به نام خدا

دانشگاه صنعتی امیرکبیر

دانشکده مهندسی کامپیوتر

پروژه پایانی شبکه‌های عصبی

شبکه‌های عصبی کانولوشنی زمانی

استاد: دکتر صفابخش

دانشجو: حلیمه رحیمی

شماره دانشجویی: ۹۹۱۳۱۰۴۳

بهار ۱۴۰۰

چکیده

در سال‌های اخیر رایج‌ترین رویکرد در برابر دنباله‌های زمانی بهره‌گیری از شبکه‌های بازگشتی از جمله شبکه شامل واحد حافظه طولانی کوتاه مدت و یا واحد بازگشتی دروازه‌دار بوده است. بعلاوه پیش از آن در موارد پیش‌بینی دنباله زمانی، از مدل‌هایی همچون مدل خود همبسته استفاده می‌شد. تمامی این روش‌ها به شکلی بازگشتی عمل کرده و با توجه به مقادیری که دنباله در گذشته داشته پیشبینی‌های خود را انجام می‌دهند (البته با تغییراتی می‌توان مقادیر هر دو جهت گذشته و آینده‌ی دنباله را مورد استفاده قرار داد).

از سویی با وجود آنکه شبکه‌های کانولوشنی در زمینه‌ی تصویر بسیار مطرح شده و در برخی موارد به دقت‌های بیشتر از انسان نیز دست یافته‌اند، تنها بر روی تصاویر به کار نمی‌روند و راه خود را در کار با انواع داده‌ها باز کرده‌اند.

شبکه‌های کانولوشنی زمانی در تلاش است مشابه شبکه‌های بازگشتی با توجه به مقادیر گذشته‌ی دنباله‌ی مورد نظر، به مدل‌سازی بپردازد. از جمله خصوصیات آن نیز دخیل نبودن مقادیر آینده است و برای حل این مسئله از کانولوشن‌های سببی استفاده می‌کند. این شبکه از آن جهت که محاسبات آن به شکلی موازی انجام می‌گیرد و نه متوالی، در آموزش سرعت بیشتری از خود نشان می‌دهد. همچنین با وجود کانولوشن سببی منبسط می‌تواند وابستگی‌های طولانی مدت در ورودی را یاد بگیرد و با استفاده از اتصالات باقی‌مانده از محوشدگی گرادیان جلوگیری کند.

در این گزارش به بررسی دقیق‌تر این نوع از شبکه می‌پردازیم و اصلاحات و تغییرات صورت گرفته را به همراه شکاف‌هایی که هنوز وجود دارد، بیان می‌داریم. در انتها نیز عملکرد آن را در کاربردهای گوناگون ارائه می‌دهیم.

واژه‌های کلیدی: شبکه‌های کانولوشنی زمانی، شبکه‌های بازگشتی، کانولوشن سببی منبسط، کانولوشن سببی

فهرست مطالب

فهرست اشکال و جداول

۱- مقدمه

یکی از انواع داده که بسیار ممکن است با آن برخورد داشته باشیم، دنباله‌های زمانی است. تفاوت مهم این نوع از داده با انواع دیگر، وابستگی آن به مقادیر پیشین خود است. کاربردهای گوناگونی در زمینه‌ی دنباله‌های زمانی از جمله قطعه‌بندی فعالیت، دسته‌بندی ویدیو، ترجمه، پیش‌بینی، مدل‌سازی دنباله‌های زمانی و غیره نیازمند ساختاری است که بتواند بعد زمان را در نظر بگیرد.

در برابر این کاربردها تا کنون رایج‌ترین رویکردها شبکه‌های بازگشتی بوده است. این نوع از شبکه‌ها شامل یک یا چند حلقه‌ی پس‌خور بوده و وجود حلقه‌ها شبکه را قادر به نگه‌داری ارائه‌ای از حالت می‌سازد. به عبارتی هر حالت نمایانگر حالات پیشین و هر آنچه که شبکه تاکنون به خود دیده، است. آموزش شبکه‌های بازگشتی ابتدایی دشوار بوده و از مشکل محوشدگی/ انفجار گرادیان رنج می‌برد که موجب می‌گردید وابستگی‌های طولانی مدت فراموش گردد. برای حل این مسئله روش‌های جدیدی از جمله دو معماری واحد حافظه طولانی کوتاه-مدت [1] و واحد بازگشتی دروازه‌دار [2] معرفی شد. البته این به معنی کنار گذاشته شدن شبکه‌های ابتدایی نیست.

از سویی شبکه‌های کانولوشنی که در ابتدا در مبحث مرتبط با تصویر وارد شده بودند [3]، در حال حاضر برای کشف ویژگی از انواع داده‌ها از جمله دنباله‌ها به کار گرفته می‌شوند. شبکه‌های کانولوشنی در وظایف پردازش زبان طبیعی به طور مثال برچسب‌زنی ادات گفتار و برچسب‌زنی نقش معنایی مورد استفاده قرار گرفته است [6]–[4]. همچنین برای دسته‌بندی جمله [8] ,[7] و دسته‌بندی متن بکار رفته‌اند.

آنچه در این گزارش به توصیف آن می‌پردازیم برگرفته از معماری شبکه‌ی ویونت[۱] است. [9] علاوه بر معرفی کانولوشن سببی منبسط و استفاده از اتصالات باقی‌مانده، با بهره‌گیری از تابع فعالیت دروازه‌دار به یادگیری و تولید صوت می‌پردازد.

[10] برای اولین بار از معماری مشابه شبکه‌ی ویونت برای تشخیص و قطعه‌بندی فعالیت در ویدیوها استفاده کرد و معماری حاصل را شبکه‌های کانولوشنی زمانی عنوان کرد. البته پیش از این نیز [11] معماری به شکل کدگذار-کدگشا را با استفاده از کانولوشن تک بعدی که مقادیر پیشین دنباله را در نظر می‌گرفت، طراحی کرده و عنوان مشابه را برای ساختار خود به کار برده بود.

---

[۱] Wavenet

برای حفظ وابستگی‌های طولانی مدت، نوعی از شبکه‌های کانولوشنی با عنوان شبکه‌های کانولوشنی زمانی معرفی شده است. به جای کانولوشن‌های معمولی، این نوع از شبکه‌ها از کانولوشن‌های سببی[۲] و منبسط[۳] بهره می‌برند تا بتوانند تنها از مقادیر گذشته استفاده کنند و میدان دید[۴] بیشتری نیز داشته باشند. بهره‌گیری از بلوک‌های باقی‌مانده نیز موجب مقاومت شبکه در برابر محوشدگی/ انفجار گرادیان می‌گردد. در مقایسه با شبکه‌های بازگشتی، این نوع شبکه از مزایا و معایبی برخوردار است که به آنها نیز خواهیم پرداخت.

