism appears to be more attentive to local dependencies than global ones [19, 20, 21], training local self-attention in shorter blocks seems to be less complex than training global selfattention in more extended blocks. Nagatsuka et al. [9] proposed a Curriculum Learning (CL) strategy concentrating on hands-on self-attention mechanism training to enhance this aspect. They applied the strategy directly to BERT pre-training, manipulating the size of the input text block in the self-attention mechanism as a measure of difficulty.

Further the world of transformer-based models, many CL studies have used sentence length, external resources, or input sequences to measure difficulty in various NLP tasks such as in parsing tasks [22], reading comprehension [23], and concept masking for pre-training of the knowledge graph-related models [24].

In this paper, to solve the gap of LLMs in learning

English, Italian and French, we studied the difficulties faced in learning more languages. We propose text complexity techniques combined with input text block-size in the context of the self-attention mechanism. The two approaches measure the difficulty of pre-training two language models: BERT [11] and GPT2 [12]. Our proposal adds to the incremental CL brought in [9], an additional light step for calculating the pre-training text complexity. Our model performs better than the baselines and methods proposed in [9] regarding loss and perplexity.

3. Our Methods

Starting from the fact that language has a structure that varies between different languages, we searched for a strategy to alleviate these divergences [25, 26]. Hence organizing the examples during pre-training could improve the model's performance. Therefore, starting from the concept of Curriculum Learning (CL) shown by Bengio et al. [8], according to which learning algorithms perform better when the data are presented following the current competencies of the model, we used the methodology proposed in [9] applying an incremental learning technique on increasing block-sizes. We propose to use these techniques in different languages and extend the work done with a generative model. Finally, we study the impact of language complexity by intruding LRC, a measure used to determine the complexity of examples during pre-training before standard CL.

The application of the CL-LRC method consists of



Table 2

Examples of the complexity values produced by the metrics defined in Section 3.1.

three steps (Figure 1): (i) sorting the corpus according to our complexity measure starting from the least complex sentences to the most complex ones; (ii) partitioning the corpus according to input blocks of predefined sizes; (iii) stepwise pre-training by increasing the block size.

3.1. Complexity

The increasing block-size techniques and complexity measures are our method's core. While the dynamic resizing technique is fixed and does not change in different scenarios, the complexity of a text example is challenging to define.

Since the tasks used in pre-training should aim to learn language from context, precisely as humans do, organizing the complexity of examples could improve CL in LLMs.

We propose combining three factors: the number of tokens or sentence length, the repetitiveness or rarity of words in the corpus, and finally, the comprehensibility or, more commonly, the Flesch-Kincaid readability metric. Aggregating these three heuristics forms d_{LRC} , one of the foundational elements of our framework. Hence, we denote our training corpus as a collection of D sentences, $\{s_i\}_{i=0}^{D}$, where each sentence is a sequence of words denoted with $s_i = \{w_0^i, w_1^i, ..., w_n^i\}$.

Number of tokens The number of occurrences or sentence length is critical since longer sequences are more difficult to encode, as the possibility of them being cut is high. Therefore, longer sentences would be more prone to losing context during the pre-training tasks. We compute sentence length for each period s_i of our corpus D:

$$d_L(s_i) = length(s_i) \tag{1}$$

Following obtaining the $d_{L_{max}}$ and $d_{L_{min}},$ we normalize the values:

$$\hat{d}_L(s_i) = \frac{d_L(s_i) - d_{L_{min}}}{d_{L_{max}} - d_{L_{min}}}, \forall i \in [0, |D|].$$
(2)

Rarity The repetitiveness of words is a significant factor. We use the metric introduced in [27] where rarity is defined as the probability product of unigrams. This metric represents sentence information since the scores of longer sentences are the sum of more words and thus are likely to be more meaningful. Given a corpus of sentences, $\{s_i\}_{i=0}^{D}$, the complexity metric for word rarity is defined as:

$$d_R(s_i) \stackrel{\Delta}{=} -\sum_{k=1}^{N_i} \log p\left(w_k^i\right) \tag{3}$$

where we use logarithms of word probabilities. The component p(w) is defined as:

$$p(w) \stackrel{\Delta}{=} \frac{1}{N_{total}} \sum_{i=1}^{M} \sum_{k=1}^{N_i} \mathbb{1}_{w_k^i = w} \tag{4}$$

for each w unique word in a corpus and $\mathbb{1}_{condition}$, is the indicator function equal to 1 if its condition is satisfied or 0. We compute this value for each sentence s_i of our corpus D, obtaining the $d_{R_{max}}$ and $d_{R_{min}}$ and we normalize the values:

$$\hat{d}_R(s_i) = \frac{d_R(s_i) - d_{R_{min}}}{d_{R_{max}} - d_{R_{min}}}, \forall i \in [0, |D|].$$
(5)

