شبکه‌های عصبی پیچشی تکه‌ای

کار مطالعاتی شبکه‌های عصبی

محمدرضا غفرانی
۴۰۰۱۳۱۰۷۶
۱۴ تیر ۱۴۰۱

فهرست مطالب

فصل ۱

مقدمه

شبکه عصبی پیچشی تکه‌ای[1] در ابتدا در سال ۲۰۱۵ توسط ژنگ و همکاران در کنفرانس ACL ارائه شد [۵]. این شبکه با هدف استخراج روابط بین دو موجودیت در متن ارائه شد. در سال‌های بعد هم «استخراج رابطه بین موجودیت‌ها» هدف بیشتر پژوهش‌هایی بود که از این شبکه عصبی استفاده کردند.

با توجه تنیدگی شبکه عصبی پیچشی تکه‌ای و استخراج رابطه نیاز است تا توضیحاتی در رابطه با شیوه استخراج رابطه بین موجودیت‌ها در متن داده شود. در استخراج رابطه موجودیت‌ها از روی جمله تلاش می‌شود با داشتن موجودیت‌ها و متن جمله، رابطه‌ای که جمله بین آن موجودیت‌ها بیان می‌کند، استخراج شود. برای روشن شدن مطلب جمله «حافظ در شیراز درگذشت.» را در نظر بگیرید. این جمله شامل دو موجودیت «حافظ» و «شیراز» بوده و رابطه بیان شده توسط این جمله برای این دو موجودیت «محل فوت» است. هدف این پژوهش‌ها نیز استخراج رابطه «محل فوت» از روی متن جمله و موجودیت‌های آن است.

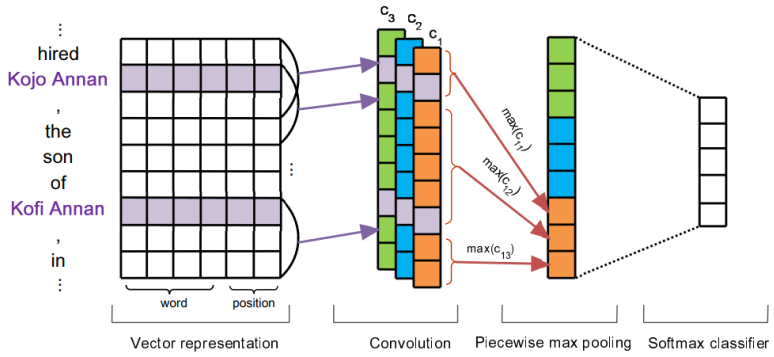
گرچه استخراج رابطه از متن در نگاه اول ساده به نظر می‌رسد اما این پژوهش‌ها با چالش‌هایی نیز مواجهند. در مثال قبلی جمله رابطه «محل فوت» را بین دو موجودیت «حافظ» و «شیراز» بیان می‌کرد اما جمله «حافظ در شیراز متولد شد» رابطه «محل تولد» را بین دو موجودیت بیان می‌کند. همان‌طور که می‌بینید با اندکی تغییر در جمله معنای جمله کاملا می‌تواند متفاوت شود و رابطه دیگری را بین موجودیت‌ها بیان کند.

برای فراهم کردن داده‌های آموزشی برای استخراج رابطه در گذشته از روش‌های بانظارت استفاده می‌شد. اما با توجه به سرآیند زیاد این روش‌ها مینتز در سال ۲۰۰۹ ایده نظارت‌ازراه‌دور را برای جمع‌آوری داده برای وظیفه استخراج رابطه از روی جمله را مطرح کرد [۶]. در این ایده از یک پایگاه دانش که شامل موجودیت‌ها و رابطه بین آن‌ها بود برای جمع‌آوری داده‌های آموزشی از سطح وب استفاده می‌شود. بدین طریق که هر جمله‌ای که شامل هر دو موجودیت بود به عنوان نمونه‌ای از رابطه‌ی بیان شده بین دو جمله در نظر گرفته می‌شود. البته همان‌طور که مشخص است این ایده ممکن است جملاتی را نیز که رابطه دیگری را بیان می‌کنند را به عنوان یک نمونه آموزشی برای رابطه مدنظر جمع‌آوری کند، مثلا این ایده هر دو مثال ارائه شده در بالا را به عنوان «محل فوت» یا «محل زندگی» در نظر می‌گیرد. اما از آن جا که حجم داده‌های آموزشی جمع‌آوری شده در این حالت بسیار زیاد است در مصالحه بین حجم داده برچسب خورده و درستی برچسب‌ها این روش بهتر عمل می‌کند.

حال که زمینه پژوهشی استخراج رابطه از متن بررسی شد، تمرکز خود را روی شبکه عصبی PCNN و تلاش‌هایی که برای بهبود آن انجام شده است می‌دهیم. در ابتدا شیوه عمل شبکه عصبی پیچشی تکه‌ای را مورد بررسی قرار می‌دهیم.

در شکل ۱.۱ ساختار شبکه عصبی پیچشی تکه‌ای مشاهده می‌شود. با ورود نمایش برداری کلمات به لایه‌های کانوولوشنی، این لایه‌ها به صورت سطری روی نمایش برداری پیمایش کرده و ویژگی‌های جمله را استخراج می‌کنند. در ادامه هر بردار ویژگی استخراج شده توسط لایه کانوولوشنی به سه قسمت تقسیم می‌شود: قسمت قبل از موجودیت اول، قسمت مابین موجودیت اول و دوم و قسمت بعد از موجودیت دوم. بیشینه هر یک از این قسمت‌ها محاسبه شده و به صورت یک بردار سه‌تایی در می‌آید. بردار نهایی بازنمایی جمله با ترکیب بردارهای سه‌تایی متناظر هر یک از لایه‌های کانوولوشنی حاصل می‌شود. در قدم بعد از این بازنمایی برای کاربرد مدنظر استفاده می‌شود. به عبارتی شبکه PCNN یک شبکه کدگذار است که با دریافت یک بازنمایی با طول متغیر آن را به یک بازنمایی با طول ثابت نگاشت می‌کند. استخراج ویژگی‌های خوب به همراه ارائه بردار با طول ثابت برای بازنمایی‌های با

[1] Piecewise Convolutional Neural Network(PCNN)



شکل ۱.۱: شبکه عصبی پیچشی تکه‌ای

طول متغیر باعث محبوبیت شبکه‌های عصبی تکه‌ای شده است.

برای بهبود این شبکه‌های عصبی راهکارهای مختلفی ارائه شده است. یکی از راهکارها مدل ارائه شده توسط پنگ و همکاران [۷] است. آن‌ها پیشنهاد کردند برای آن که شبکه عصبی PCNN بتواند وابستگی‌های با فاصله زیاد را در جمله بهتر بازنمایی کند، به جای استفاده از کانوولوشن‌های پیوسته از کانوولوشن‌های دراز[1] استفاده شود. این لایه‌های کانوولوشن به جای آن که خروجی را از روی بردارهای نزدیک به هم محاسبه کند از بردارهای با فاصله مکانی مشخص استفاده می‌کند.

شیوه دیگری که برای بهبود این شبکه‌ها استفاده شده است، ترکیب شبکه‌های عصبی بازگشتی با این شبکه‌هاست. این ایده اولین بار توسط یان و هو [۳] ارائه شد. بازنمایی ارائه شده توسط شبکه عصبی PCNN برای هر بخش بازنمایی مستقلی را ارائه می‌دهد، بنابراین آن‌ها قصد داشتند با عبور بازنمایی تولید شده از شبکه عصبی حافظه کوتاه‌مدت بلند[2] از این استقلال بکاهند.

