

به نام خدا



دانشگاه تهران
دانشکده فنی
دانشکده مهندسی برق و کامپیوتر



درس بازیابی هوشمند اطلاعات

تمرین ۳

استاد درس: خانم دکتر آزاده شاکری

سرپرست دستیاران آموزشی: سمانه پیمانی‌راد

طراح تمرین: فائزه مظفری

آذر ماه ۱۴۰۴

| | |
|---------|--|
| ۳..... | مقدمه |
| ۴..... | سوالات عملی (۵۵ نمره) |
| ۵..... | مجموعه داده |
| ۶..... | سوال ۱: پیاده سازی الگوریتم های رتبه بندی اینترنت (۳۰ نمره) |
| ۸..... | سوال ۲: پیاده سازی TF-IDF با استفاده از MapReduce (۲۵ نمره) |
| ۹..... | سوالات تئوری (۴۵ نمره) |
| ۱۰..... | سوال ۱: الگوریتم Inverted Index در MapReduce (۱۰ نمره) |
| ۱۱..... | سوال ۲: محاسبه الگوریتم ها رتبه بندی PageRank و HITS (۲۰ نمره) |
| ۱۲..... | سوال ۳: ساخت یک فهرست با استفاده از MapReduce (۱۵ نمره) |
| ۱۳..... | ملاحظات (حتما مطالعه شود) |
| ۱۵..... | استفاده مسئولانه از هوش مصنوعی |
| ۱۵..... | ۱. هدف و اصول کلی |
| ۱۵..... | ۲. استفاده مجاز از LLMها |
| ۱۶..... | ۳. استفاده غیرمجاز از LLMها |
| ۱۶..... | ۴. مستندسازی |
| ۱۶..... | ۵. آمادگی ارائه شفاهی |
| ۱۶..... | ۶. پیامدهای تخلفات |
| ۱۶..... | ۷. موارد تکمیلی |
| ۱۷..... | ۸. اظهارنامه |

در این تمرین، با الگوریتم‌های معروف رتبه‌بندی اینترنت آشنا خواهید شد. این الگوریتم‌ها ابزارهایی موثر برای تحلیل رفتار کاربران و رتبه‌بندی آن‌ها در شبکه هستند. این روش‌ها به شما کمک خواهند کرد تا نقش هر کاربر را در شبکه‌های اجتماعی و ارتباطات آن‌ها تحلیل کنید.

از طرفی با توجه به رشد روزافزون حجم اطلاعات و محدود بودن زمان خواندن این اطلاعات، نیاز به روش‌های کارآمد برای پردازش داده‌ها بیشتر از همیشه احساس می‌شود. یکی از این روش‌ها استفاده از معماری MapReduce است که به کمک آن می‌توان به صورت موازی و با سرعت بالا، حجم‌های بسیار زیاد داده‌ها را پردازش کرد.

اهداف اصلی تمرین:

- پیاده‌سازی الگوریتم‌های PageRank، HITS، SALSA و Topic-Sensitive PageRank
- یادگیری و استفاده از معماری MapReduce در پردازش داده‌های متنی

نکات قابل توجه در هنگام پاسخ به سؤالات:

- تمامی کدهای نوشته شده باید قابلیت اجرای مجدد داشته باشند. به تفسیرهایی که بدون آزمایش و صرفاً به صورت فرضی بیان گردند نمره‌ای تعلق نمی‌گیرد.
- نمره اصلی تمرین مربوط به کیفیت و درستی تفسیرهای ارائه شده است.
- بهتر است از **نمودارها** و کشف نمونه‌های مرتبط از اسناد و پرس‌وجوها برای افزایش کیفیت تفسیرها استفاده گردد.
- بدیهی است که حجم تمرین معیار نمره‌ی شما نیست، بلکه صحت انجام آزمایش‌ها و کیفیت تفسیرهای شما مهم است.
- توصیه می‌شود انجام تمرین را به روزهای آخر موکول نکنید! **اجرای آزمایش‌ها و پیدا کردن پارامتر بهینه زمان بر بوده** و تفسیر هر کدام نیازمند تحلیل و بررسی است.

سوالات عملی (۵۵ نمره)

برای پیاده‌سازی سوال اول از کتابخانه NetworkX و برای سوال دوم از کتابخانه MRJob استفاده کنید. توصیه می‌شود که مستندات این دو کتابخانه را مطالعه کنید. همچنین استفاده از کتابخانه‌های مرسوم مانند NLTK نیز مجاز می‌باشد.

