
بسم الله الرحمن الرحيم

دانشکده : مرکز آموزش الکترونیکی

رشته مهندسی کامپیوتر گرایش هوش مصنوعی

پایان نامه کارشناسی ارشد

تحلیل و مقایسه عملکرد توابع فعالساز در شبکه های حافظه کوتاه و بلندمدت

نگارنده : امیر فرزاد

استاد راهنما :

دکتر هدی مشایخی

استاد مشاور:

دکتر حمید حسن پور

تیر ماه ۱۳۹۵

دانشکده : مرکز آموزش الکترونیکی

گروه : مهندسی کامپیوتر

پایان نامه کارشناسی ارشد آقای امیر فرزاد به شماره دانشجویی: ۹۳۱۳۰۳۴

تحت عنوان:

تحلیل و مقایسه عملکرد توابع فعالساز در شبکه های حافظه کوتاه و بلندمدت

در تاریخ ..... توسط کمیته تخصصی زیر جهت اخذ مدرک کارشناسی ارشد مورد ارزیابی و با درجه ..... مورد پذیرش قرار گرفت.

اساتید راهنما	امضاء	اساتید مشاور	امضاء
نام و نام خانوادگی :		نام و نام خانوادگی :	
نام و نام خانوادگی :		نام و نام خانوادگی :	

اساتید داور	امضاء	نماینده تحصیلات تکمیلی	امضاء
نام و نام خانوادگی :		نام و نام خانوادگی :	
نام و نام خانوادگی :			
نام و نام خانوادگی :			
نام و نام خانوادگی :			

تقدیم اثر

تقدیم به پدر، مادر و خواهرم.
و به یک دوست.

تشکر و قدردانی

بدون شک جایگاه و منزلت معلم، جلیل‌تر از آن است که در مقام قدردانی از زحمات بی‌شائبه‌ی او، با زبان قاصر و دست ناتوان، چیزی بنگاریم. اما ازآنجایی‌که تجلیل از معلم، سپاس از انسانی است که هدف و غایت آفرینش را تأمین می‌کند و سلامت امانت‌هایی را که به دستش سپرده‌اند، تضمین؛ برحسب وظیفه: از استاد با کمالات و شایسته؛ سرکار خانم دکتر مشایخی که در کمال سعه‌صدر، با حسن خلق و فروتنی، از هیچ کمکی در این عرصه بر من دریغ ننمودند و زحمت راهنمایی این رساله را بر عهده گرفتند؛ از جناب آقای مهندس کریمی که بدون مساعدت ایشان، این پروژه به نتیجه مطلوب نمی‌رسید کمال تشکر و قدردانی را دارم. باشد که این خردترین، بخشی از زحمات آنان را سپاس گوید.

تعهد نامه

اینجانب امیر فرزاد دانشجوی دوره کارشناسی ارشد رشته مهندسی کامپیوتر گرایش هوش مصنوعی دانشکده مرکز آموزش الکترونیکی دانشگاه صنعتی شاهرود نویسنده پایان نامه "تحلیل و مقایسه عملکرد توابع فعالساز در شبکه های حافظه کوتاه و بلندمدت" تحت راهنمائی استاد گرامی خانم دکتر مشایخی متعهد می شوم.

- **تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.**
- **در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده است.**
- **مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.**
- **کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « Shahrood University of Technology» به چاپ خواهد رسید.**
- **حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.**
- **در کلیه مراحل انجام این پایان نامه ، در مواردی که از موجود زنده (یا بافتهای آنها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.**
- **در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است .**

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

- **کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه های رایانه ای، نرم افزار ها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می باشد. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.**
- **استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.**

چکیده

شبکه حافظه کوتاه و بلندمدت نوعی از معماری شبکه بازگشتی است که برای جلوگیری از مشکلات وابستگی های طولانی مدت در واحدهای لایه های مخفی (ناپدید شدن گرادیان) طراحی شده است. هر واحد در شبکه حافظه کوتاه و بلندمدت به صورت یک بلوک نمایش داده می شود. بلوک‌های شبکه حافظه کوتاه و بلندمدت به‌عنوان سلول‌های حافظه به صورت بازگشتی به هم متصل هستند. هر بلوک در لایه مخفی این شبکه شامل چندین دروازه با توابع فعالساز برای تنظیم اطلاعات درون آن است. توابع فعالساز مورد استفاده در اغلب شبکه‌های عصبی به‌خصوص در بلوک‌های شبکه حافظه کوتاه و بلندمدت، توابع سیگموئید و تانژانت هایپربولیک می‌باشند. توابع فعالساز دیگری نیز معرفی شده‌اند، اما در انتخاب تابع فعالساز مناسب در شبکه‌های حافظه کوتاه و بلندمدت پژوهش چندانی صورت نگرفته است. در این پایان‌نامه ابتدا به بررسی و جمع آوری توابع فعالساز مختلف قابل اعمال بر روی دروازه‌های سیگموئیدی بلوک‌های شبکه حافظه کوتاه و بلندمدت پرداخته شده، و ۲۳ تابع فعالساز مختلف انتخاب شده است. سپس به بررسی دقت رده‌بندی شبکه با استفاده از این توابع فعالساز پرداخته شده است. بدین منظور از شبکه حافظه کوتاه و بلندمدت با یک‌لایه مخفی برای کلاسه‌بندی استفاده شده و ارزیابی بر روی دو مجموعه داده IMDB و Movie Review انجام شده است. علاوه بر بررسی عملکرد شبکه با توابع مختلف, تعداد بلوک های شبکه در لایه مخفی نیز برای دو مجموعه داده مورد نظر بررسی شده است. نتایج بدست‌آمده بر روی مجموعه داده نشان می‌دهد توابع فعالساز *modified Elliott* و *cloglogm* دارای بهترین نتایج روی هر دو مجموعه داده هستند. این در حالی است که تابع فعالساز سیگموئید، که به طور معمول در دروازه‌های سیگموئیدی شبکه حافظه کوتاه و بلندمدت به کار می‌رود، عملکرد ضعیف‌تری از توابع فعالساز پیشنهادی دارد. نتایج نشان داد که بازه بزرگتر در تابع فعالساز لایه مخفی می تواند باعث بهبود عملکرد شبکه گردد. همچنین بررسی تعداد بلوک در لایه مخفی نشان داد که انتخاب تعداد بلوک ها در لایه مخفی ارتباط مستقیم با پیچیدگی و طول مجموعه داده مورد استفاده در شبکه دارد.

کلمات کلیدی

شبکه عصبی، شبکه حافظه کوتاه و بلندمدت، تابع فعالساز، دروازه سیگموئیدی.

لیست مقالات مستخرج از پایان نامه

[1] Farzad, A., Mashayekhi, H., & Hassanpour, H. (2016) A Comparative Performance Analysis of Various Activation Functions in LSTM Networks for Classification. submitted to Neural Computing and Applications Journal

[۲] فرزاد ا. و مشایخی ه،(۱۳۹۵)، "ارزیابی نقش تابع فعالساز در عملکرد شبکه‌های حافظه کوتاه و بلند مدت"، کنفرانس ملی تحقیقات بین رشته ای در مهندسی کامپیوتر، برق، مکانیک و مکاترونیک، قزوین

فهرست مطالب

فهرست تصاویر و جداول

۱- فصل اول: مقدمه

۱-۱- مقدمه

شبکه های عصبی مصنوعی[۱] به عنوان مدل‌های ریاضی و زیستی مغز ابداع شدند [۱]. ساختار اساسی یک شبکه عصبی متشکل از شبکه ای از واحدها یا گره‌ها است که به وسیله وزن‌ها به همدیگر متصل هستند. براساس مدل زیستی, گره ها نشان دهنده نورون[۲] و وزن‌های متصل کننده نشان دهنده قدرت محل تماس[۳] دو نورون هستند. شبکه با خوراندن ورودی به یک یا چند گره فعال می شود و این فعالیت به وسیله وزن‌های متصل کننده به شبکه اعمال می شود.

شبکه های عصبی متفاوت و متنوعی در سال های اخیر معرفی شده اند. دسته مهمی از شبکه های عصبی به نام شبکه عصبی پیشخور[٤] معروف هستند. از معروف ترین این شبکه ها می توان به شبکه های عصبی پرسپترون[٥] [۲]، شبکه واحد تابع شعاعی[٦] [۳]، شبکه های هاپفیلد[۷] [۴] اشاره کرد. مهمترین شبکه عصبی پیشخور که در مقالات گوناگون از آن استفاده شده است شبکه پرسپترون چندلایه[۸] [۵] است. هر شبکه پرسپترون می تواند شامل چندین لایه باشد. در هر لایه تعداد گره‌های متفاوتی می تواند وجود داشته باشد. در این شبکه هر گره در هر لایه به وسیله وزن‌ها (بردار ها) به همه گره‌های لایه بعد به صورت رو به جلو متصل است اما در یک لایه هیچ گره ای به گره دیگر متصل نیست. ورودی ها در لایه ورودی به لایه مخفی انتشار پیدا می کنند و سپس به لایه خروجی می روند. این رویه به نام انتشار رو به جلو در شبکه نامیده می شود. شبکه پرسپترون چندلایه فقط براساس ورودی حال حاضر هستند و براساس ورودی ها در گذشته یا آینده نیستند، بنابراین این شبکه‌ها بیشتر مناسب کلاس بندی الگو هستند تا اینکه مناسب کلاس بندی یا برچسب گذاری رشته ها باشند.

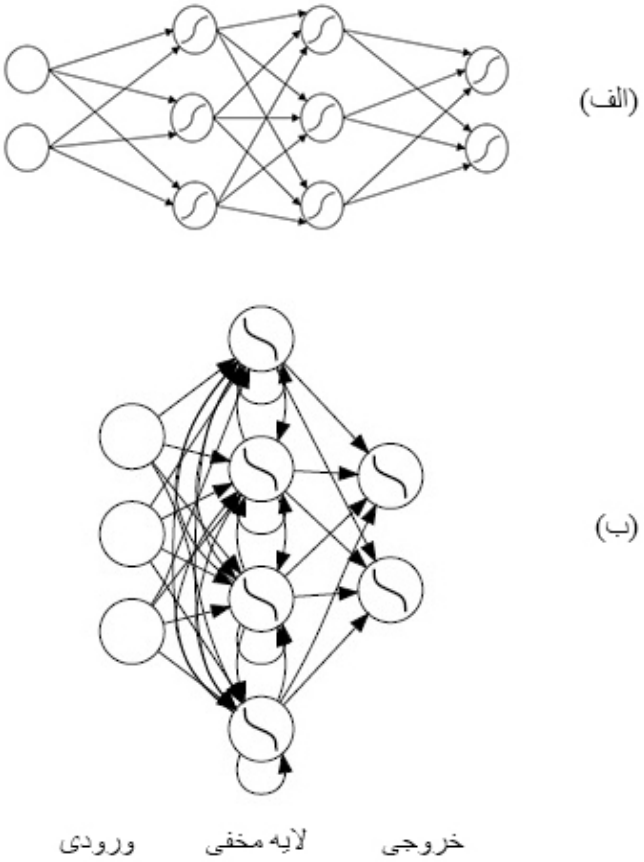
---

[۱] Artificial neural network
[۲] Neuron
[۳] Synapse
[٤] Feedforward
[٥] Perceptron
[٦] Radial basis function
[۷] Hopfield
[۸] Multilayer perceptron

۱–۲– شبکه بازگشتی استاندارد[۹]

دسته مهم دیگر از شبکه های عصبی، شبکه های بازگشتی است که نتایج مناسبی در پردازش زبان طبیعی داشته است [۶]. شبکه بازگشتی استاندارد شبیه به شبکه پرسپترون چندلایه است اما در لایه مخفی هر گره می تواند توسط بردار وزن‌ها به گره‌های دیگر در همان لایه متصل شود. گرچه این تغییر جزیی به نظر می آید اما در واقع پیچیده است. یک شبکه پرسپترون چندلایه فقط می تواند بردارها را از هر ورودی به خروجی نگاشت کند، در حالی که یک شبکه بازگشتی می تواند نگاشتی از همه ورودی های گذشته به خروجی داشته باشد. نکته اساسی این است که اتصالات بازگشتی به عنوان یک "حافظه" از ورودی های گذشته آن‌ها را در حالت داخلی خود حفظ کرده و در نتیجه می تواند روی خروجی تاثیر بگذارد. به دلیل اینکه برای کلاس بندی رشته ها, ورودی ها وابسته به ورودی‌های قبلی یا بعدی خود هستند، برای این نوع ورودی ها شبکه بازگشتی مناسب است. به طور مثال اگر یک جمله یک رشته در نظر گرفته شود، هر کلمه در جمله وابسته به کلمات قبلی یا بعدی خود در جمله است. شبکه‌های عصبی بازگشتی ساده دارای حافظه طولانی‌مدت[۱۰] به فرم وزن‌ها هستند. وزن‌ها می‌توانند به آرامی در طی آموزش تغییر کنند و دانش عمومی در رابطه با داده را یاد بگیرند. آن‌ها همچنین حافظه کوتاه مدت[۱۱] را به فرم فعالسازهایی بی دوام دارند که از هر گره به گره های پشت سر هم داده می‌شوند. مقایسه ای از معماری یک شبکه بازگشتی استاندارد و یک شبکه پرسپترون چندلایه در شکل (۱-۱) نمایش داده شده است.

[۹] Recurrent neural network
[۱۰] Long-term memory
[۱۱] Short-term memory



شکل (۱-۱): مقایسه بین شبکه پرسپترون چندلایه و شبکه بازگشتی استاندارد. (الف): شبکه عصبی پرسپترون چندلایه با دو لایه مخفی. شکل های S مانند نشان دهنده تابع فعالساز سیگموئید هستند. (ب): شبکه عصبی بازگشتی استاندارد با یک لایه مخفی [۷].

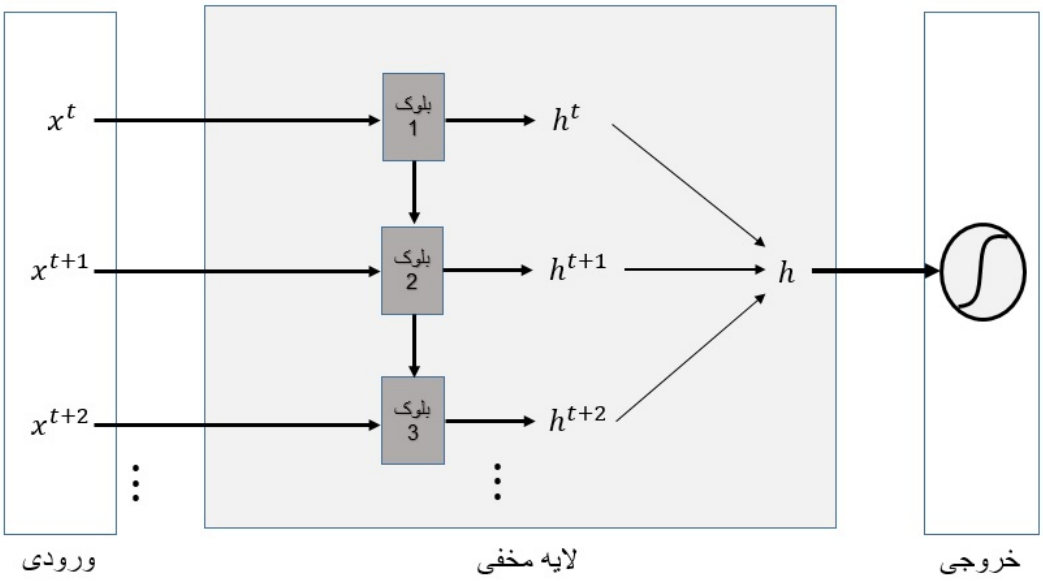
۱-۲-۱- شبکه حافظه کوتاه و بلند مدت

در سال ۱۹۹۷ شبکه‌های حافظه کوتاه و بلندمدت[۱۲] توسط Hochreiter و Schmidhuber [۸] برای غلبه بر مشکلات وابستگی طولانی‌مدت در واحدهای لایه های مخفی ارائه شد. این مدل شبیه به یک شبکه عصبی بازگشتی استاندارد با یک‌لایه مخفی است، اما هر گره در لایه مخفی با یک بلوک جایگزین

[۱۲] Long short-term memory

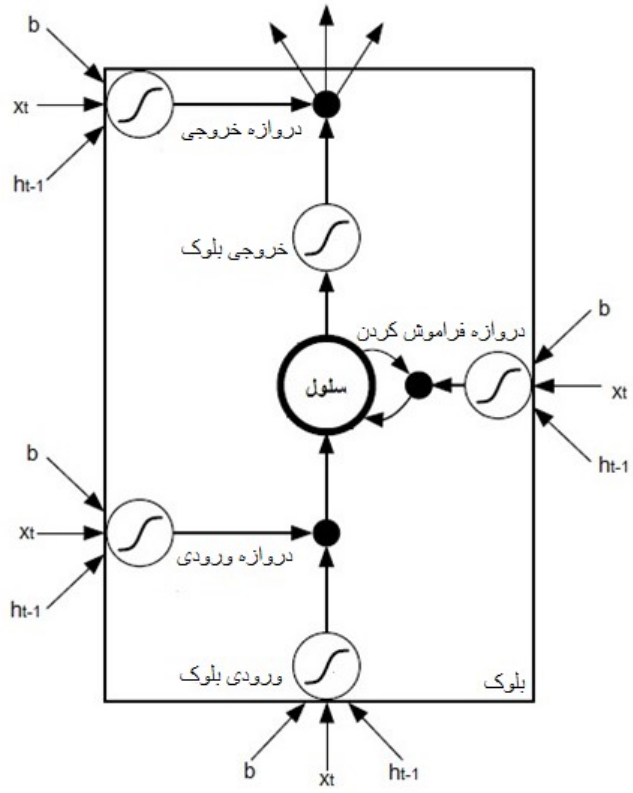
شده است. هر واحد در شبکه حافظه کوتاه و بلند مدت به صورت یک بلوک نمایش داده می شود. بلوک‌های شبکه حافظه کوتاه و بلندمدت به نوبه خود نوعی از شبکه بازگشتی است که به‌عنوان سلول‌های حافظه مشخص به صورت بازگشتی به هم متصل هستند. شکل های (۱-۲) و (۱-۳) به ترتیب معماری شبکه حافظه کوتاه و بلند و مدت و یک بلوک شبکه حافظه کوتاه و بلندمدت را نشان می‌دهد.

در تحقیقات اخیر از بلوک‌های شبکه حافظه کوتاه و بلندمدت با دروازه‌های فراموشی[۱۳] و اتصالات روزنه‌ای[۱۴] استفاده شده است [۷]. به‌طور عمومی شبکه حافظه کوتاه و بلندمدت از پنج جز اصلی شامل ورودی بلوک، دروازه ورودی، دروازه فراموشی، حالت سلول و دروازه خروجی تشکیل شده است.



شکل (۱-۲): معماری شبکه حافظه کوتاه و بلندمدت شامل چندین بلوک در یک لایه مخفی. تمام ورودی‌ها به تمام بلوک‌ها در لایه مخفی اعمال می شوند.

[۱۳] Forget gates
[۱۴] Peep-hole connections



شکل (۱-۳): یک بلوک شبکه حافظه کوتاه و بلندمدت. دروازه‌های ورودی، خروجی و فراموشی اغلب دارای تابع فعالساز سیگموئیدی است، همچنین ورودی بلوک و خروجی بلوک اغلب دارای تابع فعالساز تانژانت هایپربولیک است. دایره‌های سیاه کوچک ضرب نقطه‌ای هستند [۷].

معماری شبکه حافظه کوتاه و بلندمدت شامل بلوک‌هایی است که به‌صورت بازگشتی به همدیگر متصل هستند. هر بلوک شامل یک یا چند (اغلب یک) سلول حافظه است که به‌صورت متصل به خود به یکدیگر متصل هستند. بلوک‌ها شامل دروازه ورودی، فراموشی و خروجی می‌باشند که واحدهایی افزاینده هستند. شبکه حافظه کوتاه و بلندمدت و شبکه عصبی بازگشتی بسیار به هم شبیه هستند جز اینکه واحدهای جمع شونده در لایه مخفی شبکه بازگشتی با بلوک‌ها در شبکه حافظه کوتاه و بلندمدت جایگزین شده‌اند. لایه خروجی در هر دو شبکه یکسان است. دروازه‌ها باعث بهبود مشکل ناپدید شدن گرادیان می‌شوند این امر به وسیله این موضوع صورت می‌پذیرد که در شبکه‌های حافظه کوتاه و بلندمدت سلول‌های حافظه اجازه دسترسی و ذخیره اطلاعات برای مدت زمان طولانی را دارند [۷].

هدف این پایان‌نامه بررسی عملکرد توابع فعالساز بر روی دروازه‌های سیگموئیدی بلوک‌های شبکه حافظه کوتاه و بلندمدت است. بدین منظور بر روی مجموعه داده IMDB و Movie Review بهترین معماری (تعداد بلوک در لایه مخفی) این نوع شبکه و همچنین تأثیر و عملکرد انواع توابع فعالساز بر روی نتایج خروجی بررسی می‌گردد و بهترین توابع فعالساز و بازه برای این شبکه معرفی می‌شوند. اهمیت این موضوع از آنجاست که با توجه به عملکرد مناسب شبکه‌های حافظه کوتاه و بلندمدت بر روی رشته‌های ورودی برای وظایفی همچون کلاسه‌بندی و همچنین اهمیت توابع فعالساز بر روی شبکه‌های عصبی این نیاز احساس می‌شود تا با بررسی جامع توابع فعالساز و آزمایش آن‌ها بر روی شبکه‌های حافظه کوتاه و بلندمدت تابع فعالساز مناسبی برای این نوع شبکه‌ها معرفی گردد. با توجه به بررسی انجام شده در موضوع تاکنون مقایسه‌ای بین عملکرد توابع فعالساز بر روی شبکه‌های حافظه کوتاه و بلندمدت صورت نپذیرفته است.

۱–۳– ساختار پایان نامه

در این پایان نامه به بررسی انواع توابع فعالساز بر روی شبکه حافظه کوتاه و بلند مدت پرداخته می شود. بدین منظور در فصل دوم به تاریخچه شبکه حافظه کوتاه و بلند مدت پرداخته می شود به این صورت که ابتدا به کلیات این شبکه پرداخته می شود, وارد جزئیات آن شده و اجزای آن شرح داده می شود. سپس به بررسی معماری و مقایسه انواع شبکه های مشتق شده از شبکه حافظه کوتاه و بلند مدت پرداخته می شود و در نهایت الگوریتم یادگیری و بهینه ساز مورد استفاده در این شبکه شرح داده می شود.

در فصل سوم الگوریتم و توابع فعالساز مورد استفاده در این پایان نامه شرح داده می شود. بدین منظور در ابتدا معماری و الگوریتم شبکه حافظه کوتاه و بلند مدت مورد استفاده در این پایان نامه شرح داده می شود. سپس به بررسی انواع توابع فعالساز مورد استفاده پرداخته شده و شرح داده می شوند. در نهایت دو مجموعه داده مورد استفاده برای ارزیابی نتایج شرح داده می شوند.

در فصل چهارم نتایج خطای اشتباه در رده بندی بر روی دو مجموعه داده مورد استفاده، گزارش می شود و توابع فعالساز پیشنهادی معرفی می گردند. همچنین تعداد بلوک بهینه در لایه مخفی برای هر مجموعه داده معرفی می شود. در نهایت به نتیجه‌گیری پایان نامه پرداخته می شود.

۱-۴- نتیجه‌گیری

در این فصل به مقدمه‌ای از شبکه‌های عصبی و انواع معروف آن پرداخته شد و سپس به صورت اجمالی شبکه حافظه کوتاه و بلندمدت و معماری آن شرح داده شد. در فصل آینده به صورت کامل به طرز کار این شبکه و خواص آن پرداخته خواهد شد.

۲- فصل دوم: تاریخچه شبکه حافظه کوتاه و بلندمدت

۲–۱– مقدمه

در این بخش ابتدا به کلیات و جزئیات شبکه حافظه کوتاه و بلندمدت و مراحل کار این شبکه پرداخته می‌شود. سپس انواع شبکه‌های مشتق شده از شبکه حافظه کوتاه و بلندمدت، اجزا تغییریافته و معماری‌های گوناگون آن معرفی می‌شود و همچنین مقایسه‌ای بین انواع این معماری‌ها و گونه‌های مختلف معرفی می‌شود. درنهایت الگوریتم انتشار رو به عقب طی زمان و الگوریتم بهینه‌سازی روش نرخ یادگیری انطباقی برای این شبکه معرفی می‌شود.

۲–۲– شبکه حافظه کوتاه و بلندمدت

شبکه حافظه کوتاه و بلندمدت برای اولین بار توسط Hochreiter و Schmidhuber در سال ۱۹۹۷ برای غلبه بر مشکل ناپدید شدن گرادیان ارائه شد [۸]. این مدل شبیه یک شبکه عصبی بازگشتی استاندارد با یک‌لایه مخفی است، اما هر گره متداول در لایه مخفی با یک بلوک با یک (یا چند) سلول حافظه جایگزین شده است. مدل شبکه حافظه کوتاه و بلندمدت یک نوع از ذخیره سازی از طریق سلول حافظه را معرفی کرد.

در معماری‌هایی دیگر شبکه‌های حافظه کوتاه و بلندمدت دوطرفه[۱۵] نیز معرفی‌شده است و نتایج مناسب آن روی تشخیص دستخط، کلاسه‌بندی آوا و تشخیص گفتار تأیید شده است [۹, ۱۰, ۱۱]. همچنین این نوع شبکه به رشته‌های مقطع نگاری همدوسی اپتیکی[۱۶] برای کلاسه‌بندی سرطان ریه اعمال شده است و نتایج قابل قبولی ارائه داده است [۱۲, ۱۳]. از معماری‌های دیگر می‌توان از شبکه عصبی بازگشتی با دروازه عمقی[۱۷] برای بهبود ترجمه ماشینی و مدل‌سازی زبان [۱۴], شبکه واحد دروازه‌ای بازگشتی[۱۸] که با ترکیب دروازه‌های ورودی و فراموشی در یک دروازه به نام دروازه به‌روزرسانی باعث

---

[۱۵] Bidirectional LSTM
[۱۶] Optical coherence tomography
[۱۷] DGLSTM
[۱۸] GRU

سادگی بیشتر مدل شبکه حافظه کوتاه و بلندمدت شده است [۱۵] و همچنین شبکه حافظه کوتاه و بلندمدت مشبک[۱۹] به‌صورت شبکه‌ای چندبعدی برای بردارها، رشته‌ها و تصاویر به کار رفته است [۱۶] نیز نام برد. مقالاتی مقایسه‌ای بین معماری‌های شبکه‌های بازگشتی ارائه شده است که می‌توان به مقایسه شبکه بازگشتی بهبود یافته و شبکه حافظه کوتاه و بلندمدت [۱۷] و مقایسه عملکرد معماری‌های مختلف شبکه حافظه کوتاه و بلندمدت [۱۸, ۱۹] اشاره کرد، جایی که نشان داده شده است دروازه فراموشی دارای بیشترین اهمیت است و همچنین اندازه شبکه یکی از با اهمیت‌ترین قسمت‌های قابل تنظیم شبکه حافظه کوتاه و بلندمدت است. از مزایای شبکه‌های حافظه کوتاه و بلندمدت می‌توان به قابلیت سر و کار داشتن با وقفه‌های زمانی طولانی تا ۱۰۰۰۰ مرحله زمانی و همچنین زمان بندی دقیق، تکثیر مقدار دقیق، جمع و حتی ضرب نام برد و به همین دلایل بلوک‌های شبکه حافظه کوتاه و بلندمدت پتانسیل قابلیت‌های یادگیری مهمی را در دسته شبکه بازگشتی آزاد کرده‌اند.

شبکه‌های حافظه کوتاه و بلندمدت و حافظه کوتاه و بلندمدت دوطرفه [۱۱] نتایج مناسبی روی وظایف گوناگون مثل کلاسه‌بندی داشته‌اند. کاربردهای گوناگون این شبکه‌ها شامل شناسایی سری زمانی مربوط به بیماری [۲۰] شناسایی دست خط [۱۰, ۲۱, ۲۲, ۲۳, ۲۴], کلاسه‌بندی آوا [۷, ۱۱, ۲۵] و شناسایی فرمان آنلاین [۲۶] است. شبکه حافظه کوتاه و بلندمدت همچنین برای تولید [۲۷], ترجمه [۲۸], تشخیص احساسات به وسیله شبکه حافظه کوتاه و بلندمدت دوطرفه [۲۹], همچنین مدل‌سازی صوت در گفتار [۳۰], ترکیب گفتار [۳۱], مدل‌سازی زبان [۳۲], پیش‌بینی ساختار پروتئین [۳۳], آنالیز صدا [۳۴] و داده‌های ویدیویی [۳۵] و نیز شناسایی حواس پرتی رانندگان [۳۶] کاربرد دارد.

عموماً عملکرد شبکه‌های عصبی بر اساس معیارهایی گوناگون همچون الگوریتم یادگیری، تابع فعالساز هر گره، تعداد لایه مخفی و گره ها است اما بیشترین تأکید بر روی الگوریتم یادگیری و معماری شبکه است، بنابراین نسبت به اهمیت توابع فعالساز غفلت ورزیده شده است [۳۷, ۳۸, ۳۹]. اگرچه تابع فعالساز می‌تواند بر روی پیچیدگی و عملکرد شبکه‌های عصبی تأثیر داشته باشد و همچنین بر روی همگرایی الگوریتم‌ها تأثیر دارد [۳۸, ۳۹, ۴۰, ۴۱, ۴۲].

[۱۹] GLSTM

اجزای اصلی یک شبکه حافظه کوتاه و بلندمدت شامل موارد زیر هستند.

ورودی بلوک: این واحد که با $\tilde{C}^t$ و در رابطه (۲-۱) نشان داده شده است، یک گره است که فعالساز را از لایه ورودی x^t در مرحله زمانی فعلی و همچنین از لایه مخفی در مرحله زمانی پیشین h^{t-1} می‌گیرد. هر بلوک در لایه مخفی شبکه حافظه کوتاه و بلندمدت به صورت یک مرحله زمانی مستقل در نظر گرفته می شود. به طور مثال در شکل (۱-۳) شبکه دارای ۷ مرحله زمانی است که به صورت پشت سر هم قرار دارند. عموماً جمع وزن های ورودی طی یک تابع فعالساز تانژانت هایپربولیک اجرا می‌شوند.

$$\tilde{C}_t = \tanh\left(W_{\tilde{C}} x_t + U_{\tilde{C}} h_{t-1} + b_{\tilde{C}}\right) \quad (۱-۲)$$

دروازه ورودی: دروازه‌ها[۲۰] ویژگی مشخصه در مدل شبکه حافظه کوتاه و بلندمدت است. دروازه یک واحد سیگموئید است که همانند گره ورودی، فعالساز را از داده‌های حاظر x^t و همچنین لایه مخفی در مرحله زمانی پیشین می‌گیرد. مقدار دروازه ورودی i^t در مقدار گره ورودی ضرب می‌شود.

حالت سلول[۲۱]: در مرکز هر بلوک یک گره C^t وجود دارد. حالت سلول C^t دارای یک نوار بازگشتی متصل به خود است. به دلیل اینکه این نوار، در امتداد مراحل زمانی دارای عملگر جمع است و خطا می‌تواند طی مراحل زمانی بدون ناپدید شدن یا گسترده شدن جریان یابد. این نوار اغلب چرخش خطای ثابت[۲۲] نامیده می‌شود. در به‌روزرسانی برای حالت سلول، نماد $\odot$ یک ضرب نقطه‌ای است.

