January 2021

DOCUMENT GENERATOR BASED ON RANDOMIZED TEMPLATES

HP INC

Invention Title

Document generator based on randomized templates

Abstract

This disclosure relates to the field of synthetic document generation for training Deep Learning algorithms in order to understand document contents. Document understanding is important for applications such as document quality enhancement and information extraction pipelines. In document quality enhancement, different computer vision techniques are applied to specific regions of the document depending on the element type (text, image), for tasks like printing and/or scanning. Information extraction pipelines aim to retrieve valuable knowledge from documents in an automated fashion. Again, depending on the element type, different extractors are used. Machine Learning techniques may be applied to decompose a document into element types: text, images, equations, charts, and diagrams. Regardless of the training regime (supervised or unsupervised), data is necessary. An option could be to obtain documents from the Internet. However, there are some problems: No permissive license, unbalanced data (i.e. slides with only text elements), and difficulty to extract precise annotations for training ML models from raw documents. This disclosure presents a synthetic data generator able to create a diverse set of documents based on randomized template formats, here we focus on slide presentations.

Problems Solved

The problem of obtaining huge amounts of diverse document types, with fine grained annotations, in order to train machine learning models to recognize the information contained in any document.

Prior Solutions

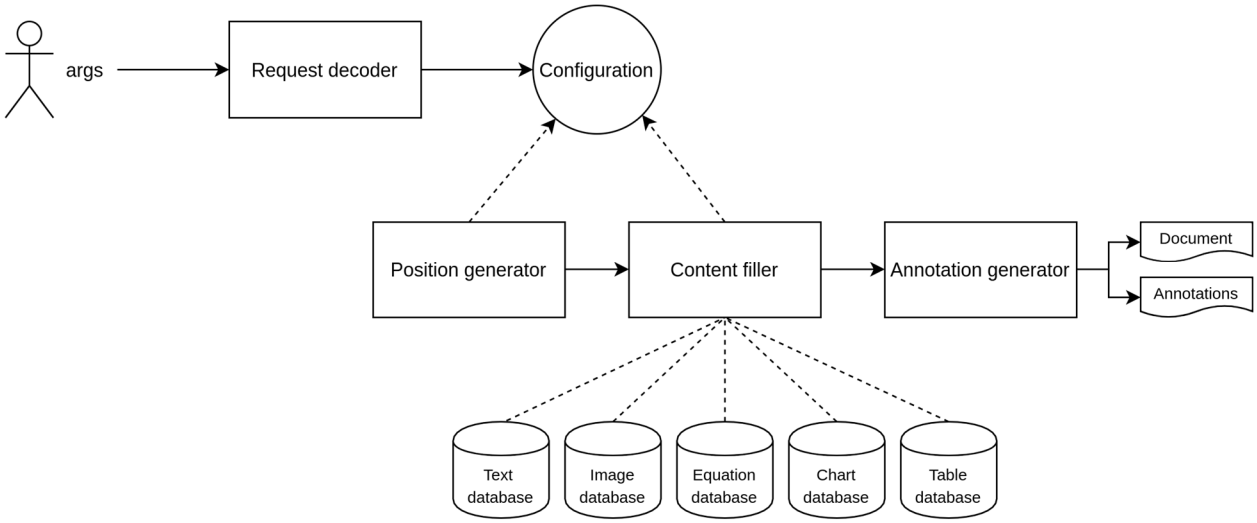
On the generation of slide-oriented content, [1] proposed an automatic algorithm to detect presentation slide transitions in lecture video streams, working towards detecting slides in videos, but without any reasoning approach over them. [2] creates presentation slides from papers. Using a tool to extract the text from PDF documents, it uses a model, composed of LSTM and MLP, to score the sentences in the text. Then, it selects the scored sentences in a greedy search, using the selected sentences in bullets points on a presentation slide. Although the approach generates slides, it is limited to academic slide styles and cannot create slides from scratch, needing real documents to perform the generation. [3] also generates presentation slides from academic papers but creating the presentation from the Lateχ source code, parsing it to an XML document. This approach, limited to the academic style and requiring prior content for slide generation, as in Sravanthi \etal~\cite{sravanthi2009slidesgen}, also needs that the sections of the work are limited to the scope of introduction, related work, model, experiments, and conclusions.

Description

The proposed application generates synthetic documents as a combination of pre-determined templates and content randomization techniques. In this description, slide presentations are shown, but the generator can be generalized to other types of documents. The generator is highly configurable, being able to introduce variations in the generated documents (e.g., font, background color, number of columns, etc.) as shown in the generated slide presentations on the figure below.

