دانشکده مهندسی کامپیوتر

بررسی مقاله یادگیری توالی‌به‌توالی با شبکه‌های عصبی

گزارش پروژه درس پردازش زبان‌های طبیعی
فاز اول

دانشجو:
مرتضی ذاکری نصرآبادی

استاد:
دکتر بهروز مینایی

آبان‌ماه ۱۳۹۶


بسم الله الرحمن الرحیم

چکیده

یادگیری ژرف شاخه‌ای نسبتا جدید از یادگیری ماشین است که در آن توابع محاسباتی به شکل گراف‌های چند سطحی یا ژرف برای شناسایی و تخمین قانون حاکم بر حل یک مسئله پیچیده به‌کار بسته می‌شوند. شبکه‌های عصبی ژرف ابزاری 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 ارایه می‌دهند. این مدل‌ها در حالت ساده برای همه وظیفه‌های یک مدل زبانی مناسب نیستند. 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 مــطالب

عنوان | صــفحه

فهرست شکل‌ها

عنوان **صـفحه**

فهرست جدول‌ها

عنوان **صـفحه**

No table of figures entries found.

جدول واژگان و نمادهای اخـتصاری

کوته‌نوشت	مفهوم کوته‌نوشت
CNN	**Convolutional Neural Networks**
DNN	**Deep Neural Network**
LM	**Language Model**
LSTM	**Long-short term memory**
NLP	**Natural Language Processing**
NLM	**Neural Language Models**
NMT	**Neural Machine Translation**
ReLU	**Rectified Linear Unit**
RNN	**Recurrent Neural Network**
SMT	**Statistical Machine Translation**

۱ مقدمه

مدل‌ها و روش‌های یادگیری به‌کمک شبکه‌های عصبی ژرف (DNNs)[۱] اخیرا، 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 کاملا 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 ارایه دهند. درحال حاضر DNNهای بزرگ می‌توانند با استفاده از الگوریتم پس‌انتشار[۲] به‌صورت بانظارت[۳] روی یک مجموعه آموزش برچسب‌زده و به‌قدر 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 (صرفا با استناد به این که مغز انسان همین مسئله را خیلی سریع حل می‌کند)، روش یادگیری پس‌انتشار این تنظیم از پارامترها (مقدارهای بهینه) را یافته و مسئله را حل می‌کند [1].

بسیاری از وظایف یادگیری ماشین به حوزه پردازش زبان طبیعی (NLP)[۴] مربوط می‌شوند؛ جایی که در آن معمولا 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 اصطلاحا مبتنی بر توالی[۵] هستند. در واقع ورودی آنها به صورت یک توالی است. شبکه‌های

[۱] deep neural networks
[۲] backpropagation
[۳] supervised
[۴] natural language processing
[۵] sequence

عصبی رو به جلو ژرف[۱] برای این دسته از وظایف خوب عمل نمی‌کنند؛ چرا که قابلیتی برای به‌خاطر سپاری و مدل‌سازی ترتیب در آنها تعبیه نشده است.

شبکه‌های عصبی مکرر (RNNs)[۲] خانواده‌ای از شبکه‌های عصبی برای پردازش وظایف مبتنی بر توالی هستند. همانطور که شبکه‌های عصبی پیچشی (CNNs)[۳]، ویژه پردازش یک تور[۴] از مقادیر، برای مثال یک تصویر، طراحی شده‌اند؛ یک RNN نیز همسو با پردازش یک توالی از مقادیر ورودی $x = < x^{(1)}, x^{(2)}, \dots x^{(T)} >$ ساخته شده است [2]. خروجی RNNها نیز مانند ورودی آنها در اغلب وظایف یک توالی است. این قابلیت پردازش توالی توسط شبکه‌های عصبی، آنها را برای استفاده در وظایف NLP، بسیار درخور ساخته است.

