# PostDoc: Generating Poster from a Long Multimodal Document Using Deep Submodular Optimization

Vijay Jaisankar<sup>a</sup>, Sambaran Bandyopadhyay<sup>b,\*</sup>, Kalp Vyas<sup>c</sup>, Varre Chaitanya<sup>c,\*\*</sup> and Shwetha Somasundaram<sup>b</sup>

<sup>a</sup>International Institute of Information Technology, Bangalore

<sup>b</sup>Adobe Research

<sup>c</sup>IIT Bombay

**Abstract.** A poster from a long input document can be considered as a one-page easy-to-read multimodal (text and images) summary presented on a nice template with good design elements. Automatic transformation of a long document into a poster is a very less studied but challenging task. It involves content summarization of the input document followed by template generation and harmonization. In this work, we propose a novel deep submodular function which can be trained on ground truth summaries to extract multimodal content from the document and explicitly ensures good coverage, diversity and alignment of text and images. Then, we use an LLM based paraphraser and propose to generate a template with various design aspects conditioned on the input content. We show the merits of our approach through extensive automated and human evaluations.

## 1 Introduction

Recent success of large language models [24, 26] and large vision language models [28, 17] have led to several new applications in the field of generative AI. In this work, we focus on transforming a long multimodal document, containing both text and images, into a poster, which is a visually rich one-page multimodal summary of the document. A poster should have a good coverage of the overall content of the document and, is generally easy-to-read and presented in a nice template with good design elements [29]. To create a poster, there are different types of design tools and products available such as Microsoft PowerPoint <sup>1</sup> and Adobe Express <sup>2</sup>. However, these products need significant manual effort to select the multimodal content from a document and choose the appropriate design elements in the poster. Such manual efforts are time consuming and often need specific domain expertise based on the document.

Automatic transformation of a document to a poster is a less studied problem in the research literature and industry [27, 34, 35]. Such a transformation process mainly involves two major steps: (i) Content summarization (or planning), which aims to select key content from the document and paraphrase them in some appropriate format to be put in the poster and (ii) Template generation and harmonization, which involves generating a suitable layout and design elements

such as background of the poster, font and size of letters, number of text and image elements etc. of the poster based on the output of the content planning step and fill up the generated template with the generated content. Content summarization for poster is challenging because of the following reasons. A poster is very limited in size (single large page), but the input document can have multiple pages. Thus, it is essential that the content in the poster (i) represents all the important aspects of the input document (coverage), (ii) has very less repetition within it (diversity) and (iii) has well aligned images and text. Coverage and diversity have been studied in text summarization literature [18]. But their interpretation in the multimodal setup (with text and images) is not well-understood.

Recent advent of zero-shot and few-shot LLMs, especially GPT-3.5-turbo (ChatGPT) and GPT-4 [24] have significantly improved the state-of-the-art (SOTA) for several natural language processing tasks like summarization. However, using them directly for the content summarization of poster generation is not feasible because: (i) Current LLM models (including GPT-4) cannot produce multimodal output and their ability to process multimodal input is still questionable; (ii) Since we focus on long documents as input, LLM based algorithms cannot process that lengthy text at single shot. There are techniques which divide long text into chunks and feed each chunk separately to an LLM for summarization [2]. Researchers have also tried to improve the context length of LLMs using length extrapolation [25] and position interpolation of transformers [5]. However, such approaches lose the global view of the document, computationally expensive or cannot be used with black-box LLMs. Moreover, LLMs tend to hallucinate and their performance drops when the input context is very long [19].

In this paper, we have addressed all the challenges mentioned above by proposing an approach to handle multimodal content jointly and avoid feeding the entire content to an LLM directly. Following are the key contributions made in this work: (1) We propose an efficient and computationally fast end-to-end pipeline, referred as *Post-Doc* (in Figure 1), for automatically generating a visually rich *Poster* from a long multimodal input *Doc*ument. (2) For selecting suitable content from the input document, we propose a novel optimization formulation using deep submodular functions which explicitly ensures coverage, diversity and multimodal content alignment in the multimodal summary. This summary is passed to an LLM (GPT-3.5-turbo, a.k.a., ChatGPT) for paraphrasing to put into a poster. (3) Based on the input document and the selected content, we generate a

<sup>\*</sup> Corresponding Author. Email: samb.bandyo@gmail.com

<sup>\*\*</sup> Vijay Jaisankar, Kalp Vyas and Varre Chaitanya were interns at Adobe Research when the work was conducted.

<sup>&</sup>lt;sup>1</sup> https://create.microsoft.com/en-us/templates/posters

<sup>&</sup>lt;sup>2</sup> https://www.adobe.com/express/

suitable poster template with different design elements to ensure that the poster looks good aesthetically. (4) We conduct thorough experimentation to validate the quality of the content and design aspects of the generated posters through automated and human evaluations.

# 2 Related Work and Background

# 2.1 Multimodal Summarization

Text summarization has been studied quite extensively in the literature [36, 39]. From the last few years, researchers have focused into multimodal summarization which involves different modalities such as text, images and videos, and leverages cross-modal information [21, 16]. In this section, we focus on works which consider both input and output to be multimodal, as required for transforming a document to a poster. Zhu et al. [40] generated abstractive multimodal summary from a document, but it is trained by the target of text modality, leading to the modality-bias problem. Zhu et al. [41] extended it by including images along with the text but is currently limited to generating summaries with only one image. Zhang et al. [38] further improved on these methods by using BART and knowledge distillation which does a better job of image selection but still treats image and text as different entities. He et al. [10] presented A2Summ, a novel unified transformer-based framework for multimodal summarization primarily focusing on text and video modalities. Zhang et al. [37] proposed a technique to summarize the document based on using the hierarchy in the form of a graph network but also faces the issue of choosing only one image for the summary.

Submodular functions have been used for multiple summarization tasks [18, 32, 22, 40] as they are a natural fit for text summarization. A set of differentiable submodular functions, also known as deep submodular functions [3, 15] have been proposed in the literature and also used for text summarization. However, they have not incorporated the multimodal aspects such as multimodal coverage, diversity, and image-text alignment terms which are key components for multimodal summarization. In this work, we address this research gap and aim to design a deep submodular function which captures a set of intrinsic properties of multimodal extractive summarization and also trainable from the ground truth data.

## 2.2 Layout Generation

Regarding the design aspects, Qiang et al. [27] address poster layout generation using recursive divisions, yet within the context of scientific papers. Gupta et al. [9] propose a layout generation method. However, it lacks conditional generation. On the other hand, Chai et al. [4] generate layouts conditionally based on specific requirements, such as the count and placement of text and image bounding boxes. Hsu et al. [11] generate layout depending on the structure of the background image being used but this method too lacks conditional generation. These methods are computationally expensive, but not effective in generating posters with non-overlapping bounding boxes. Our work addresses these limitations by adopting a simple heuristic-based layout generation approach specifically for posters. Additionally, we enhance poster aesthetics by considering design elements such as colors, fonts, and backgrounds, based on the content.

# 2.3 Document Transformation

There are few existing works related to poster creation from documents. Xu and Wan [34] generates posters from documents but relies on template retrieval from a fixed set of templates and also limiting its

applicability to research papers exclusively. There are few research works on generating slides from scientific documents automatically [31, 7]. But they often need users to come up with an outline specific to slide presentation or summarize individual sections in each slide.

## 2.4 Background on Submodular Functions

Here, we discuss some key concepts required to understand our solution. A submodular function f is a set function with diminishing returns property. In simple terms, adding an element to a smaller set provides a larger marginal gain compared to adding the same element to a larger set. Mathematically, for sets  $A \subseteq B$ , on adding extra element  $x \notin B$ ,  $f(A \cup x) - f(A) \ge f(B \cup x) - f(B)$  [18]. A key advantage of using a submodular function is that there exists a simple greedy algorithm of iteratively choosing the element that maximises the marginal gain with an approximation guarantee.

