



تحلیل هوشمند تصاویر زیست پزشکی

نیم‌سال اول ۰۳-۰۲

مدرس: محمدحسین رهبان

مهلت ارسال: ۲۴ آذر

تمرین چهارم

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص شده است.
- در طول ترم امکان ارسال با تاخیر تمرین‌ها بدون کسر نمره تا سقف ۱۲ روز وجود دارد. محل بارگزاری جواب تمرین‌ها بعد از ۴ روز بسته خواهد شد و پس از گذشت این مدت، پاسخ‌های ارسال شده پذیرفته نخواهند شد.
- توجه داشته باشید که نوت‌بوک‌های شما باید قابلیت بازاجرای ۱۰۰ درصد داشته باشند و در صورت نیاز به نصب یک کتابخانه یا دسترسی به یک فایل، مراحل نصب و دانلود (از یک محل عمومی) در نوت‌بوک وجود داشته باشد.
- هم‌فکری در انجام تمرین مانعی ندارد، فقط توجه داشته باشید که پاسخ تمرین حتماً باید توسط خود شخص نوشته شده باشد. همچنین در صورت هم‌فکری در هر تمرین، در ابتدای جواب تمرین نام افرادی که با آن‌ها هم‌فکری کرده اید را حتماً ذکر کنید.
- برای پاسخ به سوالات نظری در صورتی که از برگه خود عکس تهیه می‌کنید، حتماً توجه داشته باشید که تصویر کاملاً واضح و خوانا باشد. در صورتی که خوانایی کافی را نداشته باشد، تصحیح نخواهد شد.
- محل بارگذاری سوالات نظری و عملی در هر تمرین مجزا خواهد بود. به منظور بارگذاری بایستی تمرین تئوری در یک فایل زیپ با نام `ABI_Theo_hw4_[First-Name]_[Last-Name]_[Student-Id].zip` و تمرین عملی نیز در یک فایل مجزای زیپ با نام `ABI_Prac_hw4_[First-Name]_[Last-Name]_[Student-Id].zip` بارگذاری شوند.
- در صورت وجود هرگونه ابهام یا مشکل، در کوثرای درس آن مشکل را بیان کنید و از پیغام دادن مستقیم به دستیاران آموزشی خودداری کنید.

بخش نظری (۳۰ نمره)

سوال اول (۱۰ نمره)

یک گراف $G = (V, E)$ را در نظر بگیرید، با ویژگی‌های گره $x(v)$ برای هر $v \in V$. برای هر گره $v \in V$ ، فرض کنید $h_v^{(0)} = x(v)$ نمایه اولیه گره باشد. در هر تکرار k ، نمایه‌ها به صورت زیر به‌روزرسانی می‌شوند:

$$h_{N(v)}^{(k)} = \text{AGGREGATE}(\{h_u^{(k-1)}, \forall u \in N(v)\})$$
$$h_v^{(k)} = \text{COMBINE}(h_v^{(k-1)}, h_{N(v)}^{(k)}),$$

توجه کنید که آرگومان تابع $\text{AGGREGATE}(\cdot)$ ، $h_u^{(k-1)}, \forall u \in N(v)$ ، یک مولتی‌ست است. یعنی، از آنجا که چندین گره می‌توانند نمایه یکسانی داشته باشند، یک عنصر می‌تواند چندین بار در $h_u^{(k-1)}, \forall u \in N(v)$ رخ دهد. در نهایت، خود یک گراف می‌تواند با محاسبه برخی توابع اعمال شده بر روی مولتی‌ست تمام نمایه‌های گره در یک تکرار نهایی K ، که ما آن را به صورت زیر نمایش می‌دهیم، نمایه‌بندی شود:

$$\text{READOUT}(\{h_v^{(K)}, \forall v \in V\})$$

ما می‌خواهیم از نمایه‌های گراف فوق برای آزمایش اینکه آیا دو گراف $G_1 = (V_1, E_1)$ و $G_2 = (V_2, E_2)$ ایزومورف هستند یا خیر، استفاده کنیم. به یاد داشته باشید که این صحیح است اگر و فقط اگر یک تناظر یک به یک $\phi : V_1 \rightarrow V_2$ بین گره‌های G_1 و گره‌های G_2 وجود داشته باشد به طوری که برای هر $u, v \in V_1$

$$(u, v) \in E_1 \Leftrightarrow (\phi(u), \phi(v)) \in E_2$$

روشی که ما از مدل فوق برای آزمایش ایزومورفی استفاده می‌کنیم به شرح زیر است. برای دو گراف، اگر توابع خواندن آن‌ها متفاوت باشد، یعنی

$$\text{READOUT}(\{h_v^{(K)}, \forall v \in V_1\}) \neq \text{READOUT}(\{h_v^{(K)}, \forall v \in V_2\}),$$

ما نتیجه می‌گیریم که گراف‌ها ایزومورف نیستند. در غیر این صورت، ما نتیجه می‌گیریم که گراف‌ها ایزومورف هستند. توجه داشته باشید که این الگوریتم کامل نیست و ما به بررسی قدرت این نمایه‌های گراف خواهیم پرداخت.