دروازه فراموشی: این دروازه f^t, توسط Gers و همکارانش [۴۳] معرفی شده‌اند. یک روش است که به وسیله آن شبکه می‌تواند یاد بگیرد که محتوای حالت سلول را تخلیه کند. با دروازه‌های فراموشی، معادله (۲-۲) برای محاسبه حالتِ سلول در عبورِ روبه‌جلو نشان داده شده است.

$$C^t = f^t \odot C^{t-1} + i^t \odot \tilde{C}^t \quad (۲-۲)$$

خروجی بلوک: درنهایت مقدار h^t یا خروجی بلوک توسط یک سلول که مقدار حالت سلول (C^t) ضربدر مقدار دروازه خروجی (o^t) است حساب می‌شود. مرسوم است که حالت سلول ابتدا طی یک تابع

---

[۲۰] Gate

[۲۱] Cell state

[۲۲] Constant error carousel

فعالساز تانژانت هایپربولیک اجرا شود که با این کار خروجیِ هر سلول را به‌صورت یک دامنه یکسان به دست می‌دهد. در بعضی دیگر از تحقیقات تابع ReLU که دامنه بزرگ‌تری دارد استفاده شده است که برای آموزش ساده‌تر است. خروجی بلوک در رابطه (۲-۳) نمایش داده شده است.

$$h^t = o^t \odot \tanh\left(C^t\right) \quad (۳-۲)$$

در اینجا از i, f و o به ترتیب برای ارجاع به دروازه‌های ورودی، فراموشی و خروجی استفاده شده است [۴۴] که به ترتیب در روابط (۲-۴) تا (۲-۶) نمایش داده شده اند.

$$i^t = \sigma\left(W_i.x^t + U_i.h^{t-1} + b_i\right) \quad (۴-۲)$$

$$f^t = \sigma\left(W_f.x^t + U_f.h^{t-1} + b_f\right) \quad (۵-۲)$$

$$o^t = \sigma\left(W_o.x^t + U_o.h^{t-1} + b_o\right) \quad (۶-۲)$$

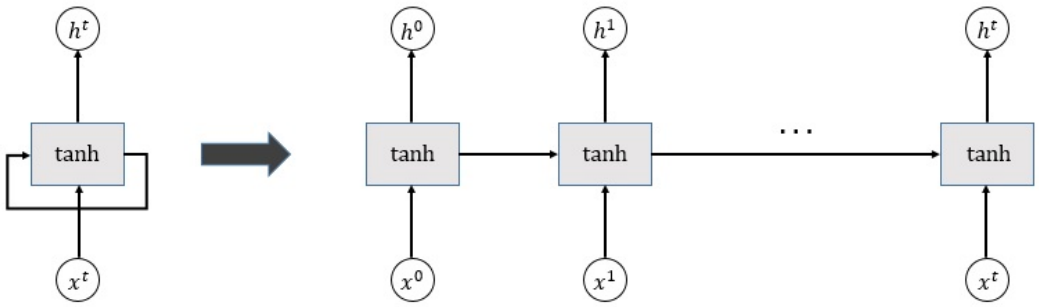
از زمان ارائه شبکه حافظه کوتاه و بلندمدت متغیرهای گوناگونی معرفی شده‌اند. دروازه‌های فراموشی در سال ۲۰۰۰ معرفی شد و جزو طراحی شبکه حافظه کوتاه و بلندمدت اصلی نبودند اما بسیار سودمند هستند. در همین سال اتصالات روزنه‌ای [۴۳] معرفی شد که مستقیماً از حالت سلول به دروازه‌های بلوک متصل است. این اتصالات عملکرد روی زمان بندی، جایی که شبکه باید اندازه گیری‌های دقیق داخلی بین رخدادها را یاد بگیرد را بهبود می‌دهند.

۲-۳- جزئیات شبکه حافظه کوتاه و بلندمدت

همه شبکه‌های عصبی بازگشتی، فرمی به شکل یک زنجیره از بلوک‌های تکرار شونده از شبکه هستند. در شبکه عصبی بازگشتی استاندارد این تکرار بلوک، ساختاری بسیار ساده مثل یک‌لایه تانژانت هایپربولیک دارد. مشکل شبکه بازگشتی ساده (و تقریبا تمام شبکه های چندلایه عمیق) این است که بازه محتوای ورودی محدود است زیرا تاثیر ورودی در لایه مخفی و خروجی در اتصال های بازگشتی شبکه می تواند بسیار کوچک یا بزرگ شود که به این مشکل ناپدید شدن گرادیان می گویند. برای حل مشکل ناپدید شدن گرادیان در شبکه حافظه کوتاه و بلند مدت هر بلوک با ذخیره و دسترسی اطلاعات در طول زمان طولانی باعث جلوگیری از این مشکل می شوند. شکل (۲-۱) نشان دهنده یک شبکه بازگشتی با یک لایه مخفی است که خروجی هر بلوک در هر مرحله زمانی، ورودی بلوک در مرحله زمانی بعدی است. در

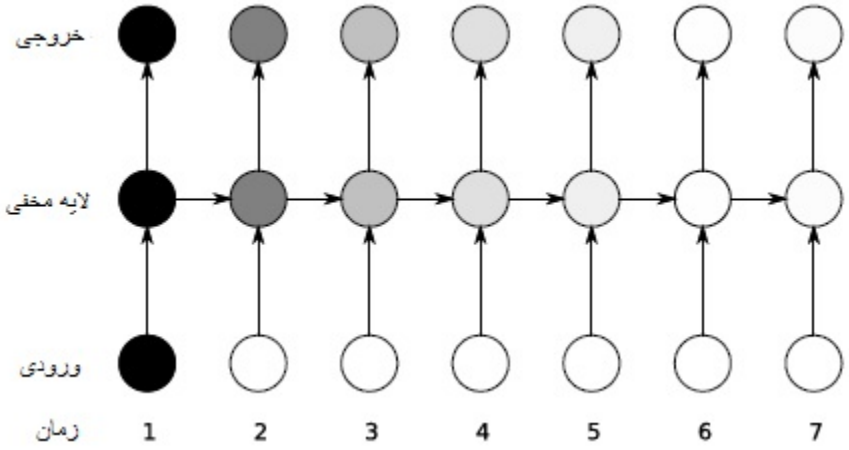
شبکه های عصبی که براساس روش های گرادیان[۲۳] و انتشار رو به عقب[۲۴] آموزش می بینند هر ماتریس وزن یک بروزرسانی به نسبت گرادیان تابع در طول آموزش دریافت می کند. توابع فعالساز معروف مثل تانژانت هایپربولیک و سیگموئید در بازه [۱،۱-] و [۰،۱] هستند و بازه گرادیان آن ها حتی کوچکتر هم می شود برای مثال بازه مشتق تابع سیگموئید [۰،۰/۲۵] است و در انتشار رو به عقب, گرادیان ها را براساس قانون زنجیره ای[۲۵] به صورت ضرب آن ها محاسبه می کند. این موضوع باعث می شود که عدد های کوچک گرادیان محاسبه شده در هر لایه در لایه های بعدی (گره ها) ضرب شود و طی چند لایه یا مرحله زمانی به صورت نمایی کاهش پیدا کند، در نتیجه بعد از چند مرحله گرادیان به شدت کاهش پیدا می کند و شبکه فراموش می کند که دقیقا به دنبال چه چیزی بوده است [۷]. شکل (۲-۲) نشان دهنده مشکل ناپدید شدن گرادیان در شبکه های بازگشتی است.

در بلوک شبکه حافظه کوتاه و بلندمدت دو عملیات ضرب و جمع برای تبدیل ورودی به خروجی وجود دارد. وجود علامت جمع در رابطه (۲-۲) راه حل مشکل ناپدید شدن گرادیان در این شبکه است. این کار باعث می شود که هنگام انتشار رو به عقب خطا ثابت بماند. به جای تعیین حالت سلول پسین به وسیله ضرب حالت کنونی اش در ورودی جدید, این دو با هم جمع می شوند. به این صورت اصطلاحا خطا در حافظه سلول گیر می افتد که از آن به نام چرخش خطای ثابت یاد می شود.



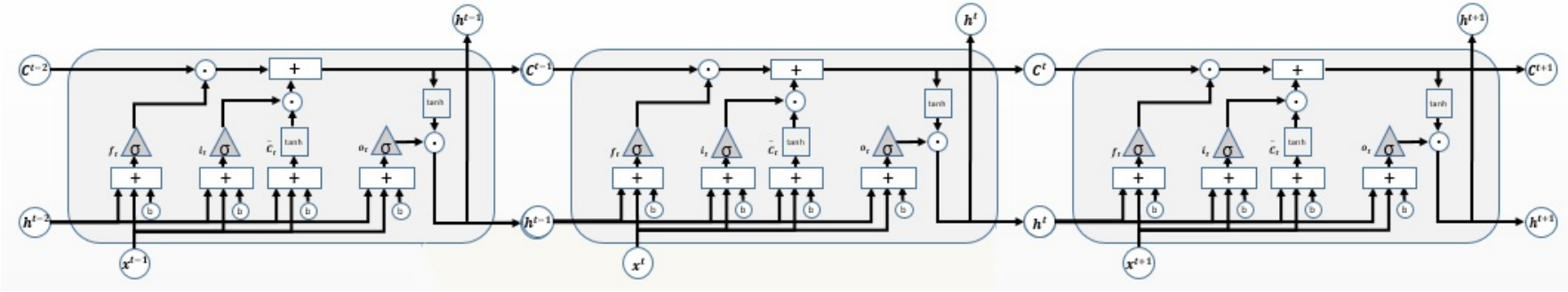
[۲۳] Gradient-based
[۲۴] Backpropagation
[۲۵] Chain rule

شکل (۲-۱): بلوک تکرار شونده در یک شبکه بازگشتی استاندارد شامل یک‌لایه مخفی.



شکل (۲-۲): مشکل ناپدید شدن گرادیان در شبکه های عصبی بازگشتی استاندارد. حاشور های پُررنگ و کم رنگ نشان دهنده میزان حساسیت به ورودی در زمان است. حاشور پُر رنگ تر به معنی حساسیت بیشتر است. میزان حساسیت طی زمان, هنگامی که ورودی های جدید بر روی فعالساز های لایه مخفی بازنویسی می شوند کاهش می یابد و در نتیجه شبکه, ورودی ابتدایی را فراموش می کند [۷].

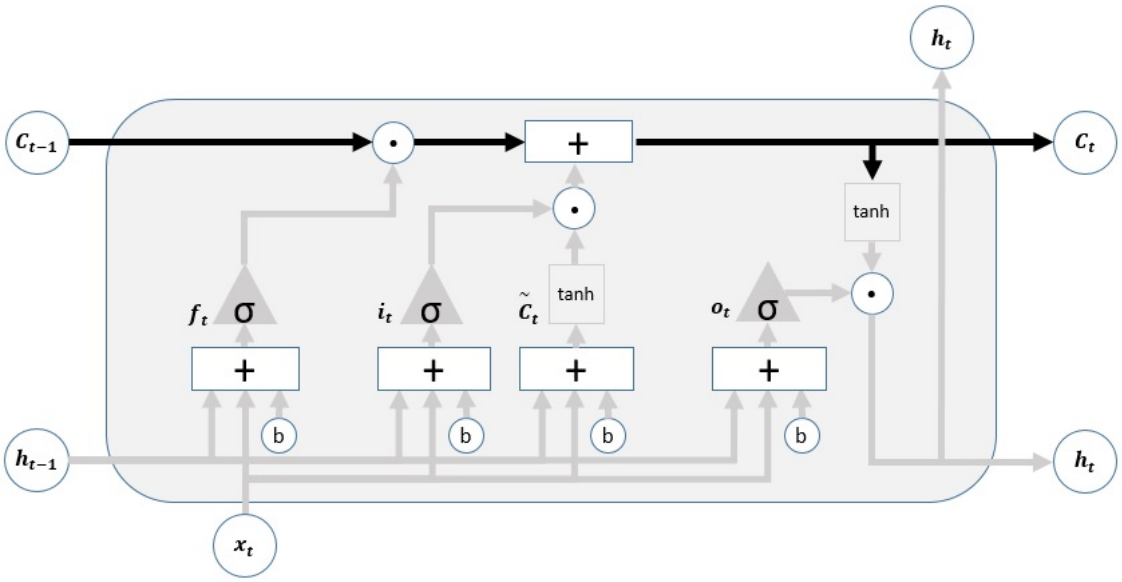
شبکه حافظه کوتاه و بلندمدت ساختار زنجیره‌ای به شکل شبکه بازگشتی دارد، اما بلوک تکرار شونده دارای یک ساختار متفاوت است. شکل (۲-۳) نشان دهنده بلوک تکرار شونده در این شبکه است. به‌جای داشتن یک‌لایه شبکه عصبی واحد، این شبکه شامل چهار لایه است که در مسیری بسیار خاص عمل می‌کنند.



شکل (۲-۳): بلوک تکرار شونده در یک شبکه حافظه کوتاه و بلندمدت شامل چهار لایه با اثر متقابل.

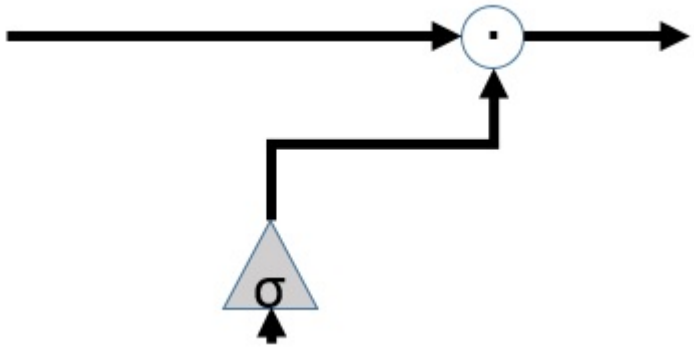
در شکل (۲-۴) هر خط، یک بردار کامل را از خروجی یک گره به ورودی‌های دیگر حمل می‌کند. نماد ⊙ نشان‌دهنده عملگرهای نقطه‌ای ضرب بردار هستند، در حالی که سه مثلث و یک مربع (قسمت پایین بلوک) لایه‌های شبکه عصبی یاد گرفته شده هستند.

موضوع کلیدی در شبکه‌های شبکه حافظه کوتاه و بلندمدت حالت سلول یا خط افقی بالای شکل است. حالت سلول شبیه به یک نوار نقاله است و از کل زنجیره، با کمی عملیات‌های خطی کوچک به‌صورت مستقیم عبور می‌کند و برای اطلاعات بسیار ساده است که طی آن بدون تغییر، جریان پیدا کنند.



شکل (۲-۴): حالت سلول در یک شبکه حافظه کوتاه و بلندمدت.

شبکه حافظه کوتاه و بلندمدت قابلیت حذف یا اضافه کردن اطلاعات، به‌صورت به دقت تنظیم شده با ساختارهایی به نام دروازه‌ها, به حالت سلول را دارد. دروازه‌ها یک راه هستند که به‌صورت اختیاری اجازه عبور اطلاعات را می‌دهند. آن‌ها از یک‌لایه سیگموئیدی و یک عملگر ضرب نقطه‌ای ساخته شده‌اند. شکل (۲-۵) نشان دهنده یک دروازه است.



شکل (۲-۵): نمایش یک دروازه در یک شبکه حافظه کوتاه و بلندمدت.

لایه دروازه سیگموئیدی اعدادی بین صفر و یک (با توجه به تابع فعالساز مورد استفاده در آن) را در خروجی به وجود می آورد. این اعداد نشان‌دهنده این هستند که هر کدام از اجزا چه مقداری را باید عبور بدهند. مقدار صفر به معنی این است که هیچ چیزی عبور نکند و مقدار یک به معنی این است که همه اطلاعات عبور کند. یک شبکه حافظه کوتاه و بلندمدت دارای سه دروازه از این نوع (دروازه‌های سیگموئیدی) برای حفاظت و کنترل حالت سلول است.

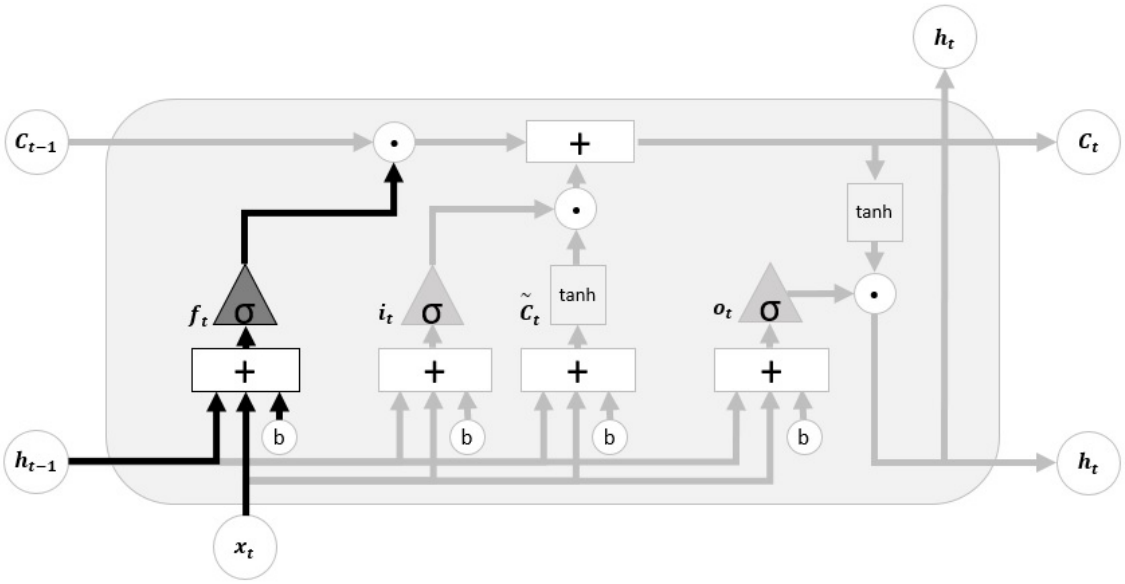
۲-۴- مراحل کار شبکه حافظه کوتاه و بلندمدت

در مراحل زیر f، i و o به ترتیب نشان‌دهنده دروازه فراموشی، دروازه ورودی و دروازه خروجی هستند. $\tilde{C}^t$ ورودی بلوک در زمان است. C^t حالت سلول حافظه در زمان است. h^t خروجی بلوک در زمان است. x^t ورودی در زمان است. W و U ماتریس‌های وزن هستند و b بردار بایاس است. نماد $\odot$ ضرب نقطه‌ای دو بردار است. توابع σ و tanh توابع فعالساز سیگموئید و تانژانت هایپربولیک هستند.

مرحله ۱: مرحله اول تصمیم‌گیری برای این است که کدام اطلاعات از حالت سلول دور ریخته شوند. این تصمیم با یک‌لایه سیگموئیدی به نام «دروازه فراموشی» گرفته می‌شود. این لایه به خروجی بلوک در زمان پیشین (h^{t-1}) و ورودی در زمان حال (x^t) نگاه می‌کند و یک عدد بین صفر و یک برای هر عدد

ورودی در حالت سلول پیشین (C^{t-1}) خارج می‌کند. شکل (۲-۶) و رابطه (۲-۷) نشان دهنده دروازه فراموشی است.

$$f^t = \sigma\left(W_f . x^t + U_f . h^{t-1} + b_f\right) \quad (۲-۷)$$

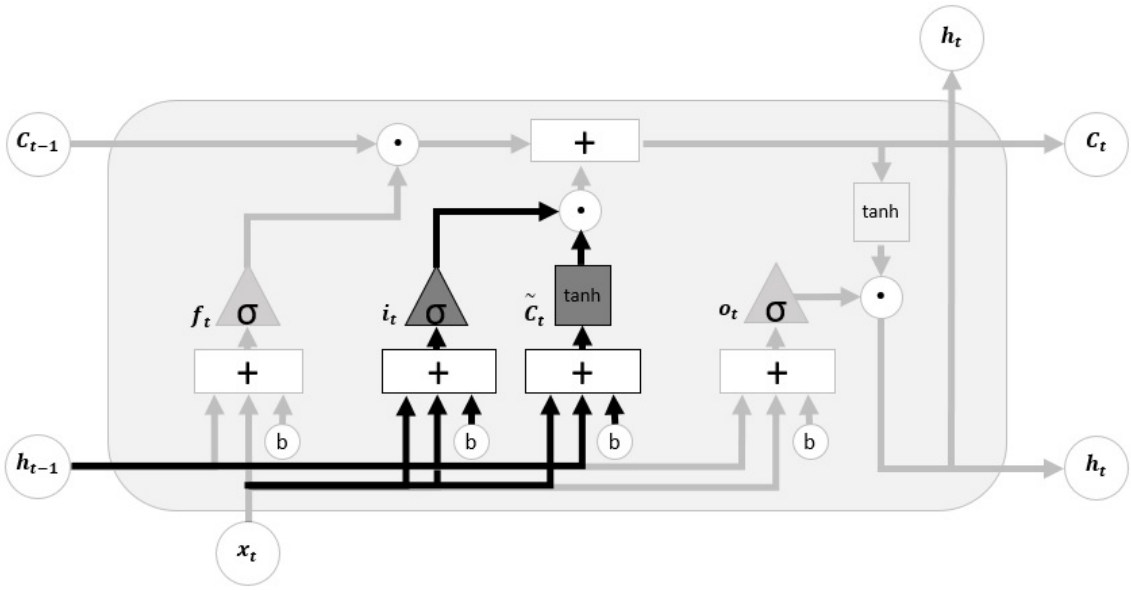


شکل (۲-۶): نمایش لایه دروازه فراموشی در شبکه حافظه کوتاه و بلندمدت.

مرحله ۲: مرحله دوم تصمیم‌گیری درباره این است که چه اطلاعات جدیدی می‌بایست در حالت سلول ذخیره شود. این مرحله دو قسمت دارد، ابتدا یک‌لایه سیگموئیدی به نام «دروازه ورودی» تصمیم می‌گیرد کدام مقادیر می‌بایست بروزرسانی شود، سپس یک‌لایه تانژانت هایپربولیک یک بردار از مقادیر کاندید جدید به عنوان ورودی بلوک ($\tilde{C}^t$) می‌سازد که می‌تواند به حالت سلول اضافه شود. در مرحله بعد این دولایه برای بروزرسانی حالت سلول با هم ترکیب می‌شوند. شکل (۲-۷) و روابط (۲-۸) و (۲-۹) نشان دهنده دروازه ورودی و ورودی بلاک است.

$$i^t = \sigma\left(W_i . x^t + U_i . h^{t-1} + b_i\right) \quad (۲-۸)$$

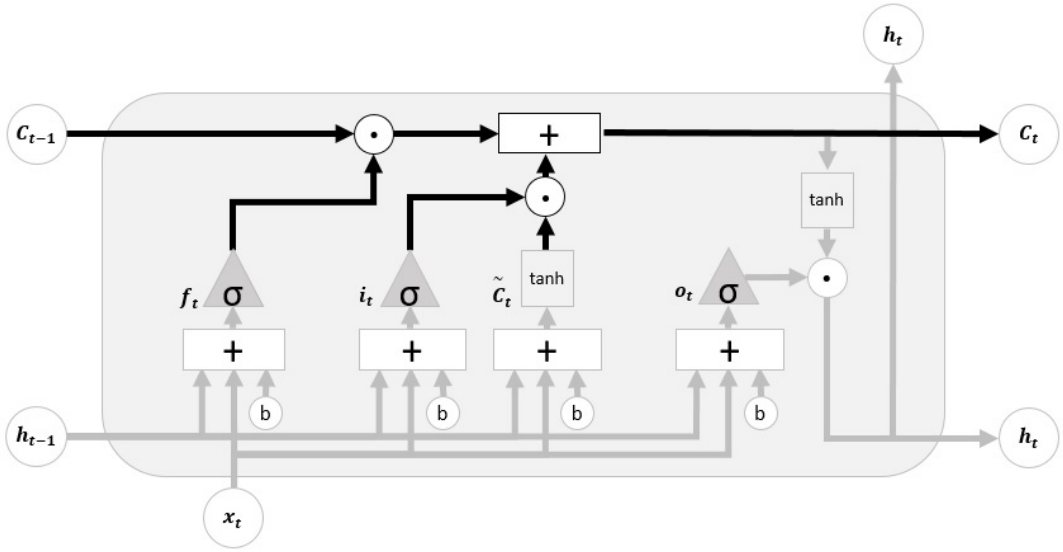
$$\tilde{C}^t = \tanh\left(W_{\tilde{C}} . x^t + U_{\tilde{C}} . h^{t-1} + b_{\tilde{C}}\right) \quad (۲-۹)$$



شکل (۲-۷): نمایش لایه دروازه ورودی و لایه تانژانت هایپربولیک برای تصمیم‌گیری در مورد اینکه چه اطلاعاتی به شبکه حافظه کوتاه و بلندمدت می‌بایست اضافه شود.

حال زمان این است که حالت سلول قدیم C^{t-1} به حالت سلول جدید C^t بروزرسانی شود. در مراحل قبل تصمیم به چگونگی این کار گرفته شده است و اینجا عملاً انجام می‌پذیرد. با ضرب حالت قدیم در دروازه فراموشی f^t اطلاعاتی که قبلاً می‌بایست فراموش شوند، فراموش می‌شوند. سپس ضرب دروازه ورودی در ورودی بلوک $i^t \odot \tilde{C}^t$ به آن اضافه می‌شود. این مقادیر جدید کاندید است که نشان می دهد شبکه, هر مقدار حالت سلول را به چه اندازه می‌خواهد به‌روزرسانی کند. شکل (۲-۸) و رابطه (۲-۱۰) نشان دهنده به‌روزرسانی توسط حالت سلول است.

$$C^t = f^t \odot C^{t-1} + i^t \odot \tilde{C}^t \qquad (۱۰-۲)$$

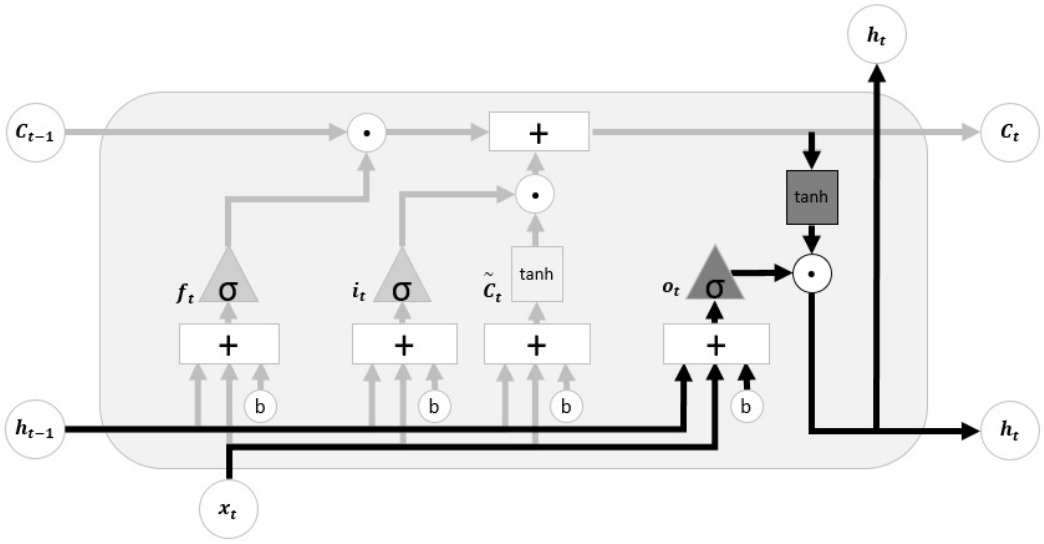


شکل (۲-۸): به‌روزرسانی حالت سلول قدیم به جدید در شبکه حافظه کوتاه و بلندمدت.

مرحله ۳: درنهایت می‌بایست تصمیم گرفته شود که چه چیزی در خروجی وجود داشته باشد. این خروجی بر اساس حالت سلول خواهد بود. ابتدا یک‌لایه سیگموئیدی به نام دروازه خروجی o^t اجرا می‌شود که تصمیم می‌گیرد چه بخش‌هایی از حالت سلول می‌بایست به خروجی داده شود، سپس حالت سلول در یک لایه تانژانت هایپربولیک گذاشته می‌شود (تا مقادیر بین ۱ و ۱- قرار بگیرد) و در دروازه خروجی ضرب می‌شود؛ بنابراین فقط قسمت‌های موردنظر و تصمیم گرفته شده به خروجی می‌روند. شکل (۲-۹) و روابط (۲-۱۱) و (۲-۱۲) نشان دهنده دروازه خروجی و خروجی بلوک (h^t) است.

$$o^t = \sigma\left(W_o x^t + U_o . h^{t-1} + b_o\right) \qquad (۱۱-۲)$$

$$h^t = o^t \odot \tanh\left(C^t\right) \qquad (۱۲-۲)$$



شکل (۲-۹): تصمیم در مورد خروجی شبکه حافظه کوتاه و بلندمدت.

۲-۵- تغییرات در شبکه های حافظه کوتاه و بلند مدت

معماری استاندارد شبکه حافظه کوتاه و بلند مدت دارای تغییرات بسیار زیادی شده است و تحقیقات گسترده ای در سال های اخیر روی عملکرد معماری های گوناگون آن صورت گرفته است و ابداع معماری های جدید روی این ادامه دارد. برای نشان دادن اهمیت این نوع شبکه در ادامه به معرفی چند معماری و تغییر معروف در این شبکه پرداخته می شود.

۲-۵-۱- شبکه حافظه کوتاه و بلندمدت دوطرفه

معماری شبکه بازگشتی استاندارد به‌عنوان یک طرفه شناخته می‌شود. این بدان معنا است که رشته داده ورودی به یک شبکه در جهت مستقیم داده می‌شود. در نتیجه سابقه ورودی، در یک «حالت گذشته» شبکه انباشته می‌گردد. به هرحال مسائلی وجود دارد که تنها حالت گذشته برای ارضای یادگیری مساله کافی نیست. از این جهت بهتر است یک محتوای آینده نیز به‌عنوان ورودی دریافت شود. این نوع محتوای آینده می‌تواند فقط زمانی تولید شود که رشته‌های ورودی به‌طور کامل در زمان محاسبه موجود باشند. برای اینکه محتوای گذشته و محتوای آینده در شبکه‌های بازگشتی در دسترس باشند ایده

شبکه عصبی بازگشتی دوطرفه ارائه شد. اگر یک شبکه کلاسیک بازگشتی با یک‌لایه مخفی بازگشتی در نظر گرفته شود، سپس یک‌لایه مخفی بازگشتی دوم به لایه ورودی متصل و اضافه گردد و همچنین به لایه خروجی نیز متصل گردد اما به لایه مخفی دیگر متصل نشود، در این حالت شبکه دوطرفه شده است. در حالی که روند محاسباتی در یک شبکه بازگشتی کلاسیک مستقیم است، محاسبه در یک شبکه بازگشتی دوطرفه سه لایه است. در ابتدا یک رشته ورودی به‌صورت مستقیم به لایه ورودی اعمال می‌شود، سپس فقط لایه مخفی اول محاسبه می‌شود و همه فعالساز هایش برای هر مرحله زمانی ذخیره می‌گردد. در مرحله بعد رشته ورودی به‌صورت برعکس شده (در جهت وارونه) ارائه می‌شود. در اینجا فقط لایه مخفی دوم عمل می‌کند و همه فعالساز هایش ذخیره می‌گردد. درنهایت لایه خروجی رشته خروجی را به وسیله ترکیب اطلاعات رسیده از گذشته (تولید شده به وسیله لایه مخفی اول) و اطلاعات رسیده از آینده (تولید شده به وسیله لایه مخفی دوم) در هر مرحله زمانی تولید می‌کند. توجه شود که محاسبه گرادیان در لایه مخفی وارونه شده مثل حالت قبل ولی به‌صورت وارونه در زمان عمل خواهد کرد. این ایده اساس ایجاد شبکه حافظه کوتاه و بلندمدت دوطرفه بود [۲۰].