اما بیشتر پژوهش‌ها تلاش کرده‌اند عملکرد شبکه‌های پیچشی تکه‌ای را با ارائه بازنمایی بهتر از ورودی‌ها بهبود ببخشند. اکثر این پژوهش‌ها تلاش دارند که با وزن‌دهی به اجزای مهم در ورودی بازنمایی بهتری را با استفاده از شبکه‌های پیچشی تکه‌ای تولید کنند [۸]، [۹]، [۱۰]، [۱۱] [۲]، [۱]. تحقیقات دیگر ایده‌های دیگری برای بهبود عملکرد این شبکه عصبی پیشنهاد کردند. برای مثلا سانگها نام[3] و همکاران [۱۲] تلاش کردند بازنمایی بهتری را در ورودی شبکه‌های عصبی پیچشی تکه‌ای برای کلماتی که چندمعنا دارند ارائه دهند. روش پیشنهادی آن‌ها برای این کار استفاده از یک ماژول ابهام‌زدایی از کلمات بود. در پژوهش دیگری که در سال ۲۰۱۹ انجام شد تلاش شد با استفاده سلسله‌مراتبی از مدل PCNN نتیجه بهتری کسب شود [۱۳].

بعضی دیگر نیز ایده بیان شده توسط شبکه‌های پیچشی تکه‌ای را به شکل دیگری استفاده کرده‌اند. در شبکه عصبی پیچشی تکه‌ای ابتدا کانوولوشن روی ورودی اعمال شده و سپس خروجی به چند تکه تقسیم می‌شود اما در مقاله ارائه شده توسط لیو[4] و همکاران پیشنهاد شد ابتدا ورودی تکه‌تکه شود و سپس عمل کانوولوشن روی هر قسمت انجام شود [۴].

به عنوان سخن آخر می‌خواهیم به کاربردهای شبکه عصبی PCNN در سایر حوزه‌ها

[1] dilated
[2] Long short-term memory
[3] Sangha Nam
[4] Liu

اشاره کنیم. برای این کار می‌توان از پژوهش‌های انجام شده توسط دو[1] و ژنگ[2] نام برد. هر دو این پژوهش‌ها تلاش کرده‌اند با استفاده از شبکه‌های عصبی پیچشی تکه‌ای به نتایج بهتری در حوزه تحلیل منظور[3] انجام دهد. دو از شبکه PCNN به عنوان یک کدگذار استفاده می‌کند و برای افزایش تعداد نمونه‌های آموزشی از شبکه‌های مولد تقابلی[4] بهره می‌گیرد [۱۴]. ژنگ نیز مشابه دو به منظور کدگذاری جملات ورودی از شبکه‌های عصبی پیچشی تکه‌ای استفاده می‌کند [۱۵].

در بخش‌های بعدی با جزئیات برخی از کارهایی که در زمینه شبکه‌های عصبی پیچشی تکه‌ای شاخص هستند، را بررسی خواهیم کرد. در انتها نیز خلاصه‌ای از مطالب ارائه شده و لیست مراجع استفاده شده را خواهیم داشت.

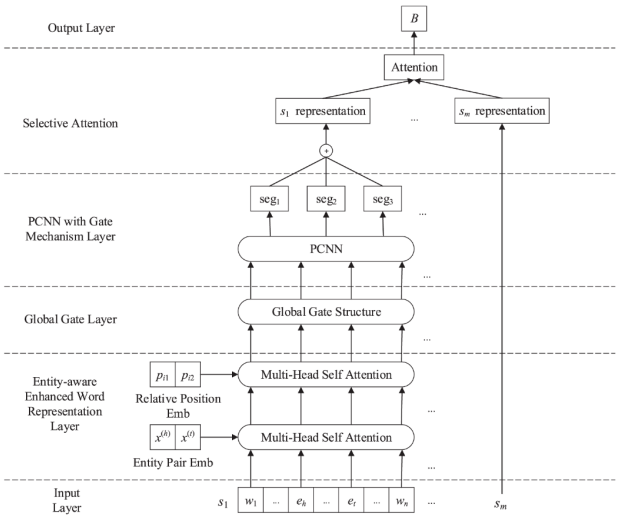
[1] Du
[2] Zhang
[3] sentiment analysis
[4] Generative Adversarial Network

فصل ۲

شرح مقالات

۱.۲ استخراج رابطه با تکنیک نظارت از راه دور با استفاده از دروازه در شبکه عصبی پیچشی تکه‌ای با تاکید بر موجودیت‌ها [۱]

محتوای جمله و به خصوص موجودیت‌ها تاثیر زیادی در معنای برداشت شده از جمله و کلمه دارند. هایخو ون[۱]، شین‌هوآ ژو[۲]، لانفانگ ژنگ[۳] و فی لی[۴] تلاش کردند با استفاده از مکانیزم توجه به خود[۵] بهتر بتوانند محتوای جمله را در معنای کلمه دخیل کنند. آن‌ها برای انجام این کار از شبکه عصبی پیچشی تکه‌ای و مکانیزم دروازه[۶] استفاده کردند.



شکل ۱.۲: معماری شبکه مقاله استخراج رابطه با تکنیک نظارت از راه دور با استفاده از دروازه در شبکه عصبی پیچشی تکه‌ای با تاکید بر موجودیت‌ها

شکل ۱.۲ خلاصه کار انجام شده توسط ون و همکاران را نشان می‌دهد. در این شبکه ابتدا هر کلمه با استفاده از مدل word2vec به بردار تبدیل می‌شود. در قدم بعدی ترکیب بردار موجودیت‌ها با بردار کلمه به لایه توجه به خود داده می‌شود تا بازنمایی بهتری برای کلمه با تاکید بر بازنمایی موجودیت‌ها تولید شود. به عبارتی اگر بردار کلمه iام را با x_i و بردار موجودیت اول را با e^h و بردار موجودیت دوم را با e^t نمایش دهیم، در این صورت بردار ورودی به لایه توجه به خود برابر خواهد بود با

$$[x_i, e^h, e^t] \tag{۱.۲}$$

[1] Haixu Wen
[2] Xinhua Zhu
[3] Lanfang Zhang
[4] Fei Li
[5] self attension
[6] gate

فرض کنیم خروجی لایه توجه به خود در قدم قبلی بردار x_i^h باشد. برای پررنگ‌تر کردن همبستگی بین هر کلمه و موجودیت، بردار x_i^h با فاصله مکانی نسبی کلمه تا هر یک از موجودیت‌ها، که آن را با $p_{i,1}$ و $p_{i,2}$ نشان می‌دهند، ترکیب شده و حاصل مجددا به لایه توجه به خود داده می‌شود. البته این لایه توجه به خود متفاوت از لایه توجه به خود توضیح داده شده در پاراگراف قبلی است. به عبارتی ورودی لایه توجه به خود در این حالت برابر خواهد بود با

