# پردازش زبان‌های طبیعی

نیم‌سال دوم ۱۴۰۳-۱۴۰۴

مدرس: دکتر عسگری

دانشکده مهندسی کامپیوتر

گزارش تمرین سری سوم، $RAG$

نام و نام‌خانوادگی: سینا بیرامی، سینا دانشگر، الهه ظهیری

شماره دانشجویی: ۴۰۰۱۰۵۴۳۳، ۴۰۰۱۰۳۵۱۶، ۴۰۱۱۰۰۳۶۸

## ۱ پیش‌پردازش داده‌ها

برای ساخت دیتا برای هر فرد به عکس و متن خلاصهٔ مربوطه نیاز داشتیم. در ابتدا تنها یکی از فایل‌های ورزشی که رکوردهای بیشتری داشت انتخاب و متون تمرین دو را به دیتای تمرین اول چسباندیم. پس از ساخت امبدینگ‌ها و دیدن نتایج که بسیار ضعیف بود به مرحلهٔ قبل برگشتیم و ابتدا دیتا را سازماندهی کردیم. برای لینک عکس‌هایی که عکس بی‌کیفیت بود، وجود نداشت یا مناسب نبود پروسهٔ کراول کردن را دوباره انجام دادیم و دیتای نسبتا کاملی از کل مشاهیر ورزشی تهیه کردیم. سپس اول از متون و عکس‌های مربوطه امبدینگ ساختیم اما با توجه به اینکه برای مشاهیر متون بیوگرافی هستند و اینفورمیشنی برای عکس ندارند باعث شد دقت مدل بسیار پایین باشد، پس تصمیم گرفتیم به جای متون تنها اسم‌ها را امبد کنیم و دوباره مراحل را انجام دهیم اما باز هم دقت بسیار پایین و نزدیک انتخاب تصادفی بود. در نتیجه اینگونه عمل کردیم که برای هر کوئری چه عکس و چه متن ابتدا نزدیک‌ترین دیتا به آن را پیدا کرده و سپس از مپینگ بین عکس و اسم افراد استفاده کنیم که باعث شد دقت مدل بسیار بهتر شود که در ادامه تحلیل و گزارش آن‌ها تشریح شده‌است.

## ۲ مدل بازیاب

پایپ‌لاین کلی مدل رتریوال (بازیاب) به شرح زیر بود:
۱) سوال ۴ گزینه‌ای به مدل داده می‌شود
۲) مدل بازیاب، $Top_k$ رکوردهایی که امبدینگ متن خلاصهٔ $(summary)$ آن‌ها، به امبدینگ کوئری (سوال) نزدیک‌تر است را، برمی‌گرداند.
۳) یک فایل گزارش (به فرمت $csv$.) از $Top_3$ نتایج مدل بازیاب برای هر سوال ساخته می‌شود.
پس از آن، این خروجی‌ها به مدل جنریتیو داده می‌شود تا پاسخ نهایی را تولید کند.
برای این بخش برای هر دسته از سوالات یک تابع تعریف کردیم به اینصورت که برای پرسش‌های متنی، کوئری به صورت متن به مدل داده می‌شود، سپس با استفاده از $cosine\ similarity$ نزدیک‌ترین متن‌ها به کوئری انتخاب شده و خروجی داده می‌شود و برای استفاده در مراحل بعدی به همراه کوئری به $LLM$ داده می‌شوند. برای حالت مالتی مودال نیز سوال به همراه چهار گزینه و لینک عکس به تابع داده شده و ابتدا برای هر گزینه نزدیک‌ترین متن و عکس متناظر آن‌ها استخراج می‌شود، سپس تصویر ورودی دانلود شده، امبد شده و با استفاده از $cosine\ similarity$ از بین ۱۲ عکس استخراج‌شده نزدیک‌ترین داده به همراه سوال برای استفاده در مراحل بعدی خروجی داده می‌شوند.

