LIAS: Layout Information-Based Article Separation in Historical Newspapers

Wenjun Sun, Hanh Thi Hong Tran, Carlos-Emiliano González-Gallardo, Mickaël Coustaty, Antoine Doucet

LIAS: Layout Information-based Article Separation in Historical Newspapers

Wenjun Sun[1][0009−0002−7857−8737], Hanh Thi Hong Tran[1,2,3][0000−0002−5993−1630], Carlos-Emiliano González-Gallardo[1,4][0000−0002−0787−2990], Mickaël Coustaty[1][0000−0002−0123−439X], and Antoine Doucet[1][0000−0001−6160−3356]

[1] L3i, University of La Rochelle, La Rochelle, France
{firstname.lastname,thi.tran}@univ-lr.fr
[2] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[3] Jožef Stefan Institute, Ljubljana, Slovenia
[4] LIFAT, University of Tours, Blois, France
carlos-emiliano.gonzalez-gallardo@univ-tours.fr

Abstract. Nowadays, the increasing digitization of historical document resources not only provides researchers with abundant materials but also introduces a new challenge: how to analyze their contents swiftly and accurately. Article separation emerges as a solution to this issue, overcoming the disorder and inefficiency associated with querying the original document material directly. However, there is limited in-depth research on this topic currently. Given that historical documents contain information in multiple modalities such as text and images, the reasonable integration and analysis of multimodal information pose a further challenge. In response to these challenges, we propose LIAS, a method based on layout information, and conduct experiments on historical newspapers. In comparison to existing work, LIAS introduces a fresh perspective to research in this area. The method initially identifies the separator lines of the newspaper, analyzes the layout information to reconstruct the information flow of the document, performs segmentation based on the semantic relationship of each text block in the information flow, and ultimately achieves article separation. The experiments encompass diverse historical newspapers in French and Finnish, featuring various layouts. Results demonstrate that LIAS outperforms current schemes in most cases, as evidenced by metrics specifically designed for article separation tasks.

Keywords: Article separation · Historical documents · Semantic relation· Layout information

1 Introduction

Historical newspapers serve as valuable resources, offering a wealth of historical information to researchers across various disciplines and playing an important role in political and economic studies [8]. However, the automatic extraction of

Fig. 1. Example of article separation on a newspaper page of the NewsEye dataset. Distinct colors indicate different articles.

information from these newspapers remains an unresolved challenge. This difficulty arises primarily from the analysis of deteriorated documents due to the passage of time and the distinctive layout of historical newspapers, which differs from contemporary counterparts that have been the primary focus of existing research. In the context of historical newspapers, information is typically organized in the form of articles. Consequently, the primary task involves separating the entire document into distinct articles corresponding to individual news items as exemplified in Figure 1 where the articles of a Finnish 19^{th} century newspaper are highlighted.

Recent research on information extraction using layout information has focused on a token- and document-level showing big advances in tasks like named entity recognition (NER) and document classification [12,13,17,18], nevertheless almost no research has focused on article-level information extraction of historical texts which can simultaneously reduce redundancy of information compared to token-level and provide more detailed information compared to document-level. Moreover, historical newspapers contain both visual and textual information which implies that the newspaper's visual layout and the text's location are also semantically rich. Current research does not manage to integrate both types

of information, most of the approaches are overly focused on token-level semantic and visual comprehension, which results in pre-trained multimodal document comprehension models that lack sufficient generalization ability when facing different layouts from the training corpus. Even though some methods have been developed to try to rearrange scattered tokens to obtain more explanatory semantics [11], all of them have difficulties in modeling the complete information flow of the document. As a result, the acquired textual information is not reasonably organized, thus limiting the understanding ability of these methods at the article level.

Textual content-wise, the defacement of historical newspapers coupled with the constraints of optical character recognition (OCR) technology, poses a challenge in obtaining precise semantic embeddings directly from visual information [2]. In addition, the spelling and semantics gap between text from historical and contemporary newspapers impacts the language model representations, leading to unsatisfactory results in information retrieval tasks [5,9]. This makes it difficult for current models trained with contemporary documents to be capable of embedding historical newspapers' text semantics and makes it necessary to process visual and textual information in a new way, rather than directly relying on existing schemes proposed for clean contemporary corpora.

Building upon existing models and acknowledging their limitations, we introduce LIAS, a layout-oriented method for article separation consisting of three principal components [5]. Initially, the text portion of a historical newspaper document is filtered by pixel expansion and erosion, and only the linear objects (separator lines) are retained. Subsequently, the layout is modeled using these separator lines, paragraphs are grouped, and the information flow of the document is simulated in a manner that mirrors human reading order. Finally, a neural network-based classifier is deployed to analyze the resulting layout and determine segmentation points, thus partitioning the entire document into distinct articles. This method introduces a novel approach to research in article-level information extraction from historical texts, primarily focusing on modeling the sequential flow of information within a document. Its notable advantage lies in its increased adaptability compared to existing approaches. Rather than embedding layout information directly into the semantic content of the text, it models the overall reading order information for each semantic unit, enhancing flexibility and applicability. When faced with a distinct layout or different reading order, the need for adaptation is addressed efficiently by redefining the information flow simulation process, eliminating the requirement for retraining or fine-tuning the entire model. We conducted experiments on the Newseye dataset [7] and in comparison to current approaches, our results suggest that LIAS demonstrates competitive and promising performance. Overall, article-level segmentation for historical documents is still an area that lacks a mature solution.

