



دانشکده مهندسی کامپیوتر

بررسی مقاله یادگیری توالی به توالی با شبکه‌های عصبی

گزارش پروژه درس پردازش زبان‌های طبیعی
فاز اول

دانشجو:

مرتضی ذاکری نصرآبادی

استاد:

دکتر بهروز مینایی

آبان‌ماه ۱۳۹۶

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

چکیده

یادگیری ژرف شاخه‌ای نسبتاً جدید از یادگیری ماشین است که در آن گراف‌های چند سطحی یا ژرف از توابع محاسباتی برای شناسایی و تخمین قانون حاکم بر حل یک مسئله پیچیده به کار بسته می‌شوند. شبکه‌های عصبی ژرف وسیله‌ای برای طراحی و پیاده‌سازی این مدل یادگیری هستند. این شبکه‌ها در بسیاری از وظایف یادگیری ماشینی سخت، موفق ظاهر شده‌اند. به‌منظور استفاده از شبکه‌های ژرف در وظایفی که ترتیب ورودی داده در انجام آن مؤثر است مانند اکثر وظایف حوزه پردازش زبان طبیعی، شبکه‌های عصبی مکرر ابداع گشتند که بازنمایی مناسبی از مدل‌های زبانی ارائه می‌دهند. در این گزارش مدل خاصی از شبکه‌های مکرر تحت عنوان مدل توالی‌به‌توالی یا کدگذار-گدگشا بررسی می‌شود که برای وظایفی که شامل توالی‌های ورودی و خروجی با طول متفاوت هستند؛ نظیر ترجمه ماشینی، توسعه داده شده و توانسته است نتایج قابل قبولی را در این زمینه تولید کند.

کلیدواژه‌ها: مدل توالی‌به‌توالی، شبکه عصبی مکرر، یادگیری ژرف، ترجمه ماشینی.

فهرست مطالب

صفحه

عنوان

فهرست شکل‌ها

صفحه	عنوان
۳	شکل (۱). یک طرح‌واره از مدل توالی‌به‌توالی متشکل از دو RNN. این مدل توالی ABC را به‌عنوان ورودی خوانده و توالی WXYZ را به‌عنوان خروجی تولید می‌کند. مدل پس از تولید نشانه <EOS> روند پیش‌بینی خود را متوقف می‌کند.
۶	شکل (۲). گراف محاسباتی مربوط به یک نوع RNN که یک توالی ورودی از مقادیر x را به یک توالی خروجی از مقادیر o نگاشت می‌کند. فرض شده است که خروجی o احتمالات نرمال نشده است، بنابراین خروجی واقعی شبکه یعنی \hat{y} از اعمال تابع بیشینه هموار روی o حاصل می‌شود. چپ: RNN به‌صورت یال بازگشتی. راست: همان شبکه به‌صورت باز شده در زمان، به‌نحوی که هر گره با یک برجسب زمانی مشخص شده است [18].
۶	شکل (۳). طرح‌واره‌ای از حالت‌های مختلف RNN. (الف): شبکه عصبی استاندارد، (ب): شبکه یک به چند، (پ): شبکه چند به یک، (ت) و (ث): شبکه‌های چند به چند [27].
۸	شکل (۴). یک نمونه از معماری RNN کدگذار-کدگشا، که برای یادگیری تولید توالی خروجی $\langle y_1, \dots, y_n \rangle$ از روی توالی ورودی $\langle x_1, \dots, x_n \rangle$ به‌کار می‌رود.
۱۱	شکل (۵). نمایش نحوه آموزش مدل توالی‌به‌توالی در وظیفه NMT.

فهرست جدول‌ها

صفحه

عنوان

No table of figures entries found.

جدول واژگان و نمادهای اختصاری

علامت اختصاری	مفهوم علامت اختصاری
DoS	Denial of Service
LSTM	Long-Short Term Memory
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
SUT	Software Under Test
$f(x, \theta)$	تابع با متغیر مستقل x و پارامتر θ
a	پارامتر اسکالر a
a	بردار a
A	ماتریس A

