Jura: Towards Automatic Compliance Assessment for Annual Reports of Listed Companies

Zhengqi Xu[1,2], Yixuan Cao[1,2], Rongyu Cao[1,2], Guoxiang Li[3], Xuanqiang Liu[3], Yan Pang[3], Yangbin Wang[3], Jianfei Zhang[3], Allie Cheung[4], Matthew Tam[4], Lukas Petrikas[4], Ping Luo[1,2]
[1]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China.
[2]University of Chinese Academy of Sciences, Beijing 100049, China.
[3] Beijing Paoding Technology Co., LTD.
[4] Hong Kong Exchanges and Clearing Limited
{luop}@ict.ac.cn

ABSTRACT

The initial public offering (IPO) market in Hong Kong is consistently one of the largest in the world. As part of its regulatory responsibilities, Hong Kong Exchanges and Clearing Limited (HKEX) reviews annual reports published by listed companies (issuers). The number of issuers has grown at a fast pace, reaching 2,538 as the end of 2020. This poses a challenge for manually reviewing these annual reports against the many diverse regulatory obligations (listing rules). We propose a system named `Jura` to improve the efficiency of annual report reviewing with the help of machine learning methods. This system checks the compliance of an issuer's published information against listing rules in four steps: panoptic document recognition, relevant passage location, fine-grained information extraction, and compliance assessment. This paper introduces in detail the passage location step, how it is critical for speeding up compliance assessment, and the various challenges faced. We argue that although a passage is a relatively independent unit, it needs to be combined with document structure and contextual information to accurately locate the relevant passages. With the help of `Jura`, HKEX reports saving 80% of the time on reviewing issuers' annual reports.

ACM Reference Format:
Zhengqi Xu[1,2], Yixuan Cao[1,2], Rongyu Cao[1,2], Guoxiang Li[3], Xuanqiang Liu[3], Yan Pang[3],, Yangbin Wang[3], Jianfei Zhang[3], Allie Cheung[4], Matthew Tam[4], Lukas Petrikas[4], Ping Luo[1,2] . 2021. Jura: Towards Automatic Compliance Assessment for Annual Reports of Listed Companies. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3459637.3481929

1 INTRODUCTION

In many public stock markets, listed companies must publish an annual report that presents their financial results, business performance and management commentary to the general public every year. In the Hong Kong stock market, Hong Kong Exchanges and Clearing Limited (HKEX), as the frontline regulator of listed companies (issuers), reviews annual reports to check, among other things, that the issuers are disclosing all the relevant information that the *listing rules* of the Stock Exchange require of them.

A listing rule specifies what information needs to be disclosed by issuers in an annual report. Example 1.1 shows a listing rule that requires the company to disclose certain information about its five highest-paid employees during the reporting year; more specifically, to disclose an analysis about the distribution of the five highest-paid individuals within different compensation bands. If such information is not disclosed in an annual report, or disclosed but not following the strict specification, the case is considered non-compliant.

Example 1.1. Listing Rule Appendix 16.25(6): Five highest paid individuals: an analysis showing the number of individuals whose remuneration (being amounts paid under (1) to (5) above) fell within bands from HK$nil up to HK$1,000,000 or into higher bands (where the higher limit of the band is an exact multiple of HK$500,000 and the range of the band is HK$499,999) must be disclosed.

Another type of listing rule that requires the identification of specific fine-grained information is shown in Example 1.2. In this example, issuers are required to disclose the price of each security of the company issued within the reporting year. The presence of this information (or lack thereof) in an annual report is insufficient to complete a full assessment, which means that retrieving and analyzing other supplementary documents (e.g. announcements made by the issuer during the year) is required. For example, a company may have announced two issuances during the year, but only the price of one security issuance is disclosed. Such a case is considered non-compliant.

Example 1.2. Listing Rule Appendix 16.11(4): Issue for cash of equity securities: the issue price of each security.

The listing rules in the scope of `Jura` are very diverse, each posing very different requirements for document structure recognition, disclosure location and compliance assessment recommendation. Within the scope of this study, nearly one hundred listing rules were considered.

HKEX has been assessing annual reports against a subset of the listing rules each year. On average, an annual report has 173 pages. The information relevant to each Listing Rule is often scattered across different pages of an annual report. It takes HKEX staff, on average, 110 minutes to review the report for listing rule

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3481929

compliance. With a rising number of listed companies, reaching 2,538 at the end of 2020, manually assessing thousands of reports against even a sub-sample of listing rules is laborious and time-consuming. Nonetheless, this work is important to improve the quality of issuers' published materials and transparency for the investing public. Thus, HKEX has long been interested in applying intelligent automation to its annual report review process.

This kind of system can be categorized as regulatory technology (RegTech), which means technology used to oversee, monitor or conform to regulatory requirements. RegTech is a rapidly growing and diverse field. But no full-fledged system can support compliance assessment of a hundred of diverse listing rules. We introduced some solutions related to this problem. The Cognitive Compliance function [26] assesses risks in financial advice documents (on average 60 pages). It considers the appropriateness of an advisor's recommendations to its client's goals, and whether the document contains asset class tables, the client's capital position, cash flow analysis, and insurance consideration. While an existing Numerical Cross-Checking system introduced by [4] and [12] may be able to help check the consistency of values of financial indexes within a financial document (e.g. some financial institutions have successfully built models to detect simple pieces of information like earnings per share and interest rate)

To bridge this gap, we built `Jura` to make reviewing annual reports of listed companies more efficient. Given an annual report, `Jura` will help reviewers assess whether the issuers are disclosing all the relevant information that the listing rules require of them, by locating the relevant passages within the lengthy document and recommending a compliance assessment. `Jura` saved 80% time compared with human review alone. This system has become a key part of HKEX's regulatory toolkit. `Jura` was recognized by Regulation Asia, a publication focused on financial regulation, with its Award for Excellence 2020 for Outstanding Project – Financial Document Analysis (selected out of 194 submissions).

The challenges of building a compliance assessment system for annual reports can be summarized into the following two aspects.

• *Diversity of listing rules.* For the purposes of this study, HKEX selected nearly *one hundred* listing rules that regulate various aspects of information that issuers must disclose in their annual reports. Each listing rule is unique regarding what information should be disclosed and how it should be disclosed (some examples are discussed in Section 2.3). Given the diversity of the selected listing rules and the human interpretation associated with them, designing separate programs for each rule is time-consuming and difficult to manage.

