VISUAL RECOGNITION(PART -II)
ASSIGNMENT-I

Q.I. Core concepts in CVPR 2022 paper " Grounded Language-Image Pretraining (GLIP)"

ANS: In the paper " Grounded Language-Image Pretraining", the authors introduce their recent efforts on building a generalizable localization model with language supervision . GLIP enables the unification between localization and vision-language understanding, paving a way towards a unified computer vision foundation model,which can be commonly used for various vision tasks.

GLIP,which is is designed to learn visual representations that are object-level, learns visual representations by unifying object detection and phrase grounding.Given an image and a corresponding caption, The phrase grounding task aims to ground each entity mentioned by a noun phrase in the caption to a region in the image.This leads to improvements in both tasks and enables the model to generate semantic-rich grounding boxes through self-training.

Main Contributions:

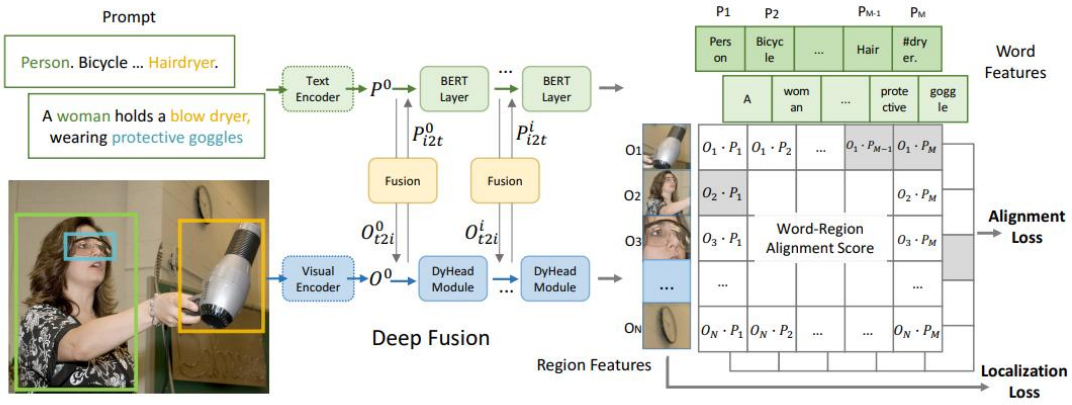
1. **Unifying detection and grounding by reformulating object detection as phrase grounding.**
2. **Scaling up visual concepts with massive image-text data**
3. **Transfer learning with GLIP: one model for all**

In the **related work** section of the paper they mentioned about **CLIP(Contrastive Language-Image Pretraining) by OpenAI.** Which is a model that combines vision and language understanding. It consists of a vision encoder and a text encoder, which are jointly trained using a contrastive loss. The vision encoder processes images, while the text encoder processes natural language descriptions. The model is trained to align the representations of matching image-text pairs and differentiate them from non-matching pairs. This enables CLIP to learn a shared embedding space where images and texts can be compared and matched. The training process involves sampling positive and negative pairs, calculating similarity scores, and optimizing the model parameters using gradient descent. This approach allows CLIP to acquire general-purpose knowledge about images and texts, enabling it to perform various image level tasks like image classification, zero-shot image generation, and text-based image retrieval.

GLIP goes through a rather complicated process in the pre-prediction stage unlike clip. One important thing to note here is that the DyHead module in GLIP architecture is borrowed from a paper published at **cvpr in 2021 called 'Dynamic Head."**It is an object detection architecture ,which dynamically adapts the head architecture based on the input image, utilizing a shared backbone network, prediction head generator, and selector. This approach aims to improve object detection accuracy by selecting the most suitable prediction head for each image.This Dyhead module does not change the shape of the feature map.So, the

input vector and output vector have the same shape and can be used by attaching several, and can be used by attaching to any detection network.