مقدمه

مدل‌ها و روش‌های یادگیری به کمک شبکه‌های عصبی ژرف (DNN)^۱ اخیراً، با افزایش قدرت محاسباتی سخت‌افزارها و نیز حل برخی از چالش‌های اساسی موجود بر سر راه آموزش و یادگیری این شبکه‌ها، بسیار مورد توجه واقع شده‌اند. DNNها در انجام وظایف سخت یادگیری ماشین مانند تشخیص گفتار، تشخیص اشیاء و غیره، فوق‌العاده قدرت‌مند ظاهر شده‌اند و در مواردی روش‌های سنتی را کاملاً کنار زده‌اند. قدرت بازنمایی زیاد DNNها به این دلیل است که قادر هستند محاسبات زیادی را به صورت موازی در چندین لایه انجام داده، با تعداد زیادی پارامتر پاسخ مسئله داده شده را تخمین زده و مدل مناسبی از آن ارائه دهند. در حال حاضر DNNهای بزرگ می‌توانند با استفاده از الگوریتم پس‌انتشار به صورت بانظارت روی یک مجموعه آموزش برچسب‌زده و به قدر کافی بزرگ آموزش ببینند. بنابراین در مواردی که ضابطه حاکم بر یک مسئله دارای پارامترهای بسیار زیادی است و یک مقدار بهینه از این پارامترها وجود دارد (صرفاً با استناد به این که مغز انسان خیلی سریع همین مسئله را حل می‌کند)، روش یادگیری پس‌انتشار این تنظیم از پارامترها (مقدارهای بهینه) را یافته و مسئله را حل می‌کند.

بسیاری از وظایف یادگیری ماشین به حوزه پردازش زبان طبیعی مربوط می‌شوند؛ جایی که در آن معمولاً ترتیب ورودی‌ها و خروجی‌ها یک مسئله مهم است. برای مثال در ترجمه ماشینی دو جمله با کلمات یکسان ولی ترتیب متفاوت، معانی (خروجی‌های) مختلفی دارند. این وظایف اصطلاحاً مبتنی بر توالی هستند. در واقع ورودی آنها به صورت یک توالی است. شبکه‌های عصبی رو به جلو ژرف برای این دسته از وظایف خوب عمل نمی‌کنند؛ چرا که قابلیتی برای به خاطر سپاری و مدل‌سازی ترتیب در آنها تعبیه نشده است.

^۱ deep neural networks

شبکه‌های عصبی مکرر (RNN)^۱ خانواده‌ای از شبکه‌های عصبی برای پردازش وظایف مبتنی بر توالی هستند. همانطور که شبکه‌های عصبی پیچشی (CNN)^۲، ویژه پردازش یک تور از مقادیر، برای مثال یک تصویر، طراحی شده‌اند؛ یک شبکه عصبی مکرر نیز برای پردازش یک توالی از مقادیر $x = \langle x^{(1)}, x^{(2)}, \dots, x^{(T)} \rangle$ ساخته شده است. خروجی شبکه‌های عصبی مکرر نیز مانند ورودی آنها در اغلب وظایف یک دنباله است. این قابلیت پردازش توالی شبکه‌های عصبی آنها را برای استفاده در مدل‌های زبانی مناسب ساخته است.

شرح مسئله و اهمیت موضوع

برخلاف انعطاف پذیری و قدرت بالای شبکه‌های عصبی مکرر، در حالت ساده این شبکه‌ها یک توالی ورودی با طول ثابت را به یک توالی خروجی با همان طول نگاشت می‌کنند. این موضوع اما یک محدودیت جدی است؛ زیرا، بسیاری از مسائل مهم، در قالب توالی‌هایی که طولشان از قبل مشخص نیست، به بهترین شکل قابل بیان هستند و در نظر گرفتن یک طول ثابت از پیش تعیین شده برای ورودی و خروجی به خوبی مسئله را مدل نمی‌کند. برای مثال ترجمه ماشینی و تشخیص گفتار مسائلی از این دست هستند. همچنین سیستم پرسش و پاسخ را نیز می‌توان به صورت نگاشت یک توالی از کلمات به عنوان پرسش به یک توالی دیگر از کلمات به عنوان پاسخ، در نظر گرفت. بنابراین واضح است که یک ایجاد روش مستقل از دامنه برای یادگیری نگاشت توالی به توالی مفید و قابل توجیه خواهد بود.

اهداف و راهکارها

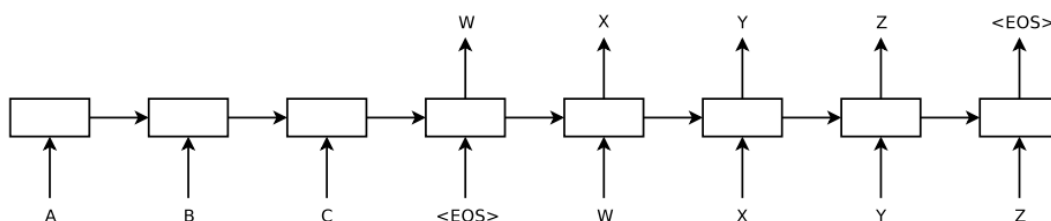
همانطور که دیدیم طیف وسیعی از وظایف NLP مبتنی بر نگاشت توالی‌های با طول نامشخص و متغیر به یکدیگر است. همچنین روش‌های سنتی مثل n-garm دارای محدودیت‌های خاص خود در حل این دسته مسائل هستند و استفاده از روش‌های یادگیری ژرف به وضوح امید بخش

