Correcting Whitespace Errors in Digitized Historical Texts

Sandeep Soni and **Lauren F. Klein** and **Jacob Eisenstein**
Georgia Institute of Technology
sandeepsoni@gatech.edu lauren.klein@lmc.gatech.edu jacobe@gmail.com

Abstract

Whitespace errors are common to digitized archives. This paper describes a lightweight unsupervised technique for recovering the original whitespace. Our approach is based on count statistics from Google n-grams, which are converted into a likelihood ratio test computed from interpolated trigram and bigram probabilities. To evaluate this approach, we annotate a small corpus of whitespace errors in a digitized corpus of newspapers from the 19th century United States. Our technique identifies and corrects most whitespace errors while introducing a minimal amount of oversegmentation: it achieves 77% recall at a false positive rate of less than 1%, and 91% recall at a false positive rate of less than 3%.

1 Introduction

The application of natural language processing to digitized archives has the potential for significant impact in the humanities. However, to realize this potential, it is necessary to ensure that digitization produces accurate representations of the original texts. Most large-scale digital corpora are produced by optical character recognition (OCR; e.g., Smith, 2007), but even the best current methods yield substantial amounts of noise when applied to historical texts, such as the nineteenth-century newspaper shown in Figure 1. Alternatively, with substantial effort, digitization can be performed manually, or by manual correction of OCR output (Tanner et al., 2009). However, even for manually "keyed-in" corpora, noise can be introduced due to errors in workflow (Haaf et al., 2013).

Whitespace is a particularly common source of digitization errors in both OCR and manually digitized corpora. Such errors, also known as *word segmentation errors* or *spacing errors*, can arise during OCR as well as during the post-digitization handling of the data (Kissos and Dershowitz, 2016). These errors can result in the elimination of whitespace between words, leading to out-of-vocabulary items like *senatoradmits* and *endowedwith*. This paper presents a set of unsupervised techniques for the identification and correction of such errors.

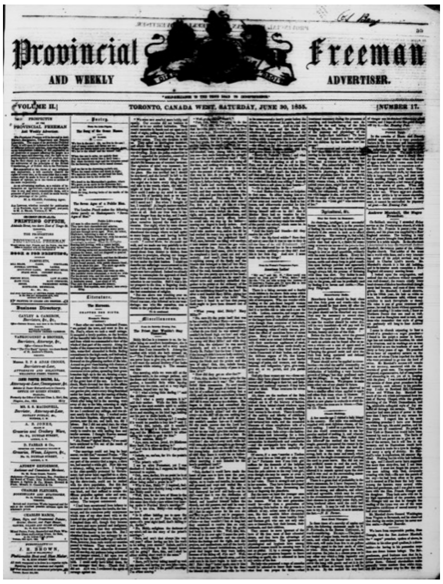

Figure 1: An example front page from the *Accessible Archives* corpus.

To resolve these errors, we apply large-scale n-gram counts from Google Books (Michel et al., 2011; Lin et al., 2012). The basic premise of this approach is that additional whitespace should be introduced in cases where a token is out-of-vocabulary, yet can be decomposed into two or more in-vocabulary tokens. By using bigram and unigram counts, it is possible to distinguish these cases, without treating membership in a predefined vocabulary as the sole and determinative indicator of whether a token should be segmented. Furthermore, by using higher-order n-gram counts, it is possible to make a contextualized judgment about whether and how whitespace

Proc. of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, pp. 98–103
Minneapolis, MN, USA, June 7, 2019. ©2019 Association for Computational Linguistics

should be introduced. We show that contextualization yields significant improvements in segmentation accuracy.