Recently, researchers explore deep (trainable) submodular functions [3, 15] where the data is projected into a suitable feature or embedding space. They can be represented as:  $f(A) = \sum_{u \in \mathbb{U}} \Phi\Big(w(u)m_u(A)\Big)$ , where  $\Phi$  is a non decreasing non-negative concave function, w(u) represents the trainable weight of the feature  $u, m_u(S) = \sum_{s \in S} m_u(s)$  is a non-negative modular function. These functions enable the learning of submodular functions through a training dataset and also their inference is fast compared to the quadratic complexity of most of the other submodular functions.

# 3 Problem Statement and Solution Approach

As discussed in Section 1, we aim to automatically generate a poster from a long document. The document may contain different types of multimodal content such as text, images, tables, charts, etc. For the ease of presentation, we use image to represent all such non-textual elements in a document. As shown in Figure 1, we use Adobe Extract API <sup>3</sup> to extract the multimodal content from the document. Then, we use the pre-trained multimodal model BLIP [17] to encode both text and image elements into a common vector space of dimension 768. We have observed that the embeddings of text and images are often not in the same scale. To overcome this issue, we first shift all the embeddings to positive coordinate of the embedding space and do a L1 normalization of the embeddings.

With this, we present the problem mathematically as follows. Given  $D = (e_1, e_2, \dots, e_N)$  is a multimodal input document with N (can vary over the documents) content elements in order where each content element can be a text sentence or an image. We assume that each  $e_i \in \mathbb{R}^d_+$  denotes a d dimensional normalized BLIP embedding (as discussed above) of the ith content element in the document (d = 768 in this case). Our goal is to select a subset of content elements  $A \subseteq D$  with  $|A| \le K$  so that content elements of A have good coverage, diversity and alignment as discussed in Section 1. Since the size of the poster is limited, we extract the corresponding maximum K>0 content elements from the document. We discuss how to fix K in Section 4. For training our content selection algorithm, we use a training set of documents with their ground truth summary. So, we assume to have the following training set  $\mathcal{T} = \{(D_1, A_1^*), (D_1, A_1^*), \cdots, (D_M, A_M^*)\}$  containing M pairs of documents and the corresponding ground truth summary. During the inference for a given document D, we will first select the subset Ausing the trained content selection model. Then we will paraphrase A into a poster friendly format. We also generate a poster template

<sup>3</sup> https://developer.adobe.com/document-services/apis/pdf-extract/

with suitable design elements based on the content and put the paraphrased content into it.

### 3.1 Multimodal Extractive Summarization

Once the embeddings of all the content elements are obtained from a document, the next step is to select only a subset of them such that some desired properties are satisfied. We propose a novel deep submodular based optimization framework for this task. As mentioned in Section 3, the normalized BLIP embeddings of the sequence of content elements (text sentences or images) from the document are represented as  $D=(e_1,e_2,\cdots,e_N)$ , with  $e_i\in\mathbb{R}^d_+$ . In this subsection, our goal is to extract a subset of content elements  $A\subseteq D$ with  $|A| \leq K$  such that the following properties are preserved in the extracted subset: (1) Coverage: We want the content elements of A to represent the whole document D well. Thus, it is expected that any content element in A to be more similar to many elements in D. (2) Diversity: Since the poster is of very limited size, we want to avoid unnecessary redundancy in content present in the poster. This property ensures that any content element in A is very different from most of the other elements in A. (3) Multimodal Alignment: The output poster contains both images and text. Images and text contain complementary information. Hence, it is important to ensure that the images and text present in the poster are aligned with each other. For e.g., the text present in the poster can brief about the image. Thus, any image element in A should be similar to some text elements in A and vice-versa. (4) Ground Truth Data Adaptability: All the above properties are different intrinsic properties expected in a summarization. However, ground truth summarization data may have other hidden properties which may not be captured above. So, we also want our algorithm to learn from the set of multimodal ground truth summary data.

Thus, A can be considered as an extractive multimodal summary of the multimodal document D where our goal is to design a summarization model which is trainable and equipped with the inductive bias as mentioned above. Next, we try to construct an objective function to achieve this.

$$f(A) = \sum_{u \in [d]} w_u \sqrt{\sum_{x \in A} \sum_{y \in D} x_u y_u - \sum_{x \in A} \sum_{y \in A} x_u y_u} + \sum_{x \in A} \sum_{y \in A_T} x_u y_u + |D| \sum_{x \in A} x_u$$
(1)

Here,  $[d]=\{1,2,\cdots,d\}$  denotes the set of dimensions of  $\mathbb{R}^d$ . We use a vector of trainable weight parameters  $w=[w_1,w_2,\cdots,w_d]^T\geq 0$  which intuitively captures the importance of each dimension of the embedding space. Ideally for a given document D, we would like to choose the subset  $A\subseteq D$  which maximizes f(A). The term  $\sum_{x\in A}\sum_{y\in D}x_uy_u$  captures the similarity of an element  $x\in A$  to an element  $y\in D$  for the uth dimension. Since both x and y are L1-normalized, this term over the outer summation contributes more when x and y are similar to each other. Thus, the first term captures the notion of coverage. Similarly, the second term  $\sum_{x\in A}\sum_{y\in A}x_uy_u$  captures the similarity of content elements within the extracted multimodal summary A. Thus, the negation of this term can be considered as the diversity of the elements within A. So far, we have not differentiate between the text and images within A. However, selected images in A need to be aligned with the text present in the poster. The third term  $\sum_{x\in A_I}\sum_{y\in A_I}x_uy_u$  in Equation 1 ensures multimodal alignment, where  $A_I$  is the set of images in A and  $A_T$  is the set of text sentences in A. The last term

 $|D|\sum_{x\in A} x_u$  is introduced to ensure some nice mathematical property of our loss function. Since,  $x\in D$  are L1-normalized, the last term is a constant if  $w_u=\frac{1}{d}, \forall u\in [d]$  (initial condition as discussed in Section 3.2).

Next, we want to design a loss function to train the parameters w of our model. As mentioned in Section 3, we are given with a training set of multimodal documents with the corresponding ground truth summaries as  $\mathcal{T} = \{(D_1, A_1^*), (D_1, A_1^*), \cdots, (D_M, A_M^*)\}$ . For any  $(D_i, A_i^*) \in \mathcal{T}$ , we want the value of the function f on our model generated summary to be close to  $f(A_i^*)$ . We consider the following hinge loss function here:

$$\min_{w \ge 0} \sum_{i=1}^{M} \left( \max \left( \max_{\substack{A \subseteq D_i \\ |A| \le K}} \{ f(A) \} - f(A_i^*), 0 \right) + \frac{\lambda}{2} ||w||_2^2 \right)$$
 (2)

In the above equation, we also use an L2 regularizer on w with a weight hyperparameter  $\lambda \geq 0$ . Based on the performance on validation set, we keep  $\lambda = 0.1$  for all our experiments. The predicted summary is obtained by maximizing f(A) w.r.t. A such that  $A \subseteq D$  with  $|A| \leq K$ . By minimizing the hinge loss w.r.t. the trainable nonnegative weight parameters w ensures that the maximum of  $\{f(A)\}$  (i.e., on model predicted summary) is not too far less from the ground truth summary on the training data. We discuss the solution strategy and training of this optimization problem next.

## 3.2 Training and Optimization

The optimization function in Equation 2 is a constrained min-max optimization on two different types of variables. Here, w is continuous and non-negative. But A is a subset of with a fixed cardinality. To solve this, we use an iterative alternating optimization strategy as discussed below.

# 3.2.1 Maximization w.r.t. A

Let us first focus on maximizing the objective w.r.t. A while keeping w fixed. Then it is essentially a subset selection problem for each  $D_i$ ,  $\forall i=1,2,\cdots,M$ . Subset selection problems are typically combinatorial in nature and often computationally infeasible. But the following theorem shows an important property of f which will help us to solve the optimization problem.