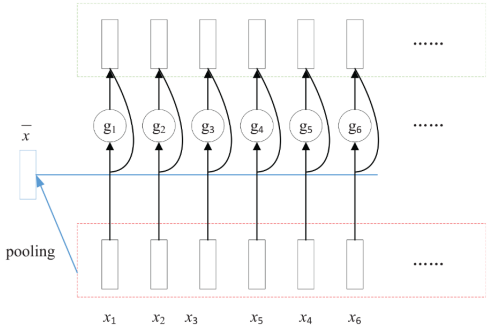
$$[x_i^h, p_{i,1}, p_{i,2}] \tag{۲.۲}$$

خروجی قسمت قبل برای هر کلمه را با علامت x_i^{ep} نمایش می‌دهیم. در گام بعدی بردار حاصل شده به لایه دروازه سراسری[1] داده می‌شود. ساختار این بخش در شکل ۲.۲ مشاهده می‌شود. در لایه دروازه سراسری ابتدا بردار میانگین یک جمله بر اساس بردارهای x_i^{ep} محاسبه می‌شود. برای محاسبه میزان همبستگی بردار میانگین با هر یک از بردارهای کلمات، حاصل ضرب نقطه‌ای بین بردار میانگین و بردار کلمه (x_i^{ep}) محاسبه می‌شود. در ادامه بردار حاصل شده از حاصل‌ضرب نقطه‌ای با عبور از یک لایه متراکم حالت ضریب به خود پیدا می‌کند. بردارهای بازنمایی نهایی با ضرب این مقادیر ضریب در هر یک از بردارها محاسبه می‌شود. به بیان ریاضی

$$\bar{x} = \frac{1}{n}\sum_i^n x_i^{ep} \tag{۳.۲}$$

$$g_i = \sigma(W^g(x_i^{ep} \odot \bar{x}) + b^g) \tag{۴.۲}$$

$$x_i^g = x_i^{ep} \odot g_i \tag{۵.۲}$$



شکل ۲.۲: ساختار قسمت دروازه سراسری

بردارهای حاصل شده از لایه دروازه سراسری (x_i^g) به شبکه PCNN داده می‌شود. شبکه PCNN برای سه قسمت جمله بازنمایی متفاوتی را ارائه می‌کند. این بازنمایی‌ها در ادامه مشابه دروازه سراسری از یک لایه متراکم عبور داده شده و وزن‌های حاصل شده در آن‌ها ضرب می‌شود. به عبارت ریاضی اگر فرض کنیم خروجی شبکه PCNN برابر $[q_{i,1}, q_{i,2}, q_{i,3}]$ باشد، در این صورت خواهیم داشت:

[1] global gate

$$g_{i,seg} = \sigma(W^s q_{i,seg} + b^s) \quad (۶.۲)$$
$$P_{i,seg} = g_{i,seg} \odot q_{i,seg} \quad (۷.۲)$$
$$s^i = \tanh([P_{i,1}; P_{i,2}, P_{i,3}]) \quad (۸.۲)$$

بردار s بردار بازنمایی نهایی از یک جمله در این روش است. در این مقاله برای نادیده گرفتن جملات نویزی از وزن‌دهی جملات یک دسته استفاده می‌شود. بیان ریاضی این قسمت به صورت زیر انجام می‌شود.

$$B = \sum_i \alpha_i s_i \quad (۹.۲)$$
$$\alpha_i = \frac{\exp(s_i A v_r)}{\sum_j \exp(s_j A v_r)} \quad (۱۰.۲)$$

بردار B برای آموزش مدل و تعیین برچسب نمونه در هنگام آزمون استفاده می‌شود.

۲.۲ استخراج رابطه با روش نظارت از راه دور با استفاده از شبکه‌های عصبی تکه‌ای با توجه مکانی و توجه به دسته‌های مشابه[۲]

در این مقاله برای بهبود استخراج روابط از جمله‌ها از دو شبکه مجزا استفاده شده است. یکی از این شبکه‌ها در سطح جمله عمل کرده و دیگری وظیفه تعیین یافتن ویژگی بین جملات یک دسته است. شبکه عصبی توجه مکانی از مدل PCNN برای کدگذاری جملات استفاده کرده و برای ارائه کدگذاری بهتر روش جدیدی را برای توجه به مکان قرارگیری کلمات پیشنهاد می‌دهد. روش دوم نیز برای رفع استخراج ویژگی از دسته‌هایی که تعداد جملات اندکی دارند ارائه شده است. در ادامه با جزئیات بیشتر با ساختار هر کدام از این شبکه‌ها آشنا می‌شویم.

۱.۲.۲ شبکه عصبی توجه مکانی

همان‌طور که بیان شد هدف از شبکه عصبی اول ایجاد یک بازنمایی بهتر برای جمله است و برای ایجاد این بازنمایی از شبکه PCNN استفاده می‌شود. برای آن که PCNN بتواند بازنمایی دقیق‌تری را ارائه دهد پیشنهاد شده است که به کلماتی که به موجودیت‌های جمله نزدیک‌تر هستند اهمیت بیشتری داده شود. شمای کلی این شبکه در شکل ۳.۲ آورده شده است.

شیوه انجام این وزن‌دهی به این صورت است که ابتدا فاصله هر کلمه تا هر یک از موجودیت‌ها محاسبه می‌شود. با انجام این کار برای هر کلمه دو عدد d_1 و d_2 به دست می‌آید که d_1 فاصله کلمه تا موجودیت اول و d_2 فاصله کلمه تا موجودیت دوم است. حال این فاصله‌ها در فرمول تابع چگالی احتمال گاوس با $\mu = 0, \sigma = 0.5$ قرار داده می‌شود تا مقادیر کوچک‌تر به اعداد بزرگ‌تر و مقادیر بزرگ‌تر به اعداد کوچک‌تری تبدیل شوند. با این تبدیل مقدار d_1 به عدد G_1 و عدد d_2 به G_2 تبدیل می‌شود.

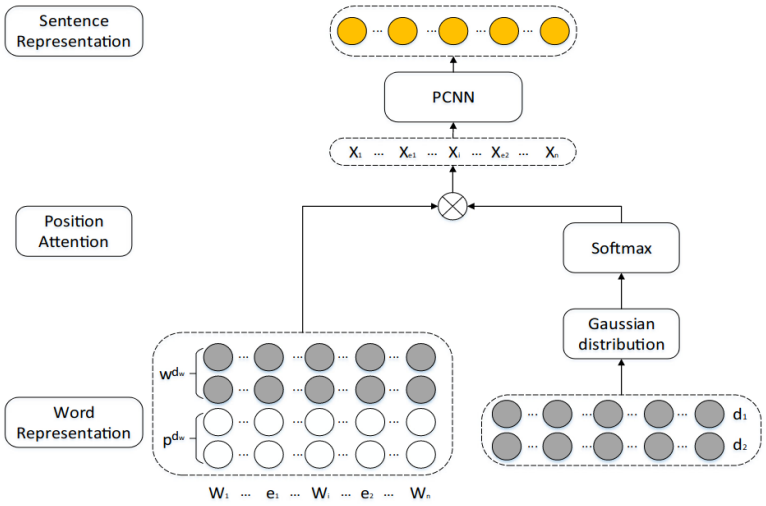
در قدم بعدی برای هر کلمه مقدار $G_1 + G_2$ را محاسبه کرده و حاصل را به تابع softmax می‌دهند. این تابع هر یک از مقادیر را به بازه $[0, 1]$ نگاشت می‌کند. از این مقادیر برای وزن‌دهی بازنمایی کلمات استفاده می‌شود. بردارهای وزن‌دهی شده برای استخراج ویژگی‌های بیشتر به شبکه PCNN داده می‌شود.

۲.۲.۲ شبکه توجه به دسته‌های با ویژگی‌های مشابه

مجموعه داده‌ای که برای این پژوهش استفاده شده است، شامل ۵۳ رابطه مختلف است. اما بیشتر برای بیشتر این روابط تنها یک نمونه وجود دارد. مشخص است که برای چنین دسته‌هایی شبکه قادر نخواهد بود ویژگی‌های مناسبی استخراج کند. روشی که در این پژوهش ارائه شده است ادغام ویژگی‌های مشابه از دسته‌های دیگر در این شبکه است.

راهکار ارائه شده به این صورت عمل می‌کند که ابتدا شباهت ویژگی‌های استخراج شده برای دسته فعلی را با ویژگی‌های تمام دسته‌های دیگر از طریق رابطه‌ی ریاضی زیر محاسبه می‌کنند.

$$\text{sim}(\text{Bag}_i, \text{Bag}_j) = \text{Bag}_i \text{Bag}_j^T \quad (۱۱.۲)$$



شکل ۳.۲: مدل بازنمایی جمله

سپس n تا از شبیه‌ترین دسته‌ها انتخاب می‌شود. در قدم بعدی ویژگی‌ها وزن‌دهی شده و بر اساس وزن‌ها ترکیب می‌شوند. وزن‌های این با استفاده از فرمول زیر استخراج می‌شود.

