مجله محاسبات نرم، جلد ۱۱، شماره ۱ (بهار و تابستان ۱۴۰۱)، صفحه ۱۲۰ـ۱۳۹
https://doi.org/10.22052/scj.2023.243182.1010

مجله محاسبات نرم
SOFT COMPUTING JOURNAL
تارنمای مجله: scj.kashanu.ac.ir

دانشگاه کاشان
University of Kashan

طبقه‌بندی متون فارسی مبتنی بر شبکه‌های عصبی عمیق ✧

محمدرضا فیضی‌درخشی[۱]، استاد، زینب متقی‌نیا[۱*]، دانشجوی دکتری، میثم عسگری چناقلو[۱]، دکتری

[۱] دانشکده مهندسی برق و کامپیوتر، دانشگاه تبریز، تبریز، ایران.

اطلاعات مقاله

تاریخچه مقاله:
دریافت ۲۵ مهر ماه ۱۴۰۰
پذیرش ۲۲ آذر ماه ۱۴۰۱

کلمات کلیدی:
طبقه‌بندی متون
شبکه‌های عصبی عمیق
بردار نمایش کلمات
تعبیه کلمات
یادگیری ماشین

چکیده

امروزه با توجه به رشد روزافزون حجم اسناد الکترونیکی، طبقه‌بندی متون بر اساس روش‌های مختلفی در میان محققین بازیابی اطلاعات و متون عمومیت یافته است. با توجه به اهمیت موضوع و کارهایی که در این زمینه در زبان‌های مختلف دنیا انجام گرفته است، نیاز به طبقه‌بندی متون فارسی به خوبی احساس می‌شود. به طور کلی روش‌های طبقه‌بندی متون را می‌توان به روش‌های سنتی (مبتنی بر انتخاب ویژگی و یادگیری ماشین) و روش‌های مبتنی بر یادگیری عمیق تقسیم‌بندی کرد. روش‌های مبتنی بر یادگیری عمیق به دلیل توانایی اشتراک وزن به طور قابل توجهی سبب کاهش تعداد متغیرهای آزاد آموزش‌پذیر شبکه و در نتیجه افزایش تعمیم‌پذیری شده است و نتایج بهتری به نسبت سایر روش‌ها می‌دهد. در زبان فارسی روش‌های مبتنی بر یادگیری عمیق بسیار اندکی برای طبقه‌بندی متون ارائه شده است. در این مقاله دو مدل شبکه عصبی عمیق شامل شبکه عصبی پیچشی ParsCNN و شبکه عصبی با حافظه بلند کوتاه‌مدت دوسویه سلسه‌مراتبی با لایه توجه ParsBiLSTM برای طبقه‌بندی متون فارسی تشریح شده است. کارایی سیستم‌های مبتنی بر شبکه عصبی عمیق بر روی مجموعه داده همشهری بررسی شده و از نظر سه معیار ارزیابی دقت، فراخوانی و مقیاس-F مورد مطالعه قرار گرفته است. نتایج آزمایش‌ها نشان می‌دهد که روش ParsCNN میزان دقت ۰/۶۹، فراخوانی ۰/۷ و مقیاس-F ۰/۶۹؛ همچنین روش ParsBiLSTM میزان دقت ۰/۷۲، فراخوانی ۰/۷۳ و مقیاس-F ۰/۷۲ دارند که نشان‌دهنده کارایی بالاتر این روش‌ها نسبت به روش‌های طبقه‌بندی متون فارسی مورد مطالعه است.

© ۱۴۰۱ نویسندگان. مقاله با دسترسی آزاد تحت مجوز CC-BY

۱. مقدمه

در سال‌های اخیر، تولید حجم عظیمی از داده‌ها سبب شده است تا روش‌های هوشمند استخراج دانش از این مجموعه داده‌ها، همچون طبقه‌بندی[۱]، خوشه‌بندی[۲] [۱]، پیش‌بینی [۲]، ردیابی الگو و ... مورد توجه پژوهشگران قرار بگیرد. طبقه‌بندی متون یکی از موضوعات مطرح در حوزه پردازش زبان طبیعی[۳] [۳]، [۴] می‌باشد که وظیفه آن نسبت دادن یک متن به یک یا چند کلاس از پیش تعریف شده است. با توجه به افزایش چشمگیر حجم مطالب متنی در سال‌های اخیر و با توجه به کاربردهای

✧ نوع مقاله: پژوهشی
* نویسنده مسئول
پست(های) الکترونیک: mfeizi@tabrizu.ac.ir (فیضی‌درخشی)
z.mottagy@gmail.com (متقی‌نیا)
m.asgari.c@gmail.com (عسگری چناقلو)

[1] Classification
[2] Clustering
[3] Natural Language Processing (NLP)

نحوه ارجاع به مقاله: فیضی‌درخشی، محمدرضا، متقی‌نیا، زینب، عسگری چناقلو، میثم، «طبقه‌بندی متون فارسی مبتنی بر شبکه‌های عصبی عمیق»، مجله محاسبات نرم، جلد ۱۱، شماره ۱، ص۱۲۰ـ۱۳۹، بهار و تابستان ۱۴۰۱.

مهم طبقه‌بندی متن، روش‌های طبقه‌بندی خودکار [۵]، [۶]، اهمیت زیادی پیدا کرده است. هم اکنون طبقه‌بندی متون در بسیاری از زمینه‌ها از شاخص‌گذاری متون بر مبنای یک لغت‌نامه کنترل شده تا پالایش متون، تولید خودکار فراداده، ابهام‌زدایی از کلمه، تولید کاتالوگ‌های سلسله مراتبی از منابع وبی، تجزیه و تحلیل احساسات[۱]، طبقه‌بندی خبرها[۲]، پرسش‌وپاسخ[۳] و استنتاج زبان طبیعی و به طور کلی در هر کاربردی که نیاز به سازمان‌دهی یا توزیع انتخابی و تطبیقی خاصی از مستندات مد نظر باشد، کاربرد دارد [۷]. در سال‌های گذشته، روش‌های طبقه‌بندی مبتنی بر انتخاب ویژگی و یادگیری ماشین بسیاری ارائه شده‌اند اما با ظهور یادگیری عمیق[۴] [۸]، [۹]، تحولی شگرف در روش‌های طبقه‌بندی متون صورت پذیرفت. یادگیری عمیق شاخه‌ای از یادگیری ماشین است که در آن با استفاده از توابع محاسباتی به شکل گراف‌های عمیق یا چندسطحی به حل مساله‌های پیچیده پرداخته می‌شود. شبکه‌های عصبی عمیق در بسیاری از وظایف یادگیری ماشینی سخت، موفق بوده‌اند. مدل‌ها و ساختارهای مختلفی مبتنی بر روش‌های یادگیری عمیق همچون شبکه‌های عصبی پیچشی[۵]، شبکه‌های عصبی برگشتی[۶] و ... در حوزه طبقه‌بندی متون پیشنهاد و بررسی شده‌اند. با توجه به دانش ما، تلاش‌های اندکی در زبان فارسی در این زمینه صورت گرفته است.

در این مقاله ابتدا مروری بر تحقیقاتی که در زمینه نمایش بردار کلمات[۷] انجام شده است [۱۰]–[۱۲] خواهیم داشت. سپس روش‌های پیشین طبقه‌بندی متون را به دو دسته روش‌های سنتی و روش‌های مبتنی بر یادگیری عمیق تقسیم‌بندی کرده و به تفصیل به شرح آنها خواهیم پرداخت. دو مدل شبکه عصبی عمیق شامل شبکه عصبی پیچشی به نام ParsCNN و شبکه عصبی با حافظه‌ی بلند کوتاه- مدت[۸] دوسویه سلسله مراتبی با لایه توجه به نام ParsBiLSTM برای طبقه‌بندی متون فارسی پیاده‌سازی و مورد مقایسه قرار خواهد گرفت. تا جایی که دانش و مطالعات ما نشان می‌دهد، این روش‌ها تقریبا جزء اولین روش‌هایی هستند که از یادگیری عمیق برای طبقه‌بندی متون در زبان فارسی استفاده می‌نمایند.

۲. کارهای پیشین

در این بخش از تحقیق مروری بر روش‌های نمایش کلمات که به روش‌های سنتی و روش‌های تعبیه کلمات تقسیم‌بندی شده‌اند، خواهیم داشت. در ادامه برخی از روش‌هایی که برای طبقه‌بندی متون در زبان انگلیسی و فارسی ارائه شده‌اند، معرفی خواهند شد.

۲,۱. نمایش کلمات

نحوه نمایش کلمات یک موضوع اساسی در پردازش زبان طبیعی است. روش‌های نمایش کلمات را می‌توان به دو دسته: روش‌های سنتی و روش‌های تعبیه کلمات تقسیم‌بندی کرد. روش‌های سنتی یا مدل‌های کیفی از کلمات[۹] روش‌هایی هستند که هر کلمه در مجموعه واژگان[۱۰] را با یک بیت در یک بردار بزرگ نمایش می‌دهند، برای مثال اگر مجموعه واژگان مشتمل بر ۱۰۰۰۰ کلمه باشد و کلمه «Hello» چهارمین کلمه از این مجموعه‌ی واژگان باشد بردار نمایش به صورت

0 0 0 1 0 0 ..... 0 0 0 0

خواهد بود که این امر باعث تنک‌شدن بردار کلمات می‌شود بنابراین هم نیاز به فضای ذخیره‌سازی زیادی خواهیم داشت و هم پیچیدگی الگوریتم‌ها و زمان اجرای آنها بسیار بالا است. در برخی آثار، ابعاد بردارهای کلمه خلوت با استفاده از روش‌هایی مانند تجزیه مقادیر منفرد[۱۱] [۱۳] یا روش تخصیص پنهان دیریکله[۱۲] [۱۴]– [۱۶] کاهش می‌یابد. به عبارت بهتر در این دسته روش‌ها، هر کلمه به عنوان یک نماد گسسته و مجزا در

[1] Sentiment analysis
[2] News categorization
[3] Question answering
[4] Deep Learning
[5] Convolutional Neural Networks (CNNs)
[6] Recurrent Neural Networks (RNNs)
[7] Word vector representation
[8] Long Short-Term Memory (LSTM)
[9] Bag Of Words (BOW) Model
[10] Vocabulary
[11] Singular Value Decomposition (SVD)
[12] Latent Dirichlet Allocation (LDA)

نظر گرفته می‌شود و از قابلیت تعمیم ضعیف رنج می‌برد؛ بنابراین ما به دنبال نمایشی از کلمات هستیم که هم مشکل خلوت بودن بردار را حل کند و هم شباهت معنایی و نحوی بین کلمات را در نظر بگیرد که روش‌های تعبیه کلمات [۱۷] این مشکل را برطرف کرده‌اند. در این دسته از روش‌ها کلماتی که شباهت معنایی دارند به احتمال زیاد در یک خوشه قرار گرفته و تشکیل یک گروه معنایی را می‌دهند. علاوه بر این، فضاهای برداری با ارائه ساختار خطی با استفاده از عملیات افزایشی ساده بردار کلمات را به طور معناداری با هم ترکیب می‌کنند [۱۲]. به طور مثال:

$$\text{vec(Germany)} + \text{vec}(Capital) \approx \text{vec}(Berlin)$$

$$\text{vec}(Athlete) + \text{vec(Football)} \approx \text{vec(Football_Player)}$$

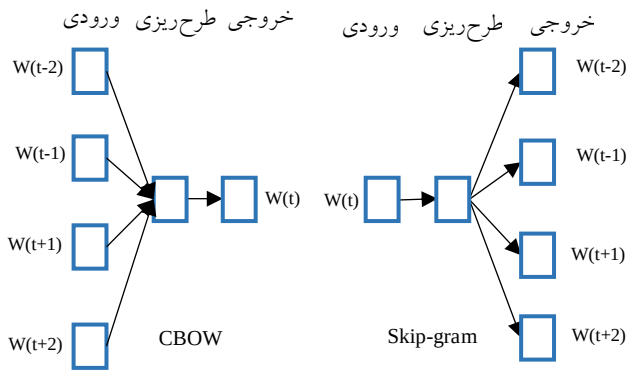
همانطور که مشاهده می‌کنید بردار کلمه «آلمان» بعلاوه بردار کلمه «پایتخت» برابر بردار کلمه «برلین» است و بردار کلمه «ورزشکار» بعلاوه بردار کلمه «فوتبال» برابر بردار کلمه «بازیکن فوتبال» است. مثال‌های بالا نشان می‌دهد که ترکیب افزایشی اغلب می‌تواند نتایج معناداری تولید کند، همچنین روابط ساختاری و مفهومی بین کلمات را نمایش می‌دهد.

