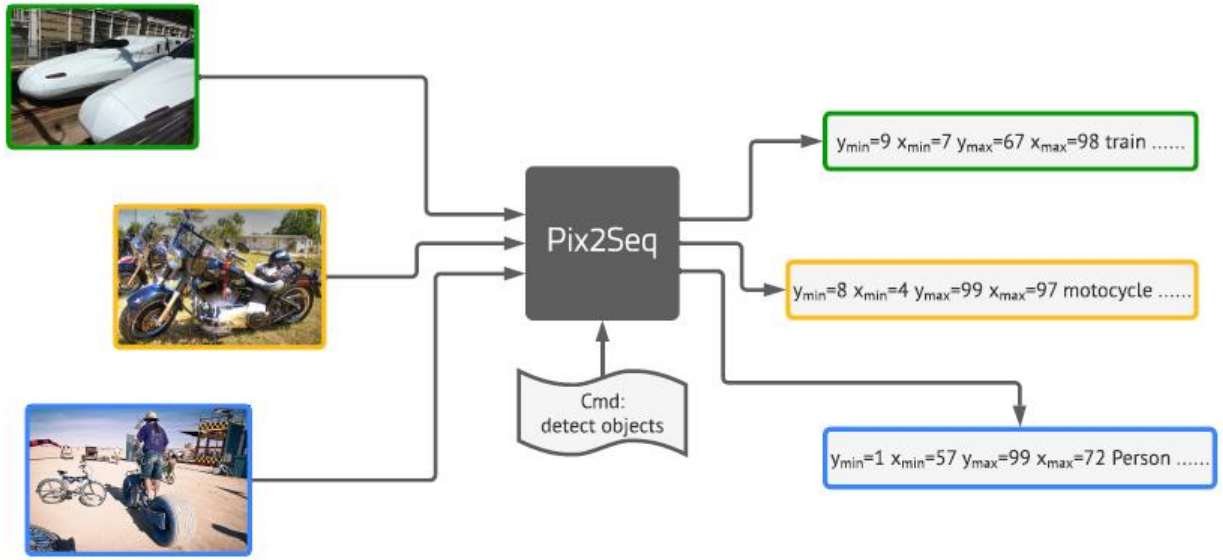


Pix2seq: A Language Modeling Framework for Object Detection

Ting Chen, Saurabh Saxena, Lala Li,
David J. Fleet, Geoffrey Hinton

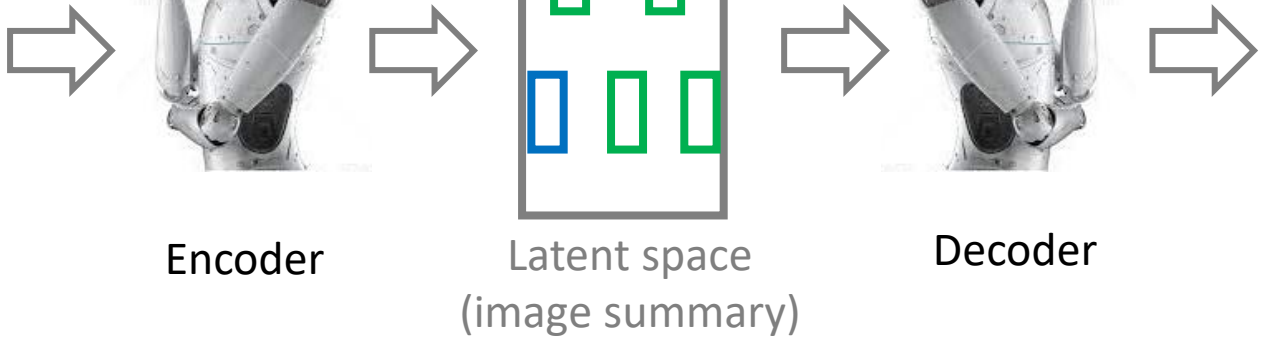
