Báo Cáo Thực Hành: Phân Vùng Ngữ Nghĩa Sử Dụng Học Sâu

Môn học: Phân tích và Xử lý Hình ảnh

Tác giả: Lê Hoàng Anh

Mssv: 22022563

1. Mục Tiêu

- Tìm hiểu các phương pháp phân vùng ngữ nghĩa (semantic segmentation) sử dụng học sâu, phân tích ưu và nhược điểm.
- Áp dụng các phương pháp tiên tiến trên một bộ dữ liệu công khai và báo cáo kết quả thực nghiệm.

2. Tìm Hiểu Các Phương Pháp Phân Vùng Ngữ Nghĩa Sử Dụng Học Sâu

2.1. Tổng Quan Về Phân Vùng Ngữ Nghĩa

Phân vùng ngữ nghĩa là một bài toán trong xử lý ảnh, nhằm phân loại từng pixel trong ảnh thành một danh mục cụ thể (ví dụ: vùng tổn thương, nền, cơ quan). Các phương pháp học sâu đã cải thiện đáng kể hiệu suất của phân vùng ngữ nghĩa, đặc biệt trong các lĩnh vực như ảnh y tế, xe tự hành, và phân tích cảnh.

2.2. Các Phương Pháp Phân Vùng Ngữ Nghĩa Tiên Tiến

Dưới đây là một số phương pháp học sâu phổ biến và phân tích ưu, nhược điểm:

1. **U-Net**:
    - **Mô tả**: U-Net là một mạng encoder-decoder với skip connections, được thiết kế cho phân vùng ảnh y tế. Encoder trích xuất đặc trưng, decoder tái tạo ảnh, và skip connections giữ lại thông tin chi tiết không gian.
    - **Ưu điểm**:
        - Hiệu quả trên dữ liệu nhỏ, phù hợp với ảnh y tế (như phân vùng tổn thương da).
        - Số lượng tham số ít, huấn luyện nhanh.
        - Giữ chi tiết ranh giới tốt nhờ skip connections.
    - **Nhược điểm**:
        - Khả năng nắm bắt ngữ cảnh toàn cục (global context) hạn chế.

- Có thể không hiệu quả với các vùng có kích thước và hình dạng đa dạng.

2. **DeepLabV3**:
   - **Mô tả**: DeepLabV3 sử dụng Atrous Convolution và Atrous Spatial Pyramid Pooling (ASPP) để thu thập thông tin ngữ cảnh ở nhiều tỷ lệ. Thường dùng backbone như ResNet50.
   - **Ưu điểm**:
     - Nắm bắt ngữ cảnh đa tỷ lệ tốt nhờ ASPP.
     - Hiệu quả trên các tập dữ liệu lớn như COCO, PASCAL VOC.
     - Độ chính xác cao trong các bài toán phức tạp.
   - **Nhược điểm**:
     - Số lượng tham số lớn, yêu cầu tài nguyên tính toán cao.
     - Hiệu suất có thể giảm trên dữ liệu nhỏ hoặc ít đa dạng.
3. **DeepLabV3+**:
   - **Mô tả**: Cải tiến của DeepLabV3, kết hợp encoder-decoder với ASPP để cải thiện chi tiết ranh giới.
   - **Ưu điểm**:
     - Kết hợp lợi thế của U-Net (chi tiết ranh giới) và DeepLabV3 (ngữ cảnh toàn cục).
     - Hiệu suất cao trên nhiều bài toán.
   - **Nhược điểm**:
     - Phức tạp hơn DeepLabV3, cần nhiều tài nguyên hơn.
     - Yêu cầu điều chỉnh siêu tham số cẩn thận.
4. **SegNet**:
   - **Mô tả**: Một mạng encoder-decoder với max-pooling indices để truyền thông tin không gian từ encoder sang decoder.
   - **Ưu điểm**:
     - Nhẹ, phù hợp với thiết bị có tài nguyên hạn chế.
     - Hiệu quả trong các ứng dụng thời gian thực.
   - **Nhược điểm**:
     - Độ chính xác thấp hơn so với U-Net hoặc DeepLab.
     - Khó nắm bắt các vùng phức tạp.

