Not All Titles are Created Equal: Financial Document Structure Extraction Shared Task

Anubhav Gupta and **Hanna Abi Akl** and **Hugues de Mazancourt**
Yseop
{agupta, habi-akl, hdemazancourt}@yseop.com

Abstract

This paper presents a multi-modal approach to *FinTOC-2021 Shared Task*. With help of a fine-tuned `Faster-RCNN` our solution achieved a Precision score comparatively better than other participants.

1 Introduction

Heading or title is a phrase that either represents an oeuvre or demarcates a text into chapters, sections etc. It serves as a milestone that helps readers find their way through a long text. Its appearance is governed by style guides. For example, AER[1] mandates that the title of a section begins with roman numerals and that of a subsection with capital letters. APA[2] and MLA limit the number of levels of titles to 5. Their guide ensures that each level is visually distinct from another. Titles at level 4 or below are "run-on heads[3]" / "run-in[4]" / in-line headings i.e. they appear along with the text.

Unfortunately no such guide is available for the prospectuses provided as part of FinTOC 2021 Financial Document Structure Extraction Shared Task (El Maarouf et al., 2021). In other words the documents do not follow the same style guide, assuming they are respecting one. As a result, the task of identifying a heading and its correct level is daunting. This is proved by the fact that the best score for the previous year's shared task (Bentabet et al., 2020) was **0.37**.

[1] https://www.aeaweb.org/journals/aer/submissions/accepted-articles/styleguide#IVA

[2] https://apastyle.apa.org/style-grammar-guidelines/paper-format/headings

[3] https://projects.iq.harvard.edu/crea-lit/headings-and-subheadings

[4] http://www.creativeglossary.com/graphic-design/run-in-heading.html

2 Data

The training set consisted of 47 prospectuses in `PDF` format along with the annotations in `json` format. The annotations had depth (level), page number, file name and raw text of each title. The test set had 10 prospectuses for each language and the task is to generate a json file as described before for each of the files.

The first challenge was to find these texts in the PDFs in order to extract more metadata viz. position, font, size etc. This process is detailed in Section 3.

The second challenge was that the number of levels that the titles can have was not defined. Also, there were quite a number of documents having multiple depth one title i.e. main heading!

A cursory glance reveals that the title level 1 is always present in the first page and is either the name of the fund or the key phrase **Prospectus** for English and **Informations clés pour l'investisseur** for French. If both the fund's name and the key phrase exist it is generally the one that appears first irrespective of the style of the sentence. There were certain documents where this wasn't the case.

We argue that **Prospectus** or **Informations clés pour l'investisseur** should always be the first level title if present in the first page since it describes the document and is consistent with other main titles of documents such as **Status**, **Reglement**, **Key Information Document**, etc.

Language	P	R	F1
English	*0.858*	0.670	0.728
French	**0.911**	0.510	0.639

Table 1: Title detection results.

Language	Inex08					Harmonic
	P	R	F1	Title Acc	Level Acc	Mean
French	46.8	28.1	34.4	47.3	16.6	22.4
English	**61.1**	50.3	*53.4*	**68.2**	12.4	20.1

Table 2: TOC extraction results.

3 System

We used `pdfminer.six`[5] to parse the files. We extracted `LTTextLine` and matched it against the annotations. If

- it is an exact match, we extract features.
- `LTTextLine` is a subtext of an annotation, we find all such subtexts, then merge them and finally, extract features.
- a annotation is a subtext of `LTTextLine`, it is ignored.

In short we ignored quite a few inline titles. This might have led the models to treat them as normal text and may have been the cause of low Recall score.

