Noname manuscript No.
(will be inserted by the editor)

Corpus Construction for Historical Newspapers: A Case Study on Public Meeting Corpus Construction using OCR Error Correction

Koji Tanaka, Chenhui Chu*, Tomoyuki Kajiwara, Yuta Nakashima, Noriko Takemura, Hajime Nagahara, Takao Fujikawa

Received: date / Accepted: date

Abstract Large text corpora are indispensable for natural language processing. However, in various fields such as literature and humanities, many documents to be studied are only scanned to images, but not converted to text data. Optical character recognition (OCR) is a technology to convert scanned document images into text data. However, OCR often misrecognizes characters due to the low quality of the scanned document images, which is a crucial factor that degrades the quality of constructed text corpora. This paper works on corpus construction for historical newspapers. We present a corpus construction method based on a pipeline of image processing, OCR, and filtering. To improve the quality, we further propose to integrate OCR error correction. To this end, we manually construct an OCR error correction dataset in the historical newspaper domain, propose methods to improve a neural OCR correction model and compare various OCR error correction models. We evaluate our corpus construction method on the accuracy of extracting articles of a specific topic to construct a historical newspaper corpus. As a result, our method improves the article extraction F-score by 1.7% via OCR error correction comparing to previous work. This verifies the effectiveness of OCR error correction for corpus construction.

Keywords OCR error correction · historical newspapers · corpus construction · public meeting

1 Introduction

Large-scale text corpora are essential for natural language processing (NLP). Most existing corpora are created from text that has already been digitized. For instance, the benchmark syntactic parsing dataset *Penn Treebank* (Marcus et al, 1993) is created by labelling part-of-speech tags and syntactic information on the digitized text from the Wall Street Journal newspapers. The parallel corpus *Europarl* (Koehn, 2005) that

* Corresponding author: Chenhui Chu
E-mail: chu@i.kyoto-u.ac.jp, Kyoto University

has been used for shared tasks in the conference on machine translation (Barrault et al, 2019), is created by aligning parallel sentences from the digitized multilingual European Parliament data.

However, in various fields including literature, humanities, and engineering drawing digitization (Moreno-García et al, 2019; Moreno-García and Elyan, 2019), many materials to be studied are not digitized, which are stored in a physical medium such as papers or just scanned to images but not transcribed into text. By digitizing and transcribing such materials into text, and structuring them via extracting specific topics, we can apply many NLP techniques to analyzing them automatically. In these research fields, digitization, text transcription, and structuring can significantly increase the value of the original materials. Therefore, corpus construction for these fields is very important.

Optical character recognition (OCR) is the technology for converting scanned images to text data. In general, OCR is implemented by character delimiter recognition, size normalization, feature extraction, and character classification (Smith, 2007). OCR is indispensable for constructing corpora in literature, humanities, and engineering drawing digitization fields. However, OCR often makes errors when the document images have defects due to, e.g., dirt and damage (Chiron et al, 2017). OCR errors may significantly decrease the quality of the constructed corpus. In particular, historical documents may suffer from severe OCR errors due to immature printing technologies and deterioration of media; therefore, automatic OCR error correction is crucial to improve the quality of the corpus.

In this paper, we work on corpus construction based on the historical newspaper database Trove (Cassidy, 2016; Sherratt, 2021)[1][2] and target "public meeting" articles in Australian historical newspapers (Fujikawa, 1990). Public meetings were the main pillar of public opinion formation for Western Europe, spanning 120 years from the 19th to 20th century (Fujikawa, 1990). Note that our targeted "public meeting" articles in this paper are the ones in the "advertisement" pages not the ones in the main news pages. We do appreciate that Trove has already provided the page boundary of advertisements in their database in the formats of both PDF files and OCRed text, but unfortunately the boundaries for individual articles including "public meeting" articles in advertisement pages are unavailable. We start our work from the available advertisement pages provided by Trove.

The knowledge obtained from "public meeting" articles is important for understanding Australian history, and it is expected that analysis of long-running "public meeting" articles will provide new insights in Australian history. In our previous study (Tanaka et al, 2020), we proposed a method to construct a "public meeting" domain corpus from Trove. However, OCR errors significantly affected the corpus construction accuracy and the effective use of the corpus. To address OCR errors, this paper extends our previous study by improving the corpus construction method integrating OCR error correction.

[1] `https://trove.nla.gov.au`

[2] Trove is an online library database service maintained by the Australian government, which covers major Australian daily newspapers and local newspapers.

To this end, firstly, we build an OCR error correction dataset for historical newspapers, especially in the "public meeting" domain. Our OCR error correction dataset consists of OCRed text and manually corrected text correspondingly. Our dataset provides knowledge on wording and typical failures especially due to immature printing technologies and deteriorated printing of historical newspapers. We conduct OCR error correction experiments, comparing one statistical and two neural network (NN)-based models (i.e., a neural machine translation (NMT)-based model and a semi-supervised NN-based model) on our dataset. For the semi-supervised NN-based model, we propose fine-tuning a model pretrained on a news domain dataset on our dataset, which improve word error rate (WER) and character error rate (CER) by 8.63% and 3.26%, respectively. In addition, we propose reranking with language models and dictionary match scores, which further improves OCR error correction for proper nouns. We also demonstrate that the statistical model performs better than the NN-based models, which reduces WER and CER by 23.43% and 9.07%, respectively, compared to vanilla OCR.

Secondly, we present a corpus construction method based on a pipeline of image processing, OCR, and filtering following Tanaka et al (2020) but with novel integration of OCR error correction. We first identify the rule lines in advertisement page images and trim the images into articles. Next, we apply OCR to the trimmed articles. Then, we use our best OCR error correction model for the OCRed text. Finally, we extract the articles with specific topic words by filtering. Evaluation conducted on manually annotated ground-truth "public meeting" articles indicates that our method achieves a F-score of 68.7% with a high recall of 93.7%, whose F-score is 1.7% improved via OCR error correction comparing to the best performance of (Tanaka et al, 2020). In addition, our method can extract 15.9% more articles without excess and deficiency, compared to a baseline that is based on linguistic features to identify beginning and ending sentences of "public meeting" articles from the OCRed text of the entire advertisement pages provided by Trove. We will release our OCR error correction dataset, corpus construction toolkit, and the constructed corpus upon acceptance. The contributions of this paper are as follows:

- We create an OCR error correction dataset in the "public meeting" domain. We further propose fine-tuning on our dataset, and reranking with language models and dictionary match scores to improve a semi-supervised NN-based OCR error correction model and compare it with one statistical and another NN-based model.
- We integrate OCR error correction to our previous "public meeting" corpus construction method, and verify the effectiveness of it.
- Although experiments are conducted on the "public meeting" domain only, our OCR error correction construction method and models can be easily applied to other domains, and our corpus construction method is general enough to be applied to any historical newspapers data.

The remaining of this paper is organized as follows. We first present related work in Section 2. After presenting our OCR error correction dataset and models in Section 3, we introduce our corpus construction method with OCR error correction in Section 4. Next, we describe the experimental settings for OCR error correction and corpus

construction in Section 5 and discuss the results in Section 6. Finally, we conclude this paper in Section 7.

2 Related Work

2.1 OCR Error Correction Methods

There are two types of OCR error correction methods, i.e., supervised ones (Kolak and Resnik, 2002; Kolak et al, 2003; Yamazoe et al, 2011) and unsupervised ones (Lund et al, 2013; Dong and Smith, 2018). Although the unsupervised methods are being improved to an accuracy that is near to supervised methods, supervised methods still outperform unsupervised methods (Dong and Smith, 2018). Therefore, we create a dataset for supervised and semi-supervised OCR error correction modeling, improve OCR error correction models based on it, and apply OCR error correction trained on our dataset to the corpus construction system.

