

تحلیل احساسات با شبکه‌های ^۱ RNN

حامد محمدزاده

(۱) چکیده

در این تمرین قصد داریم امتیاز نظر کاربران بر محصولات دسته بندی اشتراک مجلات ^۲ سایت آمازون را پیش بینی می‌شوند. برای اینکار مدل‌های RNN ، LSTM و GRU با تعداد لایه‌های میانی مختلف آموزش داده شده‌اند.

(۲) مقدمه

این دیتاست [۱] شامل حدود ۹۰ هزار نظر کاربران به صورت رشته است و هر نظر یک امتیاز از ۱ تا ۵ برای محصول مربوطه ثبت کرده. دیتاست مورد استفاده شده، علاوه بر نظر کاربران به صورت رشته و امتیاز آن‌ها به صورت عددی بین ۰ تا ۴، شامل تصاویری که کاربران در نظر خود ارسال کرده‌اند، شناسه کاربر و محصول، زمان نظر و ... هم می‌باشد.

(۳) پیش‌پردازش داده‌های متن

برای پیش‌بینی امتیاز کاربران فقط از رشته نظر آن‌ها استفاده می‌شود. در قدم اول داده‌هایی که متن نظر آن‌ها موجود نیست حذف می‌شود. برای تبدیل رشته‌های ورودی به عدد، ابتدا باید رشته‌ها tokenize شوند، برای اینکار از spacy استفاده می‌کنیم. سپس از یک لغت‌نامه ^۳ برای تبدیل کلمات به عدد استفاده می‌شود. در تمام آزمایشات، اندازه این لغت‌نامه ۴۰۰۰۰ در نظر گرفته شده‌است (در واقع ۴۰۰۰۰ توکن که بیشترین فراوانی را دارند در لغت‌نامه جای می‌گیرند) لازم به ذکر است اندازه بهینه این لغت‌نامه در آزمایشات جستجو نشده است و انتظار می‌رود و

Sentiment Analysis^۱
Magazine Subscriptions^۲
Vocabulary^۳

با انتخاب مقدار بهتر، تعمیم پذیری مدل بالا برود. برای کلماتی که در لغت نامه موجود نیستند از توکن <UNK> استفاده می شود.

نکته مهمی که در انجام آزمایشات رعایت شده، الهام گرفته از [این لینک](#) و [این پیاده سازی](#)، بجای اینکه تمامی جملات متن نظر (که ورودی مدل هستند و به عدد تبدیل شده اند) به یک اندازه pad شوند و sequence length همه ورودی ها یکسان باشد، هر چند تا batch از داده ها در کنار هم بر اساس طول متنشان مرتب شده، و آن جملاتی که در این چند batch از نظر طول رشته شبیه به هم هستند، در یک batch قرار می گیرند. این راه حل از چند نظر مفید است.

• از لحاظ محاسباتی بهینه تر است، به عنوان مثال بجای اینکه یک جمله با طول ۲ (به عنوان مثال: "great magazine") را به تا طول ۱۰۰ pad شود و محاسبات ۵۰ برابر شود، این جمله در کنار جملات با طول نزدیک به ۲ دیگر قرار می گیرد و همگی تا اندازه $p < 100$ pad می شوند و مقدار محاسبات به شدت کاهش می یابد.

• با این روش از خاصیت مدل های RNN که می توانند ورودی با طول متغیر بگیرند استفاده می شود. همچنین اطلاعات مفیدی که شبکه یاد می گیرد بیشتر می شود، به عنوان مثال فرض کنید متن نطری "great magazine" باشد. با padding بیش از حد، ورودی که شبکه می گیرد
$$\begin{bmatrix} \text{"great"} & \text{"magazine"} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$
 می شود اما با این روش شبکه ورودی
$$\begin{bmatrix} \text{"great"} & \text{"magazine"} \end{bmatrix}^T$$
 را می گیرد.

(۴) آزمایشات

سه مدل RNN، GRU و LSTM با یک، دو و سه لایه میانی آزمایش شده اند. در آزمایشات انجام شده ۱۰ درصد از داده ها برای تست، ۱۰ درصد برای validation و بقیه داده ها برای آموزش استفاده شده اند. از بهینه ساز Adam و batch size با مقدار ۶۴ برای یادگیری استفاده شده و نرخ یادگیری مربوط به هر آزمایش در جدول شماره ۱ آمده

است.