The remainder of the paper is organized as follows: Section 2 provides a review of existing research in the field of text semantic segmentation. Section 3

[5] The code is available in the code repository `https://github.com/WenjunSUN1997/seperator_ar_sep`

describes the model architecture and offers a comprehensive description of the accompanying data. The experimental setup and evaluation metrics are detailed in Section 4. Results and discussion are presented in Section 5. The paper concludes with Section 6, summarizing key findings, explaining the limitations, and suggesting potential directions for future research.

2 Related Work

Documents inherently comprise two modalities of information: text and images. Consequently, both modalities must be considered when analyzing document information. Initially, efforts have been made to incorporate location information into text semantics directly using the embedding layer [18], yielding satisfactory results in NER. Building upon this, researchers have explored enhancing the model's comprehension by establishing communication between location and text semantics generation [16]. Since obtaining logically structured input text using only the location information of tokens from OCR is challenging, attempts have been made to improve model input quality by correcting the order of the output from OCR [11]. In parallel, visual embedding has been introduced into document understanding. Initially, this involved embedding pictures of tokens [17], followed by training models to align text and images [13] or make predictions about masks through self-supervised training [12]. However, it's essential to note that these works predominantly focus on token-level understanding.

While research in contemporary document understanding is advancing, there is a growing interest in the exploration of historical texts. A primary challenge in this domain is the issue of semantic changes in historical texts. Since the training corpus has a great influence on the pre-trained language models, many experimental results indicate that language models trained on contemporary corpora face difficulties in accurately capturing semantics in historical texts [6]. To address this, some works propose further processing token embeddings by overlaying an encoder on top of the language model [2]. Alternatively, obtaining a fine-tuned model using a historical corpus has also shown success, as demonstrated by [15], particularly in NER.

Efforts have also been made to analyze the visual properties of historical documents for structural layout understanding [1]. However, article separation in historical documents remains a complex issue with limited existing research. Some methods leverage graph neural network approaches to analyze relationships between text blocks and use the final clustering results for article-level classification [3]. Nevertheless, this method may struggle to construct a reasonable text block diagram based on layout information. Another approach, independent of layout, employs a completely rule-based method for article classification by categorizing text blocks and comparing semantic similarity [8]. While this method has limitations and may not yield desired results when confronted with new data, it represents an existing avenue of exploration in historical document analysis.

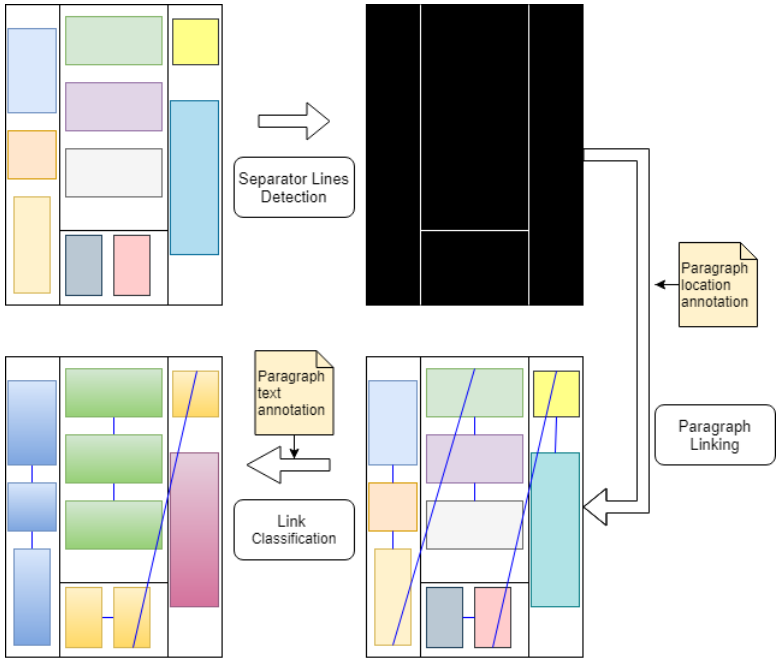

Fig. 2. The three-step procedure of the LIAS method.

3 The LIAS Model

We approach article separation of historical documents by analyzing the semantic relationships among all text blocks in the document. Beyond merely conveying textual information, these text blocks also encapsulate layout features. Thus, leveraging both modalities becomes necessary to acquire a comprehensive understanding of the semantic content within text blocks. Our method initiates by employing separator lines to deduce the document's layout information. Subsequently, we establish positional relations between text blocks, creating a simulated representation of the information flow across the entire document. The pivotal step in this process involves identifying segmentation points in the information flow, thereby achieving the separation of articles. Text blocks that keep linked together form an article. Based on this idea, we propose LIAS.