۱.۰ انتخاب تابع تجميع

انتخاب تابع $\text{AGGREGATE}(\cdot)$ برای قدرت مدل فوق مهم است. سه انتخاب رایج عبارتند از:

$$\text{AGGREGATE}_{\max}(\{h_u^{(k-1)}, \forall u \in \mathcal{N}(v)\})_i = \max_{u \in \mathcal{N}(v)} (h_u^{(k-1)})_i \text{ (حداکثر تک‌عنصری)}$$

$$\text{AGGREGATE}_{\text{mean}}(\{h_u^{(k-1)}, \forall u \in \mathcal{N}(v)\}) = \frac{1}{|\mathcal{N}(v)|} \sum_{u \in \mathcal{N}(v)} (h_u^{(k-1)})$$

$$\text{AGGREGATE}_{\text{sum}}(\{h_u^{(k-1)}, \forall u \in \mathcal{N}(v)\}) = \sum_{u \in \mathcal{N}(v)} (h_u^{(k-1)})$$

مثالی از دو گراف $G_1 = (V_1, E_1)$ و $G_2 = (V_2, E_2)$ و ویژگی‌های اولیه گره‌های آن‌ها ارائه دهید، به گونه‌ای که برای برخی گره $v_1 \in V_1$ و برخی گره $v_2 \in V_2$ با ویژگی‌های اولیه یکسان $h_{v_1}^{(0)} = h_{v_2}^{(0)}$ ، ویژگی‌های به‌روزرسانی شده $h_{v_1}^{(1)}$ و $h_{v_2}^{(1)}$ در صورت استفاده از تجميع میانگین و حداکثر یکسان باشند، اما در صورت استفاده از تجميع جمع متفاوت باشند.

راهنمایی: ویژگی‌های گره‌های شما می‌توانند اسکالر به جای بردار باشند، یعنی ویژگی‌های گره تک‌بعدی به جای چندبعدی. همچنین، شما آزاد هستید تا تعداد گره‌ها (مثلاً ۳ گره)، اتصالات آن‌ها (یعنی یال‌های بین گره‌ها) را در مثال خود به دلخواه انتخاب کنید.

۲.۰ آزمون وایسفایلر-لهمن

الگوریتم آزمایش ایزومورفی ما حداکثر به اندازه آزمون معروف وایسفایلر-لهمن (WL-Test) قدرتمند است. در هر تکرار، این الگوریتم نمایه هر گره را به مجموعه‌ای که شامل نمایه قبلی خود گره و نمایه‌های قبلی تمام همسایه‌هایش است، به‌روزرسانی می‌کند. الگوریتم کامل در شکل بعدی آمده است.

ثابت کنید که مدل عصبی ما حداکثر به اندازه آزمون WL قدرتمند است. به طور دقیق‌تر، فرض کنید G_1 و G_2 غیر ایزومورف هستند و فرض کنید که نمایه‌های گره‌های آن‌ها در طی K تکرار با همان توابع $\text{AGGREGATE}(\cdot)$ و $\text{COMBINE}(\cdot)$ به‌روزرسانی می‌شوند. نشان دهید که اگر رابطه زیر برقرار باشد آنگاه آزمون WL نیز تصمیم می‌گیرد که گراف‌ها ایزومورف نیستند.

$$\text{READOUT}(\{h_v^{(K)}, \forall v \in V_1\}) \neq \text{READOUT}(\{h_v^{(K)}, \forall v \in V_2\})$$

Algorithm 3: Weisfeiler-Lehman Test

Data: $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$, initial features $x(\cdot)$, number of iterations K

Result: Prediction of whether G_1 and G_2 are isomorphic

```
for  $v \in V_1 \cup V_2$  do
  |  $l_v^{(0)} \leftarrow x(v)$ 
end
for  $k = 1, \dots, K$  do
  | for  $v \in V_1 \cup V_2$  do
  | |  $l_v^{(k)} \leftarrow \text{HASH} \left( l_v^{(k-1)}, \{l_u^{(k-1)}, \forall u \in \mathcal{N}(v)\} \right)$ 
  | end
end
return  $\{l_v^{(K)}, \forall v \in V_1\} = \{l_v^{(K)}, \forall v \in V_2\}$ 
```

توجه: اثبات باید برای هر تابع AGGREGATE، COMBINE، READOUT عمومی باشد. به عبارت دیگر، کافی نیست که این موضوع را فقط برای یک نمونه خاص از مدل GNN نشان دهید. **راهنمایی:** می‌توانید از اثبات توسط تناقض استفاده کنید با این فرض که آزمون وایسفلر-لهمن نمی‌تواند در پایان تکرار K تصمیم بگیرد که آیا G_1 و G_2 ایزومورف هستند یا خیر.

