به نام خالق رنگین کمان

ستاره باباجانی – گزارش تمرین سری 4

سوال 1: به محاسبه احتمال جملات خواسته شده میپردازیم: (مجموع کلمات پیکره 9521 میباشد.)

محاسبه احتمال یونیگرم ها به شرح زیر است:

$$p\left(\text{ما}\right)=\frac{1872}{9521}, p\left(\text{خواندیم}\right)=\frac{1495}{9521}, p\left(\text{دیروز}\right)=\frac{2021}{9521},$$
$$p\left(\text{امروز}\right)=\frac{1943}{9521}, p\left(\text{داستان}\right)=\frac{945}{9521}, p\left(\text{کتاب}\right)=\frac{1245}{9521}$$

حال اگر در جدول بایگرم ها فرض کنیم کلمه اول در ستون سمت راست و کلمه بعدی آن در ردیف بالای آن قرار داشته باشد، خواهیم داشت:

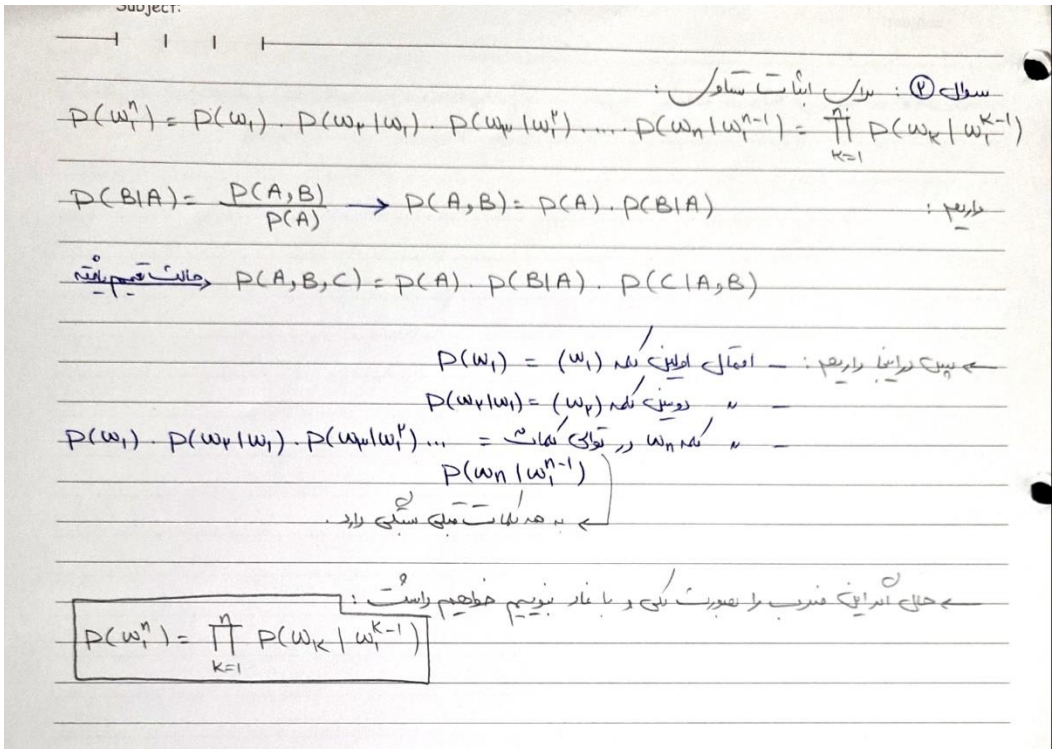
- احتمال رخداد جمله تست 1:

$$p\left(\text{ما امروز کتاب خواندیم}.\right)=p\left(\text{ما}\mid\text{امروز}\right).p\left(\text{امروز}\mid\text{کتاب}\right).p\left(\text{کتاب}\mid\text{خواندیم}\right)$$
$$=\frac{452}{1872}*\frac{231}{1943}*\frac{320}{1245}=0.0074$$

- احتمال رخداد جمله تست 2:

$$p\left(\text{ما دیروز داستان خواندیم}.\right)=$$
$$p\left(\text{ما}\mid\text{دیروز}\right).p\left(\text{دیروز}\mid\text{داستان}\right).p\left(\text{داستان}\mid\text{خواندیم}\right)=\frac{411}{1872}*\frac{68}{2021}*\frac{345}{945}=0.0027$$

سوال 2:



سوال 3:

a) در مرحله اول با توجه به k = 2، هر دو کلمه را نگه میداریم:

- “neural” with score = -0.65
- “network” with score = -0.73

b) در این مرحله خواهیم داشت:

- For “neural” with the score of -0.65:
  - Expand to “neural neural”: total score will be -1.45
  - Expand to “neural network”: total score will be -1.25
- For “network” with the score of -0.73:
  - Expand to “network neural”: total score will be -1.33
  - Expand to “network network”: total score will be -1.53