The LIAS method presented in Figure 2 is composed of a three-step procedure. The **separator lines detection** module obtains the separator lines from a digitized historical newspaper page and segments it into regions. The **paragraph linking**, in turn, groups the paragraphs of the newspaper page according to the regions and separator lines, determines their reading order, links the paragraphs, and constructs the document information flow based on the location annotation of each paragraph. Lastly, with the paragraph text annotations, the **link classification** module determines whether the links between paragraphs

need to be maintained or removed, i.e., whether the paragraphs at both ends of the link belong to the same article. As a result of this process, the paragraphs that are still linked together are recognized as forming an article, and the whole newspaper page is segmented on the article level. Figure 6 exemplifies the result of the three-step procedure on two historical newspaper pages. In the subsequent sections, we delve into the specifics of each module within this three-step procedure.

3.1 Separator Lines Detection and Region Segmentation

The complex layout of historical newspapers makes it not feasible to directly link paragraphs in a left-to-right top-to-bottom order. It is then necessary to first split the newspaper page into different regions using the existent separator lines and then analyze the paragraphs within the sub-regions. Separator lines are visual segments of text in newspaper pages that are important for determining the reading order. Added to the layout complexity, the noise introduced during the digitization process makes it necessary to highlight the existent separator lines. We thus perform binarization, pixel expansion, and erosion operations on the newspaper images, so that the text zones are filtered out and only the segmentation lines are retained.

One of the most common methods for detecting separator lines is the Hough Transformation which fits straight lines and curves by spatial transformation and voting mechanism [4]. Nevertheless, it shows complications when the lines are broken or tilted, as in newspaper pages, and tends to segment a long line into different pieces. Neural networks have been used to detect separator lines as part of layout detection models in which the separator lines correspond to one of multiple possible labels, complicating the training process. Considering that this first module is focused on detecting separator lines and is intended to be as lightweight and generalizable as possible, we opted for a rule-based approach capable of analyzing long lines with breaks and tilts.

The separator line detection process includes two scanning cycles to obtain first, page-level separators (outer cycle) and second, region-level separators (inner cycle). The outer cycle employs two sliding windows to scan the output image of the former pixel operations both horizontally and vertically regarding the newspaper page as a pixel matrix. For each window during the horizontal scan, a column is selected when white pixels appear in that column. Ultimately, the number of selected columns in the window and a proportional threshold are used to determine whether the window is a separator line or not. If the number of selected columns is higher than the threshold, this window is kept as an horizontal separator line. In the example of Figure 3, the first five columns of the zoomed window will be selected. The same process is performed for the vertical scan, where a window is kept as a vertical separator line if the number of selected rows is higher than the threshold. An special case of page-level separators are the four border lines that conform the image boundary. The inner cycle follows the same principle but it operates inside the regions delimited by the intersections of horizontal and vertical page-level separators. Finally, an index is

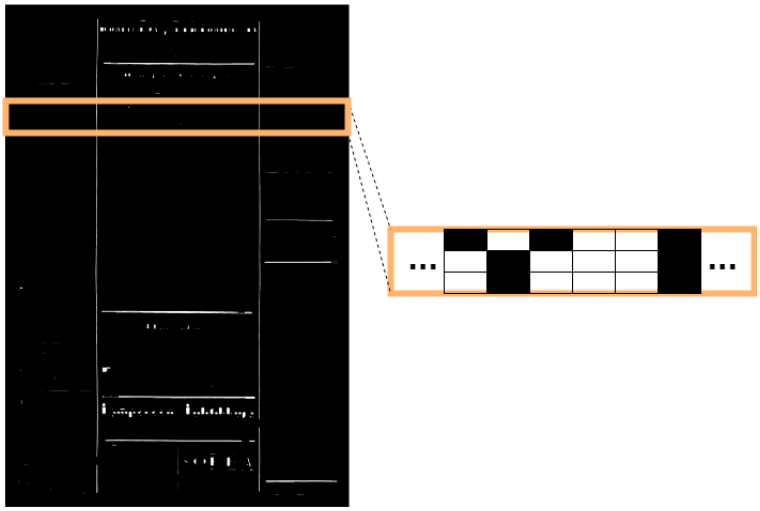

Fig. 3. Example of the pixel operations' output and separator line detection window.

assigned to each page- and region-level separator and the whole page is divided into different areas. An example is shown in Figure 4 (left) where red and green lines denote the horizontal and vertical separators respectively, and the numbers are their index. The sliding window method introduces a slight misalignment between the identified separators and their actual positions. Nevertheless, this misalignment does not impact the final result, as subsequent modules rely on the centroid coordinates of the paragraph rather than the distance between the centroid coordinates and the separator. This approach prevents any incorrect paragraph assignments despite the small misalignment.