^۱ recurrent neural networks

^۲ convolutional neural networks

بوده است. بنابراین هدف ارایه یک مدل مبتنی بر RNNها جهت نگاشت توالی‌به‌توالی است. در این گزارش راهکار مطرح شده در [1] و نتایج آن به تفصیل شرح داده می‌شود.

Sutskever و همکاران [2] نشان دادند که چگونه یک کاربرد ساده از شبکه با معماری حافظه کوتاه‌مدت بلند (LSTM) می‌تواند مسائل نگاشت توالی‌به‌توالی را حل کند. ایده اصلی استفاده از یک LSTM برای خواندن توالی ورودی، به صورت یک نمونه در هر مرحله زمانی، جهت اقتباس برداری بزرگ با بعد ثابت و سپس استفاده از یک LSTM دیگر برای استخراج توالی خروجی از آن بردار است. LSTM دوم دقیقاً یک مدل زبانی مبتنی بر RNN است با این تفاوت که به حاوی احتمال شرطی نسبت به توالی ورودی نیز هست. قابلیت LSTM در یادگیری موفق وابستگی‌های مکانی طولانی مدت موجود در توالی‌ها آن را برای استفاده در مدل پیشنهادی مناسب ساخته است. شکل ۱ یک طرح‌واره از مدل پیشنهاد شده را در به صورت عام نشان می‌دهد.



شکل (۱) یک طرح‌واره از مدل توالی‌به‌توالی متشکل از دو RNN. این مدل توالی ABC را به عنوان ورودی خوانده و توالی WXYZ را به عنوان خروجی تولید می‌کند. مدل پس از تولید نشانه <EOS> روند پیش‌بینی خود را متوقف می‌کند.

داده‌ها و نتایج

مدل پیشنهادی در بخش قبل، بر روی وظیفه ترجمه ماشینی عصبی (NMT) مورد آزمایش قرار گرفته است. برای انجام آزمایش‌ها از مجموعه داده ترجمه انگلیسی به فرانسوی WMT'14 استفاده شده است [3]. همچنین مجموعه داده کوچکتری در [۱۰۰۰ جمله فارسی] وجود دارد که برای آموزش مدل‌های کوچکتر مناسب است.

نتایج حاصل شده از این کار بدین قرار است. بر روی مجموعه داده WMT'14 با استخراج مستقیم ترجمه از پنج LSTM ژرف با ۳۸۰ میلیون پارامتر، در نهایت امتیاز BLEU معادل ۳۴,۸۱ کسب گردیده است. این امتیاز بالاترین امتیازی است که تا زمان ارایه این مقاله از طریق NMT حاصل شده است. به عنوان مقایسه امتیاز BLEU برای ترجمه ماشینی آماری (SMT) بر روی همین مجموعه داده برابر ۳۳,۳۰ است [۱]. این درحالی است که امتیاز ۳۴,۸۱ با احتساب اندازه واژه‌نامه ۸۰ هزار کلمه به دست آمده و هر جا که کلمه ظاهر شده در ترجمه مرجع در واژه‌نامه نبوده این امتیاز جریمه شده است. بنابراین نتایج نشان می‌دهد که یک معماری مبتنی بر شبکه عصبی تقریباً غیربهمینه، که نقاط زیادی برای بهبود دارد، قادر است تا روش‌های سنتی مبتنی بر عبارت سیستم SMT را شکست دهد.

مفاهیم اولیه

در این قسمت پیرامون سه مفهوم اصلی گزارش پیشرو، یعنی مدل زبانی، شبکه‌های عصبی مکرر و ترجمه ماشینی عصبی توضیحاتی ارائه می‌گردد.