2.2 OCR Error Correction for Historical Documents

There are many studies aimed at OCR error correction for historical documents (Kolak and Resnik, 2002; Kolak et al, 2003; Yamazoe et al, 2011). In particular, historical documents are difficult to perform OCR due to the underdeveloped printing technology and paper degradation. For this reason, there can be many OCR errors, making error correction more difficult. Barbaresi (2016) compared many morphological analysis systems for OCR error detection for German newspapers. Afli et al (2016) proposed to adopt statistical machine translation (SMT) to OCR error correction for historical documents and WER was reduced by 2.9% compared to vanilla OCRed text. Eger et al (2016) compared different character-level translation models for spelling error correction. Xu and Smith (2017) proposed duplication passage detection and a consensus decoding method. Dong and Smith (2018) presented an unsupervised OCR error correction model based on NMT. In this paper, we compare SMT and our improved NN-based methods for OCR error correction on our dataset. Lyu et al (2021) proposed a convolutional NN encoder with a recurrent NN (RNN) decoder for OCR error correction.

Klein and Kopel (2002) presented a post-processing system based on statistical information and dictionaries. Richter et al (2018) proposed a post-processing tool. The tool suggests multiple correction word candidates for an incorrect word, and annotators can select a suitable candidate as the correction word. As a result of annotation using the tool, they achieved 6.3% improvement in WER compared to the OCRed text. OCR post-processing also has been studied for other languages, including Arabic (Trad and Doush, 2016) and French (Afli et al, 2015). We leave OCR post-processing as one of our future work.

2.3 Historical Corpus Construction

Several studies on corpus construction for historical documents have been conducted. Davies (2012) built an American English historical corpus. They collected text from magazines, newspapers, and books from 1810 to 2000. They further lemmatized and labeled part-of-speech (POS) tags on the corpus. Rögnvaldsson et al (2012) built an Icelandic parsed historical document corpus. They collected text from the 12th to the 21st century and annotated them for parsing using the same schema as the Penn Treebank (Marcus et al, 1993). Sánchez-Martínez et al (2013) built a Spanish historical corpus. They collected text from prose, theatre, and verse from 1481 to 1748, lemmatized them, and labeled POS tags. Neudecker (2016) built a corpus for named entity recognition from historical newspapers in French, Dutch, and German. They annotated named entity tags for the *Europeana Newspaper* from the 17th to the 20th century using the INL Attestation Tool.[3] Cassidy (2016) built an Australian historical newspaper corpus and published it on a website called Trove. They converted newspapers from the 19th century to the 21st century into text data using OCR. We construct our corpus based on Trove. Different from (Cassidy, 2016) and other previous studies, we extend our previous method (Tanaka et al, 2020) to extract articles with the specific topic of "public meeting" from Trove with OCR error correction.

3 OCR Error Correction

3.1 Dataset

3.1.1 Annotation

We used the advertisement pages crawled from Trove, and the targeted articles were the ones containing the key phrase "public meeting." We searched the key phrase "public meeting" on Trove and narrowed our search range to advertisement pages only to get the advertisement page IDs. There were $407,756$ advertisement pages including the key phrase "public meeting" in Trove. Figure 1 shows an screenshot of the Trove search interface to get the "public meeting" advertisement pages. Next, we obtained the advertisement page PDF data through the API provided by Trove with the advertisement page IDs.

We first manually sampled 5 advertisement pages from 1838 to 1954 each year in Trove, which include the "public meeting" articles. We used the OCRed text provided by Trove. An advertisement page may contain multiple "public meeting" articles. As a result, we obtained 719 "public meeting" articles (including $13,543$ lines). Note that volunteers are correcting a part of the OCRed text in the Trove database as well (Evershed and Fitch, 2014). However, we wanted to create a "public meeting" domain-specific OCR error correction corpus in this paper, and thus we sampled the "public meeting" articles and annotated them by ourselves.

The OCRed text has many errors because the newspaper medium (i.e. paper) is smeared or deteriorated before scanning. This imposes a particular challenge for

[3] https://github.com/INL/AttestationTool

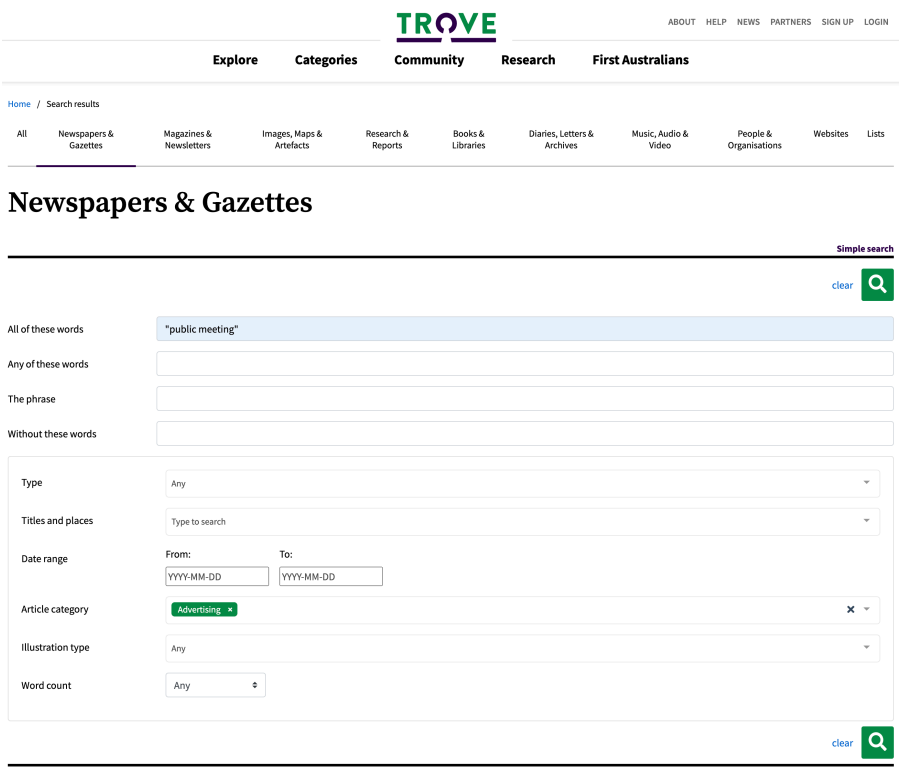

Fig. 1 The Trove search interface to get the "public meeting" advertisement pages.

proper nouns, such as person and place names in the articles. Because proper nouns are barely inferred from the context, whereas they are crucial for historical analysis. Therefore, we asked an expert of Australian history to manually correct all OCR errors including person and place names. Specifically, under the expert's supervision, 5 students who major in British history annotated the 719 "public meeting" articles by comparing the OCRed text and the original advertisement pages so that they identified OCR errors and corrected them accordingly.[4] The expert then checked the corrected text to ensure the quality.

3.1.2 Statistics

We divided the annotated dataset into 577, 71, and 71 articles for training, validation, and testing, respectively. Table 1 shows the statistics. WER is calculated for the

[4] Note that due to the large number (i.e., 407,756) of overall advertisement pages including "public meeting" articles, it is almost impossible to either digitize all of them or correct the OCR errors in all of them manually.

	Articles	Lines	Tokens	Characters	WER(%)	CER(%)
Train	577	11,575	70,914	337,896	26.50	9.82
Valid	71	1,227	6,881	33,462	25.49	9.66
Test	71	1,389	8,528	40,319	26.57	9.68

Table 1 Statistics of our dataset for the training, validation, and testing splits.

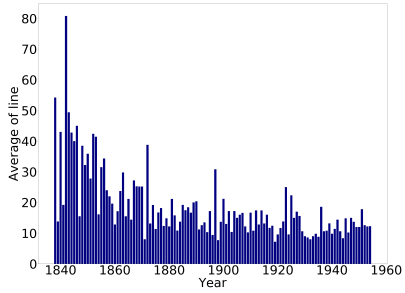

Fig. 2 Average number of lines in an article per year in the annotated OCR error correction dataset.

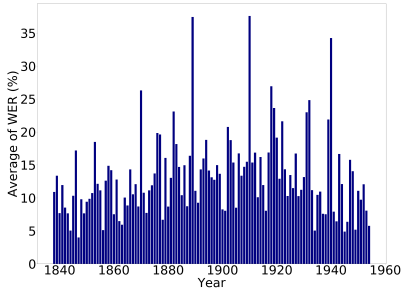

Fig. 3 Average WER in a single article per year in the annotated OCR error correction dataset.