پس در این معماری دولایه از بلوک‌های مخفی وجود دارد. هر دولایه مخفی به ورودی و خروجی متصل هستند. دولایه مخفی تفکیک شده هستند، جایی که اولین لایه مخفی اتصالات بازگشتی از مراحل زمانی گذشته را دارد در حالی که در دومین لایه مخفی جهت بازگشتی اتصالات برعکس شده‌اند، فعالساز به‌صورت رو به عقب در طول رشته حرکت می‌کند. شبکه به‌صورت عادی با انتشار رو به عقب با رشته ورودی و خروجی آموزش می‌بینند. سه معادله (۲-۱۳) تا (۲-۱۵) یک شبکه بازگشتی دوطرفه را توصیف می‌کند. جایی که h^t و z^t به ترتیب مقادیر خروجی لایه‌های مخفی در جهت روبه‌جلو و رو به عقب هستند.

$$h^t = \sigma(W_{hx}x^t + W_{hh}h^{t-1} + b_h) \qquad (۱۳-۲)$$

$$z^t = \sigma(W_{zx}x^t + W_{zz}z^{t+1} + b_z) \qquad (۱۴-۲)$$

$$y^t = \mathrm{softmax}(\mathrm{W}_{yh}h^t + W_{yz}z^t + b_y) \qquad (۱۵-۲)$$

یکی از محدودیت‌های شبکه بازگشتی دوطرفه این است که نمی‌تواند به‌صورت مداوم اجرا شود زیرا به یک نقطه مشخص پایانی در آینده و گذشته احتیاج دارد. همچنین الگوریتم یادگیری ماشین مناسبی

برای تنظیمات آنلاین وجود ندارد و این باعث می‌شود که دریافت اطلاعات از آینده، برای مثال شناخت اجزای رشته‌ای که مشاهده نشده است، نامحتمل باشد؛ اما برای پیش‌بینی روی یک رشته با طول ثابت، داشتن گزارش هر دو اجزای رشته گذشته و آینده نمایان است. اگر یک وظیفه زبان طبیعی مبنی بر برچسب زدن «قسمتی از گفتار» در نظر شود، با گرفتن هر کلمه در یک رشته، اطلاعات همه کلمه‌ها که جلوتر بودند و آن‌هایی که بعد از آن هستند برای پیش‌بینی آن کلمه در «قسمتی از گفتار» مفید است. شبکه حافظه کوتاه و بلندمدت دوطرفه نیز به همین صورت عمل می‌کند که نتایج مناسب آن روی تشخیص دستخط و کلاسه‌بندی آوا وجود دارد [۱۰, ۱۱].

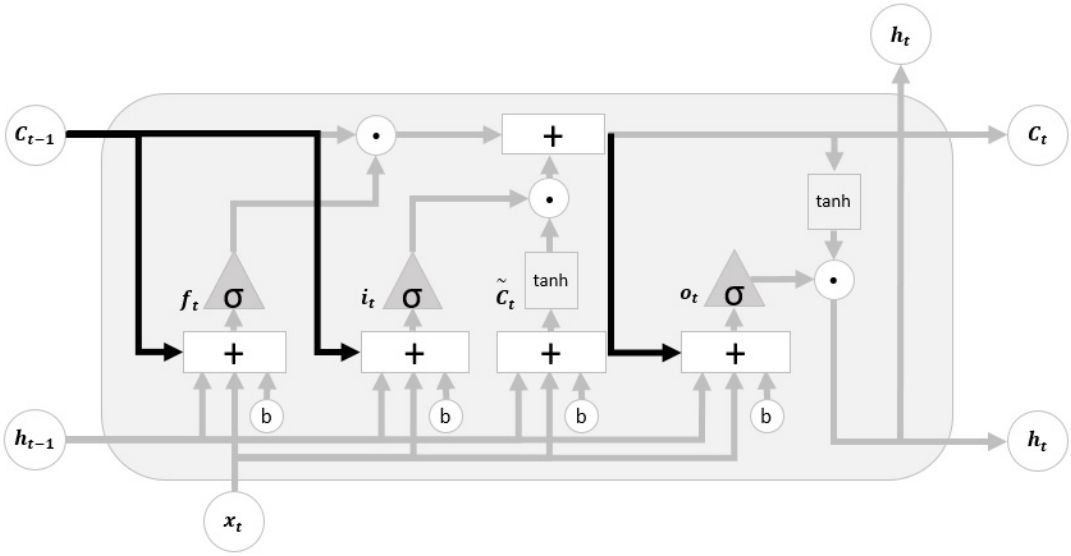
۲-۵-۲- اتصالات روزنه‌ای

یک تغییر معروف شبکه حافظه کوتاه و بلندمدت اضافه کردن اتصالات روزنه‌ای است [۴۳]. بدین معنا که اجازه داده شود تا دروازه‌ها به حالت سلول نگاه کنند. شکل (۲-۱۰) و روابط (۲-۱۶) تا (۲-۱۸) نشان دهنده اتصالات روزنه ای است، در این شکل روزنه‌هایی به همه دروازه‌ها اضافه‌شده‌اند اما در بسیاری از مقالات فقط بعضی از دروازه‌ها دارای اتصالات روزنه‌ای هستند.

$$f^t = \sigma(W_f.[C^{t-1}, h^{t-1}, x^t] + b_f) \quad (۱۶-۲)$$

$$i^t = \sigma(W_i.[C^{t-1}, h^{t-1}, x^t] + b_i) \quad (۱۷-۲)$$

$$o^t = \sigma(W_o.[C^t, h^{t-1}, x^t] + b_o) \quad (۱۸-۲)$$

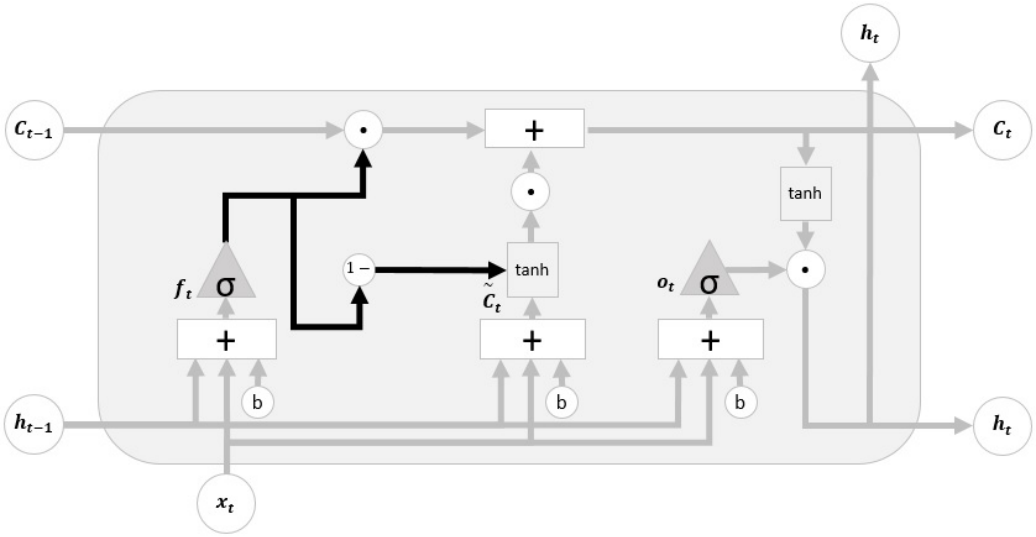


شکل (۲-۱۰): اتصالات روزنه‌ای در شبکه حافظه کوتاه و بلندمدت.

۲-۵-۳- ورودی زوج شده

تغییر دیگر استفاده از دروازه‌های فراموشی و ورودی زوج شده است. در اینجا به‌جای تصمیم به‌صورت جداگانه برای اینکه چه چیزی فراموش شود و چه اطلاعات جدیدی می‌بایست اضافه شود، این تصمیم‌ها باهم گرفته می‌شوند. تنها زمانی اطلاعات فراموش می‌شوند که چیزی در ورودی به‌جای آن وارد شود. تنها زمانی مقادیر جدید به حالت سلول وارد می‌شوند که اطلاعاتی در گذشته فراموش‌شده باشد. شکل (۲-۱۱) و رابطه (۲-۱۹) نشان دهنده ورودی زوج شده هستند.

$$C^t = f^t \odot C^{t-1} + (1 - f^t) \odot \tilde{C}^t \qquad (۱۹-۲)$$



شکل (۲-۱۱): دروازه‌های فراموشی و ورودی زوج شده در شبکه حافظه کوتاه و بلندمدت.

۲-۵-٤- شبکه‌های حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی

یک تغییر دیگر در شبکه‌های حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی[۲۶] است [۱۵]. این تغییر دروازه‌های فراموشی و ورودی را در یک دروازه واحد به نام «دروازه به‌روزرسانی» ترکیب می‌کند. همچنین حالت سلول و حالت مخفی به هم الحاق می‌شوند. شبکه ایجادشده ساده‌تر از شبکه حافظه کوتاه و بلندمدت استاندارد است. شکل (۲-۱۲) و روابط (۲-۲۰) تا (۲-۲۳) شبکه واحد دروازه ای بازگشتی است.

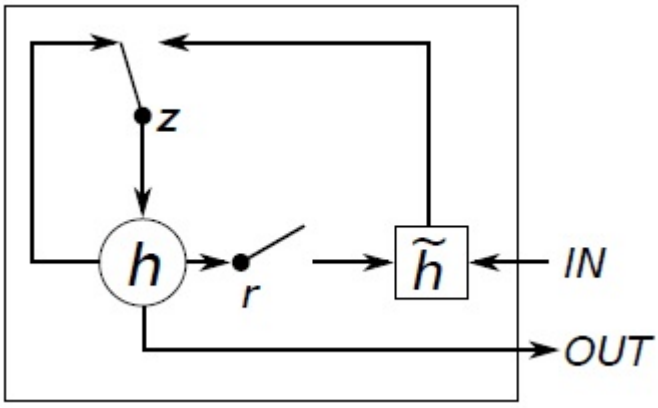
$$z^t = \sigma(W_z.[h^{t-1}, x^t]) \quad (۲۰-۲)$$

$$r^t = \sigma(W_r.[h^{t-1}, x^t]) \quad (۲۱-۲)$$

$$\tilde{h}^t = \tanh(W .[r^t \odot h^{t-1}, x^t]) \quad (۲۲-۲)$$

$$h^t = (1 - z^t) \odot h^{t-1} + z^t \odot \tilde{h}^t \quad (۲۳-۲)$$

[۲۶] GRU



شکل (۲-۱۲): مدل تغییریافته شبکه حافظه کوتاه و بلندمدت به نام شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی [۱۵].

۲-۶- مقایسه عملکرد گونه‌های مختلف شبکه حافظه کوتاه و بلندمدت

مقایسه عملکرد گونه‌های مختلف شبکه حافظه کوتاه و بلندمدت متفاوت توسط Jozefowicz و همکارانش ارائه شده است [۱۸]. گونه‌های گوناگونی از معماری شبکه حافظه کوتاه و بلندمدت برای شبکه‌های بازگشتی تاکنون ارائه شده‌اند. در اینجا آنالیزی گسترده از ۸ گونه شبکه حافظه کوتاه و بلندمدت روی وظایفی همچون تشخیص گفتار، تشخیص دست‌نویس و مدل‌سازی چندصدایی موسیقی پرداخته شده است. نشان داده است که هیچ‌گونه‌ای نمی‌تواند به‌صورت خیلی خوب معماری شبکه حافظه کوتاه و بلندمدت استاندارد [۱۱] را بهبود ببخشد و همچنین نشان داده است که دروازه فراموشی و تابع فعالساز خروجی دارای بیشترین اهمیت در بین اجزای این شبکه هستند.

۸ گونه مختلف شبکه حافظه کوتاه و بلندمدت در اینجا آزمایش شده‌اند.

۱. بدون دروازه ورودی (NIG).

۲. بدون دروازه فراموشی (NFG).

۳. بدون دروازه خروجی (NOG).

۴. بدون تابع فعالساز ورودی (NIAF).

۵. بدون تابع فعالساز خروجی (NOAF).

۶. بدون اتصالات روزنه‌ای (NP).

۷. دروازه ورودی و فراموشی زوج شده (CIFG).

۸. دروازه کاملاً بازگشتی (FGR).

گونه CIFG تنها از یک دروازه برای هر دو ورودی و سلول بازگشتی متصل به خود استفاده می‌کند (شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی). این برابر با تنظیمات $f_t = 1 - i_t$ به‌جای یادگیری وزن‌های دروازه فراموشی به‌صورت مستقل است. گونه FGR اتصالات بازگشتی بین همه دروازه‌ها در معماری استاندارد شبکه حافظه کوتاه و بلندمدت اضافه می‌کند. این کار باعث اضافه شدن ۹ ماتریس وزن بازگشتی اضافی می‌شود بنابراین به‌صورت معنی‌دار تعداد پارامترها را افزایش می‌دهد.

نتایج نشان داد که معماری شبکه حافظه کوتاه و بلندمدت استاندارد به‌خوبی روی مجموعه داده‌های گوناگون عمل می‌کند و استفاده از هر کدام از هشت گونه تغییریافته تغییر مهمی روی بهبود عملکرد شبکه حافظه کوتاه و بلندمدت ندارد. تغییرات مشخص مثل زوج کردن دروازه‌های ورودی و خروجی یا حذف اتصالات روزنه‌ای شبکه حافظه کوتاه و بلندمدت را بدون صدمه مهم به عملکرد شبکه ساده‌سازی کرده است. دروازه فراموشی و تابع فعالساز خروجی مهم‌ترین اجزای بلوک شبکه حافظه کوتاه و بلندمدت هستند. نرخ یادگیری و اندازه شبکه مهم‌ترین پارامترهای قابل تغییر شبکه حافظه کوتاه و بلندمدت هستند و در نهایت استفاده از مومنتوم بی‌اهمیت به نظر می‌رسد (برای تنظیمات گرادیان نزولی آنلاین).

۲-۶-۱- مقایسه معماری‌های شبکه‌های عصبی بازگشتی و شبکه حافظه کوتاه و بلندمدت

مقایسه دیگر مقایسه معماری‌های شبکه‌های عصبی بازگشتی و شبکه حافظه کوتاه و بلندمدت است که توسط Greff و همکارانش معرفی‌شده است [۱۹]. در اینجا تعیین می‌شود که آیا شبکه حافظه کوتاه و بلندمدت بهینه است یا معماری‌های بهتری نیز وجود دارند. این کار با ده هزار معماری مختلف شبکه‌های بازگشتی صورت پذیرفت. نشان داده شده است که اضافه کردن یک بایاس ۱ به دروازه فراموشی فاصله بین شبکه حافظه کوتاه و بلندمدت و شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی را از بین می‌برد.

جزئیات فنی در شبکه حافظه کوتاه و بلندمدت وجود دارد که یکی از مهم‌ترین آن‌ها مقداردهی اولیه بایاس دروازه فراموشی b_f است. در بیشتر معماری‌ها، شبکه حافظه کوتاه و بلندمدت با مقادیر وزن رندم کوچک مقداردهی اولیه می‌شوند که البته روی بسیاری از مسائل به‌خوبی کار می‌کند؛ اما این مقداردهی اولیه به‌صورت مؤثری دروازه فراموشی را به ۰/۵ تنظیم می‌کند. این باعث می‌شود تا گرادیان با فاکتور ۰/۵ بر مرحله زمانی ناپدید گردد که باعث می‌شود تا هر وقت مبحث وابستگی‌های طولانی‌مدت پیش بیاید این موضوع مشکل‌ساز باشد. این مشکل با مقداردهی اولیه دروازه فراموشی b_f به مقادیری مثل ۱ یا ۲ حل می‌شود. با این کار دروازه فراموشی با مقادیری نزدیک به ۱ مقداردهی اولیه می‌شود و باعث می‌شود تا جریان گرادیان فعال شود. اگر بایاس دروازه فراموشی به صورتی مناسب مقداردهی اولیه نشود باعث مشکلاتی در یادگیری شبکه خواهد شد و وابستگی‌های طولانی‌مدت دچار مشکل خواهند شد. در اینجا مقایسه‌ای بین شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی و شبکه حافظه کوتاه و بلندمدت صورت پذیرفته است که نشان می‌دهد شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی روی بیشتر وظایف غیر از مدل‌سازی زبان بهتر عمل می‌کند اما اگر بایاس دروازه فراموشی به مقدار ۱ مقداردهی اولیه شود نتایج دو شبکه بسیار شبیه به هم خواهند شد.

معماری شبکه‌های MUT شبیه به شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی به‌صورت معادلات (۲-۲۴) تا (۲-۳۲) است.

MUT1:

$$z = \sigma(W_{xz} x^t + b_z) \quad (۲-۲۴)$$

$$r = \sigma(W_{xr} x^t + W_{hr} h^t + b_r) \quad (۲-۲۵)$$

$$h^{t+1} = \tanh(W_{hh}(r \odot h^t) + \tanh(x^t) + b_h) \odot z + h^t \odot (1 - z) \quad (۲-۲۶)$$

MUT2:

$$z = \sigma(W_{xz} x^t + W_{hz} h^t + b_z) \quad (۲-۲۷)$$

$$r = \sigma(x^t + W_{hr} h^t + b_r) \quad (۲-۲۸)$$

$$h^{t+1} = \tanh(W_{hh}(r \odot h^t) + W_{xh} x^t + b_h) \odot z + h^t \odot (1 - z) \quad (۲-۲۹)$$

MUT3:

$$z = \sigma(W_{xz} x^t + W_{hz} \tanh(h^t) + b_z) \quad (۲-۳۰)$$

$$r = \sigma(W_{xr}x^t + W_{hr}h^t + b_r) \quad (۳۱-۲)$$

$$h^{t+1} = \tanh(W_{hh}(r \odot h^t) + W_{xh}x^t + b_h) \odot z + h^t \odot (1-z) \quad (۳۲-۲)$$

نتایج نشان داده است که شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی روی همه وظایف به غیر از مدل‌سازی زبان بهتر از شبکه حافظه کوتاه و بلندمدت کار می‌کند. شبکه MUT1 روی وظیفه مدل‌سازی زبان همانند شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی کار می‌کند و روی بقیه وظایف بهتر از همه کار می‌کند. شبکه حافظه کوتاه و بلندمدت به صورتی معنی‌دار روی مدل‌سازی زبان مجموعه داده PTB بهتر از بقیه شبکه‌ها کار می‌کند. شبکه حافظه کوتاه و بلندمدت با بایاس فراموشی بزرگ تقریباً روی همه وظایف بهتر از شبکه‌های حافظه کوتاه و بلندمدت و شبکه حافظه کوتاه و بلندمدت واحد دروازه‌ای بازگشتی کار می‌کند. شبکه MUT1 بهترین عملکرد را روی مجموعه داده موسیقی دارد، همچنین شبکه‌های حافظه کوتاه و بلندمدت با دروازه ورودی و دروازه خروجی بهترین نتایج را روی مجموعه داده موسیقی دارند.

با توجه به نتایج به نظر می‌آید دروازه فراموشی دارای بیشترین اهمیت باشد. وقتی دروازه فراموشی حذف شود، شبکه حافظه کوتاه و بلندمدت دارای مشکلات عدیده روی مجموعه داده‌ها خواهد شد، اگرچه این موضوع برای مجموعه داده مدل‌سازی زبان بی‌اهمیت به نظر می‌رسد. دومین دروازه مهم دروازه ورودی است. دروازه خروجی دارای کمترین اهمیت روی شبکه حافظه کوتاه و بلندمدت است. هنگام حذف، h_t به سادگی تبدیل به $\tanh h(c_t)$ می‌شود که برای عملکرد شبکه حافظه کوتاه و بلندمدت کافی است. اگرچه نتایج روی مجموعه داده موسیقی این الگو را تصدیق نمی‌کند زیرا شبکه‌های حافظه کوتاه و بلندمدت با دروازه ورودی و دروازه خروجی بهترین عملکرد را در آنجا داشتند. با اهمیت‌تر از این‌ها مشخص شد که اضافه کردن یک بایاس مثبت به دروازه فراموشی به‌خوبی عملکرد شبکه حافظه کوتاه و بلندمدت را بهبود می‌بخشد.

۲-۷- الگوریتم انتشار رو به عقب طی زمان[۲۷]

الگوریتم انتشار رو به عقب طی زمان، نوعی از الگوریتم انتشار رو به عقب است که به شبکه‌های بازگشتی اعمال می‌شود [۴۵]. این الگوریتم مشابه الگوریتم انتشار رو به عقب ساده است اما در شبکه بازگشتی خطاها در یک مرحله زمانی می‌بایست به‌صورت «در طی زمان» به همه مراحل زمانی قبلی، به عقب انتشار پیدا کند. آموزش در این الگوریتم مشابه شبکه پیشخور است غیر از اینکه هر تکرار می بایست طی خروجی مشاهده شده به صورت متوالی اجرا شود. در این قسمت معادلات روبه‌جلو و انتشار رو به عقب در طی زمان در یک‌لایه مخفی شبکه حافظه کوتاه و بلندمدت نمایش داده شده است.

معادلات (۲-۳۳) تا (۲-۵۰) برای یک بلوک واحد شبکه حافظه کوتاه و بلندمدت است، برای چندین بلوک محاسبات به‌صورت تکراری روی هر بلوک انجام می‌شود. در معادلات, w_{ij} وزن متصل از واحد i به واحد j اُم است، a_j^t ورودی شبکه به واحد j در زمان t است و b_j^t فعالساز واحد j در زمان t است. زیرنویس‌های l, ϕ و w به ترتیب نشان‌دهنده دروازه ورودی، دروازه فراموشی و دروازه خروجی در بلوک است. c نشان‌دهنده یکی از C سلول حافظه است. s_j^t بیان‌گر وضعیت سلول c در زمان t است. f تابع فعالساز دروازه‌ها است و g و h به ترتیب توابع فعالساز ورودی بلوک و خروجی بلوک هستند. I تعداد ورودی‌ها، K تعداد خروجی‌ها و H تعداد سلول‌ها در لایه مخفی است. b_c^t خروجی بلوک است و فقط این خروجی بلوک ها به بلوک های دیگر متصل هستند. اندیس h نشان‌دهنده خروجی‌های سلول از بلوک‌های دیگر در لایه مخفی است. G نمایانگر تعداد کلی ورودی‌ها به لایه مخفی، شامل سلول‌ها و دروازه‌ها است. برای یک‌لایه استاندارد شبکه حافظه کوتاه و بلندمدت با یک سلول در هر بلوک $G = 4H$ است.

مثل یک شبکه بازگشتی استاندارد انتشار روبه‌جلو برای یک ورودی با طول T برای رشته x با شروع از زمان $t=1$ محاسبه می‌شود و به‌صورت بازگشتی معادلات به‌صورت افزایشی روی t بروز رسانی می‌شوند. انتشار رو به عقب طی زمان با شروع در زمان $t=T$ محاسبه می‌شود و به‌صورت بازگشتی

[۲۷] Backpropagation through time

مشتقات واحد را به‌صورت کاهشی تا زمان ۱ محاسبه می‌کند. مشتقات وزن نهایی به وسیله جمع روی همه مشتق‌ها در هر مرحله زمانی محاسبه می‌شود.

$$\delta_j^t = \frac{\partial L}{\partial a_j^t} \qquad (۳۳-۲)$$

جایی که L تابع هزینه مورد استفاده برای آموزش است.

۲-۷-۱- انتشار روبه‌جلو

دروازه‌های ورودی

$$a_l^t = \sum_{i=1}^{I} w_{il} x_i^t + \sum_{h=1}^{H} w_{hl} b_h^{t-1} \qquad (۳۴-۲)$$

$$b_l^t = f(a_l^t) \qquad (۳۵-۲)$$

دروازه‌های فراموشی

$$a_\phi^t = \sum_{i=1}^{I} w_{i\phi} x_i^t + \sum_{h=1}^{H} w_{h\phi} b_h^{t-1} \qquad (۳۶-۲)$$

$$b_\phi^t = f(a_\phi^t) \qquad (۳۷-۲)$$

حالت سلول‌ها

$$a_c^t = \sum_{i=1}^{I} w_{ic} x_i^t + \sum_{h=1}^{H} w_{hc} b_h^{t-1} \qquad (۳۸-۲)$$

$$s_c^t = b_\phi^t s_c^{t-1} + b_l^t g(a_c^t) \qquad (۳۹-۲)$$

دروازه‌های خروجی

$$a_w^t = \sum_{i=1}^{I} w_{iw} x_i^t + \sum_{h=1}^{H} w_{hw} b_h^{t-1} \quad (۴۰-۲)$$

$$b_w^t = f(a_w^t) \quad (۴۱-۲)$$

خروجی بلوک‌ها

$$b_c^t = b_w^t h(s_c^t) \quad (۴۲-۲)$$

۲-۷-۲- انتشار رو به عقب

پس از محاسبه انتشار رو به جلو برای شبکه, می بایست خطا در هر لایه محاسبه شود. با داشتن خطای خروجی بلوک که مشتق تابع هزینه نسبت به خروجی بلوک است می توان به ترتیب خطای دروازه خروجی و خطای حالت سلول را بدست اورد و سپس خطای دروازه ورودی, دروازه فراموشی, ورودی بلاک و خطای حالت سلول در مرحله زمانی پیشین را بدست آورد. با داشتن این خطا ها می توان خطای وزن ها را محاسبه کرد و وزن ها طی الگوریتم بهینه سازی بروزرسانی می شوند. این الگوریتم بهینه سازی در بخش بعد توضیح داده خواهد شد.

$$\varepsilon_c^t = \frac{\partial L}{\partial b_c^t} \quad (۴۳-۲)$$

$$\varepsilon_s^t = \frac{\partial L}{\partial s_c^t} \quad (۴۴-۲)$$

خروجی‌های بلوک

$$\varepsilon_c^t = \sum_{k=1}^{K} w_{ck} \delta_k^t + \sum_{g=1}^{G} w_{cg} \delta_g^{t+1} \quad (۴۵-۲)$$

دروازه‌های خروجی

$$\delta_w^t = f'(a_w^t)\sum_{c=1}^{C} h(s_c^t)\varepsilon_c^t \quad (۴۶-۲)$$

حالت سلول‌ها

$$\varepsilon_s^t = b_w^t h'(s_c^t)\varepsilon_c^t + b_\phi^{t+1}\varepsilon_s^{t+1} \quad (۴۷-۲)$$

بلوک‌ها

$$\delta_c^t = b_l^t g'(a_c^t)\varepsilon_s^t \quad (۴۸-۲)$$

دروازه‌های فراموشی

$$\delta_\phi^t = f'(a_\phi^t)\sum_{c=1}^{C} s_c^{t-1}\varepsilon_s^t \quad (۴۹-۲)$$

دروازه‌های ورودی

$$\delta_l^t = f'(a_l^t)\sum_{c=1}^{C} g(a_c^t)\varepsilon_s^t \quad (۵۰-۲)$$

۲-۸- الگوریتم بهینه‌سازی روش نرخ یادگیری انطباقی

الگوریتم روش نرخ یادگیری انطباقی[۲۸] الگوریتم بهینه ساز یادگیری بر اساس گرادیان نزولی[۲۹] است که نرخ یادگیری بر روی هر پارامتر در طول زمان را تعدیل می‌کند [۴۶]. این الگوریتم بهبودی بر الگوریتم گرادیان انطباقی[۳۰] است که به پارامترها[۳۱] حساس‌تر است و ممکن است نرخ یادگیری را به‌تندی

[۲۸] ADADELTA
[۲۹] Gradient descent
[۳۰] Adagrad

کاهش دهد [۴۷]. روش نرخ یادگیری انطباقی می‌تواند برای بهینه‌سازی به‌جای گرادیان نزولی یکپارچه[۳۲] به کار رود. به‌جای اینکه همه مربع گرادیان‌های گذشته بر روی‌هم انباشته و باهم جمع شوند، این الگوریتم گرادیان‌های انباشته گذشته را به پنجره‌ای به اندازه ثابت w محدود می‌کند. به‌جای ذخیره بی‌فایده تعداد w مربع گرادیان‌های پیشین، مجموع گرادیان‌ها به‌عنوان یک میانگین کاهشی از همه مربع گرادیان‌های پیشین به‌صورت بازگشتی تعریف می‌شود. میانگین $E[g^2]_t$ در مرحله زمانی t در نظر گرفته می‌شود (γ مشابه‌ترم ممنتوم و کوچک‌تر از ۱ است) و سپس بر اساس تنها میانگین پیشین و حال حاضر گرادیان محاسبه می‌شود که در رابطه (۲-۵۱) نمایش داده شده است. g_t گرادیان تابع موردنظر است ($g^2{}_t$ گشتاور دوم گرادیان یا مشتق جزئی تابع هزینه J نسبت به پارامتر θ است).

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma) g_t^2 \qquad (۵۱-۲)$$

گرادیان نزولی به‌صورت دسته[۳۳] برای هر مثال آموزشی $x^{(i)}$ و برچسب $y^{(i)}$ در اینجا تعریف می‌شود که در آن θ پارامترهای به‌روزرسانی مدل، η نرخ یادگیری و $\nabla\theta$ مشتق جزئی تابع هزینه J نسبت به پارامتر θ است. دسته[۳۴] یک به‌روزرسانی برای هر دسته کوتاه از n مثال آموزشی را انجام می‌دهد که در رابطه (۲-۵۲) نشان داده شده است.

$$\theta = \theta - \eta . \nabla_\theta J(\theta; x^{(i:i+n)}; y^{(i:i+n)}) \qquad (۵۲-۲)$$

با این روش واریانس پارامترهای به‌روزرسانی مدل کاهش می‌یابد و همچنین از ماتریس‌های بهینه‌سازی به‌خوبی بهینه‌شده، استفاده می‌کند.

حال به‌روزرسانی SGD بر اساس بردار پارامتر به‌روزرسانی $\Delta\theta_t$ توسط روابط (۲-۵۳) و (۲-۵۴) محاسبه می‌شود.

$$\Delta\theta_t = -\eta . g_{t,i} \qquad (۵۳-۲)$$

---

[۳۱] Hyperparameters
[۳۲] Stochastic gradient descent
[۳۳] Mini-batch gradient descent
[۳۴] Mini-batch

$$\theta_{t+1} = \theta_t + \Delta\theta_t \qquad (۵۴-۲)$$

الگوریتم گرادیان انطباقی[۳۵] به‌صورت معادله (۲-۵۵) تعریف می‌شود که در آن G_t ماتریس مربعی قطری است که هر عنصر روی قطر i, i مجموع مربعات گرادیان‌ها است. اپسیلون ترم هموارسازی است که از تقسیم کسر بر صفر جلوگیری می‌کند و اغلب عددی کوچک مثل $۱۰^{-۶}$ است و $\odot$ ضرب نقطه‌ای است.

$$\Delta\theta_t = -\frac{\eta}{\sqrt{G_t + \varepsilon}} \odot g_t \qquad (۵۵-۲)$$

حال در الگوریتم روش نرخ یادگیری انطباقی، ماتریس قطری G_t با میانگین کاهشی روی مربع گرادیان‌های پیشین جایگزین می‌شود و توسط رابطه (۲-۵۶) نمایش داده شده است.

$$\Delta\theta_t = -\frac{\eta}{\sqrt{E[g^2]_t + \varepsilon}} g_t \qquad (۵۶-۲)$$

این همان خطای میانگین مربعات ریشه[۳۶] است که می‌تواند به‌صورت معادله (۲-۵۷) نوشته شود.

$$\Delta\theta_t = -\frac{\eta}{RMS[g]_t} \qquad (۵۷-۲)$$

می‌توان میانگین کاهشی را به‌گونه‌ای دیگر نوشت که این بار مربع گرادیان‌ها نیست بلکه مربع پارامترهای بروز رسانی است که در رابطه (۲-۵۸) مشخص شده است.

$$E[\Delta\theta^2]_t = \gamma E[\Delta\theta^2]_{t-1} + (1-\gamma)\Delta\theta_t^2 \qquad (۵۸-۲)$$

خطای میانگین مربعات ریشه پارامترهای به‌روزرسانی به‌صورت معادله (۲-۵۹) تغییر می‌کند.

$$RMS[\Delta\theta]_t = \sqrt{E[\Delta\theta^2]_t + \varepsilon} \qquad (۵۹-۲)$$

ازآنجایی‌که $RMS[\Delta\theta]_t$ نامشخص است، مقدارش توسط RMS پارامتر به‌روزرسانی تا مرحله زمانی قبلی تخمین زده می‌شود. با جایگزین کردن نرخ یادگیری η در قانون به‌روزرسانی پیشین با

---

[۳۵] Adagrad
[۳۶] RMS

$RMS[\Delta\theta]_{t-1}$ (که هنوز نامشخص است), درنهایت قانون به‌روزرسانی الگوریتم گرادیان انطباقی به‌صورت معادله (۲-۶۰) و (۲-۶۱) تعیین می‌شود.

$$\Delta\theta_t = -\frac{RMS[\Delta\theta]_{t-1}}{RMS[g]_t} g_t \qquad (۶۰-۲)$$

$$\theta_{t+1} = \theta_t + \Delta\theta_t \qquad (۶۱-۲)$$

۲-۹- نتیجه‌گیری

در این فصل ابتدا به کلیات و جزئیات شبکه حافظه کوتاه و بلندمدت و مراحل کار این شبکه پرداخته شد و بلوک های شبکه حافظه کوتاه و بلندمدت و طرز کار آن‌ها شرح داده شد. سپس انواع شبکه‌های مشتق شده از شبکه حافظه کوتاه و بلندمدت و معماری‌های گوناگون آن معرفی شد. ملاحظه گردید که در سال های اخیر این شبکه دارای تغییرات بسیار گسترده‌ای بوده است و تحقیقات روی این شبکه همچنان ادامه دارد. درنهایت الگوریتم انتشار رو به عقب طی زمان و الگوریتم بهینه‌سازی روش نرخ یادگیری انطباقی برای این شبکه معرفی شدند.