• *Complexity of annual reports.* Annual reports are unstructured, richly formatted and lengthy documents that are often submitted in PDF format. No two annual reports are the same: each issuer can decide how to organize and present its annual report. With 154 new company listings in Hong Kong in 2020 alone, the variety of documents is ever-expanding. An annual report document has a complex structure with multiple levels of hierarchy, and is a mixture of paragraphs, tables and charts which makes it difficult to parse. Moreover, to understand the meaning of a *passage* (paragraph or table), one has to take the document structure information into consideration. For example, in Figure 1, the second table is what the listing rule requires the issuer to disclose. However, if we only look at the table alone, we cannot necessarily infer that it is related to the five highest-paid employees. Important clues are scattered in its section headings and previous passages.

We propose corresponding approaches to address the above challenges. To address the diversity of the selected listing rules, we divide the whole assessment process into four stages: *panoptic document recognition*; *relevant passage location*; *fine-grained information extraction*; and *compliance assessment*. The first stage parses the PDF document and returns *objects* (e.g. paragraphs, charts, tables, pictures) and *document structure* (e.g. table of contents, section headings) of the document. The details are described in Section 2.1. For each listing rule, the second stage locates relevant *passages* (paragraphs or tables). This stage is considered a passage classification problem which is generic for all listing rules. The details are illustrated in Section 3. In the third stage, fine-grained information (e.g. the issue price of each security within relevant passages) is extracted (Section 2.2), and finally the fourth stage assesses the compliance based on the result of the preceding stages (Section 2.3). The third and fourth stages are more dependent on each listing rule and require customized solutions. We divide the selected listing rules into four distinct categories of "difficulty" so that rules in the same category can be assessed similarly. The staging and the categorization strategy minimizes the amount of customization required for each different listing rule by abstracting the commonalities among listing rules.

To summarize, the contributions of this work are as follows:

- We introduce `Jura`, the first full-fledged system to accelerate the compliance assessment process for complex annual reports of listed companies against a large number of listing rules. It can reduce 80% of the time for regulatory assessment of an annual report on average but keep the same accuracy compared to manual method, while greatly expanding the achievable breadth of assessment coverage.
- We propose staging and categorization strategies to address the diversity and complexity of listing rules, representing real-life regulatory obligations faced by listed companies.
- We propose machine learning models that integrate the document hierarchy in the relevant passage location stage to address the diversity and complexity of annual reports.

In the rest of the paper, we first introduce the overview of `Jura`. Then, we focus on the relevant passage location stage, introduce the problem formulation, solution, and experiments.

2 SYSTEM

`Jura` is currently designed as an assistance system for human-led compliance assessment. Though capable of automatic and autonomous compliance assessment, it currently provides suggestions for human review. When assessing an issuer's compliance against a listing rule, as shown in Figure 1, `Jura` will recommend passages that are relevant to that listing rule. The top 5 suggestions are shown in the Relevant Passage Suggestions red box, and the relevant passage (the top suggestion) is shown in the blue box in Document Display Panel. It also provides prediction results on the issuer's compliance for the reviewer's consideration, as shown in the Compliance Assessment red box. As the right side of Figure 1 shows, `Jura` consists of four stages. These four stages are as follows:

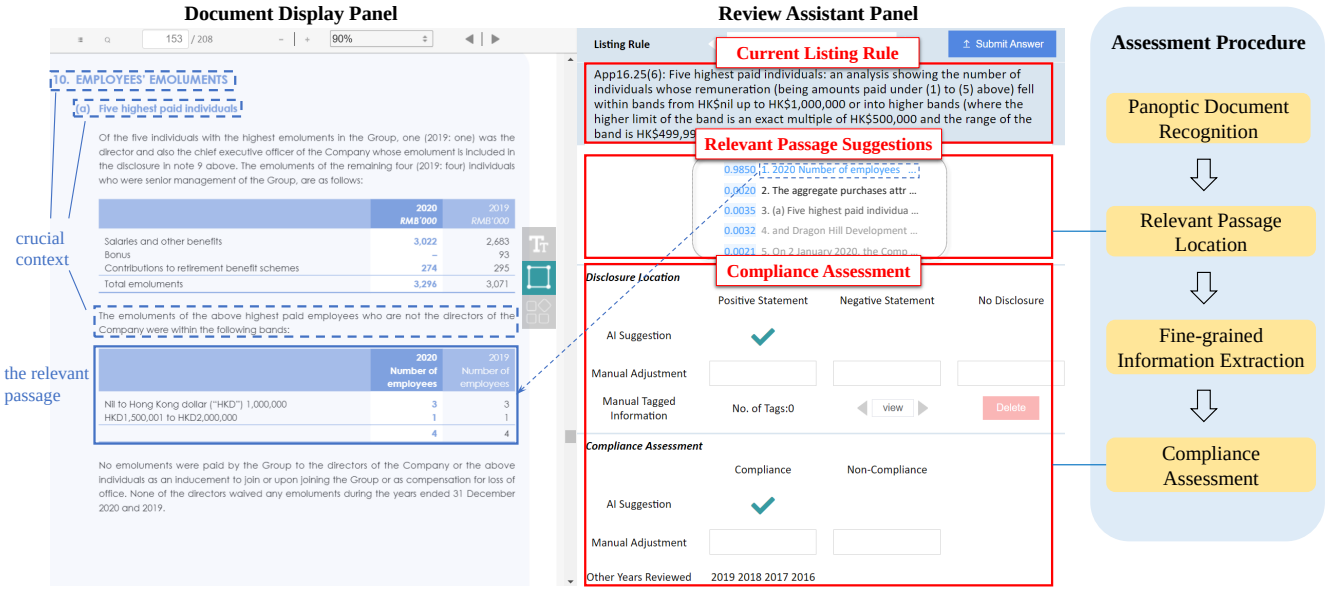


Figure 1: The interface and process of Jura.

• *Panoptic document recognition.* Most annual reports are in PDF format. But PDF only records the location and style of characters and lines instead of structural information like paragraphs, tables, and chapters. So, it is only "human readable". We need to restore the physical and logical structure of the document so that it becomes "machine readable".