روش‌های تعبیه کلمات روشی بسیار کارآمد و مناسب برای نمایش لغات و متون و پردازش آنها است که در بسیاری از کاربردهای نوین پردازش متن مانند سنجش شباهت کلمات [۱۸]، ترجمه ماشینی [۱۹]، [۲۰]، احساس کاوی [۲۱]- [۲۶]، نظرکاوی [۲۷]، [۲۸]، تحلیل معنایی اسناد و ... استفاده می‌شود. در این روش هر کلمه را به عنوان نقطه‌ای در فضا در نظر می‌گیرند که توسط برداری از تعداد ثابتی ابعاد نمایش داده می‌شوند. برای مثال کلمه «Hello» ممکن است با

[0.4, -0.11, 0.55, 0.3 … 0.1, 0.02]

نمایش داده شود. این بردارهای متراکم با استفاده از روش‌های متفاوتی که از شبکه‌های عصبی الهام گرفته شده است آموزش داده می‌شوند [۱۱]، [۱۲]، [۲۹]- [۳۱]. کار با این دسته از روش‌ها به علت اینکه با ماتریس‌های با ابعاد کوچک‌تر کار می‌کنند، آسان‌تر است.

Mikolov و همکارانش [۳۲]، برای ساخت بردار کلمات در یک مجموعه داده بزرگ دو مدل کیف کلمات پیوسته[1] و پرش-چندتایی[2] را ارائه کردند. این دو مدل هر دو یک شبکه عصبی ساده شامل سه لایه ورودی، طرح‌ریزی[3] و خروجی هستند که لایه طرح‌ریزی بین تمامی کلمات به اشتراک گذاشته شده است. این دو مدل بدون وجود لایه پنهان به کمک چند قانون ساده، بردارهای کلمات را تولید می‌کنند. در روش کیف کلمات پیوسته، ابتدا به ازای هر کلمه یک بردار با طول مشخص و با اعداد تصادفی (بین صفر و یک) تولید می‌شود. سپس به ازای هر کلمه از یک سند یا متن، تعدادی مشخص از کلمات بعد و قبل آن را به شبکه عصبی داده (به غیر از خود کلمه فعلی) و با عملیات ساده ریاضی، بردار کلمه فعلی را تولید می‌کند که این اعداد با مقادیر قبلی بردار کلمه جایگزین می‌شوند. زمانی‌که این کار بر روی تمام لغات در تمام متون انجام گیرد، بردارهای نهایی لغات همان بردارهای مطلوب ما هستند. روش پرش-چندتایی عکس روش کیف کلمات پیوسته است به این صورت که بر اساس یک کلمه داده شده، چندین لغت قبل و بعد آن کلمه را تشخیص می‌دهد (شکل (۱)).



شکل (۱): مدل CBOW و Skip-gram [۱۶]

Pennington و همکارانش [۳۳]، در دانشگاه استنفورد برای نمایش برداری کلمات روش بردار جهانی[4] را ارائه کرده‌اند. الگوریتم بردار جهانی شامل مراحل زیر است:

۱. ماتریس هم‌رخدادی کلمات با نام X بر اساس طول

[1] CBOW(Continuous Bag-Of-Words)
[2] Skip-gram
[3] Projection
[4] Global Vectors for Word Representation (Glove)

هم‌جواری[1] با شمارش هم‌رخدادی کلمات تشکیل می‌شود. هر درایه X_{ij} ماتریس نشانگر تعداد دفعات حضور کلمه i در متن مربوط به کلمه j است. بر اساس نزدیکی یا دوری دو کلمه به یکدیگر، این اعداد، وزن‌دهی می‌شوند.

۲. تعریف محدودیت نرم برای هر جفت کلمه:

$$w_i^T w_j + b_i + b_j = \log(X_{ij}) \quad (۱)$$

که w_i بردار کلمه اصلی، w_j بردار کلمه متن، b_i و b_j بایاس‌های کلمات اصلی و متن هستند.

۳. تعریف تابع هزینه:

$$J = \sum_{i=1}^{V}\sum_{j=1}^{V} f(X_{ij})\left(w_i^T w_j + b_i + b_j - \log X_{ij}\right)^2 \quad (۲)$$

که V تعداد واژگان و f تابع وزنی است که تابعی از هم‌رخدادی w_i و w_j بوده و قابل تنظیم می‌باشد و در این روش با استفاده از رابطه (۳) انتخاب شده است.

$$f(X_{ij}) = f(x) = \begin{cases} (x/x_{max}) , & if\, x < x_{max} \\ 1 , & otherwise \end{cases} \quad (۳)$$

خروجی این تابع با $x \to 0$ نزدیک به صفر خواهد شد ($f(0) = 0$)، همین طور $f(x)$ غیرنزولی است تا هم‌رخدادی بالاتر تاثیرگذاری بیشتری داشته باشد، زمانی‌که x خیلی بزرگ باشد نیز اشباع می‌گردد. در رابطه (۳) مقدار بهینه x_{max} برابر ۱۰۰ در نظر گرفته شده است.

۲،۲. طبقه‌بندی متن

طبقه‌بندی متون، از سال ۱۹۶۰ میلادی به بعد مورد توجه قرار گرفته است و با شروع دهه ۹۰ با وجود پیشرفت‌های سخت‌افزاری و نرم‌افزاری شاهد پیشرفت‌های چشم‌گیری در این حوزه بوده‌ایم. هدف از طبقه‌بندی متون نسبت دادن کلاس‌های از پیش تعریف شده به اسناد متنی است. در طبقه‌بندی، یک مجموعه آموزشی از اسناد با کلاس‌های معین وجود دارد که با استفاده از این مجموعه، مدل طبقه‌بندی معین شده و کلاس سند جدید مشخص می‌گردد. روش‌های طبقه‌بندی متون را می‌توان به دو دسته روش‌های سنتی و روش‌های مبتنی بر یادگیری عمیق تقسیم‌بندی کرد که در ادامه به تفصیل در مورد این دو دسته توضیح خواهیم داد.

۲،۲،۱. طبقه‌بندی متن به‌صورت سنتی

روش‌های طبقه‌بندی متون به صورت سنتی روش‌های مبتنی بر انتخاب ویژگی و یادگیری ماشین هستند که به طور عمده شامل بخش‌های پیش‌پردازش[2]، استخراج ویژگی[3]، انتخاب یا کاهش ویژگی[4] و طبقه‌بند یا دسته‌بند می‌باشند. در بخش پیش‌پردازش، داده‌های ورودی به شکلی که برای طبقه‌بندی مناسب باشند بازنمایی می‌شوند. در فاز استخراج ویژگی، بخشی از اطلاعات که برای طبقه‌بندی مهم نیست، حذف شده و اطلاعات مفید برای طبقه‌بندی تحت عنوان ویژگی استخراج می‌شود؛ که این امر باعث کاهش پیچیدگی محاسباتی می‌شود. انتخاب ویژگی تعداد مؤلفه‌های بردار ویژگی را کاهش می‌دهد. در فاز طبقه‌بند، طبقه‌بندی متون با روش‌های مختلفی برای زبان انگلیسی و زبان فارسی انجام پذیرفته است. در زبان انگلیسی با استفاده از بردارهای فراوانی ریشه کلمات، ماشین بردار پشتیبان[5]، روش تحلیل معنایی پنهان[6]، شبکه‌های عصبی ترکیبی[7]، تخصیص پنهان دیریکله و ... اقدام به طبقه‌بندی متون کرده‌اند [۳۴]- [۴۱]. در روش‌های سنتی ارائه شده برای طبقه‌بندی متون در زبان فارسی مؤلفه‌های طبقه‌بندی کننده اغلب مبتنی بر یکی از روش‌های مدل رگرسیون، طبقه‌بندی کننده چند نزدیک‌ترین همسایه[8]، درخت تصمیم‌گیری[9]، ماشین بردار پشتیبان، شبکه عصبی[10] و مدل احتمال بیز[11] است [۴۲]- [۴۶]. در مرجع [۴۷] روشی مبتنی بر معیار شاخص‌گذاری n-بخشی[12] ارائه گردیده است که با استفاده از الگوریتم یادگیری ماشینی بیز ساده به طبقه‌بندی اخبار

[1] Window Size

[2] Preprocessing
[3] Feature Extraction
[4] Feature Selection
[5] Support Vector Machine (SVM)
[6] Latent semantic analysis (LSA)
[7] Hybrid Neural Network
[8] K-Nearest Neighbor (KNN)
[9] Decision Tree
[10] Neural Network
[11] Naïve Bayes (NB)
[12] N-Gram