حال با توجه به k = 2 دوتا از بهترین نتایج مرحله قبل را حفظ میکنیم:

- “neural network” with score = -1.25
- “network neural” with score = -1.33

c) حال دوباره هر یک از نود های مرحله قبل را گسترش میدهیم:

- For “neural network” with the score of -1.25:
  - Expand to “neural network neural”: total score will be -2.05
  - Expand to “neural network network”: total score will be -1.85
- For “network neural” with the score of -1.33:
  - Expand to “network neural neural”: total score will be -2.13
  - Expand to “network neural network”: total score will be -1.93

حال با توجه به k = 2 دوتا از بهترین نتایج مرحله قبل را حفظ میکنیم:

- “neural network network” with score = -1.85
- “network neural network” with score = -1.93

d) خیر. همان طور که میدانیم در این مثال بهترین دنباله ها ”neural neural neural“ و ”network network network“ به ترتیب با امتیاز 1.46- , 1.54- هستند که beam search نتوانسته آنها را پیدا کند.

این الگوریتم لزوما دنباله کلی با بیشترین احتمال را برنمیگرداند بلکه بهترین را در میان تعداد محدودی از دنباله هایی که ارزیابی میکند، پیدا میکند. همچنین این الگوریتم با استفاده از پارامتر k تعادلی بین کارایی محاسباتی و کیفیت جواب نهایی برقرار میسازد.(هرچه k بیشتر، راه حل دقیق تر ولی کندتر و بالعکس.)

e) همان طور که میدانیم در هر مرحله زمانی t برای یک دنباله k دنباله هایی که در حال حاضر ذخیره شده اند، RNN امتیازات را برای همه ی m کلمه بعدی محاسبه کرده(k*m فراخوانی) و سپس بهترین k دنباله از این k*m تا برای مرحله محل انتخاب میشوند. پس از آنجایی که که این فرآیند برای هر یک از t مرحله دنباله تکرار میشود، پیچیدگی زمانی اجرای کلی آن بصورت O(t*k*m) خواهد بود.

سوال 4:

a) اگر فقط گیت forget داشته باشیم، LSTM فقط می تواند تصمیم بگیرد که کدام بخش از وضعیت سلول قبلی را حفظ کند، اما نمی تواند کنترل کند که چه اطلاعات جدیدی به وضعیت سلول اضافه می شود.

- گیت input: گیت ورودی کنترل می کند که چه مقدار از اطلاعات جدید (از ورودی فعلی و حالت پنهان قبلی) به حالت سلول اضافه می شود. بدون آن، LSTM توانایی خود را برای افزودن اطلاعات جدید به حالت سلول از دست می دهد. این بدان معنی است که وضعیت سلول فقط می تواند در طول زمان کاهش یابد یا ثابت بماند، زیرا اطلاعات فقط فراموش می شوند، هرگز به روز نمی شوند یا با ورودی های جدید تکمیل نمی شوند.
- گیت ouput: گیت خروجی کنترل می کند که چه مقدار از حالت سلول برای ایجاد حالت پنهان فعلی (خروجی) استفاده می شود. بدون آن، کل حالت سلول همیشه به عنوان خروجی در معرض دید قرار می‌گیرد، که

می‌تواند منجر به مسائل نفوذ بیش از حد از حالت سلول و عدم کنترل بر روی اینکه چه بخشی از حالت داخلی در معرض لایه‌ها یا خروجی‌های بعدی قرار می‌گیرد، شود.

- تغییرات خروجی: LSTM احتمالاً در طول زمان توانایی کمتری در یادگیری الگوهای پیچیده پیدا می کند زیرا نمی تواند اطلاعات جدید را در ورودی ها جمع آوری کند.

b) گیت forget در یک LSTM میزان حفظ حالت سلول قبلی (از آخرین مرحله زمانی) را در وضعیت سلول فعلی کنترل می کند. این کار را با اعمال یک ضرب گیت (بین 0 و 1) در حالت سلول قبلی انجام می دهد.

- بدون گیت forget: وقتی گیت فراموشی روی صفر تنظیم می شود، به طور موثری از حفظ هرگونه اطلاعات از وضعیت سلول قبلی در وضعیت سلول فعلی جلوگیری می کند. این بدان معناست که LSTM هر آنچه را که قبلاً می دانست در هر مرحله زمانی فراموش می کند و از انتقال هر گونه اطلاعات به گام های زمانی جلوگیری می کند.
- تاثیر بر یادگیری و پیش بینی: باعث از دست دادن وابستگی های طولانی مدت میشود. یکی از نقاط قوت اولیه LSTM ها توانایی آنها در به خاطر سپردن اطلاعات در دوره های طولانی است. صفر کردن گیت فراموشی این قابلیت را حذف می کند و در نتیجه توانایی LSTM را برای استفاده از بافت تاریخی یا وابستگی هایی که فراتر از یک مرحله زمانی هستند فلج می کند.