۱-۱- شرح مسئله و اهمیت موضوع

برخلاف انعطاف پذیری و قدرت بالای RNNها، 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 (MT)[۵] و تشخیص گفتار[۶] مسائلی از این دست هستند. همچنین سیستم پرسش و پاسخ را نیز می‌توان به صورت نگاشت یک توالی از واژه‌ها به‌عنوان پرسش، به یک توالی دیگر از واژه‌ها به عنوان پاسخ، در نظر گرفت. بنابراین پُر واضح است که ایجاد یک روش مستقل از دامنه برای یادگـیری نگاشت توالی‌به‌تولی مفید و قابل توجیه خواهد بود [1].

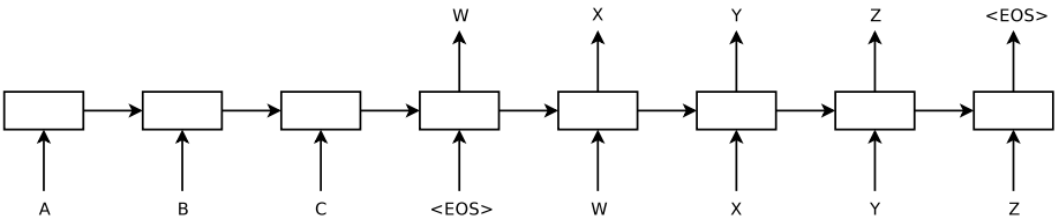
---

۱ deep feed-forward neural networks
۲ recurrent neural networks
۳ convolutional neural networks
۴ grid
۵ machine translation
۶ speech recognition

۱-۲- 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 بوده است. بنابراین هدف ارایه یک مدل مبتنی بر RNNها جهت نگاشت توالی‌به‌توالی است. در این گـزارش راهکار مطرح شده در [1] و نتایج آن به‌تفصیل شرح داده می‌شود.

Sutskever و همکاران [1] نشان دادند که چگونه یک کاربرد ساده از شبکه با معماری حافظه کوتاه‌مدت بلند (LSTM)[۱] می‌تواند مسائل نگاشت توالی‌به‌توالی را حل کند. ایده اصلی استفاده از یک LSTM برای خواندن توالی ورودی، به‌صورت 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 دقیقا یک مدل زبانی مبتنی بر RNN است با این تفاوت که حاوی احتمال شرطی نسبت به توالی ورودی نیز هست. قابلیت LSTM در یادگیری موفق وابستگی‌های مکانی طولانی مدت نهفته درون توالی‌ها، آن را برای استفاده در مدل پیشنهادی مناسب ساخته است. شکل (۱) یک طرح‌واره از این مدل را به صورت عام نشان می‌دهد.



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 [1].

[۱] long-short term memory

۱-۳- داده‌ها و نتایج

مدل پیشنهادی در بخش قبل، برروی وظیفه ترجمه ماشینی عصبی (NMT)[۱] 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 [4] وجود دارد که برای آموزش مدل‌های کوچکتر مناسب است. این مجموعه شامل ترجمه‌های انگلیسی به فارسی نیز هست.

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. به‌عنوان مقایسه امتیاز BLEU برای ترجمه ماشینی آماری (SMT)[۲] برروی همین مجموعه داده برابر ۳۳٫۳۰ است. این درحالی است که امتیاز ۳۴٫۸۱ با احتساب اندازه واژه‌نامه ۸۰هزار کلمه به‌دست آمده و هرجا 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 تقریبا غیر بهینه، که نقاط زیادی برای بهبود دارد، قادر است تا روش‌های سنتی مبتنی بر عبارتِ سیستم SMT را شکست دهد [1].

۲ مفاهیم اولیه

در این قسمت پیرامون سه مفهوم اصلی گزارش پیشرو، یعنی مدل زبانی (LM)[۳]، شبکه‌های عصبی مکرر و ترجمه ماشینی عصبی، به‌صورت مختصر توضیحاتی ارایه می‌گردد.