فارسی می‌پردازد. نتایج حاصل نشان می‌دهد که n-بخشی‌های با طول سه نتایج بهتری به دست می‌دهد، همچنین پیش‌پردازش داده‌ها بعلاوه هموارسازی[1] عقب‌گرد[2] نتایج را اندکی بهبود می‌بخشد. نویسندگان مقاله [۴۲] روشی مبتنی بر چند نزدیک‌ترین همسایه و وردنت برای طبقه‌بندی متون فارسی ارائه داده‌اند. در این روش تمامی کلمات با ریشه خود جایگزین می‌شوند، سپس کلمات با اطلاعات کم و تعداد تکرار زیاد حذف می‌شوند. کلمات موجود در سند به وردنت داده می‌شود و اگر کلمه مرتبط پیشنهادی توسط وردنت حداقل یک بار در سند استفاده شده باشد، تعداد تکرار آن با بیشینه تعداد تکرار کلمه و تعداد کلمه مرتبط جایگزین می‌شود. مقدار فراوانی عبارت[3] برای اسناد در مجموعه آموزشی با استفاده از $log(tf+1)$ به دست می‌آید. در نهایت، با استفاده از روش تحلیل اجزای اصلی[4] تعداد مشخصه‌های استخراج شده کاهش می‌یابد و کلاس‌بند چند نزدیک‌ترین همسایه اعمال می‌شود. نتایج پیاده‌سازی نشان می‌دهد که بهترین مقدار k در این برنامه ۱۰ می‌باشد. در مرجع [۴۸] یک هسته معنایی[5] ارائه شده است که به دنبال جایگزینی مشخصه‌های مفهومی به جای کلمات در اسناد متنی می‌باشد. در این روش، با بهره‌گیری از تجزیه مقادیر منفرد در ماتریس کلمه-سند و استخراج K بردار ویژه با بالاترین انرژی، اسناد را به یک فضای مشخصه K بعدی جدید منتقل می‌کند. این امر نه تنها موجب کاهش پیچیدگی محاسباتی می‌شود، بلکه کارایی طبقه‌بندی را نیز افزایش می‌دهد. نتایج ارزیابی‌ها نشان می‌دهد که استفاده از هسته معنایی پیشنهادی کارایی کلاس‌بندهای ماشین بردار پشتیبان و بیز ساده را افزایش می‌دهد. در مرجع [۴۵] الگوریتم‌های مختلف شبکه بیز برای طبقه‌بندی متون فارسی مورد مقایسه قرار گرفته است. زمانی که مشاهدات و داده‌ها از نوع پیوسته باشند، از طبقه‌بند بیز ساده گاوسی[6] استفاده شده است. طبقه‌بند بیز ساده برنولی[7] فرض را بر وجود یا عدم وجود یک ویژگی قرار می‌دهد، به این معنی که معنی صفر نشان‌دهنده «کلمه در سند نیست» و یک نشان‌دهنده «کلمه در سند رخ می‌دهد» است. زمانی که داده‌های گسسته داشته باشیم از طبقه‌بند بیز ساده چندجمله‌ای[8] استفاده می‌شود. در این روش‌ها از فراوانی عبارت و فراوانی عبارت-معکوس فراوانی سند[9] و ۳-بخشی[10] برای مدل‌سازی آماری متن استفاده شده است. در فرآیند مقایسه الگوریتم‌های ذکر شده، بهترین نتایج توسط الگوریتم طبقه‌بند بیز ساده چندجمله‌ای به دست آمده است. در این الگوریتم، صحت[11] طبقه‌بندی متن با وجود ایست‌واژه‌ها[12] ۸۳٪ و پس از حذف ایست‌واژه‌ها ۸۲/۴٪ به دست آمده است که نتایج تقریبا یکسانی می‌دهد. نویسندگان در [۴۶] از متداول‌ترین الگوریتم‌های یادگیری ماشین یعنی طبقه‌بند بیز ساده و چند نزدیک‌ترین همسایه برای طبقه‌بندی متن فارسی استفاده نموده‌اند. در این مطالعه، روشی برای بهبود عملکرد طبقه‌بندی متن با استفاده از اطلاعات متقابل[13] و الگوریتم‌های انتخاب ویژگی خی‌دو آماری (x2)[14] پیشنهاد شده است. نتایج ارزیابی‌ها نشان می‌دهد که روش انتخاب ویژگی اطلاعات متقابل می‌تواند صحت طبقه‌بندی بیز ساده را تا ۱۰٪ بهبود بخشد. صحت الگوریتم پیشنهادی به طور متوسط ۸۰٪ و مقدار میانگین پارامتر مقیاس-F1 نیز ۸۰٪ گزارش شده است.

روش‌های طبقه‌بندی متون سنتی محدودیت‌های زیادی دارند. به عنوان مثال، استخراج‌کننده‌های ویژگی دستی و پرزحمتی در این سیستم‌ها مورد استفاده قرار می‌گیرند که برای داشتن کارایی مناسب نیاز به تجزیه‌وتحلیل و مهندسی ویژگی‌ها دارند و برای مجموعه داده‌هایی با ابعاد بزرگ مقیاس‌پذیر نیستند. علاوه بر این، وابستگی شدید به دانش دامنه برای طراحی ویژگی‌ها، تعمیم این روش‌ها را برای کارهای جدید دشوار می‌کند. همچنین، این مدل‌ها نمی‌توانند از مقدار زیادی از داده‌های آموزشی کاملا استفاده کنند چون ویژگی‌ها (الگوهای ویژگی‌ها)

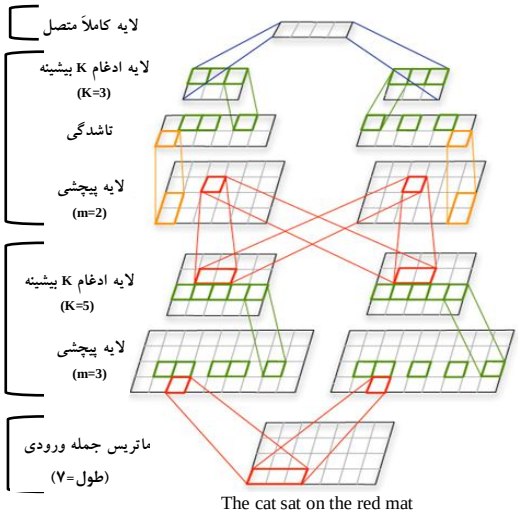
[1] Smoothing
[2] Back-off
[3] Term Frequency (TF)
[4] Principal Component Analysis (PCA)
[5] Semantic Kernel
[6] Gaussian NB
[7] Bernoulli NB
[8] Multinomial NB
[9] Term Frequency-Inverse Document Frequency (TF-IDF)
[10] Trigram
[11] Accuracy
[12] Stop Words
[13] Mutual information (MI)
[14] Chi-Square

از قبل تعریف شده‌اند.

۲،۲،۲. طبقه‌بندی متن مبتنی بر یادگیری عمیق

روش‌های یادگیری مبتنی بر شبکه‌های عصبی عمیق با افزایش قدرت محاسباتی سخت‌افزارها و رفع برخی چالش‌های موجود در آموزش و یادگیری این نوع شبکه‌ها، بیش‌ازپیش مورد توجه قرار گرفته‌اند. این شبکه‌ها قادر به انجام محاسبات فراوان به صورت موازی در چندین لایه، تخمین پاسخ مسئله داده شده با تعداد زیادی پارامتر و ارائه مدل مناسبی از آن هستند؛ بنابراین قدرت بازنمایی بالایی دارند. شبکه‌های عصبی عمیق در انجام وظایف سخت یادگیری ماشین بسیار قدرتمند عمل کرده‌اند و در مواردی روش‌های سنتی را به طور کامل کنار زده‌اند. یک مدل مبتنی بر یادگیری عمیق از مجموعه‌ای از لایه‌ها تشکیل شده است: یک لایه ورودی، یک لایه خروجی و یک یا چند لایه پنهان. لایه ورودی، ورودی را به‌صورت ماتریس دریافت می کند. لایه های پنهان انتخاب ویژگی و کاهش ابعاد را انجام می‌دهند. لایه خروجی، سند ورودی را به یکی از برچسب‌های از پیش تعریف شده طبقه‌بندی می‌کند. مدل‌های مبتنی بر یادگیری عمیق به طور خودکار ویژگی‌ها را از مجموعه داده بدون روش‌های صریح برنامه‌ریزی شده، فرامی‌گیرند. در حوزه طبقه‌بندی، روش‌های مبتنی بر یادگیری عمیق در زبان فارسی بسیار اندکی ارائه شده است؛ بنابراین در این بخش به بررسی روش‌های ارائه شده برای زبان انگلیسی نیز می‌پردازیم. روش‌های طبقه‌بندی متون مبتنی بر یادگیری عمیق در زبان انگلیسی را از نظر سطح می‌توان به چهار دسته: طبقه‌بندی در سطح اسناد [۴۹]، طبقه‌بندی در سطح جملات [۵۰]– [۵۴]، طبقه‌بندی در سطح کلمه [۵۵]–[۵۷] و طبقه‌بندی در سطح کاراکترها [۵۸]– [۶۰] تقسیم‌بندی کرد.

Kalchbrenner و همکاران [۵۱]، یک شبکه عصبی پیچشی پویا[1] برای مدل‌سازی جملات ارائه کرده‌اند. هدف از مدل کردن یک جمله، تجزیه‌وتحلیل و ارائه محتوای معنایی جمله برای اهدافی مانند طبقه‌بندی است. شبکه ارائه شده یک شبکه پیچشی با یک لایه ادغام k بیشینه[2] پویا است که ورودی‌های با طول‌های متفاوت را مدیریت می‌کند. ادغام k بیشینه پویا دارای دو ویژگی است: اول، ادغام k بیشینه بر روی یک دنباله خطی از مقادیر، دنباله‌ای از k مقدار بیشینه را به جای مقدار بیشینه منفرد برمی‌گرداند. دوم، پارامتر ادغام k تابعی از ورودی و سایر پارامترهای شبکه هست و به صورت پویا انتخاب می‌شود. در این شبکه اندازه نقشه ویژگی[3] در لایه میانی بسته به طول جمله ورودی متغیر است و همچنین بر درخت تجزیه متکی نمی‌باشد و به راحتی برای هر زبانی قابل اجرا است. شکل (۲)، DCNN ارائه شده را نمایش می‌دهد. همانطور که در این شکل نمایش داده شده است لایه پیچشی دوم در این شبکه شامل لایه تاشدگی[4] می‌باشد که وظیفه آن جمع دو سطر ماتریس نقشه ویژگی و نصف کردن اندازه آن است.



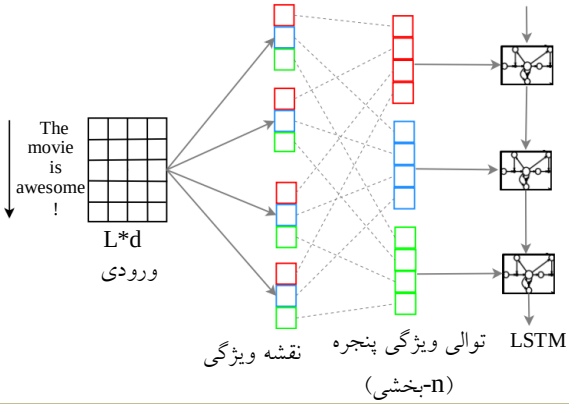
شکل (۲): DCNN برای جمله ورودی با ۷ کلمه و بردار کلمات با اندازه D=2، دو لایه پیچشی با دو نقشه ویژگی برای هر لایه، عرض فیلترها در هر دولایه به ترتیب ۳ و ۲، لایه ادغام K بیشینه با مقدار K= 5, 3 [۵۱]

در مرجع [۵۲] مدلی به نام C-LSTM برای طبقه‌بندی در سطح جملات معرفی شده است که شامل دو بخش CNN و LSTM می‌باشد (شکل (۳)). مدل C-LSTM از CNN یک لایه برای استخراج دنباله‌ای از بازنمایی‌هایی سطح بالا از عبارات استفاده می‌کند. در این بخش از چندین فیلتر برای تولید چندین نقشه

[1] Dynamic Convolutional Neural Network (DCNN)

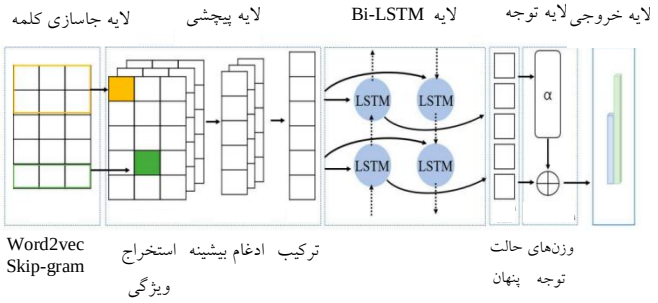
[2] K-max Pooling
[3] Feature map
[4] Folding

ویژگی استفاده می‌شود. همچنین بعد از عملیات پیچشی ادغام اعمال نمی‌شود. خروجی CNN به عنوان ویژگی‌های پنجره متوالی سازماندهی می‌شوند تا وارد LSTM شوند. به این ترتیب، به جای اینکه LSTM به طور مستقیم از جمله ورودی ساخته شود، ابتدا هر جمله به ویژگی‌های پنجره (n-بخشی) متوالی تبدیل می‌شود و LSTM بازنمایی از جمله را به دست می‌آورد. به این صورت که خروجی لایه پنهان LSTM در آخرین گام زمانی بازنمایی از سند بوده که یک لایه بیشینه هموار[1] هم به بالای مدل اضافه می‌شود.