ICLR 2021

Problem Statement

Object Detection

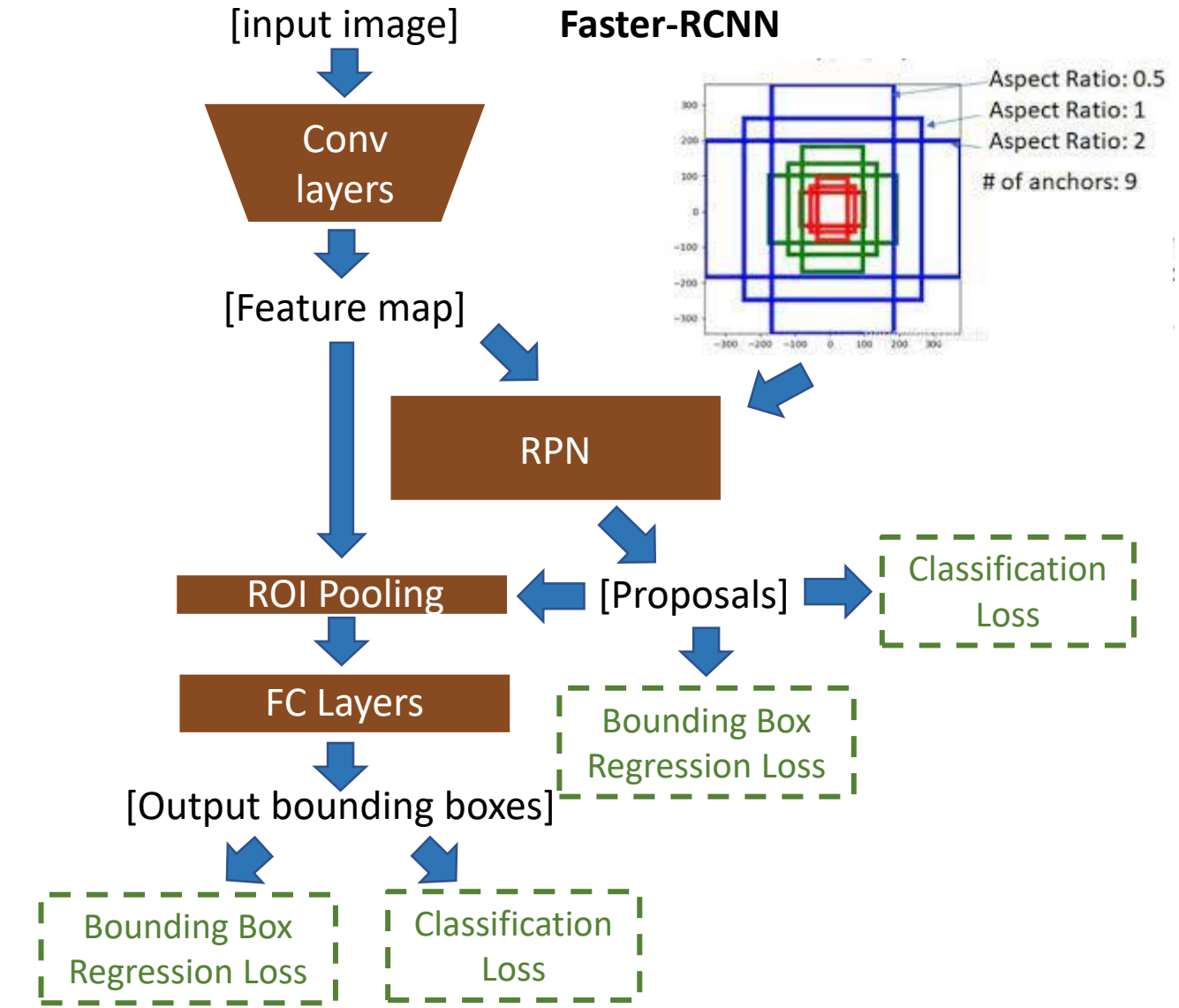
- Existing approaches:
  - Complicated
  - Needs to choose architecture and loss function. (region proposal, ROI pooling, etc.)