[۱] neural machine translation
[۲] statistical machine translation
[۳] language model

۲-۱- مدل زبانی

LM یک مفهوم پایه در NLP است که امکان پیش‌بینی نشانه بعدی در یک توالی را فراهم می‌کند. به‌بیان 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 [5]. LM برای توالی $x = < x^{(1)}, x^{(2)}, \dots x^{(n)} >$ عبارت است از:

$$p(x) = \prod_{t=1}^{n} p(x^{(t)} | x^{(<t)}) \qquad (۱)$$

مدل‌های سنتی n-gram برای غلبه بر چالش‌های محاسباتی، با استفاده از فرض مارکوف رابطه (۱) را به درنظر گرفتن تنها n-1 نشانه قبلی محدود می‌کنند. به‌همین دلیل برای توالی‌های طولانی (بیشتر از ۴ یا ۵ نشانه) و دیده نشده مناسب نیستند. مدل‌های زبانی عصبی (NLMs)[۱] که بر مبنای شبکه‌های عصبی عمل پیش‌بینی واژه بعدی را انجام می‌دهند، در ابتدا برای کمک به n-gramها با آنها ترکیب شدند که منجر به ایجاد پیچیدگی‌های زیادی شد؛ در حالی که مشکل توالی‌های طولانی همچنان وجود داشت [5]. اخیرا اما، معماری‌های جدیدی برای LM که کاملا بر اساس DNNها است، ایجاد شده‌اند. سنگ‌بنای این مجموعه معماری‌ها RNNها بوده که در بخش بعدی معرفی می‌شوند.

۲-۲- شبکه‌های عصبی مکرر

RNNها کلاسی از شبکه‌های عصبی هستند که به‌صورت یک ***گراف جهت‌دار دوری*** بیان می‌شوند. به‌عبارت دیگر ورودی هریک از لایه(های) پنهان یا خروجی علاوه بر خروجی لایه قبل، شامل ورودی از مرحله قبل به‌صورت بازخورد نیز می‌شود. (۵)شکل (۲) یک RNN را نشان می‌دهد. همانطور که پیداست، لایه پنهان از مراحل قبلی هم بازخورد می‌گیرد. در هر مرحله‌زمانی

[۱] neural language models

t از ($t = 1$ تا $t = n$) یک بردار $x^{(t)}$ از توالی ورودی $x = < x^{(1)}, x^{(2)}, \dots x^{(n)} >$ پردازش می‌شود. در حالت کلی معادله‌های بروزرسانی (گذرجلو[1]) یک RNN در t عبارتند از [2]:

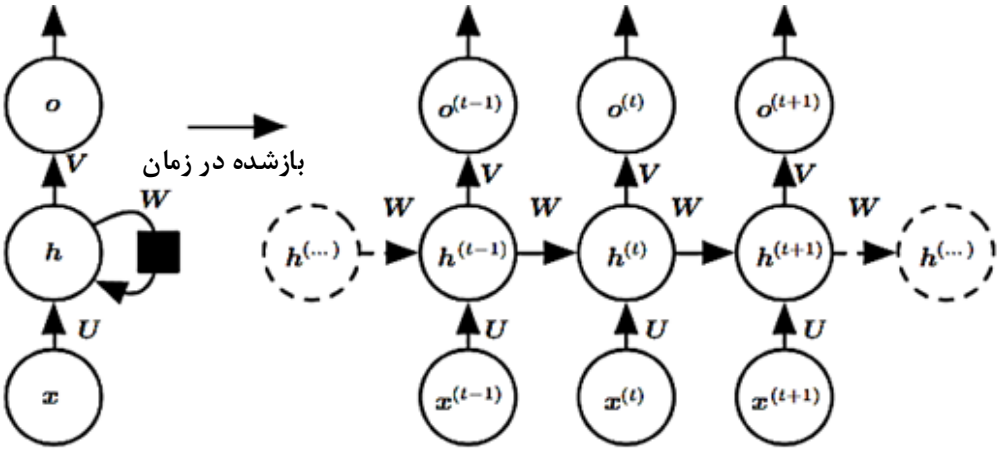
$$a^{(t)} = Ux^{(t)} + Wh^{(t-1)} + b, \quad (۲)$$