Readability Metric Comprehensibility or, more commonly, readability may be related to the speed of perception, reflex blink technique, reading speed, reading fatigue, cognitively motivated characteristics, and word

	English		Italian		French	
Model	Loss	Perplexity	Loss	Perplexity	Loss	Perplexity
Baseline (BERT)	2.74	270.42	3.96	336.38	4.19	304.20
$Baseline_{LRC}$ (BERT)	2.53	254.23	4.06	330.21	4.38	296.71
Total-Curriculum (BERT)	2.56	250.64	3.83	324.73	4.06	300.18
$Curriculum_{LRC}$ (BERT)	2.26	245.348	3.86	304.70	3.46	287.16
Baseline (GPT2)	4.33	122.37	6.24	135.48	6.36	125.05
$Baseline_{LRC}$ (GPT2)	4.20	119.36	6.46	122.32	6.83	122.26
Total-Curriculum (GPT2)	3.97	117.29	6.32	120.97	6.09	124.63
$Curriculum_{LRC}$ (GPT2)	3.55	96.66	6.43	116.65	6.38	108.23

Table 3

Loss and Perplexity after Pre-training on the test set.

difficulty for a specific reader. Unfortunately, it is not always possible to collect these characteristics.

We used the Flesch-Kincaid metric [28] as an assessment tool for text comprehension. This metric is based on the length of sentences and words within a text by quantifying difficulty with a score. The lower the score, the easier it is to read and understand the text. We use the following formula:

$$d_C(s_i) = 0.39 \frac{avg(d_L(s_i))}{100} + 11.8 \frac{avg(d_L(w_i))}{100} - 15.59$$
(6)

where $avg(d_L(s_i))$ average sentence length is the number of words in a sentence divided by the number of sentences, and $avg(d_L(w_i)$ is the average word length, i.e., does the number of words divides the number of syllables per word. The value 0.39 is used to scale the effect of the average sentence length to compare it to the effect of the average word length, weighted by 11.8. The final score is then adjusted by subtracting the value of 15.59, which adjusts the score scale to match the grading levels used in education more closely. We calculate this value for each sentence s_i and obtain the maximum $d_{C_{max}}$ and the minimum $d_{C_{min}}$ scores. Finally, we normalize these values:

$$\hat{d}_C(s_i) = \frac{d_C(s_i) - d_{C_{min}}}{d_{C_{max}} - d_{C_{min}}}, \forall i \in [0, |D|].$$
(7)

3.2. Applying Complexity Heuristics

In the first phase, we compute the complexity of each sentence $d_{LRC}(s_i)$ by adding the normalized values of length $\hat{d}_L(s_i)$, rarity $\hat{d}_R(s_i)$, and readability score $\hat{d}_C(s_i)$, that is:

$$d_{LRC}(s_i) = \hat{d}_L(s_i) + \hat{d}_R(s_i) + \hat{d}_C(s_i)$$
(8)

Then, we sort the sentences of the original corpus by order of increasing complexity before the pre-training phase. Finally, we recompose the re-ordered corpus ready for pre-training.

3.3. Splitting a Corpus-Based on Block-sizes

Secondly, following the work of Nagatsuka et al. [9], we split the original corpora into training samples of the specified size. Each input text (block) for BERT and GPT2 pre-training should not be linguistically consistent as a sentence but a fixed interval of contiguous text. Thus, it is not guaranteed that the input is a period or begins with the first word of a sentence. Moreover, after extensive experiments, Liu et al. [29] argue that the input sequence should be at most 512 tokens. However, we follow an incremental approach that differs from the static sizing of 512 tokens per batch. The difference is the order, which is the reason why it could be easier for a Transformer to learn by order of complexity. We train a Byte-Pair Encoding (BPE) at the byte level [30] to split the raw text into a sequence of tokens. Byte-level BPE allows the decomposition of words, including words outside the vocabulary likely to appear during testing, especially when using a small training dataset. In the experiment, we set the vocabulary size to 20,000.

3.4. Gradual Training

Using the corpus sorted by complexity order, we train a step model with four block sizes, namely 64, 128, 256, and 512. At first, we train the model with the shortest block-size, 64, for an arbitrary number of steps. Then, we continue to train the model with block-sizes of 128 and 256, respectively, for the same number of steps. Finally, we finish with the largest block-size of 512.