## ۳ طراحی کانتکست برای مدل مولد

کانتکستی که پس از بازیابی ساختیم و به مدل $gpt-4o$ دادیم به حالت زیر بود: یک سوال چهار گزینه‌ای به تو داده می‌شود، به همراه ۳ عدد متن خلاصه $(summary)$ که از مدل بازیاب گرفتیم، و نقش $(role)$ تو این است که تشخیص بدهی بین این $Top_3$ خروجی داده‌شده، کدام‌یک مربوط به سوال (کوئری) است. (زیرا $Top_3$ ها متون خلاصه‌شده از سه شخص متفاوت هستند و تنها یکی از آن‌ها جواب قطعی می‌باشد)
اسم آن کسی که در مورد آن، سوال شده را برگردان و در این کانتکست یک شاهد $(evidence)$ بیاور که طبق آن تضمین کرده‌ای این جواب، برای آن شخص مذکور است.

سپس از خروجی جواب $gpt-4o$، در عکس‌ها (همان داده‌های جفت که در مرحله پیش‌پردازش درست کردیم) جستجو می‌کنیم و در نهایت، خروجی شامل اسم آن شخص $(target)$، شاهد $(evidence)$ ای که از کانتکست استخراج شده که جواب از آن بوده، و عکس متناظر آن فرد نیز چاپ می‌شود و درنهایت داخل یک فایل به فرمت .csv ذخیره می‌شوند.
برای سوالات تست مالتی مدال (عکس + متن) به این صورت عمل کردیم که مثل مرحلهٔ قبل، با دادن کوئری به مدل بازیاب و استخراج $Top_3$، صرفا به جای جستجو برای خلاصهٔ متون، عکس‌های متناظر هر شخص در کوئری داده‌شده (سوال چند گزینه‌ای) گرفته و ذخیره می‌شود و سپس این عکس‌ها به همراه سوال، در کانتکست مناسب به مدل $gpt-4o$ داده می‌شوند تا با توجه به آن، جواب مناسب را تولید و بازگردانی کند.

## ۴ مدل مولد

در بخش جنریشن، پس از بازیابی نتایج $Top_K$ (که در این پروژه $k=3$ در نظر گرفته شد)، متون و متادیتای مربوط به هر نتیجه (استخراج‌شده از فایل `parquet`.) جمع‌آوری شدند. این نتایج به عنوان کانتکست به مدل زبانی $(GPT-4o)$ ارسال شدند تا پاسخ نهایی تولید شود.
فرآیند جنریشن شامل دو خروجی بود:
۱) تشخیص اینکه کدام‌یک از اسناد بازیابی‌شده بیشترین ارتباط را با کوئری دارد
۲) استخراج یک «شاهد» (evidence) از همان کانتکست مرتبط که توضیح دهد چرا پاسخ انتخاب‌شده معتبر است.
سپس نام فرد شناسایی‌شده از کانتکست استخراج می‌شد و برای بازیابی تصویر متناظر او در دیتاست استفاده می‌گردید؛ در نهایت پاسخ نهایی همراه با تصویر و شاهد ارائه می‌شد.
در حالت مالتی‌مدال، اگر پرسش به‌طور مستقیم درباره تصویر بود (مثلاً: «این عکس متعلق به چه کسی است؟»)، مدل زبانی تنها وظیفه داشت پاسخ را در قالبی از پیش تعیین‌شده ارائه دهد، به‌صورت «این عکس متعلق به X است و بیوگرافی Y را دارد». در موارد دیگر، ابتدا با استفاده از امبدینگ تصویر، متنی از میان نتایج بازیابی انتخاب می‌شد که بیشترین شباهت به تصویر داشت؛ سپس متن خلاصهٔ متناظر به مدل زبانی داده می‌شد تا مشابه مرحلهٔ متنی، پاسخ نهایی تولید گردد.