OCRed text against the corrected text as:

$$\mathrm{WER} = (C + I + D)/N,$$

where C, I, and D are the numbers of words corrected, inserted, and removed during the manual annotation; N is the number of words in the corrected text. CER is calculated in the same way as WER but on characters. Figures 2 and 3 show the average number of lines and the average WER in the articles over the respective year. Although the number of annotated samples per year is small, we would say that there are some trends in WER and the number of lines in a single article: WER increases until 1910's, and then decreases; the number of lines seems to keep decreasing over time, and this could be because of changes in the design of articles. Figure 4 shows some examples of "public meeting" articles in 1840, 1881, and 1938. We can see that a newer article has less lines. In addition, a newer article uses more diversified fonts, and important information is printed with larger characters, which can affect OCR errors.

3.2 Models

We compare one statistical and two NN-based models (an NMT-based model and a semi-supervised NN-based model) on our dataset for our corpus construction task. For the semi-supervised NN-based model, we further propose fine-tuning and reranking.

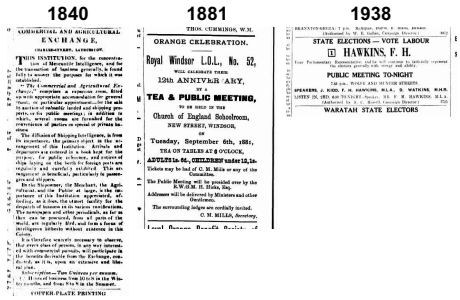

Fig. 4 Visual differences in example articles in 1840, 1881, and 1938.

3.2.1 Statistical Machine Translation (SMT) based Model

Afli et al (2016) showed that SMT is effective for OCR correction in historical documents. Therefore, we employ a SMT model for OCR correction. Let $X = \{x_i | i = 1, \ldots, M\}$ and $Y = \{y_j | j = 1, \ldots, N\}$ denote original OCRed text and corrected text, where x_i and y_j denote characters and M and N are the number of characters. SMT finds $\hat{Y}$ that maximizes the following joint probability:

$$\hat{Y} = \text{argmax}_Y \sum_a P(X, \mathbf{a} \mid Y) P(Y),$$

where $\mathbf{a}$ is the character-level alignment between X and Y. $P(Y)$ is the language model probability, defined as:

$$P(Y) = \prod_{j=1}^{N} P(y_j \mid y_{1:j-1}),$$

where N is the character numbers in Y.

3.2.2 Neural Machine Translation (NMT) based Model

Mokhtar et al (2018) showed that the character-level sequence-to-sequence model proposed by Chung et al (2016) outperforms a SMT-based OCR error correction model for modern documents. Therefore, we apply NMT as one of our NN-based OCR error correction models. The model we use is a RNN-based character-level sequence-to-sequence model. We train the model with X and Y pairs. Firstly, we obtain the encoder's hidden states $\mathbf{h}_{\text{enc}} = \mathbf{h}_M$ for the last character x_M by:

$$\mathbf{h}_{\text{enc}} = \text{RNN}_{\text{enc}}(x_M, \mathbf{h}_{M-1}),$$

where $\mathbf{h}_{M-1}$ is the hidden state vector at time step $M - 1$.

For inference, we compute the decoder's hidden state $\mathbf{h}_j$ for character y_j at time step j by]

$$\mathbf{h}_j = \text{RNN}_{\text{dec}}(y_{j-1}, \mathbf{h}_{j-1}),$$

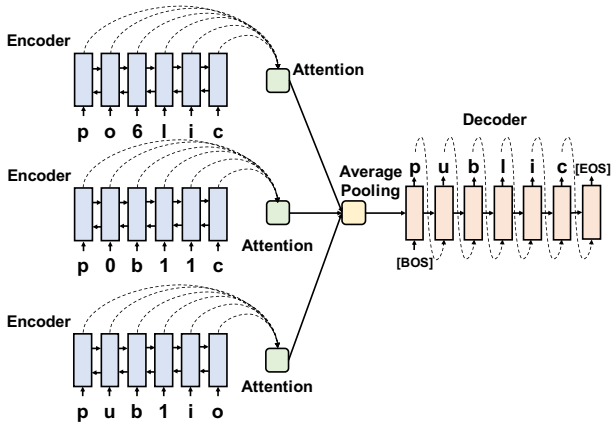

Fig. 5 Overview of the pretrained OCR error correction model by Dong and Smith (2018).

where $\mathbf{h}_{j-1}$ is the hidden state for the previous generated character y_{j-1}, and the decoder's hidden state is initialized by $\mathbf{h}_{\mathrm{enc}}$. The corrected text $\hat{Y}$ is given by maximizing the following probability:

$$
\begin{aligned}
P(Y \mid X) &= \prod_{j=1}^{N} P(y_j \mid X, y_{1:j-1}) \\
&= \prod_{j=1}^{N} \mathrm{softmax}(\phi(\mathbf{h}_j)),
\end{aligned}
$$

where the summation is computed over all possible characters and special tokens (i.e., `[BOS]` and `[EOS]`) , and ϕ is an activation function.

3.2.3 Semi-Supervised Model

We also adopt a NN-based semi-supervised model for OCR error correction. The model is based on a model pretrained with a large-scale OCRed text corpus from another domain. We further fine-tune on our OCR error correction dataset in order to be specific to the "public meeting" domain. Moreover, we propose to apply reranking to the model to correct the error correction of proper nouns in a sentence.

Pretrained Model (Pre) Although we only have a small amount of annotated data for OCR error correction in a specific domain, multiple candidates of text obtained using different OCR techniques may be available in other domains.[5] Therefore, we use the unsupervised model by Dong and Smith (2018) to correct OCR errors, where multiple OCR candidates are used for training. Figure 5 shows an overview of the model by Dong and Smith (2018). Let $\mathcal{X} = \{X_l \mid l = 1, \ldots, L\}$ denote a set of original OCRed

[5] Unfortunately, multiple OCR candidates are unavailable for our "public meeting" domain, and thus we used a corpus from a different domain in our experiment.

text of the same sentence by different OCR techniques. These lines of text are fed into the respective encoders (with shared parameters). Character-level attention weights, computed based on the decoder's j-th hidden state $\mathbf{g}_j$, are multiplied to the hidden states of each encoder. Formally, let $\mathbf{h}_{li}$ denote encoder l's hidden state for X_l's i-th character x_{li}, and $\mathbf{g}_{j-1}$ the decoder's hidden state for $j-1$-th character. The context vector $\mathbf{c}_{jl}$ from encoder l for the decoder's j-th character is given by:

$$\alpha_{jli} = \mathrm{softmax}(\mathrm{f}(\mathbf{g}_{j-1}, \mathbf{h}_{li})),$$
$$\mathbf{c}_{jl} = \sum_{i=1}^{M_l} \alpha_{jli} \mathbf{h}_{li},$$

where f is a fully-connected layer and M_l is the number of characters in X_l. The context vectors from all encoder layers are then average pooled as:

$$\mathbf{c}_j = \sum_{l=1}^{L} \frac{1}{L} \mathbf{c}_{jl}.$$

For the decoder's j-th output y_j, we compute the probability by:

$$\mathbf{g'}_j = \tanh(W_c[\mathbf{c}_j, \mathbf{g}_j] + \mathbf{b}_c),$$
$$P(y_j \mid y_{1:j-1}, X) = \mathrm{softmax}(W_s \mathbf{g'}_j + \mathbf{b}_s), \quad (1)$$

where $[\cdot, \cdot]$ denotes concatenation; W_c, $\mathbf{b}_c$, W_s, and $\mathbf{b}_s$ are parameters to be trained. Based on Eq. (1), we can define the sentence generation probability given the original OCRed text $\mathcal{X}$ as:

$$P(Y) = \prod_{j=1}^{N} P(y_j \mid y_{1:j-1}, \mathcal{X}),$$

where N is the number of output characters.