۲- مولفه‌های شبکه کانولوشنی زمانی

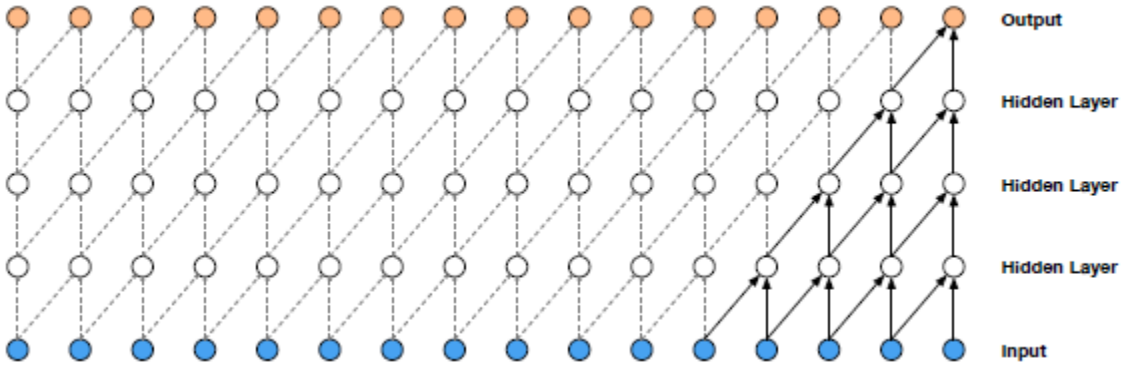
شبکه‌های کانولوشنی زمانی شامل دو مولفه اصلی هستند: کانولوشن سببی منبسط[۵] و بلوک باقی‌مانده.[۶] البته لازم به ذکر است این امکان وجود دارد که سببی بودن کانولوشن‌ها دستخوش تغییر قرار بگیرد. در اینجا به مولفه‌های این نوع شبکه می‌پردازیم. در بخش بعدی جزئیات معماری‌های مختلف از جمله ویونت بیان می‌گردد که برخی شامل نکاتی جهت ایجاد تغییرات به تناسب مسئله است.

۲-۱- کانولوشن سببی منبسط

برای آنکه مقادیر آینده در محاسبات حال نقشی نداشته باشد، لازم است کانولوشن معمول با مقداری جابجایی صورت بگیرد. به عبارتی ساختار به گونه‌ای است که هر نورون تنها به ورودی‌های پیش از خود وابسته است. در شکل (۱) نحوه اتصالات را مشاهده می‌کنید. علاوه بر این، برای آنکه اندازه‌ی ورودی با خروجی یکسان باشد، از گسترش مرز با صفر استفاده می‌شود؛ با این تفاوت که با داشتن فیلتر با اندازه‌ی K، تعداد $K-1$ صفر پیش از مقادیر دنباله قرار می‌گیرد.

---

[۲] Causal
[۳] Dilated
[۴] Receptive Field
[۵] Dilated Causal Convolution
[۶] Residual Block

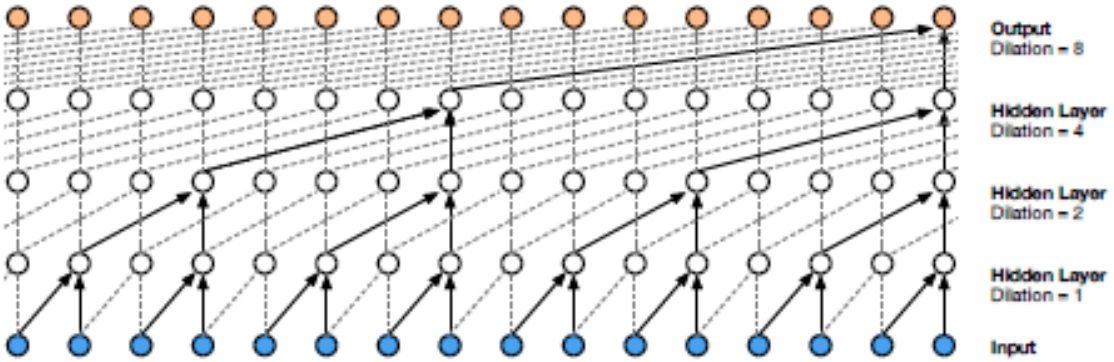


شکل ۱ - پشته ای از لایه های کانولوشنی سببی (تصویر از [9])

چنین ساختاری با وجود آنکه مسئله‌ی نبود وابستگی به آینده را حل می‌کند اما میدان دید در آن تنها متناسب با عمق شبکه و طول فیلتر طبق رابطه #layers + filter length – 1 افزایش می‌یابد. از طرفی تعداد لایه‌های بیشتر موجب محوشدگی/ انفجار گرادیان و افزایش طول فیلتر موجب بیشتر شدن تعداد پارامترها و در نتیجه محاسبات می‌گردد. بنابراین در این حالت وابستگی‌های طولانی مدت ممکن نیست. به همین دلیل مفهوم کانولوشن منبسط به ساختار کنونی افزوده می‌گردد. این نوع از کانولوشن پیش از این در موارد مختلف به کار رفته است و با ایجاد فضاهای خالی میان مولفه‌های فیلتر موجب افزایش میدان دید می‌گردد. عمل کانولوشن منبسط بر مولفه s از دنباله‌ی ورودی x با درجه‌ی انبساط d مطابق رابطه (۱) انجام می‌گیرد و $s - d.i$ در آن نشاندهنده‌ی جهت به سمت گذشته است.

$$F(s) = (x *_d f) = \sum_{i=0}^{k-1} f(i).x_{s-d.i} \qquad \text{رابطه (۱)}$$

با قرار دادن d برابر با یک، این نوع از کانولوشن به کانولوشن معمول تغییر می‌یابد. استفاده از انبساط بیشتر هر خروجی در بالاترین لایه را قادر می سازد ارائه دهنده‌ی ورودی‌ها در بازه‌ی وسیع‌تری باشند و در نتیجه میدان دید را گسترش می‌دهد. در شکل (۲) پشته‌ای از لایه‌های کانولوشن سببی منبسط را می‌بینید.