Our research is motivated by our own experience working with historical texts. We were fortunate to obtain access to a manually-digitized corpus of nineteenth-century newspapers from the United States.[1] However, the digitization process introduced whitespace errors, and the original tokenization was unrecoverable. These errors were sufficiently frequent as to substantially impact downstream analyses such as topic models and word embeddings. We undertook this research to solve this practical problem, but because we believe it generalizes beyond our specific case, we systematically analyze the performance of our solution, and release a trained system for whitespace recovery. To summarize our contributions:

- We present a new method for correcting common whitespace errors in digitized archives.
- We evaluate on new annotations of manual whitespace error corrections in a digitized historical corpus.
- We release a trained system for other researchers who face similar problems.[2]

2 Unsupervised Token Segmentation

A token is likely to contain missing whitespace if (a) the token is out-of-vocabulary; and (b) there is some segmentation of the token into substrings that are all in-vocabulary. By these conditions, the term *applebanana* is likely to contain missing whitespace. The term *watermelon* is excluded by condition (a), and *cherimoya* is excluded by condition (b).

In real scenarios, membership in a predefined vocabulary of terms is not the sole indicator of whether a token should be segmented: in some contexts, an "in-vocabulary" term should be segmented; in other cases, an out-of-vocabulary term, such as a name, should not be segmented. The premise of our approach is to approximate the notion of vocabulary inclusion with n-gram probabilities. Specifically, a segmentation is likely to be correct when the segments have high probability in a large corpus of (mostly) clean text, in comparison with both (a) the original token, and (b) other segmentations of that same token. We therefore apply a set of *likelihood ratios* to score candidate segmentations. The numerator quantifies the likelihood of a proposed segmentation, and the denominator quantifies the likelihood of the unsegmented token.

To describe our approach, we introduce the following notation. Let $w^{(t)}$ indicate token t from a corpus, where the tokenization is performed by simple whitespace pattern matching. We are concerned with the question of whether $w^{(t)}$ contains missing whitespace. Given a segmentation of $w^{(t)}$ such that i is the index of the first character in the second segment, we denote the segments as $w^{(t)}_{0,i}$ and $w^{(t)}_{i,\ell^{(t)}}$, where $\ell^{(t)}$ is the length of $w^{(t)}$ in characters.[3]

2.1 Non-contextual likelihood ratio

We first consider the probability of the bigram $(w^{(t)}_{0,i}, w^{(t)}_{i,\ell^{(t)}})$, in comparison with the unigram probability $w^{(t)}$:

$$r(w^{(t)}, i) = \frac{p_2\left(w^{(t)}_{0,i}, w^{(t)}_{i,\ell^{(t)}}\right)}{p_1(w^{(t)})}, \tag{1}$$

where p_2 is a bigram probability, and p_1 is a unigram probability. These probabilities can be computed from n-gram counts,

$$p_2(u,v) = \frac{n_2(u,v)}{\sum_{(u',v')} n_2(u',v')} \tag{2}$$

$$p_1(u) = \frac{n_1(u)}{\sum_{u'} n_1(u')}, \tag{3}$$

where n_2 and n_1 are bigram and unigram counts, respectively. The denominator of p_2 is the count of all bigrams, and the denominator of p_1 is the count of all unigrams. Both are equal to the total size of the corpus, and they cancel in Equation 1. This makes it possible to perform segmentation by directly comparing the raw counts. However, in the contextualized models that follow, it will be necessary to work with normalized probabilities.

To use Equation 1, we first identify the segmentation point with the highest score, and then compare this score against a pre-defined threshold. The threshold controls the tradeoff between recall and precision, as described in § 4.

[1] https://www.accessible-archives.com. The dataset is described in a review article by Maret (2016).

[2] https://github.com/sandeepsoni/whitespace-normalizer

[3] In our dataset, we do not encounter the situation in which a single token requires more than two segments. This problem is therefore left for future work.

In our experiments, the counts are obtained from Google n-grams (Michel et al., 2011). It is not essential that the corpus of counts be completely free of whitespace errors or other mistakes. As long as errors are independent and identically distributed across terms (in other words, each term is equally likely to have a segmentation error), the correct segmentation can still be recovered in the limit of sufficient data. This consideration prevents us from using the historical corpus, because it is possible that errors will be especially frequent for some terms, adding bias to the relevant n-gram counts.

2.2 Contextual likelihood ratio

The likelihood ratio based on word counts can be strengthened by considering additional context. Consider a term like *often*. According to Equation 1, we would be unlikely to segment *often* into *of ten*, since $p_1(\textit{often})$ exceeds $p_2(\textit{of ten})$, by a factor of 10-20 in the Google n-grams corpus.[4] Yet there are contexts in which segmentation is appropriate, such as the phrase *memory often years*.