• *Relevant passage location.* Utilizing the recognized document structure, we can locate relevant passages within annual reports and other relevant corporate communications with respect to different listing rules. These passages include all information needed to assess listing rule compliance. This is a recall stage to narrow the scope for subsequent stages.

• *Fine-grained information extraction.* For many listing rules, the simple location of passages alone is not sufficient to determine issuers' compliance. So we need to further extract fine-grained information, like Example 1.1.

• *Compliance assessment.* Using relevant passages and fine-grained information, we can arrive at a recommended assessment of whether issuers are compliant with respect to each relevant listing rule.

Among the four stages, the second stage demonstrates the challenge of this system and we formulate it as a generic passage classification problem. We discuss this stage in detail in Section 3. The other three stages are briefly introduced in this section.

It should be noted that the purpose of Jura is never to replace human regulator, but to assist human regulator to assess compliance with ease. For each listing rule, human regulator can first check the accuracy of suggestions and prediction results instead of finding relevant passages in the whole document all the time.

2.1 Panoptic Document Recognition

Before we locate the relevant paragraph(s) or table(s) and extract the fine-grained information, we have to know what constitutes paragraphs and tables in a document, as well as their relations and hierarchy. This is not a trivial task when we are dealing with PDF files. We briefly introduce what kind of information has to be recognized for this task, and share our experience on building systems with PDF inputs.

The Portable Document Format (PDF) is a file format developed by Adobe in 1993 to present documents, which has become the de facto standard for fixed-format electronic documents to date [8, 27]. To faithfully reproduce content across devices, PDF is primarily designed for the preservation of layout instead of further editing [2]. A PDF document stores a collection of programmatic instructions to draw characters and line segments on a page with visual formatting information. However, this intelligent design brings a major drawback in that the underlying physical and logical structures of the document are lost, which poses a challenge in analyzing and reusing the content of PDF files. Here, the physical structure refers to identifiable physical objects in the document (e.g. tables, graphs, paragraphs, page headers and footers, etc.) and a logical structure that reflects the parallel and containment relationships between these objects.

We name the process of extracting physical objects and integrate them into a logical document hierarchy as *panoptic document recognition*. Panoptic document recognition is a complicated task that consists of four sequential sub-tasks: physical object detection, cross-page object trimming, reading order determination and logical hierarchy discovery. Next, we introduce each sub-task with the example in Figure 2.

• *Physical object detection.* For a given document page, detect each physical object (table, graph, paragraph, page header and footer, etc [19]) on the page and delineate it with a bounding box. This is depicted in the left part of Figure 2 where we use two adjacent pages in an annual report as an example. Many existing studies focus on

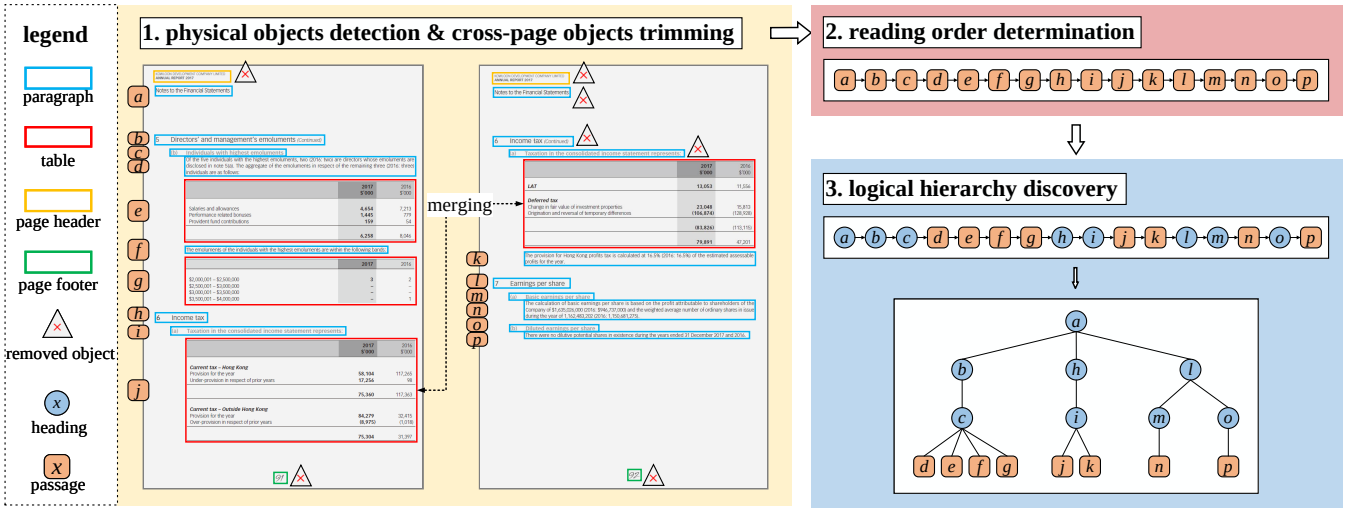


Figure 2: The framework of panoptic document recognition.

physical object detection tasks and use semantic segmentation or object detection techniques to tackle this problem [15, 31].

• *Cross-page object trimming.* In this step, we trim the detected physical objects into a collection at the document level and thus eliminate the concept of document pages. First, repeated physical objects like page header and footer should be removed at the document level, since they are redundant information for machine understanding. Second, cross-page objects are merged. For example, on the two pages in Figure 2, the table about taxation in income statement is split into two parts, which should be merged into one object.

• *Reading order determination.* A human reads a PDF document sequentially, passage-by-passage, according to a certain reading order. Thus, physical objects should be converted into an ordered sequence to obtain the progressive relationship. Typically, the conventional top-to-bottom left-to-right reading order is used in a large number of documents. As shown in the top-right part of Figure 2, these physical objects can be converted into a physical object sequence in an order from a to p. Reading order can provide position information which is helpful in the next section of relevant passage location. For more complicated document layouts, there are rule-based or learning-based methods [17, 20].