### Theorem 3.1. The set function f in Equation 1 is a monotone submodular function.

*Proof.* Let the image-text alignment term of f(A) be  $h(A) = \sum_{x \in A_I} \sum_{y \in A_T} x_u y_u$ . The function f in Equation 1 can be simplified to  $f(A) = \sum_{u \in [d]} w_u \sqrt{g(A)}$  where,

$$g(A) = \left(\sum_{x \in A} x_u\right) \left(|D| + \sum_{y \in D, y \notin A} y_u\right) + h(A) \tag{3}$$

In order to show that f(A) is monotone submodular, we can show that g(A) is monotone submodular function.

$$g(A \cup \{p\}) = \left(\sum_{x \in A} x_u + p_u\right) \left(|D| + \sum_{y \in D, y \notin A} y_u - p_u\right) + h(A \cup \{p\})$$

The proof will remain same for either  $p \in A_I$  or  $p \in A_T$ . So let us consider the case  $p \in A_I$ .

$$h(A \cup \{p\}) - h(A) = p_u \sum_{u \in A_T} y_u$$

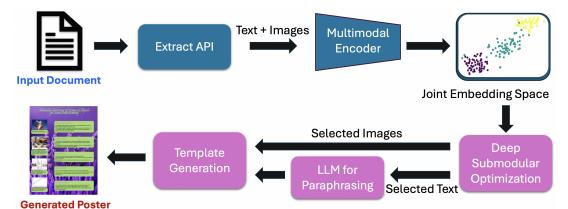


Figure 1: Block Diagram of PostDoc

$$g(A \cup \{p\}) - g(A) = p_u \left( |D| + \sum_{y \in D} y_u + \sum_{y \in A_T} y_u - 2 \sum_{x \in A} x_u \right)$$
(4)

For simplicity, lets denote  $g(A \cup \{p\}) - g(A)$  as  $g_A(p)$  Since

$$|D| > \sum_{x \in A} x_u$$
 and  $\sum_{y \in D} y_u > \sum_{x \in A} x_u$ 

$$g_A(p) = p_u \left( (|D| - \sum_{x \in A} x_u) + (\sum_{y \in D} y_u - \sum_{x \in A} x_u) + \sum_{y \in A_T} y_u \right) > 0$$

Hence g(A) is monotone. For submodularity, the property of diminishing marginal gains should hold. Let  $A \subset B$ . We need to show that  $g_A(p) \geq g_B(p)$ . Since the term  $p_u\left(|D| + \sum_{y \in D} y_u\right)$  in the equation 4 is independent of A, showing  $g_A(p) \geq g_B(p)$  dissolves to showing

$$\sum_{y \in A_T} y_u - 2\sum_{x \in A} x_u \ge \sum_{y \in B_T} y_u - 2\sum_{x \in B} x_u \tag{5}$$

since,  $\sum_{y \in A_T} y_u - 2 \sum_{x \in A} x_u = -\left(\sum_{y \in A_I} y_u + \sum_{x \in A} x_u\right)$ . As A increases the value becomes more negative, hence the inequality 5 is always valid. This proves the submodular property of g(A). Hence g(A) is monotone-submodular. Since we know that for any monotone submodular function p(A), q(A) = r(p(A)) is monotone submodular, if r is any non decreasing concave function [18], We can say that the set function f(A) is a monotone submodular function.

Please note that the submodular function f in Equation 1 proposed by us for multimodal summarization is very different from the existing set of submodular functions used for the purpose of text summarization. Most of those existing functions capture diversity rewards assuming the availability of cluster of text and ignore the multimodal aspect of the problem [18, 22]. To maximize f(A), we use the simple greedy algorithm which is a  $(1-\frac{1}{e})$  factor approximation of the optimal solution since f is monotone and submodular [18]. At each step of the greedy algorithm, we include a new content element  $x \in D \setminus A$  into A for which  $f(A \cup \{x\}) - f(A)$  is maximum till |A| = K.

### 3.2.2 Minimization w.r.t. w

Next, we assume A to be fixed and use a projected stochastic gradient descent approach to minimize the loss function in Equation 2 w.r.t  $w \geq 0$ . The subgradient of the loss w.r.t.  $w_u$  (uth dimension) on the i-th sample of the training set is calculated as  $\frac{\partial f(A)}{\partial w_u} - \frac{\partial f(A_i^*)}{\partial w_u} + \lambda w_u$ ,

where  $A=\max_{\substack{A\subseteq D_i\\|A|\leq K}}\{f(A)\}$ . This gives us the stochastic gradient descent step with a learning rate  $\alpha>0$  as (check proof of Theorem 3.1):

$$w_u = w_u - \alpha \left( \sqrt{\left( \sum_{x \in A} x_u \right) \left( |D| + \sum_{y \in D, y \notin A} y_u \right) + h(A)} \right)$$
$$- \sqrt{\left( \sum_{x \in A_i^*} x_u \right) \left( |D| + \sum_{y \in D, y \notin A_i^*} y_u \right) + h(A_i^*) + \lambda w_u \right)}$$

To ensure the non-negativity, we project it to the positive quadrant by setting  $w_u = \max(0, w_u), \forall u = 1, 2, \dots, d$ .

We iteratively solve the optimization problem in Equation 2 by maximizing it w.r.t. A by keeping w fixed, and then minimizing it w.r.t. w by keeping A fixed. We repeat these two steps until the loss converges on the validation set used in the experiments.

To calculate the value of f(A), we use a weighted sum of d terms and each term can be calculated using pre computing and using the previous term. Since we are using the greedy algorithm to find  $\max_{A\subseteq D_i} f(A)$ , this total step has time complexity of O(KNd). For the update of the weights step, we do d updates (all weights get updated) and for each update, all the summations can be done independently and then multiplied so we get the time complexity for each update as O(K) (|D| and total sum ( $\sum_{y\in D} y_u$ ) values are pre computed hence the only term calculated during run time is  $\sum_{x\in A} x_u$  term, which is then used to calculate  $\sum_{y\in D, y\notin A} y_u = \sum_{y\in D} y_u - \sum_{x\in A} x_u$ ), this gives us total training time complexity as O(NKd+Kd) for one full training update. For the inference time, we only need to use greedy algorithm to get the best possible subset (summary), hence this is will be of time complexity O(KNd)

### 3.3 Content Paraphrasing

Text sentences in the multimodal extractive summary may not be suitable to put in the poster directly. We choose ChatGPT (GPT-3.5-turbo) as it is shown to perform very well for text paraphrasing for different use cases [6]. But, applying it directly to the long text of the whole document is not possible due to the length of the document. The approaches that deal with the context length has their limitations as discussed in Section 1. However, the length of the extractive summary is limited to K. We choose K in such a way that the whole text from the extracted multimodal summary can be fed to ChatGPT within a single API call. A sample prompt for paraphrasing the text in the poster is:

"Group and rephrase the content of the following text into 5 to 8 topics without altering the order such that for each topic, there is a title and atleast 3 rephrased sentences as bullet points so that it will look good in a poster. Do not add any new content.

Text: {Extractive Summary Text}"

We use the output paraphrased text along with the images selected in the multimodal extracted summary as the content to be put in the poster.

# 3.4 Template Generation

We define the template to be a combination of different style elements such as font of the text and different colors to be used, and the layout of the poster (position of different content elements). We discuss each of them as follows.

#### 3.4.1 Font Selection

Fonts are crucial components of posters as they signal the intent of the content provided and enable mental maps for familiarity. For example, cooking books are often associated with serif and cursive styles. In this regard, we train a model based on the poster title guiding the visual attributes of the font chosen. The dataset for this task was the *Let me choose* dataset [30], which consists of pairs of titles and the most appropriate fonts for them, from a sample size of 10 fonts. We use a fine-tuned MiniLM [33] transformer model as the base encoder to the font selection model. The 384-dimensional feature vector is passed through a 2-layer fully connected network with a dropout layer and uses the LeakyReLU activation function. To find the appropriate learning rate for this process, we use LR-Finder <sup>4</sup> and trained the model for 20,000 epochs on the *train* section of the dataset.