We can resolve such cases by considering the additional context provided by the neighboring tokens $w^{(t-1)}$ and $w^{(t+1)}$:

$$r_c(w^{(t)}, i) = \frac{p\left(w^{(t)}_{0,i}, w^{(t)}_{i,\ell^{(t)}} \mid w^{(t-1)}, w^{(t+1)}\right)}{p(w^{(t)} \mid w^{(t-1)}, w^{(t+1)})}. \tag{4}$$

We decompose these terms into trigram and bigram probabilities. The numerator can be expressed as:

$$\begin{aligned} & p\left(w^{(t)}_{0,i}, w^{(t)}_{i,\ell^{(t)}} \mid w^{(t-1)}, w^{(t+1)}\right) \\ & \propto p_3(w^{(t+1)} \mid w^{(t)}_{i,\ell^{(t)}}, w^{(t)}_{0,i}) \\ & \times p_3(w^{(t)}_{i,\ell^{(t)}} \mid w^{(t)}_{0,i}, w^{(t-1)}) \\ & \times p_2(w^{(t)}_{0,i} \mid w^{(t-1)}), \end{aligned} \tag{5}$$

with p_3 and p_2 indicating trigram and bigram probabilities respectively. The denominator is similar:

$$\begin{aligned} & p\left(w^{(t)} \mid w^{(t-1)}, w^{(t+1)}\right) \\ & \propto p_3(w^{(t+1)} \mid w^{(t)}, w^{(t-1)}) \\ & \times p_2(w^{(t)} \mid w^{(t-1)}). \end{aligned} \tag{6}$$

In both the numerator and denominator, the constant of proportionality is $p(w^{(t+1)} \mid w^{(t-1)})$, which cancels from the likelihood ratio.

In the example above, the trigrams *memory of ten* and *of ten years* have relatively high conditional probabilities, and *memory often years* has a low conditional probability. This ensures that the appropriate segmentation is recovered.

Interpolation. The bigram and trigram probabilities in Equations 5 and 6 can be unreliable when counts are small. We therefore use interpolated probabilities rather than relative frequencies for p_3 and p_2:

$$\begin{aligned} p_3(u \mid v, w) = & \alpha_3 \hat{p}_3(u \mid v, w) \\ & + \beta_3 \hat{p}_2(u \mid v) \\ & + (1 - \alpha_3 - \beta_3)\hat{p}_1(u) \end{aligned} \tag{7}$$

$$p_2(u \mid v) = \beta_2 \hat{p}_2(u \mid v) + (1 - \beta_2)\hat{p}_1(u), \tag{8}$$

where $\hat{p}_n$ refers to the unsmoothed empirical n-gram probability, and $(\alpha_3, \beta_3, \beta_2)$ are hyperparameters. We manually set $\alpha_3 = 0.7, \beta_3 = 0.2, \beta_2 = 0.9$, and did not try other values.

3 Experimental Setup

We apply the segmentation techniques from the previous section to the Accessible Archives corpus, a dataset of manually digitized articles from newspapers in the nineteenth-century United States. As noted in the introduction, whitespace errors were introduced during the digitization process, likely by deleting newline characters when moving the files across operating systems. As a result, the dataset contains a relatively large number of concatenated terms, such as *andsaw, daythe, dreamsof, manufactureof, onlytwo, returningto, showsthe, theboys, thelevel*, and *thesea*.

To measure segmentation accuracy, two of the authors manually annotated a randomly-selected subset of 200 terms that occur in at least 5 contexts in the corpus. In each case, the annotator either provides the correct segmentation or indicates that no segmentation is necessary. The annotators indicated that 33 % of the terms needed a segmentation and agreed on all segmentation decisions, indicating that this problem is unambiguous for human readers. Although a high proportion of terms required segmentation, these terms were all concentrated in the long tail of the distribution of the terms by frequency. This indicates that the segmentation errors are spread across several terms in the corpus but are still rare and may not adversely affect the readability of the corpus. We tested the ability of likelihood ratio scores to recover the true

[4] From a web interface search of American books in the 19th century.