3.2 Paragraphs Grouping and Linking

This module starts by grouping each paragraph depending on the regions obtained in the previous step. Then, according to their bounding box, it obtains the coordinates of the centroid of the paragraphs. Finally, given to the centroid, it gets the index of the region where it is located. The index of the region is constructed in the order of (*left*, *right*, *up*, *down*) using the index of the separator. For example, the index of the region of the paragraph in blue of Figure 4 (left) is (9, 10, 0, 5). Then, the paragraphs that share the same region index are linked in the order of centroid coordinates from left to right and from top to bottom.

Newspaper layouts typically feature multiple columns, resulting in a scenario where a single article may span the end of one column and the beginning of the next. Consequently, paragraphs in different regions may potentially belong to the same article, exemplified by the two purple-highlighted paragraphs in Figure 4 (left). As a result, besides linking paragraphs within the same region, the two paragraphs closest to the two vertices of each vertical separator line are

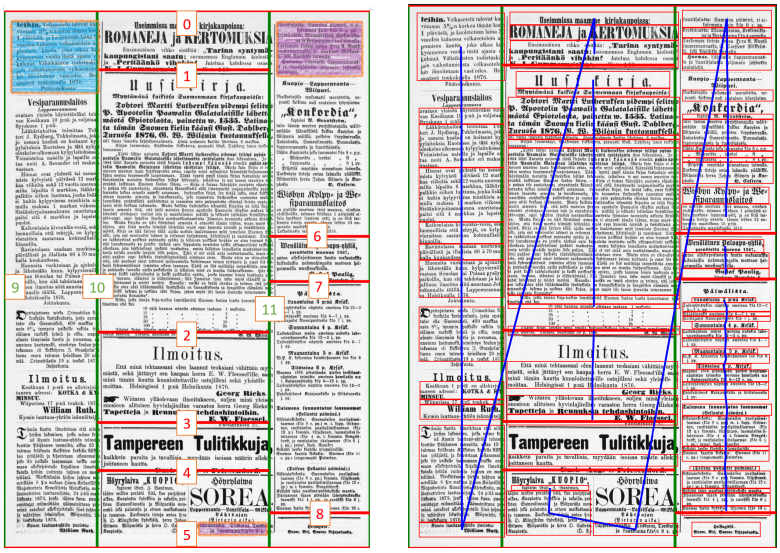

Fig. 4. Example of separator lines (left) and paragraph linking (right).

also linked. The resulting linkages are illustrated in Figure 4 (right), where the blue lines represent the connections between paragraphs.

3.3 Link Classification

Since not all of the linked paragraphs belong to the same article, it is necessary to determine whether the link needs to be kept or removed according to the semantic information of the paragraphs at both ends of the link. Additional encoder layers have been shown to mitigate semantic bias and OCR recognition errors in historical documents [2], we thus follow the same approach in processing token semantics. Initially, a backbone language model is employed to embed the tokens within the paragraph. Subsequently, an additional encoder is utilized to further process the raw output. Finally, to capture the semantics of the entire paragraph, an average pooling operation is performed. The link classification problem can be viewed as a text segmentation task for local content. We adopt the feature extraction methodology from the state-of-the-art [14] on this task to characterize the links, which utilizes the concatenation between two semantic vectors and the modulus of their difference. The final link features are fed into a linear layer head to effectuate the binary classification, determining whether the link should be retained or removed. In addition to the textual information, the positional information of the paragraph is included to obtain its semantics. An embedding layer is used to convert the centroid coordinates of the paragraph into a vector. The average value of this embedding is then added to the average vector representing the text semantics, providing a comprehensive description of

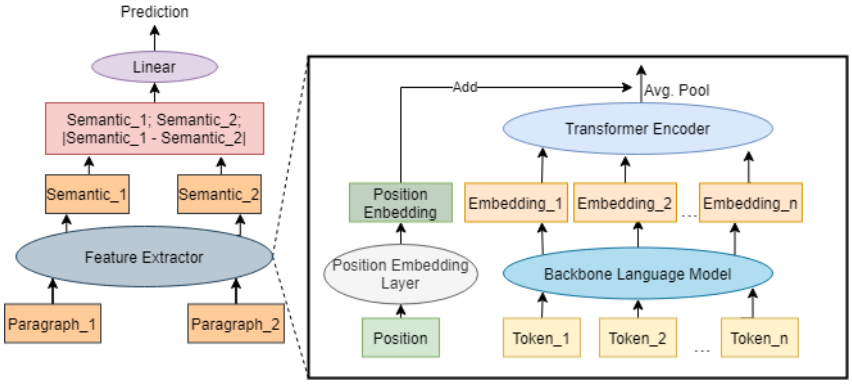

Fig. 5. The structure of the link classifier.

the entire paragraph's semantics. The structure of the link classifier is detailed in Figure 5.

4 Experiments

We evaluated the performance of our model on the Newseye dataset [7], employing metrics specifically designed for article separation to assess the effectiveness of our approach.

4.1 Dataset

The Newseye dataset comprises newspapers dating from the 19^{th} to the early 20^{th} century in French and Finnish, annotated at both paragraph and article levels. The newspapers are sourced from the national libraries of France (BNF), Finland (NLF), and Austria (ONB). In our experiments, we focused on the BNF and NLF sections of the benchmark, which statistics, including the number of pages, sentences, paragraphs, and articles are presented in Table 1.