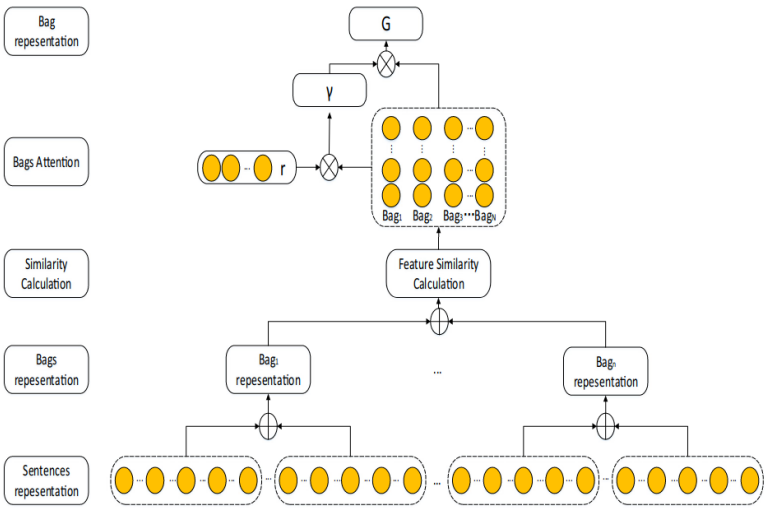
$$\gamma_i = \frac{\exp(e_i)}{\sum_k^N \exp(e_k)} \quad (۱۲.۲)$$

$$\exp(e_i) = \text{Group}_i^j Br \quad (۱۳.۲)$$

در فرمول بالا B یک ماتریس قطری وزن و r یک بازنمایی برداری از رابطه است. پس از محاسبه γ_iها ویژگی‌ها به صورت زیر با هم ترکیب می‌شوند.

$$G_i = \sum_i^N \gamma_i \text{Group}_i^j \quad (۱۴.۲)$$

برچسب نهایی با استفاده از G_iها تعیین می‌شود.



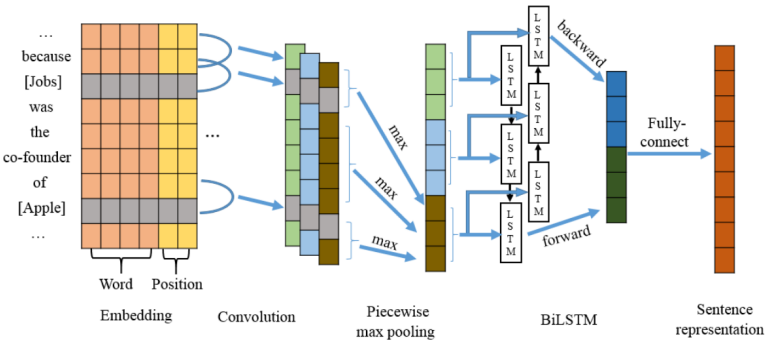
شکل ۴.۲: ساختار مدل توجه به دسته‌های با ویژگی‌های مشابه

۳.۲ استخراج رابطه با استفاده از مولد بازنمایی مشترک و شبکه‌های عصبی پیچشی کوتاه‌مدت بلند تکه‌ای [۳]

در این مقاله که در سال ۲۰۱۸ توسط دانفنگ یان[۱] و بو هو[۲] عرضه شده است، از یک شبکه عصبی مولد برای ایجاد بازنمایی رابطه استفاده شده است. این شبکه مولد برای ایجاد بازنمایی رابطه از بازنمایی جملات آن رابطه کمک می‌گیرد. بازنمایی جمله نیز با استفاده از شبکه‌های عصبی پیچشی کوتاه‌مدت بلند تکه‌ای[۳] انجام می‌شود.

۱.۳.۲ شبکه‌های عصبی کدگذار

در روش‌های ارائه شده پیش از این مقاله، کدگذاری خام ارائه شده توسط PCNN در قسمت‌های بعدی برای تعیین رابطه بیان شده در جمله استفاده می‌شد. در این مقاله اما پس از کدگذاری جمله توسط شبکه PCNN این کدگذاری‌ها مطابق شکل ۵.۲ به یک شبکه BiLSTM داده می‌شود تا ارتباط موجود بین قسمت‌های مختلف جمله در بازنمایی ارائه شده نهایی تاثیرگذار باشد. کدگذاری تولید شده توسط این شبکه در شبکه مولد که در بخش بعدی خواهیم دید، به کار گرفته می‌شود.



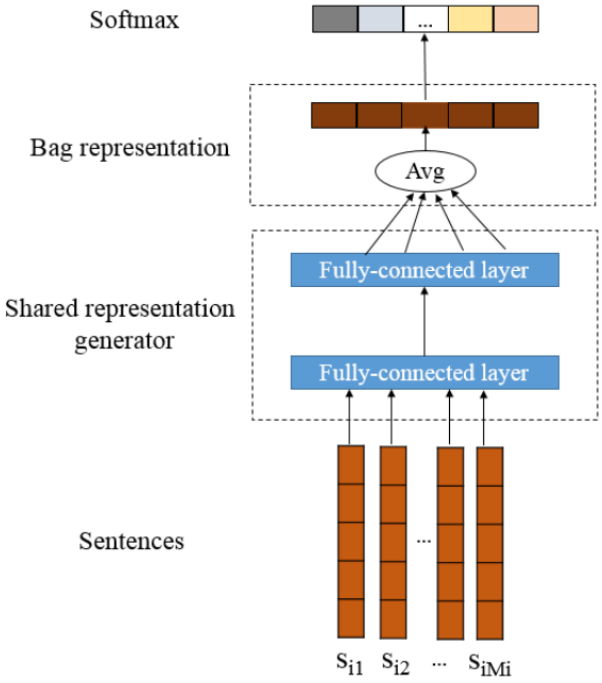
شکل ۵.۲: ترکیب شبکه عصبی PCNN و LSTM

۲.۳.۲ شبکه تولیدکننده بازنمایی مشترک

یان و هو شبکه عصبی موجود در شکل ۶.۲ را برای ایجاد بازنمایی یک دسته[۴] از کلمات ارائه کردند. آن‌ها بر خلاف پژوهش‌های دیگر به جای آن که از مجموع وزن‌دار بازنمایی هر جمله ($s_{(i,r)}$) برای تولید بازنمایی مشترک استفاده کنند، پیشنهاد یک مولد برای تولید بازنمایی جملات یک رابطه را مطرح کردند. این مدل مولد برای تولید بازنمایی رابطه از بازنمایی تولید شده برای جملات متناظر رابطه استفاده می‌کند. برای آموزش این شبکه تابع خطایی نیز معرفی شده است که ترکیبی از تابع خطای آنتروپی متقابل[۵] و میانگین فاصله است. در

[1] DANFENG YAN
[2] BO HU
[3] Piecewise-LSTM Convolutional Neural Network
[4] bag
[5] cross entropy

ادامه با جزئیات بیشتر این شبکه عصبی و تابع خطای معرفی شده آشنا می‌شویم.
شبکه عصبی مولد برای ایجاد بازنمایی رابطه از بازنمایی‌های جمله که توسط مدل کدگذار PCNN تولید شده استفاده می‌کند. بازنمایی جمله ارائه شده توسط شبکه PCNN به شبکه عصبی دولایه داده می‌شود. شبکه عصبی دولایه بازنمایی جمله iام در رابطه rام را از فضای $s_{(i,r)}$ به فضای $g_{(i,r)}$ می‌برد. هدف از این تبدیل ایجاد یک بازنمایی بهتر با تاکید بیشتر روی معنا از بازنمایی جمله است. در مدل پیشنهادی آن‌ها برای هر رابطه شبکه مولد جداگانه‌ای در نظر گرفته شده است که از شبکه مولد دیگر مستقل است.