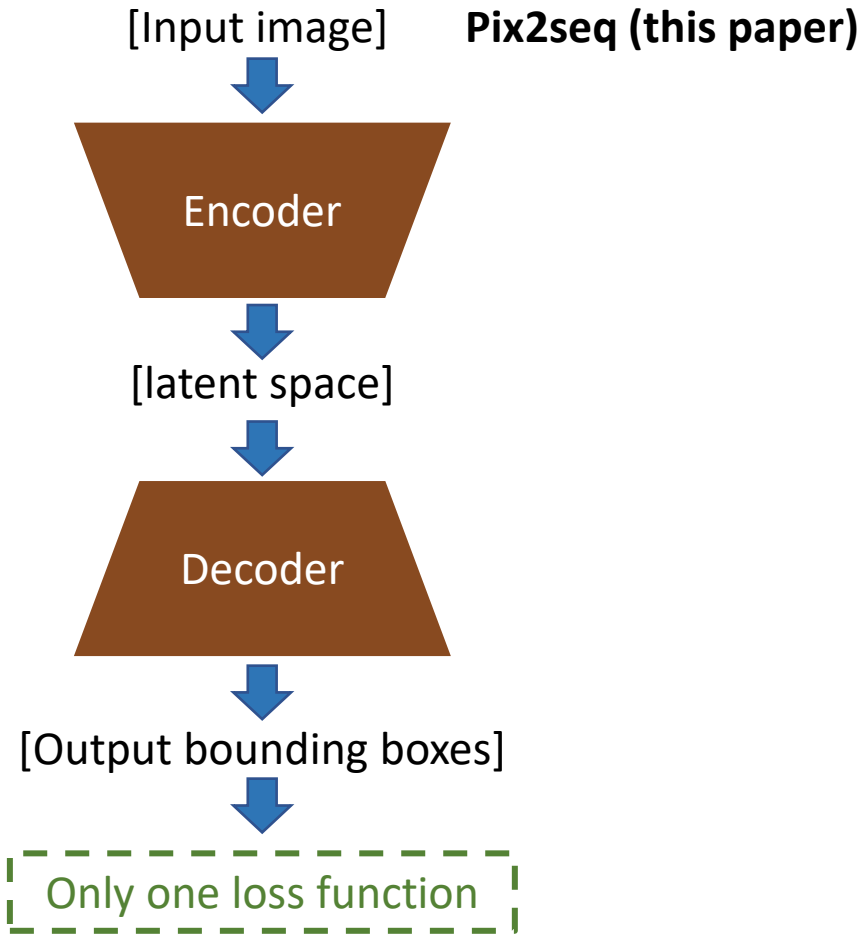
- Proposed approach:
  - Simple
  - Intuition: if a neural net knows about where and what the objects are, it just need to learn to read them out.



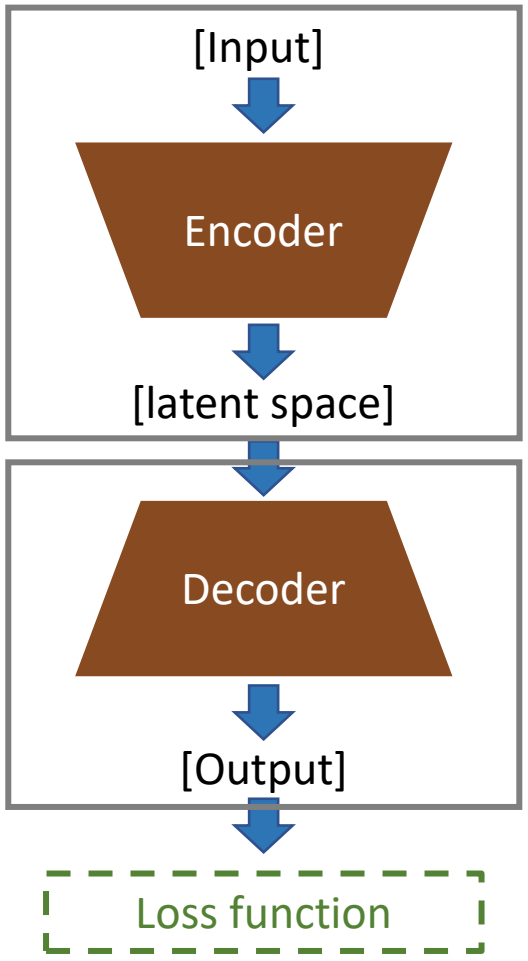
1. Position: [x1, y1; x2, y2];
   class: man
2. Position: [x3, y3; x4, y4];
   class: man
3. Position: [x5, y5; x6, y6];
   class: woman
4. ……

Faster R-CNN vs. Pix2seq





Overall design of Pix2seq

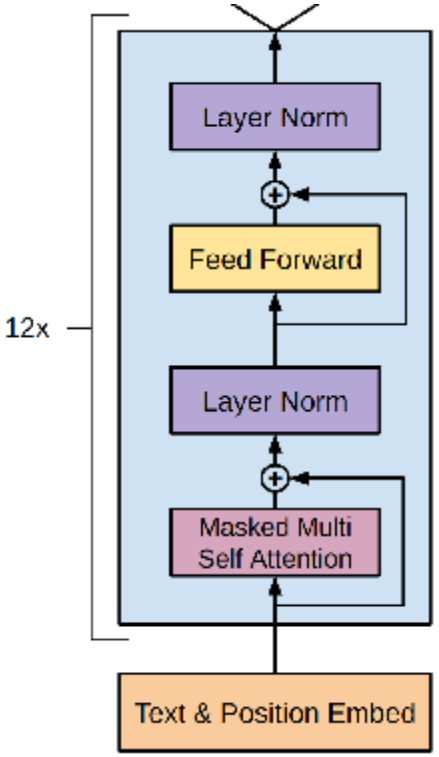


- No restriction.
- Any network that extracts features from images can be used.
- E.g. CNN, Transformer