شکل (۳): معماری مدل C-LSTM [۵۲]

در مطالعه [۵۳]، مدل ترکیبی Bi-LSTM+CNN پیشنهاد داده شده است که برای بهبود بیشتر و کاهش تعداد پارامترهای قابل یادگیری، مدل توسط مکانیسم توجه تقویت می‌شود. همان‌گونه که در شکل (۴) مشاهده می‌شود، این مدل دارای لایه‌های جاسازی کلمه، پیچشی، Bi-Lstm، توجه و لایه خروجی می‌باشد. در لایه جاسازی کلمه پس از پیش‌پردازش متن ورودی از روش نمایش کلمه Skip-gram استفاده می‌کند. این مدل از CNN برای استخراج ویژگی‌ها از مکان‌های مختلف در یک جمله استفاده می‌کند، در نتیجه مقدار ویژگی‌های ورودی را کاهش می‌دهد. سپس LSTM برای استخراج اطلاعات متنی از ویژگی‌های به دست آمده از لایه پیچشی استفاده می‌شود. مکانیسم توجه از ترازبندی بایاس بر ورودی‌ها استفاده می‌کند و به اجزای ورودی که با طبقه‌بندی همبستگی زیادی دارند وزنی اختصاص می‌دهد، از این رو تعداد پارامترهای یاد گرفته شده در طول آموزش کاهش می‌یابد. مکانیسم توجه همچنین توزیع وزن برای توالی‌های با طول متغیر را افزایش می‌دهد. نتایج نشان می‌دهد استفاده از وزن‌هایی که با بهره‌گیری از Word2vec با یک پیکره بزرگ آموزش دیده‌اند سبب افزایش صحت مدل پیشنهادی می‌شود.



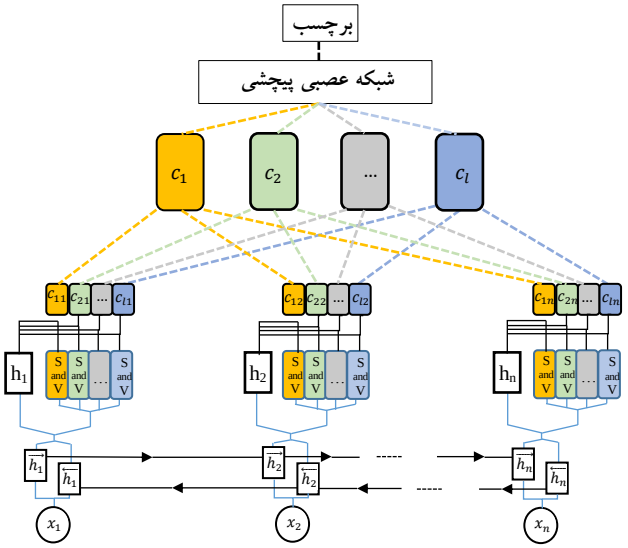
شکل (۴): معماری مدل ترکیبی Bi-LSTM + CNN [۵۳]

مقاله [۵۴] شبکه عصبی چندکاناله مبتنی بر توجه[2] را برای طبقه‌بندی متن پیشنهاد می‌کند. مدل AMCNN از یک حافظه کوتاه‌مدت بلندمدت دوجهته برای رمزگذاری تاریخچه و اطلاعات آینده کلمات در نمایش‌های با ابعاد بالا استفاده می‌کند، به طوری که می‌توان اطلاعات بعد و قبل جمله را به طور کامل بیان کرد. سپس مکانیسم توجه عددی و مکانیسم توجه برداری برای به دست آوردن نمایش‌های چندکاناله اعمال می‌شود. مکانیسم توجه عددی می‌تواند اهمیت سطح کلمه را محاسبه کند و مکانیسم توجه برداری می‌تواند اهمیت سطح ویژگی را محاسبه کند. مدل AMCNN به جای محاسبه مجموع وزنی، از ساختار CNN برای تجزیه روابط کلمه بر روی نمایش‌های تولی شده توسط مکانیسم توجه عددی و برداری استفاده می کند. به این ترتیب، توانایی CNN در استخراج ویژگی‌های n-بخشی نیز افزایش یافته است. مطابق شکل (۵) جمله‌ای با n کلمه وارد شبکه می‌شود ابتدا با استفاده از ماتریس جاسازی کلمه به بردار متراکم تبدیل می‌شود. در ادامه با استفاده از Bi-LSTM حاشیه‌نویسی از کلمات به دست می‌آید. سپس مکانیسم‌های توجه به حالت‌های پنهان اعمال شده و آنها را به یک ماتریس متصل می‌کند. در این شکل l تعداد کانال‌ها، c_{li} نمایش جدید از

[1] Softmax

[2] Attention-based Multichannel Convolutional Neural Network (AMCNN)

h_i در این کانال است. با الحاق c_{li} ها مقدار c_l به دست می‌آید. c_l ها وارد شبکه پیچیشی شده و خروجی این شبکه برچسب جملات خواهد بود.

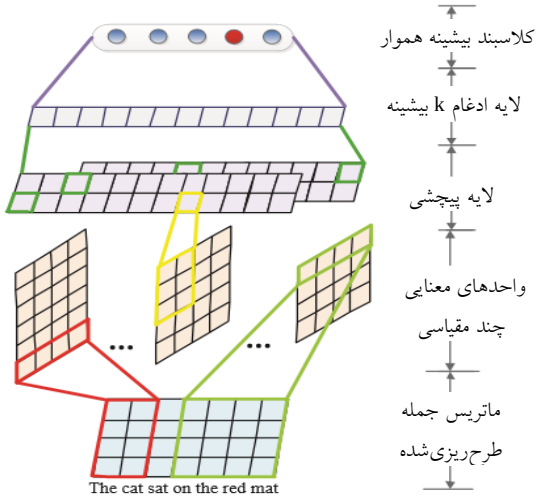


شکل (۵): ساختار شبکه عصبی چندکاناله مبتنی بر توجه، وقتی S و V به ترتیب مکانیسم توجه عددی و برداری را نشان می‌دهند. بلوک‌های همرنگ در یک کانال ادغام می‌شوند [۵۴]

در مرجع [۵۵] روشی برای مدل‌سازی متون کوتاه با استفاده از خوشه‌بندی معنایی[1] و شبکه عصبی پیچشی ارائه شده است. در مرحله اول، الگوریتم خوشه‌بندی سریع [۶۱] بر مبنای جستجوی قله تراکم[2]، برای خوشه‌بندی بردارهای کلمات و کشف گروه‌های معنایی استفاده شده است. همان‌گونه که در شکل (۶) آمده است، معماری روش پیشنهادی از نمایش برداری کلمات که از قبل آموزش دیده شده‌اند برای مقداردهی اولیه جدول جستجو[3] استفاده می‌کند که برای متن کوتاه S، ماتریس طرح‌ریزی شده[4] PM با جستجو در این جدول در لایه اول به دست می‌آید. لایه دوم برای به دست آوردن واحدهای معنایی[5] چند مقیاسی برای تشکیل ماتریس‌های معنایی به کار می‌رود که با لایه‌های پیچشی، ادغام k بیشینه و کلاس‌بند بیشینه هموار ترکیب شده است.

[1] Semantic Clustering
[2] Density peaks
[3] Lookup table
[4] Projected matrix
[5] Semantic Units (SUs)

در مرجع [۵۸] یک شبکه پیچشی در سطح کاراکتر برای طبقه‌بندی متون ارائه شده است که نیازی به داشتن اطلاعات در مورد کلمات ندارد. این روش اولین روشی است که شبکه پیچشی را فقط بر روی کاراکترها اعمال کرده است. این مدل یک رشته از کاراکترهای کد شده را به عنوان ورودی دریافت می‌کند که برای مساله‌های مختلف طول ورودی ممکن است متفاوت باشد؛ در این روش دو شبکه پیچشی، یکی بزرگ و دیگری کوچک طراحی شده است که هر کدام از آنها شامل ۶ لایه پیچشی و ۳ لایه کاملا متصل[6] است که بین این سه لایه دو ماژول حذف تصادفی برای تنظیم وجود دارد (شکل (۷)). این روش با تعداد زیادی از روش‌های سنتی و یادگیری عمیق با پایگاه داده‌های با مقیاس بزرگ مقایسه شده است که نتایج این مقایسه‌ها نشان می‌دهد که روش پیشنهادی یک روش کارآمد است و کارایی آن به عوامل زیادی همچون اندازه پایگاه داده وابسته است.



شکل (۶): معماری مدل مبتنی بر خوشه‌بندی معنایی و شبکه عصبی پیچشی [۵۵]

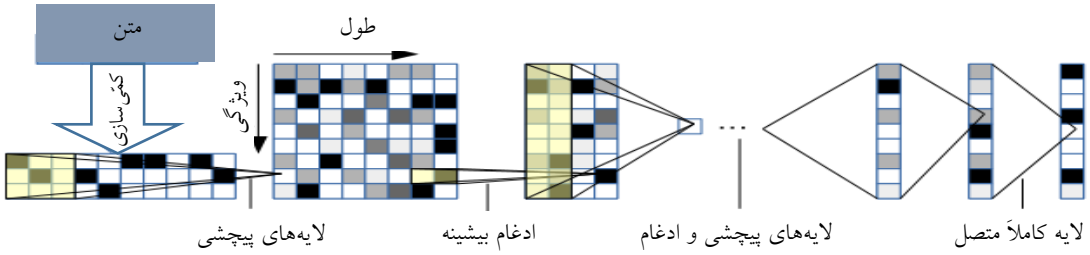
در مرجع [۶۰] شبکه طبقه‌بندی متن مبتنی بر کاراکتر به نام CharTeC-Net معرفی شده است. این مدل معماری CNN بسیار سبک (با حدود بیش از دویست هزار پارامتر) و از نظر محاسباتی کارآمد است که از چهار بلوک ساختاری (هر بلوک شامل دو لایه هستند) برای استخراج ویژگی تشکیل شده است.

