State-of-the-art of post-OCR correction with Neural Machine Translation

Guillaume THOMAS

November 8, 2022

Contents

1 Introduction

1.1 Context

Optical Character Recognition (OCR) systems enable the extraction of text from digitalized documents, or more generally from images. The field of research around this technology has been active for over the past 30 years and achieves, now, remarkable results on mainstream documents. But those systems still terribly fail on historical documents. Due to their challenging layouts, their storage conditions, or the poor quality of their original printing materials, the extraction pipeline, therefore, produces a very noisy OCR output. This noisy extracted text can be, in the end, strongly diverging from the original text, known as

the Ground Truth (GT). This noise can hamper downstream Natural Language Processing (NLP) tasks, which makes difficult the indexation, access, and exploitation of such documents.

The ANR SODUCO project is interested in the social dynamics in the heart of Paris during the XVIII-XXth century. In order to analyze these dynamics, the project relies on scanned paper directories from which the text has been extracted. Even with specific OCR well-suited for historical documents [1]([9], [8], [10]), the text remains quite noisy. An OCR post-processing is then required to correct this noise. Post-OCR approaches are also well-suited in other scenarios in which OCRization is costly and cannot be performed again or if the digitization pipeline is reserved to process newly arrived documents.

1.2 Formalization of the problem

We can define the OCR-noise correction process as a Machine Translation task. We thus consider the OCR-noisy version of a text and its ground truth as two distinct languages. The model is then trained to "translate" from one to the other. A key difference from a language translation setup is that we rely on the same character and sub-word vocabularies for input and output instead of having two distinct vocabularies.

Machine Translation task has, historically, been performed through Statistical Machine Translation (SMT) system. But with the advent of neural networks in the last decade, the state-of-the-art approaches now rely exclusively on Neural Machine Translation (NMT) methods. We, therefore, limited the scope of this state-of-the-art to this sub-domain.

Regarding the specificities of our problem, the cost of annotation is significant. Our corpus contains a few hundred directories, for a few hundred thousand pages, containing altogether about some millions of lines. Our team manually annotated about 5.000 lines to establish a first Ground Truth. This Ground Truth is useful for evaluation, but its size is too limited for any form of supervised training. We would therefore prefer to process this corpus through a fully-automated approach.

1.3 Motivation of this work

Beyond the unsupervised need for our approach, we also want to explore two other important aspects. First, our corpus is composed exclusively of directories. Those contain a lot of addresses, and therefore a lot of numbers that are ultimately misrecognized by the OCR engine. On contrary to words, numbers can not be corrected based on their syntax or their semantics. How would you correct an OCR token "11" into its ground truth "14"? Or how would you even recover a number that has not been detected at all? To tackle this issue, we need something more advanced than classical NMT or more generally syntax correction. This problem requires an indexing system or at least a redundancy detection system combined with a consensus mechanism.

Secondly, in the translation context, Named Entities (like localization, name, or organization) are better translated when processed on their own, rather than with a generic NMT model [11]. In the context of "Named Entities Translation", a pre-processing with a Named Entities Recognition (NER) system is becoming mainstream. We believe that we can use the same approach in the OCR-correction tasks. In particular, our corpus, based on directories, has clear well-defined entities which our NER system successfully extracts (triplet of patronym, profession, and address). The noise-correction of specific Named Entities can benefit from a learned apriori knowledge about them. It has been shown that it is now possible with modern architecture to develop NER systems robust to OCR-noise [1]. In this study, we aim to experiment with post-OCR and post-NER correction to understand their differences.

[1] `https://github.com/DCGM/pero-ocr`

2 State-of-the-art of main approaches

2.1 Baselines

NMT approaches for OCR correction are broadly inspired by encoder-decoder architectures combined with an attention mechanism [3]. In the following section, we will review the simplest and most representative baselines of the domain. Those baselines can easily be used for comparison purposes. We will review, the supervised approach of the CCC Team from the ICDAR 2019 competition, which seems accessible to re-implement. We will then tackle two self-supervised approaches and discuss them.

2.1.1 CCC team from ICDAR 2019 Competition, Rigaud et al. 2019

[4] tried with the ICDAR competition to set a new standard to enable meaningful comparison in the domain of post-OCR correction. Most notably, they released publicly the dataset and the evaluation pipeline used in the competition. The CCC team, winner of the 2019 edition [14], therefore seems to be a perfect supervised baseline. Even though described in simple and clear terms, it might be hard to reproduce their result. They did not released any no open-source code and no other papers have been published about their implementation. Therefore, the only description of their method is contained within a short paragraph in the ICDAR 2019 paper.