مدل زبانی

مدل زبانی (LM) یک مفهوم پایه در NLP است که امکان پیش‌بینی نشانه بعدی در یک توالی را فراهم می‌کند. به بیان دقیق‌تر LM عبارت است از یک توزیع احتمالی روی یک توالی از نشانه‌ها (اغلب واژه‌ها) که احتمال وقوع یک توالی داده شده را مشخص می‌کند. در نتیجه می‌توان بین چندین توالی داده شده برای مثال چند جمله، آن را که محتمل‌تر است، انتخاب کرد. LM برای توالی $x = \langle x^{(1)}, x^{(2)}, \dots, x^{(n)} \rangle$ عبارت است از:

$$p(x) = \prod_{t=1}^n p(x^{(t)} | x^{(<t)}) \quad (۲)$$

مدل‌های سنتی n-gram برای غلبه بر چالش‌های محاسباتی، با استفاده از فرض مارکوف رابطه ۱ را به در نظر گرفتن تنها n-1 نشانه قبلی محدود می‌کنند. به همین دلیل برای توالی‌های

طولانی (بیشتر از ۴ یا ۵ نشانه) و دیده نشده مناسب نیستند. مدل‌های زبانی عصبی (NLMS) که بر مبنای شبکه‌های عصبی عمل پیش‌بینی واژه بعدی را انجام می‌دهند، در ابتدا برای کمک به n-gram با آنها ترکیب شدند که منجر به پیچیدگی‌های زیادی شد؛ در حالی که مشکل توالی‌های طولانی همچنان وجود داشت. اخیراً، معماری‌های جدیدی برای LM که کاملاً بر اساس DNNها است، ایجاد شده‌اند. سنگ‌بنای این مجموعه معماری‌ها RNNها هستند.

شبکه‌های عصبی مکرر

RNNها کلاسی از شبکه‌های عصبی هستند که به صورت یک *گراف جهت‌دار دوری* بیان می‌شوند. به عبارت دیگر ورودی هر یک از لایه‌(های) پنهان یا خروجی علاوه بر خروجی لایه قبل، شامل ورودی از مرحله قبل به صورت بازخورد نیز می‌شود. (۶) شکل (۲) یک RNN را نشان می‌دهد. همانطور که پیداست، لایه پنهان از مراحل قبلی هم بازخورد می‌گیرد. در هر مرحله زمانی t از $t = 1$ تا $t = n$ یک بردار $x^{(t)}$ از توالی ورودی $x = \langle x^{(1)}, x^{(2)}, \dots, x^{(n)} \rangle$ پردازش می‌شود. در حالت کلی معادله‌های بروزرسانی (گذر جلو)^۱ یک RNN در t عبارتند از [۱]:

$$a^{(t)} = Ux^{(t)} + Wh^{(t-1)} + b, \quad (۳)$$

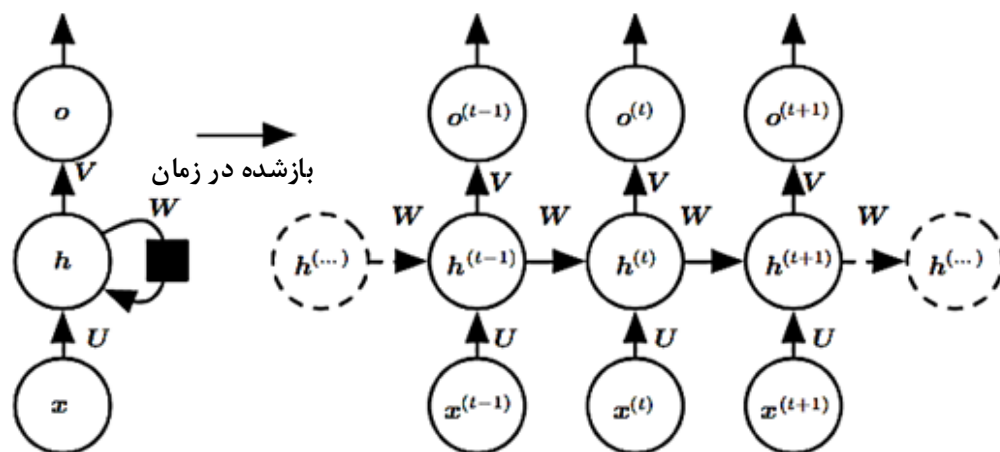
$$h^{(t)} = \Phi(a^{(t)}), \quad (۴)$$

$$o^{(t)} = Vh^{(t)} + c, \quad (۵)$$

$$\hat{y}^{(t)} = \text{softmax}(o^{(t)}), \quad (۶)$$

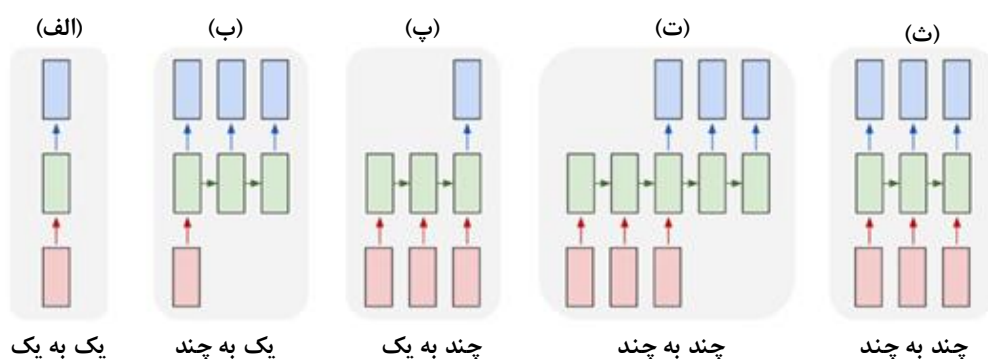
که در آن بردارهای b و c بایاس و ماتریس‌های U ، V و W به ترتیب وزن یال‌های لایه ورودی به پنهان، پنهان به خروجی و پنهان به پنهان، تشکیل‌دهنده مجموعه پارامترهای شبکه هستند. Φ تابع انگیزش است که معمولاً یکی از توابع ReLU یا سیگموئید انتخاب می‌شود. لایه آخر را نیز تابع بیشینه هموار تشکیل می‌دهد که احتمال وقوع هر نشانه خروجی را مشخص می‌کند.

^۱ forward pass



شکل (۲) گراف محاسباتی مربوط به یک نوع RNN که یک توالی ورودی از مقادیر x را به یک توالی خروجی از مقادیر o نگاشت می‌کند. فرض شده است که خروجی o احتمالات نرمال نشده است، بنابراین خروجی واقعی شبکه یعنی \hat{y} از اعمال تابع بیشینه هموار روی o حاصل می‌شود. چپ: RNN به صورت یال بازگشتی. راست: همان شبکه به صورت باز شده در زمان، به نحوی که هر گره با یک برجسب زمانی مشخص شده است [18].

در شکل (۲) RNN با یک لایه پنهان نشان داده شده است. اما می‌توان RNN ژرف با چندین لایه پنهان نیز داشت. همچنین طول توالی‌های ورودی و خروجی می‌تواند بسته به مسئله مورد نظر متفاوت باشد. karpathy [] RNNها را از منظر طول توالی ورودی و طول توالی خروجی به چند دسته تقسیم‌بندی کرده است که شکل (۳) نشان داده شده است.



شکل (۳) طرح‌واره‌ای از حالت‌های مختلف RNN. (الف): شبکه استاندارد، (ب): شبکه یک به چند، (پ): شبکه چند به یک، (ت) و (ث): شبکه‌های چند به چند [27].

تصویر karpathy از حالت‌های مختلف RNN بعد از مقاله منتخب در این گزارش می‌باشد با این حال در بخش ۴ خواهیم دید که چگونه می‌توان از این طرح‌ها برای ایده معماری توالی به توالی نیز الهام گرفت.

ترجمه ماشینی عصبی

به‌طور کلی MT را می‌توان با یک LM که به جمله زبان مبدأ مشروط شده است، مدل‌سازی کرد. بر همین اساس NMT را می‌توان یک مدل زبانی مکرر در نظر گرفت که مستقیماً احتمال شرطی $p(y|x)$ را در ترجمه جمله زبان مبدأ $x = \langle x^{(1)}, x^{(2)}, \dots, x^{(n)} \rangle$ به جمله زبان مقصد $y = \langle y^{(1)}, y^{(2)}, \dots, y^{(m)} \rangle$ مدل می‌کند. دقت شود که طول جمله مبدأ یعنی n و جمله مقصد یعنی m الزاماً برابر نیست. بنابراین در NMT هدف محاسبه این احتمال و سپس استفاده از آن در تولید جمله به زبان مقصد به کمک DNN‌ها است.