Table 1. Statistics of the NLF and BNF sections of the Newseye dataset.

Dataset	Pages	Sentences	Paragraphs	Articles
NLF	200	22,042	6,348	3,282
BNF	182	50,698	6,792	3,061

4.2 Metrics

Five metrics were considered to comprehensively evaluate the performance of our pipeline compared to the benchmarks, including mean article coverage score

($mACS$), mean proper predicted articles ($mPPA$) [8], precision (AR_P), recall (AR_R), and F1-score (AR_{F1}) [10].

$mACS$ and $mPPA$ were proposed in the STRAS method [8] and serve to evaluate the article separation system. Given n articles, the $mACS$ is calculated as:

$$mACS = 1 - \sum_{x=1}^{n} \frac{AER_x}{n} \tag{1}$$

where

$$AER_x = \frac{|PTP_x \oplus GTP_x|}{|PTP_x \cup GTP_x|} \tag{2}$$

is the article error rate of the article x (x means the index of each article), PTP_x the predicted paragraphs of x and GTP_x the ground truth paragraphs of x.

Regarding $mPPA$, given P pages of a newspaper, m ground truth articles and n correct predicted article per page, then $mPPA$ is defined as:

$$mPPA = \frac{\sum_{p=1}^{P} \frac{n}{m}}{P}. \tag{3}$$

AR_P, AR_R, and AR_{F1} are the three metrics used to describe the document text line detection and article separation performance [10]. Given the ground truth GT composed of M articles and the predictions PT composed of N articles, the $M * N$ evaluation matrix Eva is formed. For the AR_P matrix, the element on row i and column j is the precision value between GT_i and PT_j. Similarly, for the AR_R matrix, the elements are the recall value between each GT and PT. Finally, the AR_P and AR_R are calculated using the following greedy algorithm:

Algorithm 1 Greedy Algorithm

```
Form matrix Eva ∈ R^{M*N}
Result ← []
while Eva is not empty do
    max ← one of the maximal elements in Eva
    Add max into Result
    Eva ← take Eva and delete corresponding row and column of max
end while
return Avg(Result)
```

The AR_{F1} is the harmonic mean of AR_P and AR_R, which is formulated as:

$$AR_{F1} = \frac{2 * AR_P * AR_R}{AR_P + AR_R} \tag{4}$$

4.3 Benchmarks

Two benchmarks were considered, including Newseye [3] and STRAS. Both methods are proposed for the task of article separation and have so far obtained the best results on the Newseye dataset.

The Newseye method leverages a hybrid technique combining Graph Neural Networks (GNNs) and layout information. It initially creates a structural graph representation of the newspaper document by analyzing the geometric relationship of paragraphs using Delaunay triangulation. Then, the semantic information is extracted from each paragraph by GNNs. Ultimately, different clustering algorithms are applied to effectively separate articles.

In contrast, the STRAS method employs a layout-independent approach based on a rule set. It categorizes paragraphs into independent and non-independent, then further classifies the no-independent paragraphs into source, and non-source groups. Finally, it reorganizes paragraphs based on semantic similarity to determine the final article division.

4.4 Detailed Setup and Hyperparameters

For all experiments, the threshold for image binarization is set to 200, and two morphological dilating operations are performed on the binarized image with kernel sizes ($\sqrt{width} * 1.2$, 1) and (1, $\sqrt{height} * 1.2$), each operation is performed only once. The size and step of the sliding window are both set to 60 pixels. The threshold for the separator line detection is set to 75% of the length and width of the input page. In the link classifier, hmBERT [15] is used as the backbone language model. The encoder of the classifier is composed of 2 transformer layers and 4 attention heads per layer. During training, the batch size is 8, the learning rate is set to 5e-5, drop out is 0.1, and the cross-entropy loss function is used.

5 Results and Discussion

We made a comparative evaluation between LIAS and the benchmarks aforementioned. Results are reported in Table 2, results not shown in the original paper are replaced by –. $LIAS_{ideal}$ serves as a benchmark for evaluating the performance of our model's first two steps, which involve separator lines detection and paragraph linking. It represents the theoretical maximum performance attainable by our model, assuming a perfect link classifier that can categorize all links without error. $Newseye_{[dbscan,greedy,hierarchecal]}$ refer to the results of article separation achieved by employing the corresponding clustering approaches after extracting paragraph semantics using GNNs. $STRAS_{[pre,sg,cbow,ft]}$ denote the STRAS method performance using the fine-tuned Spacy, skip-gram, continuous bag-of-words, and fast-text to capture semantic similarity between paragraphs under the same rule set. *LIAS* indicates the results of inputting text only while *LIAS+bbox* is for the results of inputting both text and position.

The results of $LIAS_{ideal}$ indicate that if paragraph links were classified accurately, our approach would outperform existing methods significantly in terms of

Table 2. Results of benchmarks and LIAS on NLF and BNF datasets. **Bold** is used to indicate the best scores.