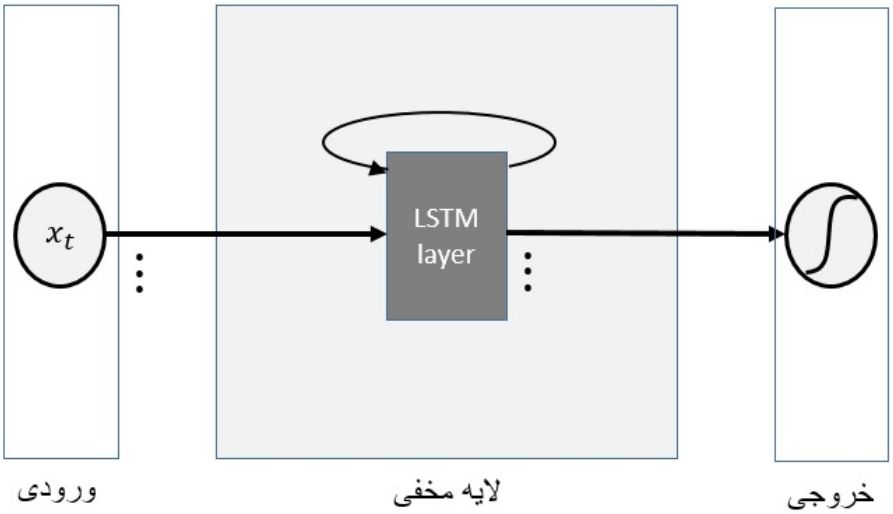
۳- فصل سوم: الگوریتم و توابع فعالساز پیشنهادی

۳–۱– مقدمه

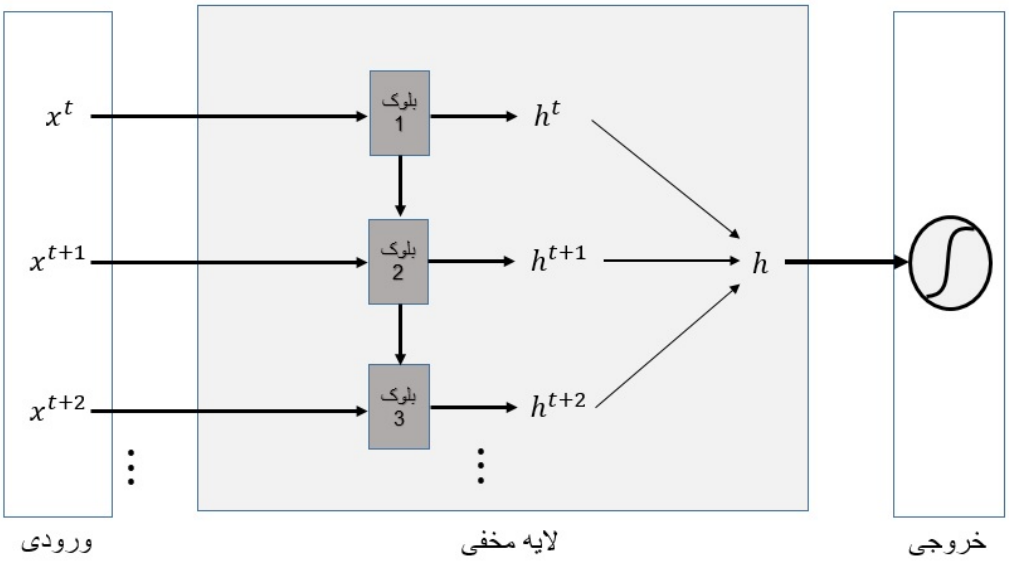
در این بخش ابتدا الگوریتم برای شبکه حافظه کوتاه و بلندمدت مورد استفاده در این پایان‌نامه شامل شرح معماری و بلوک‌های مورد استفاده معرفی می‌شود. سپس ۲۳ تابع فعالساز مورد استفاده برای مقایسه عملکرد آن‌ها در دروازه‌های سیگموئیدی شبکه حافظه کوتاه و بلندمدت، به همراه مشتق و بازه آن‌ها معرفی می‌گردند. درنهایت مجموعه داده‌های مورد استفاده برای ارزیابی عملکرد شبکه روی این توابع فعالساز معرفی می‌شوند.

۳–۲– الگوریتم مورد استفاده

در این پایان‌نامه از مدل استاندارد شبکه حافظه کوتاه و بلندمدت با دروازه فراموشی و بدون استفاده از اتصالات روزنه‌ای برای کلاسه‌بندی استفاده شده است. نشان داده شده است که اتصالات روزنه‌ای فقط پیچیدگی محاسباتی را بالا می‌برد و در بسیاری از موارد کاربرد مناسبی ندارد [۱۸]. معماری شبکه مورد استفاده دارای ۳ لایه، لایه ورودی، لایه خروجی و یک‌لایه مخفی است. لایه ورودی شامل ورودی‌های شبکه، لایه مخفی شامل بلوک‌ها است که به‌صورت بازگشتی به هم متصل هستند و لایه خروجی شامل رگرسیون لجستیک با تابع فعالساز softmax برای کلاسه‌بندی است. معماری شبکه مورد استفاده در شکل‌های (۳-۱) و (۳-۲) نمایش داده شده است. لایه مخفی شامل چندین بلوک با یک سلول حافظه است که همگی به‌صورت بازگشتی به همدیگر متصل هستند.



شکل (۳-۱): نمای شماتیک معماری شبکه حافظه کوتاه و بلندمدت شامل ۳ لایه، لایه ورودی، یک‌لایه مخفی و یک لایه خروجی. لایه مخفی شامل چندین بلوک است که به‌صورت بازگشتی به یکدیگر متصل هستند.



شکل (۳-۲): معماری شبکه حافظه کوتاه و بلندمدت. تمام ورودی‌ها به تمام بلوک‌ها در لایه مخفی متصل هستند، از خروجی تمام بلوک‌ها در طی زمان میانگین‌گیری صورت می‌پذیرد و به لایه خروجی برای رده‌بندی اعمال می‌گردد.

در لایه مخفی بلوک‌ها و هر سلول حافظه در بلوک به‌صورت بازگشتی به یکدیگر متصل هستند. شکل (۳-۳) نشان‌دهنده یک بلوک واحد در شبکه حافظه کوتاه و بلندمدت است. هر بلوک معادلاتی به شکل معادلات (۳-۱) تا (۳-۶) دارد.

$$f^t = \sigma\left(W_f . x^t + U_f . h^{t-1} + b_f\right) \quad (۱-۳)$$

$$i^t = \sigma\left(W_i . x^t + U_i . h^{t-1} + b_i\right) \quad (۲-۳)$$

$$o^t = \sigma\left(W_o . x^t + U_o . h^{t-1} + b_o\right) \quad (۳-۳)$$

$$\tilde{C}^t = \tanh\left(W_{\tilde{C}} . x^t + U_{\tilde{C}} . h^{t-1} + b_{\tilde{C}}\right) \quad (۴-۳)$$

$$C^t = f^t \odot C^{t-1} + i^t \odot \tilde{C}^t \quad (۵-۳)$$

$$h^t = o^t \odot \tanh\left(C^t\right) \quad (۶-۳)$$

معادلات (۳-۱) تا (۳-۳) به ترتیب نشان‌دهنده دروازه فراموشی، دروازه ورودی و دروازه خروجی است. دروازه ورودی تصمیم می‌گیرد چه مقادیری می‌بایست به‌روزرسانی شوند، دروازه فراموشی تصمیم می‌گیرد چه مقادیری می‌بایست حذف شوند و دروازه خروجی به همراه بلوک خروجی تصمیم می‌گیرد که چه اطلاعاتی در زمان به خارج بلوک فرستاده شوند. $\tilde{C}^t$ در معادله (۳-۴) ورودی بلوک در زمان است که یک لایه تانژانت هایپربولیک است که با دروازه ورودی تصمیم می‌گیرند چه اطلاعات جدیدی می‌بایست در حالت سلول حافظه ذخیره شوند. C^t در معادله (۳-۵) حالت سلول حافظه در زمان است که به وسیله وضعیت قبلی آن به‌روزرسانی می‌شود. h^t در معادله (۳-۶) خروجی بلوک در زمان است. x^t ورودی در زمان است. W و U ماتریس‌های وزن هستند و b بردار بایاس است. نماد $\odot$ ضرب نقطه‌ای دو بردار است. توابع σ و tanh توابع فعالساز سیگموئید و تانژانت هایپربولیک هستند. در این پایان‌نامه توابع فعاساز متفاوتی به‌جای توابع فعالساز دروازه‌های سیگموئیدی بررسی شده‌اند و توابع فعالساز tanh بدون تغییر باقی مانده‌اند. برای آموزش شبکه از الگوریتم انتشار رو به عقب به همراه بهینه‌ساز الگوریتم روش نرخ یادگیری انطباقی استفاده شده است. برای لایه خروجی از رگرسیون لجستیک به‌صورت معادله (۳-۷) استفاده شده است.

$$P(Y = i \mid h, W, b) = \text{softmax}_i(Wh + b) = \frac{e^{W_i h + b_i}}{\sum_j e^{W_j h + b_j}} \quad (۷-۳)$$

$$y_{pred} = \text{argmax}_i P(Y = i \mid h, W, b) \quad (۸-۳)$$

تابع هزینه مورد استفاده، تابع درست نمایی لگاریتمی منفی[۳۷] است و در رابطه (۳-۹) و (۳-۱۰) نمایش داده شده است.

$$L(\theta = \{W, b\}, D) = \sum_{i=0}^{|D|} \log(P(Y = y^{(i)} \mid h^{(i)}, W, b)) \quad (۹-۳)$$

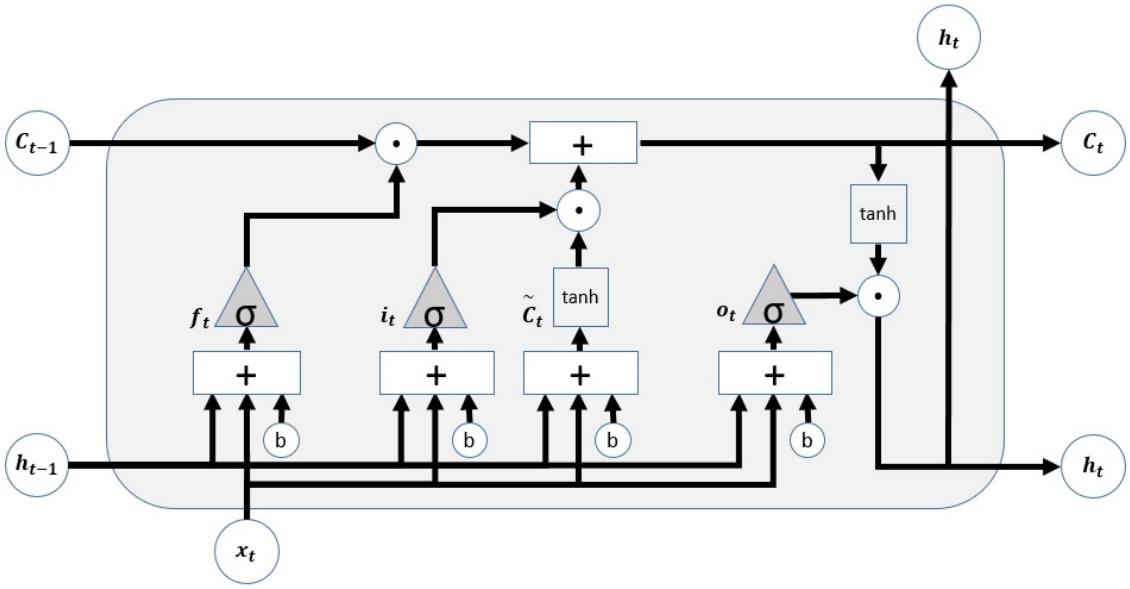
$$\ell(\theta = \{W, b\}, D) = -L(\theta = \{W, b\}, D) \quad (۱۰-۳)$$

شکل (۳-۳) نشان‌دهنده یک بلوک واحد است. سه دروازه فراموشی، ورودی و خروجی که در این پایان‌نامه دروازه‌های سیگموئیدی نامیده می‌شوند، توسط مثلث در آن نمایش داده شده‌اند و همچنین توابع فعالساز ورودی بلوک و خروجی بلوک توسط مربع نمایش داده شده‌اند. خروجی بلوک به‌صورت بازگشتی به ورودی بلوک و هر سه دروازه خروجی در بلوک بعدی متصل هستند. هر بلوک چهار ورودی کلی دارد، ابتدا بردار x^t که ورودی در زمان است که به دروازه‌ها و ورودی بلوک اعمال می‌شود، دوم خروجی بلوک در زمان ماقبل h^{t-1} که به دروازه‌ها و ورودی بلوک اعمال می‌شود و همچنین بردار بایاس که به همین صورت اعمال می‌شود، چهارم وضعیت سلول حافظه در زمان ماقبل C^{t-1} که به شکل نواری جمع کننده و ضرب کننده در طول بلوک حرکت می‌کند و وضعیت بلوک را بروز می‌کند. هر بلوک دو خروجی دارد، یکی حالت سلول در زمان و دیگری خروجی بلوک در زمان، خروجی بلوک و حالت سلول در هر بلوک به بلوک بعدی اعمال می‌شوند. همچنین از خروجی هر بلوک در زمان یک خروجی دیگر گرفته شده و بعد از میانگین‌گیری به لایه خروجی فرستاده می‌شود. یک میانگین‌گیری روی همه خروجی‌های بلوک گرفته شده و در لایه خروجی کلاسه‌بندی توسط رگرسیون لجستیک با تابع *softmax* اعمال می‌شود.

شبکه مورد استفاده به وسیله زبان پایتون و کتابخانه Theano[۳۸] نوشته شده است. کتابخانه Theano برای تعریف، بهینه‌سازی و ارزیابی عبارت‌های ریاضی و مخصوصا شبکه‌های عصبی عملکرد

[۳۷] Negative log-likelihood

مناسبی دارد. از ویژگی‌های این کتابخانه می توان به هماهنگی مناسب با کتابخانه‌های دیگر پایتون، پایداری و سرعت بالا اشاره کرد.



شکل (۳-۳): یک بلوک واحد شبکه حافظه کوتاه و بلندمدت. دروازه‌های سیگموئیدی با مثلث و توابع فعالساز ورودی و خروجی بلوک با مربع نمایش داده شده‌اند.

روند اجرا برای جملات ورودی که به صورت متن است به‌صورت زیر است.

هر کلمه واحد در ورودی کلی (کل جملات مجموعه داده) با یک عدد خاص اندیس گذاری می‌شود. به دلیل اینکه در یک تکرار[۳۹] تعداد کل ورودی ها و در نتیجه ماتریس ورودی خیلی بزرگ نشود از روش دسته[۴۰] استفاده می شود. تعداد کل ورودی‌ها (تعداد بردار برای هر جمله) تقسیم بر اندازه دسته می‌شود و عدد به‌دست‌آمده نشانگر تعداد به‌روزرسانی در هر تکرار است. اگر n نشان‌دهنده تعداد کل ورودی و b

[۳۸] https://pypi.python.org/pypi/Theano
[۳۹] Epoch
[۴۰] Mini-batch

نمایان گر تعداد دسته باشد, حداکثر $(\frac{n}{b})+1$ به‌روزرسانی نیاز است تا همه ورودی‌ها در دسته برای هر تکرار دیده شوند. به طور مثال اگر تعداد کل ورودی مجموعه داده ۸۱۶۲ جمله و اندازه دسته ۱۶ جمله باشد، ۵۱۱ به‌روزرسانی در هر تکرار وجود خواهد داشت ($(\frac{8162}{16}+1)=511$) که در هر کدام از این ۵۱۱ به‌روزرسانی، اندازه ورودی ۱۶ خواهد بود. اندازه دسته نشان‌دهنده تعداد (بردار) ورودی در هر به‌روزرسانی است. به عبارت دیگر به جای ایجاد یک ورودی کلی بسیار بزرگ که ماتریسی بزرگ (با پیچیدگی محاسباتی بالا) را در پی خواهد داشت، این ورودی به دسته های کوچک تر تقسیم بندی می شود. در هر به‌روزرسانی، به اندازه اندازه دسته ورودی به شبکه اعمال می‌شود. هر بلوک یک مرحله زمانی محسوب می‌شود.

مرحله اول: در به‌روزرسانی اول b جمله اول از ورودی کلی به‌صورت موازی به بلوک‌ها اعمال می‌شود. بدین صورت که بردار جمله اول دسته با اندازه b به تمام بلوک‌ها به‌عنوان ورودی اعمال می‌شود، سپس جمله دوم دسته به تمام بلوک‌ها به‌عنوان ورودی اعمال می‌شود و تا جمله b اُم در دسته به همین صورت اعمال می‌شود. از هر بلوک در لایه مخفی یک خروجی h وجود دارد که روی همه این خروجی‌ها میانگین‌گیری در طول زمان صورت پذیرفته و درنهایت در هر به‌روزرسانی یک h خروجی وجود دارد و محاسبات روی این b جمله در دسته به‌صورت موازی صورت می‌پذیرد، انتشار رو به عقب طی زمان به همراه بهینه‌ساز الگوریتم روش نرخ یادگیری انطباقی انجام می‌شود و برای به‌روزرسانی اول خطا محاسبه شده و وزن‌ها به‌روزرسانی می‌شوند.

مرحله سوم: به همین صورت در به‌روزرسانی دوم b جمله بعدی از ورودی کلی به‌صورت تک‌به‌تک اما موازی به شبکه اعمال شده و وزن‌ها به‌روزرسانی می‌شوند تا درنهایت همه ورودی‌ها دیده شوند. همانند شبکه‌های بازگشتی دیگر، شبکه حافظه کوتاه و بلندمدت مقدار ورودی را محاسبه می‌کند، اما مقدار خروجی این ورودی بعداً در دسترس برای استفاده خواهد بود که همان مفهوم بازگشتی است. در همین حال شبکه با دیدن ورودی جدید، این ورودی جدید بعلاوه پیش‌بینی قبلی را برای ایجاد خروجی به کار می‌گیرد.

الگوریتم کلی شبکه به صورت زیر است.

۱. وزن ها به صورت تصادفی مقدار دهی اولیه می شوند.
۲. با معادلات (۳-۱) تا (۳-۶) انتشار به صورت رو به جلو محاسبه می شود.
۳. مقدار تابع هزینه محاسبه می شود.
۴. انتشار رو به عقب برای محاسبه گرادیان محاسبه می شود.
۵. الگوریتم بهینه ساز روش نرخ یادگیری انطباقی، وزن ها را به صورتی که تابع هزینه کمینه شود، بروزرسانی می کند.
۶. تکرار مراحل ۲ تا ۵ روی هر دسته برای تمام به‌روزرسانی ها، تا زمانی که شبکه همگرا شود.

۳-۳- توابع فعالساز

سه جنبه اساسی شبکه‌های عصبی نقشی مهم در عملکرد شبکه دارند، این سه جنبه ابتدا معماری و الگوی ارتباط بین واحدها، دوم الگوریتم یادگیری و سوم توابع فعالساز مورد استفاده در شبکه هستند. بیشتر تحقیقات شبکه‌های عصبی بر روی اهمیت الگوریتم یادگیری تأکید دارند و اهمیت توابع فعالساز نادیده گرفته شده است [۳۷, ۳۸, ۳۹].

انتخاب‌های معمول برای تابع فعالساز شامل سیگموئید و تابع تانژانت هایپربولیک[۴۱] است. تابع دوم در شبکه‌های عصبی پیشخور رایج هستند و به شبکه‌های بازگشتی نیز اعمال شده‌اند [۴۴]. آنالیزی روی توابع فعالساز متفاوت در شبکه‌های عصبی و پرسپترون چند لایه[۴۲] صورت پذیرفته است که نشان می‌دهد تابع سیگموئید و تانژانت هایپربولیک عملکرد بهتری نسبت به بقیه توابع دارند [۴۸, ۴۹]. نشان داده شده است که توابع فعالساز یکی از کلیدی‌ترین اجزای شبکه‌های شبکه حافظه کوتاه و بلندمدت می‌باشند [۱۹] و همچنین نورون‌های یکسو کننده[۴۳] حتی مدل‌های بهتری از نظر زیست شناسی هستند و

[۴۱] Tanh
[۴۲] Multilayer perceptron
[۴۳] Rectifier

عملکردی مساوی یا بهتر از شبکه‌های تانژانت هایپربولیک دارند [۵۰]. دیگر توابع فعالساز مورد استفاده در شبکه‌های عصبی شامل log-log, log-log تکمیلی و probit هستند که نتایج مناسبی روی پیش‌بینی سری زمانی داشته‌اند [۵۱], همچنین توابع فعالساز متناوب [۵۲], توابع فعالساز چند جمله‌ای هرمیت[۴۴] [۵۳], توابع غیر چند جمله‌ای [۵۴, ۵۵, ۵۶], ترکیب انواع توابع مثل چند جمله‌ای، متناوب، سیگموئیدی و گاوسی [۵۷], تابع فعالساز گاوسی [۵۸] توابع فعالساز معروف دیگر هستند که در وظایف گوناگون روی مجموعه داده‌ها و شبکه‌های مختلف عصبی نتایج مناسبی ارائه کرده‌اند. همچنین توابع فعالساز sinc و sincos [۵۹], کلاس جدیدی از توابع سیگموئیدی [۳۹], یک تابع سیگموئیدی جدید که نتایج مناسبی برای مدلسازی سیستم‌های گسسته زمانی و پویا دارد [۶۰] و تابع انتقالی Lorentzian [۶۱] از این دست توابع فعالساز هستند.

خواصی که می‌بایست عموماً توسط یک تابع فعالساز برآورده شوند عبارتند از:

۱. تابع فعالساز می‌بایست پیوسته باشد [۶۲, ۶۳, ۶۴].
۲. تابع فعالساز می‌بایست کران دار باشد [۶۲, ۶۳, ۶۴, ۶۵, ۶۶].
۳. یکنواختی تابع فعالساز لازمه اجباری برای وجود خاصیت تقریب فراگیر[۴۵] نیست [۶۳, ۶۶], خاصیت تقریب فراگیر نشان می‌دهد که یک شبکه پرسپترون چندلایه استاندارد با یک لایه مخفی شامل تعداد متناهی گره در لایه مخفی با فرض استفاده از تابع فعالساز، یک تخمین زننده جامع است.
۴. تابع فعالساز می‌بایست سیگموئیدی باشد [۶۲, ۶۳, ۶۴, ۶۵, ۶۶] و یا در بازه بی‌نهایت معادلات (۳-۱۱) تا (۳-۱۳) را برآورده کند [۶۷]:

معادله ۳-۱۱ $$\lim_{x \to -\infty} f(x) = \alpha$$

معادله ۳-۱۲ $$\lim_{x \to +\infty} f(x) = \beta$$

معادله ۳-۱۳ $$with \ \alpha < \beta$$

در این پایان‌نامه با تغییر یکنواخت توابع فعالساز دروازه‌ها (دروازه ورودی، دروازه خروجی و دروازه فراموشی) که به نام دروازه‌های سیگموئیدی (معادلات (۳-۱) تا (۳-۳)) نام‌گذاری می‌شوند، به بررسی شبکه حافظه کوتاه و بلندمدت پرداخته می‌شود. ۲۳ تابع فعالساز متفاوت در دروازه‌های سیگموئیدیِ

[۴۴] Hermite
[۴۵] Universal approximation property

شبکه حافظه کوتاه و بلندمدت برای پیدا کردن بهترین تابع فعالساز مقایسه می‌شود. در ادامه این توابع فعالساز معرفی می شود. توابع فعالساز مورد استفاده, مشتق و بازه آن ها در جدول (۳-۱) قابل مشاهده است. همچنین شکل این توابع در شکل (۳-۴) قابل مشاهده است.

- تابع فعالساز Aranda: تابع فعالساز *Aranda* [۳۷] تابعی نامتقارن است. تابع اصلی Aranda-Ordaz به‌صورت معادله (۳-۱۴) است [۶۸]. جایی که λ پارامتر آزاد شیب قابل تغییر است. پارامتر λ بزرگ‌تر از صفر است. معکوس این رابطه به فرم معادله (۳-۱۵) است. هنگامی که $\lambda = 1$ باشد این تابع همان تابع سیگوئیدی است. تابع فعالساز *Aranda* نتایج مناسبی برای پیش‌بینی سری‌های زمانی با شبکه پرسپترون چند لایه داشته است و عملکرد مناسب آن در مقایسه با توابع *logit* و *cloglog* نشان داده شده است [۳۷].

$$x = \log\left[\frac{(1-\pi)^{-\lambda}}{\lambda}\right] \qquad (۱۴-۳)$$

$$\pi = 1-(1+\lambda e^{x})^{-1/\lambda} \qquad (۱۵-۳)$$

- تابع فعالساز Bi modal: شامل ۴ تابع فعالساز معرفی‌شده دونمایی است که کلاسی از مجموع دو تابع سیگموئیدی و هایپربولیک تانژانت می‌باشند و نشان داده شده است که در شبکه‌های عصبی با الگوریتم انتشار رو به عقب به همراه بهینه‌ساز RPROP و تعداد تکرار الگوریتم برابر، دارای نتایج بهتری نسبت به تابع سیگموئیدی است [۶۹]. این ۴ تابع به ترتیب با اسامی *Bi-sig1*, *Bi-sig2*, *Bi-tanh1* و *Bi-tanh2* مشخص شده اند. برای یادگیری در شبکه حافظه کوتاه و بلندمدت، به توابع *Bi-tanh1* و *Bi-tanh2* مقدار ۰,۵ اضافه شده است زیرا به‌صورت نرمال با این توابع (به دلیل بودن در بازه [۱،۱-] و هم بازه با توابع فعالساز تانژانت هایپربولیک ورودی و خروجی بلوک در شبکه) یادگیری شبکه دچار اختلال می‌شود.
- تابع فعالساز cloglog: تابع فعالساز *cloglog* تابعی سیگموئیدی است که مکملی برای تابع *loglog* است [۵۱]. نتایج آن بر روی شبیه ساز مونته کارلو برای شبکه عصبی پرسپترون

چندلایه با نشان داده است که این تابع، میانگین نتایجی بهتر نسبت به توابع *logit* و تانژانت هایپربولیک دارد. بازه این تابع بین ۰ و ۱ است.

- تابع فعالساز cloglogm: تابع فعالساز *cloglogm*، اصلاح شده تابع *cloglog* است که بر روی الگوریتم CGF نتایج مناسبی روی پیش‌بینی سری زمانی داشته است [۴۰]. همچنین بازه این تابع با افزودن ۰٫۵ بین [۱٫۵, ۰٫۵-] است.
- تابع فعالساز loglog: تابع فعالساز *loglog* شبیه به تابع سیگموئید است که نتایج مناسبی روی پیش‌بینی سری زمانی داشته است [۴۱]. همچنین بازه این تابع با افزودن ۰٫۵ بین [۱٫۵, ۰٫۵] است.
- تابع فعالساز Elliott و softsign: دو تابع معرفی‌شده توسط الیوت و همکارانش *Elliott*[۷۰] و *softsign* می‌باشند. به تابع *softsign* مقدار ۰٫۵ افزوده شده است. بازه تابع *Elliott* بین [۰،۱] و بازه تابع *softsign* بین [۱٫۵, ۰٫۵-] است.
- تابع فعالساز Gaussian: این تابع یکی از توابع فعالساز معروف است که نتایج مناسبی در شبکه‌های مختلف و به‌خصوص در شبکه‌های RBF نتایج مطلوبی دارد [۷۱]. بازه تابع گاوسی بین [۰،۱] است.
- تابع فعالساز logarithmic: تابع لگاریتمی تابعی معروف است که در روش‌های بسیاری از جمله جستجوی فضای بازی [۷۲] مورد استفاده قرار گرفته است. مقدار ۰٫۵ به این تابع اضافه شده است و بازه آن بی‌نهایت است.
- تابع فعالساز logsigm: شامل تابع *logsigm* که دارای توان ۲ و *sigmoidalm* با توان ۴ [۳۹] است که کلاسی از تابع معروف سیگموئید هستند. این توابع عملکرد مناسبی در وظایف گوناگون اعم از پیش‌بینی سری زمانی [۴۱] و شبکه‌های پیشخور [۴۰] داشته‌اند. به این توابع مقدار ۰٫۵ افزوده شده است. بازه این توابع بین [۱٫۵, ۰٫۵] است.
- تابع فعالساز log-sigmoid: تابع معروف سیگموئیدی که دارای بیشترین کاربرد در شبکه‌های عصبی است. این تابع در انواع شبکه‌ها از جمله پرسپترون چندلایه، شبکه‌های بازگشتی و شبکه‌های دیگر به‌عنوان تابع فعالساز رایج مورد استفاده قرار می‌گیرد و همچنین

سرعت همگرایی مناسبی روی الگوریتم انتشار رو به عقب دارد [۷۳]. بازه تابع سیگموئید بین [۰،۱] است.

- تابع فعالساز modified Elliott: تابع بازسازی شده الیوت، تغییریافته تابع الیوت است که در روشی برای جلوگیری از بیش برآزش داده‌های نویزی نتایج بهتری نسبت تابع الیوت ساده داشته است [۷۴]. بازه این تابع بین [۱٫۵, ۰٫۵-] است.
- تابع فعالساز rootsig: این تابع تابعی سیگموئیدی با ریشه است [۳۸]. این تابع عملکرد مناسبی برای پیش‌بینی خشک‌سالی در شبکه پرسپترون چندلایه داشته است [۷۵]. به این تابع مقدار ۰٫۵ افزوده شده است. بازه این تابع بین [۱٫۵, ۰٫۵-] است.
- تابع فعالساز saturated: تابع *saturated* تابعی ساده و خطی است که به‌صورت رابطه (۳-۴۷) است. این تابع نتایج مناسبی بر روی شبکه‌های عصبی رقابتی درجه دوم[۴۶] داشته است [۷۶]. به این تابع مقدار ۰٫۵ افزوده شده است. بازه تابع فوق بین [۱٫۵, ۰٫۵-] است.
- تابع فعالساز sech: تابع *sech* تابع متقاطع هایپربولیکی است که عملکرد مطلوبی برای الگوریتم انتشار رو به عقب دارد [۷۷]. بازه این تابع [۰،۱] است.
- تابع فعالساز sigmoidalm2: کلاسی از تابع سیگموئید است که نتایج مناسبی بر روی شبکه‌های پیشخور داشته است [۷۸]. به این تابع مقدار ۰٫۵ اضافه شده است. بازه این تابع [۱٫۵, ۰٫۵] است.
- تابع فعالساز sigt: تابع سیگموئیدی قابل تنظیمی است که برای کالیبره کردن دوربین در شبکه TAF نتایج مطلوبی را داشته است [۷۹]. بازه این تابع بین [۰،۱] است.
- تابع فعالساز skewed-sig: تابع سیگموئید با مشتق مورب است، این درحالی است که اغلب توابع فعالساز دارای مشتق با شکل متقارن نسبت به محور عمودی مثل تابع سیگموئید و تانژانت هایپربولیک هستند. نتایج روی این تابع فعالساز نشان داده است که اگر در لایه مخفی به کار رود، شبکه به خطای حداقل محلی عمیق‌تر همگرا می‌شود و همچنین شبکه

[۴۶] SOCNN

نسبت به توابع دیگر مثل سیگموئید در لایه مخفی دارای سرعت همگرایی بالاتری خواهد بود [۸۰]. به این تابع مقدار ۰٫۵ افزوده شده است. بازه این تابع بین [۰٫۵, ۱٫۵] است.

