

### سوال ۱: Pattern

در این سوال قصد داریم تا توانایی شبکه LSTM برای پیش بینی دنباله های الگودار را بررسی کنیم. دنباله های به صورت زیر را در نظر بگیرید:

$aNb$

$aaNbb$

$aaaNbbb$

...

در واقع دنباله هایی که تعداد  $k$  تا  $a$  و سپس یک  $N$  و مجدداً  $k$  تا  $b$  دارند. برای این کار لازم است تا ۴ نوع کاراکتر داشته باشیم:

$a, b, N, "end"$

که با گرفتن "end" به این نتیجه برسیم که تولید دنباله به اتمام رسیده است. هدف ما این می باشد که با دادن تعدادی از حروف دنباله به ورودی شبکه، کل دنباله را تولید کنیم. به این معنا که مثلاً برای تولید دنباله به فرمت  $aaaaaaNb$  به ورودی  $aaaaaaN$  و سپس یک تک کاراکتر از خروجی دریافت کرده و آن را به ورودی داده و این کار را ادامه می دهیم تا زمانی که در خروجی کاراکتر "end" را دریافت نماییم.

۱. دنباله های با  $k < 11$  را به عنوان داده های آموزش در نظر بگیرید. یک LSTM

با تعداد ۱۰ نرون نهان را با استفاده از ۱۰ دنباله مذکور آموزش دهید.

۲. با استفاده از مدل آموزش داده شده در قسمت قبل، نتیجه را به ازای  $k$  های مختلف بررسی نموده و بیان کنید که مدل آموزش داده شده، تا چه میزان توانایی تولید دنباله درست را دارد.

۳. به ازای دنباله با  $k = 15$  نمودار میزان فعالیت یا همان Cell State را برای همه ی نرون ها رسم نمایید. به نظر شما هر نرون چه اطلاعاتی را در خود ذخیره می کند؟ پاسخ شهودی برای این قسمت مورد نظر می باشد؛ در واقع هدف درک عملکرد حافظه در LSTM است.

۴. مراحل قبل را برای دنباله های به صورت زیر با شبکه ای با تعداد ۲۰ نرون نهان تکرار نمایید. (جمله های ۱ تا ۱۵ از این دنباله را برای آموزش استفاده نمایید.)

$abNba$

$ababNbaba$

$abababNbababa$

$ababababNbabababa$

...

دقت کنید که در این قسمت نیز، همان ۴ کاراکتر بیان شده در قسمت ۱ نیز الفبای ما هستند و هدف بررسی توانایی LSTM در تشخیص رابطه الگوهای بیش از تک حرفی می باشد.

### سوال ۲: Text Generation

در این سوال قصد داریم که با کمک یک شبکه متن تولید کنیم. از آنجا که آموزش این شبکه از نظر محاسباتی سنگین است، توصیه می شود که برای تولید متن حتماً از GPU، مانند سوال دوم تمرین قبل، استفاده نمایید. در این سوال، هر حرف از میان حروف موجود در متن را به صورت یک بردار one-hot در نظر بگیرید.

۱. فایل Book.txt که در ضمیمه قرار دارد، متن یک کتاب می باشد. مجموعه داده ها را به این صورت در نظر بگیرید که هر ۴۰ کاراکتر را به عنوان یک جمله و خروجی معادل با آن را حرف بعد از آن ۴۰ کاراکتر در نظر بگیرید. در نتیجه تعداد زیادی جمله با طول ۴۰ کاراکتر و خروجی یک کاراکتر خواهید داشت. دقت نمایید برای اینکه شبکه بتواند الگوی کلمات را یاد بگیرد، نیاز است تا جملات با یکدیگر همپوشانی داشته باشند، پس مقدار همپوشانی دو جمله متوالی را ۳۵ در نظر بگیرید.

۲. حال یک مدل GRU با ۱۲۸ نورون نهان برای یادگیری این مجموعه دادگان در نظر بگیرید. برای آموزش این مدل از بهینه‌ساز RMSprop با نرخ یادگیری ۰,۰۱ استفاده نمایید. برای خروجی نیز از تابع فعال سازی softmax استفاده نمایید. حال مدل خود را بر روی داده ورودی که در قسمت قبل ایجاد کردید آموزش دهید. مدل حداقل به ۲۰ epoch نیاز دارد تا کلمات معنادار بسازد. برای تست مدل یک جمله از جملات مجموعه دادگان را به تصادف انتخاب کرده و به عنوان ورودی به مدل بدهید و پس از آن با استفاده از احتمالات خروجی برای ایجاد کاراکتر بعدی نمونه بگیرید و به جمله پیشین اضافه نمایید. در نهایت با دادن ۲۰ حرف به شبکه (به صورتی که این ۲۰ حرف تکه‌ای از یک جمله یا متن معنی دار باشند)، خروجی آن را به طول ۴۰۰ کاراکتر نمایش دهید.

### سوال ۳: Sentiment Analysis

در این سوال قصد داریم تا با کمک word2vec، جملات را از نظر معنایی بررسی نماییم، به این صورت که به جملات با توجه به میزان منفی یا مثبت بودن آن‌ها عددی از صفر تا یک اختصاص داده شود.

۱. در اینجا، مجموعه‌ای از جملات همراه با عددی مربوط به میزان مثبت یا منفی بودن معنای آن‌ها قرار دارد. ابتدا با کمک مدل گوگل، کلمات موجود در جملات را به صورت بردار در آورید.

۲. ۶۰ درصد اول از جملات را برای آموزش و باقی را برای تست قرار دهید. حال شبکه‌ی LSTM ای با تعداد ۲۵۶ نورن پنهان را آموزش دهید، تا با گرفتن جمله در ورودی، مقدار sentiment آن را در خروجی پیش‌بینی نماید. مقدار batch size را ۶۴ در نظر بگیرید. توضیح دهید با توجه به این که طول جملات برابر نمی‌باشد، چگونه در batch آن‌ها را آموزش می‌دهید.

۳. مقدار  $MSE$  را بر روی نمونه‌های تست اعلام نموده و درصد صحت را این گونه محاسبه نمایید که اگر مقدار پیش‌بینی شده با مقدار واقعی کمتر از ۰,۰۵، فرق داشت، آن را درست در نظر گرفته و در غیر این صورت پیش‌بینی را غلط در نظر بگیرید.