- Structure used in language modeling
- Input preceding token + latent space
- Output the next token (bounding box + class)
- Generates one bounding box at a time

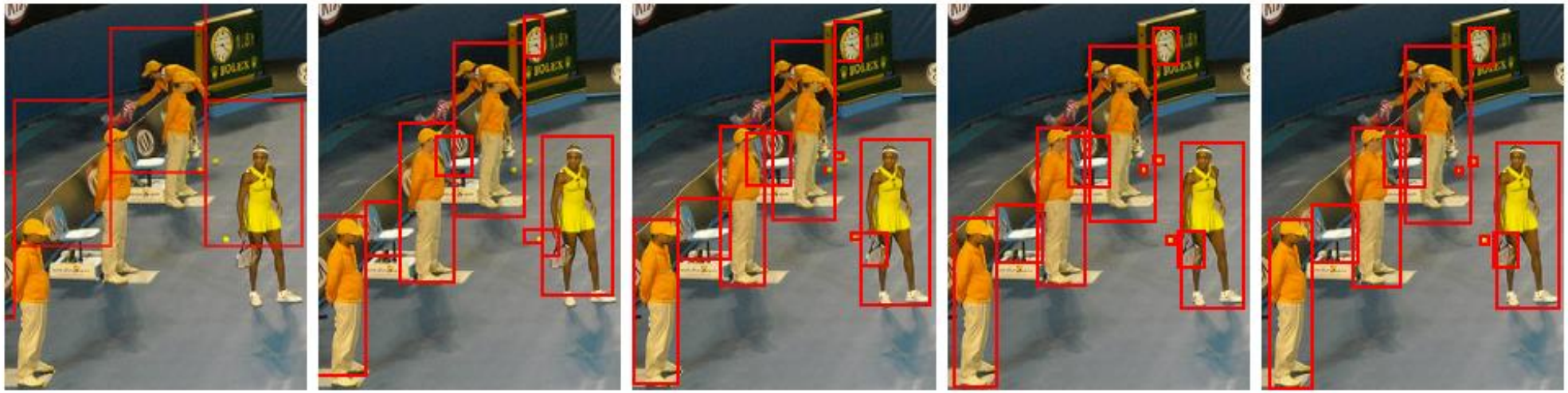
$$\text{maximize} \sum_{j=1}^{L} w_j \log P(\tilde{y}_j | x, y_{1:j-1})$$

Maximize the log likelihood of tokens conditioned on the image and the preceding tokens



Turning object into sequence (1)

- Each object is represented as [y_{min}; x_{min}; y_{max}; x_{max}; c], called a token.
- Use a shared vocabulary for all tokens
  - Cast each value of y_{min}; x_{min}; y_{max}; x_{max} into an integer in range [1, number of bins]
  - vocabulary size = number of bins + number of classes.
  - A small vocabulary can achieve high precision.

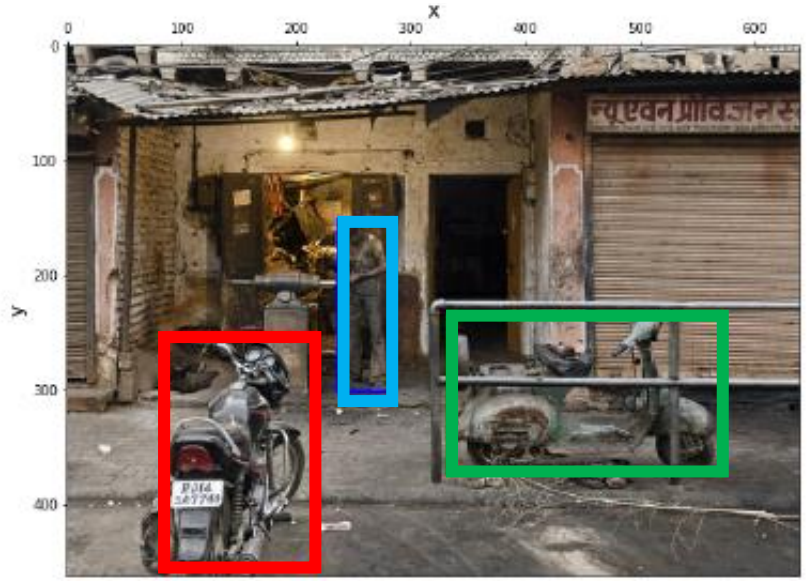


(a) $n_{\text{bins}} = 10$ (b) $n_{\text{bins}} = 50$ (c) $n_{\text{bins}} = 100$ (d) $n_{\text{bins}} = 500$ (e) Original

With a small number of bins, such as 500 bins (1 pixel/bin), it achieves high precision even for small objects.