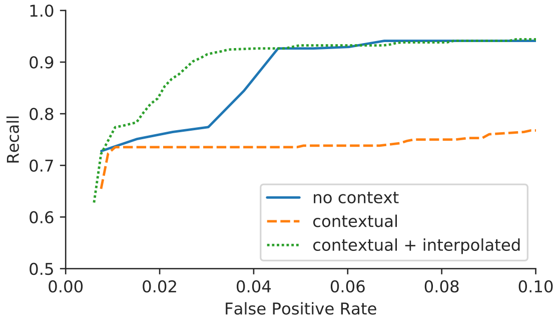

Figure 2: Performance of each method. The false positive rate is controlled by varying the threshold for segmentation.

segmentations. The evaluation is based on the following counts:

True positive: The system proposes a segmentation, and it matches the annotated segmentation.

False positive: The system proposes a segmentation, and either it does not match the annotated segmentation or the annotators marked the term as unsegmented.

False negative: A segmentation was annotated, and the system does not propose it.

True negative: A segmentation was not annotated, and the system does not propose one.

The **recall** is computed as $\text{TP}/(\text{TP}+\text{FN})$, and the **false positive rate** is computed as $\text{FP}/(\text{FP}+\text{TN})$.

4 Results

Results are shown in Figure 2 and in Table 1. The contextualized likelihood ratio obtains a recall of 0.768 at a false positive rate of 0.008, and a recall of 0.909 at a false positive rate of less than than 0.029. Contextualization substantially improves the recall at low false positive rates, but only when used in combination with interpolated probabilities. This indicates that contextualization makes it possible to segment more aggressively without suffering false positives.

We also illustrate the strengths of each method through examples. Tokens like *Themotion*, *andprovided* and *wearthese* are correctly segmented as *The motion*, *and provided* and *wear these*. However, due to sparse counts in the trigram dictionaries, merely adding the context does not lead to correct segmentations in these cases without additionally using interpolation. On the other hand, not relying on context leads to erroneous segmentations for tokens like *innumerous* (as *in numerous*), *Safeguard* (as *Safe guard*) and *Norice* (as *No rice*). Both contextualization and interpolation help in correcting these errors. Note that adding interpolation to the contextualization helps find a sweet spot between the more aggressive non-contextual model and the less aggressive contextual model.

All three methods are based on the calculation of likelihood ratio, which is crucial for their success. To show this, we additionally evaluate the performance for a rule-based baseline with the two rules described in § 2: we segment a token if it is out-of-vocabulary and some segmentation is in-vocabulary. When there are multiple valid segmentations, the segmentation with the largest second segment by length was chosen. The precision and false positive rate of this baseline is 0.24, 0.39 respectively. This shows the advantage of probabilistic segmentation over a deterministic dictionary-based alternative.

5 Related Work

Dataset "cleanliness" is an increasingly salient issue for digital humanities research. Difficulties with optical character recognition (OCR) were highlighted in a 2018 report to the Mellon Foundation (Smith and Cordell, 2018), which outlines an agenda for research and infrastructure development in handling such texts. A key point from this report is that *postprocessing* of noisily digitized texts will continue to be important, despite the obvious interest in improving the accuracy of OCR itself (e.g., Berg-Kirkpatrick et al., 2013).

Several papers tackle the more general problem of OCR post-correction. An early example is the work of Tong and Evans (1996), who employ bigram word counts and character transduction probabilities to score corrections by their log-probability. However, their approach cannot handle whitespace erorrs (which they refer to as "run-on" and "split-word" errors). Another approach is to train a supervised system from synthetic training data, using features such as proposed spelling corrections (Lund et al., 2011). Dong and Smith (2018) propose an alternative unsupervised training technique for OCR post-correction, which builds on character-level LSTMs. In their method, which they call `seq2seq-noisy`, they build an ensemble of post-processing systems. On each ex-

False positive rate:	0.01	0.03	0.05	0.1
No context likelihood ratio	0.750	0.765	0.926	0.941
Contextual likelihood ratio	0.735	0.735	0.735	0.768
Contextual likelihood ratio + Interpolation	0.768	0.909	0.932	0.944

Table 1: Maximum segmentation recall at various false positive rates.

ample, a candidate output is produced by each system in the ensemble. They then select as noisy ground truth the system output that scores highest on a character-level language model trained on clean text from a New York Times (NYT) corpus, and use this noisy ground truth to train the other members of the ensemble.