• *Logical hierarchy discovery.* This step transforms the flat structure of detected physical objects into a hierarchical structure, which reflects the parallel and containment relationships among physical objects and is essential for the next section of relevant passage location. We first recognize heading objects and then generate the logical hierarchy. For example, in the bottom-right part of Figure 2, physical objects a, b, c, h, i, l, m, o are recognized as headings; and then the logical hierarchy, with the headings as internal nodes and other concrete objects as leaf nodes, is generated. Previous studies focused on logical hierarchy discovery mainly employ three types of method: rule based, sequence-labelling based and tree-generation based [1, 3, 5, 18, 22, 23].

2.2 Fine-grained Information Extraction

For a small number of listing rules, the simple location of a passage is sufficient for determining an issuer's compliance. For these cases, compliance is automatically implied if a certain disclosure exists, and vice versa. However+, with regard to most listing rules, specific texts or values have to be extracted for consistency to be established against other parts of the annual report or other relevant corporate communications previously published by that issuer. For example, Example 1.1 requires the extraction of boundary values of bands after locating the relevant table, and Example 1.2 requires the extraction of the issue price within the paragraph. For these cases, fine-grained information has to be extracted and logical relationships established among different parts of the document(s), just as how humans would comprehend information.

To extract fine-grained information, we use reading comprehension techniques to extract the precise span from the whole passage after locating certain passages. This is a generic method that can handle various listing rules. However, there are many cases where rule-based methods work better. For example, one would prefer a rule-based program to check the range of bonds over a machine learning model for Example 1.1. This stage requires human judgment, to help determine whether rule-based methods are more suitable and effective for certain listing rules over others.

2.3 Compliance Assessment

If relevant information is collected, such as the issue prices of securities in Example 1.2, or a table about the five highest paid individuals in Example 1.1, how do we assess its compliance? As previously mentioned, different listing rules vary a lot, and each requires different ways to assess compliance. Writing different programs for each listing rule is straightforward but laborious. To simplify the problem of compliance assessment and to make it replicable with regard to future use cases, we categorize the listing rules so that rules in the same class share similar assessment methodology. From the aspect of external documents dependency,

		Other corporate communications	
		PS	NS/ND
Annual report	PS	C	NC
	NS/ND	NC	C

PS: positive statement
NS: negative statement
ND: non-disclosure
C: compliant
NC: non-compliant

Figure 3: Existence check of consistency.

there are "dependent" and "independent" rules. For "independent" rules, the presence of certain disclosures alone indicates listing rule compliance and vice versa. For "dependent" rules, disclosures from multiple sources (e.g. additional issuer announcements) are needed to verify consistency. From another aspect, there are "existence check" and "value-based check". Combining these two aspects, we get four categories. We introduce them using examples as follows.

• *Independent existence check.* Take Example 2.1 as an example, the issuer is compliant if information about its distribution to shareholders can be found in the annual report, and non-compliant if such information cannot be found.

Example 2.1. Listing Rule Appendix 16.29: A listed issuer shall include a statement of the reserves available for distribution to shareholders.

• *Independent value-based check.* Take Example 1.1 as an example, it contains certain requirements to display the distribution of the issuer's remuneration to employees for the year. We need to extract numerical values from the table and check whether these values meet the requirements.

• *Dependent existence check.* Take Example 1.2 as an example, it requires reviewers to consider both the annual report and other corporate communications that announce related events. The logic of verifying consistency is described in Figure 3. First, we locate relevant passages. If we cannot find any relevant passage, then it is non-disclosure (ND). If a relevant passage is present, it can be a positive statement (PS) or a negative statement (NS). We use BERT [6] to classify the relevant passages into PS and NS from the annual report and other corporate communications. Then, combining the classification results of the annual report and other corporate communications, we can assess whether the issuer is compliant (C) or non-compliant (NC) with regard to Figure 3.

• *Dependent value-based check.* Take Example 2.2 as an example, it involves numeric calculation and logic using values from different documents. For values in annual reports, such as number of share options granted, they are checked against various external documents (in this case, Monthly Returns of Equity Issuer on Movements in Securities) during the financial year. Then, they are checked against the following formula:

options outstanding at the end of the period
= options outstanding at the beginning + options granted
- options exercised - options cancelled - options lapsed

where the six values refer to the number of options. If the equation cannot be established, then the outcome is an assessment of potential non-compliance with regard to this listing rule.

Example 2.2. Listing Rule Chapter 17.07(2): Particulars of options granted during the financial year/period, including number of options, date of grant, vesting period, exercise period, exercise price and (for options over listed securities) the closing price of the securities immediately before the date on which the options were granted.

In summary, assessing compliance can become quite complicated as listing rules become more complex. Classifying these listing rules into above-mentioned categories can simplify the task of assessing compliance and make the logic clear.

In summary, for a given original PDF document that only contains independent characters, line segments and figures on separate pages, the physical and logical structure of the whole document can be generated via panoptic document recognition. The results will be applied to subsequent stages including relevant passage location and the aforementioned fine-grained information extraction.

3 RELEVANT PASSAGE LOCATION

Annual reports, as well as other financial documents, often consist of hundreds of pages, and we have nearly one hundred listing rules to check in this study. It is laborious to manually locate all the passages in an annual report for every listing rule. However, designing different programs for every listing rule can also be labor intensive. To find relevant passages automatically, there are mainly two ways: passage retrieval and passage classification. Passage retrieval requires well-designed queries for each listing rule and it is hard to generalize to new documents and listing rules. Although retrieval is much faster than classification, the system is not real-time and speed is not the main concern. Hence, we build a classification model for relevant passage location.

3.1 Problem Definition

A passage could be a paragraph or a table. As a passage may be relevant to multiple listing rules, multi-label classification should be applied rather than multi-class classification. Suppose there are m listing rules in total and n passages in a document. We denote the i-th passage in the document as X_i, a binary label Y_{ij} indicates whether the i-th passage is relevant to the j-th listing rule. All labels of a document form a matrix $Y \in \{0,1\}^{n \times m}$. A passage could be relevant to 0, 1 or multiple listing rules. The problem is defined as, given a document $(X_1, \cdots, X_n)$, predict the relevant matrix Y. We break this problem down into subproblems: given a passage X_i, predict whether it is relevant to the j-th listing rule (Y_{ij}).