Fine-Tuning (FT) The domain of the pretrained model is different from our "public meeting" dataset, which is a specific topic in Australian newspapers. Therefore, there are differences in wording and grammars. Moreover, historical documents can be more different over time. To alleviate this problem, we propose to fine-tune the pretrained model to adapt to the "public meeting" domain. Although the model is trained in an unsupervised manner, we fine-tune it in a supervised manner because we do have the original OCRed text and its corrected text. For this, we remove average pooling and train it with X and Y pairs on our dataset.

Reranking (RR)

Language Model Score. Our pretrained model corrects OCRed text in the character unit and does not consider fluency defined in the word unit. Therefore, we propose to rerank top-k[6] outputs obtained from beam search based on a language

[6] We used top-128 in our experiments.

model to improve the fluency. Because our dataset is too small to train a good language model, we use the pretrained GPT model[7] (Radford and Narasimhan, 2018), which is a large-scale language model for natural language generation released by OpenAI. We denote the probability of the t-th word given the sequence of words $w_{1:t-1}$ obtained from GPT by $P_{\mathrm{gpt}}(w_t \mid w_{1:t-1})$. Using P_{gpt}, we define r_{lm} indicating the fluency of the sentence based on GPT as:

$$r_{\mathrm{lm}} = \prod_{t=1}^{T} P_{\mathrm{gpt}}(w_t \mid w_{1:t-1}),$$

where T is the number of words.

Dictionary Match Score. Proper nouns are one of the most important factors for historical document analysis. Therefore, we further propose to use a proper noun dictionary dedicated for the domain to rerank. Let S denote the sequence of words in the corrected text, D the set of proper nouns that can appear in the articles. We define a dictionary match score by

$$r_{\mathrm{dict}} = \sum_{d \in D} \delta(d \in S) \frac{\mathrm{len}(d)}{\mathrm{len}(S)},$$

where $\delta(d \in S)$ is 1 if $d \in S$ and 0 otherwise, and $\mathrm{len}(\cdot)$ gives the length of the sequence in characters. r_{dict} yields 1 when all words in S are proper nouns in D.

Optimization over Reranking Score Parameters. We define the reranking score for Y by:

$$R(Y) = \alpha P(Y) + \beta r_{\mathrm{lm}} + \gamma r_{\mathrm{dict}},$$

where α, β, and γ are weights to determine the contribution of respective terms. α, β, and γ are tuned with Bayesian optimization (Snoek et al, 2012) on a validation set to minimize WER.

4 Corpus Construction

The overview of our proposed corpus construction method is shown in Figure 6. Because the OCRed text provided by Trove lacks the rule line information in the advertisement pages, it is difficult to extract only "public meeting" articles accurately. Therefore, we propose a method to address this problem by detecting rule lines from the image. We first identify the rule lines in advertisement images, and then trim the rule lines to extract images for articles. Next, we apply OCR to the extracted article images to extract text for the articles. We further apply OCR error correction to OCRed text. Finally, we filter the articles with a query phrase and thus extract only the target articles that we are interested in.

[7] https://openai.com/blog/language-unsupervised/

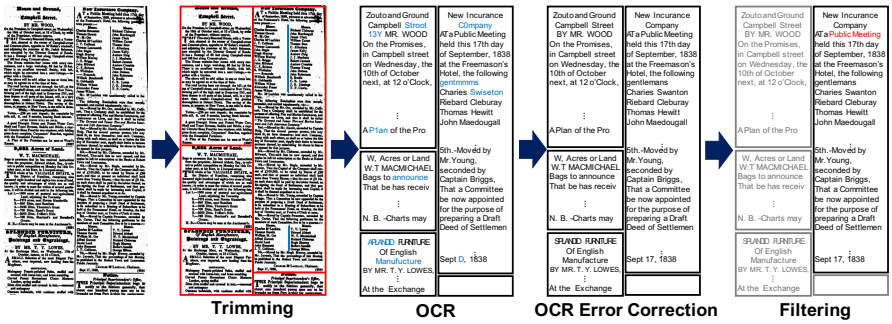

Fig. 6 Overview of the corpus construction method (We start from advertisement images containing the keyword "public meeting" obtained from Trove. Small columns in articles are shown as blue lines in the "trimming" sub-figure, and OCR errors are shown in blue fonts in the "OCR" sub-figure).

4.1 Trimming

We use OpenCV[8] for identifying the rule lines in advertisement images and trimming. Firstly, we binarize the advertisement images using the method proposed by Otsu (1979). The binarization method transfers grayscale images to white-black images by calculating the threshold that maximizes the separation degree from the histogram of picture element numbers. Next, we apply the contour tracking processing algorithm of (Suzuki and Abe, 1985) to extract the contours from the binarized images. In order to identify the contours, this algorithm calculates the boundary of the binarized images and sequentially detects the pixels that are the contour counterclockwise. Areas with a height above a threshold and a width below a threshold are identified as a column, and areas with a width above a threshold and a height below a threshold are identified as an article split in the advertisement image. The thresholds are tuned manually. After that, we can finally trim the article images accordingly.

There are small columns in articles as the blue lines shown in the sub-figure "trimming" of Figure 6. To deal with this, we propose the following method to determine the vertically split column. Firstly, we trim the column with the x coordinate (horizontal direction) value. We then compare the minimum and maximum y coordinate (vertical direction) values with the advertisement coordinate value. If the difference is above a predefined threshold, we determine it as a small column and do not use it for trimming.

4.2 OCR

OCR is generally performed following the procedures of character delimiter recognition, size normalization, feature extraction, and classification. Google open-sources the OCR method Tesseract (Smith, 2007), which achieves 98.4% and 97.4% on newspaper articles in character and word level, respectively. However, after comparing the

[8] https://opencv.org/

OCR accuracy of Google Drive[9] to Tesseract, we find that Google Drive works better. Therefore, we use the OCR function of Google Drive for extracting text from the article images.

4.3 OCR Error Correction

Because the OCRed text has errors (as shown in blue fonts in the sub-figure "OCR" of Figure 6), we apply OCR error correction (see Section 3) to the OCRed text. We use the SMT model for OCR error correction as it shows the best performance in our experiments (see Section 6.1). Note that this differs from our previous work (Tanaka et al, 2020) where we did not integrate OCR error correction. In addition, words can be separated into two continuous lines by hyphens, and our OCR error correction model can also address this issue.

4.4 Filtering

We filter the OCRed articles that are not our target with a query phrase, leaving the target articles to be extracted. In order to allow the error of character recognition by OCR, we define similarities in character level. We use the Python *difflib* module SequenceMatcher[10] for calculating similarities. In SequenceMatcher, the similarities between a character string pair is defined as:

$$\text{Similarity} = \frac{2.0 * M}{T}, \tag{2}$$

where M is the number of matched characters and T is the sum of character numbers in the character string pair.

We get word n-grams from the articles according to the number of words in the query character string. We then calculate the similarity between the n-gram and query character string, and take the articles with the highest similarity above a threshold as the target article. The threshold is tuned on a validation set, which shows the highest F-score.

5 Experimental Settings

5.1 OCR Error Correction

SMT We used the phrase based SMT (PBSMT) toolkit Moses (Koehn et al, 2007).[11] We trained a 5-gram language model[12] on the target side of the training data using the

[9] https://onlizer.com/google_drive/tesseract_ocr

[10] https://docs.python.jp/3/library/difflib.html

[11] http://www.statmt.org/moses/

[12] 5-grams language models have been used by default in SMT and we followed that for the OCR error correction task following (Chu et al, 2015).

KenLM toolkit[13] with interpolated Kneser-Ney discounting. For word alignment, we used the GIZA++ toolkit.[14] Tuning was performed by minimum error rate training (Och, 2003).

NMT We used the Open-NMT tookit[15] (Klein et al, 2017) and trained the model on our dataset. We set the dimensions of the word embedding and the RNN hidden unit to 128 and 512, respectively. The batch size was set to 16. The AdaGrad (Duchi et al, 2011) optimizer with a learning rate of 0.15 was used for optimization. We used the model with the lowest validation perplexity for testing, and set the beam size to 4.