شکل ۲- پشته ای از لایه های کانولوشنی سببی منبسط (تصویر از [9])

این اصلاحات باعث ایجاد تنوع بیشتری برای افزایش میدان دید می‌گردد: از جمله طویل‌تر کردن فیلتر و انبساط بیشتر با مقدار d بزرگ‌تر. در لایه‌ای شامل فیلترهای با اندازه K و عامل انبساط d، تاریخچه‌ی موثر برابر با $(K-1)\,d$ خواهد بود. در اینجا نیز همچون روند معمول در استفاده از کانولوشن منبسط با افزایش عمق شبکه مقدار d به طور نمایی افزایش می‌یابد. این کار باعث می‌گردد فیلتری داشته باشیم که به تمام ورودی‌ها درون تاریخچه‌ی موثر دسترسی داشته درحالیکه اجازه می‌دهد وسعت این تاریخچه به علت استفاده از شبکه‌های عمیق افزایش یابد. می‌توان این افزایش d را مقدار خاصی ادامه داد و سپس از لایه‌های بعد، مقادیر را از نو شروع کرد.

۲-۲- اتصال باقی‌مانده

در این نوع از شبکه برای افزودن به میدان دید، همچنان می‌توان از شبکه‌های عمیق‌تر بهره برد که ممکن است دچار محوشدگی گرادیان شود. برای حل این مشکل، پیشنهاد شده از بلوک باقی‌مانده استفاده گردد. این بلوک شامل اتصال شاخه ای از ورودی یک بلوک به خروجی آن و جمع آنها با یکدیگر می‌باشد. چنین عملی موجب می‌شود همواره مقدار قابل توجهی برای پس انتشار موجود باشد.

هر بلوک باقی‌مانده به طور معمول شامل لایه‌های کانولوشن سببی منبسط، نرمال‌سازی، فعالیت و دراپ-اوت[۷] است. به علاوه ممکن است از یک کانولوشن یک در یک برای همسان سازی اندازه‌ی تنسورها جهت اطمینان از یکی بودن ابعاد برای جمع عنصر به عنصر استفاده گردد.

[۷] Drop-out

۳- جزئیات در بررسی چند معماری

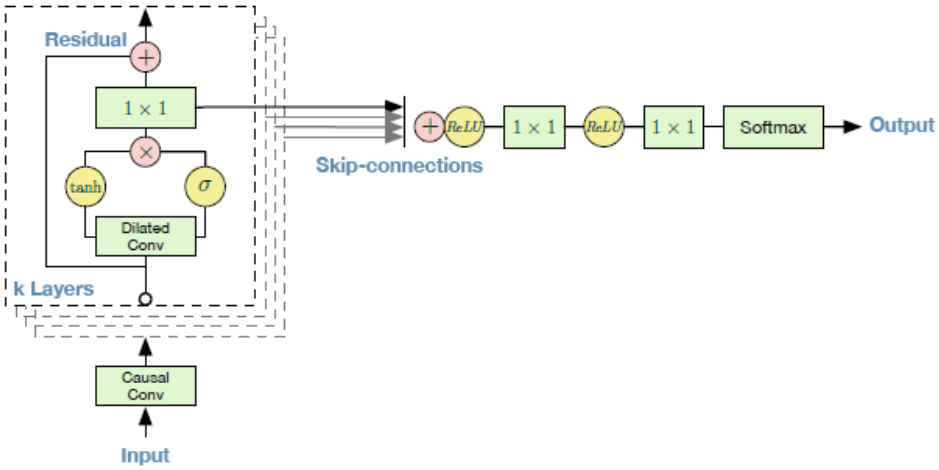
در این بخش برخی جزئیات برای درک بهتر و بهره بردن از شبکه کانولوشنی زمانی ارائه می‌گردد.

۳-۱- پیشبینی خود همبسته[۸]

شکل (۳) ساختار کلی شبکه ویونت ([9]) را نمایش می‌دهد. در هر بلوک باقی‌مانده از یک لایه کانولوشن سببی منبسط استفاده شده که خروجی آن به تابع فعالیت دروازه‌دار داده می‌شود که محاسباتی مطابق با رابطه (۲) صورت می‌گیرد. سپس نتیجه‌ی حاصل از آنها ضرب عنصر به عنصر می‌گردد.

$$z = \tanh(W_{f,k} * x) \odot \sigma(W_{g,k} * x) \qquad \text{رابطه (۲)}$$

در این شبکه علاوه بر اتصالات باقی‌مانده، اتصال پرشی نیز جهت افزایش سرعت همگرایی و افزودن بر قابلیت ساخت شبکه‌های عمیق‌تر بکار رفته است.



شکل ۳ - معماری شبکه ویونت (تصویر از [9])

مشهود است که این شبکه به شکل خود همبسته عمل می‌کند و در حین پیش‌بینی هر قدم لازم است محاسبات بسیاری انجام گیرد.

---

[۸] Auto Recursive Forecasting

۳-۲- هر نورون، نماینده قدم‌های پیشین

در [10] برای تشخیص و قطعه‌بندی فعالیت در ویدیو، از ساختار مشابه ویونت استفاده می‌شود. به دلیل سببی بودن کانولوشن‌های به کار رفته هر یک از نورن‌ها می‌تواند نشاندهنده‌ی وضعیت از زمان اولین قدم در ورودی کنونی تا به آن هنگام باشد. علاوه بر این، می‌توان این نوع شبکه را به علت ساختار سلسله مراتبی که دارد، کاشف ویژگی‌های زمانی بازه کوچک، متوسط و بزرگ دانست، برخلاف شبکه‌های بازگشتی که ورود متوالی مقادیر موجب می‌گردد ویژگی‌هایی با انتزاع بالا را کشف کنیم [11]. به عبارت ساده‌تر، شاید بتوان به هر قدم زمانی پیشین وزنی داد ولی انتزاع‌های بالاتر، حاصل از ترکیب ویژگی‌ها در انتزاع پایین‌تر نیست.

در ساختار شبکه‌ی بکار رفته، هر بلوک باقی‌مانده به ترتیب شامل یک لایه کانولوشن سببی منبسط با طول فیلتر ۲، لایه‌ی دراپ-اوت مکانی و فعالیت غیرخطی است. سپس خروجی با استفاده از کانولوشن یک در یک به ابعاد ورودی درآمده و با آن جمع خواهد شد.

در هر بلوک، خروجی طبق رابطه (۳) بدست می‌آید که پس از اضافه کردن ارتباط باقی‌مانده، خروجی رابطه (۴) را خواهیم داشت.

$$\hat{S}_t^{(j,l)} = f(W^{(1)} S_{t-s}^{(j,l-1)} + W^{(2)} S_t^{(j,l-1)} + b) \qquad \text{رابطه (۳)}$$

$$S_t^{(j,l)} = S_{t-s}^{(j,l-1)} + V\hat{S}_t^{(j,l)} + e \qquad \text{رابطه (۴)}$$

تابع فعالیت غیر خطی f می‌تواند همان فعالیت دروازه‌دار شبکه ویونت باشد.

