



Технически университет София

Проект по Технологии за големи данни

Разпознаване на бактерии в петри чинийки и сегментация на
локацията на бактерията

Йоанна Благоева 961324010, Таня Узунова 961324004 АГМПД 2024/25

Въведение	2
1. Цел на изследването	2
2. Експериментална рамка – набори данни (datasets), избор на метод и техника за анализ, използвани библиотеки и софтуерни средства	3
2.1. Набори данни (Datasets)	3
2.2. Избор на метод и техника за анализ	3
2.3. Използвани библиотеки и софтуерни средства	4
3. Обработка и анализ на данните	4
4. Представяне и визуализация на резултатите	5
5. Изводи и заключение	5
6. Речник	7
7. Източници	7

Въведение

Съвременните технологии за машинно обучение са ключови за автоматизацията и оптимизацията на процеси в различни индустрии. В този проект разглеждаме разработката на модел за разпознаване на обекти в изображения, като по-конкретно бактериийни колонии на лабораторни проби и петри чинии, използвайки предварително аотирани данни във формат COCO. Това включва използване на модерни библиотеки и техники за машинно обучение с цел създаване на ефективна система за класификация и анализ.

1. Цел на изследването

Целта на изследването е да се разработи ефективен и евтин модел за разпознаване и класификация на бактериални колонии в изображения на петри чинийки, който може да идентифицира различни видовете бактериални колонии в лабораторна среда и да бъде широко спектърен с лесна имплементация в научната сфера. Основните задачи включват обработка и анализ на изображения, използване на предварително обучени трансформер модели, като DETR, и оптимизация на представянето на модела.

2.3. Използвани библиотеки и софтуерни средства

Включват PyTorch и torchvision за дефиниране и обучение на модела, Hugging face Transformers [3] за интеграция с DETR, PyTorch Lightning [4] за управление на тренировъчния процес и Matplotlib [5] за визуализация на резултатите от модела.

3. Обработка и анализ на данните

- 3.1. На Фиг 2 се наблюдава класът за обработка на данни е специално модифициран за използване с DETR. Той комбинира стандартната функционалност на CocoDetection с допълнителни стъпки за обработка на изображенията и анотациите. За целта се използва DetrImageProcessor за нормализация и преобразуване на анотациите в подходящ формат за DETR.

```
class CocoDetection(torchvision.datasets.CocoDetection):
    def __getitem__(self, idx):
        img, target = super(CocoDetection, self).__getitem__(idx)
        encoding = self.processor(images=img, annotations=target, return_tensors="pt")
        pixel_values = encoding["pixel_values"].squeeze()
        target = encoding["labels"][0]
        return pixel_values, target
```

Фиг. 2

- 3.2. Данните се подават на модела чрез DataLoader, който пакетира изображенията и анотациите в удобен за обучение формат. Функцията collate_fn, показана на фиг. 3, осигурява консистентност в подреждането на данните и подготовката им за модела.

```
def collate_fn(batch):
    pixel_values = [item[0] for item in batch]
    encoding = processor.pad(pixel_values, return_tensors="pt")
    labels = [item[1] for item in batch]
    return {
        'pixel_values': encoding['pixel_values'],
        'pixel_mask': encoding['pixel_mask'],
        'labels': labels
    }
```

Фиг. 3

4. Представяне и визуализация на резултатите

- 4.1. Резултатите от модела се визуализират чрез Matplotlib. Изображенията се показват с наложени bounding boxes, етикети и точности, които са извлечени от резултатите на модела. Имплементацията е показва във фигура 4.

```
def plot_results(pil_img, scores, labels, boxes):
    plt.figure(figsize=(16,10))
    plt.imshow(pil_img)
    ax = plt.gca()
    colors = COLORS * 100
    for score, label, (xmin, ymin, xmax, ymax), c in zip(scores.tolist(), labels.tolist(), boxes.tolist(), colors):
        ax.add_patch(plt.Rectangle((xmin, ymin), xmax - xmin, ymax - ymin,
                                   fill=False, color=c, linewidth=3))
        text = f'{id2label[label]}: {score:0.2f}'
        ax.text(xmin, ymin, text, fontsize=15,
                bbox=dict(facecolor='yellow', alpha=0.5))
    plt.axis('off')
    plt.show()
```