شکل ۶.۲: شبکه عصبی تولیدکننده بازنمایی مشترک

پس از ایجاد بردارهای $g_{(i,r)}$، میانگین بدون وزن این بردارها محاسبه می‌شود. بردار حاصل شده G_r را بردار بازنمایی آن رابطه می‌نامند.

$$G_r = \frac{1}{M} \sum_{i=1}^{M} g_{(i,r)}$$

از آن‌جا که آموزش مدل روی نمونه‌ها به صورت دسته‌ای بوده است، بنابراین در هنگام آزمایش مدل داده‌ها به صورت دسته‌ای به مدل مولد داده شده و رابطه کلی بیان شده توسط آن دسته با استفاده از یک لایه بیشینه‌گیری نرم[1] مشخص می‌شود.
حال که جزئیات شبکه مولد بیان شد، جزئیات تابع خطای متناظر آن تشریح می‌شود. تابع خطای این شبکه دو هدف را دنبال می‌کند:

[1] softmax

۱. بازنمایی ارائه شده برای یک رابطه تا جای امکان دور از بازنمایی رابطه‌های دیگر باشد.

۲. بازنمایی ارائه شده تا جای امکان مشابه بازنمایی ارائه شده برای هر جمله آن رابطه باشد.

برای رسیدن به هدف اول، از تابع خطای آنتروپی متقابل استفاده می‌شود. بدین ترتیب که انتظار می‌رود لایه بیشینه‌گیری نرم بتواند برچسب دسته را به درستی تعیین کند. به عبارت ریاضی حاصل عبارت زیر باید بیشینه شود.

$$J_1(\theta) = \sum_{i=1}^{T} \log p(r_i|G_i, \theta) \tag{۱۵.۲}$$

برای رسیدن به هدف دوم نیاز است که فاصله بازنمایی تولید شده از هر جمله متناظر آن رابطه کمینه باشد. برای رسیدن به این هدف تابع خطای $J_2(\theta)$ به صورت زیر محاسبه می‌شود.

$$J_2(\theta) = \frac{1}{T}\sum_{i=1}^{T}(\frac{1}{M_i}\sum_{j=1}^{M_i}(g_{(j,i)} - c_i)^2) \tag{۱۶.۲}$$

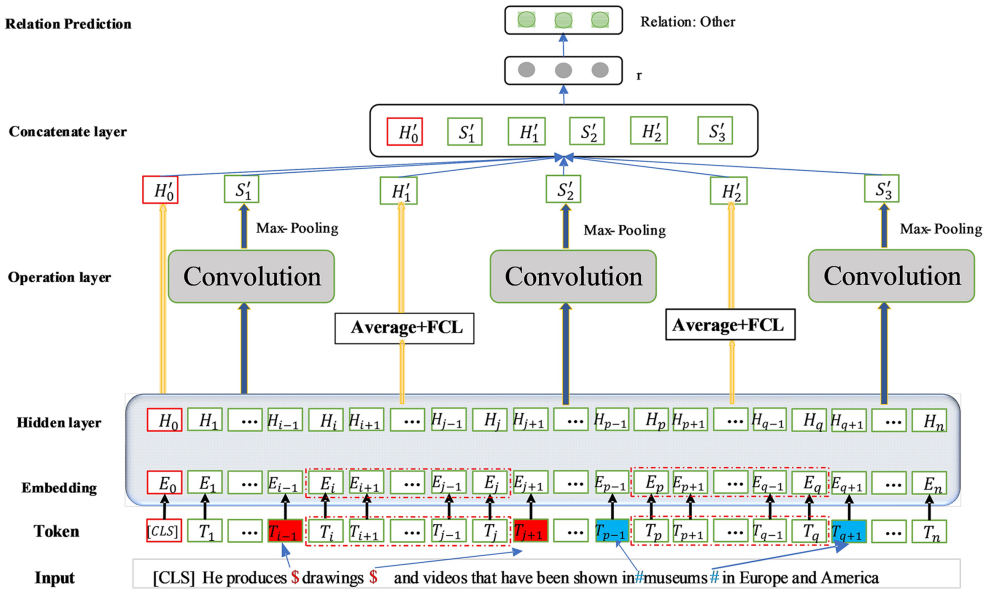
$$c_i = \frac{1}{M_i}\sum_{j=1}^{M_i} g_{(j,i)} \tag{۱۷.۲}$$

نکته عجیبی که در تابع خطای بالا وجود دارد استفاده از بازنمایی $g_{(i,r)}$ به جای $s_{(i,r)}$ است. چرا که G_r میانگین **بدون وزن** از $g_{(i,r)}$ است بنابراین محاسبه فاصله آن تا هر یک از $g_{(i,r)}$ به نظر نمی‌رسد که پارامتری را تحت تاثیر قرار دهد. اما اگر این مقدار از روی $s_{(i,r)}$ از روی محاسبه می‌شد با توجه به این که $g_{(i,r)}$ از یک شبکه عصبی محاسبه می‌شود بنابراین پارامترهای این شبکه را تحت تاثیر قرار می‌داد.
در نهایت تابع خطای کل به صورت زیر محاسبه می‌شود.

$$J(\theta) = J_1(\theta) + \alpha J_2(\theta) \tag{۱۸.۲}$$

۴.۲ دسته‌بندی رابطه با بهره‌گیری از مدل BERT و کانولوشن‌های تکه‌ای به همراه خطای کسری [۴]

در سال ۲۰۲۱ لیو[1] و همکارانش رویکرد متفاوتی را در رابطه با شبکه‌های عصبی پیچشی تکه‌ای در پیش گرفتند. آن‌ها به جای این که یک کانولوشن را روی شبکه انجام داده و خروجی آن را سه قسمت کنند سه کانولوشن متفاوت را روی سه قسمت جمله اجرا کردند. با ترکیب خروجی‌های این سه قسمت رابطه‌ای که جمله بیان می‌کند تشخیص داده می‌شود.



شکل ۷.۲: ساختار مدل پیشنهادی

مدل پیشنهادی لیو و همکارانش در شکل ۷.۲ دیده می‌شود. مدل پیشنهادی آن‌ها انتظار دارد موجودیت‌های موجود در جمله با علامت مخصوصی نظیر $ و # مشخص شده باشد. با دریافت جمله، توکن‌های [CLS] و [SEP] به جمله اضافه شده و جمله به مدل BERT داده می‌شود. پس از دریافت بازنمایی کلمات (E_i) از مدل BERT هر بازنمایی از یک لایه Dense عبور کرده و بازنمایی دیگری به نام H_i را تولید می‌کند. در این مرحله بازنمایی توکن‌ها به شش دسته تقسیم شده و برای هر بخش روند متفاوتی طی می‌شود.

- **توکن آغاز جمله BERT**: بر روی بازنمایی این توکن عملیات خاصی انجام نشده و مستقیما به خروجی منتقل می‌شود.
- **کلمات قبل از موجودیت اول**: بازنمایی متناظر این کلمات (H_i) ها از یک لایه کانولوشن عبور کرده و در قدم بعدی max pooling گرفته می‌شود. به عبارت ریاضی

$$X_{i:i+h-1} = H_i \oplus H_{i+1} \oplus H_{i+2} \oplus ... \oplus H_{i+h_1} \quad (۱۹.۲)$$

$$c_i = W * X_{i:i+h-1} + b \quad (۲۰.۲)$$

[1] Liu

در عبارت بالا منظور از عملگر $\oplus$ عملگر concatenation منظور از عملگر $*$ کانولوشن است. بنابراین c_i یک عدد بردار در فضای $\mathbb{R}^d$ است. با اعمال این کانولوشن با پنجره h روی بازنمایی کلمات ماتریس C به شکل زیر تولید می‌شود. ماتریس C در فضای $\mathbb{R}^{(n-h+1)\times d}$ است.

$$C = [c_1, c_2, ..., c_{n-h+1}] \tag{۲۱.۲}$$

حال بر روی این ماتریس عملیات max pooling انجام می‌شود. هدف از انجام این عملیات علاوه بر کاهش ابعاد به فضای $\mathbb{R}^{n-h+1}$، انتخاب مهم‌ترین ویژگی از هر یک از c_iهاست.

$$s = \max(C) \tag{۲۲.۲}$$

بردار s به مرحله بعدی منتقل می‌شود.