Turning object into “sentence” (2)

- Randomly order the tokens into a sequence. (Unlike NLP, order is not important)
- Similar to NLP, add an **EOS** token to indicate object detection is completed.



327 370 653 444 1001 544 135 987 338 1004 508 518 805 892 1004 0

$[y_{min}; x_{min}; y_{max}; x_{max}; c]$

EOS

- At inference time, use largest likelihood (argmax sampling), or using other stochastic sampling techniques to sample tokens from model likelihood, i.e., $P(y_j | x, y_{1:j-1})$.

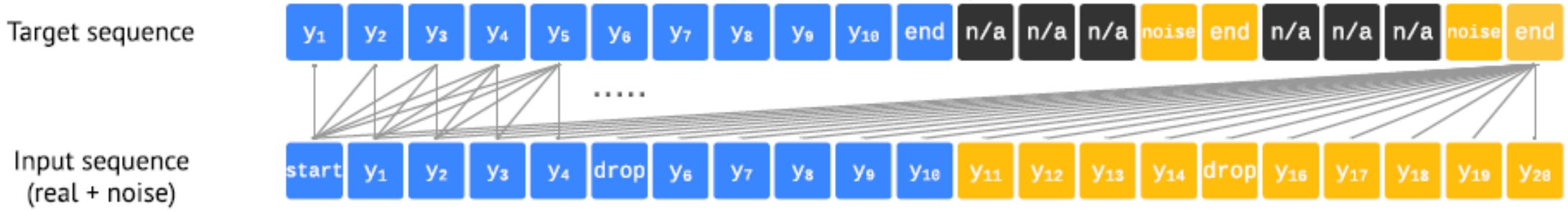
Sequence Augmentation

Problem:

- The model tends to generate **EOS** too early that not all objects have been detected.

Solution:

- Decrease the probability ($P(y_j | x, y_{1:j-1})$) to generate EOS token.
  - Leads to noisy and duplicated predictions.
- Improve robustness. Sequence Augmentation.
  - Introduce a new class called "noise".
  - In training, use both **real tokens** and synthetic **noise tokens** with "noise" label.
  - Set loss weight of **"N/A" tokens** to zero to prevent model from learning them.



Dataset

- COCO
  - 330K images (>200K labeled)
  - 1.5 million object instances
  - 80 object categories

Experiment 1

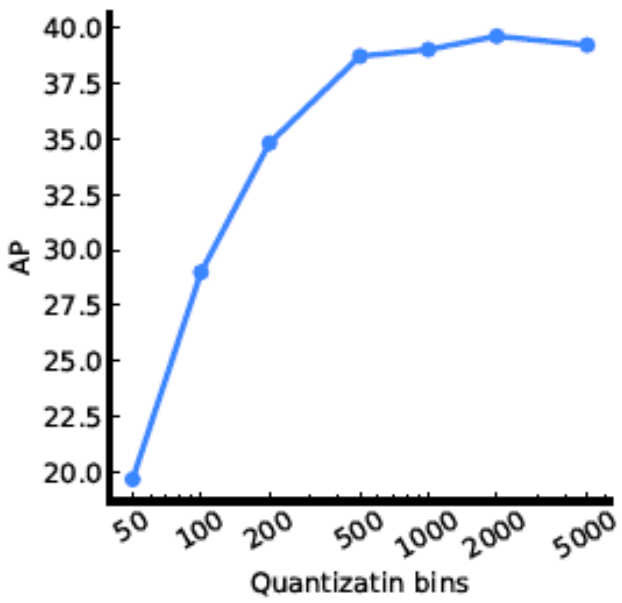
- Performance comparison on COCO dataset.