سه عدد از این بلوک‌ها در بلوکی به ترتیب با ضریب انبساط ۱، ۲ و ۴ قرار می‌گیرند و ارتباط پرشی، خروجی این لایه‌ها را در انتهای بلوک‌ها جمع می‌زند. رابطه (۵) این عمل را نمایش می‌دهد و ساختار کلی را می‌توان در شکل (۴) مشاهده نمود.

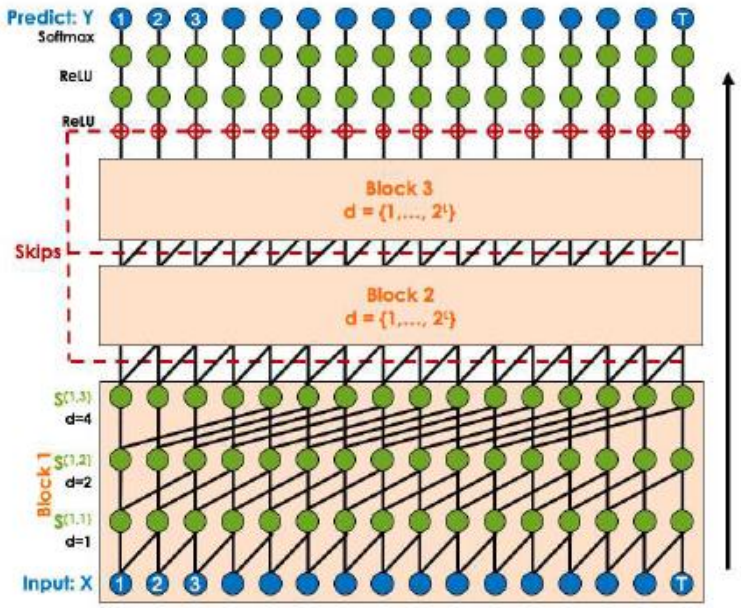
$$Z_t^{(0)} = \mathrm{ReLU}(\sum_{j=1}^{B} S_t^{(j,L)}) \qquad \text{رابطه (۵)}$$

سپس با گذر از دو لایه کانولوشنی به ترتیب با تابع فعالیت واحد خطی اصلاح‌شده (ReLU) و تابع بیشینه هموار (Softmax) به پیشبینی می‌پردازد.

$$Z_t^{(1)} = \mathrm{ReLU}(V_r Z_t^{(0)} + e_r)$$ رابطه (۶)

$$\hat{Y}_t = softmax(UZ_t^{(1)} + c)$$ رابطه (۷)

ورودی شبکه کانولوشنی زمانی ویژگی‌های ویدیو از جمله خروجی‌های مکانی یا زمانی-مکانی شبکه‌های کانولوشنی به ازای هر قاب یک ویدیو می‌باشد. هر ورودی $X_t \in \mathbb{R}^{F_0}$ است که در آن F_0 طول بردار ویژگی‌ها برای قدم زمانی $1 < t < T$ می‌باشد. تعداد قدم‌های زمانی T ممکن است برای هر دنباله‌ی ویدیو متفاوت باشد که با گسترش مرز با صفر به یک اندازه درمی‌آید. برچسب عمل برای هر قاب توسط بردار $Y_t \in \{0,1\}^C$ با C به عنوان تعداد کلاس‌ها داده می‌شود، به گونه‌ای که کلاس مورد نظر یک و باقی صفر باشند. خروجی پیشبینی شده Y_t نشاندهنده‌ی نوع عمل کنونی با توجه به قدم‌ها تا t است.



شکل ۴- معماری Dilated TCN بکار رفته برای قطعه‌بندی عمل (تصویر از [10])

۳-۳- پیشبینی غیر خود همبسته

در [12] برای مدل‌سازی دنباله درون بلوک‌ها، دو لایه‌ی متوالی متشکل از کانولوشن سببی منبسط، نرمال‌سازی وزن‌دهی شده، فعالیت ReLU و دراپ-اوت مکانی استفاده می‌شود. علاوه بر این مانند قبل لایه‌ی یک در یک نیز به کار رفته است تا جمع عنصر به عنصر با ورودی تسهیل یابد.

این نوع معماری با حذف بخش انتهای ویونت و اتصالات پرشی، باعث می‌شود بتوان پیشبینی‌های غیر خود همبسته انجام داد.

در [13] برای دسته‌بندی سیگنال‌های الکتروانسفلوگرافی، ورودی در دامنه‌ی مکانی-زمانی دریافت می‌شود و از آن جهت که همچنان پس از گذر ورودی از لایه کانولوشن زمانی، کانولوشن عمقی و سپس کانولوشن جداپذیر، کشف اطلاعات زمانی ممکن است، خروجی را به یک معماری شبکه کانولوشنی زمانی می‌دهد. اولین بلوک شبکه کانولوشنی زمانی تعداد ویژگی‌های ورودی را به مقدار خاصی می‌رساند و آن را در طول شبکه ثابت نگه می‌دارد.

معماری شبکه کانولوشنی زمانی به کار رفته در اینجا، مشابه [12] بوده اما در میان لایه‌های کانولوشنی از نرمال‌سازی دسته‌ای به جای نرمال‌سازی وزن‌دهی شده استفاده می‌کند؛ از آن جهت که [14] نشان می‌دهد این نوع نرمال‌سازی برای شبکه‌های باقی‌مانده با اندازه‌های بزرگ دقت بیشتری نسبت به نرمال‌سازی وزن‌دهی شده می‌دهد. دراپ-اوت معمولی نیز به جای دراپ-اوت مکانی قرار گرفته است؛ چرا که با وجود کانولوشن در ابتدای ساختار کلی، دیگر قاب‌های مجاور در نقشه‌های ویژگی همبستگی قوی با یکدیگر ندارند و بنابراین بهتر است به جای دور انداختن کل یک نقشه‌ی ویژگی تک بعدی، مولفه‌های تکی را برای منظم‌سازی فعالیت‌ها دور بریزیم. در اینجا از تابع فعالیت واحد خطی نمایی (ELU) به جای ReLU استفاده می‌گردد. این عمل به دلیل اینکه شبکه نتیجه بهتری را برای مسئله مورد نظر حاصل می‌داد انجام شده است.