Methods	NLF					BNF				
	$mACS$	$mPPA$	AR_P	AR_R	AR_{F1}	$mACS$	$mPPA$	AR_P	AR_R	AR_{F1}
$LIAS_{ideal}$	0.982	0.958	0.986	1.000	0.993	0.932	0.799	0.937	1.000	0.966
$Newseye_{dbscan}$	0.375	0.066	—	—	0.754	0.447	0.105	—	—	0.697
$Newseye_{greedy}$	0.327	0.054	—	—	0.757	0.356	0.094	—	—	0.652
$Newseye_{hierarchical}$	0.382	0.061	—	—	0.757	0.538	0.139	—	—	0.690
$STRAS_{pre}$	0.790	0.606	—	—	—	0.800	0.634	—	—	—
$STRAS_{sg}$	0.861	0.785	—	—	—	0.834	**0.700**	—	—	—
$STRAS_{cbow}$	0.855	**0.792**	—	—	—	0.806	0.631	—	—	—
$STRAS_{ft}$	0.827	0.700	—	—	—	0.798	0.612	—	—	—
$LIAS$	**0.907**	0.719	**0.961**	**0.955**	**0.957**	**0.870**	0.588	**0.897**	**0.943**	**0.919**
$LIAS$+$bbox$	0.886	0.688	0.949	0.958	0.952	0.804	0.470	0.888	0.916	0.901

article separation. This not only demonstrates the feasibility of our proposed approach but also highlights its substantial potential. Furthermore, the outcomes of $LIAS_{ideal}$ validate the effectiveness of the rule sets applied in the first two steps of our method. Only accurate paragraph linking can generate ideal results, showcasing the robustness of our rule formulation. Across various metrics, $LIAS$ achieved the highest score in $mACS$.

Although our method generally outperforms existing approaches, it slightly lags behind $STRAS_x$ in the $mPPA$ metric. This discrepancy can be attributed to the presence of numerous articles with only one paragraph or very few characters in the datasets. The STRAS method, before combining paragraphs, isolates these short articles and other paragraphs by setting independent paragraph rules, thereby improving their recognition accuracy in the final predictions. In contrast, LIAS links every paragraph within each sub-region, leading to occasional misidentification of short articles as part of larger articles due to classification errors in links. However, since short articles have minimal impact on $mACS$, LIAS still achieves the best results in this metric.

While LIAS analyzes document layouts, the STRAS method directly uses the semantic embedding of the text to complete article separation based on similarity, which can result in the following errors: (1) misclassifying consecutive paragraphs with low semantic similarity as separate articles; or (2) grouping paragraphs from different articles due to high semantic similarity. Additionally, STRAS's reliance on a fixed semantic similarity threshold limits its adaptability to various document formats and introduces inconsistencies in results. In contrast, LIAS groups paragraphs first, eliminating the need for a similarity threshold and making it more robust to document variations.

Compared to Newseye, LIAS exhibits superior performance across all metrics. Newseye utilizes Delaunay triangulation to represent the positional relationship between text blocks, generating connected paths but failing to capture the actual logical connections between them. This approach also undermines the

semantic relationships between text blocks. In contrast, LIAS establishes a logical information flow between text blocks, facilitating a more accurate analysis of their semantic relationships. Additionally, LIAS addresses a limitation shared by Newseye and STRAS: the inability to isolate paragraphs without semantic connections. Since all paragraphs are interconnected in Newseye's models, when GNNs communicate between nodes, paragraphs with high semantic similarity from different articles can influence the final node embeddings. This degrades the quality of node semantics and affects the subsequent similarity-based clustering. LIAS on the other hand, overcomes this limitation by grouping paragraphs first, eliminating the need for a semantic similarity threshold and making it more robust to document variations.

Unlike [18], incorporating positional embeddings into LIAS does not yield an improvement in semantic distinction. Several reasons may explain this observation. Firstly, newspaper layouts exhibit more variability compared to structured documents such as invoices, leading to inconsistent semantic relationships among textual information at the same position. Secondly, LIAS links paragraphs and classifies these links within the same region. During the training process, the data fed into the embedding layer is close in location, making it challenging for the classifier to effectively learn valuable positional information. Thirdly, regarding the link between paragraphs across regions, if two paragraphs belong to the same article but are positioned far apart, this disparity in their positions could disrupt the model's learning process. This also explains why introducing positional embedding can be beneficial in token-level tasks; in documents, neighboring tokens often belong to the same phrase or sentence, exhibiting a high degree of semantic similarity. Similar position embedding resulting from similar positions provides semantic improvement. However, when the position embedding is used to describe the positional semantics of a text block, the situation differs from the token level.

Comparing the performance on both datasets reveals that all methods perform better on the NLF dataset. This is attributed to the higher quality preservation of historical newspapers on the NLF dataset, resulting in fewer OCR annotation errors. Conversely, the BNF dataset suffers from numerous annotation errors due to complex layouts and blurry images, negatively impacting all methods. Specifically, erroneous text and positional annotations affect the training of the link classifier step of LIAS. Despite the challenges posed by varying layouts, LIAS' region-based paragraph linking method demonstrates promising potential, as indicated by the $LIAS_{ideal}$ results on both datasets. Bridging the gap between these two datasets could be achieved by improving annotation quality or employing novel classifier training methods.