Our paper approaches a special case of the general OCR post-correction problem, focusing specifically on whitespace errors, which Kissos and Dershowitz (2016) call *segmentation errors*. A key point is that these errors can and do arise even in texts that are manually keyed in, due to mishandling of file formats across operating systems. We are interested to test the applicability of general OCR post-correction systems to whitespace errors, but our results suggest that this problem can be addressed by the more lightweight solutions described here.

6 Conclusion

This paper describes an unsupervised approach for post-correcting whitespace errors, which are frequently present in digitized humanities archives. These errors can be resolved by considering two sources of information: character-level information about which surface forms are likely to be word tokens, and contextual information about which tokens are likely to appear in context. Both sources of information can be obtained from large-scale n-gram statistics, and combined using a straightforward likelihood ratio score. The resulting segmenter obtains high recall with a minimal rate of false segmentations. Tuning the interpolation coefficients on a validation set may improve performance further. Future work should test the applicability of these techniques in languages beyond English, and on other types of errors.

Acknowledgments

We thank the reviewers for their constructive feedback, as well as Georgia Tech computational linguistics lab members Ian Stewart, Sarah Wiegreffe, and Yuval Pinter for a round of edits. In addition, we thank digital humanities lab members Bin Cao and Nikita Bawa for their early efforts in cleaning this corpus. The work also benefited from discussion with Ryan Cordell and David Smith. This research was supported by the Andrew W. Mellon Foundation under the auspices of Georgia Tech's Digital Integrative Liberal Arts Center. The initial work on this project was supported by the National Endowment for the Humanities, Office of Digital Humanities grant number HD-51705-13.

References

Taylor Berg-Kirkpatrick, Greg Durrett, and Dan Klein. 2013. Unsupervised transcription of historical documents. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 207–217, Sofia, Bulgaria. Association for Computational Linguistics.

Rui Dong and David Smith. 2018. Multi-input attention for unsupervised OCR correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2363–2372, Melbourne, Australia. Association for Computational Linguistics.

Susanne Haaf, Frank Wiegand, and Alexander Geyken. 2013. Measuring the correctness of double-keying: Error classification and quality control in a large corpus of TEI-annotated historical text. In *Selected Papers from the 2011 TEI Conference*, volume 4. TEI.

Ido Kissos and Nachum Dershowitz. 2016. OCR error correction using character correction and feature-based word classification. In *IAPR Workshop on Document Analysis Systems (DAS)*, pages 198–203. IEEE.

Yuri Lin, Jean-Baptiste Michel, Erez Lieberman Aiden, Jon Orwant, Will Brockman, and Slav Petrov. 2012. Syntactic annotations for the Google Books Ngram Corpus. In *Proceedings of the ACL 2012 system demonstrations*, pages 169–174. Association for Computational Linguistics.

William B Lund, Daniel D Walker, and Eric K Ringger. 2011. Progressive alignment and discriminative error correction for multiple OCR engines. In *2011 International Conference on Document Analysis and Recognition*, pages 764–768. IEEE.

Susan Maret. 2016. Accessible archives. *The Charleston Advisor*, 18(2):17–20.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

David Smith and Ryan Cordell. 2018. A research agenda for historical and multilingual optical character recognition. http://hdl.handle.net/2047/D20297452, accessed February 2019.

Ray Smith. 2007. An overview of the tesseract ocr engine. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, volume 2, pages 629–633. IEEE.

Simon Tanner, Trevor Munoz, and Pich Hemy Ros. 2009. Measuring mass text digitization quality and usefulness: Lessons learned from assessing the OCR accuracy of the British Library's 19th century online newspaper archive. *D-Lib Magazine*.

Xiang Tong and David A Evans. 1996. A statistical approach to automatic OCR error correction in context. In *Fourth Workshop on Very Large Corpora*.