Although predicting Y_{ij} does not depend on the whole document, it does depend on information outside the passage itself. Take Figure 1 as an example, the relevant table itself contains little information about its semantic meaning. Important pieces of information are scattered over its context, including its preceding passage which acts as the table title and its headings. For the same example, figure 2 shows the hierarchical relationship among these passages. Node g is the relevant passage. f is its previous node and c is its parent node (heading). This indicates that panoptic document recognition is essential for this problem. Without the logical hierarchy of the passages, we would have to consider the whole document to predict Y_{ij}. However, with the hierarchy identified, the scope is narrowed down to siblings and ancestors of i.

Our study focuses on how to utilize the information in documents. Logistic regression (LR) and BERT are the representative models of the traditional bag-of-words methods and state-of-the-art deep learning methods. We choose them as our model respectively because of their effectiveness and high performance, as well as their widespread use in the industry.

3.2 Model

In this section, we introduce how we adopt LR and BERT as our model for passage relevance classification.

For LR, we use n-gram method to extract features from the text, because many listing rules are highly dependent on certain keywords and/or key-phrases. We apply TF-IDF [24] to measure the importance of each term. Different from common usage, the minimum unit of the task is a passage rather than a document. We regard each passage as a "document" to calculate term frequency and inverse document frequency. The feature x_i of the information within passage i is a TF-IDF vector on a predefined vocabulary calculated on the passage. As we mentioned, ancestor and preceding passages are important, so we concatenate their features to get the final feature of a passage:

$$X_i = \left[x_{anc(i)}, x_i, x_{pre(i)}\right].$$

For example, the final feature of node g is $X_g = \left[x_{a,b,c}, x_g, x_f\right]$ where $x_{a,b,c}$ is the TF-IDF vector of merged texts of nodes a, b and c. After getting all features, we train m independent binary classification LR models for m listing rules. Such an approach is suitable for listing rule modification because modifying a listing rule only requires us to re-train one of all models.

For BERT, the feature s_i of the information within passage i is the text embedded within. When the passage is a table, we flatten it into long plain texts from left to right and top to bottom. Similar to LR, the input of BERT consists of three parts: the passage itself, ancestor passages, and the preceding passage:

$$X_i = \left[s_{anc(i)}, s_i, s_{pre(i)}\right]$$

For example, the input of node g is $X_g = \left[s_{a,b,c}, s_g, s_f\right]$ where the three parts are concatenated with symbol [SEP] and $s_{a,b,c}$ indicates the token sequence of node i are concatenated with the other symbol "|". The number of "|" indicates the number of ancestors and BERT might capture the difference between different ancestors. Specifically, when the previous node of the passage is the same as the parent node of the passage, such a special case reveals positional information that the passage is the first child of its parent. It might be helpful for classification because many relevant passages have such positional properties. Due to the restriction of BERT maximum token length, we put the ancestor nodes at the beginning because they are often short and do not take up much space. Then we put the previous node behind the current node because we assume the current passage is more important than the previous node. If the length of the concatenated sequence is larger than the preset maximum length, the sequence is truncated.

Unlike LR, training m independent binary classification BERT costs too much time and space. Therefore, we jointly train one multi-label classification model. Specifically, we add a full connection layer with m units on the hidden state of the first token [CLS]. The joint loss function for multi-label classification formulates as follows:

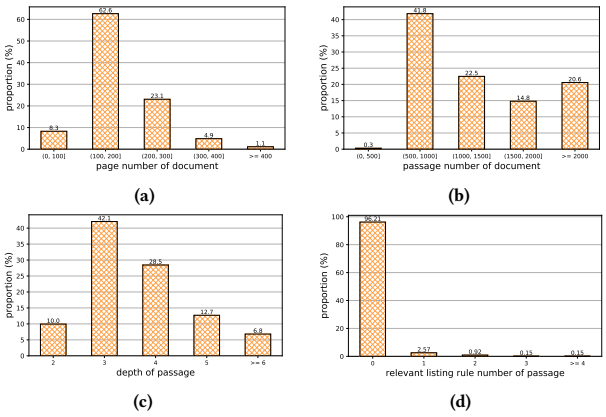


Figure 4: The distributions of documents and passages.

$$L = -\frac{1}{mn}\sum_{i}^{m}\sum_{j}^{n}\left[Y_{ij}\log\left(P_{ij}\right) + \left(1 - Y_{ij}\right)\log\left(1 - P_{ij}\right)\right] \quad (1)$$

$$P_{ij} = \text{sigmoid}\left(BERT_j\left(x_i\right)\right) \quad (2)$$

where x_i is the input feature of passage p_i, $BERT_j$ means the j-th dimension of BERT output. Then we use sigmoid function to replace softmax function for multi-label classification. P_{ij} represents output probability whether the i-th passage is relevant to the j-th listing rule. Jointly modeling can accelerate training and inference, and can also share weights for different listing rules. But it might take longer to re-train the whole model to support listing rule modification(s) and addition(s).

4 EXPERIMENTS

4.1 Data set

There are 96 different listing rules selected for the study. We collect 1,318 annual reports from HKEX, with 1.84M passages in total. We task a tagging team with a background in finance to find the correct passages relevant to each of these listing rules. Each document is labelled by two or more members of the tagging team, and any labelling conflict is resolved by a third person. We repeatedly check the wrong classification cases and re-label the documents to obtain a high-quality data set.

Figures 4a and 4b show the distribution of documents in the data set. Most documents contain more than 100 pages and 500 passages. On average, a document has 173 pages and 1,398 passages. The length of the documents and the large number of listing rules increase labelling difficulty, and some listing rules may have multiple relevant passages. Thus, we cannot rule out the possibility of unlabeled but relevant passages. The model might be misled by such unlabeled but relevant passages. To mitigate this, we find that negative sampling can accelerate training without the drop in model performance, as discussed in Section 4.3.

Figures 4c and 4d show the distributions of passages. Figure 4c shows the distribution of passage depth in the logical hierarchy. The depth of most passages ≥ 3, and the depth of nearly one fifth of all passages ≥ 5, which indicates the complex logical hierarchies of annual reports. As Figure 4d shows, 96 percent of all passages are

irrelevant to any listing rule in the scope of Jura. We define passages irrelevant to any listing rule as *negative* samples and others as *positive* samples. Hence, it is an unbalanced classification problem. Some passages are relevant to more than one listing rule. This is the reason why we have to undertake multi-label classification.