The CCC team proposed a two steps approach: detection and correction. Their detection model exploits the pre-trained multilingual BERT, for its context awareness. The BERT output of each sub-token is then plugged into convolutional layers and fully-connected layers to be classified. The correction model is a character-level sequence-to-sequence model with the attention mechanism. The encoder is a bidirectional LSTM and the character embedding is shared between the encoder and the decoder. The encoder input is characters of erroneous tokens and corresponding context information from the BERT, fine-tuned at the detection phase. The decoder generates character-level corrections and the final correction of each erroneous token can be found by using beam search.

This approach, even though simple to set up, might not work as effectively on our dataset. Indeed, their training datasets are about 100 times bigger than ours, hence the need for an unsupervised approach. But regarding their performance in the competition, this approach still constitutes a good reference baseline. Their work inspired other subsequent research like [13].

2.1.2 Dong and Smith 2018

[6] presented a self-supervised method for OCR post-correction in which duplicated texts in large corpora are used as a source of noisy target outputs. Their method, with the help of a hand-crafted attention mechanism, can be applied to both single input or multiple inputs correction (in particular correction from duplicated sequences). Their work cut the CER and WER nearly in half on single inputs and, with the addition of multi-input decoding, can rival supervised methods. Their work is based on a character-level seq-to-seq model. They compared different combinations of the hidden states generated by encoding those multiple inputs and compared different approaches to select better target sequences for the decoding time in an unsupervised manner.

Though this approach fills the unsupervised constraints of our dataset, it requires redundant text. In their context, they used the same text document scanned multiple times or OCRed several times through different OCR engines as a redundancy source. Though our dataset contains redundancy of lines through indexes of directories or different directories, regrouping them would require a redundancy detection system.

2.1.3 Hämäläinen and Hengchen 2019

[7] proposed another fully self-supervised approach. Their encoder-decoder architecture is also founded on a character-level seq-2-seq model. It establishes a training target based on semantically similar words from

the errored OCR text. Those semantically-similar corrections, along with their lemma, are then checked for correctness in the Oxford English Dictionary. This correction is then used as a training target for each original queried word.

This approach might scale difficulty on big datasets, especially around the querying system which identifies semantically similar words among the whole dataset. Therefore, our dataset containing several gigabytes of data might not fit well with this approach, even though there are some interesting ideas to keep. Moreover, their evaluation protocol was only based on 200 words, which is quite below the standard evaluation data set size. This approach might, therefore, not work as well as expected. On the other hand, their source code is available on GitHub, which may make it an easy baseline to compare with.

2.2 Summary table of main models and training procedure used

Approaches	**Training's datasets**	**Training's target of the model**	**Model architecture**	**Degree of supervision**
Dong & Smith 2018 [6]	Internet Archive + Chronicling America collection of historic U.S.newspapers	Get dupplicated line + correct with an out-of-domain uniform error model + consensus among the dupplicate to establish the target	Bi-RNN encoder and simple RNN decoder	Self-supervised
Hämäläinen & Hengchen 2019 [7]	ECCO	Spelling candidates queried from the noisy text based on semantic similarity + check correctness based on Oxford English Dictionary	Bi-LSTM encoder and simple LSTM decoder	Self-supervised
Schaefer & Neudecker 2020 [15]	German Text Archive	Ground Truth	Bi-LSTM detection & LSTM encoder-decoder correction	Supervised
Nguyen et al 2020 [13]	ICDAR 2017 & 2019	Ground Truth	BERT's NER with a FC layer detection & with a RNN encoder-decoder correction	Supervised
Amrhein & Clematide 2018 [2]	ICDAR 2017	Ground Truth	LSTM encoder-decoder	Supervised
Mokhtar et al 2018 [12]	Private english, latin and german dataset	Ground Truth	LSTM encoder-decoder	Supervised
CCC, winner of ICDAR 2019 [4]	ICDAR 2019	Ground Truth	CNN and FC layer detection & Bi-LSTM encoder-decoder correction	Supervised
Todorov & Colavizza 2020 [16]	ICDAR 2019	Ground Truth	Bi-GRU encoder and GRU decoder	Supervised

2.3 Key take-aways

2.3.1 The state-of-the-art is largely based on character-level approaches

Historically, word-level models were used, but the current state-of-the-art is now exclusively focused on character-level one ([14], [4], [13], [2], [12], [6]). [12] benchmarked NMT models' performance at both word and character levels. Their conclusion is clear regarding the character-level approaches. They explained this difference with the following reasons:

- Word-level approaches can only correct a limited set of words, specifically, the one included in the training set. On the other hand, character-level approaches learn the common character mistakes and their corrections.
- As a consequence of the previous point, word-level approaches need significantly larger datasets to learn performant word correction.
- As a second consequence of the first point, character-level approaches can correct words that have not been seen before by correcting their individual characters.
- Lastly, character-level approaches can handle corrected sequence that does not have to be the same length as the erroneous one, and can therefore deal with missing and extra characters.