در سوال دوم، قبل از پیاده‌سازی الگوریتم‌های خواسته شده، پیش پردازش‌های لازم بر روی متن مجموعه داده را انجام دهید و سپس بعد از آن به پیاده‌سازی سوالات خواسته شده بپردازید.

پاسخ سوالات این بخش را در همان نوت‌بوک تمرین خود (*IIR-CA3-Code.ipynb*) بنویسید.

در صورتی که امکان اجرای کد روی سیستم خود را ندارید می‌توانید از Google Colab استفاده کنید (نیازی به اجرا بر روی GPU نیست و در حالت CPU می‌توانید کدهای خود را اجرا کنید). برای بارگذاری فایل دادگان در گوگل کولب می‌توانید از لینک گوگل درایو قرارداده شده در سامانه ایلرن استفاده کنید.

مجموعه داده

مجموعه داده سوال اول Wikipedia Vote Network می باشد که از [این لینک](#) قابل دسترس می باشد. این مجموعه داده شامل اطلاعات مربوط به رای گیری کاربران ویکی پدیا در انتخابات مدیران است. این مجموعه داده شامل ۷,۱۱۵ گره (کاربر) و ۱۰۳,۶۸۹ یال جهت دار (رای) می باشد. در این شبکه، هر گره نشان دهنده ی یک کاربر ویکی پدیا است و هر یال جهت دار از گره i به گره j نشان می دهد که کاربر i به کاربر j برای مدیر شدن رای داده است.

مجموعه داده سوال دوم، فایل متنی کتاب “Alice’s Adventures in Wonderland” می باشد که در مسیر زیر در اختیار شما قرار گرفته شده است.

./data/ Alice’s Adventures in Wonderland.txt

سوال ۱: پیاده سازی الگوریتم های رتبه بندی اینترنت (۳۰ نمره)

در این سوال با توجه به مجموعه داده Wikipedia Vote Network، رفتار کاربران و تعامل آن ها در انتخابات مدیران ویکی پدیا را تحلیل کنید.

الف) گرافی مبتنی بر این مجموعه داده بسازید که گره ها کاربران و یال ها ارتباط بین گره ها باشد. توزیع درجه ورودی و خروجی را محاسبه کنید و تحلیل کنید که یک کاربر با in-degree و out-degree بالا چه نقشی دارد؟

ب) الگوریتم PageRank را با هایپر پارامتر $\alpha = 0.5$ و 0.3 و 0.15 و 0.05 پیاده سازی کنید. در هر تکرار بردار PageRank را به روزرسانی کرده و این فرآیند را تا زمانی ادامه دهید که اختلاف بین دو بردار متوالی کم تر از آستانه 0.001 باشد. تعداد تکرار برای هر هایپر پارامتر را گزارش کنید.

ج) الگوریتم HITS را پیاده سازی کنید. در هر تکرار امتیاز های Authority و Hub را به روز رسانی کنید و این فرآیند را تا زمانی ادامه بدهید که تغییر بین دو بردار متوالی کم تر از آستانه 0.001 باشد. تعداد تکرار را گزارش کنید. امتیاز های Authority در الگوریتم HITS را با ده کاربر برتر PageRank مقایسه کنید. نتایج را تحلیل کنید. همچنین امتیاز های Hub در HITS را با ده کاربر دارای بیش ترین out-degree مقایسه کنید. نتایج به دست آمده را تحلیل کنید.

د) الگوریتم SALSA را پیاده سازی کنید. در هر تکرار امتیاز های Authority و Hub را به روز رسانی کنید و این فرآیند را تا زمانی ادامه بدهید که تغییر بین دو بردار متوالی کم تر از آستانه 0.001 باشد. تعداد تکرار را گزارش کنید. نتایج به دست آمده را با HITS مقایسه کنید. در ادامه الگوریتم SALSA را روی یک زیرگراف با گره های با in-degree بیشتر از ۱۰ پیاده سازی کنید و امتیاز های Authority و Hub برترین گره ها را با برترین گره ها روی گراف کامل مقایسه کنید. چگونه برترین گره ها هنگام اعمال SALSA به زیرگراف نسبت به گراف کامل تغییر می کنند؟

ه) دو Topic زیر را در نظر بگیرید.