4. Experimental Results and Discussion

We evaluated our proposed CL-LRC approach in model performance in the experiments. Therefore, we show that performances increase to the proposed state of the art in [9]. We use Wikitext-2 [31] to reproduce the results proposed. Hence, we perform the pre-training from scratch for BERT [11] and GPT2 [30]. Therefore, we investigated perplexity, loss, and learning curves during and at the end of the pre-training. All experiments were performed on two NVIDIA RTX A6000 with 48 GB of memory. The code and model will be released for further research.

4.1. Data

BERT and GPT2 are pre-trained with huge corpora, i.e., bookcorpus and Wikipedia-dump with about 3 billion words [32]. In this work, we used Wikitext-2 [31], a small corpus for simulations, allowing pre-training with a limited computational resource. Wikitext-2 is a standard language model corpus with 720 good-quality articles from English Wikipedia. In addition, we introduced two further corpora from the Italian and French translations of Wikitext-2.

4.2. Experimental setup

We use the same corpus in three different languages to analyze learning divergences between different languages. Hence, we perform pre-training from scratch with the baseline methods, and then with complexity metrics (*Baseline*_{LRC}), the Total-Curriculum (CL proposed in [9]), and our CL-LRC called $Curriculum_{LRC}$ using the settings proposed in [9]. In particular, in our $Curriculum_{LRC}$, we sort the corpora according to complexity, split the corpora according to the difficulty level of the training samples, and perform the pre-training phase by increasing the block size. We performed these steps for all corpora and pre-train BERT and GPT2 from scratch. Finally, we report the losses during learning, the final losses on the evaluation set, and the average perplexity of different cuts of the evaluation set.

4.3. Results

Difficulties in learning a language depend on the complexity of the language itself. However, it can be alleviated using curricular techniques and greatly improved using linguistically motivated methods, maintaining reduced training times as shown in Table 6. These conclusions derive from the pre-training results from scratch in three languages using Baseline, Total-Curriculum, and our CL-LRC techniques visible in Table 3. In Figure 5, it can be observed from the baselines of the different corpora that English language learners, on average, are less perplexed. Moreover, the $Curriculum_{LRC}$ outperforms the others in all corpora. However, the batch-size increase supports the performance achieved by Curriculum Learning. Finally, in Figure 4, learning curves explain the trade-off between pre-training steps and loss values.

4.3.1. Our Methods vs CL & Baseline

The linguistically motivated pre-training by our metrics has improved the technique proposed in [9] and outperformed the baseline models. In particular, $Curriculum_{LRC}$ (BERT) outperforms the version without LRC of 5 points for English and more than 30 points for Italian and French over perplexity scores. The same is true for GPT2 with less striking results (ranging from 16 to 4 points). Hence, this measure seems to have less impact on the Italian and French, as we can observe from *Baseline*_{LRC} models for English pre-training and others. Finally, in Fig. 5, we can observe a clear gap in perplexity in the presence of portions of text with a small number of tokens, which is reduced to zero or almost zero when the number of tokens is more significant.

4.3.2. Languages over Complexity

With the aim of studying intrinsic learning difficulties, we propose our line of experiments from the same corpus translated into three different languages: English (original), French, and Italian. We can observe that the models started from scratch have more difficulty learning the French and Italian corpora than the English ones. We believe this result's origin stems from the structure and complexity of the languages concerned. It is widely known that being both Romance languages, French and Italian have a very complex grammatical structure, very different from English. Regarding verb conjugation, while English verbs have relatively simple and regular conjugation patterns, French and Italian ones are very intricate, with various tenses, moods, aspects, and verb endings. For the agreement rules, unlike French and Italian, English has no grammatical gender distinction, so there is no agreement based on gender. Moreover, in contrast to the skinny use in English, French, and Italian have complex systems of clauses and subordination. Therefore, it is more difficult for a non-native speaker of Italian or French to learn these two languages from scratch, for the same reasons it is also for the models we tested.

4.4. Convergence Speed & Training time

Our CL-LRC outperforms the *Total-Curriculum* regarding loss during pre-training. However, in Figure 4, it can be seen that the loss of the basic model converges to around 50; in contrast, both models with curriculum steadily decrease and reach a higher convergence rate. Moreover, it can be observed that the loss of the curriculumbased model decreased steadily whenever the difficulty of the training samples was changed. Finally, in Table 6, it is possible to observe how curricular approaches can significantly reduce training time and consecutively consumption and costs.