$$h^{(t)} = \Phi(a^{(t)}), \quad (۳)$$

$$o^{(t)} = Vh^{(t)} + c, \quad (۴)$$

$$\hat{y}^{(t)} = softmax(o^{(t)}), \quad (۵)$$

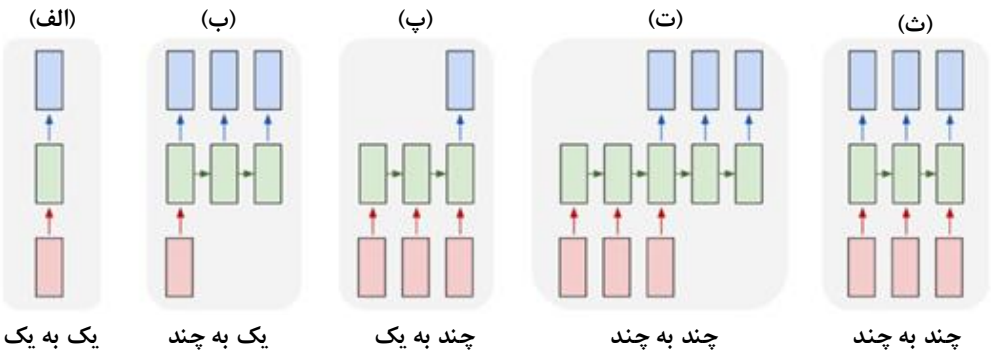
که در آن بردارهای b و c بایاس و ماتریس‌های U، V و W به‌ترتیب وزن یال‌های لایه ورودی به پنهان، پنهان به خروجی و پنهان به پنهان، تشکیل‌دهنده مجموعه پارامترهای شبکه هستند. Φ تابع انگیزش است که معمولا یکی از توابع ReLU[2] یا سیگموید[3] انتخاب می‌شود. لایه آخر را نیز تابع بیشینه هموار[4] تشکیل می‌دهد که احتمال وقوع هر نشانه خروجی را مشخص می‌کند.



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 برچسب زمانی مشخص شده است [2].

[1] forward pass
[2] rectified linear unit
[3] sigmoid
[4] softmax function

در شکل (۲)، RNN با یک لایه پنهان نشان داده شده است. اما می‌توان RNNژرف با چندین لایه پنهان نیز داشت. همچنین طول توالی‌های ورودی و خروجی می‌تواند بسته به مسئله مورد نظر متفاوت باشد. karpathy [6] RNNها را از منظر طول توالی ورودی و طول توالی خروجی به چند دسته تقسیم‌بندی کرده است. شکل (۳) این دسته‌بندی را نشان می‌دهد.



شکل (۳) طرح واره‌ای 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 [6].

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

۲-۳- ترجمه ماشینی عصبی

به‌طور کلی MT را می توان 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 مستقیما احتمال شرطی $p(y|x)$ را در ترجمه جمله زبان مبدأ $x = < x^{(1)}, x^{(2)}, \dots x^{(n)} >$ به جمله زبان مقصد $y = < y^{(1)}, y^{(2)}, \dots y^{(m)} >$ مدل می‌کند. دقت شود که طول جمله مبدأ یعنی n و جمله مقصد یعنی m الزاما برابر نیست. بنابراین در NMT هدف محاسبه این احتمال و سپس استفاده از آن در تولید جمله به زبان مقصد، هر دو به کمک DNNها است [5].