[6] Fully Connected (FC) Layer

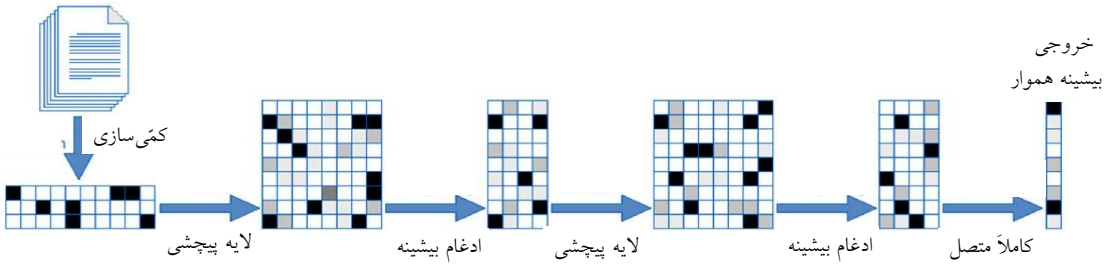
هر یک از این بلوک‌های ساختاری، به جز مورد آخر، از لایه پیچشی نقطه‌ای ۱×۱ برای افزودن غیرخطی بودن بیشتر به شبکه و افزایش ابعاد درون هر بلوک ساختاری استفاده می‌کند. علاوه بر این، اتصال های میانبر در هر بلوک ساختاری برای تسهیل جریان گرادیان‌ها از طریق شبکه استفاده می‌شود، اتصال‌های میانبر از لایه‌های سطح بالاتر به لایه‌های سطح پایین‌تر استفاده می‌شود. اما مهم‌تر از همه این است که سیگنال اصلی موجود در داده‌های آموزش در هر بلوک ساختاری شبکه به اشتراک گذاشته شود.

در حال حاضر، روش‌های طبقه‌بندی متون فارسی مبتنی بر یادگیری کم و محدود هستند. در مرجع [۶۲] روش مبتنی بر یادگیری عمیق در سطح کاراکتر برای طبقه‌بندی متون فارسی ارائه شده است. همان‌گونه که در شکل (۸) مشاهده می‌کنید این مدل متشکل از دو لایه پیچشی، دو لایه ادغام بیشینه، بخش کمّی‌سازی (quantization) و لایه کاملا متصل می‌باشد. بخش کمّی‌سازی اسناد را به دنباله‌ای از داده‌ها تبدیل می‌کند و آنها را به عنوان ورودی لایه پیچشی آماده می‌نماید. در لایه‌های پیچشی این مدل فیلترهای با اندازه ۴ و گام ۲ اعمال می‌شود. همچنین تعداد کلاس‌های خروجی لایه کاملا متصل ۲۰ می‌باشد.

تمامی روش‌های طبقه‌بندی متن مبتنی بر یادگیری عمیق مورد مطالعه در جدول (۱) آمده است. بهترین معماری شبکه عصبی برای طبقه‌بندی متن بسته به ماهیت وظیفه و دامنه هدف، در دسترس بودن برچسب‌های درون دامنه، محدودیت‌های تاخیر و ظرفیت برنامه‌ها و غیره بسیار متفاوت است [۶۳].



شکل (۷): نمایش مدل شبکه پیچشی در سطح کاراکتر برای طبقه‌بندی متون انگلیسی [۵۸]



شکل (۸): ساختار مدل شبکه پیچشی در سطح کاراکتر برای طبقه‌بندی متون فارسی [۶۲]

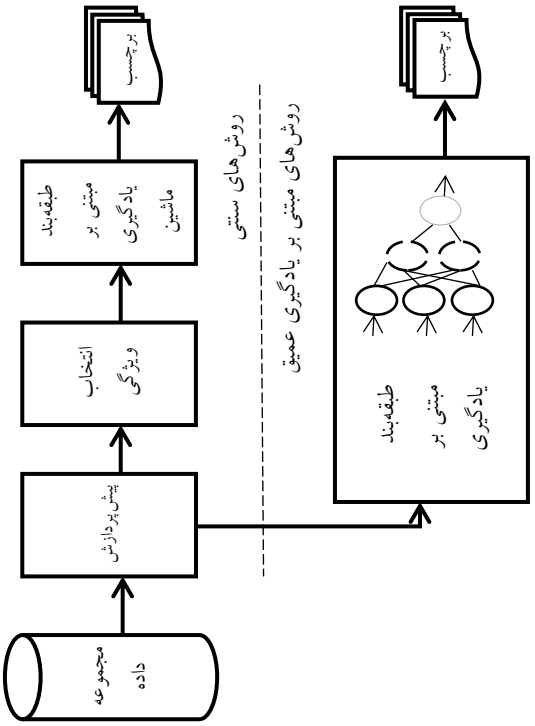
۲،۲،۳. تفاوت روش‌های طبقه‌بندی متن سنتی و مبتنی بر یادگیری عمیق

روند کار طبقه‌بندی متن هر دو روش سنتی و مبتنی بر یادگیری عمیق در شکل (۹) نشان داده شده است. همان‌گونه که پیش از این توضیح داده شد، ابتدا مجموعه داده پیش‌پردازش می‌شود تا برای ورود به یک طبقه‌بندی مناسب باشد. پس از پیش‌پردازش، در روش‌های سنتی، روش‌های انتخاب ویژگی برای انتخاب ویژگی‌های ارزشمند از فضای ویژگی بزرگ استفاده شده است. یک طبقه‌بندی‌کننده، ویژگی‌های انتخاب شده را برای طبقه‌بندی اسناد متنی فرامی‌گیرد. پس از پیش‌پردازش، در مدل‌های مبتنی بر یادگیری عمیق، ویژگی‌ها به طور خودکار توسط لایه‌های عمیق استخراج می‌شوند. در مقابل روش‌های سنتی، یک مدل مبتنی بر یادگیری عمیق می‌تواند الگوهای پیچیده مجموعه داده را بیاموزد اما این مدل ها برای آموزش به داده های بیشتری نیاز دارند و همچنین برای محاسبه سریع ماتریس به سخت‌افزار

واحد پردازنده گرافیکی ویژه نیاز دارند. تفاوت بین روش‌های سنتی و مبتنی بر یادگیری عمیق به طور خلاصه در جدول (۲) آمده است.

جدول (۱): روش‌های طبقه‌بندی متون مبتنی بر یادگیری عمیق

مدل و مرجع	معماری	مجموعه داده	معیارهای ارزیابی	زبان مورد مطالعه
DCNN [۵۱]	CNN	1. movie reviews in the Stanford Sentiment Treebank [64] 2. TREC questions dataset [65] 3. twitter	Accuracy	انگلیسی
C-LSTM [۵۲]	CNN+LSTM	1. movie reviews in the Stanford Sentiment Treebank 2. TREC questions dataset	Accuracy	انگلیسی
Bi-LSTM+CNN attention hybrid model [۵۳]	attention-based Bi-LSTM + CNN	Internet Movie Database (IMDB) movie review	Accuracy, Precision Recall, F1-score	انگلیسی
AMCNN [۵۴]	Attention-based CNN + Bi-LSTM	1. movie review 2. Subjectivity dataset 3. Stanford Sentiment Treebank 4. MPQA	Accuracy	انگلیسی
Semantic-CNN [۵۵]	semantic clustering +CNN	1. Google Snippets [66] 2. TREC	Accuracy	انگلیسی
ConvNets [۵۸]	CNN	1. AG's news 2. Sogou news 3. DBPedia 4. Yelp reviews 5. Yahoo! Answers 6. Amazon reviews	Relative Errors	انگلیسی
CharTeC-Net [۶۰]	CNN	1. AG's news 2. Sogou news 3. DBPedia 4. Yelp reviews 5. Yahoo! Answers 6. Amazon reviews	Accuracy	انگلیسی
Persian Text character-level convolutional networks [۶۲]	CNN	Hamshahri	Accuracy, Precision Recall, F1-score	فارسی



شکل (۹): روند کار روش‌های طبقه‌بندی متون سنتی و مبتنی بر یادگیری عمیق

جدول (۲): تفاوت روش‌های طبقه‌بندی سنتی و مبتنی بر یادگیری [۶۷]

روش‌های سنتی	روش‌های مبتنی بر یادگیری عمیق
کارایی خوب بر روی‌داده‌های کوچک	کارایی خوب بر روی‌داده‌های بزرگ
وابسته به روش‌های انتخاب ویژگی	مخفی و خودکار بودن روند استخراج ویژگی
زمان آموزش کم‌تر اما زمان آزمودن بیشتر	زمان آموزش بیش‌تر اما زمان آزمودن کمتر
فقط ویژگی‌های مستقیم را یاد می‌گیرند	طیف وسیعی از ویژگی‌ها از سطح پایین به سطح بالا را یاد می‌گیرند
نتایج قابل تفسیر	نتایج غیرقابل تفسیر
کند به دلیل دستگاه‌های پایین رده	برای سرعت بخشیدن به کارایی از GPU استفاده می‌کنند
کم‌هزینه	پرهزینه

۳. شبکه‌های عصبی عمیق برای طبقه‌بندی متون فارسی

در این مقاله دو شبکه عصبی عمیق برای طبقه‌بندی متون فارسی با استفاده از شبکه عصبی پیچشی به نام ParsCNN و همچنین شبکه عصبی LSTM دوسویه با لایه توجه به نام ParsBi-LSTM ارائه شده است که در ادامه به شرح آنها خواهیم پرداخت.