Figure 2: A sample poster generated by PostDoc for a research paper

# 3.4.2 Color Selection

Colors are also important for posters as they capture attention and contextualise the content. Our pipeline has three main colors: (1) Background color of the poster; (2) Box fill color that serves as the color for the bounding boxes that house textual content; and (3) Text fill color - the font color of the texts. To generate the colors used in the poster, we use a model trained on the TPN architecture as proposed in Text2Colors [1]. This model, called *TPNSmall*, is trained

for 7,000 epochs on the *Text2Colors* dataset. For a given poster title, we first pass it through a QA (Question Answering) model with the prompt "What is the main point here?". This gives the intent of the poster. TPNSmall outputs a palette of hex codes, given the intent. We make use of a publicly available fine-tuned Question Answering model <sup>5</sup>. The dominant color in the palette is chosen to be the background color of the poster, and its complement is chosen to be the box fill color. The text text fill color is chosen to be black or white based on its contrast with the box fill color. We then use the background color in a prompt to Firefly <sup>6</sup> that generates a background grounded on it.

#### 3.4.3 Layout Generation

We propose a heuristic based approach for generating a balanced layout conditioned on the paraphrased content. For each topic with the associated bullet points from the paraphrased content, we create a text box in the poster layout. Similarly, for each image (with the associated caption when available), we create an image box. We kept the images on the left and text boxes on the right by dividing the space into two parts vertically. The width of the text boxes is fixed where as that of the images is adjusted based on number of images. To estimate the height of the text boxes, we consider the content length. The detailed calculations are provided in the supplementary material. By following this approach, we achieve a well-organized and visually appealing layout for posters, adapting the design based on the number of text and image boxes required. A sample poster is shown in Figure 2.

# 4 Experimental Analysis

In this section, we discuss the details about the experimental setup and results obtained from both automated and human evaluation to understand the quality of the posters generated from PostDoc.

#### 4.1 Datasets Used

We use the MSMO Dataset collected by Zhu et al. [40] for training and testing our method for multimodal summarization. The MSMO Dataset has 312,581 samples of which only the test set (10,261 samples) has image annotations present as part of the multimodal summary. We require image annotations for training our deep submodular function. So, we use 9000 samples from the test set for our training, 261 samples for validation and the remaining 1000 samples for testing. We report the performance of our multimodal summarization method on these 1000 samples in Table 1.

In order to test the content generated by our multimodal summarization module to that with the actual posters, we make use of the NJU Fudan Dataset [27]. It contains 85 pairs of scientific papers and their corresponding posters. We extract the text and images from the papers and posters using Adobe Extract. We filter paper-poster pairs so that they each have at least one image after being processed by Extract. After this step, we have a filtered dataset of 76 paper-poster pairs. We do not use any portion of this dataset for training. We use it only for testing our summarization method on out of domain data to analyze the generalization ability in Table 2.

For training the font selection model, we make use of the Let Me Choose Dataset [30] as discussed in Section 3.4. It contains 1309 short texts which are mapped to one of 10 fonts. We follow the same

<sup>&</sup>lt;sup>4</sup> https://github.com/davidtvs/pytorch-lr-finder

<sup>&</sup>lt;sup>5</sup> https://huggingface.co/deepset/roberta-base-squad2

<sup>6</sup> https://firefly.adobe.com/

Table 1: Comparison of various multimodal summarization methods on MSMO dataset [40]

Method	ROUGE-L	ROUGE-1	ROUGE-2	Coverage	Diversity	Image Precision	Inference time (sec)
MemSum + BLIP	$0.24 \pm 0.11$	$0.37 \pm 0.12$	$0.15 \pm 0.11$	$0.29 \pm 0.06$	$0.31 \pm 0.11$	$0.73 \pm 0.33$	3.27
BRIO + BLIP	$0.36 \pm 0.11$	$0.43 \pm 0.11$	$0.18 \pm 0.10$	$0.37 \pm 0.06$	$0.45 \pm 0.20$	$0.75 \pm 0.32$	1.04
GPT-3.5 + BLIP	$0.27 \pm 0.08$	$0.34 \pm 0.08$	$0.11 \pm 0.06$	$\textbf{0.38} \pm \textbf{0.06}$	$0.54 \pm 0.19$	$\textbf{0.75} \pm \textbf{0.32}$	14.88
PostDoc	$\textbf{0.68} \pm \textbf{0.14}$	$\textbf{0.70} \pm \textbf{0.10}$	$\textbf{0.36} \pm \textbf{0.13}$	$0.30\pm0.03$	$\textbf{0.58} \pm \textbf{0.06}$	$0.74 \pm 0.34$	0.68

Table 2: Comparison of various multimodal summarization methods on NJU-Fudan dataset [27]

	Table 2. Companison of various maramoun summarization metalogs on the Tablah dataset [27]							
Method	ROUGE-L	ROUGE-1	ROUGE-2	Coverage	Diversity	Image Precision	Inference time (sec)	
MemSum + BLIP	$\begin{array}{c c} 0.27 \pm 0.03 \\ 0.07 \pm 0.02 \end{array}$	$0.27 \pm 0.03$	$0.17 \pm 0.03$	$0.38 \pm 0.05$	$0.64 \pm 0.05$	$0.39 \pm 0.28$	10.35	
BRIO + BLIP		$0.07 \pm 0.02$	$0.03 \pm 0.01$	$0.27 \pm 0.06$	$0.66 \pm 0.06$	$0.38 \pm 0.26$	6.09	
GPT-3.5 + BLIP	$0.13 \pm 0.05$	$0.13 \pm 0.05$	$0.06 \pm 0.04$	$0.31 \pm 0.06$	$0.65 \pm 0.05$	$0.39 \pm 0.28$	47.37	
PostDoc	$0.50 \pm 0.04$	$0.48 \pm 0.03$	$0.33 \pm 0.03$	$0.42 \pm 0.03$	$0.61 \pm 0.04$	$0.53 \pm 0.32$	<b>4.25</b>	

split mentioned by the authors for training (70%),validation (10%) and testing (20%).

#### 4.2 Baseline Methods

We could not find any replicable source code for existing document-to-poster works [27, 34, 35] to make an end-to-end comparison. So, for a thorough evaluation, we analyze each module of PostDoc with the corresponding baselines.

**Multimodal Summarization** To the best of our understanding, there is not publicly available replicable implementation for any of the existing multimodal-in and multimodal-out summarization approaches [40, 41, 37, 38]. Additionally, as we are using a subset of the MSMO test set (§4.1) for training, we will not be able to carry over the metrics reported in these works. So, we compare our performance with baselines for text summarization and include images in the summary as a post-processing step. For the text summarization, we use the following.

- 1. Extractive Summarization: We use the publicly available Mem-Sum architecture [8] which currently has the SOTA performance on the GovReport dataset [13].
- 2. **Abstractive Summarization**: We use the publicly available **BRIO** [20] architecture, which currently has the SOTA performance on the CNN/Daily Mail dataset [23]
- 3. **GPT-3.5-turbo**: In this baseline, we chunk the input text and summarize each chunk with GPT-3.5-turbo. We concatenate the summaries and finally paraphrase it further with another GPT-3.5-turbo

With each of the above text summaries, the images are included in the summary by choosing the top  $K_I$  images from the input document based on their similarity with the summarized text. Here the similarity is calculated by the cosine of the BLIP embeddings [17] of text and images. From the training set, we calculate the average ratio of the number of images in the summary and that in the input document. To fix  $K_I$  during inference on a document, we multiply the number of images present in the document with that ratio.