کارهای مرتبط

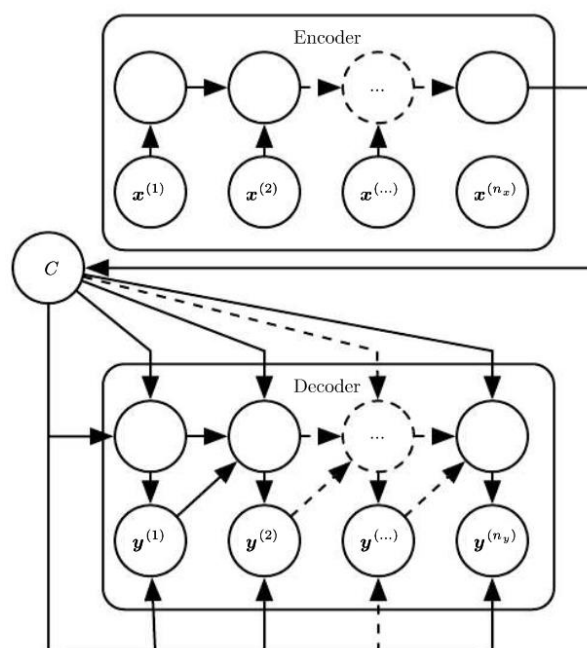
کارهای زیادی در زمینه NLMs انجام شده است. در بیشتر این کارها از شبکه‌های عصبی روبه‌جلو یا مکرر استفاده شده و کاربرد آن معمولاً در یک وظیفه MT با امتیازدهی مجدد n-best list اعمال شده و نتایج آن معمولاً نشان از بهبود امتیازهای قبلی داشته است.

اخیراً کارهایی در زمینه فشردن اطلاعات زبان مبدأ در NLM انجام شده است. برای نمونه Auli و همکاران [1] NLM را مدل عنوان^۱ جمله ورودی ترکیب کرده‌اند که نتایج بهبود بخشی داشته است. کار انجام شده در مقاله [2] به کار [3] بسیار نزدیک است. آنها برای اولین بار توالی ورودی را به یک بردار نگاشت داده و سپس آن را به توالی خروجی تبدیل کردند. البته در این کار، برای تبدیل توالی به بردار، از CNNs استفاده شده که ترتیب واژه‌ها را حفظ نمی‌کند. چو و همکاران یک معماری شبه LSTM را برای نگاشت توالی ورودی به بردار و سپس استخراج توالی خروجی و نهایتاً ترکیب آن با SMT استفاده کرده‌اند. معماری آنها از دو RNN با عنوان‌های

^۱ topic model

کدگذار و کدگشا تشکیل شده که RNN اول وظیفه تبدیل یک توالی با طول متغیر به یک بردار با طول ثابت را قابل یک سلول زمینه c دارد و RNN دوم وظیفه تولید توالی خروجی را با لحاظ کردن c و نماد شروع جمله مقصد بر عهده دارد. معماری پیشنهادی آنها تحت عنوان کلی RNN کدگذار-کدگشا در شکل ۳ نشان داده شده است. چون آنها از LSTM استفاده نکرده و بیشتر تلاش خود را معطوف به ترکیب این روش با مدل‌های قبلی SMT کرده‌اند، برای توالی‌های ورودی و خروجی طولانی همچنان مشکل عدم حفظ حافظه وجود دارد.

Bahdanau و همکاران یک روش ترجمه مستقیم با استفاده از شبکه عصبی پیشنهاد داده‌اند که از سازوکار توجه برای غلبه بر کارآمدی ضعیف روش چو روی جملات طولانی استفاده می‌کند و به نتایج مطلوبی دست یافتند.



شکل (۴) یک نمونه از معماری RNN کدگذار-کدگشا، که برای یادگیری تولید توالی خروجی $\langle y^{(1)}, \dots, y^{(n_y)} \rangle$ از روی توالی ورودی $\langle x^{(1)}, \dots, x^{(n_x)} \rangle$ به کار می‌رود.

مدل توالی به توالی

در این مدل از دو RNN با واحدهای LSTM استفاده شده است. هدف LSTM در اینجا تخمین احتمال شرطی $p(\langle y^{(1)}, y^{(2)}, \dots, y^{(m)} \rangle | \langle x^{(1)}, x^{(2)}, \dots, x^{(n)} \rangle)$ است که قبلاً هم دیده بودیم (بخش ۲-۳). LSTM این احتمال شرطی را ابتدا با اقتباس بازنمایی بعد ثابت v برای توالی ورودی $\langle x^{(1)}, x^{(2)}, \dots, x^{(n)} \rangle$ از آخرین مقدار حالت پنهان و در ادامه با محاسبه احتمال $\langle y^{(1)}, y^{(2)}, \dots, y^{(m)} \rangle$ از رابطه استاندارد مطرح در LM (رابطه (۱)) و در نظر گرفتن v برای حالت پنهان آغازین به صورت داده شده در رابطه زیر، حساب می‌کند:

$$p(\langle y^{(1)}, y^{(2)}, \dots, y^{(m)} \rangle | \langle x^{(1)}, x^{(2)}, \dots, x^{(n)} \rangle) = \prod_{t=1}^m p(y^{(t)} | v, y^{(1)}, \dots, y^{(t-1)}) \quad (7)$$