۳,۱. شبکه عصبی پیچشی برای طبقه‌بندی متون فارسی

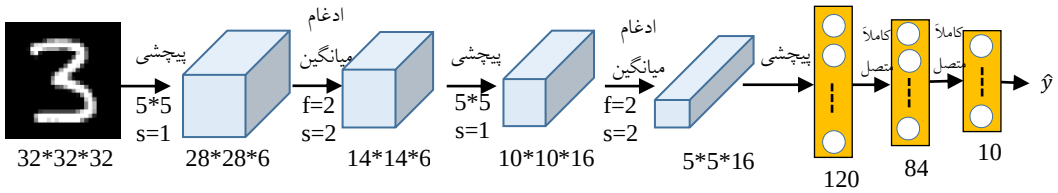
زمانی که از شبکه‌های عصبی پیچشی سخن به میان می‌آید به طور معمول اذهان به سمت بینایی کامپیوتر[1] سوق می‌یابد [۶۸]، در حالی که با وجود CNNها پیشرفت‌های زیادی در زمینه بینایی کامپیوتر [۶۹] به خصوص طبقه‌بندی تصاویر صورت گرفته است، اخیرا نیز تلاش‌هایی برای حل مشکلات پردازش زبان طبیعی با استفاده از CNNها صورت گرفته است. به طور کلی شبکه‌های عصبی پیچشی نوعی شبکه‌های عصبی مصنوعی هستند که از سه نوع اصلی لایه‌ها برای ساخت یک CNN استفاده می‌کنند که این لایه‌ها عبارت‌اند از: لایه پیچشی،لایه ادغام و لایه کاملا متصل. اولین شبکه عصبی پیچشی با معماری LeNet-5 شروع شده است. معماری شبکه پیچشی LeNet-5 [۷۰] مطابق شکل (۱۰) است. در LeNet-5 ابعاد ورودی ۱*۳۲*۳۲ است و دو لایه پیچشی با گام ۲ و لایه ادغام میانگین[2] با گام ۱ قرار دارد. در انتهای شبکه لایه‌های کاملا متصل با تابع فعال‌سازی بیشینه هموار در لایه خروجی قرار دارند. تعداد پارامترهای این شبکه ۶۰۰۰۰ است. در شبکه‌های پیچشی گرایش به سمت فیلترهای کوچک‌تر و ساختارهای عمیق‌تر و حذف کامل لایه‌های ادغام و لایه‌های کاملا متصل است. یک الگوی رایج برای ساختار شبکه‌های پیچشی به صورت رابطه (۴) است.

$$[(Conv - RELU) * N - POOL?] * M - (FC - RELU) * K, SOFTMAX \quad (۴)$$

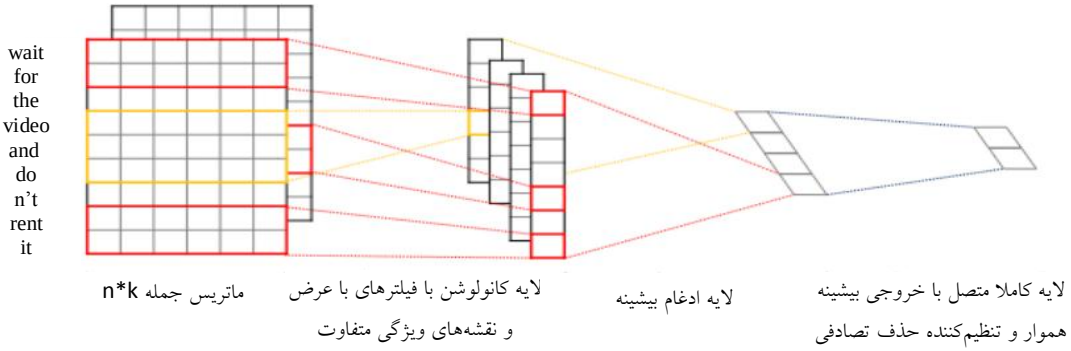
به طوری که N حداکثر برابر ۵، M یک مقدار بزرگ و K عددی صحیح مابین ۰ و ۲ است. البته پیشرفت‌های اخیر همچون ResNet و GoogleNet این الگو را به چالش کشیده‌اند.

در مساله‌های پردازش زبان طبیعی ورودی شبکه‌های CNN جملات یا اسناد می‌باشند که به شکل یک ماتریس نمایش داده می‌شوند که هر سطر ماتریس مربوط به یک توکن، به طور معمول یک کلمه یا کاراکتر، است؛ به عبارت دیگر، هر سطر ماتریس برداری است که یک کلمه را نمایش می‌دهد که به این بردارها تعبیه کلمات[3] گفته می‌شود. مدلی که در مرجع [۵۰] ارائه شده است یک مدل زبانی عصبی بدون نظارت است که یک شبکه عصبی پیچشی با یک لایه پیچشی را آموزش می‌دهد. در این مقاله ما با الهام گرفتن از این روش، شبکه عصبی پیچشی برای طبقه‌بندی متون فارسی به نام ParsCNN را پیاده‌سازی نمودیم. مدل شبکه به صورت شکل (۱۱) می‌باشد که در لایه اول کلمات به صورت بردارهای با ابعاد کم تبدیل می‌شوند. در لایه بعدی با استفاده از فیلترهای با ابعاد متفاوت پیچشی بر روی بردارهای کلمات انجام می‌شود. سپس با استفاده از لایه ادغام بیشینه نتایج لایه پیچشی به یک بردار ویژگی طولانی تبدیل شده و تنظیم‌کننده حذف تصادفی به آن اضافه می‌شود و با استفاده از لایه بیشینه هموار نتایج کلاس‌بندی می‌شود. مدل ParsCNN متن را به عنوان یک تصویر یک‌بعدی پردازش می‌کند و یک CNN تک‌لایه برای ضبط ارتباطات نهفته بین کلمات همسایه استفاده می‌شود. مدل ParsCNN هیچ‌گونه پیش‌پردازشی بر روی داده‌های ورودی انجام نمی‌دهد و ما را از وابسته بودن به تمامی روش‌های پیش‌پردازش زبان فارسی که یکی از دغدغه‌ها و چالش‌های مطرح در زبان فارسی است بی‌نیاز می‌سازد. نتایج روش‌های طبقه‌بندی وابسته به روش‌های پیش‌پردازش نبوده و همچنین زمانی صرف عملیات پیش‌پردازش نمی‌شود. در حقیقت این ویژگی اجازه می‌دهد تا طبقه‌بندی متون بدون داشتن هیچ دانشی از روش‌های پیش‌پردازش و نقاط ضعف و قوت آنها انجام شود.

[1] Computer vision
[2] Average Pooling
[3] word embedding



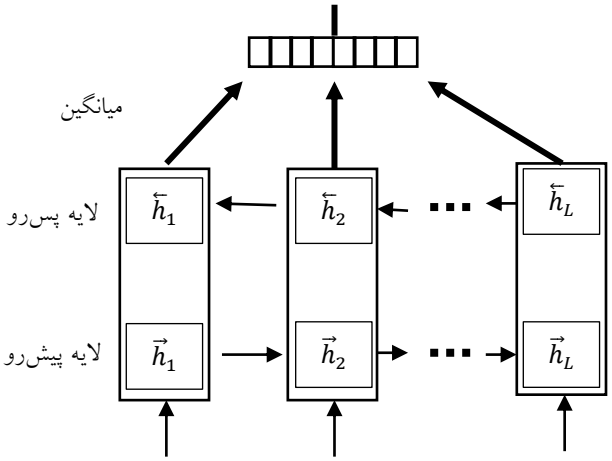
شکل (۱۰): معماری شبکه عصبی پیچشی LeNet-5 [۷۰]



شکل (۱۱): معماری شبکه عصبی پیچشی برای طبقه‌بندی متون [۵۰]

۳,۲. شبکه عصبی LSTM دوسویه سلسه‌مراتبی با لایه توجه

در این روش یک مدل سلسله‌مراتبی LSTM دوسویه با لایه توجه برای طبقه‌بندی متون فارسی با نام ParsBi-LSTM با الهام از روش مرجع [۷۱] ارائه شده است. معماری کلی شبکه‌های LSTM [۷۲]، [۷۳] همانند RNNها هستند با این تفاوت که به‌روزرسانی لایه پنهان با سلول‌های حافظه جایگزین می‌شود. در نتیجه، در یادگیری وابستگی‌های بلندمدت بهتر عمل می‌کنند. توانایی قدرتمند LSTM در استخراج اطلاعات پیشرفته متن، نقش مهمی در طبقه‌بندی متن دارد. در مدل LSTM معمولی تنها مقادیر دنباله‌های گذشته در محاسبه خروجی لایه مخفی به کار گرفته می‌شوند. لیکن در LSTM دوسویه [۷۴] می‌توان از هر دو ویژگی گذشته و آینده در یک زمان معین استفاده نمود. در این معماری از دو واحد مجزا استفاده می‌شود که یکی وظیفه محاسبه خروجی با توجه به دنباله آینده را بر عهده دارد ($\overleftarrow{h}$) و دیگری وظیفه محاسبه خروجی با توجه به دنباله گذشته را بر عهده دارد ($\overrightarrow{h}$)، در نهایت این خروجی‌ها با هم ترکیب شده و بازنمایی کلی جمله به دست می‌آید (شکل (۱۲)). مدل ParsBi-LSTM از چندین قسمت تشکیل شده است: کدگذار توالی کلمه[1]، لایه توجه در سطح کلمه[2]، کدگذار جمله[3] و لایه توجه در سطح جمله[4] (شکل (۱۳)).

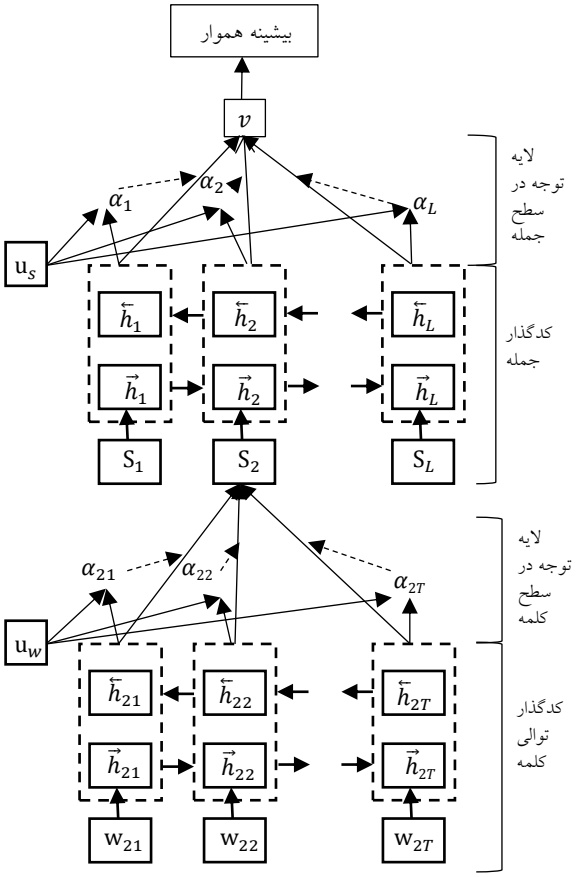


شکل (۱۲): LSTM دوسویه

کدگذار توالی کلمه: فرض کنید یک سند دارای L جملات s_i است و هر جمله حاوی کلمات T_i است. w_{it} با $t \in [1,T]$ نشان‌دهنده کلمات در جمله i است. ورودی این ماژول کلمات یک جمله است، ابتدا بردار تعبیه کلمات سند ایجاد می‌شوند. سپس با استفاده از LSTM دوسویه حاشیه‌نویسی از کلمات به

[1] Word encoder
[2] Word attention
[3] Sentence encoder
[4] Sentence attention

دست می‌آید. در این مرحله یک حاشیه‌نویسی $h_{it} = [\overrightarrow{h}_{it}, \overleftarrow{h}_{it}]$ برای کلمه w_{it} به دست می‌آید که اطلاعات کل جمله را حول محور w_{it} خلاصه می‌کند.



