Building A Text Recognition AI

TRANSLUE

Hochschule für
Wirtschaft und Recht Berlin
Berlin School of Economics and Law

→ How could we build our own text recognition AI?

→ What do we have to consider here theoretically?

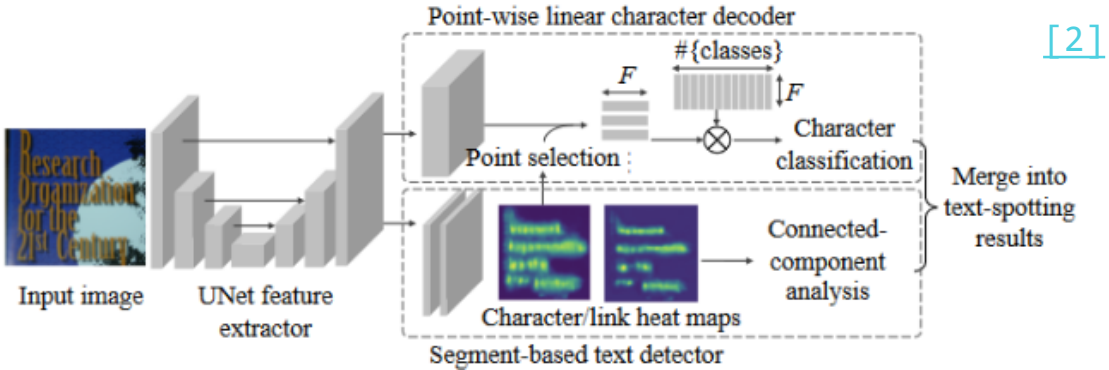
→ What kinds of pre-defined models exist?

→ What kind of data/things do we need to train better?

How could we build our own text recognition AI?

Common Ingredients:

- Image Scaler
  - Resize or crop images to have uniform dimensions (AI *often* needs uniform input) [1]
- Feature Extraction Unit
  - Convolutional Neural Network to locate image patches with text [2] [3]
- Character Classification Unit
  - Convolutional Neural Network to find characters in obtained image patches [2] [3]



[2]

How could we build our own text recognition AI?

Such a system was in fact deployed by Yahoo in 2021. It ran on

iPhone 11 Pro (4GB of memory)
with latencies of ~54ms

[2]

How could we build our own text recognition AI?

- In case of Yahoo:
  Scaler, extraction unit and classifiction unit were prototyped and iterated upon in **Python** using **PyTorch**

- For iPhone: Final logic was transfered to Apple Core ML [2]
- For Android (possibility): Transfer final logic to Google ML Kit

Yahoo‘s system achieved **82.9%** mean accuracy on iPhone.

How could we build our own text recognition AI?

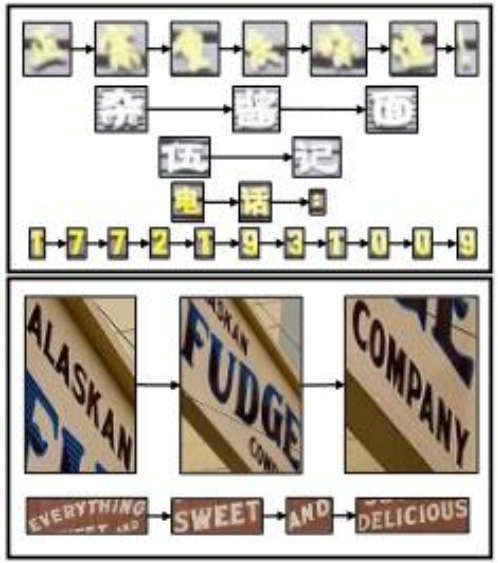
One more thing…

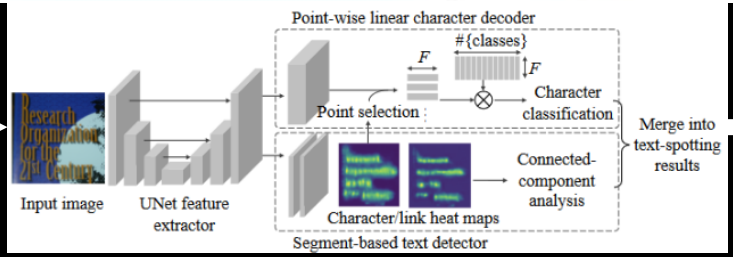
Still missing, but **part of Translue‘s USP**
and **never actually done** on mobile before:

Contextual Segmentation

How could we build our own text recognition AI?