Details Of GLIP:



The figure below shows the overall structure of GLIP. As mentioned briefly in the Introduction, GLIP is a network to perform the task of integrating Object Detection and Phrase Grounding.The structure is very similar to the structure of the clip described above. However, In GLIP

- The input is entered as a noun phrase describing each object rather than a sentence that describes the entire image because the task has to be performed at the object level rather than the image level. By using the NLP Parser introduced in this book called 'NLP with python', only the part corresponding to the noun phrase in the sentence is extracted and used in this way. So, the noun phrases that describe the object in each sentence create text data in a format that is distinguished through the dot. For the 24M web-crawled image-text pairs, there are 78.1M high-confidence (> 0.5) phrase-box pseudo annotations,with 58.4M unique noun phrases.

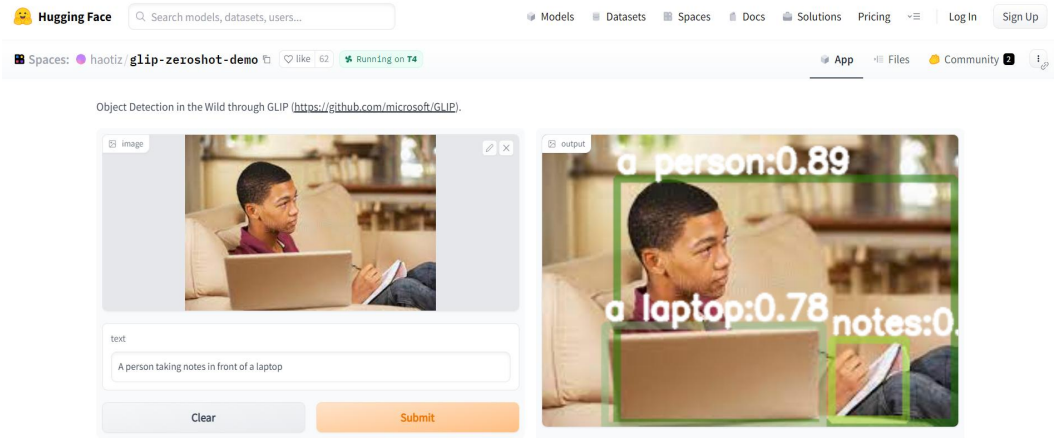
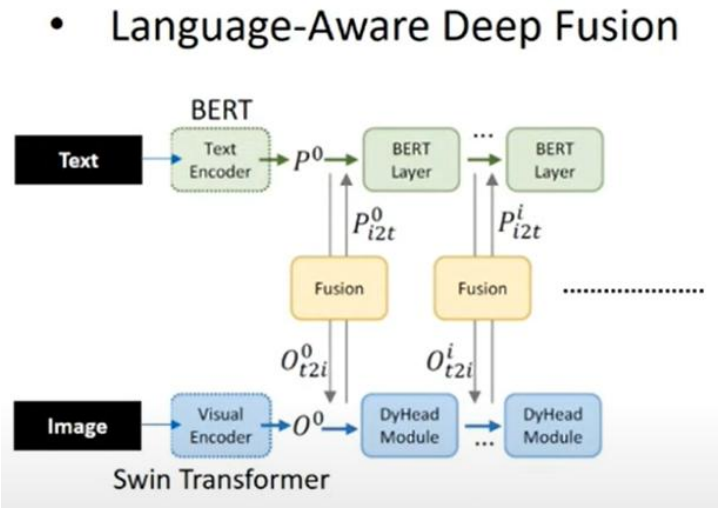


Image generated by me using hugging-face spaces deployed zero shot GLIP (given a text prompt automatic ground noun phrases to image regions)

- Then each of these images and prompts goes through a visual encoder(Swin transformer) and a text encoder(BERT)

- Then to reflect the relationship between image embedding and text embedding they performed a process called Deep Fusion based on head attention.



- Attention-based diffusion process encodes image data and text data and derives the relationship of output information between each encoder.This process is carried out from the output information of the encoders to the prediction head, propagating it through each module.

$$O^i_{t2i}, P^i_{i2t} = \text{X-MHA}(O^i, P^i), \quad i \in \{0, 1, .., L-1\} \quad (4)$$

$$O^{i+1} = \text{DyHeadModule}(O^i + O^i_{t2i}), \quad O = O^L, \quad (5)$$

$$P^{i+1} = \text{BERTLayer}(P^i + P^i_{i2t}), \quad P = P^L, \quad (6)$$

- Equation 4, 5, and 6 represent the iteration of the Deep Fusion module, where step 4 establishes the relationship between output information of visual encoder(O) and text encoder(P) using multi-head attention.In steps 5 and 6, the output and relation information are merged within each image and text processing module before progressing to the next module.
- Multi-head attention proceeds through the formula below. Each image and text data is multiplied by a weight to create an image query and text query, and attention is calculated through the performance calculation between the two queries.