Method	Backbone	#params	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster R-CNN	R50-FPN	42M	40.2	61.0	43.8	24.2	43.5	52.0
Faster R-CNN+	R50-FPN	42M	42.0	62.1	45.5	26.6	45.4	53.4
DETR	R50	41M	42.0	62.4	44.2	20.5	45.8	61.1
Pix2seq (Ours)	R50	37M	**43.0**	61.0	45.6	25.1	46.9	59.4
Faster R-CNN	R101-FPN	60M	42.0	62.5	45.9	25.2	45.6	54.6
Faster R-CNN+	R101-FPN	60M	44.0	63.9	47.8	27.2	48.1	56.0
DETR	R101	60M	43.5	63.8	46.4	21.9	48.0	61.8
Pix2seq (Ours)	R101	56M	**44.5**	62.8	47.5	26.0	48.2	60.3
Faster R-CNN	R50-DC5	166M	39.0	60.5	42.3	21.4	43.5	52.5
Faster R-CNN+	R50-DC5	166M	41.1	61.4	44.3	22.9	45.9	55.0
DETR	R50-DC5	41M	**43.3**	63.1	45.9	22.5	47.3	61.1
Pix2seq (Ours)	R50-DC5	38M	**43.2**	61.0	46.1	26.6	47.0	58.6
DETR	R101-DC5	60M	**44.9**	64.7	47.7	23.7	49.5	62.3
Pix2seq (Ours)	R101-DC5	57M	**45.0**	63.2	48.6	28.2	48.9	60.4

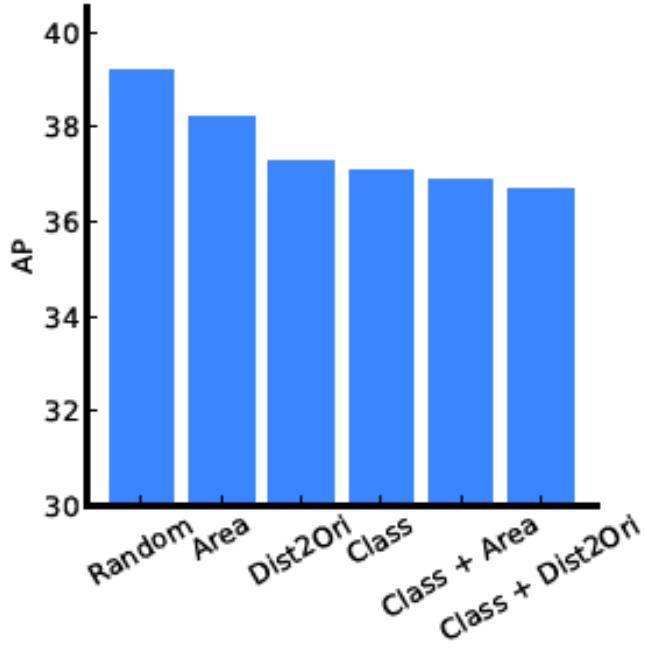
- Better than Faster R-CNN
- Competitive results compared with DETR

Experiment 2

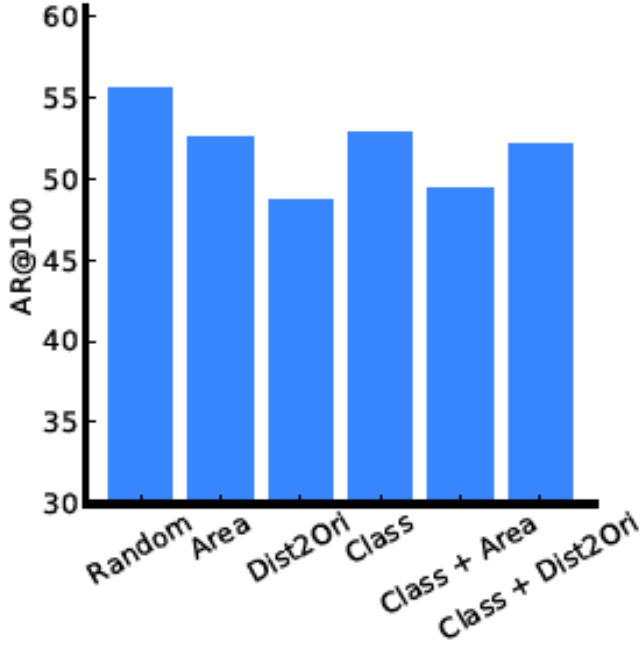
a) Number of bins' influence on performance.
b) Different object ordering strategies' influence on performance.
c) Similar to b), the metric changes.