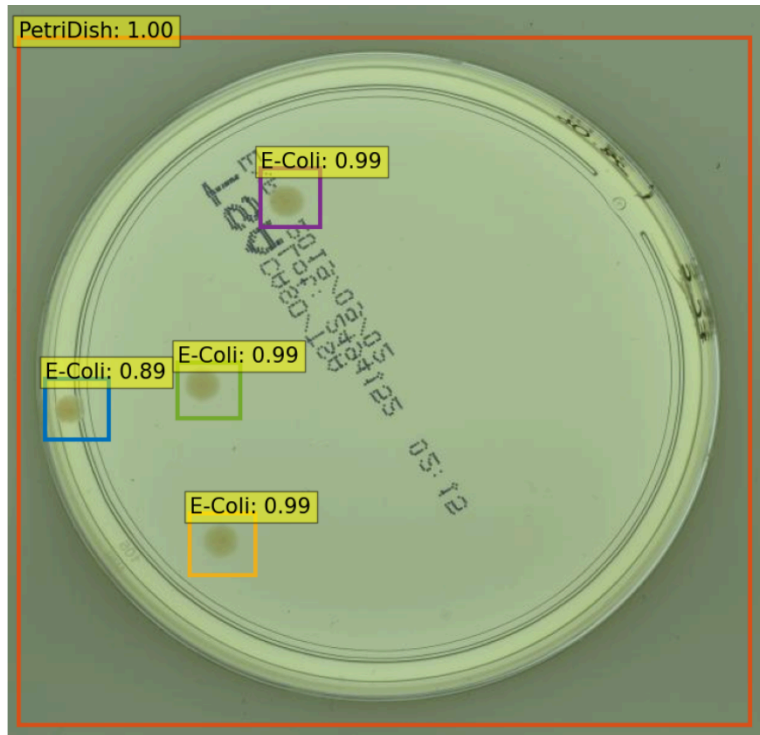
Фиг. 4

- 4.2. Оценката на модела е извършена с помощта на COCO метрики. Основните резултати показват средна точност (AP - Average Precision) от 0.375 при IoU(Intersection over Union)=0.50:0.95 и средно припомняне (AR - Average Recall) от 0.450 при IoU=0.50:0.95. Тези метрики демонстрират конкурентни резултати за големи обекти, но показват предизвикателства при малки обекти.

5. Изводи и заключение

Настоящият проект демонстрира успешното приложение на трансформер - базирания модел DETR за задачата по разпознаване на бактериални колонии в лабораторни изображения.

Средната точност (AP) от 0.375 и средното припомняне (AR) от 0.450 показват, че моделът е способен да идентифицира и класифицира обекти с висока точност, особено когато става дума за големи обекти. Въпреки това, малките обекти представляват значително предизвикателство, което предполага нуждата от допълнителни подобрения в модела или в набора от данни. Интеграцията на COCO формат за анотации осигурява стандартен и добре документиран начин за обучение и оценка на модела.



Фиг. 5: Увереност в модела да разпознава бактерии от клас E-Coli.

Заклученията от проекта показват, че DETR е способен да се адаптира към специализирани задачи, като разпознаване на лабораторни проби, когато бъде обучен с подходящи данни и конфигурации. Бъдещата работа може да включва разширяване на набора от данни чрез включване на повече категории и изображения, както и използване на техники за усилване на обучението, като semi-supervised learning или data augmentation.

6. Речник

DETR - DEtECTION TRansformer - засичащ трансформатор

AP - Average Precision - средна точност

AR - Average Recall - средно припомняне

IoU - Intersection over Union - пресечна точка над обединение

7. Източници

Kaggle линк към notebook с резултати:

<https://www.kaggle.com/code/jonicornpotato/big-data-tu>

Github линк към notebook с резултати и документация:

https://github.com/joannaBlagoeva/tu_petri_4nika

[1] Detr-resnet-50 модел от Hugging face - <https://huggingface.co/facebook/detr-resnet-50>

[2] Статия за resnet-50 [\[1512.03385\] Deep Residual Learning for Image Recognition](#)

[3] Hugging face Transformer библиотека - <https://huggingface.co/docs/transformers/index>

[4] Lightning AI Py-torch библиотека - <https://lightning.ai/docs/pytorch/stable/>

[5] Matplotlib библиотека - [Matplotlib documentation — Matplotlib 3.10.0 documentation](#)

[6] Makrai, L., Fodróczy, B., Nagy, S.Á. *et al.* Annotated dataset for deep-learning-based bacterial colony detection. *Sci Data* 10, 497 (2023). <https://doi.org/10.1038/s41597-023-02404-8>

[7] Yang F, Zhong Y, Yang H, Wan Y, Hu Z, Peng S. Microbial Colony Detection Based on Deep Learning. *Applied Sciences*. 2023; 13(19):10568. <https://doi.org/10.3390/app131910568>