Phương pháp	Độ chính xác	Độ phức tạp	Phù hợp với dữ liệu nhỏ	Chi tiết ranh giới	Ngữ cảnh toàn cục
U-Net	Trung bình - Cao	Thấp	Có	Cao	Thấp
DeepLabV3	Cao	Cao	Không	Trung bình	Cao
DeepLabV3+	Rất cao	Rất cao	Không	Cao	Cao
SegNet	Trung bình	Thấp	Có	Trung bình	Thấp

3. Thực Nghiệm Với Bộ Dữ Liệu

3.1. Thông Tin Dữ Liệu

- **Tập dữ liệu**:
  - **Tập train**: 1087 cặp ảnh và mask (ghép đôi thành công từ 1087 ảnh và 1087 mask).
  - **Chia dữ liệu**:

        - **Train**: 978 mẫu (90%).
        - **Validation**: 109 mẫu (10%).
    - **Tập test**: 192 ảnh (không có mask).
- **Tiền xử lý**:
    - **Transforms train**: Resize 512×512, lật ngang (p=0.5), điều chỉnh sáng/tương phản (p=0.2), chuẩn hóa (mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]).
    - **Transforms validation/test**: Resize 512×512, chuẩn hóa.
    - Mask nhị phân hóa (threshold > 127).
- **DataLoader**:
    - Train: Batch size=4, shuffle=True, num_workers=2, pin_memory=True.
    - Validation/Test: Batch size=1, shuffle=False, num_workers=2, pin_memory=True.

3.2. Quá Trình Huấn Luyện

- **Mô hình**:
    - **DeepLabV3**: Backbone ResNet50, pretrained trên COCO, đầu ra 1 kênh (phân vùng nhị phân).
    - **U-Net**: Backbone ResNet50, pretrained trên ImageNet, đầu ra 1 kênh.
- **Hàm mất mát**: Kết hợp BCEWithLogitsLoss và Dice Loss (tỷ lệ 0.5:0.5).
- **Optimizer**: Adam, learning rate=1e-4.
- **Số epoch**: 10.
- **Thiết bị**: GPU (Kaggle environment).
- **Metrics theo dõi**: Train Loss, Val Loss, IoU, Dice, Precision, Recall, F1.
- **TensorBoard**: Ghi lại loss và metrics.

3.3. Kết Quả Huấn Luyện

3.3.1. DeepLabV3

- **Thời gian huấn luyện**:
    - Train: ~2:56 phút/epoch (~1.39 it/s).
    - Validation: ~7 giây/epoch (~14.5 it/s).
- **Kết quả qua 10 epoch** (trên tập validation):

Epoch	Train Loss	Val Loss	IoU	Dice	Precision	Recall	F1
1	0.2653	0.1821	0.8136	0.7801	0.8724	0.9365	0.8905
2	0.1741	0.1391	0.8448	0.8340	0.9137	0.9216	0.9100
3	0.1365	0.1357	0.8307	0.8505	0.9346	0.8903	0.9019
4	0.1089	0.1312	0.8242	0.8629	0.9720	0.8490	0.8980
5	0.1420	0.1450	0.8240	0.8535	0.9209	0.8893	0.8911
6	0.1066	0.1234	0.8441	0.8795	0.9164	0.9237	0.9117
7	0.1000	0.1103	0.8521	0.8890	0.9351	0.9135	0.9166
8	0.0900	**0.1040**	**0.8626**	**0.8989**	0.9239	**0.9354**	**0.9232**
9	0.0845	0.1181	0.8472	0.8877	**0.9400**	0.9038	0.9130
10	**0.0798**	0.1081	0.8453	0.8940	0.9272	0.9133	0.9130

- **Nhận xét**:
  - **Train Loss**: Giảm đều từ 0.2653 xuống 0.0798.
  - **Val Loss**: Giảm từ 0.1821 xuống thấp nhất 0.1040 (epoch 8), nhưng tăng nhẹ ở epoch 9.
  - **Dice**: Tăng từ 0.7801 lên cao nhất 0.8989 (epoch 8).
  - **IoU**: Cao nhất 0.8626 (epoch 8).
  - **F1**: Cao nhất 0.9232 (epoch 8).
  - Hiệu suất tốt nhất đạt ở **epoch 8**, với Dice và IoU cao nhất, nhưng có dấu hiệu dao động nhẹ ở epoch 9-10.