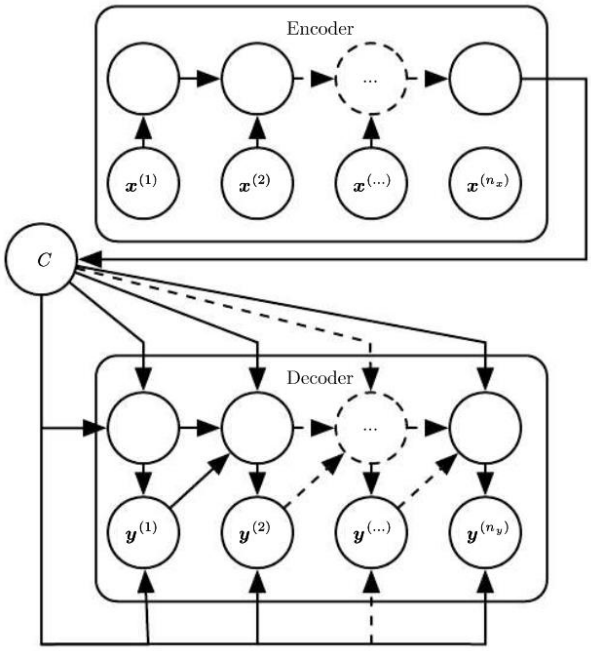
۳ کارهای مرتبط

کارهای زیادی در زمینه NLMs انجام شده است. در بیشتر این کارها از شبکه‌های عصبی روبه‌جلو یا مکرر استفاده شده و کاربرد آن معمولا در یک وظیفه MT با امتیازدهی مجدد n فهرست بهتر، اعمال شده و نتایج آن معمولا نشان از بهبود امتیازهای قبلی داشته است [1].

اخیرا کارهایی در زمینه فشردن اطلاعات زبان مبدأ در NLM انجام شده است. برای نمونه Auli و همکاران [7] NLM را با مدل عنوان[۱] جمله ورودی ترکیب کرده‌اند که نتایج بهبود بخشی داشته است. کار انجام شده در مقاله [1] به کار [8] بسیار نزدیک است. در مقاله [8] نویسندگان برای اولین بار توالی ورودی را در یک بردار فشرده کرده 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. چـــو و همکاران [9] یک معماری شبهِ LSTM را برای نگاشت توالی ورودی به بردار و سپس استخراج توالی خروجی و نهایتا ترکیب آن با SMT استفاده کرده‌اند. معماری آنها از دو RNN با عنوان‌های 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 (۴) 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 [10] یک روش ترجمه مستقیم با استفاده از شبکه عصبی پیشنهاد داده‌اند که از سازوکار *attention* برای غلبه بر کارآمدی ضعیف روش [9] روی جملات طولانی استفاده می‌کند و به نتایج مطلوبی دست یافتند.

[۱] topic model



شکل (۴) مدل RNN کدگذار-کدگشا، که برای یادگیری تولید توالی خروجی $< y^{(1)}, \dots y^{(n_y)} >$ از روی توالی ورودی $< x^{(1)}, \dots x^{(n_x)} >$ به‌کار می‌رود [2].

۴ مدل توالی‌به‌توالی

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

$$p(< y^{(1)}, \dots y^{(m)} > | < x^{(1)}, \dots x^{(n)} >) = \prod_{t=1}^{m} p(y^{(t)} | v, y^{(1)}, \dots, y^{(t-1)}) \qquad (۶)$$

در رابطه (۶) هر توزیع احتمالی $p(y^{(t)} | v, y^{(1)}, \dots, y^{(t-1)})$ به‌وسیله یک تابع بیشینه هموار روی همه واژه‌های داخل واژه‌نامه بازنمایی می‌شود. برای LSTM از روابط [11] استفاده شده است. هر جمله در این مدل نیاز است تا با یک علامت خاص مثل <EOS> 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 $\ll A', B', C', < EOS \gg$ را حساب و سپس از این بازنمایی برای محاسبه احتمال توالی خروجی $< W', X', Y', Z', < EOS >>$ استفاده می‌کند. در عین حال این مدل را می‌توان ترکیبی از قسمت‌های پ و ت ۲-۲-(۵)شکل (۳) دانست.