$$O^{(q)} = OW^{(q,I)}, P^{(q)} = PW^{(q,L)}, Attn = O^{(q)}(P^{(q)})^\top/\sqrt{d},$$

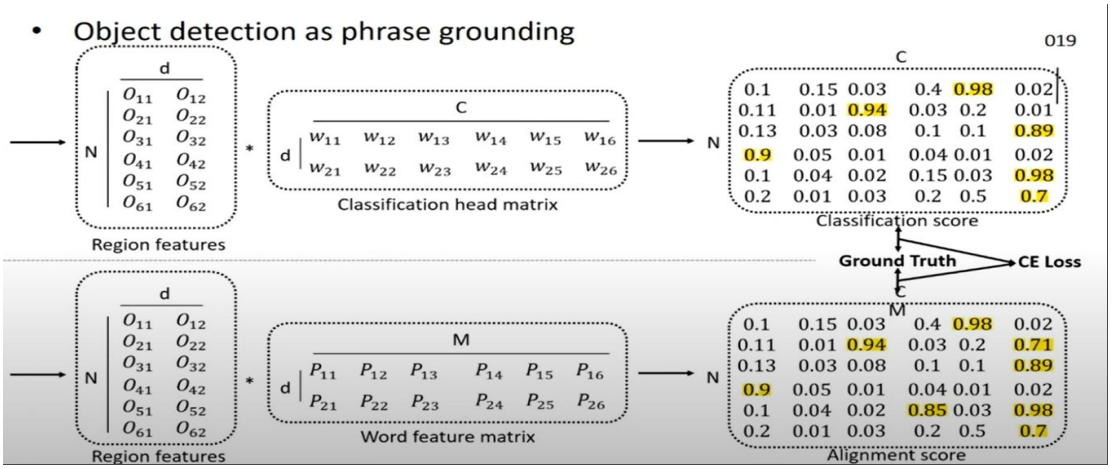
$$P^{(v)} = PW^{(v,L)}, \quad O_{t2i} = \text{SoftMax}(Attn)P^{(v)}W^{(out,I)},$$

$$O^{(v)} = OW^{(v,I)}, \quad P_{i2t} = \text{SoftMax}(Attn^\top)O^{(v)}W^{(out,L)},$$