شکل (۱۳): ساختار شبکه عصبی LSTM دوسویه سلسه‌مراتبی با لایه توجه [۷۱]

لایه توجه در سطح کلمه: تمامی کلمات در بازنمایی معنای جمله مشارکت یکسانی ندارند. از این رو، مکانیسم توجه برای استخراج کلماتی که در معنی جمله تاثیر بیشتری دارند و تجمیع بازنمایی‌های این کلمات برای تشکیل بردار جمله استفاده می‌شود.

$$u_{it} = \tanh(W_w h_{it} + b_w) \quad (۵)$$

$$a_{it} = \frac{\exp(u_{it}^T u_w)}{\sum_t \exp(u_{it}^T u_w)} \quad (۶)$$

$$s_i = \sum_t a_{it} h_{it} \quad (۷)$$

ابتدا حاشیه‌نویسی کلمه h_{it} وارد یک پرسپترون چند لایه[1] تک لایه می‌شود و u_{it} را به عنوان نمایشی پنهان از h_{it} دریافت می‌کنیم، سپس اهمیت کلمه را به عنوان شباهت u_{it} با بردار زمینه کلمه[2] u_w اندازه‌گیری می‌کنیم، وزن اهمیت نرمال‌شده a_{it} از خروجی تابع بیشینه هموار به دست می‌آید. پس از آن بردار جمله s_i را به عنوان مجموع وزن‌دار از حاشیه‌نویسی کلمات بر اساس وزن‌ها محاسبه می‌کنیم. u_w به طور تصادفی مقداردهی اولیه می‌شود و به طور مشترک در طول فرآیند آموزش آموخته می‌شود.

کدگذار جمله: بردار جملات s_i با استفاده از LSTM دوسویه جملات کدگذاری می‌شوند و یک بازنمایی از جملات به عنوان خروجی ایجاد می‌شود. h_i جملات همسایه را در اطراف جمله i خلاصه می‌کند اما همچنان روی جمله i تمرکز می‌کند.

لایه توجه در سطح جمله: در این مرحله به جملاتی که تاثیر بیشتری در طبقه‌بندی سند دارند توجه بیشتری می‌شود و خروجی v بازنمایی سطح بالایی از سند است که تمام اطلاعات همه جملات سند در آن خلاصه شده است.

$$u_i = \tanh(W_s h_i + b_s) \quad (۸)$$

$$a_i = \frac{\exp(u_i^T u_s)}{\sum_t \exp(u_i^T u_s)} \quad (۹)$$

$$v = \sum_i a_i h_i \quad (۱۰)$$

در روابط بالا، h_i، u_i، a_i و u_s به‌ترتیب برابر حاشیه‌نویسی جمله i، نمایش پنهان از h_i، وزن اهمیت نرمال‌شده جمله i و بردار زمینه سطح جمله می‌باشند. به طور مشابه، بردار زمینه سطح جمله u_s را می‌توان به طور تصادفی مقداردهی اولیه کرد و به طور مشترک در طول فرآیند آموزش یاد گرفت. خروجی لایه توجه در سطح جمله به عنوان ویژگی برای طبقه‌بندی سند استفاده می‌شود که برای این امر از تابع بیشینه هموار استفاده می‌شود.

این مدل، به تدریج با تجمیع کلمات مهم به بردارهای جمله و

[1] Multi Layer Perceptron (MLP)
[2] word context vector

سپس تجمیع بردارهای جملات مهم به بردارهای سند، یک بردار سند را می‌سازد. مدل ParsBi-LSTM دارای دو ویژگی است: (۱) ساختار سلسه‌مراتبی دارد که نشان‌دهنده ساختار سلسه‌مراتبی اسناد است به این معنی که کلمات، جملات را تشکیل می‌دهند و جملات، یک سند را می‌سازند. (۲) سطح مکانیسم توجه در دو سطح کلمه و جمله اعمال می‌شود. مکانیسم توجه به عنوان یک فن نوظهور برای افزایش کارایی پیشنهاد شده است. مکانیسم توجه ما را قادر می‌سازد که در ساخت بازنمایی از سند بین محتوای کم‌اهمیت و پراهمیت تفاوت قائل شویم.

۴. مجموعه داده‌ها و پیاده‌سازی

در این بخش از مقاله به بیان مجموعه داده مورد استفاده، معیارهای ارزیابی و جزئیات پیاده‌سازی می‌پردازیم.

۴،۱. مجموعه داده همشهری

مجموعه داده همشهری [۷۵] پیکره‌ای است که با خزش[1] وبگاه همشهری و چندین مرحله پیش‌پردازش و برچسب‌زنی حاصل شده است. این مجموعه داده مربوط به متن اخبار و گروه‌های خبری منتشر شده در روزنامه همشهری مربوط به سال‌های ۱۳۷۵ تا ۱۳۸۷ می‌باشد. همه اسناد مجموعه همشهری دارای برچسب هستند که نشان می‌دهد هر سند در چه رده‌ای شامل اقتصادی، سیاسی و ... است. این پیکره توسط گروه تحقیقاتی پایگاه داده دانشگاه تهران و با حمایت مرکز تحقیقات مخابرات ایران تهیه شده است [۷۶]. در مجموعه داده همشهری نسخه یک، ۱۶۶ هزار خبر با دسته‌بندی‌های موضوعی؛ سیاست، اخبار شهری، اقتصاد، گزارش‌ها، سرمقاله‌ها، ادبیات، علوم، جامعه، اخبار خارجی، ورزش و غیره وجود دارد. حجم اسناد از اخبار کوتاه (زیر یک کیلو بایت) تا اخبار به نسبت طولانی (۱۴۰ کیلو بایت) با میانگین اندازه متوسط ۱/۸ کیلوبایت متفاوت است. در این مقاله، آزمایش‌ها بر روی مجموعه داده‌های همشهری که یکی از معتبرترین منابع فارسی‌زبان در زمینه زبان طبیعی، برای استخراج ویژگی‌ها و طبقه‌بندی متن انجام می‌شود.

۴،۲. معیارهای ارزیابی

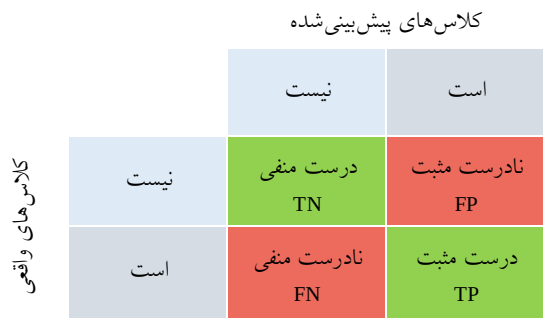
در مقالات مختلف معیارهای ارزیابی مختلفی ارائه شده است که در این مقاله برای ارزیابی نتایج خود و روش‌های پیشنهادی دیگر از سه معیار رایج دقت[2]، فراخوانی[3] و مقیاس F-[4] استفاده خواهیم نمود که به‌ترتیب می‌توان با استفاده از روابط (۱۱)، (۱۲) و (۱۳) محاسبه کرد.

$$precision = \frac{TP}{TP + FP} \quad (۱۱)$$

$$recall = \frac{TP}{TP + FN} \quad (۱۲)$$

$$F - measure = 2 * \frac{recall * precision}{recall + precision} \quad (۱۳)$$

که در آن TP[5] یا درست مثبت نشانگر تعداد اسنادی است که به‌درستی به عنوان مرتبط شناسایی شده‌اند؛ FP[6] یا نادرست مثبت تعداد اسنادی است که سیستم به غلط به عنوان مرتبط شناسایی کرده است؛ FN[7] یا نادرست منفی تعداد اسناد مرتبطی است که سیستم نتوانسته آنها را شناسایی کند و TN[8] یا درست منفی تعداد اسنادی است که سیستم به‌درستی به عنوان غیرمرتبط شناسایی کرده است. ماتریس درهم‌ریختگی یا اغتشاش[9] (شکل (۱۴)) این موارد را نمایش می‌دهد.



شکل (۱۴): ماتریس درهم‌ریختگی یا اغتشاش

[1] Crawl

[2] Precision
[3] Recall
[4] F-Measure
[5] True Positive
[6] False Positive
[7] False Negative
[8] True Negative
[9] Confusion Matrix

به عبارت بهتر، دقت، حاصل تقسیم تعداد متن‌های طبقه‌بندی شده درست مثبت بر تعداد کل متن‌های طبقه‌بندی شده به عنوان مثبت است. فراخوانی، حاصل تقسیم تعداد متن‌های طبقه‌بندی شده درست مثبت، بر کل متن‌های مثبت مجموعه آموزشی می‌باشد. مقیاس-F نیز میانگین هارمونیک دقت و فراخوانی می‌باشد.

۴،۳. پیاده‌سازی

در پیاده‌سازی شبکه عصبی پیچشی ParsCNN از تابع فعالیت غیرخطی واحد یک‌سوشده‌ی خطی[1]، تابع هزینه آنتروپی متقاطع[2] و قانون بهینه‌سازی Adam استفاده‌شده است. همچنین مدل نمایش کلمات مورداستفاده CNN-rand است که در آن تمامی کلمات به‌صورت تصادفی مقداردهی اولیه می‌شوند و در طول آموزش تغییر می‌کنند. همچنین برخلاف مقاله اصلی که از دو کانال داده ورودی، بردارهای کلمات استاتیک و غیر استاتیک استفاده کرده بود، در این مقاله ما فقط از یک کانال استفاده نمودیم. جزئیات پیاده‌سازی در جدول ۳ آمده است.

جدول (۳): جزئیات پیاده‌سازی شبکه عصبی پیچشی ParsCNN

متغیر	مقدار
اندازه فیلترها	۳، ۴ و ۵
تعداد فیلترها	۱۲۸
نرخ حذف تصادفی در آموزش	۰/۵
نرخ حذف تصادفی در آزمون	صفر (غیرفعال کردن حذف تصادفی)
اندازه دسته[3]	۶۴
تعداد دوره[4]ها	۳۰
ابعاد Embedding	۲۰۰
تعداد checkpoints	۵

در شبکه LSTM دوسویه سلسله‌مراتبی با لایه توجه ParsBi-LSTM بردار تعبیه کلمات به صورت تصادفی مقداردهی اولیه می‌شوند. جزئیات پیاده‌سازی این شبکه در جدول (۴) نمایش داده شده است.