→ ? →

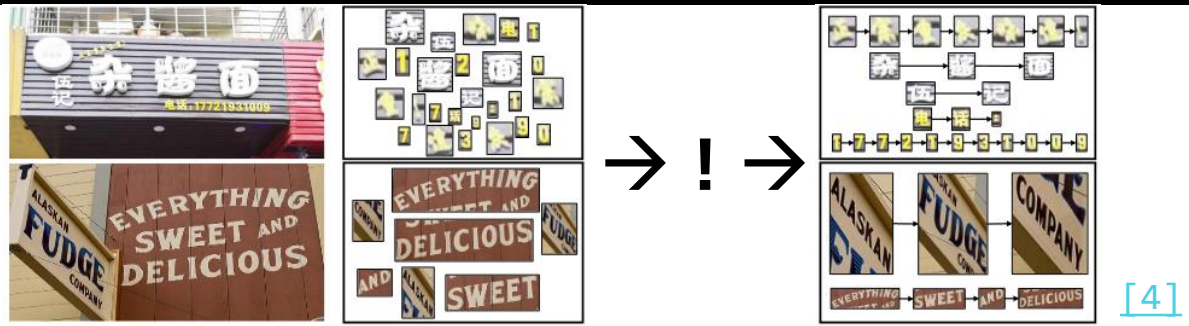




[2]

[4]

How could we build our own text recognition AI?



[4]

Idea:

After detecting text blocks/chunks (conventionally), they should be

- Grouped by common appearance characteristics
- Interpreted in natural reading order, instead of the previously shown „only interpret one textblock individually“ approach

How could we build our own text recognition AI?

Proposed Ingredients:

- Image Scaler
  - Resize or crop images to have uniform dimensions (AI *often* needs uniform input) [1]
- Feature Extraction Unit
  - Convolutional Neural Network to locate image patches with text [2] [3]
- Integral Embedding Extractor
  - Learns visual and contextual feature embeddings for each detected integral text unit [4]
- Contextual Text Block Generator
  - Groups and arranges the detected integral texts in reading order to produce contextual text blocks [4]
- Character Classification Unit
  - Convolutional Neural Network to find characters in obtained image patches [2] [3]

→ How could we build our own text recognition AI?

→ What do we have to consider here theoretically?

→ What kinds of pre-defined models exist?

→ What kind of data/things do we need to train better?

What do we have to consider here theoretically?

- Processing Speed to allow for best possible UX
  - Processing times up to ~100ms seem to be ok for users
- High accuracy in recognition and contextual segmentation, even during swiping
  - **82.9%** was achieved, accuracy **may have to increase**
- Overall high efficiency / small impact on device and battery

- The proposed system is **modular**
  → It has to be evaluated, which parts to build In-House, if any, and which to outsource, if any, e.g. to Google Vision
- Latter might come with making compromises regarding accuracy, former puts more load on device

→ How could we build our own text recognition AI?

→ What do we have to consider here theoretically?

→ What kinds of pre-defined models exist?

→ What kind of data/things do we need to train better?

What kinds of pre-defined models exist?

- A model for a mobile-optimized text detection and recognition is described in theory in [2], but no code is provided
- A model for contextual grouping does not exist, the authors of [4] claim to be first proposal of its kind

- Solutions on text recognition for “standard” machine/server architectures exist with [3] [5] [6], never specify if mobile application is possible

→ How could we build our own text recognition AI?

→ What do we have to consider here theoretically?

→ What kinds of pre-defined models exist?

→ What kind of data/things do we need to train better?

What kind of data/things do we need to train better?

"It's not who has the best algorithm that wins. It's who has the most data.“ – Jean-Claude Heudin

- Two things required, **if** a system is to be built In-House:
  - Large annotated image datasets specifically for text recognition
    - Kaggle
    - ICDAR2019
    - ICDAR2015
  - Perform training on data in epochs
    - Run exact same dataset repeatedly but shuffled through the system
    - Specifically avoid Overfitting (High Variance) though, but get the most out of the data

References

[1] - https://medium.com/mindboard/image-classification-with-variable-input-resolution-in-keras-cbfbe576126f, 30.08.22
[2] - Yoshihashi, Ryota, et al. "Context-Free TextSpotter for real-time and mobile end-to-end text detection and recognition.“ (Yahoo)
[3] - Bartz, Christian, Haojin Yang, and Christoph Meinel. "STN-OCR: A single neural network for text detection and text recognition.“ (HPI)
[4] - Xue, Chuhui, et al. "Contextual Text Block Detection towards Scene Text Understanding.“ (ByteDance)
[5] - Wang, Xiqi, et al. "R-YOLO: A real-time text detector for natural scenes with arbitrary rotation.“ (Wuhan University)
[6] - Zhang, Chengquan, et al. "Look more than once: An accurate detector for text of arbitrary shapes.“ (Xiamen University)