[15] به طور غیر خود همبسته[۹] و با استفاده از ورودی‌های برون‌زا[۱۰] به پیشبینی احتمالاتی مقادیر در چند قدم آینده می‌پردازد. در مدل‌های غیر خود همبسته به جای آنکه مقادیر چند قدم به طور متوالی پیشبینی شود به طور موازی انجام می‌گیرد.

مدل‌های رگرسیون پویا (همچون آریماکس[۱۱]) مدل کلاسیک دنباله‌ی زمانی را گسترش می‌دهد تا هم اطلاعات مشاهدات پیشین و هم متغیرهای برون‌زا را شامل شود. می‌توان مدل‌های رگرسیون پویا را به صورت زیر ارائه کرد:

$$y_t^{(i)} = v_B\left(X_t^{(i)}\right) + n_t^{(i)} \qquad \text{رابطه (۸)}$$

در این رابطه $v_B(.)$ تابع انتقالی است که تغییرات ورودی‌های برون‌زای $X_t^{(i)}$ به $y_t^{(i)}$ را توصیف می‌کند و $n_t^{(i)}$

---

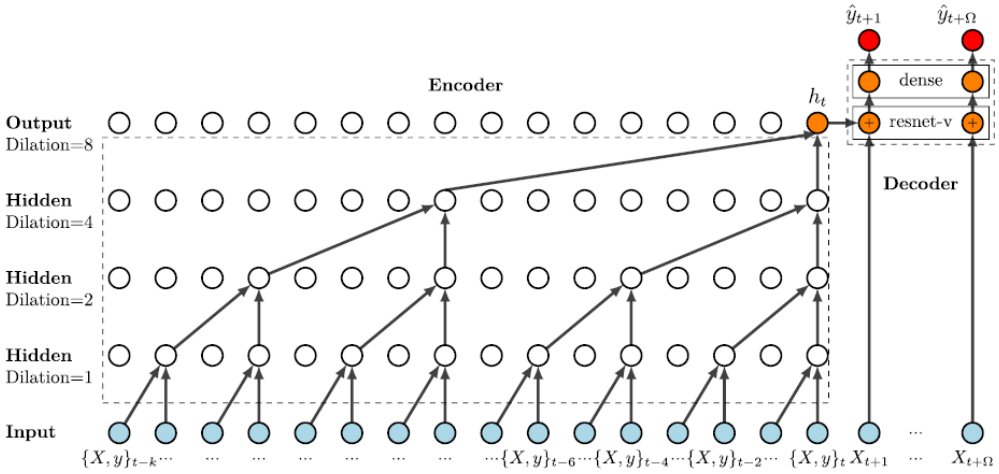
[۹] Non-Auto Recursive

[۱۰] Exogenous

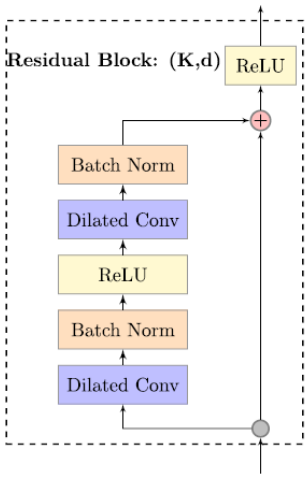
[۱۱] ARIMAX

فرآیند دنباله‌ی زمانی تصادفی (همچون فرآیند آریما[۱۲]) است که یک پیشبینی از $y_t^{(i)}$ را براساس مقادیر پیشین دنباله به‌دست می‌اورد.

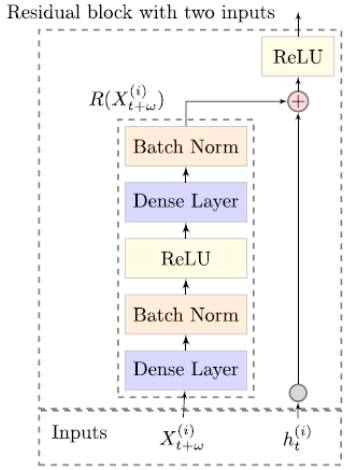
برای گسترش مدل رگرسیون پویا به نوعی که از متغیرهای دیگر ورودی بگیرد، [15] از یک ارتباط باقی‌مانده متفاوت استفاده می‌کند که دو نوع ورودی را از خود عبور دهد. ساختار کلی این شبکه در شکل (۵-a) آمده است.



(a) Architecture of **DeepTCN**



(b) Encoder module



(c) Decoder module

شکل ۵ – معماری Deep TCN بکار رفته برای پیشبینی دنباله زمانی، a) ساختار کلی شبکه، b) بخش کدگذار شبکه، c) بخش کدگشای شبکه (تصویر از [15])

[۱۲] ARIMA

در اینجا معماری کلی به دو بخش کدگذار و کدگشا تقسیم می‌شود. بخش کدگذار آن شامل بلوک‌های باقی‌مانده است که در شکل (۵-b) جزئیاتش به تصویر کشیده شده است. همانطور که پیش از این نیز بیان شد، فیلتری وجود دارد که به تمام ورودی‌ها درون تاریخچه‌ی موثر دسترسی دارد. این فیلتر در نورون T آخرین لایه درون بلوک باقی‌مانده قرار می‌گیرد. خروجی این نورون وارد بلوک باقی‌مانده‌ی بخش رمزگشا شده و طبق شکل (۵-c) با خروجی حاصل از توابع اعمال شده بر ورودی‌های برون‌زا جمع می‌شود.

تابعی که در بخش کدگذار انجام می‌گیرد طبق رابطه (۹) می‌باشد که در آن $h_t^{(i)}$ خروجی پنهان رمزگذار و $X_{t+\omega}^{(i)}$ ورودی‌های برون‌زای پسین است. بنابراین تابع غیرخطی نقش تابع انتقال را ایفا می‌کند که باقی‌مانده‌ی مقدار هدف را با پیشبینی‌هایی که تنها براساس مقادیر پیشین دنباله بدست آمده‌اند، روشن می‌کند.

$$\delta_{t+\omega}^{(i)} = \mathrm{R}\left(X_{t+\omega}^{(i)}\right) + h_t^{(i)} \qquad \text{رابطه (۹)}$$

۳-۴- شبکه دو جهته

[16] برای تشخیص و مکان‌یابی رویداد صوتی، معماری [17] را با استفاده از شبکه کانولوشنی زمانی پیاده سازی می‌کند. در این معماری به جای واحدهای بازگشتی دروازه‌دار دوجهته، یک شبکه‌ی کانولوشنی زمانی قرار می‌گیرد. این شبکه مشابه مدل ویونت بوده اما برای دوجهته کردن آن، خاصیت سببی بودن را از کانولوشن حذف کرده و تنها از کانولوشن‌های منبسط استفاده می‌شود. بعلاوه یک لایه نرمال‌سازی دسته‌ای پس از هر لایه کانولوشن اضافه شده است. این شبکه شامل ده بلوک باقی‌مانده با درجه انبساط صفر الی ۹ است.