در رابطه (۶) هر توزیع احتمالی $p(y^{(t)} | v, y^{(1)}, \dots, y^{(t-1)})$ به وسیله یک تابع بیشینه هموار روی همه واژه‌های داخل واژه‌نامه بازنمایی می‌شود. برای LSTM از روابط [Graves] استفاده شده است. هر جمله در این مدل نیاز است تا با یک علامت خاص مثل $\langle \text{EOS} \rangle$ خاتمه یابد. این امر مدل را قادر می‌سازد تا بتواند توزیع احتمالی را روی توالی با هر طول دلخواهی تعریف کند. شمای کلی مدل در شکل (۱) نشان داده شده است. در این شکل LSTM بازنمایی توالی ورودی $\langle 'A', 'B', 'C', \langle \text{EOS} \rangle \rangle$ را حساب و سپس از این بازنمایی برای محاسبه احتمال توالی خروجی $\langle 'W', 'X', 'Y', 'Z', \langle \text{EOS} \rangle \rangle$ استفاده می‌کند.

مدل پیاده‌سازی شده در عمل از سه جنبه با مدل معرفی شده در بالا تفاوت دارد. اول، از دو LSTM جداگانه استفاده شده است: یکی برای توالی ورودی و دیگری برای توالی خروجی؛ زیرا، انجام این کار پارامترهای مدل را با هزینه محاسباتی اندکی، به تعداد بسیار زیادی افزایش می‌دهد. دوم اینکه LSTM‌های ژرف به شکل قابل توجهی LSTM‌های سطحی را شکست می‌دهند، به همین دلیل LSTM با ژرفای چهار لایه به کار گرفته شده است. سوم اینکه نویسندگان در این مقاله یافته‌اند که وارون کردن توالی ورودی در سرعت همگرایی آموزش شبکه و نیز دقت پیش‌بینی آن تأثیر شگرفی ایفا می‌کند. بنابراین به جای نگاشت مستقیم توالی $\langle a, b, c \rangle$ به توالی $\langle \alpha, \beta, \gamma \rangle$ ، LSTM برای نگاشت $\langle c, b, a \rangle$ به $\langle \alpha, \beta, \gamma \rangle$ آموزش داده می‌شود.

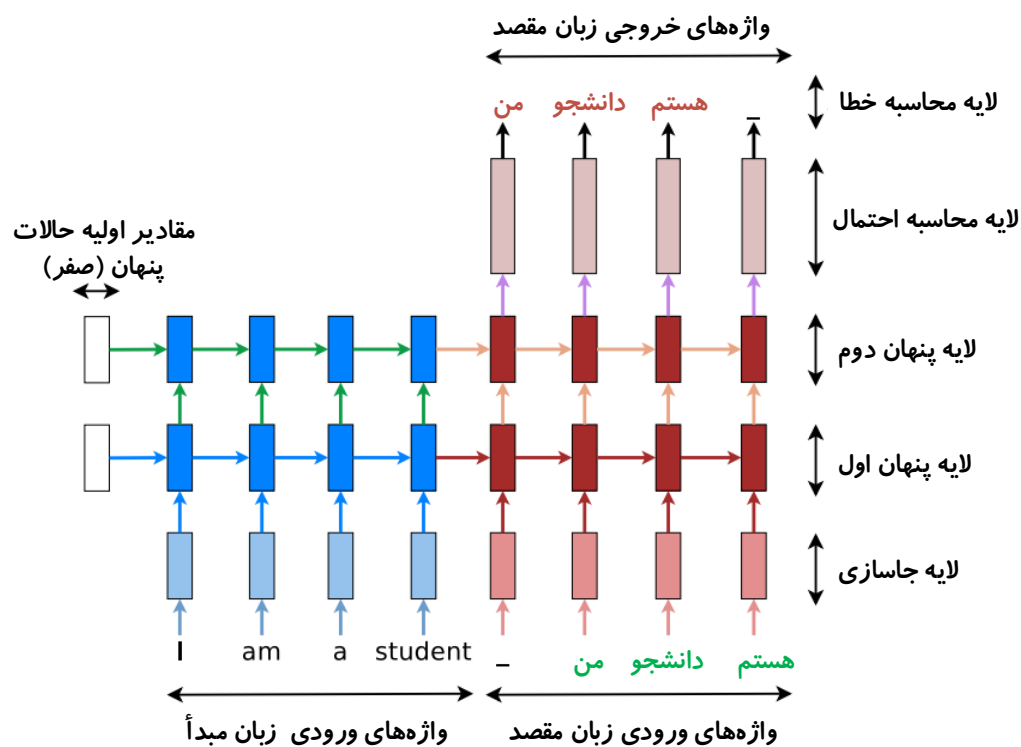
که در آن $\alpha \beta \gamma$ ترجمه متناظر با $a b c$ است. توجیه اثر این پدیده آن است که در نگاشت به روش وارون ابتدای عبارت‌ها که متناظر با یکدیگر هستند به هم نزدیک شده و این امر سبب زودتر همگرا شدن الگوریتم SGD و نزدیک شدن به مقادیر بهینه می‌شود.