- Topic 1 : کاربران با ۵٪ out-degree بالا
- Topic 2 : کاربران با ۵٪ in-degree بالا

سپس الگوریتم Topic-Sensitive PageRank را پیاده سازی کنید. این فرآیند را تکرار کنید تا تغییر بین دو بردار متوالی کم تر از آستانه 0.001 باشد. تعداد تکرار را برای هر موضوع گزارش کنید. ۱۰ کاربر برتر برای

هر موضوع را به دست آورید و نتایج را تحلیل کنید. ۱۰ کاربر برتر برای هر موضوع را با کاربران برتر برای الگوریتم PageRank را مقایسه کنید و تحلیل کنید.

سوال ۲: پیاده سازی TF-IDF با استفاده از MAPREDUCE (۲۵ نمره)

ابتدا پیش پردازش های لازم بر روی متن داده شده را انجام بدهید و سپس به سوالات زیر پاسخ دهید.

(الف) متن کتابی که در اختیار شما قرار داده شده است شامل ۱۲ فصل می باشد، آن را به ۱۲ فایل مجزا، برای هر فصل یک فایل، تقسیم کنید. این ۱۲ فایل سند های شما می باشد.

(ب) در ابتدا با استفاده از MapReduce فراوانی کلمات، IDF را محاسبه کنید و در نهایت مقدار امتیاز نهایی TF-IDF را محاسبه کرده که خروجی باید به صورت $(word, doc_id, tf_idf_score)$ باشد. وزن های TF-IDF را برای این کلمات را با استفاده از فرمول های زیر TF و IDF محاسبه کنید:

$$TF(t, d) = \frac{\text{Frequency of term } t \text{ in document } d}{\text{Total number of words in document } d}$$

$$IDF(t) = \ln\left(\frac{N}{df(t)}\right)$$

که در آن N تعداد کل سند ها و $df(t)$ تعداد اسنادی است که شامل کلمه t می باشد.

(ج) در تمامی هر ۱۲ فصل ۱۰ کلمه ای که بالاترین امتیاز TF-IDF را دارد را استخراج کنید. این ۱۰ کلمه و امتیازات آن ها را لیست کنید. آیا این کلمات از نظر موضوعی به این فصل ها مرتبط هستند؟

(د) کلمه alice را در خروجی خود پیدا کنید و برای این کلمه در تمامی فصل ها مقدار TF، DF، IDF و TF-IDF آن را محاسبه کنید. مقادیر به دست آمده را تحلیل کنید.

(ه) کلمه ای پیدا کنید که دارای TF بالایی باشد، سپس مقدار امتیاز TF-IDF آن را محاسبه کنید و بگویید چرا مقدار TF-IDF آن پائین می باشد. همچنین کلمه ای را پیدا کنید که TF پائینی دارد و مقدار TF-IDF آن را محاسبه کنید، توضیح دهید که این مقادیر به دست آمده چرا بدین گونه می باشد و چرا این رفتار برای سیستم های بازیابی اطلاعات مطلوب هستند.

سوالات تئوری (۴۵ نمره)

باتوجه به مباحث تدریس شده سوالات زیر را حل کنید. پاسخ سوالات این بخش باید در قالب مربوط به

سوالات تئوری (IIR-CA3-Theory) بنویسید.

نکات سوالات تئوری:

- صرفاً به جواب نهایی نمره تعلق نمی‌گیرد و راه‌حل نوشته شده بخش اصلی نمره‌ی شماست.
- می‌توانید جهت اطمینان از محاسبات خود از کدنویسی استفاده کنید، اما باید راه‌حل به طور کلی به صورت ریاضی در گزارش تمرین آمده باشد و صرفاً قراردادن کد راه‌حل باعث کسر نمره شما می‌شود.

سوال ۱: الگوریتم INVERTED INDEX در MAPREDUCE (۱۰ نمره)

شبه کد زیر برای یک الگوریتم پایه‌ی Inverted Index در MapReduce می‌باشد:

```
1: class Mapper
2: procedure Map(docid n, doc d)
3:   H ← new AssociativeArray
4:   for all term t ∈ doc d do
5:     H{t} ← H{t} + 1
6:   for all term t ∈ H do
7:     Emit(term t, posting <n, H{t}>))

1: class Reducer
2: procedure Reduce(term t, postings <[n1, f1], <n2, f2> ...>))
3:   P ← new List
4:   for all posting <a, f> ∈ postings <[n1, f1], <n2, f2> ...> do
5:     Append(P, <a, f>)
6:   Sort(P)
7:   Emit(term t, postings P)
```