4.2 Metrics

To accurately evaluate model performance, training samples should be separate from test samples. Hence, the training set and test set are divided by document. In our experiments, the splitting ratio is 9:1. Due to the fact that the number of positive samples for different listing rules varies a lot, both micro and macro metrics are used to measure model performance from different aspects. The micro metric calculates the overall metric regardless of different listing rules. The macro precision and recall is the average of the metrics of different listing rules. The macro F1 is the arithmetic mean of class-wise F-scores [21].

$$
\begin{aligned}
P_{\text{micro}} &= \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m} (TP_i + FP_i)} \qquad P_{\text{macro}} = \frac{1}{m}\sum_{j=1}^{m} \frac{TP_j}{TP_j + FP_j} \\
R_{\text{micro}} &= \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m} (TP_i + FN_i)} \qquad R_{\text{macro}} = \frac{1}{m}\sum_{j=1}^{m} \frac{TP_j}{TP_j + FN_j} \\
F_{\text{micro}} &= \frac{2P_{\text{micro}}R_{\text{micro}}}{P_{\text{micro}} + R_{\text{micro}}} \qquad F_{\text{macro}} = \frac{1}{m}\sum_{j=1}^{m} \frac{TP_j}{TP_j + \frac{1}{2}\left(FP_j + FN_j\right)}
\end{aligned} \tag{3}
$$

where TP_j, FP_j, FN_j are the numbers of true positives, false positives and false negatives of the j-th listing rule.

Apart from precision, recall and F1, we also evaluate model performance on recall@k. Recall@k is different from the recall metric mentioned before. On the one hand, we do not set a threshold to calculate recall@k. On the other hand, for each document and each listing rule, we regard the top k results with the highest probabilities as predicted answers and then calculate recall@k. The recall@k measure is directly related to our system user experience. For each document and each listing rule, our system always returns the top 5 results, each of which has a probability of relevance, as shown in Figure 1. Thus, recall@k, and especially recall@5, indicates the percentage of relevant passages that are shown to reviewers. There are also micro and macro metrics for recall@k. They are formulated in a similar way as follows:

$$
R@k_{micro} = \frac{\sum_{j=1}^{m}\sum_{d\in D} Q_{j,k}^{d}}{\sum_{j=1}^{m}\sum_{d\in D} Q_{j,n_d}^{d}} \qquad R@k_{macro} = \frac{1}{m}\sum_{j=1}^{m} \frac{\sum_{d\in D} Q_{j,k}^{d}}{\sum_{d\in D} Q_{j,n_d}^{d}} \tag{4}
$$

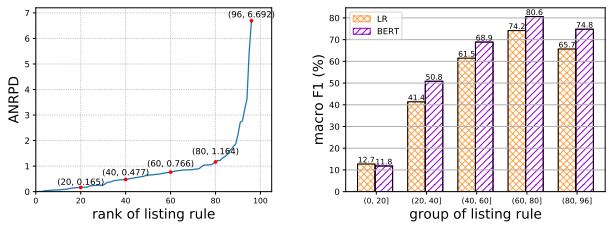
where d is a document in the data set D. For listing rule j and document d, $Q_{j,k}^{d}$ is the number of recalled passages if we only consider the top k predicted passages.

In this experiment, we explore how to utilize the information of documents to improve model performance. We use different contextual information to form the final feature of a passage, including the passage itself, ancestors (including its parent), and preceding passage.

To mitigate the problem of overfitting the unlabeled but relevant data, we select different negative samples for training in each epoch and test different *Negative Sampling Ratio (NSR)*, the ratio of preserved negative samples, for training. Another benefit of lower NSR is to accelerate the training process, especially for the BERT model, considering negative samples contain limited useful information and are nearly 20 times more than positive samples.

Table 1: Result of different models and settings.

Features	NSR(%)	Micro (%)			Macro (%)		
		P	R	F	P	R	F
LR							
all features	50	72.8	57.6	**64.3**	62.5	44.8	50.5
self+pre	50	67.5	44.2	53.4	59.1	40.8	46.7
par+self+pre	50	66.6	52.3	58.6	62.2	43.7	49.4
anc+self	50	71.7	54.8	62.1	58.4	42.0	47.7
all features	10	55.1	**68.6**	61.2	52.2	**53.6**	51.2
all features	20	63.1	63.8	63.4	56.7	50.4	**51.8**
all features	100	**78.6**	50.4	61.5	**65.1**	39.0	47.2
BERT							
all features	50	**74.8**	71.5	**73.1**	60.6	54.7	**56.7**
self+pre	50	67.4	59.7	63.3	54.8	54.4	53.4
par+self+pre	50	67.0	67.0	67.0	56.8	54.1	53.8
anc+self	50	74.6	68.3	71.3	**61.2**	51.7	54.1
all features	10	60.3	**78.4**	68.2	50.6	**58.9**	53.1
all features	20	68.6	76.5	72.4	54.0	58.6	55.4
all features	100	74.8	64.9	69.5	60.4	47.4	50.7



(a) (b)

Figure 5: The relevance between the number of relevant passages and F1. Listing rules are ranked according to the average number of relevant passages per document (ANRPD) in (a). Listing rules are split into five groups according to their rank in (b). Macro F1 is used to represent the average performance of each group and corresponding model.

4.3 Results Analysis

Table 1 summarizes the results under different settings. The top block is the result of the LR model, and the bottom block is about the BERT model.

The best result is achieved when NSR is 50% and all features are used, including ancestors (anc), the passage itself (self) and the preceding node (pre). Under such experimental settings, LR can attain 64.3% in micro F1 and 50.5% in macro F1. BERT outperforms LR and attains 73.1% in micro F1 and 56.7% in macro F1. We then analyze the ablation results as follows.

• *Logical hierarchy.* To test the importance of the logical hierarchy of the document, we remove the feature from ancestors. The performances of LR and BERT both drop sharply (over 10%). This

indicates that the logical hierarchy is very useful for locating relevant passages. We also conduct experiments to explore whether the information of the parent node is enough for this task. We preserve the parent node but drop the other ancestor nodes, which is the "par+self+pre" row. Under this setting, model performance still decreases, indicating a likelihood that information is conveyed through the logical hierarchy of the document. Therefore, it is necessary to restore the complete hierarchy.