- تابع فعالساز wave: تابع موج تابع فعالساز معروفی است که در سال ۱۹۹۴ معرفی‌شده است و نتایج خوبی در شبکه‌های عصبی پیشخور داشته است [۸۱]. بازه تابع موج بین [۱, ۰٫۰۵۵-] است.

جدول (۳-۱): ردیف، نام، معادله تابع فعالساز، مشتق آن و بازه هر تابع فعالساز. به توابعی که در کنار آن‌ها ستاره (*) وجود دارد مقدار ۰/۵ (به تابع اصلی) افزوده شده است.

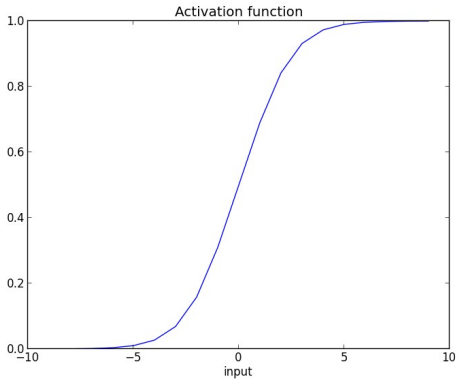
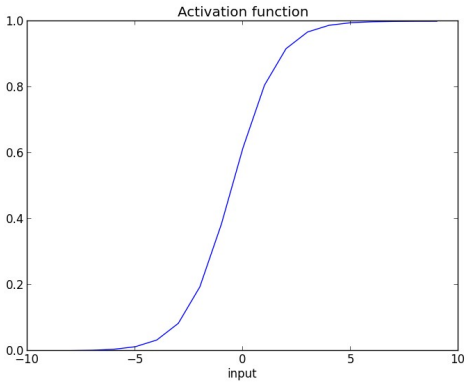
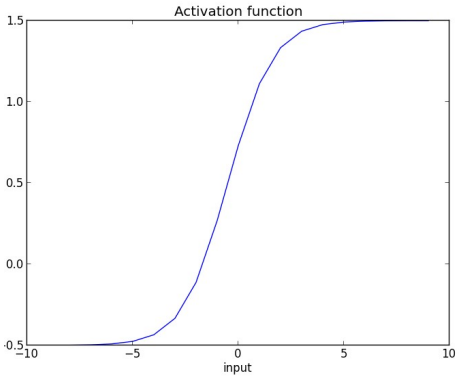
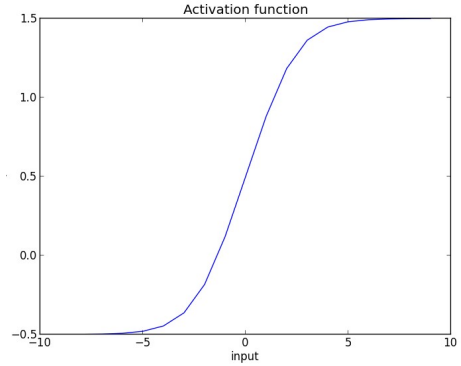
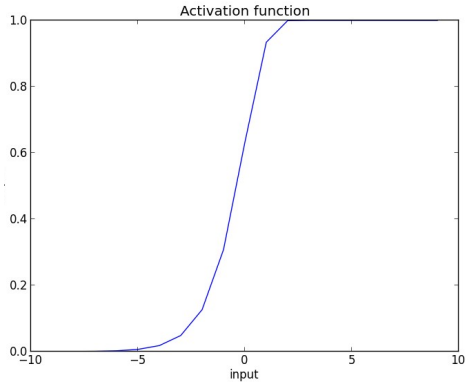
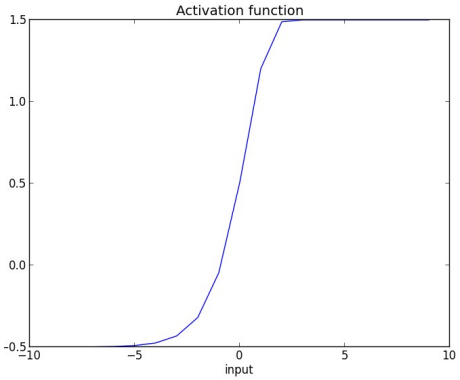
ردیف	نام	تابع فعالساز	مشتق تابع	بازه
۱	Aranda[۲۶]	$f(x)=1-(1+2e^{x})^{-1/2}$	$f'(x)=e^{x}(2e^{x}+1)^{-3/2}$	[۰,۱]
۲	Bi-sig1[۵۶]	$f(x)=\frac{1}{2}\left(\frac{1}{1+e^{-x+1}}+\frac{1}{1+e^{-x-1}}\right)$	$f'(x)=\frac{\frac{e^{1-x}}{(e^{1-x}+1)^2}+\frac{e^{-x-1}}{(e^{-x-1}+1)^2}}{2}$	[۰,۱]
۳	Bi-sig2[۵۶]	$f(x)=\frac{1}{2}\left(\frac{1}{1+e^{-x}}+\frac{1}{1+e^{-x-1}}\right)$	$f'(x)=\frac{\frac{e^{-x}}{(e^{-x}+1)^2}+\frac{e^{-x-1}}{(e^{-x-1}+1)^2}}{2}$	[۰,۱]
۴	Bi-tanh1[۵۶]*	$f(x)=\frac{1}{2}\left[\tanh\left(\frac{x}{2}\right)+\tanh\left(\frac{x+1}{2}\right)\right]+0.5$	$f'(x)=\frac{\operatorname{sech}^2(\frac{x+1}{2})+\operatorname{sech}^2(\frac{x}{2})}{4}$	[-۰,۵ ,۱,۵]
۵	Bi-tanh2[۵۶]*	$f(x)=\frac{1}{2}\left[\tanh\left(\frac{x-1}{2}\right)+\tanh\left(\frac{x+1}{2}\right)\right]+0.5$	$f'(x)=\frac{\operatorname{sech}^2(\frac{x+1}{2})+\operatorname{sech}^2(\frac{x-1}{2})}{4}$	[-۰,۵ ,۱,۵]
۶	cloglog [۳۲]	$f(x)=1-e^{-e^{x}}$	$f'(x)=e^{x-e^{x}}$	[۰,۱]
۷	cloglogm[۳۰]*	$f(x)=1-2e^{-0.7e^{x}}+0.5$	$f'(x)=7e^{x-0.7e^{x}}/5$	[-۰,۵ ,۱,۵]

جدول (۳-۱): (ادامه)

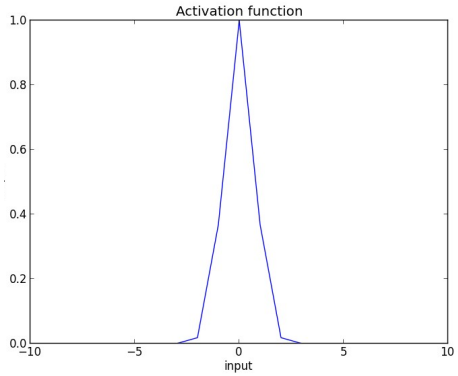
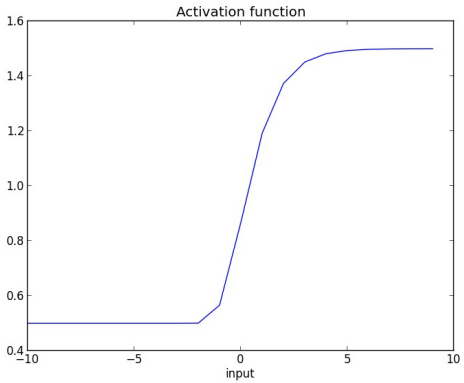
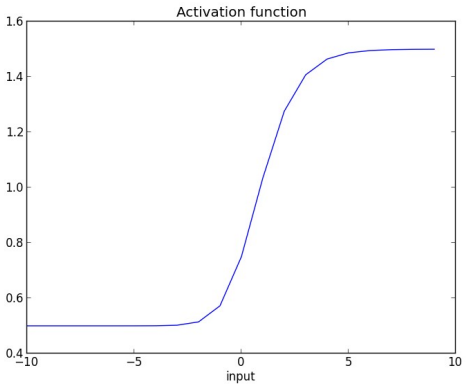
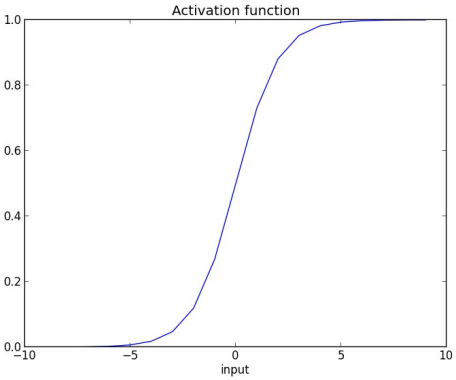
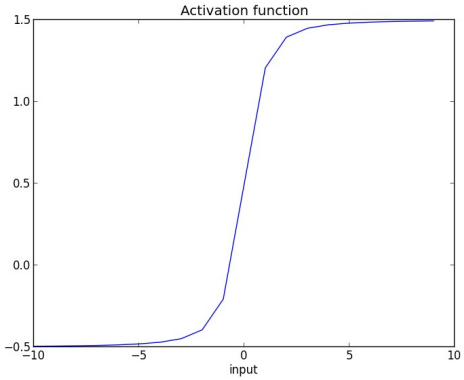
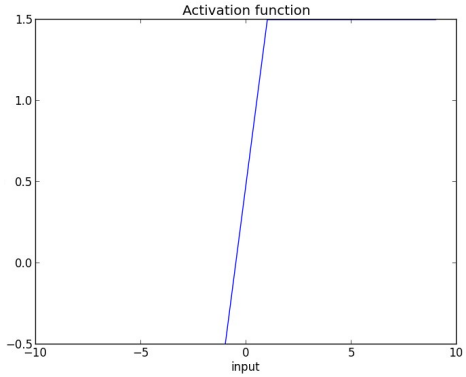
۸	Elliott [۵۹]	$f(x)=\frac{0.5x}{1+\lvert x\rvert}+0.5$	$f'(x)=\frac{0.5}{(1+\lvert x\rvert)^2}$	[۰,۱]
۹	Gaussian	$f(x)=e^{-x^2}$	$f'(x)=-2xe^{-x^2}$	[۰,۱]
۱۰	logarithmic*	$f(x)=\begin{cases}\ln(1+x)+0.5 & x\geq 0\\ -\ln(1-x)+0.5 & x<0\end{cases}$	$f'(x)=\begin{cases}\frac{1}{x+1} & x\geq 0\\ \frac{1}{1-x} & x<0\end{cases}$	[-∞, +∞]
۱۱	loglog [۳۰]*	$f(x)=e^{-e^{-x}}+0.5$	$f'(x)=e^{-e^{-x}-x}$	[۰,۵ ,۱,۵]
۱۲	logsigm [۲۸]*	$f(x)=\left(\frac{1}{1+e^{-x}}\right)^2+0.5$	$f'(x)=\frac{2e^{-x}}{(e^{-x}+1)^3}$	[۰,۵ ,۱,۵]
۱۳	log-sigmoid	$f(x)=\frac{1}{1+e^{-x}}$	$f'(x)=\frac{e^{-x}}{(e^{-x}+1)^2}$	[۰,۱]
۱۴	modified Elliott [۶۰]	$f(x)=\frac{x}{\sqrt{1+x^2}}+0.5$	$f'(x)=\frac{1}{(x^2+1)^{3/2}}$	[-۰,۵ ,۱,۵]
۱۵	rootsig [۲۷]*	$f(x)=\frac{x}{1+\sqrt{1+x^2}}+0.5$	$f'(x)=\frac{1}{\sqrt{x^2+1}+x^2+1}$	[-۰,۵ ,۱,۵]
۱۶	saturated*	$f(x)=\frac{\lvert x+1\rvert-\lvert x-1\rvert}{2}+0.5$	$f'(x)=\frac{\frac{x+1}{\lvert x+1\rvert}-\frac{x-1}{\lvert x-1\rvert}}{2}$	[-۰,۵ ,۱,۵]

جدول (۳-۱): (ادامه)

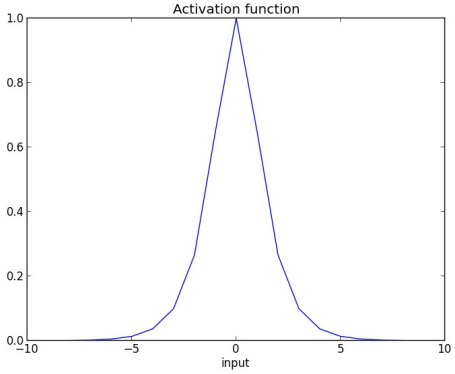
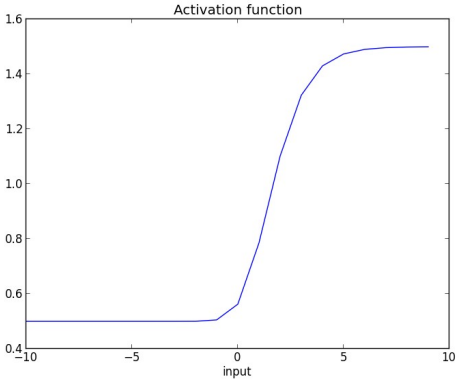
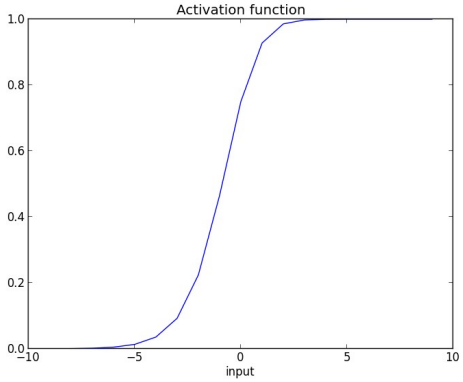
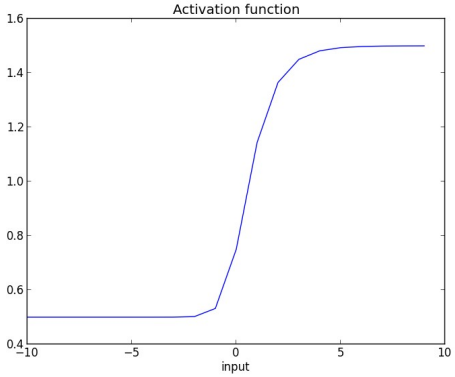
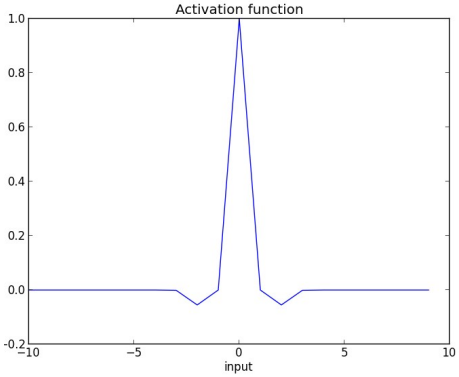
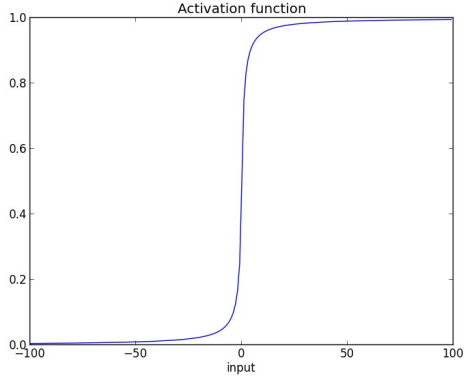
۱۷	Sech	$f(x)=\frac{2}{e^{x}+e^{-x}}$	$f'(x)=-\frac{2(e^{x}-e^{-x})}{(e^{x}+e^{-x})^{2}}$	[۰,۱]
۱۸	sigmoidalm [۲۸]*	$f(x)=\left(\frac{1}{1+e^{-x}}\right)^{4}+0.5$	$f'(x)=\frac{4e^{-x}}{(e^{-x}+1)^{5}}$	[۰,۵ ,۱,۵]
۱۹	Sigmoidalm2 [۶۱]*	$f(x)=\left(\frac{1}{1+e^{-x/2}}\right)^{4}+0.5$	$f'(x)=\frac{2e^{-x/2}}{(e^{-x/2}+1)^{5}}$	[۰,۵ ,۱,۵]
۲۰	sigt [۵۷]	$f(x)=\frac{1}{1+e^{-x}}+\frac{1}{1+e^{-x}}\left(1-\frac{1}{1+e^{-x}}\right)$	$f'(x)=\frac{2e^{x}}{(e^{x}+1)^{3}}$	[۰,۱]
۲۱	skewed-sig [۵۸]*	$f(x)=\left(\frac{1}{1+e^{-x}}\right)\left(\frac{1}{1+e^{-2x}}\right)+0.5$	$f'(x)=\frac{(e^{2x}+2e^{x}+3)e^{3x}}{(e^{x}+1)^{2}(e^{2x}+1)^{2}}$	[۰,۵ ,۱,۵]
۲۲	softsign [۵۹]*	$f(x)=\frac{x}{1+\lvert x\rvert}+0.5$	$f'(x)=\frac{1}{(1+\lvert x\rvert)^{2}}$	[-۰,۵ ,۱,۵]
۲۳	wave [۶۲]	$f(x)=(1-x^{2})e^{-x^{2}}$	$f'(x)=2x(x^{2}-2)e^{-x^{2}}$	[-۰,۰۵۵ ,۱]

Bi-sig1	Bi-sig2	Bi-tanh1
		
Bi-tanh2	**Cloglog**	**Cloglogm**
		

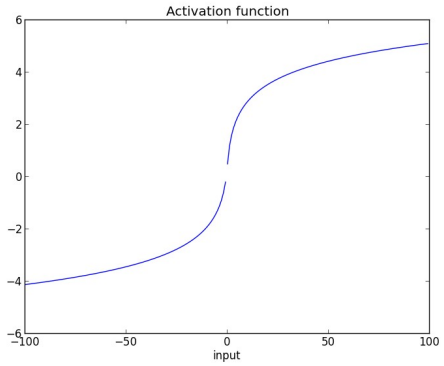
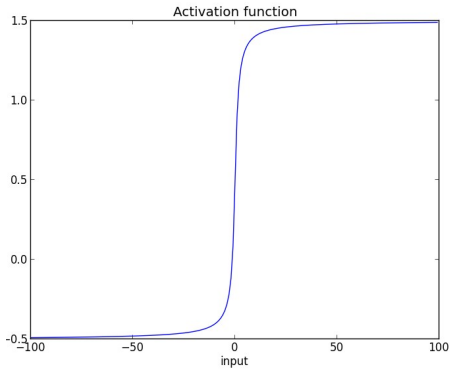
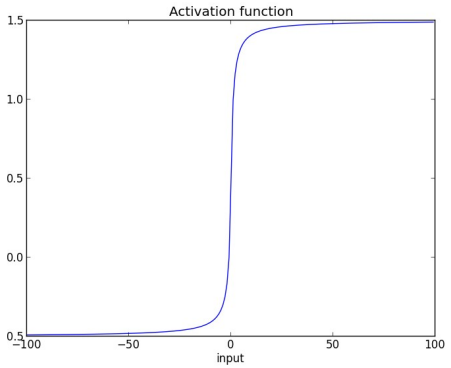
شکل (۳-۴): شکل توابع فعالساز مورد استفاده. به ترتیب از بالا به پایین و چپ به راست شامل توابع فعالساز Bi-sig1، Bi-sig2، Bi-tanh1، Bi-tanh2، cloglog، cloglogm، Gaussian، loglog، logsigm، log-sigmoid، modified Elliott، saturated، sech، sigmoidalm، sigt، skewed-sig، wave، Elliott، logarithmic، rootsig، softsign، Aranda و sigmoidalm2

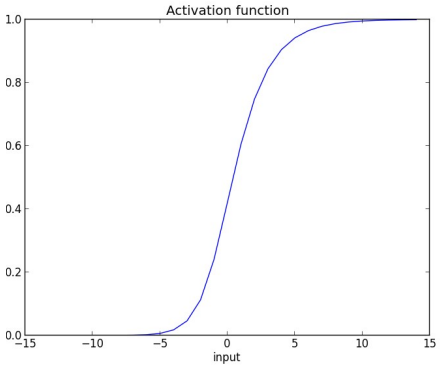
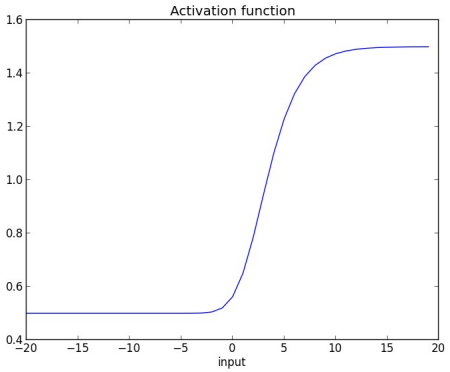
Gaussian	Loglog	Logsigm
		
Log-sigmoid	**modified Elliott**	**Saturated**
		

شکل (۳-۴): (ادامه)

Sech	Sigmoidalm	Sigt
		
Skewed-sig	**Wave**	**Elliott**
		

شکل (۳-۴): (ادامه)

Logarithmic	Rootsig	Softsign
		

Aranda	Sigmoidalm2
	

شکل (۳-۴): (ادامه)

۳-۴- مجموعه داده

برای کلاسه‌بندی توسط شبکه حافظه کوتاه و بلندمدت دو مجموعه داده مرتبط به مرور نظرات فیلم به‌صورت مثبت و منفی (احساسی) استفاده شد. اولین مجموعه داده در این پایان‌نامه Movie Review[۴۷] [۸۲] و دیگری IMDB[۴۸] [۸۳] نام‌گذاری شدند.

مجموعه داده Movie Review شامل ۱۰۶۶۲ جمله نقد فیلم است که از این تعداد ۵۳۳۱ جمله آن نقد مثبت و همین تعداد جمله نقد منفی در آن وجود دارد. برای شبکه موردنظر از ۸۱۶۲ جمله در مجموعه آموزشی و باقی در مجموعه آزمایشی مورد استفاده قرار گرفت. تعداد نقدهای مثبت و منفی در هر قسمت مجموعه آموزشی و مجموعه آزمایشی به‌صورت برابر در نظر گرفته شد. نتایج این مجموعه داده توسط Cheeti [۸۴] نشان داده است که به وسیله شبکه SVM دارای دقت ۷۴٫۳۳٪ می باشد. همچنین Kim و همکارانش [۸۵] نشان داده اند که به وسیله کانولوشن کرنل های متفاوت دارای میانگین دقت ۷۸٪ می باشد. نتایج این مجموعه داده توسط Dai و Le [۸۶] بر روی شبکه حافظه کوتاه و بلند مدت ساده دقتی برابر با ۷۶٫۷٪ داشته است. در این پایان نامه با تابع فعالساز پیشنهادی modified Elliott و شبکه حافظه کوتاه و بلند مدت ساده میانگین دقتی برابر ۷۷٫۶۴٪ برای این مجموعه داده بدست آمده است (بالاترین دقت ثبت شده با این تابع برابر ۷۸٫۱۲٪ بود).

از مجموعه داده IMDB, ۲۰۰۰ جمله برای مجموعه آموزشی و ۵۰۰ جمله برای مجموعه آزمایشی در نظر گرفته شد. تعداد نقدهای مثبت و منفی در هر قسمت مجموعه آموزشی و مجموعه آزمایشی این مجموعه داده نیز به‌صورت برابر در نظر گرفته شد. همچنین طول جملات در نظر گرفته شده کمتر ۱۰۰ کلمه در هر جمله است. نتایج این مجموعه داده توسط Fernandez [۸۷] نشان داده است که به وسیله رده بند SGD به صورت ساده دارای دقتی برابر ۸۸٪ می باشد و همچنین توسط رده بند RF دارای دقتی برابر با ۸۲٪ می باشد. نتایج این مجموعه داده توسط Dai و Le [۸۶] بر روی شبکه حافظه کوتاه و بلند مدت ساده دقتی برابر با ۸۶٫۵٪ داشته است. در این پایان نامه با تابع فعالساز پیشنهادی modified

[۴۷] http://www.cs.cornell.edu/people/pabo/movie-review-data/rt-polaritydata.tar.gz
[۴۸] http://ai.stanford.edu/amaas/data/sentiment/

Elliott و شبکه حافظه کوتاه و بلند مدت ساده میانگین دقتی برابر ۸۷/۵۴٪ برای این مجموعه داده بدست آمده است (بالاترین دقت ثبت شده با این تابع برابر ۸۸/۴٪ بود).

۳-۵- نتیجه‌گیری

در این فصل ابتدا الگوریتم برای شبکه حافظه کوتاه و بلندمدت مورد استفاده معرفی شد. سپس ۲۳ تابع فعالساز مورد استفاده در دروازه های سیگموئیدی این شبکه معرفی شدند. درنهایت دو مجموعه داده مورد استفاده برای ارزیابی عملکرد شبکه روی این توابع فعالساز معرفی شدند.

۴- فصل چهارم: نتایج و نتیجه‌گیری

۴-۱- مقدمه

در این بخش ابتدا نتایج بدست آمده از آزمایش ها توسط جداول و شکل‌های مربوط به دو مجموعه داده مورد بحث و بررسی قرار می گیرد و سپس نتیجه‌گیری صورت می پذیرد.

۴-۲- نتایج

برای بررسی عملکرد شبکه آزمایش‌های گوناگونی با تعداد واحد در لایه مخفی و تغییر توابع فعالساز موجود در جدول (۳-۱) در دروازه‌های سیگموئیدی ورودی، خروجی و فراموشی شبکه حافظه کوتاه و بلندمدت انجام شد. همچنین از روش دسته‌بندی استفاده شد جایی که اندازه مجموعه برای فاز آموزشی ۱۶ و برای فاز آزمایشی ۶۴ استفاده شد. انتخاب اندازه مجموعه بر اساس نتایج آزمایش‌های عملی به‌دست‌آمده است. برای الگوریتم یادگیری از الگوریتم انتشار رو به عقب در طی زمان به همراه بهینه‌ساز الگوریتم روش نرخ یادگیری انطباقی با پارامتر اپسیلون برابر با $۱۰^{-۶}$ استفاده شده است. پارامترهای گوناگون در شبکه به‌صورت ثابت استفاده شده است و تغییر داده نشده اند. خطای اشتباه در رده‌بندی برای رتبه‌بندی توابع فعالساز استفاده شد. هر آزمایش ۳ بار تکرار شده است.

آزمایش توابع فعالساز روی هرکدام از مجموعه داده‌ها ۳ بار تکرار شد. نتایج میانگین خطا به همراه بازه اطمینان[۴۹] ٪۹۵ برای مجموعه داده Movie Review در جدول (۴-۱) به نمایش در آمده است. همچنین تعداد بلوک شبکه حافظه کوتاه و بلندمدت در لایه مخفی بر روی مجموعه {۲, ۴, ۸, ۱۶, ۳۲, ۶۴} گرفته شد. تعداد تکرار در این مجموعه داده ۲۰ تکرار برای هر آزمایش در نظر گرفته شد. در مجموعه داده Movie Review بهترین میانگین خطا برای تابع فعالساز *modified Elliott* با درصد خطای میانگین ٪۲۲٫۳۶ است. تعداد واحد مخفی برای کمترین میزان میانگین خطا ۲ واحد بود. بعد از تابع *modified Elliott* با بازه [۱٫۵, ۰٫۵-]، توابع *cloglogm* ([۱٫۵, ۰٫۵-])، *log-sigmoid* ([۱،۰]), *Bi-*

[۴۹] Confidence interval

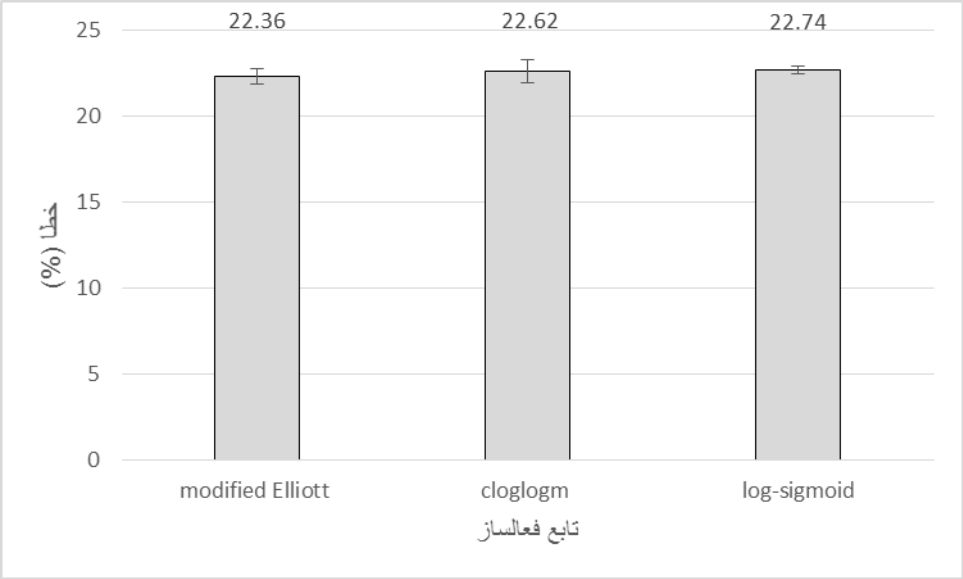
tanh1 ([۱٫۵, ۰٫۵-]) و *saturated* ([۱٫۵, ۰٫۵-]) با به ترتیب ٪۲۲٫۶۲, ٪۲۲٫۷۴, ٪۲۲٫۸ و ٪۲۲٫۸ دارای کمترین میانگین خطا بودند.

نتایج میانگین خطا به همراه بازه اطمینان ٪۹۵ برای مجموعه داده IMDB در جدول (۴-۲) به نمایش در آمده است. همچنین تعداد بلوک شبکه حافظه کوتاه و بلندمدت در لایه مخفی بر روی مجموعه {۴, ۸, ۱۶, ۳۲, ۶۴, ۱۲۸, ۲۵۶} گرفته شد. تعداد تکرار در این مجموعه داده ۵۰ تکرار برای هر آزمایش در نظر گرفته شد. در مجموعه داده IMDB بهترین میانگین خطا متعلق به تابع *modified Elliott* با میانگین خطای ٪۱۲٫۴۶ بود. تعداد واحد مخفی برای کمترین میزان میانگین خطا ۲۵۶ واحد بود. بعد از تابع *modified Elliott* با بازه [۱٫۵, ۰٫۵-], توابع *cloglogm* ([۱٫۵, ۰٫۵-]), *saturated* ([۱٫۵, ۰٫۵-]) و *Bi-sig2* ([۰،۱]) با به ترتیب ٪۱۲٫۸۶, ٪۱۲٫۸۶ و ٪۱۲٫۸۶ دارای کمترین میانگین خطا بودند. در این مجموعه داده تابع *log-sigmoid* با میانگین خطای ٪۱۳٫۶ دارای مکان یازدهم از ۲۳ تابع فعالساز مورد استفاده بود. مقایسه نتایج توابع فعالساز *modified Elliott*, *cloglogm* و *log-sigmoid* بر روی مجموعه داده Movie Review و IMDB به ترتیب در شکل های (۴-۱) و (۴-۲) نمایش داده شده است، جایی که بازه اطمینان ٪۹۵ و میانگین خطا هر سه تابع نمایش داده شده است. همچنین شکل سه تابع فعالساز *modified Elliott*, *cloglogm* و *log-sigmoid* به‌صورت مقایسه‌ای در شکل (۴-۳) نمایش داده شده است، بر اساس شکل (۴-۳), *modified Elliott* و *cloglogm* حول نقطه صفر دارای شیب بیشتری نسبت به *log-sigmoid* هستند و همچنین دارای بازه بزرگ‌تری نیز می‌باشند. همچنین نمودار میانگین خطا برای مجموعه داده Movie Review و IMDB به ترتیب در شکل‌های (۴-۴) و (۴-۵) نمایش داده شده است.