Semi-Supervised Model We used the implementation from (Dong and Smith, 2018)[16] for pretraining the OCR error correction model. We used gated recurrent unit (GRU) (Cho et al, 2014) as RNN layers in the model, and set the dimensions of the GRU hidden unit to 512. The batch size was set to 128. The cross-entropy loss was used for the loss function, and the Adam (Kingma and Ba, 2015) optimizer with a learning rate of $3e-4$ was used for optimization. We trained the pretrained model on the Richmond Daily Dispatch (RDD) newspaper corpus.[17] RDD is a dataset in the *American newspaper* domain. RDD contains $1,384$ issues ($2.2M$ lines) from 1860 to 1865. As RDD does not have multiple OCRed text by different OCR techniques, following (Dong and Smith, 2018) we retrieved 3 similar lines of text from other issues in RDD. We applied the text reuse approach (Smith et al, 2014; Wilkerson et al, 2015) to retrieve similar lines. To be specific, we created an inverted index of word 5-gram hashes and extract all lines from different issues. When fine-tuning, we used the pretrained model with the lowest perplexity in the RDD validation data for initializing the model. We used the model with the lowest perplexity in our validation data for testing, and the beam size was set to 128. We collected person and place names from the Australian Dictionary of Biography,[18] Australia Post,[19] and Ghostly Gazetteer of Australia[20] for calculating the dictionary match score. As a result, we collected $13,127$ person names and $6,071$ place names.

Evaluation Metrics We used WER and CER to evaluate comparison models' performance. In addition to WER and CER, we used a domain-dictionary based evaluation metric $\mathrm{m_{dic}}$. $\mathrm{m_{dic}}$ is calculated as:

$$\mathrm{m_{dic}} = \frac{\mathrm{count}(U_h \cap U_r)}{\mathrm{count}(U_r)}$$

where U_h is the words in the domain-dictionary included in generated text, U_r is the words in the domain-dictionary included in reference text, and count is a function that counts the number of words appeared in text.

[13] https://github.com/kpu/kenlm/
[14] http://code.google.com/p/giza-pp
[15] http://opennmt.net/
[16] https://github.com/Doreenruirui/ACL2018_Multi_Input_OCR
[17] http://dlxs.richmond.edu/d/ddr/
[18] http://adb.anu.edu.au/
[19] https://auspost.com.au/
[20] http://www.let.osaka-u.ac.jp/seiyousi/Ghost-Gazetteer/index.htm

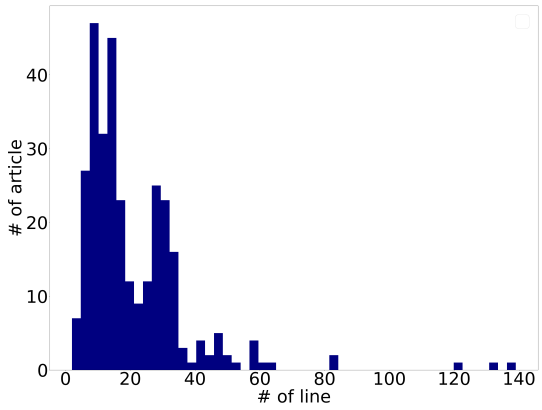

Fig. 7 Line number distribution of the ground-truth article data.

5.2 Corpus Construction

5.2.1 Data

We manually created the ground-truth data for "public meeting" articles in advertisement pages, in order to evaluate the accuracy of article extraction. As OpenCV cannot handle PDF files, we converted the PDF data of "public meeting" advertisement pages to PNG with ImageMagick.[21]

In our experiments, we manually extracted 307 articles about "public meeting" in advertisement pages spanning from 1838 to 1954, and split them into 149 and 158 articles for validation and testing, respectively. Figure 7 shows the line number distribution of the ground-truth data used for our evaluation. We can see that most ground-truth articles contain less than 60 lines, but there are also exceptions.

5.2.2 Comparison

We compared the following methods in our experiments:

- **Baseline**: We compared a baseline method, which is based on text features to identify articles from OCR error corrected text.[22] The baseline method extracts features from the beginning and ending sentences of articles for article identification. The features are as follows:
  - Beginning sentence: Take 2 sentences before the sentence that contains "public meeting."

[21] `https://www.imagemagick.org/`

[22] Note that here, we applied our SMT OCR correction model to the OCRed text provided by Trove. Therefore, this baseline can be treated as an improved version on how good we can get "public meeting" articles from the advertisement pages using the functions provided by Trove.

  - Ending sentence: We first apply named entity recognition using the Stanford parser.[23] Then we take the sentence containing LOCATION, DATE, PERSON tags, but the following sentence that does not contain these tags as the ending sentence.
- **Baseline w/o Correction**: Apply the Baseline feature extraction method directly on the OCRed text provided by the Trove website.
- **Proposed**: This is our proposed method presented in Section 4.
- **Proposed w/o Correction (Tanaka et al, 2020)**: This is our proposed method presented in Section 4, but does not apply OCR error correction, i.e., our previous method (Tanaka et al, 2020).
- **Baseline+Proposed**: Use Proposed for articles that the Baseline fail to extract.[24] Baseline fails to extract articles when the ending sentence corresponding to the beginning sentence is not found.
- **Baseline+Proposed w/o Correction (Tanaka et al, 2020)**: Use Proposed w/o correction for articles that the Baseline w/o Correction fail to extract. This is another method used in our previous work (Tanaka et al, 2020).

5.2.3 Parameter Tuning

To tune the thresholds for the rule line and small column identification described in Section 4.1, we used the advertisement data spanning in one month and determined the thresholds empirically. For the threshold used for filtering as described in Section 4.4, we tuned it on the validation data and chose the one achieving the highest F-score. We tuned the threshold from 0 to 1 with an increment of 0.05, and it turned out that 0.8 was the best and thus we used the threshold of 0.8 for filtering.

5.2.4 Evaluation Methods

In our experiments, we conducted an article level evaluation to check if the articles were successfully extracted. In addition, we also conducted a line level evaluation to verify the accuracy for article extraction. These two evaluation methods are described as follows:

Article level evaluation method
: We calculated the similarity following Equation (2) between the sentences containing the keyword "public meeting" in the extracted article and ground-truth article, respectively. Note that for methods with OCR error correction, we calculated the similarity on ground-truth articles after OCR error correction; while for methods without OCR error correction, we used the ground-truth articles with OCRed text to calculate the similarity. If the similarity is higher than a threshold then the extraction is evaluated as success, otherwise it is failure. The threshold was empirically determined to be 0.6. Then we calculated the precision, recall, and F-score for the baseline and proposed methods.

[23] `https://nlp.stanford.edu/software/lex-parser.shtml`

[24] As advertisement pages in our data contain the key phrase "public meeting," there must be target articles to be extracted. If no articles are extracted by Baseline from a advertisement page, we treat it as a failure.

	WER (%)	CER (%)	$m_{dic}(\%)$
OCRed	26.57	9.68	80.64
SMT	**3.14**	**0.61**	**99.19**
NMT	18.4	8.72	84.68
Pre (Dong and Smith, 2018)	22.70	8.79	86.29
Ours: Pre+FT	14.07	5.53	88.71
Ours: Pre+RR	22.67	8.84	87.10
Ours: Pre+FT+RR	14.16	5.56	89.11

Table 2 Results of OCR error correction on our dataset described in Section 3.1.

Line level evaluation method

We compared the beginning and ending lines of the extracted articles to the ground-truth articles to investigate the difference. Then we calculated the ratio of excess and deficiency lines between the extracted and ground-truth articles.