**Template Generation** For font selection, we compare our method (§3.4.3) with the Let Me Choose BERT Model [30] on the Let Me Choose Dataset as shown in Table 4. For layout generation, we compare our method (§3.4.3) with LayoutDM [14] on the NJU Fudan Dataset, in Table 4.

# 4.3 Evaluation Metrics

We evaluate the generated posters on the following aspects.

#### 4.3.1 Multimodal Summarization

To evaluate the salience of the text generated by our method, we employ the standard text summarization metric ROUGE-1, ROUGE-2 and ROUGE-L which compares the generated summary with the ground truth. To evaluate the generated multimodal summary, we use coverage and diversity [15] along with image precision and image recall. Coverage is measured between the generated multimodal summary and the multimodal source document.

$$Coverage(A) = \frac{1}{|D||A|} \sum_{x \in D} \sum_{y \in A} cosine(x, y)$$

$$Diversity(A) = 1 - \frac{1}{|A|^2} \sum_{x,y \in A} cosine(x,y)$$

Here, D contains the BLIP embeddings for all the content elements of the input multimodal document and A contains the BLIP embeddings from the generated summary. An ideal summary will have high coverage and high diversity. Following Zhu et al. [40], we use image precision  $I_P$  to evaluate the images selected by our method.  $I_P = \frac{|I_A \cap I_G|}{|I_A|}$ , where  $I_A$  and  $I_G$  refer to the images in generated summary and the images in the ground truth summary. We also report average inference time as a metric for computational overhead.

### 4.3.2 Template Generation

Font Selection: Following Shirani et al. [30], we measure the performance of font selection models using the average weighted F1-Score over top k where  $k = \{1, 3\}$ .

**Layout Generation**: To evaluate the layouts generated, we use a combination of the following NGOMetrics <sup>7</sup>: We use a weighted combination of the equilibrium of the bounding boxes, padding in the layout, density of the bounding boxes with respect to the overall layout, and overlap between the bounding boxes. This allows us to score layouts based on their aesthetic properties. Please refer to the details about the computation of these metrics in the supplementary material.

## 4.4 Performance Analysis

**Multimodal Summarization** Tables 1 and 2 show the mean and standard deviation of the performance of PostDoc and the baselines on MSMO and NJU Fudan datasets respectively. By investigating the results, we note the following: (1): PostDoc outperforms all of the baselines on the basis of all the ROUGE metrics with significant margins. This shows that the our multimodal summarization module captures relevant information from the source document which aligns

<sup>&</sup>lt;sup>7</sup> http://www.mi.sanu.ac.rs/vismath/ngo/index.html

Table 3: Model ablation study of PostDoc on MSMO and NJU-Fudan Datasets

		MSMO Datset				NJU-Fudan Dataset			
Method	ROUGE-L	Coverage	Diversity	Image Precision	ROUGE-L	Coverage	Diversity	Image Precision	
PostDoc w/o dsf	$0.57 \pm 0.13$	$0.23 \pm 0.04$	$0.61 \pm 0.09$	$0.71 \pm 0.44$	$0.43 \pm 0.10$	$0.34 \pm 0.07$	$0.65 \pm 0.07$	$0.37 \pm 0.17$	
PostDoc w/o coverage	$0.51 \pm 0.08$	$0.24 \pm 0.06$	$0.61 \pm 0.14$	$\textbf{0.75} \pm \textbf{0.27}$	$0.48 \pm 0.05$	$\textbf{0.43} \pm \textbf{0.04}$	$0.57 \pm 0.06$	$0.52 \pm 0.32$	
PostDoc w/o diversity	$0.50 \pm 0.13$	$0.24 \pm 0.03$	$0.62 \pm 0.14$	$0.75 \pm 0.27$	$0.48 \pm 0.05$	$0.43 \pm 0.04$	$0.58 \pm 0.06$	$0.53 \pm 0.32$	
PostDoc w/o alignment	$0.56 \pm 0.12$	$0.25 \pm 0.06$	$\textbf{0.63} \pm \textbf{0.13}$	$0.75 \pm 0.28$	$0.45 \pm 0.04$	$0.36 \pm 0.03$	$\textbf{0.67} \pm \textbf{0.03}$	$0.34 \pm 0.19$	
PostDoc	$0.68 \pm 0.14$	$\textbf{0.30} \pm \textbf{0.03}$	$0.58\pm0.05$	$0.74 \pm 0.34$	$0.50\pm0.04$	$0.42\pm0.03$	$0.61\pm0.04$	$\textbf{0.53} \pm \textbf{0.32}$	

**Table 4**: Results of font recommendation on Let Me Choose Dataset and conditional layout generation on the NJU-Fudan Dataset

Font R	ecommendat	Layout Generation		
Method	Top-1 F1	Top-3 F1	Method	NGOMetric
BERT Model PostDoc	0.2697 <b>0.4301</b>	0.5191 <b>0.5950</b>	LayoutDM PostDoc	0.27 <b>0.46</b>

with the ground truth. (2): Regarding the visual modality metric image precision, our model performs on par or worse than the baselines on MSMO dataset, but significantly outperforms all the baselines on NJU-Fudan dataset. It is important to note that the number of images selected in the baseline is pre-fixed based on the average number of images present in the document on the training set. However, for PostDoc, we do not differentiate between the text and images in the selection criteria during the inference. This gives a benefit to the baselines on the MSMO dataset for image selection. (3): As mentioned in Section 4.2, we consider extractive summarization (Mem-Sum), abstractive summzarization (BRIO) and GPT-3.5-turbo for the text summarization module of the baselines. MemSum and BRIO are trained on 17517 and 200000 samples respectively and GPT-3.5-turbo is trained on large text databases available on the internet. Our model is able to perform competitively with these baselines even though the data it is trained on (9000 samples) is significantly lesser than the training data of these text summarization methods. (4): The GPT-3.5-turbo+BLIP baseline performs competitively on coverage and diversity. But the latency and the cost associated with GPT calls is very high. On average, the number of tokens processed by GPT-3.5 for MSMO dataset is 838.19 and the NJU Fudan dataset is 5323.85. PostDoc is much faster compared to all the baselines in terms of average inference time on a document.

**Template Generation** Table 4 shows the performance of our model and the baseline on the Let Me Choose dataset for font selection. Our model outperforms the baseline on both the Top-1 F1 Score and the Top-3 F1 Score. Table 4 also compares our layout generation module with LayoutDM. As LayoutDM wasn't explicitly trained on posters, we initially generate 250 candidate layouts using LayoutDM and choose the layout which gives the highest equiweighted NGOMetrics Score. Our layout generation module, which relies on heuristics, achieves a score that is almost twice the score achieved by LayoutDM.

# 4.5 Model Ablation Study

To understand the importance of different components of PostDoc, we perform the following ablation experiments on MSMO and NJU Fudan datasets, and report the results in Table 3.

PostDoc w/o DSF: Instead of using the deep submodular function proposed in Equation 1, we make use of a simple feature-based submodular function (without any trainable parameters) which takes coverage and diversity into account:  $f(A) = \lambda \sum_{x \in A} \sum_{y \in D} x_u y_u - \sum_{x \in A} \sum_{y \in A} x_u y_u$ . For inference we choose the subset A which gives the maximum value of f(A). The value of  $\lambda$  was set as 0.2 for this experiment.

PostDoc w/o Coverage: We remove the first term  $\sum_{x \in A} \sum_{y \in D} x_u y_u$  from Equation 1 and proceed with the min-max optimization procedure mentioned in Section 3.2.

PostDoc w/o Diversity: We remove the second term  $\sum_{x \in A} \sum_{y \in A} x_u y_u$  from Equation 1 and proceed with the min-max optimization procedure mentioned in Section 3.2.

*PostDoc w/o Alignment*: We remove the third term  $\sum_{x \in A_I} \sum_{y \in A_T} x_u y_u$  from Equation 1 and proceed with the min-max optimization procedure mentioned in Section 3.2.