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

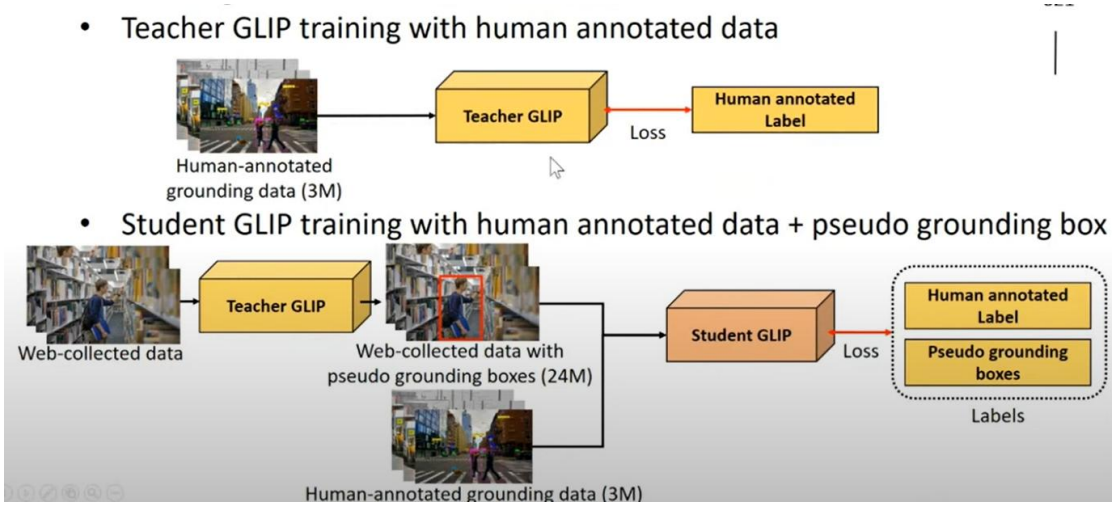
Image Text

- So, the attention goes through the softmax, multiplied by the image value and text value and then multiplied by the weight matrix to derive the result value through a process very similar to the transformer attention that derives the result value. The derived result value is added with the output information of the existing encoder and the process of moving to the next module is repeated several times.
- So, after going through the deep fusion process, the final alignment loss is obtained through the process of finding the cosine similarity between the word feature and the region feature.
- The process of obtaining the alignment score is similar to obtaining the classification score. In the classification process, the region feature matrix, representing n object candidates(d-dim), is multiplied by the class head matrix to derive scores for each class. The largest score is compared with Ground Truth using cross-entropy loss to determine the object's class. Similarly, in the alignment process, the word feature matrix replaces the class head matrix, representing words that describe the object. Multiple scores can be activated for one object since it can be expressed using multiple words. Learning in this paper follows the same cross-entropy loss approach as in the classification process.



Pre-training with Scalable Semantic -Rich Data:

The proposed method involves pseudo-labeling through a teacher network.It allows the student network to learn more information by utilizing noun phrases from grounding data instead of simple categories. The grounding data contains a significantly larger amount of information compared to object detection datasets. The process of creating pseudo-labels in GLIP involves training the teacher GLIP using human annotated data(3M), generating additional data, creating pseudo-labels for web-collected image-text data(24M) through the teacher network, and training the student GLIP using a combination of the pseudo-labeled data and the existing teacher network data. This approach enables the student GLIP to learn from a larger amount of data and outperform the teacher GLIP.



One model for all detection tasks through prompt tuning: GLIP takes a language prompt as input; thus one could change the model predictions by tuning only the prompt embeddings. This is similar to linear probing but the key difference is that the language and visual representation in GLIP is deeply fused. In the results of the paper we see,prompt tuning on GLIP almost matches full fine-tuning, while linear probing a conventional object detection cannot. This makes deploying GLIP efficient : one GLIP model can simultaneously perform well on all downstream tasks, reducing the fine-tuning and deployment cost.GLIP's performance is compared to a fully-supervised Dynamic Head model on 13 downstream object detection tasks. Even with just one-shot training, GLIP is shown to be comparable to the fully-supervised approach.

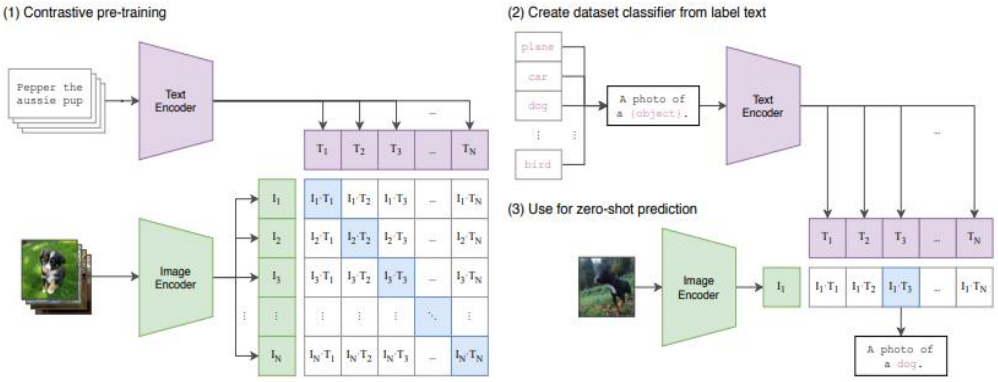
Conclusion:
The learned representations from GLIP demonstrate strong transferability to various object-level recognition tasks, even in zero-shot and few-shot scenarios. The model outperforms supervised baselines on the COCO and LVIS datasets.After fine-tuning on the COCO dataset, GLIP achieves higher average precision (AP) scores on both the validation and test-dev sets, surpassing previous state-of-the-art methods.

