Pipeline construction for the automated text retrieval, editing, and deletion in comic illustrations

Jordi van Setten

TU Delft

Abstract

This paper proposes leveraging popular comic series like Dilbert to overcome the shortage of high-quality data in Machine Learning and AI. The approach involves extracting and manipulating text data from comic strips to create new high-quality data. The output enables experiments in generative comics, humor detection, translation, and more.

In the end we achieved an accuracy of 98.0% (3.77% improvement from baseline) and successful editing and deletion of text boxes within comic panels. We believe the results are suitable for select use cases.

Research Questions

- How accurately can we get current OCR models to extract text from the comics
- How can we create the ability to edit and delete existing text boxes

Methodology

For text extraction we utilize Panel Segmentation, Text Segmentation, and ultimately OCR to retrieve the text.

- Panel Segmentation uses a contour detection algorithm in order to detect edges around the panels and use that to segment them.
- Text segmentation we are identifying the locations of the text boxes within the panel using object detection models.
- OCR we use a model to read the text inside of the located text boxes

This sequence gives us our final result

Comic Images Input → Panel Segmentation → Text Segmentation → OCR Model Text Extraction →

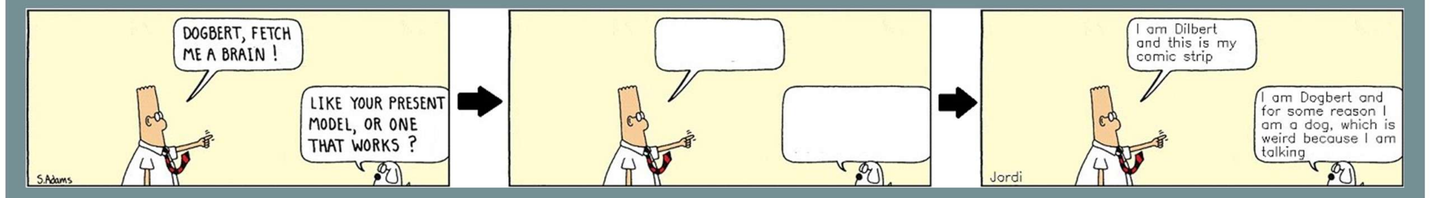
AND WHAT CAN I GET FOR YOU LITTLE FELLOW?

I WANT IT ALL !! HA HA ! GIVE ME MOUTHWASH, GUM, DEODERANT, LIPSTICK, KRAZY GLUE, ROLAIDS AND ... AND ...

I PANICKED.

Text editing and deletion we use:

- Object detection model to identify the position of the text boxes
- OpenCV to insert a new box over the old one with which:
  - The color is decided based on the most frequently occurring color within the text box
  - The text is wrapped and fitted depending on the dimensions of the text box
  - Or left blank in case of wanting to delete the text



Results

Table 1: Model and Object Detection Comparison

Exp. no.	OCR Model		Object Detection Model		
	Tesseract	Vision API	None	General	Fine tuned
Exp. #1	✓		✓		
Exp. #2	✓			✓	
Exp. #3	✓				✓
Exp. #4		✓	✓		
Exp. #5		✓		✓	
Exp. #6		✓			✓

C/WER = Character/Word Error Rate, which measures the number of Character/Word insertions, substitutions, and deletions necessary in order to make two strings match.

LDA = Latent Dirichlet allocation, which measures how similar the semantic meaning is between two strings.

Table 2: Performance Metrics Comparison

Exp. no.	CER	WER	LDA
Exp. #1	81.97%	69.26%	89.06%
Exp. #2	84.2%	64.11%	90.29%
Exp. #3	86.38%	71.02%	92.8%
Exp. #4	84.21%	80.0%	94.23%
Exp. #5	87.51%	81.47%	95.77%
Exp. #6	**94.07%**	**88.35%**	**98.0%**

Conclusion

The result is a pipeline that creates a high-quality dataset which can be used for various fields of machine learning and computer vision.
However, there are future improvements, such as further improving the finetuned model, and utilizing generative AI or mathematical morphology to improve upon the editing and deletion of text in the comics.

Limitations

With text extraction, a lot of accuracy is lost on some shortcomings of the finetuning process, with some unique text boxes not being recognized.

For the editing and deletion, non-rectangular shapes leave behind odd artifacts when overwritten. Moreover, comics after 2008 use a unique background which impacts this process and more.