6 Results

6.1 OCR Error Correction

Table 2 shows the evaluation OCR error correction results on our dataset described in Section 3.1. Ours: Pre+FT, Ours: Pre+RR, and Ours: Pre+FT+RR denote three different combinations of pretraining (Pre), fine-tuning (FT), and reranking (RR) of our proposals for the semi-supervised NN-based model. We can see all models show better performance compared to the OCRed text, and SMT achieves the best scores for all metrics. Zoph et al (2016) showed that in low-resource MT, SMT outperforms NMT. We think that the same phenomenon occurs in our OCR error correction experiments. Comparing the semi-supervised NN-based models, Ours: Pre+FT achieves the best scores for both WER and CER. Ours: Pre+FT also achieves better scores than NMT in both WER and CER, indicating that error correction can be more accurate by fine-tuning on our dataset after pre-training on the open domain dataset, compared to directly training on our dataset with NMT. Therefore, in the case of domain-specific OCR error correction, it is better to create a small amount of data and train SMT, or fine-tune the model pretrained on OCRed only datasets with NMT for comparison. In the domain-dictionary based metrics $\mathrm{m_{dic}}$, Ours: Pre+FT+RR gives the best score among the NN-based models. Therefore, reranking is effective for OCR error correction for domain-specific proper nouns.

Table 3 shows some OCR error correction examples of SMT and Ours: Pre+FT+RR. The first example shows that SMT can correct serious errors in the OCRed text. On the other hand Ours: Pre+FT+RR cannot correct them. Therefore, it can be seen that SMT is robust against terrible errors. The second example shows that both SMT and Ours: Pre+FT+RR can correct a word in the domain-dictionary. The red words in Table 3 indicate the word in the domain-dictionary. Although SMT does not use a domain-dictionary, SMT can correct the proper noun “Federal.” We can see that SMT

	OCRed	SMT	Pre+FT+RR	References
1	June 30, 1851.'$E flAILY TEfcE-GRAWT	June 30, 1851.DAILY TELE-GRAPH	WALLY TEKEGRAW.	June 30, 1851. DAILY TELE-GRAPH
2	the'F&Ieral Bill. Ladies iiivrfE?3.-..	the Federal Bill. Ladies invited	the Federal Bill. Ladies invessed	the Federal Bill. Ladies invited
3	NOYVETHj.of Osborne Flat, near Y'ackan-	NOWETH, of Osborne Flat, near Yackan	NOWVETRY of Osborne Flat, near Yackan-	NOWETH, of Osborne Flat, near Yackan-
4	etreet ; Mr Duncan's, Georgt-jireeL; or at	street, Mr Duncan's, George-street, or at	street; Mr Duncan's, Georgia, or at	street; Mr Duncan's, George-street; or at

Table 3 Examples of SMT and Pre+FT+RR. The red word indicates the word in the domain-dictionary. The blue word indicates the word which SMT mistook to correct.

is also robust against domain-specific words. 31.2% of SMT mistakes are cases that do not include symbols in the generated sentences, as shown in the third example. 12.6% of SMT mistakes are cases that generate a symbol different from the reference as shown in the fourth example. However, we think that these errors will not significantly affect document extraction and analysis. We recalculated WER ignoring these symbol-based errors, and SMT achieved a 1.51% WER.

6.2 Corpus Construction

6.2.1 Article Level Evaluation

Table 4 shows the results for article level evaluation. We can see that Baseline+Proposed has the highest F-score among all methods. This is because Proposed correctly extracts "public meeting" articles, which are failed to be extracted in Baseline. However, Baseline has a higher F-score than Proposed. The reason for this is that Baseline uses the feature of a sentence that includes "public meeting" when getting the first lines of the article, and thus the sentence used for article level evaluation is extracted. However, there are still many failures in the extraction of Baseline. This is because firstly there can be multiple "public meeting" articles in an advertisement image, secondly there are OCR errors about the keyword "public meeting." Also, Proposed has a low precision. This is because the extracted article by Proposed is OCRed by Google Drive, but the ground-truth article is OCRed by Trove. This difference causes the similarity between extracted articles by Proposed and ground-truth articles to be lower, which decreases the precision. After combining Proposed with the Baseline, the article extraction results are improved significantly.

As for the effect of OCR error correction for article extraction, all the methods with OCR error correction show higher F-scores than the methods without OCR error correction. The improvements are 2.6%, 2.5%, and 1.7% F-scores for Baseline, Proposed, and Baseline+Proposed, respectively. Therefore, we can conclude that OCR error correction is very effective for our corpus construction task.

Method	Precision	Recall	F-score
Baseline w/o Correction	**76.1**	56.3	64.7
Baseline	71.1	63.9	67.3
Proposed w/o Correction (Tanaka et al, 2020)	59.4	51.9	55.4
Proposed	61.9	54.4	57.9
Baseline+Proposed w/o Correction (Tanaka et al, 2020)	53.1	91.1	67.1
Baseline+Proposed	54.2	**93.7**	**68.7**

Table 4 Article extraction evaluation results in article level.

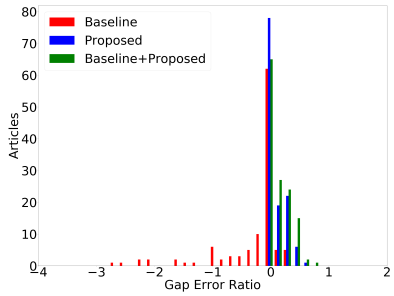

Fig. 8 Line level evaluation results (beginning line).

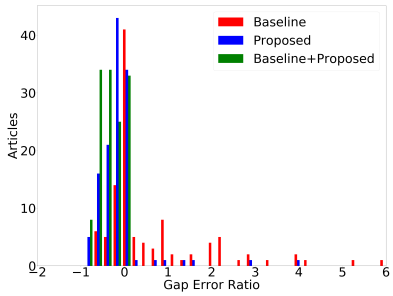

Fig. 9 Line level evaluation results (ending line).

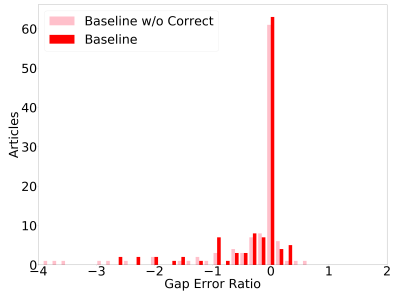

Fig. 10 Line level evaluation results for Baseline w/o Correction and Baseline (beginning line).

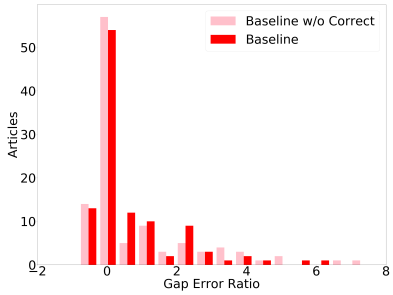

Fig. 11 Line level evaluation results for Baseline w/o Correction and Baseline (ending line).

6.2.2 Line Level Evaluation

Figures 8 and 9 show the line level evaluation results for the beginning and ending lines, respectively. The horizontal axis represents the gap ratio of the number of the excess and deficiency lines against the entire number of lines in an article. The vertical axis represents the number of articles. We can see that on both the beginning and ending lines, Proposed extracted significantly more articles without excess and deficiency than Baseline. In addition, for the case of articles without excess and de-

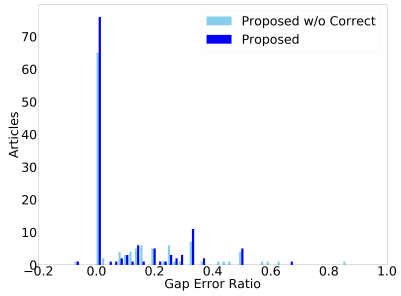

Fig. 12 Line level evaluation results for Proposed w/o Correction and Proposed (beginning line).

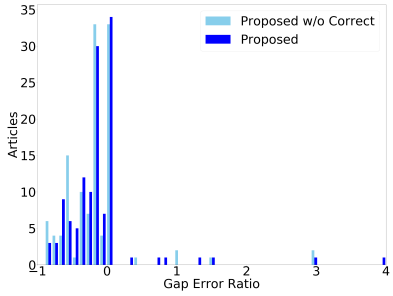

Fig. 13 Line level evaluation results for Proposed w/o Correction and Proposed (ending line).