این الگوریتم برای مجموعه‌های کوچک به درستی کار می‌کند، اما دارای یک scalability bottleneck می‌باشد. کلاس Mapper، posting ها را با کلید term منتشر می‌کند. سپس MapReduce تمام posting های مربوط به یک کلمه واحد را گروه‌بندی کرده و کل آن لیست مرتب‌نشده را به Reduce ارسال می‌کند. مشکل در خط ۶ کلاس Reducer است. Reducer ابتدا باید تمام posting های مربوط به یک کلمه را در لیستی به نام P در حافظه بافر کند و تنها پس از آن می‌تواند تابع Sort(P) را برای مرتب‌سازی آن‌ها بر اساس document ID فراخوانی کند.

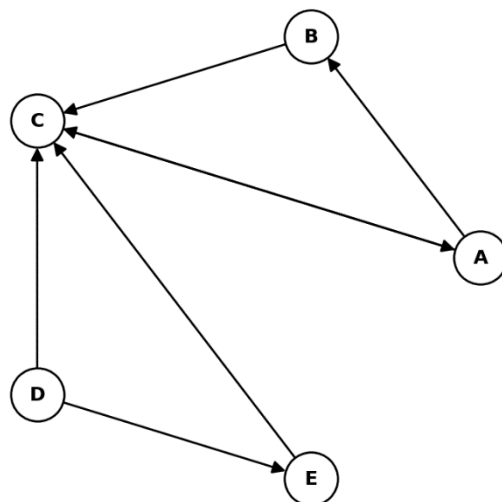
برای یک مجموعه بسیار بزرگ، یک کلمه رایج میلیون‌ها posting خواهد داشت. Reduce تلاش می‌کند تا این لیست عظیم را در حافظه بافر کند، که باعث تمام شدن حافظه و fail آن می‌شود.

الف) تغییرات مورد نیاز در جفت key-value میانی توسط Mapper باید ایجاد کنید تا این مشکل را حل کنید. این تغییر را اعمال کنید و توضیح دهید چگونه این تغییر مشکل را حل می‌کند و نیاز به Sort(P) در Reducer را از بین می‌برد.

ب) با توجه به تغییرات بالا شبه کد جدیدی را بنویسید.

سوال ۲: محاسبه الگوریتم‌ها رتبه بندی PAGERANK و HITS (۲۰ نمره)

شبکه ساده زیر را در نظر بگیرید.



الف) امتیاز هر گره را با الگوریتم PageRank تا ۲ تکرار برای هایپر پارامتر $\alpha = 0.15$ محاسبه کنید.

ب) امتیاز هر گره را با الگوریتم PageRank تا ۲ تکرار برای هایپر پارامتر $\alpha = 0$ محاسبه کنید.

ج) امتیاز Hub و Authority را برای هر گره با الگوریتم HITS تا ۲ تکرار محاسبه کنید.

د) امتیازات نهایی PageRank از بخش الف و ب را مقایسه کنید. برای گره‌های D و E، در مورد نقش damping factor چه تحلیلی می‌توانید داشته باشید؟

ه) امتیازات نهایی Authority از بخش ج را با امتیازات PageRank از بخش الف مقایسه کنید. آیا گره دارای بیش‌ترین Authority در HITS با گره دارای بالاترین رتبه در PageRank مطابقت دارد؟

سوال ۳: ساخت یک فهرست با استفاده از MAPREDUCE (۱۵ نمره)

در این سوال از شما می خواهیم تا یک فهرست PageRank-Weighted Positional Index با استفاده از MapReduce بسازید. این فهرست به موتور جستجو اجازه می دهد تا نتایج را نه تنها بر اساس ارتباط، بلکه بر اساس اعتبار PageRank نیز رتبه بندی کند.

مجموعه داده های ورودی به صورت زیر می باشد:

شما دو منبع ورودی بر اساس گراف تمرین ۲ (A, B, C, D, E) دارید.

1. فایل امتیازات PageRank

این فایل شامل امتیازات نهایی PageRank از تمرین ۲، بخش الف با تکرار ۲ و هایپر پارامتر $\alpha = 0.15$ است.

```
(A, PageRank_Score)
(B, PageRank_Score)
(C, PageRank_Score)
(D, PageRank_Score)
(E, PageRank_Score)
```

2. مجموعه محتوای ۵ سند، متناظر با گره های A-E است.