5. Conclusion

In this paper, we explored the effectiveness of Curriculum Learning (CL) in reducing the cost of pre-training and increasing the results. We trained LLMs by organizing examples from the simplest to the most complex, thereby leveraging the concept of complexity measures. Hence, we pre-trained from scratch BERT and GPT2 using standard baselines and CL approaches. After deep analysis, we show that divergence in learning can be mitigated using CL approaches reinforced by measures to determine the complexity of examples. These measures, applied during pre-training to sort the corpus according to complexity, show outstanding results. While the original approach was tested and validated for the English language, this research aimed to investigate whether CL and its associated complexity measure could be applied to other languages without significant adaptation. Experiments conducted in a low-resource environment show that the proposed method leads to better performance in terms of loss during learning and perplexity on test data.

In future works, we will continue to propose pedagogically motivated mechanisms to analyze weaknesses [33] and empower Cross-lingual abilities to deliver multistepreasoning answers [34].

References

- 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.
- [2] L. Ranaldi, A. Nourbakhsh, E. S. Ruzzetti, A. Patrizi, D. Onorati, M. Mastromattei, F. Fallucchi, F. M. Zanzotto, The dark side of the language: Pretrained transformers in the DarkNet, in: R. Mitkov, G. Angelova (Eds.), Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing, INCOMA Ltd., Shoumen, Bulgaria, Varna, Bulgaria, 2023, pp. 949–960. URL: https://aclanthology.org/2023.ranlp-1.102.
- [3] L. Ranaldi, G. Pucci, Knowing knowledge: Epistemological study of knowledge in transformers, Applied Sciences 13 (2023). URL: https://www.md pi.com/2076-3417/13/2/677. doi:10.3390/app130 20677.
- [4] L. Ranaldi, E. S. Ruzzetti, F. M. Zanzotto, PreCog: Exploring the relation between memorization and performance in pre-trained language models, in:

R. Mitkov, G. Angelova (Eds.), Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing, INCOMA Ltd., Shoumen, Bulgaria, Varna, Bulgaria, 2023, pp. 961– 967. URL: https://aclanthology.org/2023.ranlp-1.1 03.

- [5] E. Strubell, A. Ganesh, A. McCallum, Energy and policy considerations for deep learning in NLP, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 3645–3650. URL: https://aclanthology.org /P19-1355. doi:10.18653/v1/P19-1355.
- [6] L. Ranaldi, A. Nourbakhsh, E. S. Ruzzetti, A. Patrizi, D. Onorati, F. Fallucchi, F. M. Zanzotto, The dark side of the language: Pre-trained transformers in the darknet, in: Proceedings of RANLP, 2023.
- [7] L. Ranaldi, E. S. Ruzzetti, F. M. Zanzotto, Precog: Exploring the relation between memorization and performance in pre-trained language models, in: Proceedings of RANLP, 2023.
- [8] Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in: Proceedings of the 26th annual international conference on machine learning, 2009, pp. 41–48.
- [9] K. Nagatsuka, C. Broni-Bediako, M. Atsumi, Pretraining a BERT with curriculum learning by increasing block-size of input text, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), IN-COMA Ltd., Held Online, 2021, pp. 989–996. URL: https://aclanthology.org/2021.ranlp-1.112.
- [10] P. Soviany, R. T. Ionescu, P. Rota, N. Sebe, Curriculum learning: A survey, 2022. arXiv:2101.10382.
- [11] 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.
- [12] A. Radford, K. Narasimhan, Improving language understanding by generative pre-training, 2018.
- [13] L. Ranaldi, G. Pucci, F. M. Zanzotto, Modeling easiness for training transformers with curriculum learning, in: R. Mitkov, G. Angelova (Eds.), Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing, INCOMA Ltd., Shoumen, Bulgaria, Varna, Bulgaria, 2023, pp. 937–948. URL: https://aclanthology.org/2 023.ranlp-1.101.