سوال دوم (۱۰ نمره)

مقاله [Hierarchical Graph Representations in Digital Pathology](#) را بخوانید و به سوالات زیر پاسخ دهید.

۱. رویکرد نوآورانه‌ای که در مقاله در مورد استفاده از نمایش‌های گرافی سلسله مراتبی در پاتولوژی دیجیتال معرفی شده چیست و چگونه از روش‌های پیشین در تحلیل تصاویر پاتولوژیک متفاوت است؟

۲. در زمینه نمایش‌های گرافی سلول-به-بافت سلسله مراتبی (HACT) که در مقاله استفاده شده است، شبکه عصبی گرافی سلسله مراتبی (HACT-Net) چگونه تصاویر هیستولوژی ورودی را برای طبقه‌بندی ساختارهای بافتی را پردازش می‌کند؟

مقاله [SlideGraph⁺](#) در کلاس معرفی شده است. با توجه به آن به سوالات زیر پاسخ دهید.

۱. چگونه [SlideGraph⁺](#) از مفهوم پیش‌بینی‌های سطح گره در معماری شبکه خود استفاده می‌کند و این رویکرد چه مزایایی نسبت به روش‌های سنتی تحلیل تصویر کلی اسلاید (WSI) دارد؟

۲. نقش مدل رگرسیون چگالی DAB در [SlideGraph⁺](#) چیست و چگونه این مدل پیش‌بینی وضعیت HER2 از تصاویر رنگ‌آمیزی شده H&E را بهبود می‌بخشد؟

منابع پیشنهادی برای مطالعه بیشتر

- [Graph Neural Networks and Their Current Applications in Bioinformatics](#)
- [Representation Learning of Histopathology Images Using Graph Neural Networks](#)
- [Visualization for Histopathology Images using Graph Convolutional Neural Networks](#)

سوال سوم (۱۰ نمره)

مدل VoxelMorph علی‌رغم پیشرفت‌هایش، در registration تصاویر پزشکی محدودیت‌هایی دارد. از جمله آن‌ها می‌توان به محدودیت آن در یادگیری ویژگی‌های سطح بالا و ارائه‌هایی deformation با حفظ توپولوژی اشاره کرد. در سال‌های اخیر مدل‌های مبتنی بر ترنسفورمر برای وظیفه رجیستریشن تصاویر پزشکی ارائه شده‌اند، از جمله:

- ViT-V-Net: Vision Transformer for Unsupervised Volumetric Medical Image Registration
- TransMorph: Transformer for Unsupervised Medical Image Registration

با بررسی این دو مقاله، به سوالات زیر پاسخ دهید:

- هر کدام چه محدودیتی از VoxelMorph را حل کرده‌اند؟ راه حل ارائه شده آن‌ها را مختصراً توضیح دهید.
- معماری این دو مدل را بررسی کرده و شباهت‌ها و تفاوت‌های آن‌ها با مدل VoxelMorph را بیان کنید.

Multiple Instance Learning (۲۵ نمره)

در این سوال، پیاده‌سازی و آموزش یک مدل ارائه شده بر مبنای Multiple Instance Learning بر روی یک مجموعه داده از تصاویر هیستوپاتولوژی را خواهیم داشت. اطلاعات مربوط به مقاله ارائه دهنده این مدل و نحوه دریافت مجموعه داده در ابتدای نوت‌بوک آورده شده است، اما برای دسترسی سریع‌تر به مقاله اصلی می‌توانید از لینک زیر استفاده کنید:

[لینک مربوط به مقاله](#)

در ادامه چند نکته مهم درباره انجام این سوال را مرور خواهیم کرد که در نوت‌بوک نیز به آن‌ها اشاره شده است:

- در انجام تمرین می‌توانید هر تغییری که لازم دارید در نوت‌بوک و کد از پیش نوشته شده‌ی آن اعمال کنید و لزومی به پیاده‌سازی هر بخش در سلول مشخص شده آن نیست. اما برای راحت‌تر شدن نمره‌دهی تمرین، حتماً محل پیاده‌سازی هر بخش را به صورت کامنت در سلول از پیش مشخص شده برای آن تعیین کنید.
- برای اطلاعات بیشتر درباره مدل ارائه شده، مقاله دیگری نیز در نوت‌بوک معرفی شده است که می‌توانید به آن مراجعه کنید. ولی با توجه به پیچیده و وقت‌گیر بودن، مطالعه دقیق آن پیشنهاد نمی‌شود.
- نحوه دقیق استفاده از transformation در نوت‌بوک ذکر شده است. برای گرفتن بهترین نتیجه لازم است مطابق توضیح داده شده از آن استفاده کنید.
- در نوت‌بوک خواهید دید که تغییرات زیر در روش ارائه شده توسط مقاله ایجاد شده است که لازم است به آن‌ها توجه کنید:

- تغییر در نحوه تقسیم کردن مجموعه داده
- تغییر جزئی در لایه آخر معماری ارائه شده
- تغییر در نحوه استفاده از داده برای آموزش مدل در هر اپیک (استفاده از ۲۰۰ تصویر به جای تمام تصاویر مجموعه آموزش) و مشخص کردن حداقل اپیک لازم

Graph Neural Networks (۲۵ نمره)

هدف از این تمرین، کسب تجربه عملی در استفاده از شبکه‌های عصبی گرافی (GNNs) در تحلیل تصاویر پزشکی، به خصوص تصاویر پاتولوژی است. این تمرین کل فرآیند از دانلود و پیش‌پردازش داده‌ها تا پیاده‌سازی و آزمایش مدل را شامل می‌شود. در ادامه شرح مختصری از کارهایی که باید انجام دهید آورده شده است. با توجه به آن‌ها نوت‌بوک داده شده را تکمیل کنید.

مراحل:

۱. دانلود مجموعه داده‌ها: مجموعه داده مورد نظر را با استفاده از اسکریپت کوتاهی که در نوت‌بوک قرار داده شده است دانلود کرده و توضیحات مربوط به آن را مطالعه نمایید.
۲. پیش‌پردازش داده‌ها: تابع `images_preprocess` را برای آماده‌سازی تصاویر برای پردازش پیاده‌سازی کنید.
۳. استخراج هسته‌ها: تابع `nuclei_extracting` را برای شناسایی مکان و ابعاد هسته‌ها در تصاویر توسعه دهید.
۴. پیاده‌سازی مدل پایه:

- یک مدل GCN را با استفاده از کتابخانه‌های PyTorch و Geometric PyTorch پیاده‌سازی کنید.
- معماری GCN شامل لایه‌های convolutional و batch normalization را تعریف کنید.
- تابع forward را بر اساس یک نمودار معماری فراهم شده در نوت بوک پیاده‌سازی کنید.

۵. مدل پیش‌بینی گراف:

- یک مدل پیش‌بینی گراف GCN را با استفاده از جاسازی‌های گره از مدل GCN و global pooling برای ایجاد امبدینگ سطح گراف ایجاد کنید.
- توابع آموزش و ارزیابی را با استفاده از قالب‌های فراهم شده تعریف کنید.
- از global pooling و یک لایه خطی خروجی برای پیش‌بینی ویژگی‌های گراف استفاده کنید.

۶. آزمایش‌ها: آزمایش‌هایی را برای بررسی اثربخشی مدل انجام دهید و با لایه‌های global pooling مختلف آزمایش کرده و تغییرات عملکرد مدل را مشاهده کنید.

۷. بصری‌سازی: برای بررسی دقت و کیفیت مدل پیاده‌سازی شده نتایج را نمایش دهید. همچنین برای درک بهتر فرایندها پس از پیش پردازش و استخراج هسته‌ها، نتایج خود را به صورت تصویر نشان داده و اجزای آنرا کمی توضیح دهید.

دستورالعمل‌ها:

- تمام قسمت‌های نوت بوک را کامل کنید.
- در حالی که تغییرات لازم را ایجاد می‌کنید، ساختار و اهداف کلی تمرین را دنبال کنید.
- اطمینان حاصل کنید که پس از وارد کردن کد خود، هر سلول اجرا شده باشد.
- از کتابخانه‌های مرتبط استفاده کرده و توضیحات مختصری برای انتخاب‌های خود ارائه دهید.

(۲۰ نمره) Image Registration via VoxelMorph

ثبت تصویر یا image registration چندین تصویر از یک شی یا صحنه با زاویه دیدهای مختلف و ثبت‌شده در زمان‌های متفاوت را تراز می‌کند. این فرآیند امکان مقایسه دقیق، تجزیه و تحلیل و یکپارچه سازی اطلاعات تصویر را فراهم می‌کند.

در این تمرین لازم است فایل HW4_Image_Registration.ipynb را تکمیل کنید. در این نوت‌بوک تسک image registration با استفاده از مدل VoxelMorph بر روی مجموعه دادگانی شامل عکس شبکه چشم انجام می‌گیرد. دستورالعمل‌ها را دنبال کنید، کد را تکمیل کرده و خروجی بگیرید.