(a)



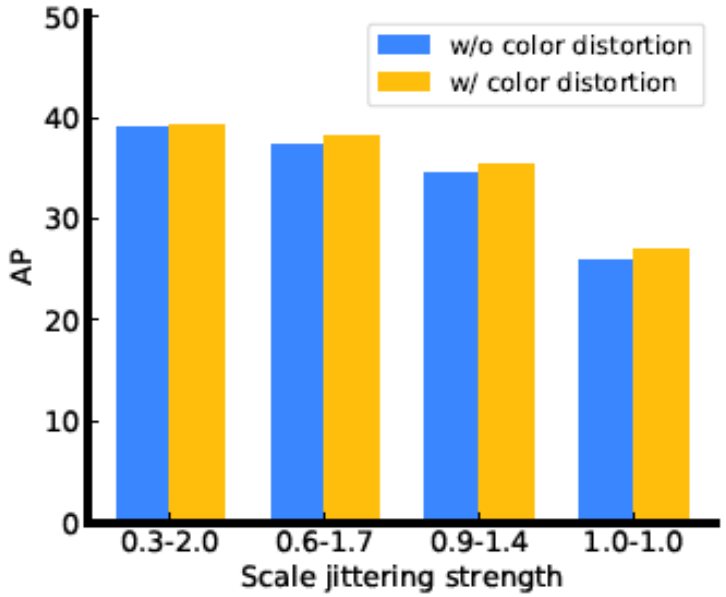
(b)



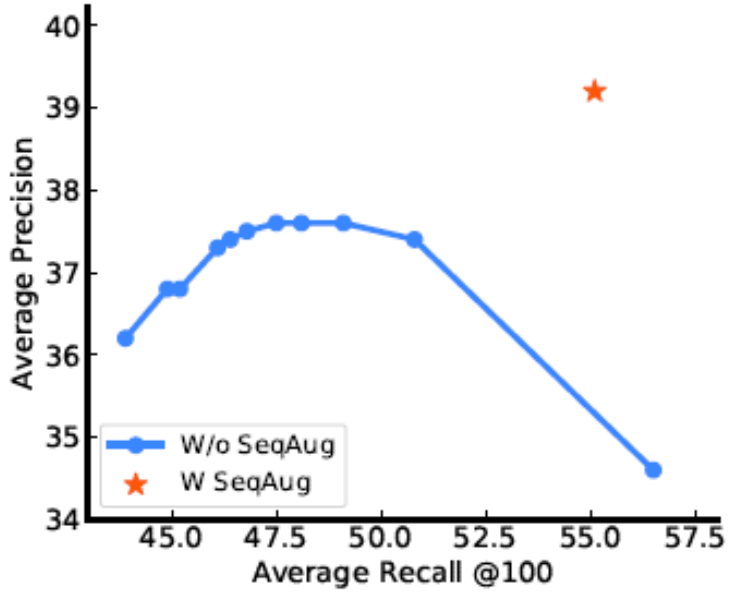
(c)

Experiment 3

a) Image augmentation is important at preventing overfitting
b) Sequence augmentation helps the model achieve both high recall and high precision.



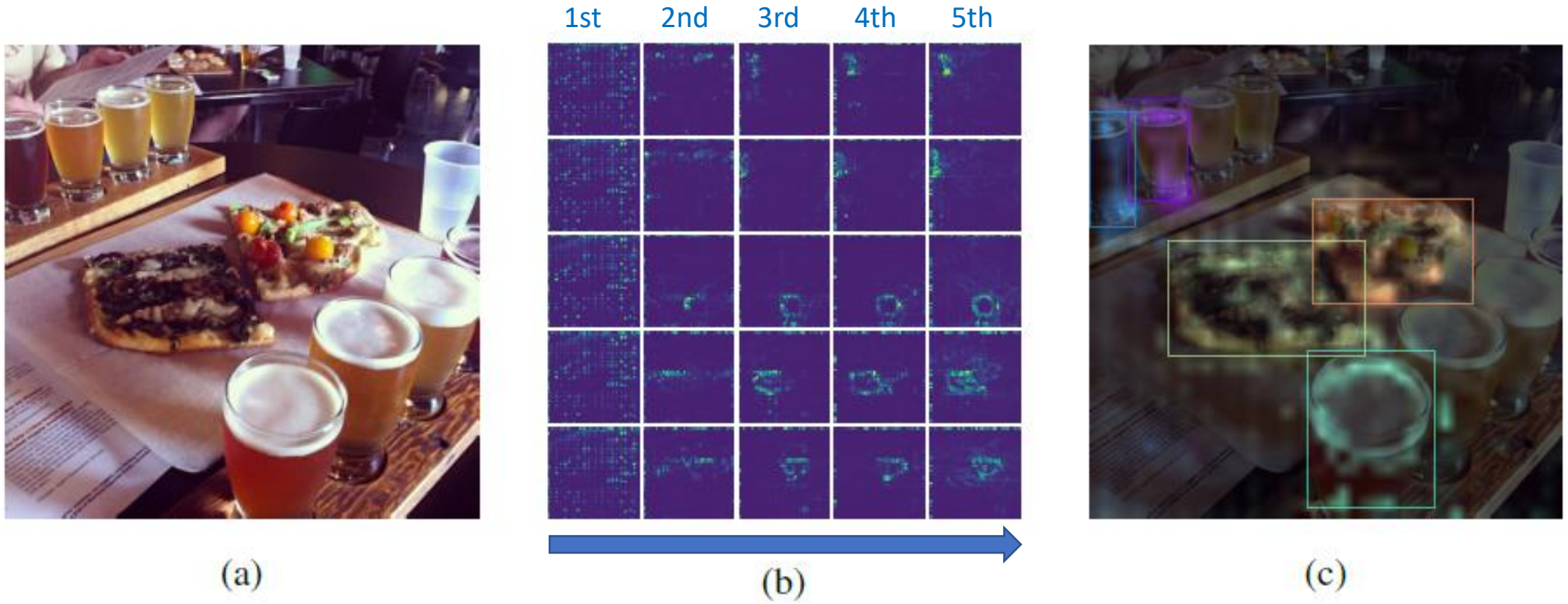
(a) Effects of image scale augmentation.