همچنین [10] که پیش از این معرفی شد شبکه‌ی خود را با کانولوشن منبسط نیز می‌آزماید.

۴- آزمایشات

برای مشاهده نتایج بیشتر به منابع ارجاع داده می‌شود.

۴-۱- تشخیص و قطعه‌بندی عمل

روش‌های قطعه‌بندی عمل پیشبینی می‌کند چه عملی در حال انجام در هر قاب است و روش‌های تشخیص، مجموعه تنکی از قطعات عمل را به عنوان خروجی می‌دهد که با یک زمان ورودی، خروجی و برچسب کلاس همراه است. با قرار دادن قطعات خالی/پس‌زمینه می‌توان از پیشبینی قطعه‌بندی به تشخیص رسید.

شبکه بکار رفته در [10] که پیش از این معرفی شد، بر مجموعه داده‌های مختلفی آزمایش شده است؛ ازجمله MERL Shopping که نتایج آن در جدول (۱) قابل مشاهده است. دقت در اینجا با توجه به هر قاب محاسبه می‌شود که مسئله ازدیاد قطعه‌بندی را در نظر نمی‌گیرد. منظور از ازدیاد قطعه‌بندی، انتساب قطعاتی از یک عمل، به عملی دیگر است. به همین دلیل معیار F1@k را معرفی می‌کند که این مسئله را در خود لحاظ کرده است.

جدول ۱ – مقایسه عملکرد برای تشخیص و قطعه‌بندی عمل بر روی مجموعه داده‌ی MERL Shopping (جدول از [10])

MERL (acausal)	F1@{10, 25, 50}	mAP	Acc
MSN Det [28]	46.4, 42.6, 25.6	**81.9**	64.6
MSN Seg [28]	80.0, 78.3, 65.4	69.8	76.3
Dilated TCN	79.9, 78.0, 67.5	75.6	76.4
ED-TCN	**86.7, 85.1, 72.9**	74.4	**79.0**
MERL (causal)	**F1@{10, 25, 50}**	**mAP**	**Acc**
MSN Det [28]	-	**77.6**	-
Dilated TCN	72.7, 70.6, 56.5	72.2	73.0
ED-TCN	**82.1, 79.8, 64.0**	64.2	**74.1**

در اینجا دو نوع شبکه معرفی شده است؛ یکی براساس معماری کدگذار-کدگشا (ED-TCN) که از ساختاری بدون اتصال باقی‌مانده و لایه‌ی کانولوشنی منبسط استفاده می‌کند، و دیگری آنچه در بخش (۳-۲) معرفی شد (Dilated TCN). احتمال می‌رود علت آنکه شبکه‌ی دوم در دقت و F1@k به خوبی شبکه‌ی اول عمل نکرده، به دلیل فیلتر کوچک‌تر باشد. فیلتر در Dilated TCN برابر با ۲ و در ED-TCN برابر با ۴ می‌باشد.

مشاهده می‌شود که با دو جهته کردن کانولوشن، نتایج همانطور که انتظار می‌رود، بهبود می‌یابد.

۴-۲- مدل‌سازی دنباله

در مدل‌سازی دنباله با داشتن دنباله‌ی ورودی $x_0, \dots, x_T$ سعی در پیش‌بینی خروجی‌های $y_0, \dots, y_T$ داریم. برای پیشبینی خروجی y_t به ازای زمان t محدود شده‌ایم تا تنها از مقادیری که تا آن زمان مشاهده شده‌اند $(x_0, \dots, x_t)$ استفاده کنیم.

[12] به مقایسه عملکرد شبکه کانولوشنی زمانی با چند مدل دیگر در وظیفه‌ی مدل‌سازی دنباله بر مجموعه داده‌های مختلف پرداخته است.

جدول ۲- مقایسه عملکرد برای مدل‌سازی دنباله زمانی (جدول از [12])

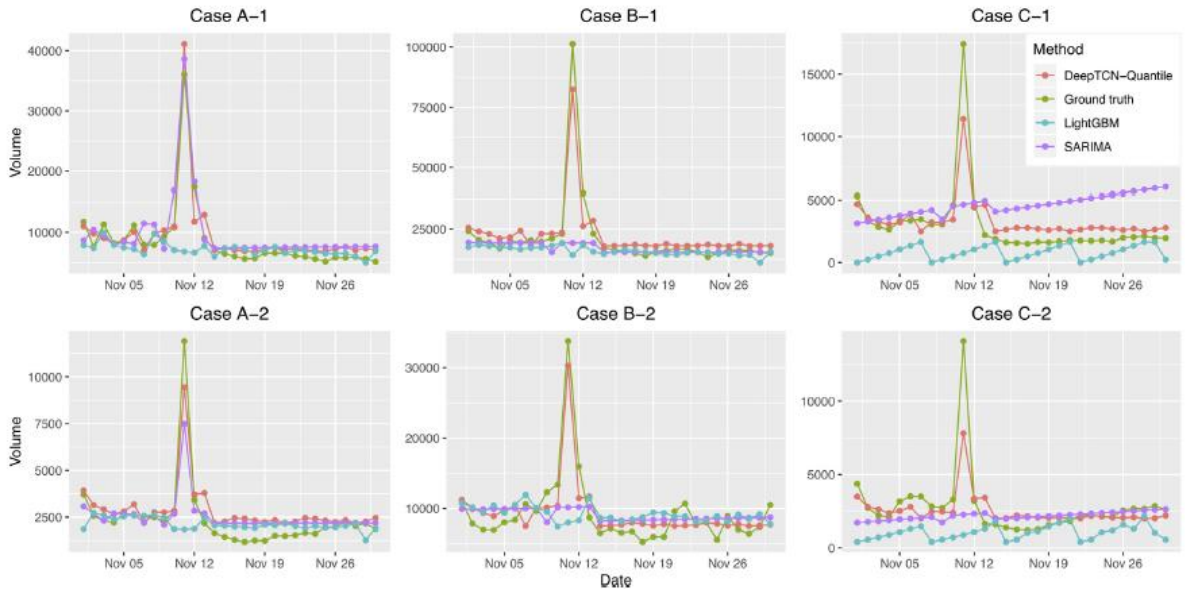
Sequence Modeling Task	Model Size (≈)	Models			
		LSTM	GRU	RNN	TCN
Seq. MNIST (accuracy^h)	70K	87.2	96.2	21.5	**99.0**
Permuted MNIST (accuracy)	70K	85.7	87.3	25.3	**97.2**
Adding problem T=600 (loss$^\ell$)	70K	0.164	**5.3e-5**	0.177	**5.8e-5**
Copy memory T=1000 (loss)	16K	0.0204	0.0197	0.0202	**3.5e-5**
Music JSB Chorales (loss)	300K	8.45	8.43	8.91	**8.10**
Music Nottingham (loss)	1M	3.29	3.46	4.05	**3.07**
Word-level PTB (perplexity$^\ell$)	13M	**78.93**	92.48	114.50	88.68
Word-level Wiki-103 (perplexity)	-	48.4	-	-	**45.19**
Word-level LAMBADA (perplexity)	-	4186	-	14725	**1279**
Char-level PTB (bpc$^\ell$)	3M	1.36	1.37	1.48	**1.31**
Char-level text8 (bpc)	5M	1.50	1.53	1.69	**1.45**