2.3.2 BERT pre-processing is on the rise

BERT is a multi-layer bidirectional transformer encoder [5]. It is pre-trained on unlabeled data over two different tasks, including Masked Language Model (MLM), and Next Sentence Prediction (NSP). BERT models can be fine-tuned to handle NLP problems. Across all the NLP tasks, BERT pre-processing becomes the norm as it can simply bring significant improvements. This assumption also holds for the post-OCR correction task. As evocated before, the CCC method, the winner of the ICDAR 2019 competition [14], was based on a fine-tuned BERT preprocessing in the error detection phase and used it as a context embedding for the correction phase. [13] pursued their research. They used a BERT-based NER fine-tuned for error correction. Its fine tuning was composed of a single Fully-Connected layer, while the CCC team, and more generally the state-of-the-art, combine it with a Fully-connected layer and a convolutional layer. Nguyen et al showed that the use of a BERT preprocessing in the context of the OCR detection task of ICDAR 2019 brought significant improvement (about 15-20% absolute improvement).

2.3.3 NMT models perform best on in-domain datasets

[2] came to this conclusion from two independent experimentations. First, they compared the performance of NMT models on four different datasets from two different languages (French and English) and two different document types (monograph and periodical). While tweaking hyperparameters and model settings, they struggled to observe constant improvements over all the datasets. They concluded that tuning supervised learning for OCR post-correction of texts from different sources, error distribution, text types, periods and languages is a difficult task: the data, on which the MT systems are trained, have a large influence on which methods and features work best. And therefore, conclusive and generally applicable insights or models usable in out-of-domain contexts is hard to achieve.

Second, they observed that NMT models' error correction performance mostly decreases when training them on several datasets altogether. They concluded that OCR errors are particular to a text type and that NMT approaches specifically adapt to the specific OCR error distribution (frequency and types) of a training set. This trend is even stronger when cross-linguistically combining larger training sets. This conclusion goes in the opposite direction of previous SMT approaches that seemed to always benefit from more data.

2.3.4 OCR post-correction becomes more difficult on low CER dataset

[15] initially experimented with the approach of [2] on their dataset. Their dataset has the particularity of an especially low original CER (about 1.1%). This particularity failed the tested approach as it increased the CER by over 30%. They conclude that NMT models perform well for data sets with a high amount of OCR errors. But the post-correction becomes more difficult when the CER is less severe.

To tackle this low-CER aspect, they experimented with a new pipeline composed of 2 steps: detection and then correction. This approach reduced their CER by 18%. They claimed that this success is due to two aspects:

- By decreasing the proportion of correct OCRed data, the CER of the data fed into the correction model is artificially increased, which is assumed to improve the translation results.
- By excluding the majority of correct sequences from the translation step, they avoid mistakenly insert errors in them.

3 Metrics & Datasets

3.1 Metrics

In the context of the post OCR-correction task, several metrics are commonly used. Here is a list of them:

3.1.1 Character Error Rate (CER)

It indicates the percentage of characters that were incorrectly predicted. The lower the value, the better the performance of the system with a CER of 0 being a perfect score. It has the downside to require an alignment of the characters between the corrected and the ground-truth sequence. The Character error rate can also be computed from the Levenshtein distance algorithm with the following formula:

$$\frac{substitution + deletion + insertion}{substitution + deletion + correct\ charaters} \tag{1}$$

It is one of the simplest metrics, both to use and to implement. It makes it especially useful for comparison with other methods. For our study, we will also use a variant of the CER: CER per named entity. It will highlight if some methods are especially powerful to correct errors on the number from the addresses, as this is a meaningful part of the error from our datasets.

3.1.2 Word Error Rate (WER)

Word error rate is a common metric of the performance of various NLP tasks, including machine translation systems. Even though it would be well suited for OCR correction, this metric is hard for us to use without proper word boundary segmentation. We chose to not include it in our experimentation.

3.1.3 Summed Levenshtein distances

Originally, the Levenshtein distance is a string metric for measuring the difference between two sequences. Informally, the Levenshtein distance between two words is the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one word into the other.

The ICDAR competition held in 2017 and 2019 used a metric derived from the Levenshtein distance as the main metric for the error correction task. The competition came along with a publicly available dataset and an evaluation tool. [4] tried with this competition to set a new standard to enable meaningful comparison in the domain.

The metric used was a weighted sum of the Levenshtein distances between the correction candidates list and the corresponding token in the Ground Truth. The weights used come from the confidence probability

from the correction system associated with each candidate from the candidates' list. This scope is useful when used in semi-automated scenarios (with human supervision to select the best correction candidate from a list, for example). This evaluation protocol can also be used in a fully-automated scenario. In this case, only the top-most candidate is used. The evaluation metric is therefore a simple summed Levenshtein distance. It is also worth pointing out that the competition used a specific format for the output of the model, rather than simply using fully corrected sequences. To prevent the alignment problem between the ground truth and the corrected sequences, they used an alternative approach. Only the position and the length of the erroneous token in the noisy OCR text are indicated along with its correction candidate. The evaluation tool is available online[2].