(b) Effects of sequence augmentation.

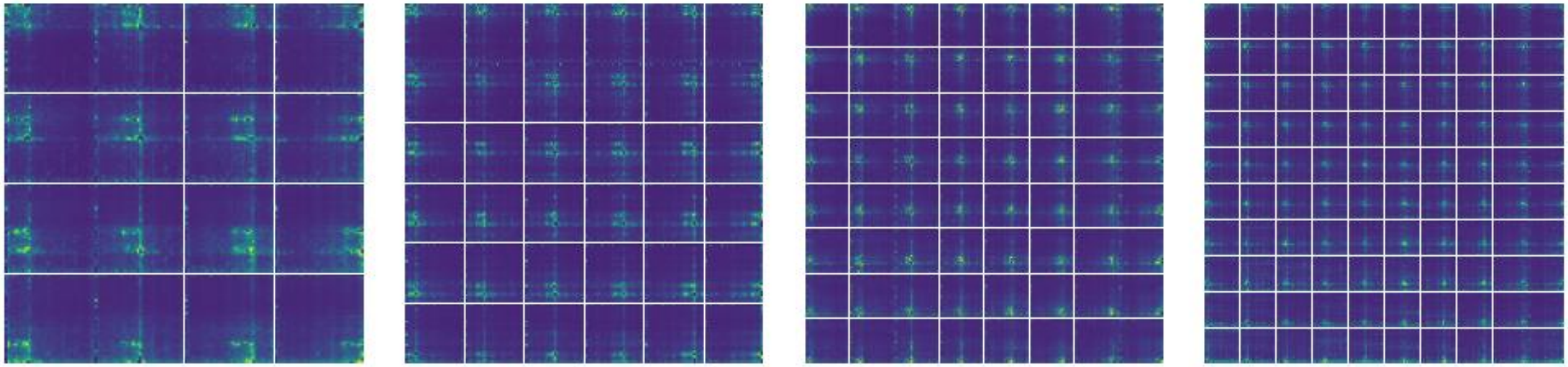
Experiment 4

a) Input image
b) Attention in decoder.
c) Overlay of the cross attention (when predicting the class token) on the original image.



Experiment 5

- Divide an image into regions, specified by a sequence of coordinates for its bounding box.
- The decoder reads the sequence of coordinates for each region.
- Visualize decoder's attention after reading.
- The model can pay attention to the specified region at different scales.



(a) 4×4 grids (b) 6×6 grids (c) 8×8 grids (d) 10×10 grids

Conclusion

- The paper proposes an elegant approach to object detection.
- Number of bounding box isn’t fixed. Flexible.
- When inferencing, the model can be slower than traditional computer vision models that can generate boxes in parallel.

Discussion

- The model did not fully outperform baseline models. And the authors did not make too much novelty. Why 3 / 5 reviewers in ICLR think it’s a good paper? (And received positive reviews from other reviewers)

- Which do you prefer? DETR or Pix2seq?

Thank you!