From each `LTTextLine` we collected:

- coordinates (normalized: divided by page dimensions)
- percentage of characters in **bold**
- percentage of characters in *italics*
- mode of character sizes (min-max normalized)
- height (min-max normalized)
- page number (min-max normalized)
- inverse length
- percentage of capital letters
- does it start with numbering
- does it start with a capital
- is it all capital letters
- average character area
- MinHash[6] encoding of the first 10 characters
- normalized text (only alphabets without accents) in lowercase to compute `tf-idf`

The scikit-learn (Pedregosa et al., 2011) was used to get TFIDF with the following arguments:

```
analyzer = 'char'
ngram_range = (3, 3)
max_df = 0.93
max_features = 3000
sublinear_tf = True
```

As mentioned above, some `LTTextLines` were needed to be combined to match a title in the annotations. This was done as follows:

- find `LTTextLine` that matches the beginning of the annotation
- if this subtext along with previously matched `LTTextLines` has the least area then keep it
- update the annotation by removing the prefix that matched `LTTextLine`
- if annotation is an empty string then stop else repeat

Once we identified the titles along with features in the PDFs we converted the documents into images with the help of `pdf2image`[7]. The coordinates of the titles were multiplied by 4 to get the bounding boxes and then saved in COCO format[8]. This was used to fine-tune the PubLayNet (Zhong et al., 2019) Faster-RCNN model as explained on their github repository[9] with the hyperparameters of Table 3.

The fine-tuned model was used to obtain IoU and probability of being title for each `LTTextLine`.

[5] https://pdfminersix.readthedocs.io/en/latest/

[6] https://dirty-cat.github.io/stable/generated/dirty_cat.MinHashEncoder.html#dirty_cat.MinHashEncoder

[7] https://github.com/Belval/pdf2image

[8] https://cocodataset.org/#format-data

[9] https://github.com/ibm-aur-nlp/PubLayNet/tree/master/pre-trained-models

Hyperparameters	English	French
BASE_LR	0.005	0.001
MAX_ITER	36000	50000
STEPS	[0, 24000 , 32000]	[0, 30000 , 40000]

Table 3: Hyperparameters to finetune PubLayNet.

These two values along with the features listed above was fed to a Gradient Boosting Classifier with parameters:

```
n_estimators = 200
learning_rate = 0.2
max_leaf_nodes = 10
min_samples_leaf = 15
max_depth = 20
random_state = 10
```

We trained one model for each language.

At test time, the fine-tuned PubLayNet was used to merge `LTTextLines` and then extract features for classification.

After classification the titles were sorted by their size. The largest titles were attributed a depth of 1. The next in order were given 2 as depth and so forth.

4 Results

Our model, despite having the highest precision for French and second-highest precision for English, came second in the title detection task (see Table 1).

In case of TOC extraction, the English model had the highest Inex08 Precision and Inex08 Title Accuracy among the competing methods. We could not achieve the same performance for French due to less time allotted for fine-tuning the PubLayNet model. Since the models scored low on Inex08 Level Accuracy, we were nowhere near the top performing team that achieved the Harmonic Mean greater than **0.5**.

5 Discussion

We can further improve the Inex08 scores related to title detection for French by better fine-tuning the PubLayNet model.

We would also like to compare this model with LayoutLM[10,11](Xu et al., 2020), also based on `Faster-RCNN`.

[10]https://github.com/microsoft/unilm/tree/master/layoutlm

[11]https://huggingface.co/transformers/model_doc/layoutlm.html

However, a model that can correctly identify the title levels and be ported to other domains remains elusive.

6 Conclusion

We feel that lack of an annotation guide makes it difficult to analyse the errors related to title levels and as a result improve the results. The use of a vision-based model improves the title detection and can be generalized to other domains.

Acknowledgements

We would like to thank Fortia Financial Solutions for organizing this task and thus contributing to the advancement of document structure analysis.

References

Najah-Imane Bentabet, Remi Juge, Ismail El Maarouf, Virginie Mouilleron, Dialekti Valsamou-Stanislawski, and Mahmoud El-Haj. 2020. The Financial Document Structure Extraction Shared Task (FinToc 2020). In *The 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation (FNP-FNS 2020)*, Barcelona, Spain.

Ismail El Maarouf, Juyeon Kang, Abderrahim Aitazzi, Sandra Bellato, Mei Gan, and Mahmoud El-Haj. 2021. The Financial Document Structure Extraction Shared Task (FinToc 2021). In *The Third Financial Narrative Processing Workshop (FNP 2021)*, Lancaster, UK.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. 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*.

Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. Publaynet: largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022. IEEE.