مدل	نرخ یادگیری
RNN	10^{-3}
GRU	10^{-3}
LSTM	5×10^{-3}

جدول ۱: مشخصات نرخ یادگیری مدل‌ها - این نرخ یادگیری برای تمام آزمایشات هر مدل است، یعنی Deep RNN با تعداد لایه‌های میانی ۲ و ۳ هم با همین نرخ یادگیری آموزش دیده‌اند

برای پیش‌بینی امتیاز نظرات، خروجی (Output) آخرین Cell هر مدل به یک لایه Fully Connected متصل شده و با اعمال Softmax و Cross Entropy Loss خروجی نهایی مدل محاسبه شده.

دقت شود که منظور از آخرین Cell هر مدل، آخرین Cell از نظر تعداد لایه‌های میانی است. همین‌طور در تمامی آزمایشات از اندازه لایه مخفی ۲۵۶ و اندازه Embedding ۱۲۸ استفاده شده. نتیجه دقت داده تست بعد از ۵ مرحله آموزش برای هر کدام از مدل‌ها در جدول شماره ۲ آمده است.

نام مدل	دقت داده تست
lstm<2>	70.6447
gru<2>	71.1912
rnn<2>	61.3875
rnn<1>	59.8706
gru<1>	71.3362
lstm<1>	70.6893
lstm<3>	69.7078
rnn<3>	11.6551
gru<3>	70.9458
rnn<3>	65.0457

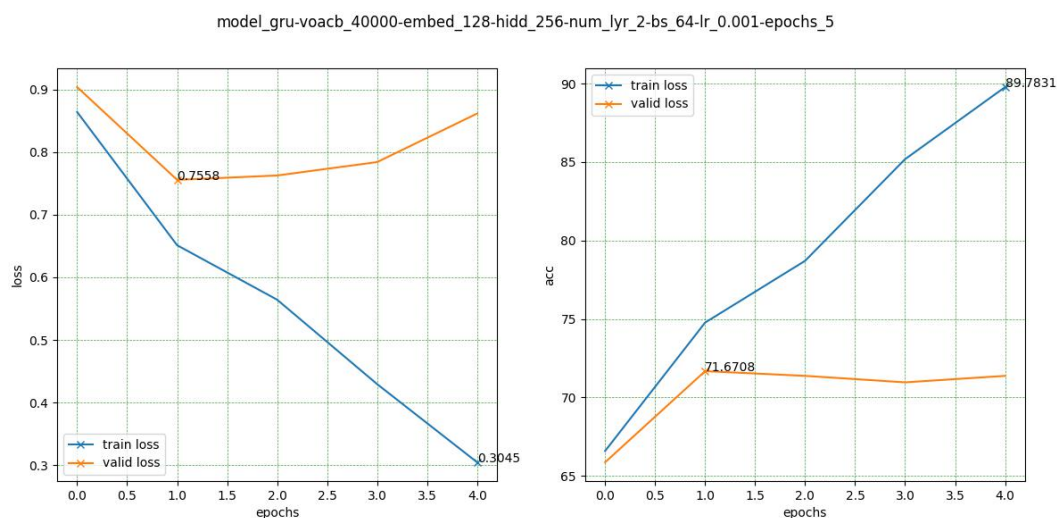
جدول ۲: نتیجه دقت داده تست برای مدل‌های مختلف، دقت کنید <K> Model Name به این معنی است که این مدل K لایه میانی داشته است

ابتدا در قسمت بعدی بهترین نتیجه هرکدام از مدل‌های RNN و LSTM و GRU را با هم مقایسه می‌کنیم و سپس تاثیر تعداد لایه‌های میانی در هرنوع را به تفکیک بررسی می‌کنیم.