## ۵ تحلیل نتایج

- آیا خروجی‌ها در حالت تصویری، متنی، یا ترکیبی تفاوت معناداری دارند؟

  در بخش متنی، خروجی‌ها بسیار دقیق‌تر بودند و به اطلاعات بیوگرافی ورزشکار مربوط می‌شدند. (چون متن مستقیم با خلاصه‌ها ایندکس شده بود) در بخش تصویری، خروجی‌ها بیشتر به شباهت ظاهری ورزشکاران (عکس‌ها) متکی بود و گاهی ربط ضعیف‌تری به محتوای بیوگرافی داشت. در قسمت ترکیبی (multimodal) که بهترین نتایج معمولا اینجا به‌دست می‌آمد، چون متن <-> تصویر مکمل یکدیگر بودند. این ترکیب، کمک می‌کرد که اگر تصویر گمراه‌کننده بود، متن مسیر درست را طی کند و همینطور برعکس.

- آیا بازیابی موفق و مرتبط انجام شده؟

  در بخش رتریوال، روی معیار $Hit@K$ برای سوالات چهارگزینه‌ای متنی، به $Hit@1 = 97.9\%$ و $Hit@3 = 100\%$ رسیدیم.

```
================================================================
MCQ RETRIEVAL EVALUATION (Hit@k)
================================================================
Total questions              : 48
Questions w/o detectable entity in Q: 0
----------------------------------------------------------------
Entity Hit@k (did we retrieve the person?)
  hit@1 : 0.979
  hit@3 : 1.000
  hit@5 : 1.000
  hit@10: 1.000
----------------------------------------------------------------
Answer-string Hit@k (did top-k context contain the correct option?)
  hit@1 : 0.812
  hit@3 : 0.833
  hit@5 : 0.833
  hit@10: 0.833
================================================================
```

- نقش prompt ها یا تنظیمات مدل چه بوده است؟

پرامپت‌ها در بخش تولید پاسخ (بعد از بازیابی) تعیین‌کنندهٔ لحن و نوع (فرمت) خروجی بودند. برای بازیابی، تنظیمات شامل انتخاب مدل امبدینگ OpenCLIP) برای تصویر و SentenceTransformer برای متن) و نحوهٔ نرمال‌سازی داده‌ها بود. به دلیل مشکلاتی که لایبرری هضم (فارسی) داشت، سعی کردیم توابع داخل آن را به صورت دستی پیاده کنیم و آن را در پروژه، با کیفیت بسیار بالاتری بهبود بدهیم. پرامپت در RAG وظیفه داشت که اسناد بازیابی‌شده را به شکل یک متن یکپارچه برای پاسخ به سوال کاربر قالب‌بندی کند. در بعضی جاها نیز، پرامپت مشخص می‌کرد که مدل باید چه توضیحی برای خروجی مدنظر ارائه دهد (مثلا شاهد ارائه کند، یا ...)