• *Preceding passage.* The content of a document should be coherent and two neighboring passages should be connected by logic. Especially when the passage is a table, the previous passage is likely to explain the content of the table. The performance drops about 2% after removing the preceding passage, which indicates the preceding information is useful.

• *Negative sampling.* When NSR increases from 10% to 100%, for both LR and BERT, the precision drops and the recall increases steadily. Using only 10% of the negative samples can attain similar F1 compared to using all data. The best result is obtained when NSR equals 50%, showing the balance of precision and recall. Another possible reason is that using different negative samples to train the model in different epoch mitigates overfitting.

It can be observed that macro F1 is smaller than micro F1 in general. This is because different listing rules vary a lot in the number of positive samples available for training. The micro metric focuses on the sum of different listing rules. Listing rules with large numbers of positive samples dominate the micro metric, and the model can learn better with large data. Therefore, the micro metric is higher than the macro metric.

To further analyze this phenomenon, we explore the relevance between the Average Number of Relevant Passages per Document (ANRPD) of different listing rules and their performance. The result is shown in Figure 5. Figure 5a shows the ANRPD of each listing rule sorted in ascending order, which is diverse. ANRPD of the last listing rule is 6.692, but is less than 0.165 for 20 rules. We then split the 96 listing rules into 5 groups according to their ranks. The first four groups contain 20 listing rules and the last group contains 16 listing rules. Then, we calculate macro F1 within each group as Figure 5b shows. The results are evaluated under the best setting with NSR as 50%. In the first group, both LR and BERT perform poorly (about 10% F1 score). As the rank increases, performance also increases, reaching 80.6% for the (60, 80] group. It indicates that performance improves with the increasing availability of tagged and annotated data. It also verifies our analysis about the difference between macro and micro F1. But the last group shows a performance drop. We think this might be due to the fact that relevant passages might be missed/unlabeled when there are multiple relevant passages, so that the tagging/annotation in the last group might have more noises of unlabeled but relevant passages.

Finally, we evaluate the recall@k metric for the best models of LR and BERT in Table 2, following Equation 4. Different from Table 1, Table 2 has an upper bound of recall@k. The upper bound is obtained by using the correct passages as predicted answers. The upper bound of recall@5 cannot reach 100% because there are listing rules that have more than 5 relevant passages in a document. The upper bound of macro recall@k is much higher than the upper bound of micro recall@k because of the diverse ANRPD. The performance of our model is close to the upper bound. It means that, in most cases, users of our system only need check whether the returned results are relevant, without reading the whole document.

Table 2: Results on recall@k

Models	Micro (%)			Macro (%)		
	R@1	R@3	R@5	R@1	R@3	R@5
upper bound	**52.5**	**72.6**	**80.8**	**80.2**	**93.7**	**96.3**
LR	43.9	64.0	72.4	59.4	80.3	88.2
BERT	45.9	67.1	75.4	61.6	84.0	89.9

To summarize, logical hierarchy information is helpful for locating relevant passages. We need a panoptic document recognition result to get all the ancestors of a passage, in order to get its full semantic meaning. A proper ratio of negative sampling may help balance recall and precision, reduce training time and overfitting on the unlabeled but relevant data. The performance of the model against different listing rules differs greatly because of the diverse ANRPD. More tagged/annotated data is the key to success. Overall, BERT outperforms LR and its recall@5 is close to the upper bound, utilizing logical hierarchy and an appropriate NSR.

5 RELATED WORK

5.1 Compliance System

Currently, there are several researches about automated compliance assessment on regulatory documents. In the construction field, Zhang et al. [30] design a system for automated compliance checking using a set of pattern-matching-based rules and conflict resolution rules in information extraction. In the data protection field, John [10] uses rule-based and machine learning methods to support checklist interpretation and risk analysis. In the finance field, the Cognitive Compliance function [26] assesses risks in financial advice documents and the Numerical Cross-Checking system [4, 12] checks the consistency of values of financial indexes within a financial document. But no full-fledged solution can incorporate a hundred of diverse listing rules. Our purpose is to explore the way towards automated compliance assessment on complex documents and against various rules. We also noted many financial institutions have successfully constructed models to detect simple pieces of information inside similar documents (e.g. earnings per share, publication date, interest rate…).

5.2 Passage Location

To locate relevant passages with regard to each listing rule, there are mainly two ways: a query-dependent method or a query-independent method. As for the query-dependent method, the task of locating relevant passages can be formulated as a passage retrieval problem. The retrieval problem has been studied for a long time, and there are many classical models, including Boolean model [11], vector space model [25], probabilistic model [24] and feature-based model [16]. Among them, BM25 is representative of probabilistic models and often used as a baseline because of its excellent performance. After BERT [6] was proposed, learning-to-rank methods gradually attracted the attention of many researchers. As for the query-independent method, the task can be formulated as a classification problem. For each listing rule, we can classify each passage

as to whether it is relevant or not. There is plenty of work proposed to solve the classification problem, which can be divided into two groups: traditional methods [7, 28] and deep-learning methods [6, 9]. Among traditional methods, LR is widely used due to its high performance and accuracy. Among deep-learning methods, BERT is the state-of-the-art model because of its attention mechanism and pre-training process on large corpus.

5.3 Physical and Logical Structure Recognition

Physical structure recognition focuses on dividing documents into flat segmentations, that is, an ordered list of physical objects (e.g. tables, paragraphs, figures, etc.). Most current methods consider the task as an object detection or semantic segmentation problem using frameworks like Faster R-CNN and Mask R-CNN to predict the bounding box of each physical object [13, 14], or detect the contour of each physical object and classify pixels in it (by FCN, VGG) to determine the type [29].

The discovery of logical document hierarchy is a conventional task. We introduce some learning-based methods. The first category is sequence labelling-based which classifies the absolute hierarchical depth of each physical object on the object sequence using RNN and LSTM [1]. The second category is tree generation-based. Pembe et. al. [22] propose a tree-based learning approach to generate a logical hierarchy node by node by considering the containment and parallel relationships between nodes.