From Table 3, we can see that PostDoc is able to achieve the best performance on Rouge metrics which shows the importance of the combination of all the components present in it. On the other three metrics, we do not see any consistent pattern among the different model variants.

Questions	GPT-3.5 + BLIP	PostDoc
Coverage	$3.13 \pm 0.83$	$\textbf{3.40} \pm \textbf{0.64}$
Duplication	$\textbf{4.26} \pm \textbf{0.27}$	$4.13 \pm 0.37$
Content Ordering	$3.33 \pm 0.62$	$3.33 \pm 0.47$
Image Selection	$2.66 \pm 0.47$	$\textbf{3.0} \pm \textbf{0.70}$
Template	$2.33 \pm 0.5$	$\textbf{3.46} \pm \textbf{0.18}$
Run Time	$1.86 \pm 0.29$	$\textbf{3.60} \pm \textbf{0.64}$

Table 5: Human evaluation on a subset of NJU-Fudan Dataset

## 4.6 Human Evaluation

We have conducted a small scale human survey to understand the quality of the generated posters from the actual human perspective. We hired 3 experts in AI as reviewers for this task. We randomly chose 5 research papers from NJU Fudan dataset. We used GPT-3.5+BLIP as the only baseline for this study as GPT-3.5 is often considered to be close to humans in generative tasks. Each reviewer generated the posters by the two algorithms from each of the selected 5 documents and gave a rating on a scale of 1 (worst) to 5 (best) to each poster for each of the following questions: (1) How good is the poster to *cover* the document? (2) Is there any *duplication* of content in the poster? (3) How good is the *ordering of content* in the poster? (4) How good are the *selected images* in the poster? (5) How good is the *template* of the poster? (6) Are you satisfied with the *run time* of the algorithm?

In Table 5, we compute average and standard deviation of the ratings provided by all the reviewers on all the documents. Interestingly, the human evaluation results correlate well with the automated evaluation on NJU Fudan dataset in Table 2. The human evaluation shows that PostDoc is as per or better in terms of output content quality and much better in terms of user satisfaction with the layout and runtime.

#### 5 Discussion and Conclusion

We have presented PostDoc, an end-end pipeline to automatically generate a poster from a long multimodal document. It is interesting to find that PostDoc, by using a novel combination of deep submodular functions and a single call to LLM (ChatGPT) is able to achieve

better or comparable performance than direct calls to the same LLM which is expensive both in terms of computation and cost. In the current work, the performance of PostDoc is limited for non-natural images such as flow-chart and neural diagrams, and other structured elements like tables, etc. We plan to fine-tune VLMs on documents containing such elements as a future work.

#### References

- [1] H. Bahng, S. Yoo, W. Cho, D. Park, Z. Wu, X. Ma, and J. Choo. Coloring with Words: Guiding Image Colorization Through Text-Based Palette Generation: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part XII, pages 443–459. 09 2018. ISBN 978-3-030-01257-1. doi: 10.1007/978-3-030-01258-8\_27.
- [2] A. Bhaskar, A. Fabbri, and G. Durrett. Prompted opinion summarization with gpt-3.5. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9282–9300, 2023.
- [3] J. Bilmes and W. Bai. Deep submodular functions, 2017.
- [4] S. Chai, L. Zhuang, and F. Yan. Layoutdm: Transformer-based diffusion model for layout generation, 2023.
- [5] S. Chen, S. Wong, L. Chen, and Y. Tian. Extending context window of large language models via positional interpolation. arXiv preprint arXiv:2306.15595, 2023.
- [6] X. Chen, J. Ye, C. Zu, N. Xu, R. Zheng, M. Peng, J. Zhou, T. Gui, Q. Zhang, and X. Huang. How robust is gpt-3.5 to predecessors? a comprehensive study on language understanding tasks. arXiv preprint arXiv:2303.00293, 2023.
- [7] T.-J. Fu, W. Y. Wang, D. McDuff, and Y. Song. Doc2ppt: automatic presentation slides generation from scientific documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 634–642, 2022.
- [8] N. Gu, E. Ash, and R. H. Hahnloser. Memsum: Extractive summarization of long documents using multi-step episodic markov decision processes. arXiv preprint arXiv:2107.08929, 2021.
- [9] K. Gupta, J. Lazarow, A. Achille, L. S. Davis, V. Mahadevan, and A. Shrivastava. Layouttransformer: Layout generation and completion with self-attention. In *Proceedings of the IEEE/CVF International Con*ference on Computer Vision (ICCV), pages 1004–1014, October 2021.
- [10] B. He, J. Wang, J. Qiu, T. Bui, A. Shrivastava, and Z. Wang. Align and attend: Multimodal summarization with dual contrastive losses. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 14867–14878, June 2023.
- [11] H. Hsu, X. He, Y. Peng, H. Kong, and Q. Zhang. Posterlayout: A new benchmark and approach for content-aware visual-textual presentation layout, 2023.
- [12] H. Y. Hsu, X. He, Y. Peng, H. Kong, and Q. Zhang. Posterlayout: A new benchmark and approach for content-aware visual-textual presentation layout. In *Proceedings of the IEEE/CVF Conference on Computer Vi*sion and Pattern Recognition (CVPR), pages 6018–6026, June 2023.
- [13] L. Huang, S. Cao, N. Parulian, H. Ji, and L. Wang. Efficient attentions for long document summarization. arXiv preprint arXiv:2104.02112, 2021.
- [14] N. Inoue, K. Kikuchi, E. Simo-Serra, M. Otani, and K. Yamaguchi. LayoutDM: Discrete Diffusion Model for Controllable Layout Generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 10167–10176, 2023.
- [15] S. Kothawade, J. Girdhar, C. Lavania, and R. Iyer. Deep sub-modular networks for extractive data summarization. arXiv preprint arXiv:2010.08593, 2020.
- [16] H. Li, J. Zhu, C. Ma, J. Zhang, and C. Zong. Multi-modal summarization for asynchronous collection of text, image, audio and video. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 1092–1102, 2017.
- [17] J. Li, D. Li, C. Xiong, and S. Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.
- [18] H. Lin and J. Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 510–520, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1052.
- [19] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang. Lost in the middle: How language models use long contexts. arXiv preprint arXiv:2307.03172, 2023.