آموزش شبکه

مدل توالی‌به‌توالی پس از معرفی توسط Sutskever و همکاران، بارها و بارها تا به امروز مورد ارجاع دیگران قرار گرفته و تبدیل به یک مدل مرجع در NMT شده است. این مدل در رساله دکتری آقای Luong به تفصیل و همراه با برخی اصلاحات توضیح داده شده است. در این بخش به برخی جزئیات آموزش شبکه مدل توالی‌به‌توالی می‌پردازیم.

شکل (۵) یک نمایش دقیق‌تر از مدل ذکر شده در شکل (۱) را نشان می‌دهد. آموزش شبکه بدین نحو است: ابتدا جمله زبان مقصد سمت راست جمله متناظر خود در زبان مبدأ قرار داده می‌شود. نشان ‘-’ نقش <EOS> را دارد که البته می‌تواند پایان جمله مبدأ یا آغاز جمله مقصد را مشخص کند. بنابراین به هر کدام از دو گروه قابل تعلق است. LSTM سمت چپ یا همان شبکه کدگذار، در هر مرحله زمانی یک واژه از جمله زبان مبدأ را خوانده پس از تبدیل به نمایش مناسب حالت داخلی لایه پنهان را بروزسانی می‌کند. در مرحله پردازش آخرین واژه مقادیر لایه‌های پنهان بردار ثابت v که اکنون نماینده کل جمله ورودی زبان مبدأ است را تشکیل می‌دهد. سپس LSTM دوم یا شبکه کدگشا اولین واژه زبان مقصد را به همراه بردار v ، به‌عنوان ورودی دریافت می‌کند و پیشبینی خود را انجام می‌دهد. برچسب واقعی این داده در واقع واژه بعدی در جمله زبان مقصد است. پس از مقایسه و محاسبه خطا، الگوریتم پس‌انتشار روی هر دوشبکه با شروع از شبکه کدگشا اجرا می‌شود و پارامترها را در خلاف جهت گرادیان تنظیم می‌کند. این روند تا پایان یافتن جمله زبان مقصد ادامه پیدا می‌کند. البته در عمل ممکن است ورودی به صورت یک دسته به شبکه داده شود.

در مرحله آزمون به جای مقایسه با برچسب و محاسبه خطا فقط احتمال آمدن کلمه بعدی محاسبه و کلمه از روی واژگان پیدا می‌شود. سپس خروجی مرحله t به عنوان ورودی مرحله $t+1$ به شبکه کدگشا داده می‌شود.



شکل (۵) نمایش نحوه آموزش مدل توالی به توالی در وظیفه NMT.

جزئیات آموزش شبکه

در مقاله [] از LSTM ژرف با چهار لایه و ۱۰۰۰ سلول حافظه در هر لایه استفاده شده است. همچنین اندازه واژگان ورودی ۱۶۰,۰۰۰ و اندازه واژگان خروجی ۸۰,۰۰۰ کلمه است. حاصل کار یک شبکه LSTM با مجموع ۳۸۰ میلیون پارامتر بوده که ۶۴ میلیون آن اتصالات برگشتی هستند. دیگر جزئیات عبارتند از:

- پارامترها با مقادیر تصادفی از توزیع یکنواخت در بازه $[-0.08, +0.08]$ مقداردهی اولیه شده‌اند.

- برای آموزش از SGD استاندارد با نرخ یادگیری 0.7 استفاده شده است. بعد از گذشت پنج دوره^۱، نرخ یادگیری در هر نیم‌دور، نصف می‌شود. تعداد کل دوره‌های آموزش برابر

آزمایش‌ها

نتیجه‌گیری و کارهای آتی

^۱ epoch

مراجع

مراجع استفاده شده

واژه‌نامه

معادل انگلیسی

Instrumenting

Noise

واژه‌ی فارسی

ابزار گذاری

اختلال



**Iran University of Science and Technology
School of Computer Engineering**

**A Survey of Sequence-to-Sequence Architectures
with Neural Networks**

A Project Submitted in IUST NLP course

By:
Morteza Zakeri Nasrabadi

Instructor:
Dr. Behrouz Minaei

November 2017