- **موجودیت اول:** از آن جا که ممکن است موجودیت اول خود شامل چند کلمه و در نتیجه دارای بازنمایی چندبعدی باشد بنابراین ابتدا میانگین این بازنمایی را پیدا کرده و به عنوان بازنمایی موجودیت اول در نظر می‌گیریم. با عبور این بازنمایی از یک لایه Dense مقدار H'_1 تولید می‌شود. به عبارت ریاضی

$$H'_1 = W_1(\tanh(\frac{1}{j-i+1}\sum_{t=i}^{j} H_t)) + b_1 \tag{۲۳.۲}$$

- **کلمات بین دو موجودیت:** برای این کلمات علمیات مشابهی مانند عملیات انجام شده بر روی کلمات قبل از موجودیت اول انجام می‌شود.
- **موجودیت دوم:** در این حالت نیز عملیات مشابهی نظیر عملیات انجام شده روی موجودیت اول انجام می‌شود.
- **کلمات بعد از موجودیت دوم:** عملیات انجام شده در این قسمت مشابه عملیات انجام شده روی کلمات قبل از موجودیت اول است.

پس از استخراج بردارهای متناظر هر قسمت یعنی H'_0، H'_1، H'_2، S'_1، S'_2 و S'_3 این بردارها با هم ترکیب شده و به یک لایه Dense با تابع فعال‌سازی softmax داده می‌شوند. بیان ریاضی این عملیات به شرح زیر است.

$$r = W_r(H'_0 \oplus S'_1 \oplus H'_1 \oplus S'_2 \oplus H'_2 \oplus S'_3) + b_r \tag{۲۴.۲}$$
$$\tilde{y} = \sigma(r) \tag{۲۵.۲}$$

در روابط بالا منظور از σ تابع softmax است. این تابع برای هر دسته یک احتمال نسبت می‌دهد. دسته‌ای که بیشترین احتمال را داشته باشد به عنوان خروجی نهایی مدل در نظر گرفته می‌شود.

۱.۴.۲ تابع خطا

بیشتر پژوهش‌هایی که در زمینه استخراج رابطه انجام شده‌اند از تابع خطای آنتروپی متقابل[1] که به شرح زیر تعریف می‌شود برای محاسبه خطای مدل استفاده کرده‌اند. مشکلی که این تابع دارد این است که نسبت به کلاس‌های با تعداد نمونه زیاد (و در نتیجه آسان بودن یادگیری این کلاس‌ها) بایاس داشته و اعداد بالایی را برای آن‌ها گزارش می‌دهد.

$$J(\theta) = -\sum_i y_i \log(p_t) + (1 - y_i) \log(1 - p_t) \qquad (۲۶.۲)$$

برای رفع مشکل این مشکل لیو و همکاران از ایده‌ای که در سال ۲۰۱۷ توسط لین و همکارانش [۱۶] ارائه شد، استفاده کرده‌اند. تابع خطای معرفی شده توسط لین تابع خطای کسری[2] نامیده شده و به شرح زیر تعریف می‌شود.

$$FL(p_t) = -a_t(1 - p_t)^\gamma \log(p_t) \qquad (۲۷.۲)$$

در این فرمول α_t فاکتور وزن‌دهی نامیده شده و مقداری بین $[0, 1]$ دارد. γ نیز پارامتر توجه نامیده می‌شود. p_t نیز احتمالی است که مدل با آن احتمال درصد اطمینان را گزارش می‌دهد. برای روشن‌تر شدن مفهوم هر یک از این پارامترها از یک مثال استفاده می‌کنیم.
فرض کنید دو دسته داریم که فراوانی هر یک از تعداد کل داده‌ها به ترتیب 0.1 و 0.25 باشد. همچنین فرض کنید دو مدل مختلف نیز داریم. مدل اول روی دسته با فراوانی کم‌تر با اطمینان 0.8 و روی دسته با فراوانی بیشتر با اطمینان 0.85 درصد پیش‌بینی انجام می‌دهد. مدل دوم در دسته با فراوانی کم‌تر با اطمینان 0.6 و در دسته با فراوانی زیاد با اطمینان 0.95 درصد پیش‌بینی انجام می‌کند. اگر بخواهیم این عملکرد این دو مدل را با تابع خطای cross entropy نسبت به هم بسنجیم، خواهیم داشت.

$$FL(\text{model}_1) = -0.1 \times (1 - 0.8)^2 \times \log(0.8) - 0.25 \times (1 - 0.85)^2 \times \log(0.85) \simeq 0.002 \qquad (۲۸.۲)$$

$$FL(\text{model}_2) = -0.1 \times (1 - 0.6)^2 \times \log(0.6) - 0.25 \times (1 - 0.95)^2 \times \log(0.95) \simeq 0.008 \qquad (۲۹.۲)$$

همان‌طور که مشاهده می‌شود تابع خطای کسری برای مدل دوم عدد بزرگتری را گزارش می‌دهد که منطقی است. چرا که مدل دوم در زمانی که داده بیشتری داشته بهتر و زمانی که با کمبود داده مواجه بوده است عملکرد ضعیفی داشته است. در این مثال مقدار $\gamma = 2$ و مقدار α_t را نسبت داده‌های دسته به تعداد کل داده‌ها در نظر گرفتیم. در این مقاله نیز مقدار α_t به همین شکل انتخاب می‌شود.
حال که با چگونگی رفتار تابع خطا آشنا شدیم، چگونگی اعمال این ایده در تابع آنتروپی متقابل را مطالعه می‌کنیم. با اعمال این ایده تابع خطای آنتروپی متقابل به شکل زیر در می‌آید.

[1] cross entropy
[2] focal loss

$$J(\theta) = -\sum_i \alpha_t y_i (1-p_t)^\gamma \log(p_t) + \alpha_t (1-y_i) p_t^\gamma \log(1-p_t) \quad (۳۰.۲)$$

این تابع خطا به خوبی دسته‌های با برچسب کم را در نظر می‌گیرد. در این مقاله علاوه بر این کار یک مقدار منظم‌سازی نیز روی تابع اعمال شده و تابع خطا به شکل زیر حاصل می‌شود.

$$J(\theta) = -\sum_i \alpha_t y_i (1-p_t)^\gamma \log(p_t) + \alpha_t (1-y_i) p_t^\gamma \log(1-p_t) + w||\theta||^2 \quad (۳۱.۲)$$

وزن‌های مدل با استفاده از این تابع خطا به روز می‌شود.

فصل ۳

مجموعه داده

مقالات بررسی شده از مجموعه داده‌های مختلفی استفاده کرده‌اند که در این جا به صورت خلاصه جزئیات آن‌ها بررسی می‌شود.

۱.۳ مجموعه داده روزنامه نیویورک تایمز

مجموعه داده نیویورک تایمز [1] در سال ۲۰۱۰ توسط ریدال [۱۷] ارائه شد. این مجموعه داده که شامل ۵۳ رابطه مختلف است، یکی از مجموعه داده‌های پرکاربرد در حوزه استخراج رابطه بوده و توسط مقالات مختلف استفاده شده است. برای ساخت این مجموعه داده از روش نظارت از راه دور استفاده شده و جملات روزنامه نیویورک‌تایمز با استفاده از پایگاه دانش فری‌بیس[2] برچسب‌گذاری شده است. این مجموعه داده شامل ۵۲۲۶۱۱ جمله برای آموزش و ۱۷۲۴۴۸ جمله برای آزمون است.