- [20] Y. Liu, P. Liu, D. Radev, and G. Neubig. Brio: Bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903, 2022.
- [21] I. Mademlis, A. Tefas, N. Nikolaidis, and I. Pitas. Multimodal stereoscopic movie summarization conforming to narrative characteristics. *IEEE Transactions on Image Processing*, 25(12):5828–5840, 2016.
- [22] N. Modani, P. Maneriker, G. Hiranandani, A. R. Sinha, Utpal, V. Subramanian, and S. Gupta. Summarizing multimedia content. In Web Information Systems Engineering—WISE 2016: 17th International Conference, Shanghai, China, November 8-10, 2016, Proceedings, Part II 17, pages 340–348. Springer, 2016.
- [23] R. Nallapati, B. Zhou, C. dos Santos, C. Gulcehre, and B. Xiang. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/K16-1028. URL https://aclanthology.org/K16-1028.
- [24] OpenAI. Gpt-4 technical report, 2023.
- [25] B. Peng, J. Quesnelle, H. Fan, and E. Shippole. Yarn: Efficient context window extension of large language models. arXiv preprint arXiv:2309.00071, 2023.
- [26] F. Petroni, P. Lewis, A. Piktus, T. Rocktäschel, Y. Wu, A. H. Miller, and S. Riedel. How context affects language models' factual predictions. In *Automated Knowledge Base Construction*, 2020. URL https://openreview.net/forum?id=025X0zPfn.
- [27] Y.-T. Qiang, Y.-W. Fu, X. Yu, Y.-W. Guo, Z.-H. Zhou, and L. Sigal. Learning to generate posters of scientific papers by probabilistic graphical models. *Journal of Computer Science and Technology*, 34:155–169, 2019.
- [28] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International con*ference on machine learning, pages 8748–8763. PMLR, 2021.
- [29] D. C. Shelledy. How to make an effective poster. Respiratory Care, 49 (10):1213–1216, 2004.
- [30] A. Shirani, F. Dernoncourt, J. Echevarria, P. Asente, N. Lipka, and T. Solorio. Let me choose: From verbal context to font selection. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020.
- [31] E. Sun, Y. Hou, D. Wang, Y. Zhang, and N. X. Wang. D2s: Document-to-slide generation via query-based text summarization. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1405–1418, 2021.
- [32] S. Tschiatschek, R. K. Iyer, H. Wei, and J. A. Bilmes. Learning mixtures of submodular functions for image collection summarization. *Advances in neural information processing systems*, 27, 2014.
- [33] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pretrained transformers. *CoRR*, abs/2002.10957, 2020. URL https://arxiv. org/abs/2002.10957.
- [34] S. Xu and X. Wan. Neural content extraction for poster generation of scientific papers. arXiv preprint arXiv:2112.08550, 2021.
- [35] S. Xu and X. Wan. Posterbot: A system for generating posters of scientific papers with neural models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 13233–13235, 2022.
- [36] J. Zhang, Y. Zhao, M. Saleh, and P. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020.
   [37] L. Zhang, X. Zhang, J. Pan, and F. Huang. Hierarchical cross-modality
- [37] L. Zhang, X. Zhang, J. Pan, and F. Huang. Hierarchical cross-modality semantic correlation learning model for multimodal summarization, 2021.
- [38] Z. Zhang, X. Meng, Y. Wang, X. Jiang, Q. Liu, and Z. Yang. Unims: A unified framework for multimodal summarization with knowledge distillation, 2022.
- [39] M. Zhong, P. Liu, Y. Chen, D. Wang, X. Qiu, and X.-J. Huang. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, 2020.
- [40] J. Zhu, H. Li, T. Liu, Y. Zhou, J. Zhang, and C. Zong. MSMO: Multimodal summarization with multimodal output. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4154–4164, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1448. URL https://aclanthology.org/D18-1448.
- [41] J. Zhu, Y. Zhou, J. Zhang, H. Li, C. Zong, and C. Li. Multimodal summarization with guidance of multimodal reference. *Proceedings*

of the AAAI Conference on Artificial Intelligence, 34(05):9749–9756, Apr. 2020. doi: 10.1609/aaai.v34i05.6525. URL https://ojs.aaai.org/index.php/AAAI/article/view/6525.

# A Template Generation for PostDoc

In this section, we provide the details of the color selection and layout generation modules which we had to skip in the main paper for the limited space.

### A.1 Color Selection

Colors are also important for posters as they capture attention and contextualise the content <sup>8</sup>. Our pipeline has three main colors:

- Background color that underpins the poster
- Box fill color that serves as the color for the bounding boxes that house textual content
- Text fill color the font color of the texts themselves.

To select the color palette for the poster, we used the Text2Colors' TPN model [1] to generate a candidate palette.

From this, we used the relative CLAB ratios to find the most contrastive color within the palette and colored the background of the poster with this color. We then colored the text bounding boxes with its complement.

The *dominant color* in the palette is chosen to be the background color of the poster, and its *complement* is chosen to be the box fill color. The text text fill color is chosen to be black or white based on its contrast with the box fill color. We then use the background color in a prompt to Firefly <sup>9</sup> that generates a background grounded on it.

For this, we make use of *thecolorapi.com* to get a textual name for the background/dominant color.

We now describe the equation that governs the selection of the dominant color. Given two colors C and C', the relative luminance of C w.r.t C' is

$$RL(C,C') = \frac{max(L(C),L(C')+0.05}{min(L(C),L(C')+0.05}$$

where L(C) is the luminance (perceived brightness) of the RGB color C as defined in the WCAG standard  $^{\rm 10}$ 

Given a set of colors  $\mathbb{C}$ , dominant color  $C_d$  is calculated as follows:

$$C_d = \max_{C \in \mathbb{C}} \sum_{\substack{C' \in \mathbb{C} \\ C' \neq C}} RL(C, C')$$

## A.2 Layout Generation

We propose a heuristic based approach for generating a balanced layout conditioned on the paraphrased content. For each topic with the associated bullet points from the paraphrased content, we create a text box in the poster layout. Similarly, for each image (with the associated caption when available), we create an image box. Let us denote  $N_T$  and  $N_I$  as the required number of text and image boxes in the layout respectively. We begin by fixing the bounding box for the title, leaving the remaining space (l) for distributing image and text boxes. We kept the images on the left and text boxes on the right by dividing the space into two parts vertically (b1 and b2). The width of the text boxes is fixed (say  $b2 - \alpha$ ) where as that of the images is adjusted based on number of images (say  $b1-\beta/N_I$ ). Here,  $\alpha$  and  $\beta$  are to ensure some marginal gaps between the boxes and the page margin. The width of the captions box is same as that of images. To estimate the height of the text boxes, we consider the content length and the  $N_T$ and  $N_I$  for captions. For images,  $height = width/aspect\_ratio$ . The distance between the text boxes  $(dh_1)$  and between the image boxes  $(dh_2)$  is calculated depending on the values of  $N_T$  and  $N_I$ , respectively. Refer to the figure 3 for better understanding.

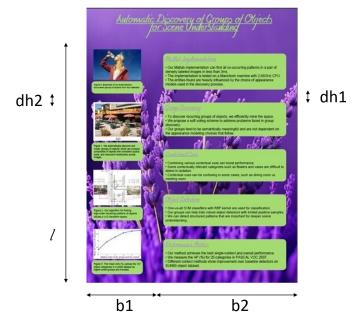


Figure 3: A sample layout generated by this method ( $N_I = 4$ ,  $N_T = 5$ ) However, when the number of images is less than three, we elongate the bottom text boxes towards the image (left) side and reduce  $dh_2$  to ensure visual balance.  $dh_1$  and  $dh_2$  are calculated as shown below:

$$\begin{split} dh_1 &= \frac{l - \sum_i h_T^i}{N_T + 1} \\ if \ N_I > 3, \ dh_2 &= \frac{l - \sum_i \left( h_I^i + h_C^i \right) - k_1 \cdot N_I}{N_I + 1} \\ else \ if \ N_I <= 2, \ dh_2 = dh_1 \end{split}$$

Once we get the heights and widths of the bounding boxes, we can use them to calculate the top left vertex of the bounding box. Their Y coordinates of these points are calculated as shown:

$$Y_T^i = l - (i \cdot dh_1) - \sum_{j=1}^{i-1} h_T^j$$

$$Y_I^i = l - (i \cdot dh_2) - \sum_{j=1}^{i-1} (h_I^j + h_C^j) - (i-1) \cdot k_2$$

$$Y_C^i = Y_I^i - h_I^i - k_2$$

<sup>&</sup>lt;sup>8</sup> https://www.snap.com.au/blog/colours-and-advertising-posters

<sup>9</sup> https://firefly.adobe.com/

<sup>10</sup> https://www.w3.org/WAI/GL/wiki/Relative\_luminance

Method	ROUGE-L	ROUGE-1	ROUGE-2	Coverage	Diversity	Image Precision
PostDoc with submodular function	$0.579509 \pm 0.130993$	$0.465446 \pm 0.051633$	$0.276391 \pm 0.028582$	$0.228677 \pm 0.044838$	$0.617691 \pm 0.093911$	$0.716180 \pm 0.437901$
PostDoc optimized w/o coverage	$0.512333 \pm 0.086352$	$0.526364 \pm 0.053763$	$0.248168 \pm 0.030499$	$0.248681 \pm 0.064308$	$0.619039 \pm 0.146075$	$0.753445 \pm 0.275050$
PostDoc optimized w/o diversity	$0.507014 \pm 0.130131$	$0.521063 \pm 0.055315$	$0.245248 \pm 0.030504$	$0.248617 \pm 0.248617$	$0.617401 \pm 0.146460$	$0.752585 \pm 0.276562$
PostDoc optimized w/o alignment	$0.562540 \pm 0.129673$	$0.577946 \pm 0.047215$	$0.276391 \pm 0.033015$	$0.252465 \pm 0.064187$	$0.634624 \pm 0.135463$	$0.748600 \pm 0.282186$
PostDoc	$0.67703 \pm 0.142195$	$\textbf{0.703324} \pm \textbf{0.100135}$	$0.357304 \pm 0.135787$	$0.296999 \pm 0.032629$	$0.576494 \pm 0.059301$	$0.739570 \pm 0.341569$