Q.II.
Discuss how this paper is different from previous paper (CLIP) which did pre-training for image language alignment.

Ans: CLIP (Contrastive Language-Image Pretraining) and GLIP (Grounded Language-Image Pretraining) are both models that integrate language and vision for various tasks, but they have some key differences:

- **Focus:** CLIP focuses on the alignment of images and text by learning a shared representation space, enabling the model to understand the relationship between visual and textual information and tries to perform **image level task like image classification** .GLIP, on the other hand, specifically targets object-level, language-aware, and semantic-rich visual representations by unifying object detection and phrase grounding for pre-training.It focuses on **object level task like object detection.**

- **Text Data Format:While pre-training in case of GLIP,**the input is entered as a **noun phrase** describing each object rather than a sentence that describes the entire image because the task has to be performed at the object level rather than the image level. By using the NLP Parser introduced in this book called 'NLP with python', only the part corresponding to the noun phrase in the sentence is extracted and used in this way. So, the noun phrases that describe the object in each sentence create text data in a format that is distinguished through the dot. **In case of CLIP web crawled raw text is provided as input because it performs image level task.**
- **Pretraining Data**: CLIP is pretrained on a large-scale dataset that pairs images with their corresponding text captions or surrounding text, creating a multimodal dataset. GLIP, on the other hand, utilizes grounding data, which includes both human-annotated(3M) and web-crawled image-text pairs(24M) This grounding data covers a larger vocabulary of visual concepts than traditional detection data.
- **Approach:** CLIP employs a contrastive learning framework, where it maximizes agreement between representations of matching image-text pairs and minimizes agreement with mismatched pairs. **GLIP, on the other hand, incorporates an attention-based diffusion process to encode image and text data and derive the relationship between them. It also utilizes multi-head attention and merges the output information with relation information in** each image and text processing module.



CLIP

- **Training Strategy:** CLIP uses a self-supervised learning strategy for pretraining, where it learns from the alignment of images and text without explicit human annotations. GLIP, on the other hand, employs a combination of human-annotated detection and grounding data for pretraining. **It utilizes a teacher-student training approach, where the teacher model predicts boxes for web-collected image-text data, and the student model is trained using both teacher's training data and pseudo grounding data generated by the teacher.**

Q.III

In the proposed paper, 24 Million Image – text pairs are web crawled (without fine-grained human supervision) to create a massive image-text pair for pre-training. Suggest a possible direction you will take, in case you want to emulate similar approach in grounding videos ?

Ans: If you want to emulate a similar approach to grounding videos with a large-scale dataset, you can consider the following possible direction:

Data Collection: Web Crawl Videos - Similar to web crawling for image-text pairs, you would need to crawl a large number of videos from various sources on the web. This would involve collecting video data along with any accompanying textual information, such as video titles, descriptions, transcripts,and captions.

Preprocessing and Annotation: Preprocess the textual information by cleaning and normalizing the text. Additionally, you might consider performing natural language processing (NLP) techniques to extract more structured information from the text, such as named entities like nouns for object detection in video frame or verb for action detection . This preprocessing step can help in enriching the textual information associated with each video.

Alignment and Grounding: The next step is to align the video content with the textual information. This can involve techniques like Object detection in each frame,temporal segmentation, where you divide the videos into meaningful segments or shots. Then, you can use methods such as shot boundary detection, object tracking(SORT and Deep SORT) in various frames, or scene classification to identify and annotate relevant visual information in the videos. The goal is to establish a connection between the video content and the corresponding text.

Model Training: With the video-text pairs and their corresponding annotations, you can proceed to pretrain a model similar to GLIP but designed for video grounding. The model should be trained to learn the relationship between video content and textual information, leveraging the aligned annotations. This training process can involve attention mechanisms, diffusion processes, or other techniques suitable for modeling the complex interactions between video and text.

Evaluation and Fine-tuning: Once the model is pretrained, you can evaluate its performance on various video grounding tasks, such as object localization, activity recognition, or video captioning. This evaluation will help in knowing the effectiveness of the pretrained model in understanding and grounding textual information within videos. Fine-tuning can then be applied using specific downstream tasks and datasets to further optimize the model's performance.