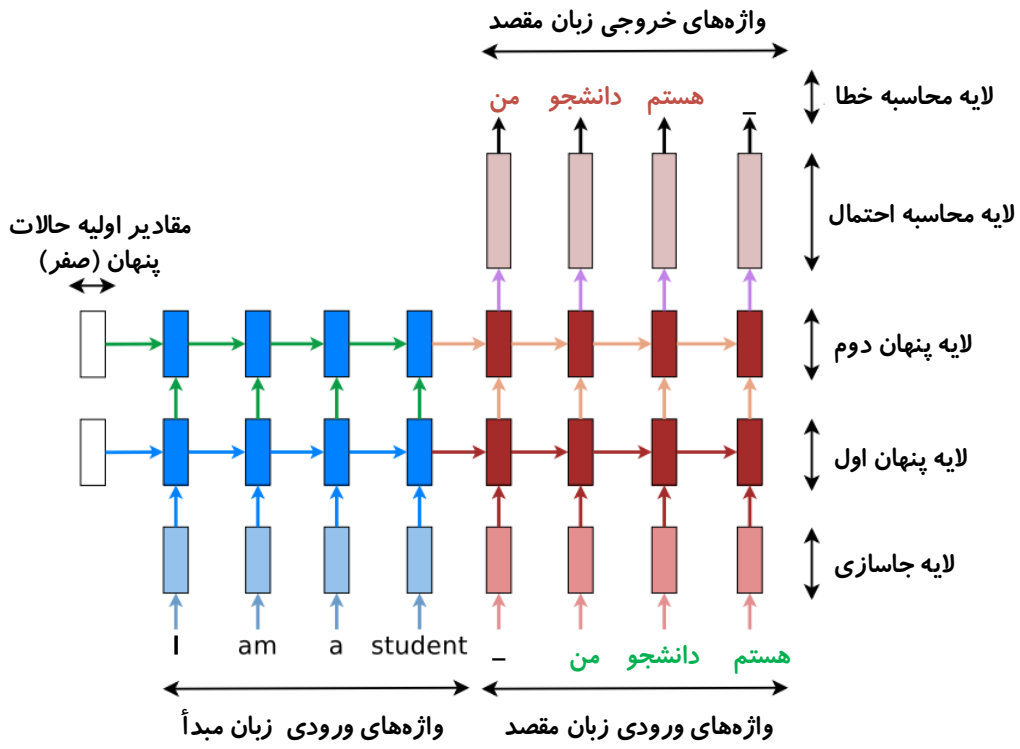
مدل پیاده‌سازی شده در عمل از سه جنبه با مدل معرفی شده در بالا تفاوت دارد. اول، از دو LSTM جداگانه استفاده شده است: یکی برای توالی ورودی و دیگری برای توالی خروجی؛ زیرا، انجام این کار پارامترهای مدل را با هزینه محاسباتی اندکی، به تعداد بسیار زیادی افزایش می‌دهد. دوم اینکه LSTMهای ژرف به‌شکل قابل توجهی LSTMهای سطحی را شکست می‌دهند، به همین دلیل LSTM با ژرفای چهار لایه به‌کار گرفته شده است. سوم اینکه نویسندگان در این مقاله یافته‌اند که وارون کردن توالی ورودی در سرعتِ همگرایی آموزش شبکه و نیز دقت پیش‌بینی آن تأثیر شگرفی ایفا می‌کند. بنابراین به‌جای نگاشت مستقیم توالی $< a, b, c >$ به توالی$< \alpha, \beta, \gamma >$، LSTM برای نگاشت $< c, b, a >$ به $< \alpha, \beta, \gamma >$ 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 به‌هم نزدیک شده و این امر سبب زودتر همگرا شدن الگوریتم SGD و نزدیک شدن به مقادیر بهینه می‌شود [1].

۴-۱- آموزش شبکه

مدل توالی‌به‌توالی پس از معرفی توسط Sutskever و همکاران [1]، 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 [5] به‌تفصیل 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 ۱-۲-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 مرحله‌زمانی یک واژه از جمله زبان مبدأ را خوانده پس از تبدیل به نمایش مناسب حالت داخلی لایه پنهان را بروزرسانی 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 دو شبکه 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.



شکل (۵) نمایش نحوه آموزش مدل توالی‌به‌توالی روی وظیفه NMT [5].

در مرحله آزمون به جای مقایسه با برچسب و محاسبه خطا فقط احتمال آمدن واژه بعدی محاسبه و واژه از روی واژگان پیدا می‌شود. سپس خروجی مرحله t به عنوان ورودی مرحله t+1 به شبکه کدگشا داده می‌شود. این روش اصطلاحا teacher forcing نامیده می‌شود[2].