Table 6: Ablation study of PostDoc on MSMO Dataset

Method	ROUGE-L	ROUGE-1	ROUGE-2	Coverage	Diversity	Image Precision
PostDoc with submodular function	$0.426815 \pm 0.103294$	$0.514749 \pm 0.044017$	$0.384185 \pm 0.044100$	$0.343835 \pm 0.065151$	$0.651721 \pm 0.078240$	$0.366457 \pm 0.174725$
PostDoc optimized w/o coverage	$0.475431 \pm 0.050257$	$0.455375 \pm 0.048136$	$0.325059 \pm 0.047058$	$\textbf{0.432243} \pm \textbf{0.046794}$	$0.577893 \pm 0.065549$	$0.529320 \pm 0.321356$
PostDoc optimized w/o diversity	$0.486038 \pm 0.052330$	$0.465534 \pm 0.050122$	$0.332959 \pm 0.047065$	$0.431614 \pm 0.046335$	$0.580118 \pm 0.065608$	$0.530137 \pm 0.320465$
PostDoc optimized w/o alignment	$0.455443 \pm 0.046483$	$0.503130 \pm 0.051569$	$0.362864 \pm 0.050940$	$0.361852 \pm 0.039532$	$0.677698 \pm 0.031189$	$0.343208 \pm 0.194419$
PostDoc	$0.503238 \pm 0.041803$	$0.482009 \pm 0.036599$	$0.331308 \pm 0.034393$	$0.421352 \pm 0.031297$	$0.609323 \pm 0.043384$	$0.531805 \pm 0.325263$

Table 7: Ablation study of PostDoc on NJU-Fudan [27] Dataset

where  $h_T^i,\,h_I^i,\,h_C^i$  are the estimated heights of the  $i^{th}$  text, image and caption box respectively.  $k_1$  is added to make  $dh_2$  depend on the number of images.  $k_2$  is the gap between the image and caption boxes. The X coordinates of these points can be linearly interpolated based on the widths of the bounding boxes, b1, and b2. Once we get these coordinates , height, and width, we can calculate the other coordinates. By following this approach, we achieve a well-organized and visually appealing layout for posters, adapting the design based on the number of text and image boxes available.

# **B** Additional Results for Model Ablation Study

Table 6 and Table 7 show the performance of model variants on all the metrics we used in the other experiments. <sup>11</sup> Our model, Post-Doc, performs best against the ablations on the MSMO test dataset in terms of the ROUGE-1 and ROUGE-2 scores as well.

## C Details of NGOMetrics

The metric to evaluate the quality of the generated layouts was built on top of design best practices, as defined by NGO-Metrics <sup>12</sup>. We use a combination of selected processed NGOMetrics.

We now formally define the NGOMetrics' scoring mechanisms and the notations for the same

- The overall frame has a width W and height H  $\Rightarrow$  The area of the whole frame is  $W \cdot H$
- Each rectangular bounding box i has an area  $a_i$ , with x-coordinates ranging from  $x_1^i$  to  $x_2^i$ , and y-coordinates ranging from  $y_1^i$  to  $y_2^i$ , so its center of mass is  $[COM_x^i, COM_y^i] = \frac{x_1^i + x_2^i}{2}, \frac{y_1^i + y_2^i}{2}$
- $\bullet$  We condition the layouts on a total of  $\mathbb B$  rectangular bounding boxes.

**Equilibrium** This represents the distance of the center of mass of the bounding boxes (when taken together) and the center of mass of the layout.

This metric is calculated as

$$equilibrium = 1 - \frac{|EM|_x + |EM|_y}{2}$$

where

$$EM_x = \frac{2 \cdot \sum_{i=1}^{i=\mathbb{B}} [a_i \cdot (COM_x^i - W/2)]}{\mathbb{B} \cdot W \cdot \sum_{i=1}^{i=\mathbb{B}} a_i}$$

and

$$EM_y = \frac{2 \cdot \sum_{i=1}^{i=\mathbb{B}} [a_i \cdot (COM_y^i - H/2)]}{\mathbb{B} \cdot H \cdot \sum_{i=1}^{i=\mathbb{B}} a_i}$$

**Padding:** This represents the area between the boundary of the layout and the topmost and rightmost edges of the bounding boxes. This metric is calculated as

$$padding = 1 - \frac{[R\_{max} - L\_{min}] \cdot [T\_{max} - B\_{min}]}{W \cdot H}$$

where

$$R\_max = max_{1 \le i \le \mathbb{B}}[x_2^i]$$

$$L\_min = min_{1 \le i \le \mathbb{B}}[x_1^i]$$

$$T\_max = max_{1 \le i \le \mathbb{B}}[y_2^i]$$

$$B\_min = min_{1 \le i \le \mathbb{B}}[y_i^i]$$

**Density**: This represents the proportion of the frame layout covered by the bounding boxes.

This metric is calculated as

$$density = max(1, \frac{\sum_{i=1}^{i=B} a_i}{W \cdot H \cdot \mathbb{B}})$$

**Overlap:** This represents the sum of areas of intersection between the bounding boxes in the layout.

This metric is calculated as

$$overlap = 1 - max(1, \frac{\sum_{i=1}^{i=\mathbb{B}-1} \sum_{j=i+1}^{j=\mathbb{B}} O_{eff}(i,j))}{W \cdot H})$$

where

$$O_{eff}(i,j) = \mathbb{1}_{\Delta x(i,j)>0} \cdot \mathbb{1}_{\Delta y(i,j)>0} \cdot \Delta x(i,j) \cdot \Delta y(i,j)$$

having

$$\Delta x(i,j) = min[x_2^i, x_2^j] - max[x_1^i, x_1^j]$$

$$\Delta y(i,j) = min[y_2^i, y_2^j] - max[y_1^i, y_1^j]$$

Please note that, due to a lack of space in the main paper, we did not include the ROUGE-1 and ROUGE-2 score in the model ablation tables of the main paper.

<sup>12</sup> http://www.mi.sanu.ac.rs/vismath/ngo/index.html

Method	Equilibrium score	Padding score	Density score	Overlap score
LayoutDM	0.450	0.282	0.092	0.266
PostDoc	0.600	0.213	0.256	0.774

Table 8: Comparison study of the conditional layout generation models on the test section of NJU-Fudan dataset

Overall scoring function The overall scoring function is as follows:  $w=0.25\cdot[equilibrium]+0.25\cdot[padding]+0.25\cdot[density]+0.25\cdot[overlap]$ 

We now present the average scores of these NGO Metrics for the test section of the NJU-Fudan dataset. The conditions for the LayoutDM model [4] was the (number of topics provided by GPT 3.5 after paraprhasing the textual content + number of images selected) text boxes + (number of images selected) image boxes.

While LayoutDM works with conditional inputs, our observations was that with increased number of text and image boxes, the overlap increases and the layouts look more cluttered, particularly near the center of the document. Our algorithm scales better with an increase in inputs in lieu of density and overlap scores.

Future work includes finetuning LayoutDM on posters and new conventions to order images and text bounding boxes. Furthermore, papers like [12] which generate layouts from an image but do not explicitly take number of bounding boxes as inputs, so we are not using it in our work.