In addition to dataset quality, the choice of the backbone language model in the classifier is also responsible for the disparate results on both datasets. In the classifier, hmBERT is selected to obtain the semantic embedding of the tokens, and the quality of these semantic embeddings influences the final article separation performance. According to the results of hmBERT's NER experiments, it achieved an F1-score of 0.801 in Finnish but only 0.751 in French. This indi-

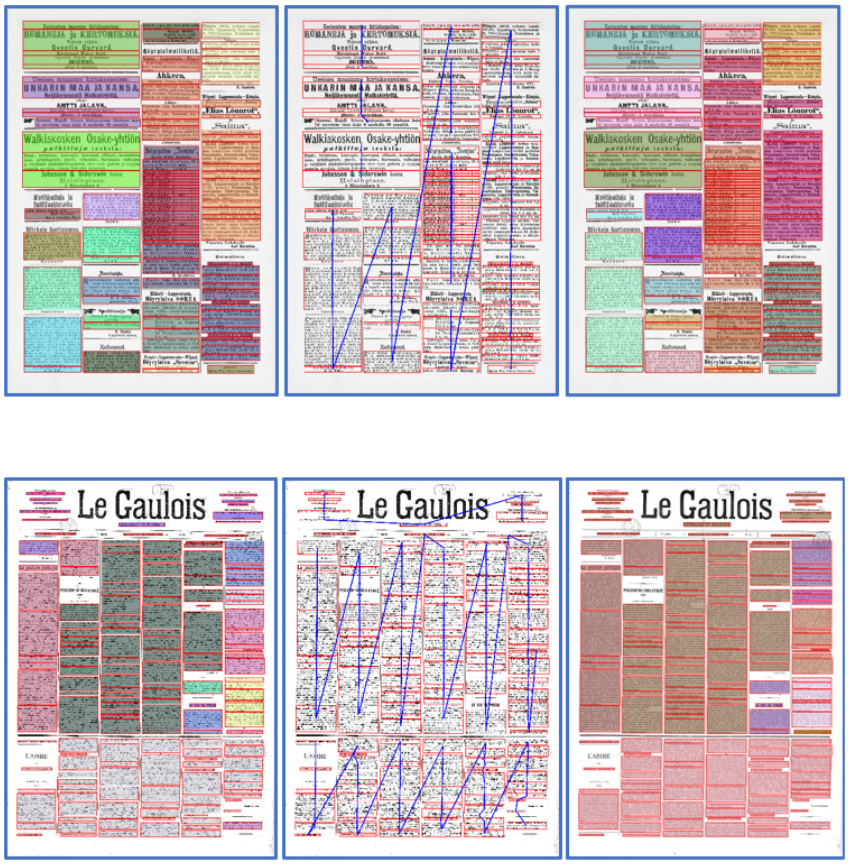

Fig. 6. Example of article separation result using LIAS. The left column corresponds the ground truth, the middle column is the output of the paragraph linking step, and the right column is the final prediction. Paragraphs in the same color mean they belong to the same article.

cates that hmBERT is less capable of processing the semantics of French tokens compared to Finnish. The varying quality of the semantic vectors of the encoder input to the classifier makes a difference in the subsequent processing, resulting in better performance on the Finnish dataset.

According to the performance analysis and method comparisons, improving the performance of the article separation model requires two key points. Firstly, we need to provide a better description of the logical relationship between the paragraphs in the whole document. Secondly, we need to improve the semantic extraction method for the paragraphs, especially embedding the semantic information of the layout of the paragraphs into the original text semantics.

5.1 Ablation Study

To investigate the impact of the encoder module on the classifier's performance, we conducted an ablation study. We removed the encoder module from the classifier and replaced it with direct average pooling on the output of the token embedding generated by the backbone language model. This modification aimed to assess the effectiveness of the encoder in capturing semantic information from paragraphs. Results after removing the encoder are depicted in Table 3. They reveal a significant decline in the classifier's performance when using only average pooling. This observation suggests that the encoder plays a crucial role in improving the model's ability to understand the semantic context of paragraphs. In our experiments, we utilized a language model trained on historical corpora. However, when dealing with noisy or potentially incorrect input content, additional processing of the output token embeddings is still necessary to obtain reliable results.

Table 3. Ablation results on both NLF and BNF datasets.

	NLF					BNF				
	$mACS$	$mPPA$	AR_P	AR_R	AR_{F1}	$mACS$	$mPPA$	AR_P	AR_R	AR_{F1}
$LIAS_{noEncoder}$	0.891	0.687	0.952	0.956	0.953	0.839	0.438	0.845	0.942	0.889

6 Conclusion and Limitations

To address the challenges of historical newspaper article separation, we introduced LIAS, a novel layout-based approach that simplifies complex page layouts and correlates them with semantic relationships. LIAS initiates the process by identifying separator lines from scanned images of newspaper pages, segmenting the newspaper into multiple sub-regions. Within each sub-region, paragraphs are linked in sequence based on their layout position (e.g., left-to-right, top-to-bottom). Horizontal and vertical separator lines are then utilized to link paragraphs across sub-regions, addressing cross-column article spanning issues. Finally, a classifier is trained to determine whether to remove a link based on the semantic information of the linked paragraphs. The actual article separation is performed using these links identified by the classifier. Our method showcases promising results on historical newspaper datasets, achieving competitive performance across various metrics.