۲.۳ مجموعه داده SemEval-2010

مجموعه داده SemEval-2010 نیز در سال ۲۰۱۰ ارائه شده است [۱۸]. این مجموعه داده نسبت به مجموعه داده نیویورک تایمز کوچک‌تر بوده و شامل ۱۹ رابطه مختلف است. این مجموعه داده شامل ۱۰۷۱۷ جمله است که از این تعداد ۸۰۰۰ جمله برای آموزش و باقی برای آزمون استفاده می‌شود.

۳.۳ مجموعه داده SemEval-2018

این مجموعه داده در سال ۲۰۱۸ با برچسب‌گذاری داده‌های گزارش حملات امینتی ارائه شد [۱۹]. تعداد جملات برچسب‌گذاری شده این مجموعه داده مشابه مجموعه داده SemEval-2010 بوده اما تعداد رابطه‌های آن نسبت به مجموعه داده SemEval-2010 بسیار کم‌تر است. این مجموعه داده شامل ۱۰۱۸۲ جمله است که از این تعداد ۸۹۱۹ جمله برای آموزش مدل و باقی ۱۲۶۳ جمله برای آزمون استفاده می‌شود. این مجموعه داده شامل ۴ رابطه مختلف است.

۴.۳ مجموعه داده UW

این مجموعه داده در سال ۲۰۱۶ توسط لیو[۲۰] ارائه شده است. این مجموعه داده شامل ۵ رابطه بوده اما تعداد جملاتی که برای هر رابطه ارائه کرده است در اندازه مجموعه داده نیویورک تایمز است. این مجموعه داده دارای حدود ۵۰۰ هزار جمله برای آموزش و حدود ۳۷۲۴ جمله برای آزمایش مدل است.

[1] NYT-dataset
[2] freebase

فصل ۴

نتایج

مقالاتی که در فصل‌های قبل‌تر معرفی شد بیشتر در سطح دسته[1] عمل می‌کنند. بدین معنی که در هنگام آموزش برای آن که نویز حاصل از برچسب‌زنی با روش نظارت از راه دور را کم کنند، تمامی جملاتی را که برچسب رابطه یکسان دارند و همچنین شامل دو موجودیت مد نظر هستند به صورت یکجا برای استخراج ویژگی‌ها استفاده می‌کنند.

مقالاتی که بر طبق روش بالا عمل می‌کنند از روش‌های مختلفی ارزیابی می‌شوند. برای ارزیابی این روش‌ها از روشی موسوم به نام «بسط»[2] استفاده می‌شود. در این روش نیمی از نمونه جملات متناظر هر رابطه را به عنوان داده آموزشی و نیم دیگر را به عنوان داده آزمون در نظر می‌گیرند. از معیارهای مختلفی نظیر منحنی دقت‌ـ‌بازیابی[3]، سطح زیر نمودار[4] و روشی به نام P@N ،برای بررسی عملکرد مدل روی داده‌های آزمون استفاده می‌کنند. علاوه بر این معیارها گاهی عامل انسانی نیز برای ارزیابی عملکرد مدل استفاده می‌شود.

در مقالاتی که بر اساس تک جمله کار می‌کنند یعنی هم در مرحله آموزش و هم در مرحله آزمون از تک جمله بهره می‌گیرند از معیارهایی نظیر دقت[5]، بازیابی[6] و F1 برای ارزیابی کار خود استفاده می‌کنند.

در ادامه نتایج مقالات را بر طبق معیارهای معرفی شده ارائه کرده و مقایسه می‌کنیم. در این جا برای سادگی، مدل‌های ارائه شده را با نام مخفف انگلیسی آن‌ها اسم می‌بریم. در ادامه اسم مخفف هر یک از مقالات معرفی می‌شود.

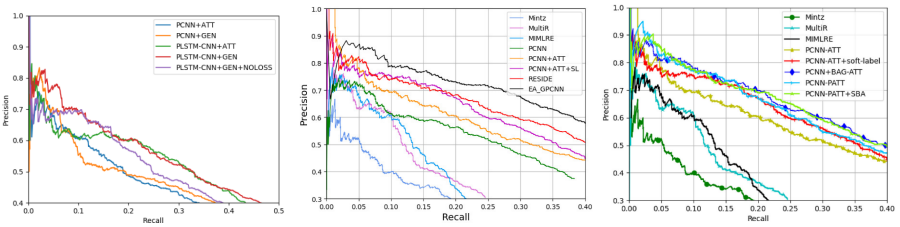
- EA-GPCNN: شبکه عصبی پیچشی تکه‌ای با تاکید بر موجودیت‌ها
- PCNN-PATT+SBA: شبکه عصبی پیچشی تکه‌ای با توجه مکانی و توجه به دسته‌های مشابه
- PLSTM-CNN: شبکه عصبی پیچشی کوتاه‌مدت بلند تکه‌ای
- BERT-PCNN: شبکه عصبی پیچشی تکه‌ای بر پایه مدل برت[7]

سه مقاله اول در هنگام آموزش بر اساس ویژگی‌هایی که از دسته استخراج کرده‌اند آموزش می‌بینند در حالی که روش آخر یعنی همواره BERT-PCNN در سطح جمله عمل می‌کند. بنابراین سه روش اول را به صورت جدای از روش آخر بررسی خواهیم کرد.

نمودار دقت‌ـ‌بازیابی برای سه روش EA-GPCNN، PCNN-PATT+SBA و PLSTM-CNN در شکل ۱.۴ مشاهده می‌شود. همان‌طور که مشاهده می‌شود متاسفانه هیچ یک از مقالات نتیجه خود را با نتیجه مقاله دیگر مقایسه نکرده است، بنابراین این روش‌ها را صرفا می‌توان از روی شکل ارائه شده با هم مقایسه کرد. بر اساس شکل‌ها عملکرد روش EA-GPCNN از دو روش دیگر بهتر است چرا که بر طبق نمودار به ازای مقادیر یکسان بازیابی نتایج بهتری برای دقت ارائه کرده است.

روش دیگری که این سه مدل را می‌توان بررسی کرد روشی به نام P@N است. در این روش دقت عملکرد مدل به ازای N نمونه آزمون بررسی می‌شود نتایج این بررسی در جدول ۱.۴ مشاهده می‌شود. همان‌طور که مشاهده می‌شود در این معیار ارزیابی نیز روش EA-GPCNN برتر از دو روش دیگر عمل کرده است.

[1] bag
[2] hold out
[3] precision-recall curve
[4] area under the curve
[5] Precision
[6] Recall
[7] BERT



(ج) عملکرد مدل PLSTM-CNN (ب) عملکرد مدل EA-GPCNN (آ) عملکرد مدل PCNN-PATT+SBA

شکل ۱.۴: بررسی عملکرد سه روش EA-GPCNN، PCNN-PATT+SBA و PLSTM-CNN در مجموعه داده نیویورک تایمز

Table 4.1: بررسی عملکرد سه روش EA-GPCNN، PCNN-PATT+SBA و PLSTM-CNN در مجموعه داده نیویورک تایمز بر معیار P@N

P@N	100	200	300	mean
EA-GCPNN	91	87.5	82.0	86.8
PLSTM-CNN	76.3	65.6	60.2	67.4
PCNN-PATT+SBA	86	81	76.7	81.2

بر طبق معیارهای ارزیابی بالا بهترین روش EA-GCPNN است. این روش بازنمایی هر کلمه را بر اساس فاصله آن از موجودیت وزن‌دهی می‌کند. در مقام بعدی روش PCNN-PATT+SBA است که از مکانیزم توجه استفاده کرده و دسته‌های با ویژگی‌های مشابه را در هم ادغام می‌کرد. در نهایت روش PLSTM-CNN است که از شبکه عصبی LSTM برای بهبود کدگذاری ارائه شده توسط PCNN بهره می‌برد. این نتایج نشان از اهمیت دخیل کردن بازنمایی موجودیت در بازنمایی سایر کلمات دارد چرا که روش‌هایی که به این بخش بیشتر اهمیت داده‌اند نتایج بهتری گرفته‌اند.