A: "search retrieval HITS"

B: "web search graph"

C: "HITS algorithm graph retrieval"

D: "web link analysis"

E: "link graph search"

خروجی نهایی شما باید یک فهرست جدید باشد که این دو ورودی را به هم متصل کند. Value برای هر کلید باید یک لیست واحد از postings ها باشد. هر postings باید شامل (DocID, PageRank_Score, [positions]) باشد و لیست باید به ترتیب نزولی بر اساس PageRank_Score مرتب شده باشد.

• این مسئله به دو شبه کد نیاز دارد.

تمامی نتایج شما باید در یک فایل فشرده با عنوان IIR-CA3-StudentID تحویل داده شود.

- خوانایی و دقت بررسی‌ها در گزارش نهایی از اهمیت ویژه‌ای برخوردار است. به تمرین‌هایی که به صورت کاغذی تحویل داده شوند یا به صورت عکس در سایت بارگذاری شوند، ترتیب اثری داده نخواهد شد. **دقت کنید که حتما گزارشات خود را در قالب ارائه شده برای تحویل تکالیف که در سامانه برای شما بارگذاری شده است ارسال بفرمایید.**

- کدهای نوشته شده برای هر بخش را با نام مناسب مشخص کرده و به همراه گزارش تکلیف ارسال کنید. همه‌ی کدهای پیوست گزارش بایستی قابلیت اجرای مجدد داشته باشند. در صورتی که برای اجرا مجدد آن‌ها نیاز به تنظیمات خاصی می‌باشد بایستی تنظیمات مورد نیاز را نیز در گزارش خود ذکر کنید. **دقت کنید که تمامی کدها باید توسط شما اجرا شده باشند و نتایج اجرا در فایل کدهای ارسالی مشخص باشد. به کدهایی که نتایج اجرای آن‌ها در فایل ارسالی مشخص نباشد نمره‌ای تعلق نمی‌گیرد.**

- تمرین تا یک هفته بعد از مهلت تعیین شده با تاخیر تحویل گرفته می‌شود. دقت کنید که شما جمعا برای تمام تکالیف، ۱۴ روز زمان تحویل بدون جریمه دارید که تنها از ۷ روز آن برای هر تمرین می‌توانید استفاده کنید، در صورتی که این ۱۴ روز به اتمام رسیده باشد، به ازای هر روز تأخیر در ارسال تمرین، ده درصد جریمه می‌شوید.

- **توجه کنید این تمرین باید به صورت تک نفره انجام شود و پاسخ‌های ارائه شده باید نتیجه فعالیت فرد نویسنده باشد (همفکری و به اتفاق هم نوشتن تمرین نیز ممنوع است).** در صورت مشاهده تشابه به همه افراد مشارکت کننده، نمره ۵۰- تعلق می‌گیرد و به استاد نیز گزارش می‌گردد.

- در صورت بروز هرگونه مشکل با ایمیل زیر در ارتباط باشید:

<mailto:faezehxmozaffari@gmail.com>

| | |
|-------------|---------------------------------------|
| ۲ آذر ۱۴۰۴ | تاریخ آپلود تمرین |
| ۱۲ آذر ۱۴۰۴ | مهلت تحویل بدون جریمه |
| ۱۹ آذر ۱۴۰۴ | مهلت تحویل با تأخیر، با جریمه ۱۰ درصد |

استفاده مسئولانه از هوش مصنوعی

۱. هدف و اصول کلی

هدف

- ترویج استفاده اخلاقی و مسئولانه از LLMها (مانند ChatGPT، Deepseek) به عنوان ابزار کمکی
- اطمینان از مشارکت فعال دانشجویان در تکالیف و درک راه‌حل‌های آن‌ها
- حفظ صداقت علمی در عین بهره‌گیری از ابزارهای مدرن هوش مصنوعی

اصول کلی

- تمرین باید نتیجه تلاش و زحمت شخصی شما باشد.
- باید به تمام بخش‌های تمرین، اعم از پیاده‌سازی و تحلیل نتایج مسلط باشید.
- تمامی کدها باید توسط خود شما اجرا شده و نتایج قابل مشاهده باشند.
- تمام مراحل انجام تمرین باید مستند و قابل پیگیری باشد.
- هرگونه نتیجه‌گیری و تحلیل باید بر اساس درک شخصی شما باشد.
- LLMها ممکن است پاسخ‌های نادرست یا قدیمی تولید کنند، اولویت با مطالب و کارگاه‌های درس است.