- آیا خروجی تولیدشده مبتنی بر اسناد بازیابی‌شده بوده یا صرفاً حدس مدل بوده است؟

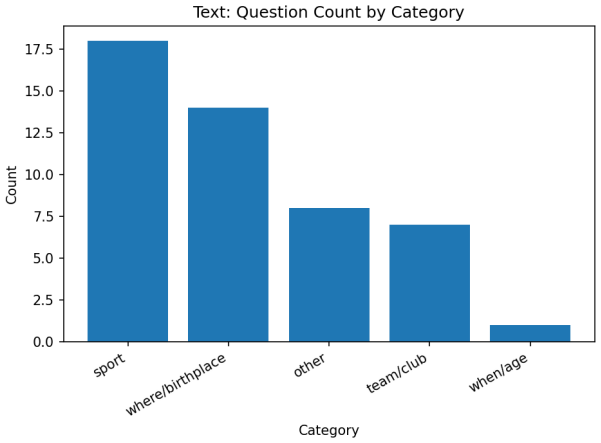
در مراحل ابتدایی کار، ما عکس و متن را باهمدیگر امبد می‌کردیم اما به دلیل محدودیت‌هایی که در دادگان داشتیم (مثل متون ضعیفی که از خروجی تمرین قبل گروه‌های دیگر در اختیارمان قرار داده بودند) مدل بازیاب، رفتار تصادفی از خود نشان می‌داد و وقتی که نتایج را محض تست به مدل مولد متن دادیم، دقتی حدود ٪۴۷ دریافت کردیم که قابل اتکا نبود. برای راه‌حل این مشکل، به جای متن (چون مستقل از عکس است) فقط اسم فرد را امبد کردیم. اما در نهایت دوباره با امبدینگ متن کار می‌کردیم و چون به‌طور موازی روی کیفیت و کمیّت داده‌ها کار می‌کردیم، نتایج به مرور بهتر شدند و مدل مولد نیز دقیق‌تر می‌شد. ایدهٔ نهایی‌ای که روی مدل رتریوال (بازیاب) زدیم نیز این بود که ابتدا بین متون شباهت می‌گرفتیم، و چون یک *lookup table* در مرحلهٔ اول درست کرده بودیم، بین نزدیکترین آن‌ها، شباهت عکس را بررسی می‌کردیم (از تناظر عکس <-> متن استفاده می‌بردیم)

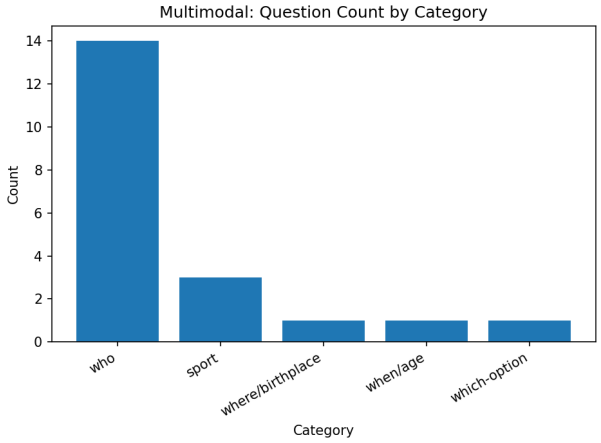
- چه چالش‌هایی در زمینه زبان، ساختار سوال، یا محتوای چندرسانه‌ای مشاهده شده است؟

در مورد چالش زبانی، بعضی رکوردها چندزبانه یا نامنظم بودند (مثلا در بیوگرافی، کلمات انگلیسی وجود داشت یا اینکه کاراکترهای خاصی باعث به‌هم‌ریختگی متن می‌شدند) و این موضوع روی امبدینگ متنی تاثیر گذاشته‌بود. در مورد ساختار سوال، با دسته‌بندی سوالات به حوزه‌های مختلف، سعی کردیم دقت را روی آن‌ها بررسی کنیم تا ببینیم که مدل دقیقا در چه حیطه‌ای بهتر عمل می‌کند و نیاز داریم که برای بخش پروژه، در چه حیطه‌هایی قدرت مدل را بهبود بدهیم. در زمینهٔ محتوای چندرسانه‌ای، یک مشکل بزرگ کمبود داده‌ها بودند به شکلی که مقداری از داده‌ها، یا کلا عکس نداشتند، یا اینکه عکس نامرتبطی با آن‌ها جفت شده بود (مثلا پرچم ایران، مدال برنز، یا...) همینطور کیفیت پایین بعضی عکس‌های ورزشکاران، باعث بازیابی ضعیف مدل می‌شد. در ترکیب متن و تصویر، وزن‌دهی به هر مدالیتی مهم بود، گاهی متن غالب می‌شد و تصویر کلا نادیده گرفته می‌شد، که این مورد هم نتایج رتریوال را خراب می‌کرد. یک چالش دیگر نیز هماهنگ‌سازی بین امبدینگ‌ها (متن <-> تصویر) بود که چون scale یا space هر یک متفاوت بود، سعی میکردیم با انتخاب مدل‌های دیگر، به نتایج بهتری برسیم.