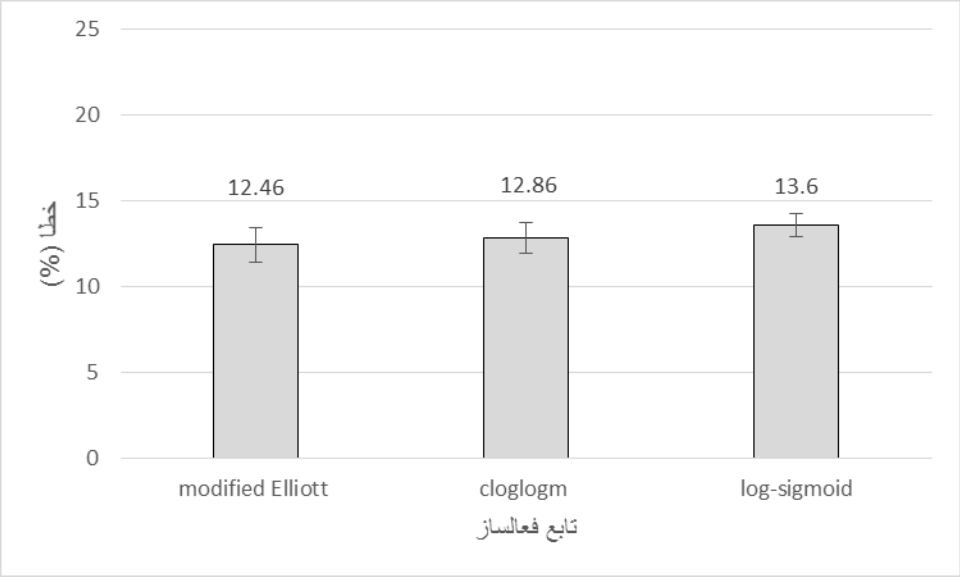
نتایج نشان می‌دهد که برای هر دو مجموعه داده تابع فعالساز *modified Elliott* دارای بهترین عملکرد نسبت به ۲۲ تابع فعالساز دیگر است. استفاده از این تابع فعالساز در مواردی که مجموعه داده بزرگ‌تر باشد نیاز به تنظیم پارامترها (مثل اپسیلون با عدد کوچک‌تر در بهینه‌ساز الگوریتم روش نرخ یادگیری انطباقی) دارد. به‌طور کلی نشان داده شد که تابع فعالساز *log-sigmoid* که بیشترین مورد استفاده در شبکه‌های عصبی و به‌خصوص شبکه حافظه کوتاه و بلندمدت را دارد دارای بهترین عملکرد در دروازه‌های سیگموئیدی نیست و بهتر است توابع فعالساز جایگزین استفاده شوند. همچنین نشان داده شد

بازه [۱/۵, ۰/۵-] برای توابع فعالساز بهتر از [۰،۱] در دروازه‌های سیگموئیدی عمل می‌کند. به دلیل اینکه حداکثر طول جمله‌ها در مجموعه داده Movie Review برابر با ۶۴ کلمه است در حالی که حداکثر طول در نظر گرفته شده برای مجموعه داده IMDB, ۱۰۰ کلمه است، شبکه حافظه کوتاه و بلندمدت برای مجموعه داده Movie Review نیازمند تعداد بلوک در لایه مخفی کمتری است. این در حالی است که با توجه به طول بیشتر و پیچیدگی بیشتر مجموعه داده IMDB شبکه برای همگرایی نیازمند تعداد بلوک بیشتر در لایه مخفی و همین‌طور تعداد تکرار بیشتر است.

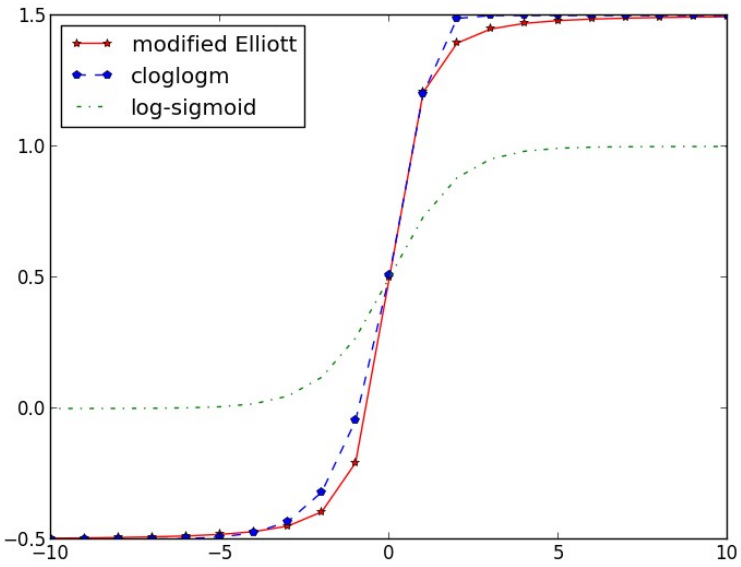
در این پایان‌نامه تابع فعالساز *modified Elliott* نسبت به ۲۲ تابع فعالساز مورد استفاده دیگر دارای نتایج بهتری بود. در مقاله‌ای از Burhani و همکارانش [۷۴] نشان داده شده است که در رهیافت حل مشکل بیش برآزش، تابع فعالساز *modified Elliott* دارای بهترین نتایج بخصوص در مقایسه با تابع فعالساز سیگموئید است. همچنین در این پایان‌نامه تابع فعالساز *cloglogm* دارای رتبه دوم بهترین نتایج بود که توسط Gomes و همکارانش [۴۱] نشان داده شده است که تابع *cloglogm* نتایج مناسبی برای پیش‌بینی سری‌های زمانی مالی دارد.



شکل (۴-۱): مقایسه کمترین میانگین خطا (به همراه بازه اطمینان ۹۵٪) برای توابع فعالساز modified Elliott, cloglogm و log-sigmoid در مجموعه داده Movie Review.



شکل (۴-۲): مقایسه کمترین میانگین خطا (به همراه بازه اطمینان ۹۵٪) برای توابع فعالساز modified Elliott, cloglogm و log-sigmoid در مجموعه داده IMDB.



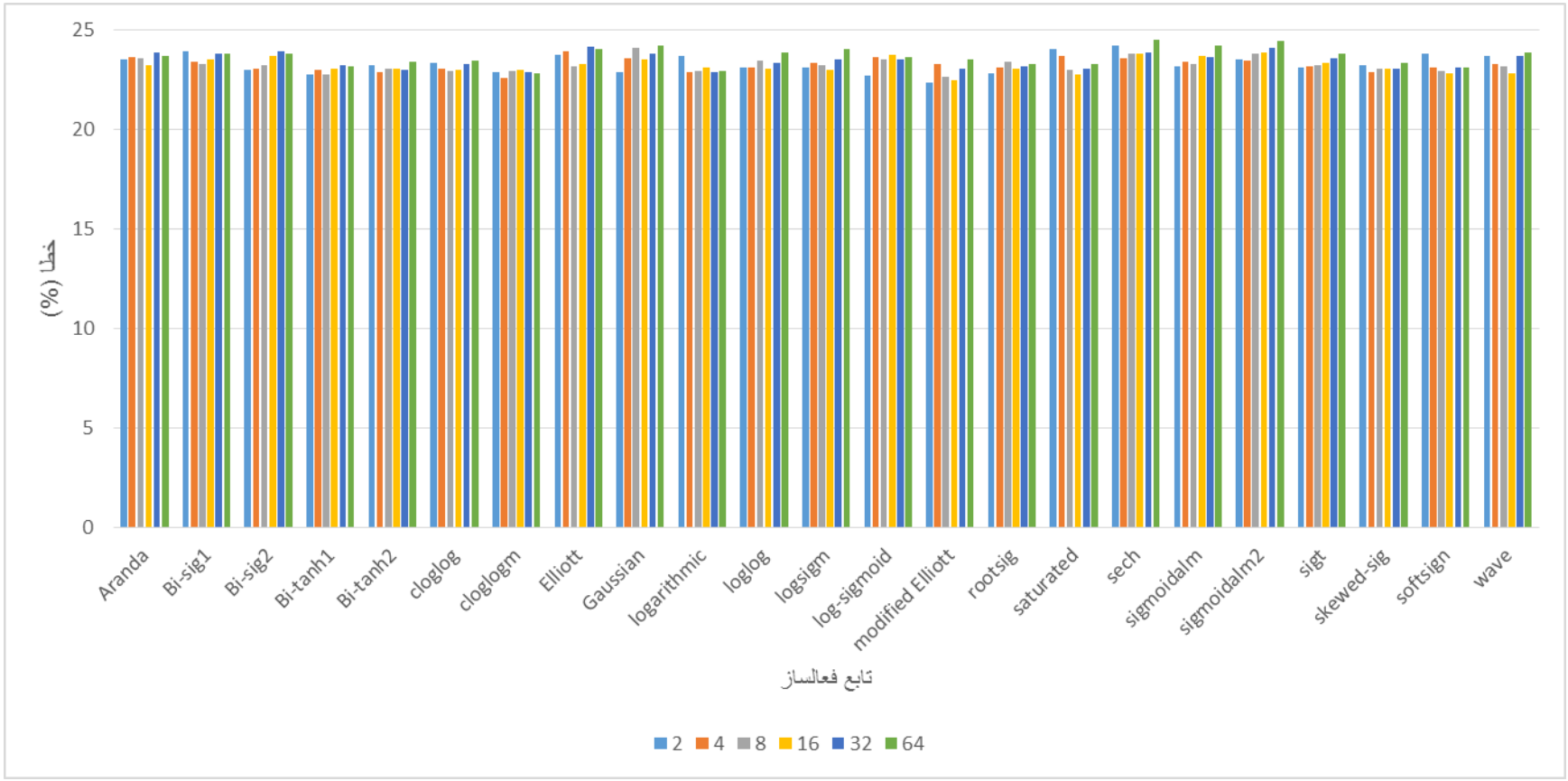
شکل (۴-۳): مقایسه شکل توابع فعالساز modified Elliott, cloglogm و log-sigmoid.

جدول (۴-۱): نتایج خطای میانگین برای مجموعه داده Movie Review. عدد داخل پرانتز بازه اطمینان ٪۹۵ است. کمترین میانگین برای هر تابع فعالساز پُر رنگ شده است. کمترین میانگین کلی با گذاشتن خط زیر آن مشخص شده است.

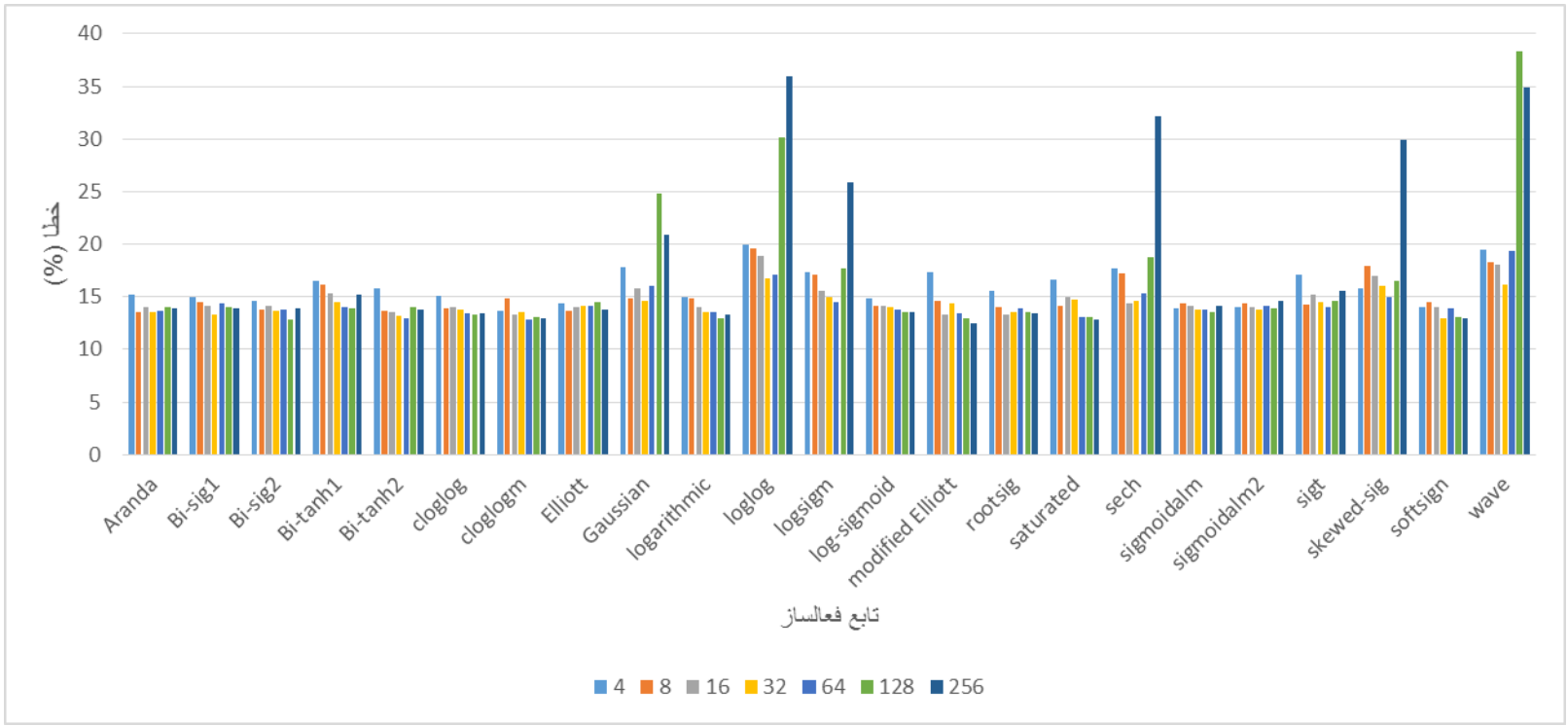
تعداد بلوک در لایه مخفی / تابع فعالساز	۲	۴	۸	۱۶	۳۲	۶۴
Aranda	۲۳٫۵۶ (۲۳٫۰۳-۲۴٫۰۸)	۲۳٫۶۸ (۲۲٫۲۸-۲۵٫۰۷)	۲۳٫۶۲ (۲۲٫۲۷-۲۴٫۹۸)	**۲۳٫۲۴** (۲۲٫۴۵-۲۴٫۰۲)	۲۳٫۸۸ (۲۳٫۲۸-۲۴٫۴۷)	۲۳٫۷۴ (۲۲٫۷۸-۲۴٫۷۱)
Bi-sig1	۲۳٫۹۷ (۲۲٫۲۶-۲۵٫۶۷)	۲۳٫۴۵ (۲۲٫۳۶-۲۴٫۵۴)	**۲۳٫۳۳** (۲۳٫۰۱-۲۳٫۶۵)	۲۳٫۵۲ (۲۲٫۵۱-۲۴٫۵۲)	۲۳٫۸۶ (۲۳٫۵۱-۲۴٫۲۱)	۲۳٫۸۵ (۲۳٫۴۵-۲۴٫۲۵)
Bi-sig2	**۲۳٫۰۱** (۲۲٫۱۶-۲۳٫۸۵)	۲۳٫۰۹ (۲۱٫۱۳-۲۵٫۰۵)	۲۳٫۲۴ (۲۲٫۸-۲۳٫۶۷)	۲۳٫۷۲ (۲۲٫۹-۲۴٫۵۳)	۲۳٫۹۲ (۲۲٫۵۲-۲۵٫۳۱)	۲۳٫۸۶ (۲۲٫۵۸-۲۵٫۱۴)
Bi-tanh1	۲۲٫۸۱ (۲۱٫۶۹-۲۳٫۹۳)	۲۳٫۰۱ (۲۲٫۱۱-۲۳٫۹۱)	**۲۲٫۸** (۲۱٫۵۹-۲۴)	۲۳٫۱ (۲۲٫۴۴-۲۳٫۷۶)	۲۳٫۲۶ (۲۲٫۱۱-۲۴٫۴۱)	۲۳٫۲ (۲۲٫۷۶-۲۳٫۶۳)
Bi-tanh2	۲۳٫۲۶ (۲۲٫۳۲-۲۴٫۲۱)	**۲۲٫۹۲** (۲۲-۲۳٫۸۳)	۲۳٫۰۶ (۲۲٫۲۳-۲۳٫۹)	۲۳٫۱ (۲۲٫۸-۲۳٫۴۱)	۲۳ (۲۲٫۶۴-۲۳٫۳۵)	۲۳٫۴ (۲۲٫۵۳-۲۴٫۲۶)
cloglog	۲۳٫۳۷ (۲۲٫۴۵-۲۴٫۲۹)	۲۳٫۰۶ (۲۲٫۳۶-۲۳٫۷۷)	**۲۲٫۹۸** (۲۲٫۳۷-۲۳٫۵۹)	۲۳ (۲۲٫۳۹-۲۳٫۶)	۲۳٫۳ (۲۲٫۶۶-۲۳٫۹۵)	۲۳٫۵ (۲۲٫۷۷-۲۴٫۲۳)
cloglogm	۲۲٫۹۳ (۲۲٫۶۱-۲۳٫۲۵)	**۲۲٫۶۲** (۲۱٫۱۱-۲۴٫۱۳)	۲۲٫۹۸ (۲۲٫۳۷-۲۳٫۵۹)	۲۳٫۰۱ (۲۲٫۶۳-۲۳٫۳۸)	۲۲٫۸۹ (۲۲٫۱۴-۲۳٫۶۳)	۲۲٫۸۴ (۲۱٫۸۸-۲۳٫۷۹)
Elliott	۲۳٫۸ (۲۲٫۰۵-۲۵٫۵۴)	۲۳٫۹۷ (۲۰٫۸۵-۲۷٫۰۹)	**۲۳٫۲** (۲۲٫۰۲-۲۴٫۳۷)	۲۳٫۳۳ (۲۲٫۴۸-۲۴٫۱۷)	۲۴٫۱۷ (۲۳٫۷۷-۲۴٫۵۷)	۲۴٫۰۵ (۲۳٫۶۷-۲۴٫۴۲)
Gaussian	**۲۲٫۸۸** (۲۱٫۲۵-۲۴٫۵۱)	۲۳٫۵۸ (۲۳٫۲۹-۲۳٫۸۷)	۲۴٫۱۳ (۲۲٫۷۳-۲۵٫۵۲)	۲۳٫۵۳ (۲۳٫۱۳-۲۳٫۹۳)	۲۳٫۸۵ (۲۳٫۲۷-۲۴٫۴۳)	۲۴٫۲۲ (۲۳٫۱۱-۲۵٫۳۳)
logarithmic	۲۳٫۶۹ (۲۲٫۲۴-۲۵٫۱۴)	۲۲٫۹ (۲۱٫۹۱-۲۳٫۹)	۲۲٫۹۴ (۲۲٫۵۴-۲۳٫۳۴)	۲۳٫۱۲ (۲۲٫۳۳-۲۳٫۹)	**۲۲٫۸۸** (۲۲٫۷۸-۲۲٫۹۷)	۲۲٫۹۴ (۲۲٫۵۴-۲۳٫۳۴)
loglog	۲۳٫۱۶ (۲۱٫۷۸-۲۴٫۵۳)	۲۳٫۱۶ (۲۲٫۶-۲۳٫۷۱)	۲۳٫۴۸ (۲۳٫۲۱-۲۳٫۷۴)	**۲۳٫۱** (۲۲٫۸۵-۲۳٫۳۵)	۲۳٫۳۸ (۲۳٫۰۳-۲۳٫۷۳)	۲۳٫۹ (۲۳٫۶۵-۲۴٫۱۵)
logsigm	۲۳٫۱۶ (۲۲٫۵۵-۲۳٫۷۶)	۲۳٫۳۶ (۲۲٫۶-۲۴٫۱۱)	۲۳٫۲۴ (۲۲٫۰۳-۲۴٫۴۴)	**۲۳٫۰۱** (۲۲٫۴۶-۲۳٫۵۶)	۲۳٫۵۴ (۲۲٫۹۴-۲۴٫۱۴)	۲۴٫۰۸ (۲۳٫۲۹-۲۴٫۸۶)
log-sigmoid	**۲۲٫۷۴** (۲۲٫۲۷-۲۳٫۲۱)	۲۳٫۶۸ (۲۱٫۲۹-۲۶٫۰۶)	۲۳٫۵۲ (۲۲٫۱۵-۲۴٫۸۸)	۲۳٫۷۸ (۲۳٫۰۸-۲۴٫۴۹)	۲۳٫۵۲ (۲۳٫۱۶-۲۳٫۸۷)	۲۳٫۶۵ (۲۳٫۵-۲۳٫۸)
modified Elliott	**۲۲٫۳۶** (۲۱٫۴۱-۲۳٫۳)	۲۳٫۳۳ (۲۲٫۲۷-۲۴٫۳۹)	۲۲٫۶۸ (۲۱٫۵۳-۲۳٫۸۲)	۲۲٫۵۲ (۲۱٫۰۵-۲۳٫۹۸)	۲۳٫۰۶ (۲۲٫۶۵-۲۳٫۴۸)	۲۳٫۵۷ (۲۲٫۹۷-۲۴٫۱۷)
rootsig	**۲۲٫۸۴** (۲۱٫۵۳-۲۴٫۱۴)	۲۳٫۱۲ (۲۲٫۳۴-۲۳٫۸۹)	۲۳٫۴۵ (۲۳٫۲-۲۳٫۷)	۲۳٫۱ (۲۲٫۹۹-۲۳٫۲۲)	۲۳٫۲۲ (۲۲٫۱۷-۲۴٫۲۷)	۲۳٫۳۲ (۲۲٫۰۱-۲۴٫۶۲)
saturated	۲۴٫۰۹ (۲۱٫۰۲-۲۷٫۱۶)	۲۳٫۷۲ (۲۰٫۸۳-۲۶٫۶)	۲۳٫۰۴ (۲۱٫۷۹-۲۴٫۲۸)	**۲۲٫۸** (۲۲٫۷-۲۲٫۸۹)	۲۳٫۰۶ (۲۲٫۲۵-۲۳٫۸۷)	۲۳٫۲۹ (۲۲٫۳۷-۲۴٫۲۱)
sech	۲۴٫۲۴ (۲۲٫۶۵-۲۵٫۸۲)	**۲۳٫۶۲** (۲۲٫۸۸-۲۴٫۳۷)	۲۳٫۸۵ (۲۲٫۸۱-۲۴٫۸۸)	۲۳٫۸۵ (۲۳٫۰۴-۲۴٫۶۶)	۲۳٫۸۹ (۲۳٫۳۲-۲۴٫۴۵)	۲۴٫۵۲ (۲۲٫۶۲-۲۶٫۴۱)
sigmoidalm	**۲۳٫۲۲** (۲۲٫۵۵-۲۳٫۹)	۲۳٫۴ (۲۲٫۶۸-۲۴٫۱۱)	۲۳٫۳۳ (۲۳٫۰۴-۲۳٫۶۲)	۲۳٫۷ (۲۲٫۸۷-۲۴٫۵۴)	۲۳٫۶۶ (۲۲٫۸۵-۲۴٫۴۷)	۲۴٫۲۵ (۲۳٫۶۸-۲۴٫۸۱)
sigmoidalm2	۲۳٫۵۶ (۲۲٫۱۵-۲۴٫۹۶)	**۲۳٫۴۹** (۲۳٫۱۱-۲۳٫۸۶)	۲۳٫۸۵ (۲۳٫۲۸-۲۴٫۴۱)	۲۳٫۸۸ (۲۲٫۵۷-۲۵٫۱۸)	۲۴٫۱۴ (۲۳٫۲۵-۲۵٫۰۳)	۲۴٫۴۸ (۲۴٫۰۴-۲۴٫۹۱)
sigt	**۲۳٫۱۲** (۲۲٫۱۴-۲۴٫۰۹)	۲۳٫۱۷ (۲۱٫۸۹-۲۴٫۴۵)	۲۳٫۲۴ (۲۲٫۶۳-۲۳٫۸۴)	۲۳٫۳۶ (۲۲٫۹۲-۲۳٫۷۹)	۲۳٫۶ (۲۳٫۰۸-۲۴٫۱۱)	۲۳٫۸۲ (۲۳٫۲۱-۲۴٫۴۳)
skewed-sig	۲۳٫۲۶ (۲۲٫۱-۲۴٫۴۲)	**۲۲٫۹۲** (۲۲٫۶۵-۲۳٫۱۸)	۲۳٫۰۹ (۲۱٫۳۴-۲۴٫۸۴)	۲۳٫۰۹ (۲۲٫۹۷-۲۳٫۲)	۲۳٫۰۹ (۲۱٫۰۶-۲۵٫۱۲)	۲۳٫۳۴ (۲۱٫۹۹-۲۴٫۷)
softsign	۲۳٫۸۱ (۲۲٫۳۶-۲۵٫۲۶)	۲۳٫۱۴ (۲۱٫۵۵-۲۴٫۷۴)	۲۲٫۹۴ (۲۱٫۷۹-۲۴٫۰۹)	**۲۲٫۸۵** (۲۲٫۳۴-۲۳٫۳۶)	۲۳٫۱۴ (۲۲٫۲۲-۲۴٫۰۶)	۲۳٫۱۲ (۲۱٫۹۷-۲۴٫۲۶)
wave	۲۳٫۶۹ (۱۹٫۴۶-۲۷٫۹۲)	۲۳٫۳ (۲۲٫۲۵-۲۴٫۳۵)	۲۳٫۱۸ (۲۲٫۷۸-۲۳٫۵۸)	**۲۲٫۸۲** (۲۱٫۹۸-۲۳٫۶۷)	۲۳٫۷۲ (۲۲٫۲۳-۲۵٫۲)	۲۳٫۹ (۲۲٫۹۸-۲۴٫۸۲)

جدول (۴-۲): نتایج خطای میانگین برای مجموعه داده IMDB. عدد داخل پرانتز بازه اطمینان ٪۹۵ است. کمترین میانگین برای هر تابع فعالساز پُر رنگ شده است. کمترین میانگین کلی با گذاشتن خط زیر آن مشخص شده است.