تا به این جا نتایج عملکرد سه مدل EA-GPCNN، PCNN-PATT+SBA و PLSTM-CNN را با هم مقایسه کردیم. در ادامه می‌خواهیم نتایج عملکرد مدل BERT-PCNN که رویکرد متفاوتی نسبت به روش‌های نامبرده دارد را بررسی می‌کنیم.

ارزیابی روش BERT-PCNN بر روی مجموعه داده‌های SemEval-2010 و SemEval-2018 انجام شده است. این مدل می‌تواند روی مجموعه داده SemEval-2010 به نتیجه 89.95 درصد روی معیار F1 برسد. عملکرد مدل روی مجموعه داده SemEval-2018 از این هم بهتر بوده و توانسته به نتیجه 99.52 روی معیار F1 برسد. به نظر می‌رسد که نتیجه بهتر روی مجموعه داده SemEval 2018 به خاطر خود مجموعه داده باشد چرا که در مقاله توضیحی در بابت اختلاف نتایج داده نشده است.

فصل ۵

جمع‌بندی

شبکه‌های کدگذار متفاوتی به منظورهای مختلف ارائه شده است. یکی از این شبکه‌های کدگذار که بیشتر به منظور کدگذاری جملت استفاده می‌شود شبکه عصبی پیچشی چندتکه‌ای است. مزیت این شبکه عصبی استخراج ویژگی‌های جمله و ارائه یک بردار با طول ثابت از جمله است.

در این گزارش مرور کوتاهی بر ایده‌های پیشنهادی با هدف بهبود شبکه‌های عصبی پیچشی تکه‌ای انجام شد. مختصرا کارهای انجام شده در دسته‌های کلی بررسی شده و سپس کلیدی‌ترین تحقیقات به صورت دقیق‌تر بررسی شد. با توجه به این که این شبکه اکثرا در حوزه استخراج رابطه استفاده می‌شود بنابراین مرور کوتاهی بر این حوزه نیز انجام شد.

با مقایسه‌ای که از نتایج حاصل مقالات مطالعه شده داشتیم مشخص شد که در روش‌هایی که پایه دسته‌ای از جملات رابطه را یاد می‌گیرند، روش ارائه شده ون و همکاران عملکرد بهتری نسبت به دو روش دیگر داشت. همچنین نتایج مدل لیو که در سطح جمله عمل می‌کرد نیز ارائه شد.

گرچه در حال حاضر شبکه پیچشی تکه‌ای در سایر حوزه‌ها کاربرد چندانی ندارد اما به نظر می‌رسد ایده مطرح شده توسط آن می‌تواند در حوزه‌های دیگر به تنهایی یا با ترکیب سایر ایده‌ها استفاده شود. برای مثال در حوزه تصویر می‌توان تصویر ورودی را به تکه‌هایی تقسیم کرده و پس از انجام عمل کانوولوشن ماکزیمم این تکه‌ها را برداشت. چنین کاری می‌تواند عمل تشخیص یک شی در تصویر را راحت‌تر بکند.

منابع و مراجع

[1] H. Wen, X. Zhu, L. Zhang, and F. Li, "A gated piecewise cnn with entity-aware enhancement for distantly supervised relation extraction," *Information Processing & Management*, vol.57, no.6, p.102373, 2020.

[2] W. Li, Q. Wang, J. Wu, and Z. Yu, "Piecewise convolutional neural networks with position attention and similar bag attention for distant supervision relation extraction," *Applied Intelligence*, vol.52, no.4, pp.4599–4609, 2022.

[3] D. Yan and B. Hu, "Shared representation generator for relation extraction with piecewise-lstm convolutional neural networks," *IEEE Access*, vol.7, pp.31672–31680, 2019.

[4] J. Liu, X. Duan, R. Zhang, Y. Sun, L. Guan, and B. Lin, "Relation classification via bert with piecewise convolution and focal loss," *PLOS ONE*, vol.16, no.9, pp.1–23, 2021.

[5] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp.1753–1762, Association for Computational Linguistics, 2015.

[6] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp.1003–1011, Association for Computational Linguistics, 2009.

[7] M. Peng, W. Hu, G. Tian, B. Wang, H. Wang, and G. Wang, "Dilated convolutional networks incorporating soft entity type constraints for distant supervised relation extraction," in *2019 International Joint Conference on Neural Networks (IJCNN)*, pp.1–7, 2019.

[8] V.-N. Nguyen, H.-T. Nguyen, D.-H. Vo, and L.-M. Nguyen, "Relation extraction in vietnamese text via piecewise convolution neural network with word-level attention," in *2018 5th NAFOSTED Conference on Information and Computer Science (NICS)*, pp.99–103, 2018.

[9] X. Li, Y. Chen, J. Xu, and Y. Zhang, "Attention-based gated convolutional neural networks for distant supervised relation extraction," in *Chinese Computational Linguistics*, pp.246–257, Springer International Publishing, 2019.

[10] E. Haihong, X. Zhou, and M. Song, "Distant supervised relation extraction based on recurrent convolutional piecewise neural network," in *Proceedings of the 2019 International Symposium on Signal Processing Systems*, p.169–175, Association for Computing Machinery, 2019.

[11] N. Rusnachenko and N. Loukachevitch, "Neural network approach for extracting aggregated opinions from analytical articles," in *Data Analytics and Management in Data Intensive Domains*, pp.167–179, Springer International Publishing, 2019.

[12] S. Nam, K. Han, E.-K. Kim, and K.-S. Choi, "Distant supervision for relation extraction with multi-sense word embedding," in *Proceedings of the 9th Global Wordnet Conference*, pp.239–244, Global Wordnet Association, 2018.

[13] D. Puspitaningrum, "Improving performance of relation extraction algorithm via leveled adversarial pcnn and database expansion," in *2019 7th International Conference on Cyber and IT Service Management (CITSM)*, pp.1–6, 2019.

[14] C. Du and L. Huang, "Sentiment analysis method based on piecewise convolutional neural network and generative adversarial network," *International Journal of Computers Communications & Control*, vol.14, no.1, pp.7–20, 2019.

[15] Y. Zhang, "Sentiment classification based on piecewise pooling convolutional neural network," *Cmc-computers Materials & Continua*, vol.56, pp.285–297, 2018.

[16] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.42, no.2, pp.318–327, 2020.

[17] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Machine Learning and Knowledge Discovery in Databases*, pp.148–163, Springer Berlin Heidelberg, 2010.

[18] I. Hendrickx, S. N. Kim, Z. Kozareva, P. Nakov, D. Ó Séaghdha, S. Padó, M. Pennacchiotti, L. Romano, and S. Szpakowicz, "SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, pp.33–38, Association for Computational Linguistics, 2010.

[19] P. Phandi, A. Silva, and W. Lu, "SemEval-2018 task 8: Semantic extraction from CybersecUrity REports using natural language processing (SecureNLP)," in *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp.697–706, Association for Computational Linguistics, 2018.

[20] A. Liu, S. Soderland, J. Bragg, C. H. Lin, X. Ling, and D. S. Weld, "Effective crowd annotation for relation extraction," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp.897–906, Association for Computational Linguistics, 2016.