6 CONCLUSIONS

An effective and accurate compliance regulatory assessment tool is highly important for improving the quality of issuers' published materials and the transparency of information for the investing public. However, the goal of evaluating hundreds of types of regulatory disclosures inside annual reports posed some formidable challenges. Due to the diversity and complexity of listing rules and annual reports, the cost of training a human reviewer is high. Even for a seasoned reviewer, assessing an annual report may still consume, on average, 110 minutes. Therefore, we design the `Jura` system to assist with disclosure and compliance assessment, and effectively reduce the time taken by around 80% on average without the decrease of accuracy, while expanding the achievable breadth of assessment coverage. This system has become a part of HKEX's regulatory toolkit.

To tackle the diversity of listing rules and complexity of annual reports, we break down the whole process into four steps: panoptic document recognition, relevant passage location, fine-grained information extraction and compliance assessment. The categorization groups similar listing rules together so that similar methods can be used for the same group of listing rules. Particularly, the relevant passage location step uses a unified passage classification approach for all listing rules. Leveraging document hierarchy extracted from the first step, the relevant passage location model achieves a near 90% with regard to the metric recall@5.

7 ACKNOWLEDGEMENTS

The research work supported by the National Key Research and Development Program of China under Grant No. 2017YFB1002104, the National Natural Science Foundation of China under Grant No. 62076231, U1811461.

REFERENCES

[1] Najah-imane Bentabet. 2019. Table-Of-Contents generation on contemporary documents. In *ICDAR*.
[2] Jean-Luc Bloechle. 2010. *Physical and logical structure recognition of pdf documents*. Ph.D. Dissertation. University of Fribourg.
[3] Rongyu Cao, Yixuan Cao, Ganbin Zhou, and Ping Luo. 2021. Extracting Variable-Depth Logical Document Hierarchy from Long Documents: Method, Evaluation, and Application. *Journal of Computer Sciecne and Technology* (2021).
[4] Yixuan Cao, Hongwei Li, Ping Luo, and Jiaquan Yao. 2018. Towards Automatic Numerical Cross-Checking: Extracting Formulas from Text. In *WWW*.
[5] Alan Conway. 1993. Page grammars and page parsing: A syntactic approach to documentlayout recognition. In *ICDAR*.
[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*.
[7] Thorsten Joachims. 1997. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization.
[8] Duff Johnson. 2014. The 8 most popular document formats on the web. *Duff Johnson Strategy and Communications Blog* (2014).
[9] Yoon Kim. 2014. Convolutional neural networks for sentence classification. (2014).
[10] John Kingston. 2017. Using artificial intelligence to support compliance with the general data protection regulation. *Artificial Intelligence and Law* (2017).
[11] Arash Habibi Lashkari, Fereshteh Mahdavi, and Vahid Ghomi. 2009. A boolean model in information retrieval for search engines. In *2009 International Conference on Information Management and Engineering*.
[12] Hongwei Li, Qingping Yang, Yixuan Cao, Jiaquan Yao, and Ping Luo. 2020. Cracking Tabular Presentation Diversity for Automatic Cross-Checking over Numerical Facts. In *KDD*.
[13] Kai Li, Curtis Wigington, Chris Tensmeyer, Handong Zhao, Nikolaos Barmpalios, Vlad I. Morariu, Varun Manjunatha, Tong Sun, and Yun Fu. 2020. Cross-Domain Document Object Detection: Benchmark Suite and Method. In *CVPR*.
[14] Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. 2019. TableBank: Table Benchmark for Image-based Table Detection and Recognition. *International Conference on Document Analysis and Recognition (ICDAR)*.
[15] Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. 2020. DocBank: A Benchmark Dataset for Document Layout Analysis. *ArXiv* (2020).
[16] Tie-Yan Liu. 2009. Learning to Rank for Information Retrieval. *Found. Trends Inf. Retr.* (2009).
[17] Donato Malerba, Michelangelo Ceci, and Margherita Berardi. 2008. Machine learning for reading order detection in document image understanding. *Studies in Computational Intelligence* (2008).
[18] Tomohiro Manabe and Keishi Tajima. 2015. Extracting logical hierarchical structure of HTML documents based on headings. In *VLDB*.
[19] Song Mao, Azriel Rosenfeld, and Tapas Kanungo. 2003. Document Structure Analysis Algorithms: a Literature Survey. *Document Recognition and Retrieval* (2003).
[20] Jean Luc Meunier. 2005. Optimized XY-cut for determining a page reading order. In *ICDAR*.
[21] Juri Opitz and Sebastian Burst. 2019. Macro F1 and Macro F1. *ArXiv* (2019).
[22] F. Canan Pembe and Tunga Güngör. 2010. A Tree Learning Approach to Web Document Sectional Hierarchy Extraction. In *ICAART*.
[23] Muhammad Mahbubur Rahman and Tim Finin. 2017. Understanding the Logical and Semantic Structure of Large Documents. *CoRR* (2017).
[24] Stephen E Robertson and K Sparck Jones. 1976. Relevance weighting of search terms. *Journal of the American Society for Information science* (1976).
[25] Gerard Salton, Anita Wong, and Chung-Shu Yang. 1975. A vector space model for automatic indexing. *Commun. ACM* (1975).
[26] Wanita Sherchan, Sue Ann Chen, Simon Harris, Nebula Alam, Khoi-Nguyen Tran, and Christopher J Butler. 2020. Cognitive Compliance: Assessing Regulatory Risk in Financial Advice Documents.. In *AAAI*. 13636–13637.
[27] Alfred Z Spector, Joshua J Bloch, Dean S Daniels, Richard Draves, Dan Duchamp, Jeffrey L Eppinger, Sherri G Menees, and Dean S Thompson. 1986. The camelot project. *IEEE Database Eng. Bull.* (1986).
[28] Raymond E Wright. 1995. Logistic regression. (1995).
[29] Yang Xiao, Ersin Yumer, Paul Asente, Mike Kraley, Daniel Kifer, and C. Lee Giles. 2017. Learning to Extract Semantic Structure from Documents Using Multimodal Fully Convolutional Neural Network. In *CVPR*.
[30] Jiansong Zhang and Nora M El-Gohary. 2016. Semantic NLP-based information extraction from construction regulatory documents for automated compliance checking. *Journal of Computing in Civil Engineering* (2016).
[31] Xu Zhong, Jianbin Tang, and Antonio Jimeno Yepes. 2019. PubLayNet: largest dataset ever for document layout analysis. In *International Conference on Document Analysis and Recognition (ICDAR)*.