تعداد بلوک در لایه مخفی / تابع فعالساز	۴	۸	۱۶	۳۲	۶۴	۱۲۸	۲۵۶
Aranda	۱۵/۲ (۱۰/۷۸-۱۹/۶۱)	**۱۳/۶** (۱۳/۱-۱۴/۰۹)	۱۴ (۱۳-۱۴/۹۹)	**۱۳/۶** (۱۲/۲۸-۱۴/۹۱)	۱۳/۶۶ (۱۲/۴۱-۱۴/۹۱)	۱۴/۰۶ (۱۳/۷۷-۱۴/۳۵)	۱۳/۹۳ (۱۱-۹۲-۱۵/۹۴)
Bi-sig1	۱۵ (۱۴/۵-۱۵/۴۹)	۱۴/۴۶ (۱۳/۰۳-۱۵/۹)	۱۴/۲ (۱۳/۳۳-۱۵/۰۶)	**۱۳/۳۳** (۱۲/۵۷-۱۴/۰۹)	۱۴/۴ (۱۲/۱۲-۱۶/۶۷)	۱۴ (۱۳/۵-۱۴/۴۹)	۱۳/۹۳ (۱۲/۶۸-۱۵/۱۸)
Bi-sig2	۱۴/۶۶ (۱۴/۰۹-۱۵/۲۴)	۱۳/۸ (۱۱/۲۱-۱۶/۳۸)	۱۴/۲ (۱۲/۸۸-۱۵/۵۱)	۱۳/۶۶ (۱۲/۱۴-۱۵/۱۸)	۱۳/۷۳ (۱۲/۹۷-۱۴/۴۹)	**۱۲/۸۶** (۱۱/۶۱-۱۴/۱۱)	۱۳/۹۳ (۱۱/۴۸-۱۶/۳۸)
Bi-tanh1	۱۶/۵۳ (۱۱/۷۵-۲۱/۳)	۱۶/۱۳ (۱۳/۶۸-۱۸/۵۸)	۱۵/۳۳ (۱۳/۷۳-۱۶/۹۳)	۱۴/۵۳ (۱۲/۹۳-۱۶/۱۳)	۱۴ (۱۲/۶۸-۱۵/۳۱)	**۱۳/۹۳** (۱۲/۱۸-۱۵/۶۷)	۱۵/۲ (۱۱/۶۱-۱۸/۷۸)
Bi-tanh2	۱۵/۸ (۱۲/۸۱-۱۸/۷۸)	۱۳/۶۶ (۱۱/۷۸-۱۵/۵۴)	۱۳/۶ (۱۱/۶۱-۱۵/۵۸)	۱۳/۲ (۱۲/۷-۱۳/۶۹)	**۱۲/۹۳** (۱۱/۸۹-۱۳/۹۶)	۱۴/۰۶ (۱۲/۸۱-۱۵/۳۱)	۱۳/۸ (۱۲/۴۸-۱۵/۱۱)
cloglog	۱۵/۱۳ (۱۲/۸۳-۱۷/۴۲)	۱۳/۹۳ (۱۲/۰۵-۱۵/۸۱)	۱۴/۰۶ (۱۱/۸۲-۱۶/۳)	۱۳/۸ (۱۲-۱۵/۵۹)	۱۳/۴۶ (۱۲/۷-۱۴/۲۲)	**۱۳/۲۶** (۱۲/۱۱-۱۴/۴۱)	۱۳/۴ (۱۱/۶۷-۱۵/۱۲)
cloglogm	۱۳/۶۶ (۱۲/۰۶-۱۵/۲۶)	۱۴/۸ (۱۰/۴۹-۱۹/۱)	۱۳/۳۳ (۱۲/۵۷-۱۴/۰۹)	۱۳/۵۳ (۱۲/۷۷-۱۴/۲۹)	**۱۲/۸۶** (۱۰/۸۵-۱۴/۸۷)	۱۳/۱۳ (۱۲/۳۷-۱۳/۸۹)	۱۳ (۱۰/۷۲-۱۵/۲۷)
Elliott	۱۴/۴ (۱۱/۹۱-۱۶/۸۸)	**۱۳/۶۶** (۱۱/۵۹-۱۵/۷۳)	۱۴/۰۶ (۱۲/۹۱-۱۵/۲۱)	۱۴/۱۳ (۱۲/۵۳-۱۵/۷۳)	۱۴/۲ (۱۲/۰۳-۱۶/۳۶)	۱۴/۵۳ (۱۳/۹۵-۱۵/۱)	۱۳/۷۳ (۱۰/۴۲-۱۷/۰۴)
Gaussian	۱۷/۸۶ (۳/۷۷-۳۱/۹۵)	۱۴/۸ (۱۰/۳۸-۱۹/۲۱)	۱۵/۸ (۱۰/۲۶-۲۱/۳۳)	**۱۴/۶۶** (۱۴/۳۷-۱۴/۹۵)	۱۶ (۱۲/۲۴-۱۹/۷۵)	۲۴/۸ (۱۷/۵۸-۳۲/۰۱)	۲۰/۹۳ (۱۱/۱۹-۳۰/۶۷)
logarithmic	۱۴/۹۳ (۱۳/۰۵-۱۶/۸۱)	۱۴/۸۶ (۱۳/۲۶-۱۶/۴۶)	۱۴/۰۶ (۱۱/۱۹-۱۶/۹۳)	۱۳/۶ (۱۲/۶-۱۴/۵۹)	۱۳/۶ *	**۱۳** (۱۲/۱۳-۱۳/۸۶)	۱۳/۳۳ (۱۰/۹۸-۱۵/۶۸)
loglog	۲۰ (۱۶/۵۲-۲۳/۴۷)	۱۹/۶ (۷/۷۶-۳۱/۴۳)	۱۸/۸۶ (۱۰/۱۴-۲۷/۵۹)	**۱۶/۸** (۹/۴۸-۲۴/۱۱)	۱۷/۱۳ (۹/۸۹-۲۴/۳۷)	۳۰/۲ (۲۸/۸۸-۳۱/۵۱)	۳۵/۹۳ (۳۲/۱۳-۳۹/۷۲)
logsigm	۱۷/۴ (۱۵/۲۳-۱۹/۵۶)	۱۷/۰۶ (۱۱/۴۱-۲۲/۷۱)	۱۵/۵۳ (۱۱/۶۷-۱۹/۳۹)	۱۴/۹۳ (۱۲/۹۲-۱۶/۹۴)	**۱۴/۵۳** (۱۳/۳۸-۱۵/۶۸)	۱۷/۶۶ (۱۰/۷-۲۴/۶۲)	۲۵/۹۳ (۱/۹۳-۴۹/۹۲)
log-sigmoid	۱۴/۸ (۱۳/۹۳-۱۵/۶۶)	۱۴/۱۳ (۱۳/۳۷-۱۴/۸۹)	۱۴/۲ (۱۲/۸۸-۱۵/۵۱)	۱۴/۰۶ (۱۱/۶۱-۱۶/۵)	۱۳/۷۳ (۱۲/۹۷-۱۴/۴۹)	**۱۳/۶** (۱۲/۱-۱۵/۰۹)	**۱۳/۶** (۱۲/۷۳-۱۴/۴۶)
modified Elliott	۱۷/۳۳ (۱۰/۶۲-۲۴/۰۴)	۱۴/۶ (۱۲/۴۳-۱۶/۷۶)	۱۳/۲۶ (۹/۷۴-۱۶/۷۹)	۱۴/۴ (۱۲/۲۳-۱۶/۵۶)	۱۳/۴۶ (۱۱/۲۲-۱۵/۷)	۱۲/۹۳ (۱۰/۰۲-۱۵/۸۴)	**۱۲/۴۶** (۱۰/۲۲-۱۴/۷)
rootsig	۱۵/۶ (۱۱/۲۶-۱۹/۹۳)	۱۴/۰۶ (۱۱/۸۲-۱۶/۳)	**۱۳/۲۶** (۱۱/۶۶-۱۴/۸۶)	۱۳/۵۳ (۱۲/۷۷-۱۴/۲۹)	۱۳/۸۶ (۱۳/۵۷-۱۴/۱۵)	۱۳/۵۳ (۱۱/۴۶-۱۵/۶)	۱۳/۴ (۱۲/۰۸-۱۴/۷۱)
saturated	۱۶/۶ (۱۲/۷۱-۲۰/۴۸)	۱۴/۱۳ (۱۲/۸۸-۱۵/۳۸)	۱۵ (۱۲/۵۱-۱۷/۴۸)	۱۴/۷۳ (۹/۴۲-۲۰/۰۴)	۱۳/۱۳ (۹/۹۷-۱۶/۲۸)	۱۳/۰۶ (۱۰/۷۱-۱۵/۴۱)	**۱۲/۸۶** (۱۱/۸۳-۱۳/۹)
sech	۱۷/۷۳ (۱۴/۹۹-۲۰/۴۶)	۱۷/۲۶ (۱۰/۸۷-۲۳/۶۵)	**۱۴/۴** (۱۱/۸۱-۱۶/۹۸)	۱۴/۶ (۱۴/۱-۱۵/۰۹)	۱۵/۳۳ (۱۳/۳۲-۱۷/۳۴)	۱۸/۸ (۱۳/۵۶-۲۴/۰۳)	۳۲/۲ (۲۷/۷۸-۳۶/۶۱)
sigmoidalm	۱۳/۹۳ (۱۱/۶۹-۱۶/۱۷)	۱۴/۳۳ (۱۳/۵۷-۱۵/۰۹)	۱۴/۱۳ (۱۱/۶۸-۱۶/۵۸)	۱۳/۸ (۱۲/۰۷-۱۵/۵۲)	۱۳/۷۳ (۱۳/۱۵-۱۴/۳)	**۱۳/۶** (۱۲/۱-۱۵/۰۹)	۱۴/۲ (۱۳/۳۳-۱۵/۰۶)
sigmoidalm2	۱۴/۰۶ (۱۲/۹۱-۱۵/۲۱)	۱۴/۳۳ (۱۳/۱۸-۱۵/۴۸)	۱۴ (۱۲/۰۱-۱۵/۹۸)	**۱۳/۸** (۱۲/۹۳-۱۴/۶۶)	۱۴/۱۳ (۱۳/۳۷-۱۴/۸۹)	۱۳/۸۶ (۱۳/۱-۱۴/۶۲)	۱۴/۶ (۱۳/۶-۱۵/۵۹)
sigt	۱۷/۰۶ (۱۳/۳۳-۲۰/۷۹)	۱۴/۲۶ (۱۳/۶۹-۱۴/۸۴)	۱۵/۲ (۱۲/۷۱-۱۷/۶۸)	۱۴/۴۶ (۱۱/۴۳-۱۷/۵)	**۱۴/۰۶** (۱۱/۶۱-۱۶/۵۱)	۱۴/۶۶ (۱۳/۰۶-۱۶/۲۶)	۱۵/۵۳ (۱۲/۷-۱۸/۳۵)
skewed-sig	۱۵/۸ (۱۰/۰۷-۲۱/۵۲)	۱۷/۹۳ (۱۴/۶۲-۲۱/۲۴)	۱۷ (۱۲/۷۵-۲۱/۲۴)	۱۶/۰۶ (۱۲/۸۷-۱۹/۲۶)	**۱۴/۹۳** (۱۳/۴۱-۱۶/۴۵)	۱۶/۵۳ (۱۳/۶۶-۱۹/۴)	۲۹/۹۳ (۱/۸۲-۵۸/۰۳)
softsign	۱۴/۰۶ (۱۱/۹۹-۱۶/۱۳)	۱۴/۴۶ (۱۳/۲۱-۱۵/۷۱)	۱۴/۰۶ (۱۳/۰۳-۱۵/۱)	**۱۲/۹۳** (۱۲/۳۵-۱۳/۵)	۱۳/۸۶ (۱۱/۳۶-۱۶/۳۶)	۱۳/۱۳ (۱۲/۰۹-۱۴/۱۶)	۱۳ (۱۰/۷۲-۱۵/۲۷)
wave	۱۹,۴۶ (۱۲,۲۲-۲۶,۷)	۱۸,۳۳ (۷,۶۲-۲۹,۰۳)	۱۸,۰۶ (۱۵,۹۹-۲۰,۱۳)	**۱۶,۲** (۱۳,۹۲-۱۸,۴۷)	۱۹,۴ (۱۶,۲۹-۲۲,۵)	۳۸,۳۳ (۲۷,۶۵-۴۹,۰۱)	۳۴,۹۳ (۴,۵۸-۶۵,۲۷)



شکل (۴-۴): نمودار نتایج میانگین خطای آزمایشی برای مجموعه داده Movie Reviewبرای تعداد بلوک های مختلف لایه مخفی .

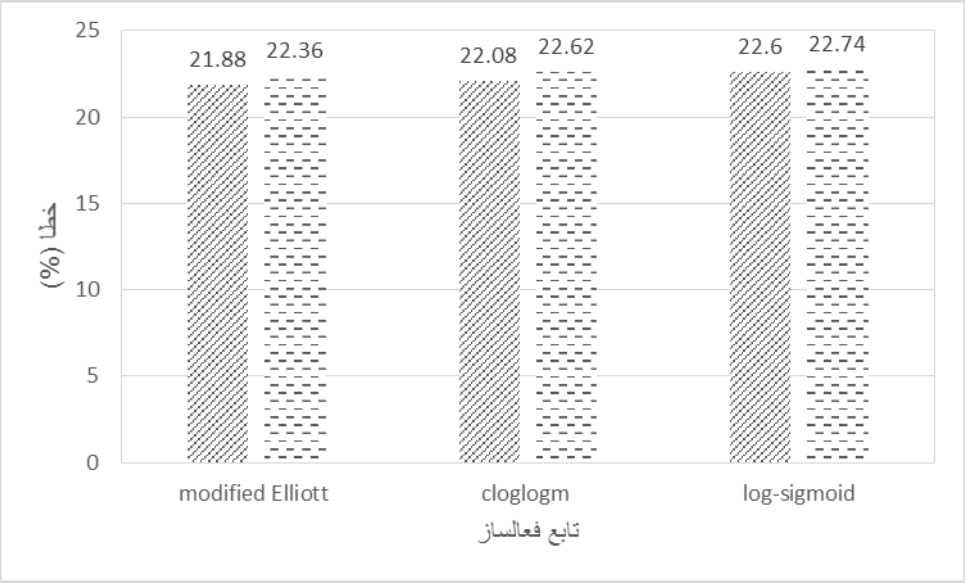


شکل (۴-۵): نمودار نتایج میانگین خطای آزمایشی برای مجموعه داده IMDB برای تعداد بلوک های مختلف لایه مخفی.

آزمایش توابع فعالساز روی هرکدام از مجموعه داده‌ها ۳ بار تکرار شد و نتایج مجموعه داده Movie Review برای حداقل و حداکثر خطای آزمایشی برای هر تابع فعالساز در جدول (۴-۳) قابل مشاهده است. بر روی این مجموعه داده کمترین خطا در هر مرحله برای تابع فعالساز *modified Elliott* با میزان خطای ٪۲۱٫۸۸ بود و همچنین کمترین خطای میانگین نیز برای همین تابع فعالساز با میزان خطای ٪۳۶٫۲۲ بود. تعداد واحد مخفی برای کمترین میزان خطا ۱۶ واحد و برای کمترین میزان میانگین خطا ۲ واحد بودند. بعد از تابع فعالساز *modified Elliott* با بازه [۱٫۵, ۰٫۵-] توابع فعالساز *cloglogm* ([۱٫۵, ۰٫۵-]) و *skewed-sig* ([۱٫۵, ۰٫۵]) و *Gaussian* ([۱, ۰]) و *rootsig* ([۱٫۵, ۰٫۵-]) دارای کمترین خطا با میزان به ترتیب ٪۲۱٫۸۸, ٪۲۲٫۰۸, ٪۲۲٫۱۶, ٪۲۲٫۱۸ و ٪۲۲٫۲۸ بودند.

با سه بار اجرای مجزا روی هر تابع فعالساز، نتایج خطای آزمایشی در اجرای اول برای مجموعه داده Movie Review نشان داد که در این دور تابع *modified Elliott* دارای کمترین میزان خطا با ٪۲۲٫۱۲ بود. در اجرای دوم تابع *modified Elliott* دارای کمترین میزان خطا با ٪۲۱٫۸۸ بود و در اجرای سوم تابع *cloglogm* دارای کمترین میزان خطا با ٪۲۲٫۰۸ بود.

نکته جالب توجه این بود که تابع فعالساز سیگموئید (*log-sigmoid*) در بهترین پنج نتایج اولیه نبود و رتبه آن بین توابع فعالساز مورد استفاده ۱۴ از ۲۳, با حداقل درصد خطای ٪۲۲٫۶ بود. مقایسه نتایج توابع فعالساز *modified Elliott*, *cloglogm* و *log-sigmoid* بر روی مجموعه داده Movie Review در شکل (۴-۶) نمایش داده شده است، جایی که حداقل خطا و حداقل میانگین خطا هر سه تابع نمایش داده شده است.

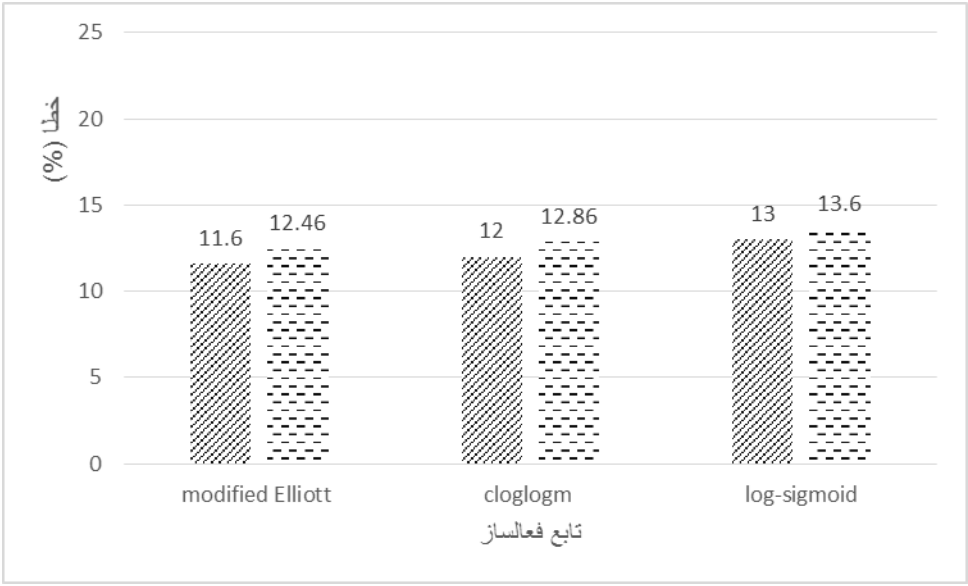


شکل (۴-۶): مقایسه نتایج حداقل خطا و حداقل میانگین خطای توابع فعالساز modified Elliott, cloglogm و log-sigmoid بر روی مجموعه داده Movie Review.

نتایج مجموعه داده IMDB در جدول (۴-۴) قابل مشاهده است، جایی که حداقل و حداکثر خطا برای هر تابع فعالساز گزارش شده است. همچنین تعداد واحد در لایه مخفی بر روی مجموعه {۴, ۸, ۱۶, ۳۲, ۶۴, ۱۲۸, ۲۵۶} گرفته شد. تعداد تکرار در این مجموعه داده ۵۰ تکرار برای هر آزمایش در نظر گرفته شد. بر روی این مجموعه داده کمترین خطا در هر مرحله برای تابع فعالساز *modified Elliott* با میزان خطای ٪۱۱/۶ بود و کمترین خطای میانگین نیز برای همین تابع فعالساز با میزان خطای ٪۱۲/۴۶ بود. تعداد واحد مخفی برای کمترین میزان خطا ۱۲۸ و ۲۵۶ واحد و برای کمترین میزان میانگین خطا ۲۵۶ واحد بودند. بعد از تابع فعالساز *modified Elliott* با بازه [۱/۵, ۰/۵-]، توابع فعالساز *cloglogm* ([۱/۵, ۰/۵-]) و *softsign* ([۱/۵, ۰/۵-]) و *saturated* ([۱/۵, ۰/۵-]) و *Bi-sig2* ([۰،۱]) دارای کمترین خطا با میزان به ترتیب ٪۱۱/۶, ٪۱۲, ٪۱۲, ٪۱۲ و ٪۱۲/۴ بودند.

با سه بار اجرای مجزا روی هر تابع فعالساز، نتایج خطای آزمایشی در اجرای اول برای مجموعه داده IMDB نشان داد که در این دور تابع *modified Elliott* دارای کمترین میزان خطا با ٪۱۱/۶ بود. در

اجرای دوم تابع *modified Elliott* دارای کمترین میزان خطا با ٪۱۲ بود و در اجرای سوم تابع *modified Elliott* دارای کمترین میزان خطا با ٪۱۱/۶ بود. نکته جالب توجه این بود که تابع فعالساز سیگموئید (*log-sigmoid*) در بهترین پنج نتایج اولیه نبود و رتبه آن بین توابع فعالساز مورد استفاده ۱۲ از ۲۳, با حداقل درصد خطای ٪۱۳ بود. مقایسه نتایج توابع فعالساز *modified Elliott*, *cloglogm* و log-*sigmoid* بر روی مجموعه داده IMDB در شکل (۴-۷) نمایش داده شده است، جایی که حداقل خطا و حداقل میانگین خطا هر سه تابع نمایش داده شده است.



شکل (۴-۷): مقایسه نتایج حداقل خطا و حداقل میانگین خطای توابع فعالساز modified Elliott, cloglogmو log-sigmoid بر روی مجموعه داده IMDB.

جدول (۴-۳): نتایج حداقل و حداکثر خطای آزمایشی برای هر تابع فعالساز برای مجموعه داده Movie Review. هر اجرا بر روی تعداد بلوک مشخص در لایه مخفی شبکه حافظه کوتاه و بلندمدت گرفته شده است.

تعداد بلوک در لایه مخفی / تابع فعالساز	۲	۴	۸	۱۶	۳۲	۶۴
Aranda	۲۳,۴-۲۳,۸	۲۳,۰۴-۲۴,۰۸	۲۳,۰۴-۲۴,۱۲	۲۲,۸۸-۲۳,۴۸	۲۳,۶۴-۲۴,۱۲	۲۳,۳۲-۲۴,۰۸
Bi-sig1	۲۳,۲۴-۲۴,۶	۲۳,۲-۲۳,۹۶	۲۳,۲۴-۲۳,۴۸	۲۳,۰۸-۲۳,۸۸	۲۳,۷۲-۲۴	۲۳,۶۸-۲۴
Bi-sig2	۲۲,۷۶-۲۳,۴	۲۲,۵۶-۲۴	۲۳,۰۴-۲۳,۳۶	۲۳,۴۴-۲۴,۰۸	۲۳,۳۶-۲۴,۴۸	۲۳,۴۴-۲۴,۴۴
Bi-tanh1	۲۲,۳۲-۲۳,۲	۲۲,۶-۲۳,۲۸	۲۲,۴۸-۲۳,۳۶	۲۲,۸-۲۳,۲۸	۲۲,۹۶-۲۳,۸	۲۳,۰۸-۲۳,۴
Bi-tanh2	۲۲,۸۸-۲۳,۶۴	۲۲,۶-۲۳,۳۲	۲۲,۶۸-۲۳,۲۸	۲۳-۲۳,۲۴	۲۲,۸۴-۲۳,۱۲	۲۳,۱۶-۲۳,۸
cloglog	۲۳,۱۲-۲۳,۸	۲۲,۷۶-۲۳,۳۲	۲۲,۷۲-۲۳,۲	۲۲,۸۴-۲۳,۲۸	۲۳,۰۴-۲۳,۵۶	۲۳,۲۸-۲۳,۸۴
cloglogm	۲۲,۸۴-۲۳,۰۸	**۲۲,۰۸-۲۳,۲۸**	۲۲,۷۲-۲۳,۲	۲۲,۸۴-۲۳,۱۲	۲۲,۷۲-۲۳,۲۴	۲۲,۵۶-۲۳,۲۸
Elliott	۲۳,۰۸-۲۴,۴۸	۲۳,۰۴-۲۵,۴	۲۲,۸-۲۳,۷۲	۲۳,۰۸-۲۳,۷۲	۲۴,۰۸-۲۴,۳۶	۲۳,۸۸-۲۴,۱۶
Gaussian	۲۲,۱۸-۲۳,۴۸	۲۳,۵۲-۲۳,۷۲	۲۳,۶-۲۴,۷۲	۲۳,۴۴-۲۳,۷۲	۲۳,۶۸-۲۴,۱۲	۲۳,۷۲-۲۴,۵۶
logarithmic	۲۳,۰۴-۲۴,۱۶	۲۲,۶-۲۳,۳۶	۲۲,۷۶-۲۳,۰۴	۲۲,۷۶-۲۳,۳۶	۲۲,۸۴-۲۲,۹۲	۲۲,۸-۲۳,۱۲
loglog	۲۲,۵۲-۲۳,۴۸	۲۲,۹۶-۲۳,۴	۲۳,۳۶-۲۳,۵۶	۲۳-۲۳,۲	۲۳,۲۴-۲۳,۵۲	۲۳,۸-۲۴
logsigm	۲۳-۲۳,۴۴	۲۳,۰۸-۲۳,۶۸	۲۲,۹۶-۲۳,۸	۲۲,۷۶-۲۳,۱۶	۲۳,۳۲-۲۳,۸	۲۳,۷۲-۲۴,۳۲
log-sigmoid	۲۲,۶-۲۲,۹۶	۲۲,۹۲-۲۴,۷۶	۲۳,۰۴-۲۴,۱۲	۲۳,۴۸-۲۴,۰۴	۲۳,۴-۲۳,۶۸	۲۳,۶-۲۳,۷۲
modified Elliott	۲۲,۱۲-۲۲,۸	۲۲,۸۴-۲۳,۶	۲۲,۲-۲۳,۱۲	**۲۱,۸۸-۲۳,۰۴**	۲۲,۸۸-۲۳,۲	۲۳,۳۲-۲۳,۸
rootsig	۲۲,۲۸-۲۳,۳۲	۲۲,۹۲-۲۳,۴۸	۲۳,۳۶-۲۳,۵۶	۲۳,۰۸-۲۳,۱۶	۲۲,۸۴-۲۳,۶۸	۲۲,۸۴-۲۳,۸۸
saturated	۲۳,۳۲-۲۵,۵۲	۲۲,۵۲-۲۴,۸۴	۲۲,۴۸-۲۳,۴۴	۲۲,۷۶-۲۲,۸۴	۲۲,۸۴-۲۳,۴۴	۲۳,۰۴-۲۳,۷۲
sech	۲۳,۶-۲۴,۸۸	۲۳,۳۲-۲۳,۹۲	۲۳,۵۲-۲۴,۳۲	۲۳,۴۸-۲۴,۰۸	۲۳,۶۴-۲۴,۰۸	۲۳,۶۴-۲۵
sigmoidalm	۲۲,۹۲-۲۳,۴۴	۲۳,۱۶-۲۳,۷۲	۲۳,۲-۲۳,۴	۲۳,۳۲-۲۳,۹۲	۲۳,۴۴-۲۴,۰۴	۲۴-۲۴,۴۴
sigmoidalm2	۲۳,۰۴-۲۴,۱۶	۲۳,۳۲-۲۳,۶	۲۳,۶-۲۴,۰۴	۲۳,۳۲-۲۴,۳۶	۲۳,۹۲-۲۴,۵۶	۲۴,۲۸-۲۴,۶
sigt	۲۲,۶۸-۲۳,۴۴	۲۲,۸-۲۳,۷۶	۲۲,۹۶-۲۳,۴	۲۳,۱۶-۲۳,۴۸	۲۳,۴۸-۲۳,۸۴	۲۳,۵۶-۲۴,۰۴
skewed-sig	۲۲,۷۶-۲۳,۶۸	۲۲,۸-۲۳	۲۲,۵۲-۲۳,۸۸	۲۳,۰۴-۲۳,۱۲	۲۲,۱۶-۲۳,۶۸	۲۲,۷۲-۲۳,۷۲
softsign	۲۳,۲-۲۴,۳۶	۲۲,۴۸-۲۳,۷۶	۲۲,۵۲-۲۳,۴۴	۲۲,۶۸-۲۳,۰۸	۲۲,۷۲-۲۳,۳۶	۲۲,۷۶-۲۳,۶۴
wave	۲۲,۴۸-۲۵,۶۴	۲۲,۹۲-۲۳,۷۶	۲۳,۰۴-۲۳,۳۶	۲۲,۴۸-۲۳,۱۶	۲۳,۲۸-۲۴,۴	۲۳,۶-۲۴,۳۲

جدول (۴-۴): نتایج حداقل و حداکثر خطای آزمایشی برای هر تابع فعالساز برای مجموعه داده IMDB. هر اجرا بر روی تعداد بلوک مشخص در لایه مخفی شبکه حافظه کوتاه و بلندمدت گرفته شده است.

تعداد بلوک در لایه مخفی / تابع فعالساز	۴	۸	۱۶	۳۲	۶۴	۱۲۸	۲۵۶
Aranda	۱۳,۸-۱۷,۲	۱۳,۴-۱۳,۸	۱۳,۶-۱۴,۴	۱۳,۲-۱۴,۲	۱۳,۲-۱۴,۲	۱۴-۱۴,۲	۱۳,۲-۱۴,۸
Bi-sig1	۱۴,۸-۱۵,۲	۱۳,۸-۱۴,۸	۱۳,۸-۱۴,۴	۱۳-۱۳,۶	۱۳,۶-۱۵,۴	۱۳,۸-۱۴,۲	۱۳,۴-۱۴,۴
Bi-sig2	۱۴,۴-۱۴,۸	۱۲,۶-۱۴,۴	۱۳,۶-۱۴,۶	۱۳-۱۴,۲	۱۳,۴-۱۴	۱۲,۴-۱۳,۴	۱۲,۸-۱۴,۶
Bi-tanh1	۱۴,۸-۱۸,۶	۱۵-۱۶,۸	۱۴,۶-۱۵,۸	۱۳,۸-۱۵	۱۳,۶-۱۴,۶	۱۳,۲-۱۴,۶	۱۳,۶-۱۶,۴
Bi-tanh2	۱۴,۶-۱۷	۱۲,۸-۱۴,۲	۱۲,۸-۱۴,۴	۱۳-۱۳,۴	۱۲,۶-۱۳,۴	۱۳,۶-۱۴,۶	۱۳,۲-۱۴,۲
cloglog	۱۴,۶-۱۶,۲	۱۳,۴-۱۴,۸	۱۳,۲-۱۵	۱۳-۱۴,۴	۱۳,۲-۱۳,۸	۱۳-۱۳,۸	۱۳-۱۴,۲
cloglogm	۱۳,۲-۱۴,۴	۱۳,۸-۱۶,۸	۱۳-۱۳,۶	۱۳,۲-۱۳,۸	**۱۲**-۱۳,۶	۱۲,۸-۱۳,۴	۱۲,۲-۱۴
Elliott	۱۳,۴-۱۵,۴	۱۳-۱۴,۶	۱۳,۸-۱۴,۶	۱۳,۴-۱۴,۶	۱۳,۶-۱۵,۲	۱۴,۴-۱۴,۸	۱۲,۶-۱۵,۲
Gaussian	۱۴,۲-۲۴,۴	۱۳,۴-۱۶,۸	۱۳,۸-۱۸,۲	۱۴,۶-۱۴,۸	۱۴,۶-۱۷,۶	۲۲-۲۷,۸	۱۶,۸-۲۴,۶
logarithmic	۱۴,۴-۱۵,۸	۱۴,۴-۱۵,۶	۱۳,۴-۱۵,۴	۱۳,۲-۱۴	۱۳,۶-۱۳,۶	۱۲,۸-۱۳,۴	۱۲,۶-۱۴,۴
loglog	۱۸,۶-۲۱,۴	۱۶-۲۵	۱۵,۲-۲۲,۲	۱۵-۲۰,۲	۱۴,۸-۲۰,۴	۲۹,۶-۳۰,۶	۳۴,۶-۳۷,۶
logsigm	۱۶,۸-۱۸,۴	۱۵,۲-۱۹,۶	۱۳,۸-۱۶,۸	۱۴,۲-۱۵,۸	۱۴-۱۴,۸	۱۵,۴-۲۰,۸	۱۹,۲-۳۷
log-sigmoid	۱۴,۴-۱۵	۱۳,۸-۱۴,۴	۱۳,۶-۱۴,۶	۱۳,۴-۱۵,۲	۱۳,۴-۱۴	۱۳-۱۴,۲	۱۳,۴-۱۴
modified Elliott	۱۴,۶-۲۰	۱۴-۱۵,۶	۱۲-۱۴,۸	۱۳,۸-۱۵,۴	۱۲,۶-۱۴,۴	**۱۱,۶**-۱۳,۸	**۱۱,۶**-۱۳,۴
rootsig	۱۴,۴-۱۷,۶	۱۳,۲-۱۵	۱۲,۸-۱۴	۱۳,۲-۱۳,۸	۱۳,۸-۱۴	۱۲,۶-۱۴,۲	۱۲,۸-۱۳,۸
saturated	۱۵,۶-۱۸,۴	۱۳,۶-۱۴,۶	۱۴-۱۶	۱۳,۴-۱۷,۲	۱۲,۴-۱۴,۶	۱۲-۱۳,۸	۱۲,۴-۱۳,۲
sech	۱۷-۱۹	۱۵,۴-۲۰,۲	۱۳,۸-۱۵,۶	۱۴,۴-۱۴,۸	۱۴,۶-۱۶,۲	۱۶,۸-۲۱	۳۰,۸-۳۴,۲
sigmoidalm	۱۳-۱۴,۸	۱۴-۱۴,۶	۱۳-۱۴,۸	۱۳,۴-۱۴,۶	۱۳,۶-۱۴	۱۳-۱۴,۲	۱۴-۱۴,۶
sigmoidalm2	۱۳,۸-۱۴,۶	۱۳,۸-۱۴,۶	۱۳,۲-۱۴,۸	۱۳,۴-۱۴	۱۳,۸-۱۴,۴	۱۳,۶-۱۴,۲	۱۴,۲-۱۵
sigt	۱۶,۲-۱۸,۸	۱۴-۱۴,۴	۱۴,۲-۱۶,۲	۱۳,۴-۱۵,۸	۱۳,۴-۱۵,۲	۱۴,۲-۱۵,۴	۱۴,۶-۱۶,۸
skewed-sig	۱۴-۱۸,۴	۱۶,۸-۱۹,۴	۱۵,۲-۱۸,۶	۱۴,۶-۱۷	۱۴,۴-۱۵,۶	۱۵,۲-۱۷,۲	۱۷-۳۸
softsign	۱۳,۴-۱۵	۱۴-۱۵	۱۳,۶-۱۴,۴	۱۲,۸-۱۳,۲	۱۲,۸-۱۴,۸	۱۲,۸-۱۳,۶	۱۲-۱۳,۸
wave	۱۶,۲-۲۱,۸	۱۵-۲۳,۲	۱۷,۴-۱۹	۱۵,۲-۱۷	۱۸-۲۰,۴	۳۴-۴۲,۶	۲۱-۴۳,۸

مقدار متوسط خطا برای همه اجراها در داده‌های آموزشی هنگام گزارش کمترین میزان خطای آزمایشی در جداول (۴-۵) و (۴-۶) برای به ترتیب مجموعه داده Movie Review و IMDB گزارش می‌شود.

جدول (۴-۵): مقدار متوسط خطا در داده‌های آموزشی هنگام گزارش کمترین میزان خطای آزمایشی برای مجموعه داده Movie Review.

تعداد بلوک در لایه مخفی / تابع فعالساز	۲	۴	۸	۱۶	۳۲	۶۴
Aranda	۱۱/۸	۸/۴۳	۷/۱۶	۴/۰۶	۶/۰۳	۳/۶
Bi-sig1	۱۱/۵۳	۹/۱	۶/۶۶	۵/۵۳	۴/۸	۵/۳
Bi-sig2	۹/۰۳	۸	۷/۹۳	۶/۹۶	۸/۱۶	۵/۴۳
Bi-tanh1	۷/۴	۵/۹۶	۶/۸۶	۴/۷۳	۶/۲	۷/۷۶
Bi-tanh2	۹/۸۶	۸/۳۳	۷/۵۳	۷/۲۶	۸/۱	۳/۸۳
cloglog	۸/۲۶	۷/۴۶	۶/۱۳	۵/۴	۴/۵۶	۸/۷۳
cloglogm	۹/۷	۸/۲	۷/۸۳	۷/۰۶	۹/۱	۵/۰۳
Elliott	۹/۳۶	۱۰/۵	۶/۷	۵/۸۶	۷/۵۶	۷/۴۶
Gaussian	۶/۸۳	۶/۰۳	۸/۲۶	۴/۶۶	۸/۲	۳/۸
logarithmic	۱۱/۲۳	۹/۴	۷/۷	۵/۹۳	۷/۱۳	۸/۵۳
loglog	۸/۵۶	۶/۶	۶/۷۶	۶/۴۳	۵	۲/۹۶
logsigm	۷/۸۶	۷/۶۳	۷/۰۳	۴/۵۳	۵/۳	۵/۲۶
log-sigmoid	۱۰/۷	۱۰/۰۶	۶/۵۶	۴/۷۳	۳/۹۳	۴/۴۶
modified Elliott	۹/۴۶	۹/۰۶	۶/۵	۷/۱	۵/۶۶	۴/۵۶
rootsig	۱۱/۵۳	۹/۵	۶/۵۳	۶/۴۳	۶/۱۶	۴/۹۶
saturated	۱۰/۲۶	۱۰/۴۳	۷/۲۳	۷/۸۳	۴/۵۶	۷/۵
sech	۱۰/۵۳	۷/۴۳	۸/۱	۵/۰۳	۵/۱	۵/۸۳
sigmoidalm	۹/۴۶	۹	۶/۵۳	۷/۱۳	۸/۱	۸/۱
sigmoidalm2	۹/۰۶	۷/۵۳	۷/۲	۸/۶۶	۵/۷۶۶	۵/۶۳
sigt	۸/۳	۷/۲۳	۶/۰۶	۵/۸۶	۴/۸۶	۵/۰۶
skewed-sig	۸/۸۳	۸/۱	۷/۶۳	۶/۸	۶/۸۶	۷/۰۶
softsign	۱۱/۸۶	۸/۲۶	۶	۴/۹۳	۵	۶/۶۶
wave	۱۰/۹۳	۷/۶	۵/۵۶	۷/۶۶	۷/۲۳	۴/۵۳

جدول (۴-۶): مقدار متوسط خطا در داده‌های آموزشی هنگام گزارش کمترین میزان خطای آزمایشی برای مجموعه داده IMDB.