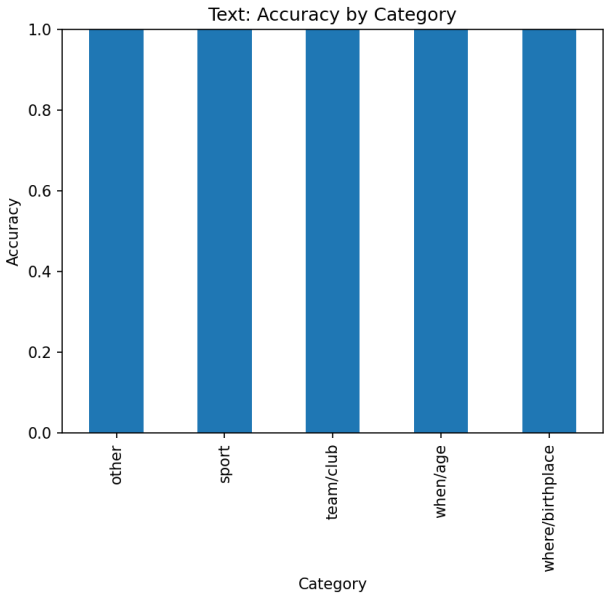
# ۶ داده‌های آماری

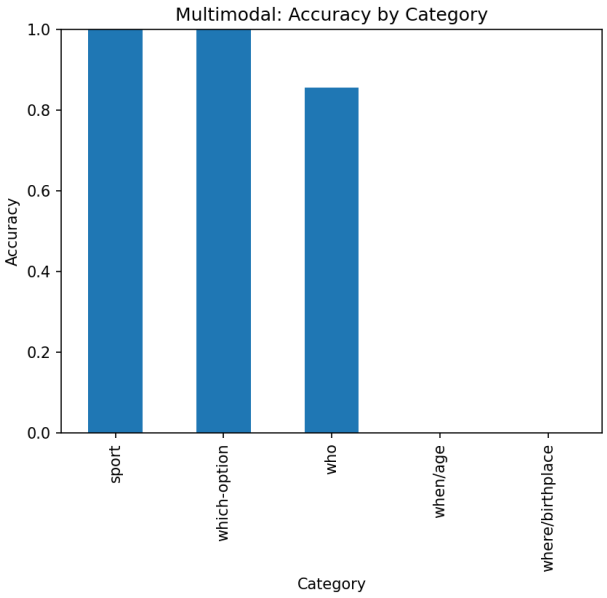
توزیع سوالات برای هر دسته از سوالات و هر کتگوری را در نمودار زیر مشاهده می‌کنید:



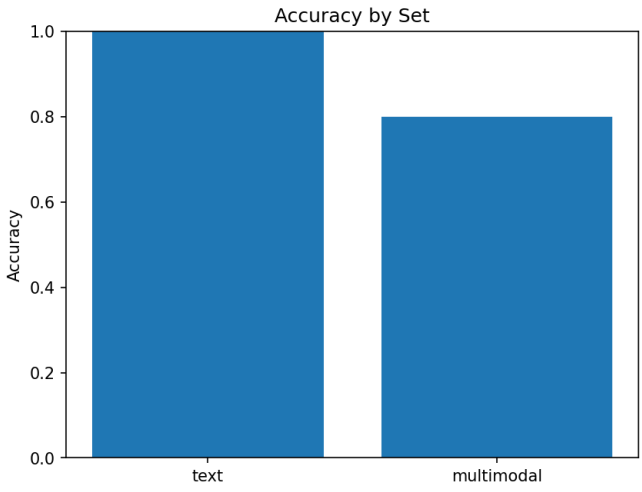




دقت مدل برای پاسخ به هر کتگوری در دسته‌های مالتی مودال و متن:




نتایج نهایی:



مقایسهٔ عملکرد مدل بدون RAG و مدل RAG :