مجموعه داده‌ی Sequential MNIST و P-MNIST بارها برای آزمودن قابلیت شبکه برای حفظ اطلاعات در مدت زمان طولانی مورد استفاده قرار گرفته است. مشاهده می‌شود که شبکه کانولوشنی زمانی قابلیت بیشتری را از خود نشان داده است. همچنین مجموعه داده‌ی LAMBADA در مدل‌سازی زبان، نیازمند چنین قابلیتی می‌باشد.

۴-۳- پیش‌بینی دنباله زمانی

برای نمایش بهبود عملکرد، معماری [15] با چندی از روش‌های آخرین پیشرفت‌های علمی در این زمینه در شکل (۶) مقایسه شده است.

این نتایج مرتبط با مجموعه داده‌ی JD-Shipment از JD.com بزرگترین خرده‌فروشی آنلاین چین می‌باشد. A-1 و A-2 مرتبط به مدلی است که بر داده‌های دو سال آموزش دیده، B-1 و B-2 بر داده‌هایی که شامل زمان‌های جشنواره‌ی خرید نبوده، و C-1 و C-2 بر داده‌های بسیار کمی (کمتر از سه روز) آموزش دیده است.



شکل ۶ – مقایسه عملکرد برای پیش‌بینی دنباله زمانی بر روی شش مورد تصادفی از مجموعه داده‌ی JD-Shipment (تصویر از [15])

۴-۴- دسته‌بندی سیگنال‌های تصویرساز محرک[۱۳]

رابط‌های ماشین-مغز ارتباط مستقیم بین انسان و دستگاه‌های خارجی را با تحلیل فعالیت‌های عصبی مغز (به طور معمول با ثبت سیگنال‌های الکتروانسفلوگرافی[۱۴]) ممکن می‌سازد. یکی از رویکردها براساس تصویرسازی محرک است که در واقع فرآیند شناختی تفکر درباره‌ی حرکت دادن بخشی از بدن بدون انجام آن می‌باشد. کدگشایی موفقیت‌آمیز سیگنال‌های الکتروانسفلوگرافی به دلیل نسبت سیگنال به نویز و واریانس بالا میان افراد مختلف که از ایجاد یک مدل برای همه‌ی افراد نمونه جلوگیری می‌کند، همچنان دشوار است. به همین دلیل در [13] از دو نوع مدل استفاده شده؛ ایستا (EEG-TCNet) که برای آن پارامترها با توجه به داده‌ی همه افراد تنظیم شده و متغیر (Variable EEG-TCNet) که پارامترها برای هر فرد نمونه به طور جدا تنظیم شده است.

جدول (۳) مقایسه‌ای میان شبکه‌های مختلف بر روی مجموعه داده‌ی BCI COMPETITION IV-2A است.

---

[۱۳] Motor Imagery

[۱۴] Electroencephalography

جدول ۳ – مقایسه عملکرد برای دسته‌بندی سیگنال‌های تصویرسازی محرک بر روی مجموعه داده‌ی BCI Competition IV-2A (جدول از [13])

	Mean Accuracy	Parameters	Mean MACs	Feature Map [kB]
EEGNet*[10]	72.40	2.63 k	13.1 M	396
Shallow ConvNet*[9]	74.31	47.3 k	63.0 M	1013
FBCSP [8]	73.70	261 k	104 M	50
Riemannian [8]	74.77	50.0 k	-	49
MSFBCNN [27]	75.80	155 k	202 M	5775
EEG-TCNET	**77.34**	**4.27 k**	**6.8 M**	**396**
CNN++ [12]	81.10	220 k	18.2 M	499
TPCT [13]	88.87	7.78 M	1.73 G	524
Variable EEGNet	79.02	15.6 k	42.6 M	1584
DFFN (variable) [11]	79.71	1.07 M	132 M	650
***Variable* EEG-TCNET**	**83.84**	**20.5 k**	**12.1 M**	**792**

*Reproduced

۴-۵- تشخیص و مکان‌یابی رویداد صوتی

تشخیص و مکان‌یابی رویداد صوتی وظیفه‌ی شناسایی مشترک محل زمانی و مکانی هر رویداد صوتی در ورودی می‌باشد.

جدول (۴) با مقایسه‌ی دو شبکه معرفی شده در [16] و [17] که تنها در استفاده از شبکه کانولوشنی زمانی با یکدیگر متفاوت‌اند، بهبود عملکرد را بر روی چهار مجموعه داده‌ی ذکر شده در جدول نمایش می‌دهد.