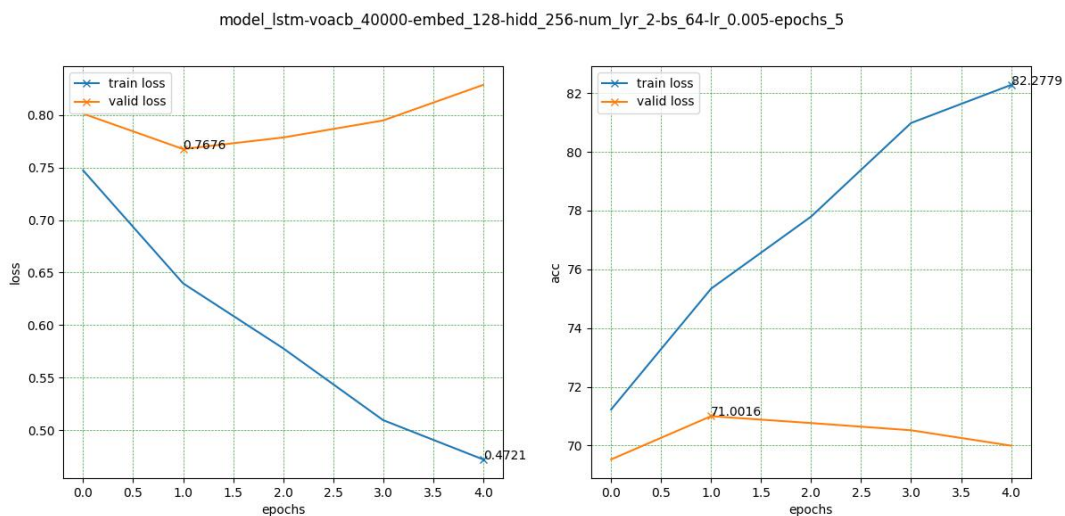
۱.۴ مقایسه نتایج RNN و LSTM و GRU

با مقایسه نتایج بهترین مدل‌های GRU، LSTM و RNN به ترتیب در شکل‌های ۱، ۲ و ۳ می‌توان به چندین نتیجه رسید:

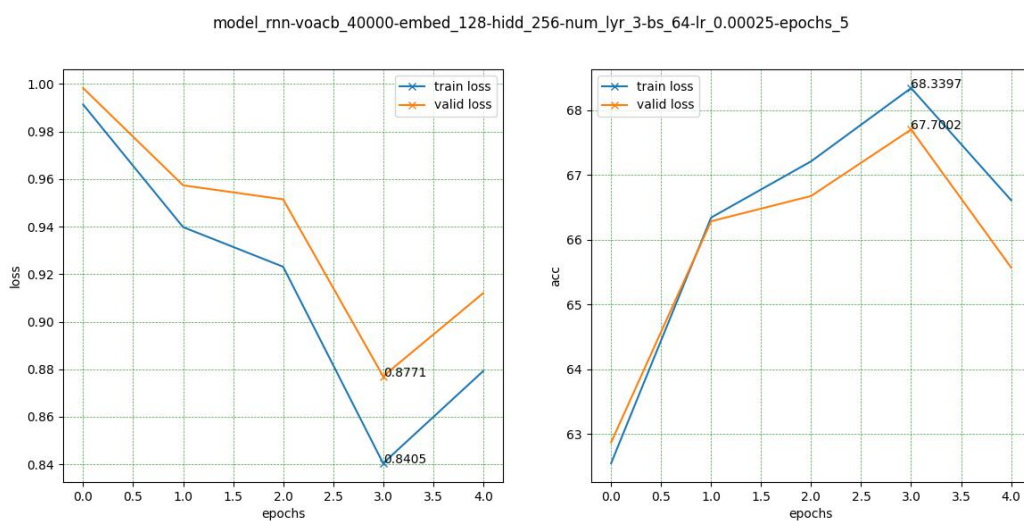
- در هر سه مدل به خصوص در LSTM و GRU مشکل Overfitting رخ داده، پارامترهای مختلف سائز لغت‌نامه، اندازه لایه‌های مخفی و ... تست شد اما این مشکل حل نشد، حدس زده می‌شود مشکل از کم بودن داده آموزشی باشد.
- عملکرد مدل‌های LSTM و GRU نسبت به RNN بهتر است، و بین این دو GRU کمی بهتر عمل کرده است.
- Gate های LSTM، این مدل مشکل ناپدید شدن گرادیان را کمتر حس می‌کند و حتی با ۳ لایه مخفی هم عملکرد خوبی داشته.



شکل ۱: نمودار یادگیری مدل GRU با ۲ لایه میانی، شکل چپ نمودار تابع زیان و شکل راست نمودار دقت برای داده‌های آموزشی و Validation است



شکل ۲: نمودار یادگیری مدل LSTM با ۲ لایه میانی، شکل چپ نمودار تابع زیان و شکل راست نمودار دقت برای داده‌های آموزشی و Validation است



شکل ۳: نمودار یادگیری مدل RNN با ۳ لایه میانی، شکل چپ نمودار تابع زیان و شکل راست نمودار دقت برای داده‌های آموزشی و Validation است

نمودار یادگیری تمامی مدل‌های آزمایش شده در فایل ارسالی به ویوو موجود است، طبق بررسی آن‌ها مشاهده شد که در LSTM، افزایش تعداد لایه‌های میانی از ۲ به ۳ عملکرد را کاهش می‌دهد اما از ۱ به ۲ باعث بهبود می‌شود. این مورد دقیقاً برای GRU هم برقرار است.

مراجع

distantly- using recommendations Justifying McAuley. Julian and Li, Jiacheng Ni, Jianmo [١]

.٢٠١٩ ٠١ ،١٩٧-١٨٨ pages aspects. fine-grained and reviews labeled