۴-۱-۲- جزئیات آموزش شبکه

در مقاله [1] از LSTMژرف 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 ۳۸۰میلیون پارامتر بوده که ۶۴میلیون آن اتصالات برگشتی هستند. دیگر جزئیات پارامترها و آموزش شبکه 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 7.5 بوده است.
- گرادیان بر روی دسته‌های ۱۲۸تایی از توالی‌ها محاسبه شده و به اندازه دسته، یعنی ۱۲۸، تقسیم می‌شود.
- هرچند LSTMها از معضل میرایی گرادیان[۲] رنج نمی‌برند، اما ممکن است مشکل انفجار گرادیان[۳] را داشته باشند. بنابراین محدودیت سختی بر مقدار نورم گرادیان اعمال می‌شود به‌این نحو که هنگامی که نورم از مقدار آستانه‌ای بیشتر شد، مجددا تنظیم شود. برای هر دسته در مجموعه آموزش مقدار $s = ||g||_2$ محاسبه می‌شود که در آن g مقدار گرادیان پس از تقسیم بر ۱۲۸ است. اگر $s > 5$ شد آنگاه قرار داده می‌شود: $g = \frac{5g}{s}$.

[۱] epoch
[۲] vanishing gradient
[۳] exploding gradient

- جملات مختلف طول‌های مختلفی دارند. بیشتر آنها کوتاه هستند (طولی بین ۲۰ تا ۳۰ دارند) اما برخی از آنها طولانی هستند (طولی بیشتر از ۱۰۰ دارند)؛ بنابراین دسته‌های ۱۲۸تایی از جملات که تصادفی انتخاب می‌شوند تعداد کمی جمله طولانی داشته و تعداد زیادی جمله کوتاه و در نتیجه سبب می‌شود تا بیشتر محاسبات داخل هر دسته هدر روند. برای غلبه بر این موضوع سعی شده است همه جملات داخل یک دسته طول تقریبا مساوی داشته باشند. این امر انجام محاسبات را تا ۲برابر تسریع کرده است.

۵ آزمایش‌ها

روش یادگیری توالی‌به‌توالی معرفی شده روی وظیفه ترجمه ماشینی انگلیسی به فرانسوی در دو حالت مختلف آزمایش گردیده است. در حالت اول مدل، برای ترجمه مستقیم جملات انگلیسی به فرانسوی به‌کار گرفته شده و در حالت دوم برای امتیاز دهی مجدد n لیست بهتر[۱] از جملات در وظیفه SMT استفاده شده است. در این قسمت نتایج آزمایش‌های انجام گرفته در قالب امتیازهای ترجمه کسب شده، نمونه جملات ترجمه شده و بلاخره مصورسازی بازنمایی جملات ورودی، بیان شده است.

۵-۱- پیاده‌سازی

پیاده‌سازی مدل اولیه با C++ انجام شده است. این پیاده‌سازی از LSTMژرف با پیکربندی شرح داده شده در بخش ۴-۱-۲ روی یک GPU، تقریبا ۱۷۰۰ واژه بر ثانیه را پردازش می‌کند. این سرعت بسیار پایین است. برای این منظور مدل به صورت موازی شده روی ۸ عدد GPU اجرا می‌شود. هر لایه از LSTM روی یک GPU اجرا شده و فعالیت‌های خود را به محض محاسبه به GPU یا لایه بعدی می‌دهد. چون مدل چهار لایه دارد، چهار GPU دیگر برای موازی‌سازی بیشینه هموار استفاده شده‌اند بنابراین هر GPU مسئول محاسبه یک ضرب ماتریسی (ماتریس با اندازه ۲۰۰۰ × ۱۰۰۰) است. نتیجه حاصل از این موازی‌سازی در سطح GPU، رسیدن به

[۱] n-best list

سرعت پردازش ۶۳۰۰ واژه بر ثانیه است. فرایند آموزش در این شیوه پیاده‌سازی، ۱۰ روز به طول انجامید [1].