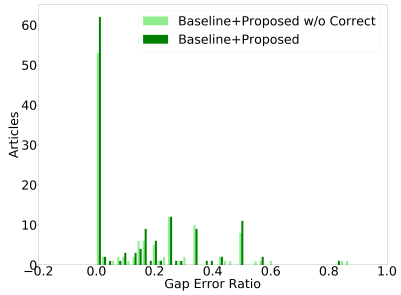

Fig. 14 Line level evaluation results between Baseline+Proposed w/o Correction and Baseline+Proposed (beginning line).

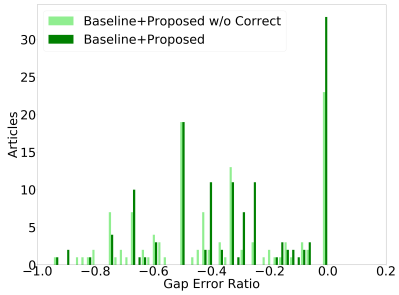

Fig. 15 Line level evaluation results between Baseline+Proposed w/o Correction and Baseline+Proposed (ending line).

ficiency in both the beginning and ending lines, Baseline only successfully extracted 13 (8.2%) articles but Proposed extracted 38 (24.1%) articles. Therefore, we can say that the proposed method that uses visual features to identify the article split, is more effective for extracting articles with specific topics. In the beginning line, Baseline+Proposed extracted more articles without excess and deficiency than Baseline. This is because Baseline+Proposed includes articles that could not be extracted by Baseline but could be extracted by Proposed. In the ending line, Baseline+Proposed is also helpful for preventing extracting articles with large error gap ratios compared to Baseline and Baseline+Proposed.

Figures 10 and 11 show the line level evaluation results in the beginning and ending lines for Baseline w/o Correction and Baseline. We can see that in the beginning line, OCR error correction is effective for extraction, but it is not effective in the ending line. We analyzed and found the reason for this was that OCR error correction incorrectly replaced proper nouns with other words, making them not being proper nouns anymore. Because we use named entities as the features for the

before correction	after correction
usual Publie Meetide of	usual Public Meetide of
# PUILIO MEETING of	#PUILIOMEETING of

Table 5 Examples of success to extract and failure to extract due to OCR error correction.

ending line identification in the baseline method, the incorrect replacement of proper nouns affect the results. For the case of articles without excess and deficiency in both the beginning and ending lines, Baseline w/o Correction extracted 9 (5.7%) articles. As such, Baseline can extract more than Baseline w/o Correction without excess or deficiency.

Figures 12 and 13 show the line level evaluation results in the beginning and ending lines for Proposed w/o Correction and Proposed. We can see that in both of the beginning and ending lines, OCR error correction is effective for extraction. The reason for this is that before applying OCR error correction it is difficult to align lines for evaluation due to differences in the OCR systems, i.e., Google Drive is used for Proposed, but Trove OCR is used for ground-truth articles; applying OCR error correction makes it easier to align lines for evaluation. For the case of articles without excess and deficiency in both the beginning and ending lines, Proposed w/o Correction extracted 24 (15.2%) articles. Therefore, OCR error correction is also effective for line level accuracy improvement for Proposed. Figures 14 and 15 show the line level evaluation results in the beginning and ending lines for Baseline+Proposed w/o Correction and Baseline+Proposed. We can see that it significantly improves both the beginning and ending line results. The reason for this is the boost from both the Baseline and Proposed by OCR error correction.

6.2.3 Discussion

Table 5 (upper) shows an example where Proposed succeeds to extract an article because of OCR error correction. Before OCR error correction, an OCR error "publie meetide" exists, and extraction fails. By applying OCR error correction, "publie" is corrected to "public", making extraction successful. 3.8% of the articles are successfully extracted after OCR error correction.

On the other hand, Table 5 (bottom) shows an example where Proposed fails to extract the article after OCR error correction. Before OCR error correction, an OCR error "PUILIO MEETING" exists, but the similarity with "public meeting" is high. However, by applying OCR error correction, it is corrected as "#PUILIOMEETING", so the similarity with "public meeting" becomes low, and thus extraction fails. 1.3% of the articles are failed to be extracted due to OCR error correction mistakes.

Articles that fail to be extracted by Baseline+Proposed are the ones that do not match the features used in Baseline or are cut across columns by trimming. Figure 16 (left) shows examples that Baseline+Proposed failed to extract due to trimming mistakes. The phrase "public meeting" is cut off by unexpected trimming. The unexpected trimming is due to the horizontal line on the left. Therefore, the phrase "public meeting" could not be detected in the filtering process. Figure 16 (right) shows an ex-

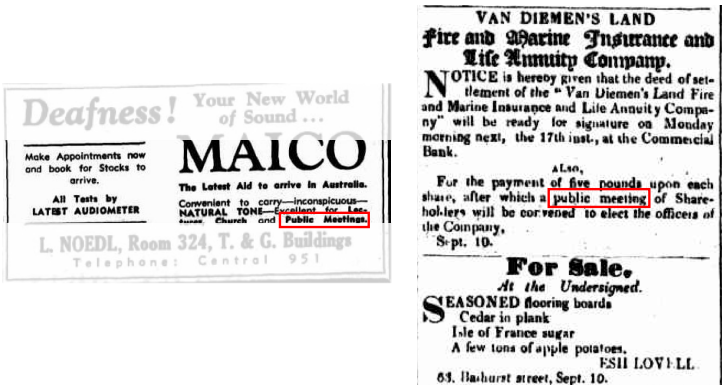

Fig. 16 Examples of failed extraction by Baseline+Proposed due to trimming mistakes. The red rectangle indicates where "public meeting" is written. The lightly colored part indicates the part cut out by trimming.

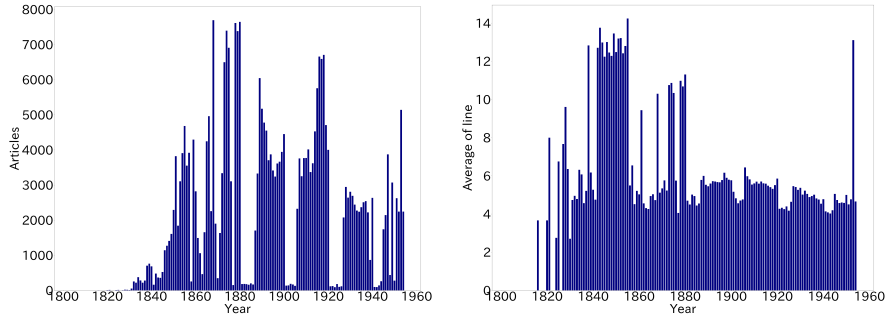

Fig. 17 Statistics on the number of articles per year. **Fig. 18** Statistics on the average of lines per year.

ample that Baseline+Proposed extracted an extra article. Extra extraction is caused by the failure to detect the horizontal line in the middle. 6.3% of the articles are failed to be extracted due to such reasons. Therefore, we believe that the accuracy of extraction can be further improved by improving the trimming accuracy of Proposed.

6.3 Statistics of the Constructed Corpus

We first searched the keyword "public meeting" in advisement pages using the search function in Trove to get all the 407,756 advisement pages in PDF. After that, we applied our corpus construction method, and extracted "public meeting" articles in the advisement pages from 1804 to 1954. In the constructed corpus, the total number of articles is 305,821, and the total number of lines is 2,181,869, the total number of vocabulary is 2,496,765. Note that the number of "public meeting" articles are less than the entire number (which is around 1.7M results) that we can get from Trove via searching the keyword "public meeting." This is because our targeted articles are only the ones in the advertisement pages, but not in all newspaper pages. Figure 17

shows statistics on the number of articles per year in the constructed corpus. We can see that "public meeting" articles begin to be published actively from 1840, and the number of articles reaches the maximum from 1860 to 1880, and gradually decreases from 1880 to 1960. Figure 18 shows statistics on the average of lines per year in the constructed corpus. Until 1860, the constructed corpus has many articles with a large number of lines; on the other hand, since 1860, it has many articles with a small number of lines. This is because newer articles tend to write only the information they want to convey.