3.3.2. U-Net

- **Thời gian huấn luyện**:
  - Train: ~46 giây/epoch (~5.23 it/s).
  - Validation: ~3 giây/epoch (~29 it/s).
- **Kết quả qua 10 epoch** (trên tập validation):

Epoch	Train Loss	Val Loss	IoU	Dice	Precision	Recall	F1
1	0.3601	0.2961	0.7621	0.6557	0.7953	**0.9597**	0.8540
2	0.2216	0.2117	0.8069	0.7518	0.9037	0.8872	0.8823
3	0.1810	0.1735	0.8020	0.8070	0.8522	0.9447	0.8849
4	0.1477	0.1331	0.8379	0.8616	**0.9272**	0.9053	0.9085
5	0.1342	0.1467	0.8251	0.8508	0.8945	0.9256	0.8985
6	0.1163	0.1333	0.8322	0.8648	0.9247	0.9028	0.9039
7	0.1141	0.1370	0.8283	0.8657	0.9007	0.9218	0.9004
8	0.1087	0.1239	0.8365	**0.8749**	0.9080	0.9250	0.9046
9	**0.0954**	0.1292	0.8239	0.8691	0.8816	0.9372	0.8965
10	0.0997	**0.1171**	**0.8425**	0.8744	0.9117	0.9258	**0.9099**

- **Nhận xét**:
  - **Train Loss**: Giảm từ 0.3601 xuống 0.0954 (epoch 9), tăng nhẹ ở epoch 10.
  - **Val Loss**: Giảm từ 0.2961 xuống 0.1171 (epoch 10).
  - **Dice**: Tăng từ 0.6557 lên cao nhất 0.8749 (epoch 8).
  - **IoU**: Cao nhất 0.8425 (epoch 10).
  - **F1**: Cao nhất 0.9099 (epoch 10).
  - Hiệu suất cải thiện ổn định, đạt tốt nhất ở **epoch 8-10**, với Dice và F1 cao nhưng vẫn thấp hơn DeepLabV3.

---

4. So Sánh DeepLabV3 và U-Net

- **Hiệu suất**:
  - **DeepLabV3**: Đạt **Dice 0.8989** và **IoU 0.8626** (epoch 8), cao hơn U-Net (**Dice 0.8749**, **IoU 0.8425**).
  - **U-Net**: Tiến bộ ổn định hơn, nhưng hiệu suất tổng thể thấp hơn DeepLabV3.
- **Tốc độ huấn luyện**:
  - **U-Net**: Nhanh hơn đáng kể (~46 giây/epoch so với ~2:56 phút/epoch của DeepLabV3).
  - **DeepLabV3**: Tốn tài nguyên hơn do số tham số lớn và cấu trúc phức tạp.
- **Độ chính xác và chi tiết**:

- **DeepLabV3**: Tốt hơn trong việc nắm bắt ngữ cảnh toàn cục, ranh giới mượt hơn (dựa trên trực quan hóa).
- **U-Net**: Giữ chi tiết cục bộ tốt hơn, phù hợp với dữ liệu y tế nhỏ.

- **Tổng quát hóa**:
  - DeepLabV3 có dấu hiệu dao động ở epoch 9-10, có thể do overfitting trên dữ liệu nhỏ.
  - U-Net ổn định hơn, phù hợp với tập dữ liệu hạn chế (1087 mẫu).

---

5. Trực Quan Hóa Kết Quả

- **Tập test**: 5 ảnh ngẫu nhiên được dự đoán bởi cả hai mô hình.
- **Kết quả** (dựa trên notebook):
  - **DeepLabV3**: Ranh giới dự đoán mượt mà, phù hợp với các vùng lớn.
  - **U-Net**: Chi tiết cục bộ rõ nét hơn, nhưng có thể bỏ sót một số vùng phức tạp.

---

6. Phân Tích

- **Phân tích**:
  - DeepLabV3 vượt trội về hiệu suất (Dice, IoU, F1) nhờ khả năng nắm bắt ngữ cảnh đa tỷ lệ, nhưng thời gian huấn luyện dài và dễ dao động trên dữ liệu nhỏ.
  - U-Net nhanh, ổn định, và phù hợp hơn cho bài toán y tế với dữ liệu hạn chế.
  - Dữ liệu nhỏ (1087 mẫu) có thể gây overfitting, đặc biệt với DeepLabV3.

---

7. Kết Luận

- Cả **DeepLabV3** và **U-Net** đều hoạt động tốt trên bộ dữ liệu y tế, với DeepLabV3 đạt hiệu suất cao hơn (Dice 0.8989) nhưng U-Net nhanh và ổn định hơn.
- Kết quả cho thấy tầm quan trọng của việc chọn mô hình phù hợp với kích thước dữ liệu và yêu cầu bài toán.
- Cần cải tiến thêm về dữ liệu và huấn luyện để đạt hiệu suất tối ưu.

---

8. Tài Liệu Tham Khảo

1. Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *MICCAI*.
2. Chen, L. C., et al. (2017). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE TPAMI*.
3. Chen, L. C., et al. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *ECCV*.
4. Kaggle Dataset: /kaggle/input/2425-ii-ait-3002-medical-image-segmentation/Dataset.