جدول ۴ – مقایسه عملکرد دو شبکه SELDNet و SELD-TCN بکار رفته برای تشخیص و مکان‌یابی رویداد صوتی با درجه نمونه‌برداری مختلف (جدول از [16])

Method	Sampling Rate	ANSYN				MANSYN				REAL				MREAL			
		F1	ER	FR	DE	F1	ER	FR	DE	F1	ER	FR	DE	F1	ER	FR	DE
SELDnet	44.1 [kHz]	93.4	0.11	81.2	17.5	94.6	0.10	90.7	14.2	74.1	0.39	48.2	38.1	70.5	0.44	46.2	**41.2**
SELD-TCN		**95.5**	**0.08**	**86.8**	**16.0**	**95.4**	**0.09**	**92.7**	**13.5**	**75.1**	**0.39**	**52.4**	**35.8**	**72.2**	**0.42**	**46.6**	41.7
SELDnet	16 [kHz]	94.6	0.09	83.1	16.0	95.8	0.07	90.4	14.0	76.1	0.36	51.6	36.5	71.6	0.42	46.6	**42.7**
SELD-TCN		**96.0**	**0.07**	**86.6**	**15.7**	**96.0**	**0.06**	**91.5**	**13.7**	**79.7**	**0.32**	**55.9**	**34.0**	**73.1**	**0.40**	**46.6**	43.9
SELDnet	8 [kHz]	93.1	0.12	76.9	19.9	95.4	0.08	87.8	16.6	75.0	0.37	52.1	38.6	72.2	0.40	45.7	**42.8**
SELD-TCN		**95.2**	**0.08**	**82.3**	**18.1**	**95.5**	**0.08**	**88.6**	**16.3**	**77.8**	**0.34**	**53.5**	**37.3**	**72.9**	**0.39**	**47.5**	44.5

۵- بحث و نتیجه‌گیری

با توجه به آنچه بیان شد، می‌توان مزایای شبکه‌های کانولوشنی زمانی را چنین برشمرد:

- پردازش موازی: برعکس شبکه‌های بازگشتی که به شکل متوالی مقادیر را دریافت کرده و آموزش می‌بینند، این نوع از شبکه‌ها قادر به آموزش موازی هستند.
- اندازه انعطاف پذیر میدان دید: میدان دید یک شبکه‌ی کانولوشنی زمانی را می‌توان به چند طریق افزایش داد؛ برای مثال روی هم قرار دادن چندین لایه‌ی کانولوشن منبسط، استفاده از مقادیر بزرگتر برای درجه انبساط، افزودن اندازه‌ی فیلتر و یا عمق شبکه، همگی ممکن هستند.
- گرادیان‌های باثبات: شبکه کانولوشنی زمانی مسیر پس انتشار متفاوتی از جهت زمانی دنباله دارد و بنابراین مشکل محوشدگی/انفجار گرادیان را که از مشکلات شبکه بازگشتی بود و منجر به ایجاد معماری‌های جدید گردید، ندارد.
- حافظه‌ی کم مورد نیاز برای آموزش: بخصوص در موارد دنباله‌های ورودی طویل، LSTM و GRU می‌تواند حافظه‌ی بسیاری را جهت ذخیره‌ی نتایج جزئی برای دروازه‌های خود پر کند. در حالیکه در شبکه کانولوشنی زمانی فیلترها در یک لایه مشترک بوده و مسیر پس انتشار نیز تنها به عمق شبکه وابسته است.
- ورودی‌های با طول متغیر: همانند شبکه‌های بازگشتی که ورودی‌های با طول متغیر را به صورت بازگشتی مدل می‌کند، شبکه کانولوشنی زمانی نیز با کانولوشن‌های تک بعدی قادر به دریافت هر ورودی با طول متفاوت می‌باشد. به این معنی که می‌توان شبکه کانولوشنی زمانی را در معماری‌ها به راحتی به جای یک شبکه بازگشتی قرار داد.

این شبکه معایبی نیز دارد:

- ذخیره‌ی داده هنگام ارزیابی: در طول ارزیابی شبکه‌های بازگشتی تنها لازم است یک حالت مخفی را نگه دارند و یک ورودی x_t را دریافت کنند. به عبارتی خلاصه‌ای از تمام تاریخچه‌ی مقادیر قدم‌های پیشین در بردارهای با طول ثابت h_t خواهیم داشت و دنباله‌ی مشاهده شده می‌تواند حذف شود. درحالیکه شبکه کانولوشنی زمانی نیاز دارد دنباله‌ی خام را تا طول تاریخچه‌ی موثر دریافت کند و بنابراین به حافظه‌ی بیشتری هنگام ارزیابی نیاز دارد.
- پتانسیل تغییر پارامتر برای انتقال دامنه: دامنه‌های مختلف ممکن است الزامات متفاوتی درباره‌ی میزان تاریخچه‌ی مورد نیاز مدل برای پیشبینی داشته باشد. بنابراین هنگام انتقال مدل از دامنه‌ای که حافظه‌ی کوتاهی را ملزم می‌سازد به دامنه‌ای نیازمند حافظه‌ی بسیار طولانی‌تر، شبکه کانولوشنی زمانی به علت کوچکی میدان دید، عملکرد ضعیفی از خود نشان خواهد داد.

در این گزارش به بررسی مولفه‌های شبکه کانولوشنی زمانی پرداختیم و چند معماری را جهت توجه به جزئیات بررسی کردیم. نتایج به دست آمده از آزمایشات نشان می‌دهد این نوع شبکه توانایی جایگزینی شبکه‌های بازگشتی را متناسب با مسئله دارد.

منابع

[1] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[2] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv Prepr. arXiv1406.1078*, 2014.

[3] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[4] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 160–167.

[5] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12, no. ARTICLE, pp. 2493–2537, 2011.

[6] C. Dos Santos and B. Zadrozny, "Learning character-level representations for part-of-speech tagging," in *International Conference on Machine Learning*, 2014, pp. 1818–1826.

[7] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," *arXiv Prepr. arXiv1404.2188*, 2014.

[8] Y. Kim, "Convolutional neural networks for sentence classification. EMNLP." Association for Computational Linguistics1746--1751, 2014.

[9] A. van den Oord *et al.*, "Wavenet: A generative model for raw audio," *arXiv Prepr. arXiv1609.03499*, 2016.

[10] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.

[11] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in *European Conference on Computer Vision*, 2016, pp. 47–54.

[12] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv Prepr. arXiv1803.01271*, 2018.

[13] T. M. Ingolfsson, M. Hersche, X. Wang, N. Kobayashi, L. Cavigelli, and L. Benini, "EEG-TCNet: An Accurate Temporal Convolutional Network for Embedded Motor-Imagery Brain--Machine Interfaces," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2020, pp. 2958–2965.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[15] Y. Chen, Y. Kang, Y. Chen, and Z. Wang, "Probabilistic forecasting with temporal convolutional neural network," *Neurocomputing*, vol. 399, pp. 491–501, 2020.

[16] K. Guirguis, C. Schorn, A. Guntoro, S. Abdulatif, and B. Yang, "SELD-TCN: sound event localization & detection via temporal convolutional networks," in *2020 28th European Signal Processing Conference (EUSIPCO)*, 2021, pp. 16–20.

[17] S. Adavanne, A. Politis, and T. Virtanen, "Localization, detection and tracking of multiple moving sound sources with a convolutional recurrent neural network," *arXiv Prepr. arXiv1904.12769*, 2019.