موارد ذکر شده در ادامه این سند، به عنوان راهنمایی بیشتر برای انجام تمرین آورده شده‌اند. با این حال، مسئولیت تطبیق کار با اصول کلی فوق بر عهده شماست. توجه داشته باشید که ممکن است مواردی در ادامه ذکر نشده باشند که با اصول کلی ذکر شده در تضاد باشند. در چنین مواردی به تشخیص دستیار آموزشی و دستیار مسئول، شما موظف به پاسخ‌گویی در قبال تمرین خود هستید. عدم رعایت هر یک از اصول فوق می‌تواند منجر به کسر نمره یا عدم پذیرش تمرین شود.

۲. استفاده مجاز از LLMها

شما می‌توانید از LLMها برای موارد زیر استفاده کنید:

- روشن‌سازی مفاهیم (مثال: "خوشه‌بندی DBSCAN چگونه کار می‌کند؟")
- کمک در اشکال‌زدایی (مثال: شناسایی خطاهای گرامری یا منطقی در کد)
- ایده‌پردازی رویکردها (مثال: "روش‌های مدیریت داده‌های missing را پیشنهاد دهید")

الزامات استفاده مجاز:

- ثبت تعاملات اصلی: (به بخش ۴ مراجعه کنید).
- درک راه‌حل: باید قادر به توضیح هر خط کد یا منطق استفاده شده باشید.

۳. استفاده غیرمجاز از LLMها

اقدامات ممنوع شامل:

- کپی-پیست مستقیم خروجی‌های LLM بدون تغییر
- استفاده از LLMها برای حل اصلی مسائل (مثال: "این سؤال تکلیف را برای من حل کن")
- گرفتن کد از سایر دانشجویان به هر شکل غیر مجاز است، تغییر و پارافریز کردن کد دیگران توسط LLM نیز قابل قبول نیست.
- هرگونه استفاده که منجر به عدم احاطه شما به موضوع تمرین شود.

۴. مستندسازی

ارجاع به مشارکت‌های LLM: افزودن پانویس یا توضیح (مثال: کد با رعایت قوانین به کمک ChatGPT نوشته شده است).

- نیازی به اشتراک گذاری پرامپت‌ها و سابقه چت نیست.
- مستندسازی تک تک تعاملات با هوش مصنوعی هدف این بخش نیست. اشاره کوتاه و کلی در بخش‌های مورد استفاده کافی است. در نظر داشته باشید که مستندسازی به معنای رفع مسئولیت نبوده و باید اصول کلی را رعایت کنید.

۵. آمادگی ارائه شفاهی

آماده دفاع از کار خود باشید: در صورت درخواست دستیار تمرین در بازه زمانی اعلام شده برای ارائه شفاهی، باید:

- رویکرد، کد یا نتایج خود را توضیح دهید.
- درک مفاهیم کلیدی را نشان دهید (مثلاً چرا یک الگوریتم خاص انتخاب شده است)
- عدم توضیح کافی کار شما ممکن است منجر به جریمه شود (بخش ۶)

۶. پیامدهای تخلفات

- تخلفات جزئی (مثل مستندسازی ناقص): کاهش نمره
- تخلفات عمده (مثل کپی-پیست بدون تغییر): نمره ۵۰- در تکلیف
- تخلفات مکرر: نمره ۵۰- در تکلیف و گزارش به استاد

۷. موارد تکمیلی

- از LLMها به عنوان معلم استفاده کنید، نه پاسخنامه تمرین‌ها: اولویت را به مهارت‌های حل مسئله خود بدهید.
- خروجی‌ها را متقابلاً تأیید کنید: پیشنهادات LLM را با کتاب مرجع درس، اسلایدها و کارگاه‌ها مقایسه کنید.
- از دستیاران آموزشی کمک بگیرید: اگر پاسخ LLM یا نحوه استفاده شما را گیج می‌کند، در ساعات متعارف از دستیاران آموزشی کمک بگیرید.

۸. اظهارنامه

این عبارت را در تکلیف ارسالی خود قرار دهید:

"تأیید می‌کنم که از LLMها مطابق با دستورالعمل‌های بارگذاری شده در سامانه Elearn درس به طور مسئولانه استفاده کرده‌ام. تمام اجزای کار خود را درک می‌کنم و آماده بحث شفاهی درباره آنها هستم."