c)

- قابلیت های یادگیری پیشرفته: افزودن لایه های LSTM بیشتر به شبکه اجازه می دهد تا بازنمایی های پیچیده تری را بیاموزد. هر لایه می تواند سطح متفاوتی از انتزاع را بیاموزد. لایه های بیشتر به طور کلی ظرفیت شبکه را افزایش می دهد و به آن اجازه می دهد تا روابط و وابستگی های پیچیده تری را در داده ها مدل کند.
- بهبود درک متنی: شبکه های عمیق تر LSTM به طور بالقوه می توانند وابستگی های طولانی مدت را بهتر از شبکه های کم عمق تر جذب کنند، زیرا اطلاعات دارای مراحل بیشتری برای پردازش و پالایش است که هر یک سطحی از درک زمینه را اضافه می کند. چندین لایه LSTM می توانند اطلاعات را به صورت سلسله مراتبی پردازش کنند. لایه‌های پایین‌تر می‌توانند وابستگی‌های کوتاه‌مدت را مدیریت کنند، در حالی که لایه‌های بالاتر می‌توانند این وابستگی‌ها را در بازه‌های طولانی‌تر ادغام کنند، بنابراین درک کلی توالی را بهبود می‌بخشند.

پس بطور خلاصه، افزایش تعداد لایه‌های LSTM می‌تواند توانایی مدل را برای یادگیری الگوهای پیچیده و بهبود عملکرد در وظایفی که نیاز به درک وابستگی‌های بلندمدت دارند، افزایش دهد. با این حال، این امر با افزایش هزینه محاسباتی، پیچیدگی آموزش، و پتانسیل بیش از حد برازش همراه است.

سوال 5: قسمت های خواسته شده برای کد، در فایل ضمیمه شده، قرار دارد. حال برای قسمت i خواهیم داشت:

همانطور که میدانیم متغیر timestep تعریف شده، میزان توجه به گذشته را شرح میدهد. حال در اینجا به بررسی اثر افزایش و یا کاهش آن میپردازیم:

- افزایش timestep:
  - اطلاعات تاریخی(گذشته) بیشتر: با افزایش گام زمانی، مدل پنجره بزرگتری از قیمت های تاریخی را برای هر پیش بینی در نظر می گیرد. این بدان معناست که هر دنباله ورودی مدل حاوی نقاط داده گذشته بیشتری است.
  - پتانسیل برای درک بهتر روندها: با زمینه تاریخی بیشتر، مدل ممکن است روندهای بلندمدت و الگوهای فصلی قیمت بیت کوین را بهتر به تصویر بکشد، که می تواند برای پیش بینی قیمت های آینده بسیار مهم باشد.
  - افزایش پیچیدگی مدل و هزینه محاسباتی آن: مراحل زمانی بزرگتر می تواند به مدل های پیچیده تری منجر شود، زیرا مدل نیاز به یادگیری از زمینه وسیع تری دارد. این می تواند زمان آموزش و منابع محاسباتی مورد نیاز را افزایش دهد.
  - خطر تطبیق بیش از حد: اگر گام زمانی خیلی بزرگ باشد، این خطر وجود دارد که مدل با داده های تاریخی بیش از حد مطابقت داشته باشد، به خصوص اگر داده های آموزشی متنوع کافی در دسترس نباشد. برازش بیش از حد می تواند باعث شود که مدل در داده های آموزشی عملکرد خوبی داشته باشد اما در داده های دیده نشده ضعیف باشد.
- کاهش timestep:
  - داده های تاریخی(گذشته) کمتر: کاهش مرحله زمانی به این معنی است که مدل از نقاط داده تاریخی کمتری برای هر پیش بینی استفاده می کند. این مدل را بر روندهای اخیر متمرکز می کند.

- آموزش سریعتر مدل: با داده های کمتری برای پردازش برای هر پیش بینی، ممکن است مدل سریعتر آموزش ببیند و به منابع محاسباتی کمتری نیاز داشته باشد.
- پتانسیل برای عدم تناسب: اگر گام زمانی خیلی کوچک باشد، ممکن است مدل به اندازه کافی زمینه تاریخی را برای پیش بینی های دقیق ثبت نکند و به طور بالقوه منجر به عدم تناسب شود. این بدان معناست که مدل ممکن است برای ثبت الگوهای اساسی در داده ها بسیار ساده باشد.

انتخاب گام زمانی یک اقدام متعادل کننده است. ما باید مقداری را انتخاب کنیم که زمینه تاریخی کافی برای ثبت الگوهای مرتبط را بدون تحت تأثیر قرار دادن مدل یا تمرکز آن بر روی داده های نامربوط فراهم کند.