While LIAS has proven to be competitive, it still encounters limitations in accurately identifying standalone paragraphs or articles with very few characters. Moreover, the quality of annotations significantly impacts our method's performance. Additionally, our current approach relies on manual annotations. In future work, we aspire to establish a fully automated pipeline for article separation that addresses these limitations. At the same time, this method is limited

by the effectiveness of text block regions and character recognition, and recognition errors can have a negative impact on the next step of semantic extraction.

Acknowledgments. This work has been supported by the ANNA (2019-1R40226), TERMITRAD (2020-2019-8510010), Pypa (AAPR2021-2021-12263410), and Actuadata (AAPR2022-2021-17014610) projects funded by the Nouvelle-Aquitaine Region (France).

Disclosure of Interests. The authors have no competing interests to declare relevant to this article's content.

References

1. Biswas, S., Riba, P., Lladós, J., Pal, U.: Beyond document object detection: instance-level segmentation of complex layouts. International Journal on Document Analysis and Recognition (IJDAR) **24**(3), 269–281 (2021)
2. Boroş, E., Hamdi, A., Pontes, E.L., Cabrera-Diego, L.A., Moreno, J.G., Sidere, N., Doucet, A.: Alleviating digitization errors in named entity recognition for historical documents. In: Proceedings of the 24th conference on computational natural language learning. pp. 431–441 (2020)
3. Doucet, A., Gasteiner, M., Granroth-Wilding, M., Kaiser, M., Kaukonen, M., Labahn, R., Moreux, J.P., Muehlberger, G., Pfanzelter, E., Therenty, M.E., et al.: Newseye: A digital investigator for historical newspapers. In: 15th Annual International Conference of the Alliance of Digital Humanities Organizations, DH 2020 (2020)
4. Duda, R.O., Hart, P.E.: Use of the hough transformation to detect lines and curves in pictures. Communications of the ACM **15**(1), 11–15 (1972)
5. Ehrmann, M., Hamdi, A., Pontes, E.L., Romanello, M., Doucet, A.: Named entity recognition and classification in historical documents: A survey. ACM Computing Surveys **56**(2), 1–47 (2023)
6. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: Overview of clef hipe 2020: Named entity recognition and linking on historical newspapers. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings 11. pp. 288–310. Springer (2020)
7. Girdhar, N., Coustaty, M., Doucet, A.: Benchmarking nas for article separation in historical newspapers. In: International Conference on Asian Digital Libraries. pp. 76–88. Springer (2023)
8. Girdhar, N., Coustaty, M., Doucet, A.: Stras: A semantic textual-cues leveraged rule-based approach for article separation in historical newspapers. In: International Conference on Asian Digital Libraries. pp. 89–105. Springer (2023)
9. González-Gallardo, C.E., Boros, E., Girdhar, N., Hamdi, A., Moreno, J.G., Doucet, A.: Yes but.. can chatgpt identify entities in historical documents? In: The ACM/IEEE-CS Joint Conference on Digital Libraries. pp. 184–189 (2023)
10. Grüning, T., Labahn, R., Diem, M., Kleber, F., Fiel, S.: Read-bad: A new dataset and evaluation scheme for baseline detection in archival documents. In: 2018 13th IAPR International Workshop on Document Analysis Systems (DAS). pp. 351–356. IEEE (2018)

11. Gu, Z., Meng, C., Wang, K., Lan, J., Wang, W., Gu, M., Zhang, L.: Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4583–4592 (2022)
12. Hong, T., Kim, D., Ji, M., Hwang, W., Nam, D., Park, S.: Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 10767–10775 (2022)
13. Huang, Y., Lv, T., Cui, L., Lu, Y., Wei, F.: Layoutlmv3: Pre-training for document ai with unified text and image masking. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 4083–4091 (2022)
14. Lee, J., Han, J., Baek, S., Song, M.: Topic segmentation model focusing on local context. In: AAAI-23 Workshop Program (2023)
15. Schweter, S., März, L., Schmid, K., Çano, E.: hmbert: Historical multilingual language models for named entity recognition. In: Conference and Labs of the Evaluation Forum (CLEF 2020) (2022). (2022)
16. Wang, J., Jin, L., Ding, K.: Lilt: A simple yet effective language-independent layout transformer for structured document understanding. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 7747–7757 (2022)
17. Xu, Y., Xu, Y., Lv, T., Cui, L., Wei, F., Wang, G., Lu, Y., Florencio, D., Zhang, C., Che, W., et al.: Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 2579–2591 (2021)
18. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: Layoutlm: Pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1192–1200 (2020)