3.2 Datasets

Here is a list of a few datasets, previously evocated, that can be used for training or comparison purposes:

- Rigaud et al, ICDAR 2019, 22 million characters in 10 European languages, along with corresponding Ground Truth, `https://sites.google.com/view/icdar2017-postcorrectionocr`
- Hill and Hengchen (2019): Eighteenth Century Collections Online (ECCO), and its Ground Truth: ECCO-TCP (Text Creation Partnership), 180,000 titles (200,000 volumes) and more than 32 millions page, `https://www.gale.com/primary-sources/eighteenth-century-collections-online`
- "Chronicling America" collection of historic U.S., Two million newspaper issues in english from the 18th to the 20th century (no Ground Truth), `https://chroniclingamerica.loc.gov`
- Internet Archive, 3 million books in english since the begining of the 20th century (no Ground Truth), `https://archive.org/details/texts`
- German Text Archive, 35 German works that were published from the 17th to the 19th century, and their Ground Truth transcription, `http://www.deutschestextarchiv.de/`

4 Conclusion

We have seen several datasets and evaluation pipelines, most notably the one from the ICDAR 2019 Competition, along with several supervised and unsupervised baselines. Those can be easily used to compare any new approaches to the current state-of-the-art reliably. This state-of-the-art also illustrates some gaps that the upcoming research will hopefully fill.

References

[1] N. Abadie, E. Carlinet, J. Chazalon, and B. Duménieu. A benchmark of named entity recognition approaches in historical documents application to 19 th century french directories. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13237 LNCS:445–460, 2022.

[2] Chantal Amrhein and Simon; https://orcid.org/0000 0003-1365-0662 Clematide. Supervised ocr error detection and correction using statistical and neural machine translation methods. *Amrhein, Chantal; Clematide, Simon (2018). Supervised OCR Error Detection and Correction Using Statistical and Neural Machine Translation Methods. Journal for Language Technology and Computational Linguistics (JLCL), 33(1):49-76.*, 33:49–76, 2018.

[2] `https://git.univ-lr.fr/gchiro01/icdar2017`

[3] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 9 2014.

[4] Guillaume Chiron, Antoine Doucet, Mickael Coustaty, and Jean Philippe Moreux. Icdar2017 competition on post-ocr text correction. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 1:1423–1428, 7 2017.

[5] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1:4171–4186, 10 2018.

[6] Rui Dong and David A. Smith. Multi-input attention for unsupervised ocr correction. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 1:2363–2372, 2018.

[7] Mika Hämäläinen and Simon Hengchen. From the paft to the fiiture: a fully automatic nmt and word embeddings method for ocr post-correction. *International Conference Recent Advances in Natural Language Processing, RANLP*, 2019-September:431–436, 10 2019.

[8] Martin Kišš, Karel Beneš, and Michal Hradiš. At-st: Self-training adaptation strategy for ocr in domains with limited transcriptions. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12824 LNCS:463–477, 2021.

[9] Oldřich Kodym and Michal Hradiš. Page layout analysis system for unconstrained historic documents. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12822 LNCS:492–506, 2021.

[10] Jan Kohút and Michal Hradiš. Ts-net: Ocr trained to switch between text transcription styles. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12824 LNCS:478–493, 2021.

[11] Maciej Modrzejewski, Thanh-Le Ha, Alexander Waibel, Miriam Exel, and Bianka Buschbeck. Incorporating external annotation to improve named entity translation in nmt. pages 45–51, 2020.

[12] Kareem Mokhtar, Syed Saqib Bukhari, and Andreas Dengel. Ocr error correction: State-of-the-art vs an nmt-based approach. *Proceedings - 13th IAPR International Workshop on Document Analysis Systems, DAS 2018*, pages 429–434, 6 2018.

[13] Thi Tuyet Hai Nguyen, Adam Jatowt, Nhu Van Nguyen, Mickael Coustaty, and Antoine Doucet. Neural machine translation with bert for post-ocr error detection and correction. *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, pages 333–336, 8 2020.

[14] Christophe Rigaud, Antoine Doucet, Mickael Coustaty, and Jean Philippe Moreux. Icdar 2019 competition on post-ocr text correction. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pages 1588–1593, 9 2019.

[15] Schaefer Robin and Clemens Neudecker. A two-step approach for automatic ocr post-correction. pages 52–57, 2020.

[16] K. Todorov and G. Colavizza. Transfer learning for historical corpora: An assessment on post-ocr correction and named entity recognition, 2020.

(2)