- [14] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, ELECTRA: Pre-training text encoders as discriminators rather than generators, in: ICLR, 2020. URL: https://openreview.net/pdf?id=r1xMH1BtvB.
- [15] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, ArXiv abs/1910.01108 (2019).
- [16] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-rank adaptation of large language models, in: International Conference on Learning Representations, 2022. URL: https://openreview.net/forum?id=nZeV KeeFYf9.
- [17] X. L. Li, P. Liang, Prefix-tuning: Optimizing continuous prompts for generation, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 4582–4597. URL: https://aclanthology.org/2021.acl-long.353. doi:10.18653/v1/2021.acl-long.353.
- [18] X. Liu, K. Ji, Y. Fu, W. L. Tam, Z. Du, Z. Yang, J. Tang, P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks, 2022. arXiv: 2110.07602.
- [19] O. Kovaleva, A. Romanov, A. Rogers, A. Rumshisky, Revealing the dark secrets of BERT, 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), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 4365–4374. URL: https://aclanthology.org/D19-144 5. doi:10.18653/v1/D19-1445.
- [20] S. Sukhbaatar, E. Grave, P. Bojanowski, A. Joulin, Adaptive attention span in transformers, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 331–335. URL: https://aclanthology.org/P19-1032. doi:10.18653/v1/P19-1032.
- [21] M. Podkorytov, D. Biś, X. Liu, How can the [mask] know? the sources and limitations of knowledge in bert, in: 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1–8. doi:10.1 109/IJCNN52387.2021.9534299.
- [22] V. I. Spitkovsky, H. Alshawi, D. Jurafsky, From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Los Angeles, California, 2010, pp. 751–759. URL: https://aclanthology.org/N10-1

116.

- [23] B. Xu, L. Zhang, Z. Mao, Q. Wang, H. Xie, Y. Zhang, Curriculum learning for natural language understanding, in: Annual Meeting of the Association for Computational Linguistics, 2020.
- [24] M. Lee, J.-H. Park, J. Kim, K.-M. Kim, S. Lee, Efficient pre-training of masked language model via concept-based curriculum masking, in: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 7417–7427. URL: https: //aclanthology.org/2022.emnlp-main.502.
- [25] F. M. Zanzotto, A. Santilli, L. Ranaldi, D. Onorati, P. Tommasino, F. Fallucchi, KERMIT: Complementing transformer architectures with encoders of explicit syntactic interpretations, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 256– 267. URL: https://aclanthology.org/2020.emnlp-mai n.18. doi:10.18653/v1/2020.emnlp-main.18.
- [26] L. Ranaldi, F. Fallucchi, F. M. Zanzotto, Dis-cover ai minds to preserve human knowledge, Future Internet 14 (2022). URL: https://www.mdpi.com/1 999-5903/14/1/10. doi:10.3390/fi14010010.
- [27] E. A. Platanios, O. Stretcu, G. Neubig, B. Poczos, T. Mitchell, Competence-based curriculum learning for neural machine translation, 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. 1162–1172. URL: https://aclanthology.org/N19 -1119. doi:10.18653/v1/N19-1119.
- [28] J. Talburt, The flesch index: An easily programmable readability analysis algorithm, in: Proceedings of the 4th Annual International Conference on Systems Documentation, SIGDOC '85, Association for Computing Machinery, New York, NY, USA, 1986, p. 114–122. URL: https://doi.org/10.114 5/10563.10583. doi:10.1145/10563.10583.
- [29] 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, ArXiv abs/1907.11692 (2019).
- [30] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, 2019.
- [31] S. Merity, C. Xiong, J. Bradbury, R. Socher, Pointer sentinel mixture models, ArXiv abs/1609.07843 (2017).
- [32] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, S. Fidler, Aligning books and

movies: Towards story-like visual explanations by watching movies and reading books, in: 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 19–27. doi:10.1109/ICCV.2015.11.

- [33] L. Ranaldi, F. M. Zanzotto, Hans, are you clever? clever hans effect analysis of neural systems, 2023. arXiv:2309.12481.
- [34] L. Ranaldi, F. M. Zanzotto, Empowering multi-step reasoning across languages via tree-of-thoughts, 2023. arXiv:2311.08097.

Appendix A



Table 4Loss during training phase.





Table 5Perplexity scores over different chunck of tokens of testset.

Appendix C

Model	Training Time (English)	Training Time (Italian)	Training Time (French)
Baseline (BERT)	5:22:33	5:41:11	5:52:33
$Baseline_{LRC}$ (BERT)	5:20:15	5:43:26	5:50:51
Total-Curriculum (BERT)	4:37:11	4:31:38	4:42:27
$Curriculum_{LRC}$ (BERT)	4:35:46	4:37:16	4:40:04
Baseline (GPT2)	6:37:21	6:42:28	6:58:13
Baseline $_{LRC}$ (GPT2)	6:37:21	6:44:09	7:02:51
Total-Curriculum (GPT2)	5:10:29	5:19:05	6:16:18
$Curriculum_{LRC}$ (GPT2)	5:06:46	5:20:16	6:09:16

Table 6

Statistics of the training time (hours) of the baseline and Curriculum Learning models.