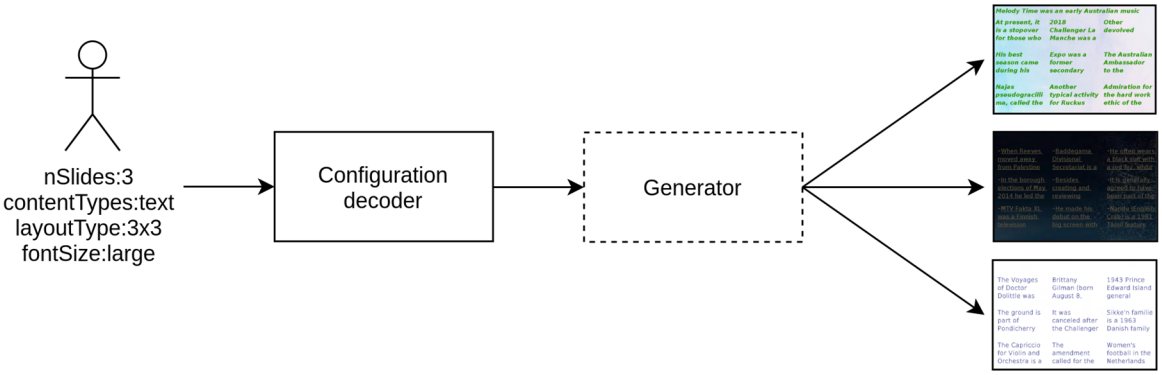
The pipeline for generating slides is as follows:

- Decodes the provided configuration into instructions to create a set of slides.
- Defines the base template for the slide document.
- Selects the global style for the elements in the slide.
- Fills the slide with content in randomized positions according to the base template.
- Generates the associated annotations for ML model training


args
Request decoder
Configuration
Position generator
Content filler
Annotation generator
Document
Annotations
Text database
Image database
Equation database
Chart database
Table database

1 - Configuration Decoder

Transcribes the high-level user configuration format into low level instructions according to the generator internal APIs.



As described in the next sections, all elements which are not explicitly defined are selected according to a random distribution. That allows the control over the generated slides, at the same time that randomization is added over all other visual aspects.

2 - Template Format

According to a generation criteria manual or random distribution, selects a template for the slide. Some examples of templates are:

- Single column content with title
- Two columns with title and subtitle
- Three columns with title and footer
- Grid layout content (nxn) with no title

3 - Global Style

According to a generation criteria manual or random distribution, selects properties such as:

- Background slide color
- Title font size, family, and color
- Text font size, family, and color for normal text
- Bullet format: circle, triangle, dash

4 - Content Randomization

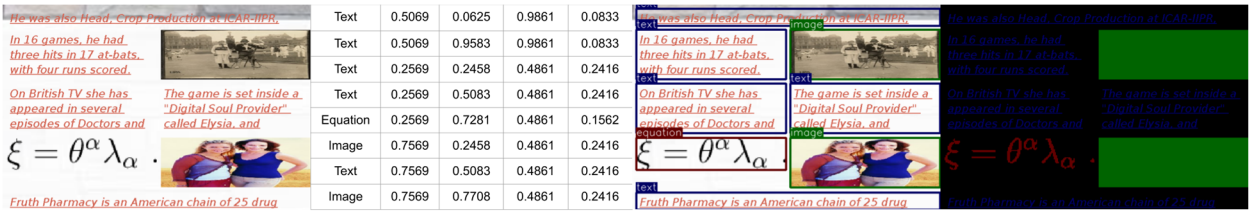
That is the heart of the generation process. According to the layout defined in the base template (single/multiple columns, or grid) the document is filled with contents. The process consists of an iterative procedure executed until no more free space is available in the slide.

First the element size and placement are randomly defined based on the available free space in the document.

Then the element type is selected: text, bullet points, tables, equations, charts, or photos. According to the type, the content itself is retrieved from the associated dataset. The datasets and the associations with element types are configurable in the application. For example, the ImageNet dataset can be associated with photos, and the Wikipedia Sentences with text. This customization allows the synthetic data to be closer to a particular information domain.

5 - Annotations

With all elements placed in the document, the last step is the generation of the annotations for training ML models. Currently, it is created annotations for object detection and segmentation tasks. For object detection the bounding boxes along with the corresponding types for each element are saved. For segmentation, the pixels contained into the mask reflect the type and position of each element.



Advantages

- Can be used as part of document enhancement pipelines, for training ML models to recognize different element types in printed/scanned documents. We believe it may be particularly useful for the Printing BU services.
- Mitigates the problem of obtaining huge amounts of data, as well as the expensive labeling process.
- Another advantage of controlling the generation process is that we can further extend the solution to other document types.
- Generated content can be customized by associating external datasets to element types
- Can prepare the synthetic data to be consumed by different Machine Learning tasks, e.g., for object detection or segmentation.

References

[1] Baoquan Zhao, Shujin Lin, Xin Qi, RuomeiWang, and Xiaonan Luo. A novel approa ch to automatic detection of presentation slides in educational videos. Neural Co mputing and Applications, 29(5):1369–1382, 2018.
[2] Athar Sefid, Jian Wu, Prasenjit Mitra, and C Lee Giles. Automatic slide gener ation for scientific papers. In SciKnow@ K-CAP, pages 11–16, 2019.
[3] M Sravanthi, C Ravindranath Chowdary, and P Sreenivasa Kumar. Slidesgen: Auto matic generation of presentation slides for a technical paper using summarization . In Twenty-Second International FLAIRS Conference, 2009.

Disclosed by Juliano Vacaro, Luana Müller, Andrey de Aguiar Salvi, Eduardo Henrique Pais Pooch, Alessandra Helena Jandrey, Vinicius Parmeggiani Vianna and Pedro Portella Possamai, HP Inc