علاوه بر پیاده‌سازی اولیه، پیاده‌سازی‌های دیگری نیز از این مدل در زبان‌ها و چهارچوب‌های مختلف ارایه شده است؛ از جمله دو پیاده‌سازی خوب با زبان پایتون و روی چهارچوب‌های کاری Tensorflow و Keras. پیاده‌سازی Tensorflow سازوکارهای جدیدتر مثل سازوکار *attention* را نیز اضافه کرده است [12]. پیاده‌سازی Keras هم به جای واژه، در سطح کاراکتر انجام شده است [13].

۵-۱-۱- جزئیات مجموعه داده

این قسمت در فاز دوم تکمیل می‌گردد.

۵-۱-۲- کدگشایی و امتیازدهی مجدد

این قسمت در فاز دوم تکمیل می‌گردد.

۵-۱-۳- وارون‌سازی جملات مبدأ

این قسمت در فاز دوم تکمیل می‌گردد.

۵-۱-۴- ارزیابی نتایج

این قسمت در فاز دوم تکمیل می‌گردد.

۵-۱-۵- کارآمدی روی جملات طولانی

این قسمت در فاز دوم تکمیل می‌گردد.

۵-۱-۶- تحلیل مدل

این قسمت در فاز دوم تکمیل می‌گردد.

۶ نتیجه‌گیری و کارهای آتی

این قسمت در فاز دوم تکمیل می‌گردد.

مراجع

[1] Q. V. Le Ilya Sutskever, Oriol Vinyals, I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Nips*, pp. 1–9, 2014.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, 2016.

[3] "ACL 2014 ninth workshop on statistical machine translation." [Online]. Available: http://www.statmt.org/wmt14/medical-task/index.html. [Accessed: 13-Nov-2017].

[4] "Tab-delimited bilingual bentence pairs from the tatoeba project (good for anki and similar flashcard applications)." [Online]. Available: http://www.manythings.org/anki/. [Accessed: 13-Nov-2017].

[5] M. T. Luong, "Meural machine translation," Stanford university, 2016.

[6] A. Karpathy, "Connecting images and natural language," Stanford University, 2016.

[7] M. Auli, M. Galley, C. Quirk, and G. Zweig, "Joint language and translation modeling with recurrent neural networks.," *Emnlp*, no. October, pp. 1044–1054, 2013.

[8] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," *Emnlp*, no. October, pp. 1700–1709, 2013.

[9] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014.

[10] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," pp. 1–15, 2014.

[11] A. Graves, "Generating sequences with recurrent neural networks," pp. 1–43, 2013.

[12] M.-T. Luong, E. Brevdo, and R. Zhao, "Neural machine translation (seq2seq) tutorial," *https://github.com/tensorflow/nmt*, 2017.

[13] "Sequence to sequence example in Keras (character-level)," 2017. [Online]. Available: https://github.com/fchollet/keras/blob/master/examples/lstm_seq2seq.py. [Accessed: 13-Nov-2017].

واژه‌نامه

واژه‌نامه فارسی به انگلیسی

واژه‌ی فارسی	معادل انگلیسی
انفجار گرادیان	Exploding Gradient
بانظارت	Supervised
بیشینه هموار	Softmax Function
پردازش زبان طبیعی	Natural Language Processing
پس‌انتشار	Backpropagation
ترجمه ماشینی	Machine Translation
ترجمه ماشینی آماری	Statistical Machine Translation
ترجمه ماشینی عصبی	Neural Machine Translation
تشخیص گفتار	Speech Recognition
توالی	Sequence
حافظه کوتاه مدت بلند	Long-Short Term Memory
دوره	Epoch
شبکه عصبی پیچشی	Convolutional Neural Network
شبکه عصبی رو به جلو ژرف	Deep Feed-forward Neural Networks
شبکه عصبی ژرف	Deep Neural Network
شبکه عصبی مکرر	Recurrent Neural Network
گذر جلو	Forward Pass
مدل زبانی	Language Model
مدل زبانی عصبی	Neural Language Models
میرایی گرادیان	Vanishing Gradient

❊❊❊

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