تعداد بلوک در لایه مخفی / تابع فعالساز	۴	۸	۱۶	۳۲	۶۴	۱۲۸	۲۵۶
Aranda	۳/۴۶	۱/۳	۲/۲۳	۱	۱/۰۳	۱/۱	۳/۱۶
Bi-sig1	۱/۹	۱/۴	۱/۲	۱/۳	۱/۳	۱/۲۳	۱/۱۶
Bi-sig2	۱/۷۳	۲/۴	۱/۷۶	۱/۴۳	۱/۴۶	۱/۶۳	۱/۴
Bi-tanh1	۲/۱	۱/۱۶	۱/۹۳	۱/۱۳	۱/۹۶	۱/۰۳	۴/۰۳
Bi-tanh2	۴/۹	۱/۸۶	۱/۲۳	۱/۶۶	۱/۸۳	۱	۱/۲
cloglog	۲/۱۳	۲/۷	۱/۵۳	۱/۳۶	۲/۶۶	۱/۵	۱/۲۳
cloglogm	۴/۳۳	۲/۵	۲/۴	۲/۶۳	۱/۴	۱/۱۶	۱/۳۳
Elliott	۱/۵۳	۱/۵	۰/۹۳	۱/۲	۱/۰۶	۱/۲۶	۴/۱۶
Gaussian	۳/۱۳	۱/۶	۲/۶۶	۲/۳۶	۵/۲۳	۱۸/۴۶	۱۱/۷
logarithmic	۱/۵۶	۳/۰۶	۱/۵	۱/۱۳	۱	۱/۰۶	۲/۷۳
loglog	۴/۷۳	۲/۴	۲/۵	۳/۳	۳/۲	۲۲/۳۳	۲۴/۳
logsigm	۱/۸	۳/۸۶	۱/۶۳	۱/۲۶	۱/۱۶	۳/۱۶	۱۱/۳
log-sigmoid	۱/۳	۱/۵	۱/۴	۲/۴۶	۲/۵۶	۱/۴۳	۱/۲۳
modified Elliott	۲/۳	۳/۴۳	۱/۶۶	۵/۱	۱/۶۶	۱/۰۶	۱/۴۳
rootsig	۱/۷۶	۱/۸۳	۱/۲۳	۱/۶۳	۱/۰۶	۱/۳۶	۱/۰۶
saturated	۳/۸۶	۱/۹	۱/۶۳	۱/۸۳	۱/۶	۱/۲۳	۱/۱۶
sech	۱/۱	۲/۱	۲/۲۶	۱/۲۳	۳/۱	۶/۳۶	۲۲/۰۶
sigmoidalm	۱/۳۶	۱/۶۶	۲/۶۳	۲/۳	۳/۱۶	۱/۰۳	۲/۵۳
sigmoidalm2	۱/۶	۲/۷۶	۲/۱۳	۱/۱	۱/۰۶	۱/۹۶	۱/۳۶
sigt	۱/۷۳	۱/۵۳	۱/۰۳	۴/۱۶	۱/۲۳	۲/۱۳	۳/۱۳
skewed-sig	۳/۳	۲/۳۶	۲/۴۳	۱/۶۶	۱/۷۳	۳/۲۳	۱۷/۷
softsign	۲/۵	۱/۲۶	۲/۴	۱/۸	۱/۳	۱/۲۳	۳/۵
wave	۷	۸/۷۳	۱۰/۳۶	۴/۶	۱۲/۸	۳۵/۲	۳۱/۰۶

تعداد متوسط دور تا همگرایی برای همه اجراها در جدول های (۴-۷) و (۴-۸) برای به ترتیب مجموعه داده Movie Review و IMDB نشان داده می شود.

جدول (۴-۷): متوسط تعداد دور تا همگرایی برای مجموعه داده Movie Review.

تعداد بلوک در لایه مخفی / تابع فعالساز	۲	۴	۸	۱۶	۳۲	۶۴
Aranda	۱۸	۱۹/۳۳	۱۸	۱۷/۶۶	۱۲/۳۳	۱۴
Bi-sig1	۲۰	۱۶/۶۶	۱۷/۳۳	۱۶/۶۶	۱۳/۶۶	۱۱/۳۳
Bi-sig2	۱۷	۱۳/۶۶	۹/۶۶	۹/۶۶	۸/۳۳	۱۰
Bi-tanh1	۱۷/۳۳	۱۴/۶۶	۹/۳۳	۱۰	۸	۷
Bi-tanh2	۱۹	۱۹	۱۶	۱۲/۳۳	۱۰/۳۳	۱۲
cloglog	۱۴/۶۶	۱۴/۶۶	۱۱/۶۶	۱۰/۳۳	۱۰/۳۳	۶/۳۳
cloglogm	۱۸	۱۸/۶۶	۱۶/۶۶	۱۲/۳۳	۸/۶۶	۱۰
Elliott	۱۹/۳۳	۱۸/۳۳	۱۸/۳۳	۱۵/۳۳	۱۲	۱۱
Gaussian	۱۹/۶۶	۱۷/۳۳	۹/۶۶	۱۳/۶۶	۱۰	۱۵
logarithmic	۱۸/۳۳	۱۳/۶۶	۱۶	۱۴/۳۳	۱۱	۸/۶۶
loglog	۱۸	۱۵/۶۶	۱۰/۶۶	۱۰/۶۶	۱۱/۶۶	۱۴
logsigm	۱۷/۳۳	۱۱	۹/۳۳	۱۱/۶۶	۱۰/۳۳	۹/۳۳
log-sigmoid	۱۷/۳۳	۱۸	۱۷/۳۳	۱۶/۳۳	۱۳/۳۳	۱۲
modified Elliott	۱۹	۱۸/۳۳	۱۴/۶۶	۱۲/۳۳	۱۱/۶۶	۱۰/۳۳
rootsig	۱۶/۶۶	۱۵	۱۶	۱۲/۶۶	۱۲/۶۶	۱۱/۳۳
saturated	۱۹	۱۹/۶۶	۱۷/۶۶	۱۳	۱۲	۹/۳۳
sech	۱۶/۳۳	۱۶/۳۳	۱۱/۶۶	۱۴/۶۶	۱۴/۳۳	۱۲
sigmoidalm	۱۷	۱۴	۱۴/۳۳	۱۰/۶۶	۹	۸
sigmoidalm2	۱۷	۱۳/۶۶	۱۴/۶۶	۱۰	۱۲/۶۶	۱۱/۳۳
sigt	۱۵/۳۳	۱۱/۳۳	۱۲/۳۳	۱۰/۳۳	۹/۶۶	۹/۳۳
skewed-sig	۱۴/۳۳	۱۰/۶۶	۱۰/۳۳	۸/۶۶	۸/۳۳	۷/۶۶
softsign	۱۹/۶۶	۱۸/۶۶	۱۵/۶۶	۱۵/۳۳	۱۰/۳۳	۱۰/۳۳
wave	۱۸	۱۳/۳۳	۱۵/۳۳	۱۲	۱۱/۳۳	۱۷/۶۶

جدول (۴-۸): متوسط تعداد دور تا همگرایی برای مجموعه داده IMDB.

تعداد بلوک در لایه مخفی / تابع فعالساز	۴	۸	۱۶	۳۲	۶۴	۱۲۸	۲۵۶
Aranda	۴۶	۳۵	۲۵	۳۲	۳۰	۲۸	۲۳
Bi-sig1	۳۸	۳۱	۳۰	۳۹	۲۸	۲۹	۲۸
Bi-sig2	۲۳	۱۶	۲۳	۲۷	۲۹	۲۰	۲۳
Bi-tanh1	۲۸	۲۵	۳۰	۳۶	۳۳	۴۰	۳۵
Bi-tanh2	۳۸	۲۷	۳۸	۲۴	۲۳	۳۵	۲۹
cloglog	۲۴	۱۹	۱۶	۲۶	۱۸	۳۵/۶۶	۲۶
cloglogm	۲۹	۳۵	۲۵	۲۳	۲۱	۲۳	۲۷
Elliott	۳۷	۳۳	۳۶	۳۴	۲۷	۲۷	۲۵
Gaussian	۴۰	۴۰	۴۴	۴۲	۴۵	۴۷	۴۵
logarithmic	۳۹	۲۹	۲۹	۲۵	۳۴	۲۳/۶۶	۲۲
loglog	۴۱	۳۹	۴۴	۴۵	۴۸	۴۶	۴۴
logsigm	۳۰	۲۷	۳۵	۳۸	۴۳	۴۶	۴۷
log-sigmoid	۳۴	۲۵	۲۸	۲۴	۲۲	۲۳	۲۵
modified Elliott	۴۳	۳۴	۳۱	۲۵	۲۳	۲۶	۲۸
rootsig	۳۶	۳۲	۲۸	۲۷	۲۶	۲۰/۶۶	۲۳
saturated	۴۷	۳۸	۴۷	۳۰	۲۲	۳۲	۲۹
sech	۴۷	۴۸	۴۵	۴۷	۴۷	۴۸	۴۷
sigmoidalm	۲۷	۲۲	۲۲	۳۹	۳۲	۳۱	۱۹/۶۶
sigmoidalm2	۲۶	۲۳	۱۷	۲۸	۲۹	۲۱	۲۴
sigt	۲۹	۲۶	۲۸	۳۵	۳۸	۴۴	۴۶
skewed-sig	۲۹	۳۲	۳۹	۴۳	۳۸	۴۶	۴۷
softsign	۴۰	۳۸	۲۷	۲۲	۲۴	۲۲	۲۰
wave	۳۶	۴۴	۴۶	۳۹	۴۷	۴۷	۴۵

۴-۳- نتیجه‌گیری

در این پایان‌نامه به بررسی ۲۳ توابع فعالساز مختلف در دروازه سیگموئیدی شبکه حافظه کوتاه و بلندمدت برای کلاسه‌بندی پرداخته شد. در بلوک‌های شبکه حافظه کوتاه و بلندمدت دو تابع فعالساز سیگموئید و تانژانت هایپربولیک دارای بیشترین استفاده هستند. دروازه سیگموئیدی به‌صورت یکنواخت تغییر داده شد و آزمایش‌ها روی دو مجموعه داده IMDB و Movie Review به‌صورت ۳ بار جداگانه گرفته شد و نتایج زیر به دست آمد.

۱. تابع فعالساز *modified Elliott* دارای بهترین نتایج روی هر دو مجموعه داده با کمترین خطا و همچنین کمترین میانگین خطا بود.

۲. دامنه [۱/۵, ۰/۵-] دارای بهترین نتایج روی هر دو مجموعه داده بود و این نشان می‌دهد دامنه بزرگ‌تر غیر از [۰،۱] می‌تواند نتایج بهتری تولید کند.

۳. تابع فعالساز سیگموئید نسبت به اغلب توابع فعالساز استفاده شده دارای نتایج بدتری بود و می‌توان با توابع فعالساز دیگر در شرایط برابر نتایج بهتری به دست آورد.

۴. تابع فعالساز *cloglogm* بعد از تابع فعالساز *modified Elliott* با توجه به نتایج روی هر دو مجموعه داده، دارای نتایج بهتری نسبت به بقیه توابع استفاده شده است.

۵. درنهایت پیشنهاد می‌شود برای توابع فعالساز در دروازه‌های سیگموئیدی شبکه حافظه کوتاه و بلندمدت از توابع *modified Elliot* یا *cloglogm* استفاده گردد.

در آینده می‌توان به بررسی توابع فعالساز ورودی و خروجی بلوک (تانژانت هایپربولیک) در بلوک‌های شبکه حافظه کوتاه و بلندمدت پرداخت و نتایج را با تابع فعالساز تانژانت هایپربولیک که در اغلب موارد برای این شبکه و شبکه‌های دیگر به کار می‌رود مقایسه کرد. عملکرد توابع فعالساز پیشنهادی می تواند در شبکه های مشتق شده از شبکه حافظه کوتاه و بلندمدت و معماری های گوناگون آن ارزیابی شود. همچنین تحقیقات بیشتری در این زمینه برای وظایف گوناگون با مجموعه داده‌های دیگر نیاز است.

منابع و مراجع

[1] Malsburg, C. V. D. (1986). Frank Rosenblatt: Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. In D. G. Palm & D. A. Aertsen (Eds.), *Brain Theory*, Springer Berlin Heidelberg, 245–248.

[2] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. doi:10.1037/h0042519

[3] Broomhead, D., & Lowe, D. (1988). Multivariable Functional Interpolation and Adaptive Networks. *Complex Systems*, *2*, 321–355.

[4] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, *79*(8), 2554–2558.

[5] Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. PhD thesis. New York, NY, USA: Oxford University Press, Inc.

[6] De Mulder, W., Bethard, S., & Moens, M.-F. (2015). A survey on the application of recurrent neural networks to statistical language modeling. *Computer Speech & Language*, *30*(1), 61–98. doi:10.1016/j.csl.2014.09.005

[7] Graves, A. (2012). *Supervised sequence labelling with recurrent neural networks*, Springer, 385, 1-56.

[8] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Comput.*, *9*(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735

[9] Graves, A., Jaitly, N., & Mohamed, A. (2013). *Hybrid speech recognition with deep Bidirectional LSTM*, Automatic Speech Recognition and Understanding (ASRU), IEEE Workshop on, 273 – 278.

[10] Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. (2009). A Novel Connectionist System for Unconstrained Handwriting Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(5), 855–868. doi:10.1109/TPAMI.2008.137

[11] Graves, A., & Schmidhuber, J. (2005). *Framewise phoneme classification with bidirectional LSTM and other neural network architectures*, Neural Networks, 602-610.

[12] Otte, C., Otte, S., Wittig, L., Hüttmann, G., Kugler, C., Drömann, D., Zell, A., & Schlaefer, A. (2014). *Investigating recurrent neural networks for OCT A-scan BasedTissue analysis*, Methods Inf Med 5, 245–249.

[13] Otte, S., Otte, C., Schlaefer, A., Wittig, L., Huttmann, G., Dromann, D., & Zell, A. (2013). *OCT A-Scan based lung tumor tissue classification with Bidirectional Long Short Term Memory networks*, Machine Learning for Signal Processing (MLSP), IEEE International Workshop on, 1-6.

[14] Yao, K., Cohn, T., Vylomova, K., Duh, K., & Dyer, C. (2015). *Depth-gated LSTM*, arXiv:1508.03790v2 [cs.NE].

[15] Cho, K., Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning phrase representations using RNN Encoder-Decoder for statistical machine translation*, arXiv:1406.1078v3 [cs.CL].

[16] Kalchbrenner, N., Danihelka, I., & Graves, A. (2015). *Grid long short-term memory*, arXiv:1507.01526v1 [cs.NE].

[17] Le, Q., Jaitly, N., & Hinton, G. (2015). *A simple way to initialize recurrent networks of rectified linear units*, arXiv:1504.00941v2 [cs.NE].

[18] Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015). *An empirical exploration of recurrent network architectures*, Proceedings of the 32th International Conference on Machine Learning, Lille, France, JMLR: W&CP, 37.

[19] Greff, K., Srivastava, R., Koutnık, J., Steunebrink, B., & Schmidhuber, J. (2015). *LSTM: A search space odyssey*, arXiv:1503.04069v1 [cs.NE].

[20] Lipton, Z. C., Kale, D. C., Elkan, C., & Wetzell, R. (2015). Learning to Diagnose with LSTM Recurrent Neural Networks. *arXiv:1511.03677 [cs]*.

[21] Liwicki, M., Graves, A., Bunke, H., & Schmidhuber, J. (2007). A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks. In *In Proceedings of the 9th International Conference on Document Analysis and Recognition, ICDAR 2007*.

[22] Bluche, T., Louradour, J., & Messina, R. (2016). Scan, Attend and Read: End-to-End Handwritten Paragraph Recognition with MDLSTM Attention. *arXiv:1604.03286 [cs]*.

[23] Pham, V., Bluche, T., Kermorvant, C., & Louradour, J. (2013). Dropout improves Recurrent Neural Networks for Handwriting Recognition. *arXiv:1312.4569 [cs]*.

[24] Doetsch, P., Kozielski, M., & Ney, H. (2014). Fast and Robust Training of Recurrent Neural Networks for Offline Handwriting Recognition, IEEE, 279–284. doi:10.1109/ICFHR.2014.54

[25] Graves, A., Fernández, S., & Schmidhuber, J. (2005). Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition. In W. Duch, J. Kacprzyk, E. Oja, &

S. Zadrożny (Eds.), *Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005,* Springer Berlin Heidelberg, 799–804.

[26] Otte, S., Krechel, D., Liwicki, M., & Dengel, A. (2012). Local Feature Based Online Mode Detection with Recurrent Neural Networks. In *2012 International Conference on Frontiers in Handwriting Recognition (ICFHR),* 533–537. doi:10.1109/ICFHR.2012.229

[27] Graves, A., Mohamed, A., & Hinton, G. (2013). Speech Recognition with Deep Recurrent Neural Networks. *arXiv:1303.5778 [cs]*.

[28] Thang Luong, I. S. (2014). Addressing the Rare Word Problem in Neural Machine Translation. doi:10.3115/v1/P15-1002

[29] Wöllmer, M., Metallinou, A., Eyben, F., Schuller, B., & Narayanan, S. (2010). Context-sensitive multimodal emotion recognition from speech and facial expression using bidirectional LSTM modeling. In *in Proc. of Interspeech, Makuhari*, 2362–2365.

[30] Sak, H., Senior, A., & Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *In Proceedings of the Annual Conference of International Speech Communication Association (INTERSPEECH)*, 338–342.

[31] Fan, Y., Qian, Y., Xie, F., & Soong, F. K. (2014). TTS synthesis with bidirectional LSTM based Recurrent Neural Networks. *In Proc. Interspeech*, 1964–1968.

[32] Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent Neural Network Regularization. *arXiv:1409.2329 [cs]*.

[33] Sønderby, S. K., & Winther, O. (2014). Protein Secondary Structure Prediction with Long Short Term Memory Networks. *arXiv:1412.7828 [cs, Q-Bio]*.

[34] Marchi, E., Ferroni, G., Eyben, F., Gabrielli, L., Squartini, S., & Schuller, B. (2014). Multi-resolution linear prediction based features for audio onset detection with bidirectional LSTM neural networks. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2164–2168. doi:10.1109/ICASSP.2014.6853982

[35] Donahue, J., Hendricks, L. A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2014). Long-term Recurrent Convolutional Networks for Visual Recognition and Description. *arXiv:1411.4389 [cs]*.

[36] Wollmer, M., Blaschke, C., Schindl, T., Schuller, B., Farber, B., Mayer, S., & Trefflich, B. (2011). Online Driver Distraction Detection Using Long Short-Term Memory. *IEEE Transactions on Intelligent Transportation Systems*, *12*(2), 574–582. doi:10.1109/TITS.2011.2119483

[37] Gomes, G. S. da S., & Ludermir, T. B. (2013). Optimization of the weights and asymmetric activation function family of neural network for time series forecasting. *Expert Systems with Applications*, *40*(16), 6438–6446. doi:10.1016/j.eswa.2013.05.053

[38] Duch, W., & Jankowski, N. (1999). Survey of Neural Transfer Functions. *Neural Computing Surveys*, *2*, 163–213.

[39] Singh Sodhi, S., & Chandra, P. (2003). A class +1 sigmoidal activation functions for FFANNs. *Journal of Economic Dynamics and Control*, *28*(1), 183–187.

[40] Chandra, P., & Singh, Y. (2004). An activation function adapting training

algorithm for sigmoidal feedforward networks. *Neurocomputing*, *61*, 429–437. doi:10.1016/j.neucom.2004.04.001

[41] Gomes, G. S. da S., Ludermir, T. B., & Lima, L. M. M. R. (2010). Comparison of new activation functions in neural network for forecasting financial time series. *Neural Computing and Applications*, *20*(3), 417–439. doi:10.1007/s00521-010-0407-3

[42] Duch, W., & Jankowski, N. (2001). Transfer Functions: Hidden Possibilities for Better Neural Networks. In *9th European Symposium on Artificial Neural Networks (ESANN), Brugge 2001. De-facto publications*, 81–94.

[43] Gers, F.A., & Schmidhuber, J. (2000). *Recurrent nets that time and count*, In Proc. IJCNN'2000,Int. Joint Conf. on Neural Networks, Como, Italy.

[44] Sutskever, I,. Martens, J,.& Hinton, G. (2011). *Generating text with recurrent neural networks*, In Proceedings of the 28th International Conference on Machine Learning (ICML-11), 1017-1024.

[45] Williams, R. J., & Zipser, D. (1995). Gradient-based learning algorithms for recurrent networks and their computational complexity. In Y. Chauvin & D. E. Rumelhart (Eds.), *Back-propagation: Theory, Architectures and Applications*, Hillsdale, NJ, USA: L. Erlbaum Associates Inc, 433–486.

[46] Zeiler, M. D. (2012). ADADELTA: An Adaptive Learning Rate Method. *arXiv:1212.5701 [cs]*.

[47] Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, *12*(Jul), 2121–2159.

[48] Karlik, B., & Vehbi, A. (2011). *Performance analysis of various activation functions in generalized MLP architectures of neural networks*,International Journal of Artificial Intelligence and Expert Systems (IJAE), 111-122.

[49] Sibi, P., Jones, S., & Siddarth, P. (2013). *Analysis of different activation functions using back propagation neural networks*, Journal of Theoretical and Applied Information Technology, 47 (3), 1264-1268.

[50] Glorot, X., Bordes, A., & Bengio, Y. (2011). *Deep sparse rectifier neural networks*, 14th International Con-ference on Artificial Intelligence and Statistics (AISTATS), 15, 315-323.

[51] Gomes, G. S. d S., & Ludermir, T. B. (2008). Complementary Log-Log and Probit: Activation Functions Implemented in Artificial Neural Networks. In *Eighth International Conference on Hybrid Intelligent Systems, 2008. HIS '08*, 939–942. doi:10.1109/HIS.2008.40

[52] Michal Rosen-Zvi, M. B. (1998). Learnability of periodic activation functions: General results. *Physical Review E*, *58*(3), 3606–3609. doi:10.1103/PhysRevE.58.3606

[53] Ma, L., & Khorasani, K. (2005). Constructive feedforward neural networks using Hermite polynomial activation functions. *IEEE Transactions on Neural Networks*, *16*(4), 821–833. doi:10.1109/TNN.2005.851786

[54] Hornik, K. (1993). Some new results on neural network approximation. *Neural Networks*, *6*(8), 1069–1072. doi:10.1016/S0893-6080(09)80018-X

[55] Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, *4*(2), 251–257. doi:10.1016/0893-6080(91)90009-T

[56] Leshno, M., Lin, V. Y., Pinkus, A., & Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, *6*(6), 861–867. doi:10.1016/S0893-6080(05)80131-5

[57] Pao, Y.-H. (1989). *Adaptive Pattern Recognition and Neural Networks*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

[58] Hartman, E., Keeler, J. D., & Kowalski, J. M. (1990). Layered Neural Networks with Gaussian Hidden Units As Universal Approximations. *Neural Comput.*, *2*(2), 210–215. doi:10.1162/neco.1990.2.2.210

[59] Efe, M. Ö. (2008). Novel Neuronal Activation Functions for Feedforward Neural Networks. *Neural Processing Letters*, *28*(2), 63–79. doi:10.1007/s11063-008-9082-0

[60] Skoundrianos, E. N., & Tzafestas, S. G. (2004). Modelling and FDI of Dynamic Discrete Time Systems Using a MLP with a New Sigmoidal Activation Function. *J. Intell. Robotics Syst.*, *41*(1), 19–36. doi:10.1023/B:JINT.0000049175.78893.2f

[61] Giraud, B. G., Lapedes, A., & Liu, L. C. (1995). *Lorentzian Neural Nets*.

[62] Carroll, S. M., & Dickinson, B. W. (1989). Construction of neural nets using the radon transform. In *, International Joint Conference on Neural Networks, 1989. IJCNN, 1,* 607–611. doi:10.1109/IJCNN.1989.118639

[63] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, *2*(4), 303–314. doi:10.1007/BF02551274

[64] Funahashi, K. (1989). On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Netw.*, *2*(3), 183–192. doi:10.1016/0893-6080(89)90003-8

[65] Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer Feedforward Networks Are Universal Approximators. *Neural Netw.*, *2*(5), 359–366. doi:10.1016/0893-6080(89)90020-8

[66] Jones, L. K. (1990). Constructive approximations for neural networks by sigmoidal functions. *Proceedings of the IEEE*, *78*(10), 1586–1589. doi:10.1109/5.58342

[67] Chandra, P., & Singh, Y. (2004). Feedforward sigmoidal networks - equicontinuity and fault-tolerance properties. *IEEE Transactions on Neural Networks*, *15*(6), 1350–1366. doi:10.1109/TNN.2004.831198

[68] F. J. Aranda-Ordaz, "On two families of transformations to additivity for binary response data," *Biometrika*, *68*, pp. 357–364, 1981.

[69] Singh Sodhi, S., & Chandra, P. (2014). Bi-modal derivative activation function for sigmoidal feedforward networks. *Neurocomputing*, *143*, 182–196. doi:10.1016/j.neucom.2014.06.007

[70] Elliott, D. L., & Elliott, D. L. (1993). *A better Activation Function for Artificial Neural Networks*. Technical Report ISR TR 93–8, University of Maryland.

[71] Orr, M. J. L. (1996). *Introduction to Radial Basis Function Networks*. Centre for Cognitive Science.

[72] Tan, T. G., Teo, J., & Anthony, P. (2014). A Comparative Investigation of Non-linear Activation Functions in Neural Controllers for Search-based Game AI Engineering. *Artif. Intell. Rev.*, *41*(1), 1–25. doi:10.1007/s10462-011-9294-y

[73] Han, J., & Moraga, C. (1995). The influence of the sigmoid function parameters on the speed of backpropagation learning. In J. Mira & F. Sandoval (Eds.), *From Natural to Artificial Neural Computation*, Springer Berlin Heidelberg, 195–201.

[74] Burhani, H., Feng, W., & Hu, G. (2015). Denoising AutoEncoder in Neural Networks with Modified Elliott Activation Function and Sparsity-Favoring Cost Function. In *2015 3rd International Conference on Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence (ACIT-CSI)*, 343–348. doi:10.1109/ACIT-CSI.2015.67

[75] Maca, P., & Pech, P. (2016). Forecasting SPEI and SPI Drought Indices Using the Integrated Artificial Neural Networks. *Computational Intelligence and Neuroscience*, *2016*, 3868519. doi:10.1155/2016/3868519

[76] Nie, X., & Cao, J. (2011). Multistability of Second-Order Competitive Neural Networks With Nondecreasing Saturated Activation Functions. *IEEE Transactions on Neural Networks*, *22*(11), 1694–1708. doi:10.1109/TNN.2011.2164934

[77] Kenue, S. K. (1991). Efficient activation functions for the back-propagation neural network. In *, IJCNN-91-Seattle International Joint Conference on Neural Networks, 1991, 2,* 946. doi:10.1109/IJCNN.1991.155549

[78] Chandra, P., & Singh, Y. (2004). A case for the self-adaptation of activation functions in FFANNs. *Neurocomputing*, *56*, 447–454. doi:10.1016/j.neucom.2003.08.005

[79] Yuan, M., Hu, H., Jiang, Y., & Hang, S. (2013). A New Camera Calibration Based on Neural Network with Tunable Activation Function in Intelligent Space. In *2013 Sixth*

International Symposium on Computational Intelligence and Design (ISCID), 1, 371–374. doi:10.1109/ISCID.2013.99

[80] Chandra, P., & Sodhi, S. S. (2014). A skewed derivative activation function for SFFANNs. In *Recent Advances and Innovations in Engineering (ICRAIE), 2014*, 1–6. doi:10.1109/ICRAIE.2014.6909324

[81] Hara, K., & Nakayamma, K. (1994). Comparison of activation functions in multilayer neural network for pattern classification. In *, 1994 IEEE International Conference on Neural Networks, 1994. IEEE World Congress on Computational Intelligence*, *5,* 2997–3002. doi:10.1109/ICNN.1994.374710

[82] Pang, B., & Lee, L. (2005). Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Stroudsburg, PA, USA: Association for Computational Linguistics, 115–124. doi:10.3115/1219840.1219855

[83] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies,* Stroudsburg, PA, USA: Association for Computational Linguistics, *1*, 142–150.

[84] Cheeti, S. (2012). *Cross-Domain Sentiment Classification Using Grams Derived From Syntax Trees And An Adapted Naive Bayes Approach*. PhD thesis. B.Tech, Jawaharlal Nehru Technology University (JNTU).

[85] Kim, J., Rousseau, F., & Vazirgiannis, M. (2015). Convolutional Sentence Kernel from Word Embeddings for Short Text Categorization. *EMNLP*, 775–780.

[86] Dai, A. M., & Le, Q. V. (2015). Semi-supervised Sequence Learning. *arXiv:1511.01432 [cs]*.

[87] Fernandez, B. B. (2015). *The degree of polarity as a factor for deep-learning based Sentiment Analysis*. PhD thesis. UNIVERSITY OF VIGO.

Abstract

Long Short-Term Memory (LSTM) is a kind of Recurrent neural network (RNN) which designed for prevent of vanishing gradient. Each unit of LSTM have been shown with a block. Blocks in this network is kind of memory cell which connected recurrently. Each block contains several gates with activation functions. In Recurrent neural networks such as the LSTM, the sigmoid and hyperbolic tangent functions are commonly used as activation functions in the network units. Other activation functions developed for the neural networks are not thoroughly analyzed in LSTMs. While many researchers have adopted LSTM networks for classification tasks, no comprehensive study is available on the choice of activation functions for the gates in these networks. In this thesis, we gather and compare 23 different kinds of activation functions in a basic LSTM network with one hidden layer. Performance of different activation functions and different number of LSTM blocks in the hidden layer are analyzed for classification of records in the IMDB and the Movie Review data sets. Additionally we compare number of blocks in hidden layer. The quantitative results on both data sets demonstrate that the least average error is achieved with the *modified Elliott* and *cloglogm* activation functions. Specifically, this function exhibits better results than the *sigmoid* activation function which is prevalent in LSTM networks. Results have been shown bigger range of activation function have better results and selecting the optimum number of blocks in hidden layer have intercommunicated with complexity and length of data sets.

Keywords: neural network, LSTM, activation function, sigmoidal gate

Shahrood University of Technology

Faculty of E-Learning

MSc Thesis in Computer Engineering, Artificial Intelligence

Analysis and comparison of different activation functions in LSTM networks

By: Amir Farzad

Supervisor:

Dr. Hoda Mashayekhi

July 2016