7 Conclusion

In this paper, firstly, we constructed an OCR error correction dataset for the "public meeting" articles in the advertisement pages of Australian historical newspapers in order to adapt an OCR correction method to that domain. We proposed fine-tuning, and reranking with language models and dictionary match scores for the semi-supervised NN-based OCR error correction model, which improved WER and CER by 8.54% and 3.23%, respectively. As a result of comparing statistical and NN-based models on our OCR error correction dataset, SMT achieved 3.14% WER and 0.61% CER. Moreover, 99.2% of domain-specific proper nouns in the references could be corrected accurately. Secondly, we constructed a corpus of "public meeting" articles via image processing, OCR, OCR error correction, and filtering. Experiments conducted on the advertisement data from Trove indicated that the proposed method can successfully extract 93.7% of the targeted articles and 24.1% of the extracted articles are without excess and deficiency. In addition, our proposed method improved 1.7% F-score by applying OCR error correction comparing to previous work.

As future work, we plan to conduct automatic analysis of the constructed "public meeting" corpus by identifying the participants, times, places, and purposes of "public meetings" in a historical span.

Acknowledgements This work was supported by Grant-in-Aid for Scientific Research (B) #19H01330, JSPS.

Conflict of Interest Statement

The authors declare that they do not have a financial or personal relationship with a third party whose interests could be positively or negatively influenced by the article's content.

References

Afli H, Barrault L, Schwenk H (2015) OCR error correction using statistical machine translation. International Journal of Computational Linguistics and Applications 7(1):175–191

Afli H, Qiu Z, Way A, Sheridan P (2016) Using SMT for OCR error correction of historical texts. In: Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016), pp 962–966

Barbaresi A (2016) Bootstrapped OCR error detection for a less-resourced language variant. In: 13th Conference on Natural Language Processing (KONVENS 2016), pp 21–26

Barrault L, Bojar O, Costa-jussà MR, Federmann C, Fishel M, Graham Y, Haddow B, Huck M, Koehn P, Malmasi S, Monz C, Müller M, Pal S, Post M, Zampieri M (2019) Findings of the 2019 conference on machine translation (WMT19). In: Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1), pp 1–61

Cassidy S (2016) Publishing the Trove newspaper corpus. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp 4520–4525

Chiron G, Doucet A, Coustaty M, Visani M, Moreux JP (2017) Impact of OCR errors on the use of digital libraries: Towards a better access to information. In: Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries, JCDL '17, pp 249–252

Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp 1724–1734

Chu C, Nakazawa T, Kurohashi S (2015) Integrated parallel sentence and fragment extraction from comparable corpora: A case study on chinese–japanese wikipedia. ACM Transactions on Asian and Low-Resource Language Information Processing 15(2):10:1–10:22

Chung J, Cho K, Bengio Y (2016) A character-level decoder without explicit segmentation for neural machine translation. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp 1693–1703

Davies M (2012) Expanding horizons in historical linguistics with the 400-million word corpus of historical american english. Corpora 7:121–157

Dong R, Smith D (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), pp 2363–2372

Duchi J, Hazan E, Singer Y (2011) Adaptive subgradient methods for online learning and stochastic optimization. J Mach Learn Res 12:2121–2159

Eger S, vor der Brück T, Mehler A (2016) A comparison of four character-level string-to-string translation models for (ocr) spelling error correction. The Prague Bulletin of Mathematical Linguistics 106:77–99

Evershed J, Fitch K (2014) Correcting noisy OCR: Context beats confusion. In: Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage, pp 45–51

Fujikawa T (1990) Public meetings in New South Wales: 1871-1901. Journal of the Royal Australian Historical Society 76:45–61

Kingma D, Ba J (2015) Adam: A method for stochastic optimization. In: International Conference on Learning Representations

Klein G, Kim Y, Deng Y, Senellart J, Rush A (2017) OpenNMT: Open-source toolkit for neural machine translation. In: Proceedings of ACL 2017, System Demonstrations, pp 67–72

Klein S, Kopel M (2002) A voting system for automatic ocr correction

Koehn P (2005) Europarl: A Parallel Corpus for Statistical Machine Translation. In: Proceedings of Machine Translation Summit, pp 79–86

Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, Dyer C, Bojar O, Constantin A, Herbst E (2007) Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, ACL '07, pp 177–180

Kolak O, Resnik P (2002) OCR error correction using a noisy channel model. In: Proceedings of the Second International Conference on Human Language Technology Research, pp 257–262

Kolak O, Byrne W, Resnik P (2003) A generative probabilistic OCR model for NLP applications. In: Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp 134–141

Lund WB, Kennard DJ, Ringger EK (2013) Combining multiple thresholding binarization values to improve OCR output. In: Document Recognition and Retrieval XX, vol 8658, pp 254–264

Lyu L, Koutraki M, Krickl M, Fetahu B (2021) Neural OCR Post-Hoc Correction of Historical Corpora. Transactions of the Association for Computational Linguistics 9:479–493

Marcus MP, Marcinkiewicz MA, Santorini B (1993) Building a large annotated corpus of English: The Penn Treebank. Computational Linguistics 19(2):313–330

Mokhtar K, Bukhari SS, Dengel A (2018) OCR error correction: State-of-the-art vs an nmt-based approach. In: Proceedings of the 2018 13th IAPR International Workshop on Document Analysis Systems (DAS), pp 429–434

Moreno-García C, Elyan E (2019) Digitisation of assets from the oil gas industry: Challenges and opportunities. In: Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), pp 2–5

Moreno-García C, Elyan E, Jayne C (2019) New trends on digitisation of complex engineering drawings 31(6):1695–1712

Neudecker C (2016) An open corpus for named entity recognition in historic newspapers. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp 4348–4352

Och FJ (2003) Minimum error rate training in statistical machine translation. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp 160–167

Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics 9(1):62–66

Radford A, Narasimhan K (2018) Improving language understanding by generative pre-training

Richter C, Wickes M, Beser D, Marcus M (2018) Low-resource post processing of noisy OCR output for historical corpus digitisation. In: Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC-2018), pp 2331–2339

Rögnvaldsson E, Ingason AK, Sigurðsson EF, Wallenberg J (2012) The Icelandic parsed historical corpus (IcePaHC). In: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pp 1977–1984

Sánchez-Martínez F, Martínez-Sempere I, Ivars-Ribes X, Carrasco R (2013) An open diachronic corpus of historical Spanish. Language Resources and Evaluation 47:1327–1,342

Sherratt T (2021) Glam workbench – using the trove newspaper gazette harvester (the web app version)

Smith DA, Cordel R, Dillon EM, Stramp N, Wilkerson J (2014) Detecting and modeling local text reuse. In: IEEE/ACM Joint Conference on Digital Libraries, pp 183–192

Smith R (2007) An overview of the Tesseract OCR engine. In: Proc. of International Conference on Document Analysis and Recognition, vol 2, pp 629–633

Snoek J, Larochelle H, Adams RP (2012) Practical bayesian optimization of machine learning algorithms. In: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2, pp 2951–2959

Suzuki S, Abe K (1985) Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing 30(1):32–46

Tanaka K, Chu C, Ren H, Renoust B, Nakashima Y, Takemura N, Nagahara H, Fujikawa T (2020) Constructing a public meeting corpus. In: Proceedings of The 12th Language Resources and Evaluation Conference, pp 1934–1940

Trad A, Doush I (2016) Improving post-processing optical character recognition documents with arabic language using spelling error detection and correction. International Journal of Reasoning-based Intelligent Systems 8:91

Wilkerson J, Smith D, Stramp N (2015) Tracing the flow of policy ideas in legislatures: A text reuse approach. American Journal of Political Science 59(4)

Xu S, Smith D (2017) Retrieving and combining repeated passages to improve ocr. In: 2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL), pp 1–4

Yamazoe T, Etoh M, Yoshimura T, Tsujino K (2011) Hypothesis preservation approach to scene text recognition with weighted finite-state transducer. In: Proceedings of the 2011 International Conference on Document Analysis and Recognition, pp 359–363

Zoph B, Yuret D, May J, Knight K (2016) Transfer learning for low-resource neural machine translation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp 1568–1575