جدول (۴): جزئیات پیاده‌سازی شبکه عصبی LSTM دوسویه سلسله‌مراتبی با لایه توجه ParsBi-LSTM

متغیر	مقدار
ابعاد Embedding	۱۰۰
ابعاد BiLSTM	۲۰۰
اندازه دسته	۶۴
نرخ یادگیری	۰/۰۰۳
تعداد دوره‌ها	۲۵

۴،۳،۱. نتایج پیاده‌سازی

پس از پیاده‌سازی الگوریتم‌ها، کارایی آنها بر اساس معیارهای ارزیابی مورد مقایسه قرار می‌گیرد. نتایج به دست آمده برای دو روش پیشنهادی ParsCNN و ParsBi-LSTM بدون انجام پیش‌پردازش بر روی متون است که این امر باعث شده است که این روش‌ها وابسته به ابزارهای پیش‌پردازش متون نباشند و زمانی برای پیش‌پردازش و همچنین یافتن مناسب‌ترین روش پیش‌پردازش صرف نشود. دو روش مبتنی بر شبکه عصبی عمیق را با روش‌های طبقه‌بندی سنتی SVM و بیز ساده که دو الگوریتم یادگیری ماشین موفق هستند و در ارزیابی‌های مقالات بسیاری مورد ارجاع قرار گرفته است [۴۹]، [۵۱]، [۷۷]، [۷۸] مقایسه می‌نماییم. در آزمایش‌ها نمایش اسناد را به دو صورت: ۱) در سطح کاراکتر و بر اساس مدل فراوانی عبارت و ۲) مدل ۲-بخشی[5] در نظر می‌گیریم. همچنین روش‌های دیگری که روش‌های ما با آنها مقایسه شده‌اند روش‌های NB & SemanticKernel+SVM [۴۸] و Char+CNN [۶۲] هستند که به‌ترتیب در بخش‌های ۲،۱،۱ و ۲،۲،۲ به توضیح آنها پرداختیم. همانگونه که در جدول (۵) مشاهده می‌کنید روش‌های مبتنی بر یادگیری عمیق نسبت به روش‌های سنتی معیارهای ارزیابی موفقیت را بهبود بخشیده است. هر دو روش پیشنهادی نسبت به سایر روش‌ها نتایج بهتری کسب نموده‌اند. مدل شبکه عصبی

[1] Rectified Linear Unit (ReLU)
[2] cross entropy
[3] Batch
[4] Epoch
[5] Bigram

LSTM دوسویه سلسله‌مراتبی با لایه توجه ParsBi-LSTM بهترین نتایج را به دست آورده است. اگرچه BiLSTM می‌تواند اطلاعات متنی را به دست آورد، اما تمرکز روی اطلاعات مهم در اطلاعات متنی به‌دست‌آمده امکان‌پذیر نیست. تمرکز بر روی اطلاعات مهم باعث بهبود کارایی طبقه‌بندی می‌شود که این امر با افزودن مکانیسم توجه به روش ParsBi-LSTM میسر شده است. مکانیسم توجه می‌تواند با تعیین وزن‌های مختلف، اطلاعات مهم را از اطلاعات متنی برجسته کند. ترکیبی از BiLSTM و مکانیسم توجه کارایی طبقه‌بندی را بیشتر بهبود بخشیده است. همچنین روش ParsBi-LSTM برخلاف سایر روش‌ها از ساختار سلسله‌مراتبی اسناد بهره برده است که این امر نیز در بهبود نتایج بی‌تاثیر نمی‌باشد. با توجه به مجموعه داده مورد استفاده در ارزیابی‌ها روش ParsBi-LSTM نتایج به نسبت بهتری از ParsCNN نشان داده است. دو روش ParsCNN و ParsBi-LSTM به‌ترتیب شبکه CNN و LSTM (نوعی شبکه RNN) هستند و تمامی ویژگی‌های این شبکه‌ها را دارند. RNNها و متعاقب آن LSTMها برای تشخیص الگوها در طول زمان آموزش می‌بینند، در حالی که CNNها یاد می‌گیرند که الگوها را در سراسر فضا تشخیص دهند [۷۰]؛ به عبارت دیگر، مدل‌های مبتنی بر LSTMها، متن را به عنوان دنباله‌ای از کلمات مشاهده می‌کنند و وابستگی‌های واژه‌ها و ساختارهای متن را دریافت می‌کنند. مدل‌های مبتنی بر CNN بیشتر برای تشخیص الگوهای متن، مانند عبارات کلیدی، برای طبقه‌بندی متون آموزش دیده‌اند. RNNها برای وظایف NLP مانند برچسب‌گذاری POS یا پرسش و پاسخ که درک معنایی دوربرد مورد نیاز است، به‌خوبی کار می‌کنند، در حالی که CNNها در جایی که تشخیص الگوهای محلی و تغییرناپذیر موقعیت مهم است، به‌خوبی کار می‌کنند. این الگوها می‌توانند عبارات کلیدی باشند که احساسات خاصی مانند «من دوست دارم» یا موضوعی مانند «گونه‌های در حال انقراض» را بیان می‌کنند[۶۳]. از دیگر ویژگی‌های CNNها می‌توان به موازی‌سازی آسان در پردازنده‌های گرافیکی و سریع‌تر بودن نسبت به RNNها اشاره کرد؛ بنابراین با توجه به مزیت‌هایی که هر کدام از شبکه‌های CNN و LSTM دارند و میزان کارایی نزدیک به هم دو روش پیشنهادی ParsCNN و ParsBi-LSTM (جدول (۵)) می‌توان یکی از این دو روش را با در نظر گرفتن کاربردی که طبقه‌بندی متون برای آن استفاده می‌شود همچون تجزیه‌وتحلیل احساسات، طبقه‌بندی اخبار، پرسش و پاسخ و استنتاج زبان طبیعی و ... استفاده نمود.

جدول (۵): مقایسه کارایی روش‌های طبقه‌بندی متون فارسی

مدل	مرجع	دقت	فراخوانی	مقیاس-F
Char.p+NB	---	۰/۳۷	۰/۲۹	۰/۳۲
Bigram.TF+NB	[۷۸]	۰/۳۸	۰/۲۸	۰/۳۲
Char.TF+SVM	---	۰/۴۵	۰/۴۵	۰/۴۵
Bigram.TF+SVM	[۷۸]	۰/۴۳	۰/۳۸	۰/۴۰
SemanticKernel+NB	[۴۸]	۰/۳۹	۰/۴۰	۰/۳۹
SemanticKernel+SVM	[۴۸]	۰/۴۷	۰/۴۶	۰/۴۶
Char+CNN	[۶۲]	۰/۵۱	۰/۵۰	۰/۵۰
ParsCNN	---	۰/۶۹	۰/۷۰	۰/۶۹
ParsBi-LSTM	---	۰/۷۲	۰/۷۳	۰/۷۲

۵. نتیجه‌گیری

در گذشته اکثر کارهایی که برای طبقه‌بندی متون انجام گرفته است بر روی متون زبان انگلیسی بوده است. در زبان فارسی کارهای اندکی که ارائه شده است اکثرا مبتنی بر انتخاب ویژگی و یادگیری ماشین بوده است که در مقایسه با روش‌های نوین مبتنی بر یادگیری عمیق نتایج خوبی را ارائه نمی‌نمایند. در این مقاله دو روش مبتنی بر یادگیری عمیق برای طبقه‌بندی متون فارسی پیاده‌سازی شده است که شامل یک شبکه عصبی پیچشی با یک لایه پیچشی به نام ParsCNN و شبکه عصبی LSTM دوسویه سلسله‌مراتبی با لایه توجه به نام ParsBi-LSTM می‌باشد. نتایج آزمایش‌ها نشان می‌دهد که این دو روش نسبت به روش‌های پیشین دقت و فراخوانی بالاتری دارند. روش ParsCNN قادر به یادگیری پاسخ محلی از داده‌های زمانی یا مکانی است اما توانایی یادگیری همبستگی‌های متوالی را ندارد. از سوی دیگر، روش ParsBi-LSTM برای مدل‌سازی متوالی

مشخص شده است، اما قادر به استخراج ویژگی‌ها به صورت موازی نیست.

تعارض منافع: نویسندگان اعلام می‌کنند که هیچ تعارض منافعی ندارند.

مراجع

[1] M. Nabiloo and N. Daneshpour, "A clustering algorithm for categorical data with combining measures," Soft Comput. J., vol. 5, no. 1, pp. 14-25, 2016 [In Persian].

[2] A. Khosravi, H. Abdulmaleki, and M. Fayazi, "Predicting the academic status of admitted applicants based on educational and admission data using data mining techniques," Soft Comput. J., vol. 9, no. 2, pp. 94-113, 2021, doi: 10.22052/scj.2021.242837.0 [In Persian].

[3] J.F. Allen, Natural language processing, in Encyclopedia of computer science, pp. 1218–1222, 2003.

[4] J. Eisenstein, Introduction to natural language processing, MIT press, 2019.

[5] M.K. Dalal and M.A. Zaveri, "Automatic text classification: a technical review," Int. J. Comput. Appl., vol. 28, no. 2, pp. 37-40, 2011, doi: 10.5120/3358-4633.

[6] A.I. Kadhim, "Survey on supervised machine learning techniques for automatic text classification," Artif. Intell. Rev., vol. 52, no. 1, pp. 273-292, 2019, doi: 10.1007/S10462-018-09677-1.

[7] A. Addis, "Study and Development of Novel Techniques for Hierarchical Text Categorization," PhD Thesis, Electrical and Electronic Engineering Dept., University of Cagliari, Italy, 2010.

[8] Y. Bengio, I. Goodfellow, and A. Courville, Deep learning, vol. 1, MIT press Massachusetts, USA, 2017.

[9] J.D. Kelleher, Deep learning, MIT press, 2019.

[10] W.T. Yih, K. Toutanova, J.C. Platt, and C. Meek, "Learning discriminative projections for text similarity measures," in Proc. 5th Conf. Comput. Natural Lang. Learn., Portland, Oregon, USA, June 23-24, 2011, pp. 247–256.

[11] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model", J. Mach. Learn. Res., vol. 3, pp. 1137-1155, 2003.

[12] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Advances in neural information processing systems, Lake Tahoe, Nevada, United States, 2013, pp. 3111–3119.

[13] J.A. Bullinaria and J.P. Levy, "Extracting semantic representations from word co-occurrence statistics: A computational study," Behav. Res. methods, vol. 39, no. 3, pp. 510–526, 2007, doi: 10.3758/BF03193020.

[14] R. Cohen, Y. Goldberg, and M. Elhadad, "Domain adaptation of a dependency parser with a class-class selectional preference model," in Proceedings of ACL 2012 Student Research Workshop, 2012, pp. 43–48.

[15] A. Ritter and O. Etzioni, "A latent dirichlet allocation method for selectional preferences," in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 2010, pp. 424–434.

[16] D.O. Seaghdha, "Latent variable models of selectional preference", in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 2010, pp. 435–444.

[17] M.T. Pilehvar and J. Camacho-Collados, "Embeddings in Natural Language Processing: Theory and Advances in Vector Representations of Meaning," Synthesis Lectures on Human Language Technologies, vol. 13, no. 4, pp. 1–175, 2020, doi: 10.2200/S01057ED1V01Y202009HLT047.

[18] Y. Lee, H. Ke, T. Yen, H. Huang, and H. Chen, "Combining and learning word embedding with WordNet for semantic relatedness and similarity measurement," J. Assoc. info. Sci. Technol., vol. 71, no. 6, pp. 657–670, 2020, doi: 10.1002/ASI.24289.

[19] K. Cho, B. Van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Doha, Qatar, 2014, pp. 1724-1734, doi: 10.3115/V1/D14-1179.

[20] H. Xu, J. van Genabith, D. Xiong, Q. Liu, and J. Zhang, "Learning Source Phrase Representations for Neural Machine Translation," Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL, 2020, pp. 386-396, doi: 10.18653/v1/2020.acl-main.37.

[21] C. dos Santos and M. Gatti, "Deep convolutional neural networks for sentiment analysis of short texts," in Proceedings of COLING 2014, the 25th International Conference on Computational

Linguistics: Technical Papers, Dublin, Ireland, 2014, pp. 69–78.

[22] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 2015, pp. 959-962, doi: 10.1145/2766462.2767830.

[23] A. Agarwal, A. Yadav, and D.K. Vishwakarma, "Multimodal sentiment analysis via RNN variants," in 2019 IEEE International Conference on Big Data, Cloud Computing, Data Science and Engineering (BCD), Honolulu, HI, USA, 2019, pp. 19–23, doi: 10.1109/BCD.2019.8885108.

[24] B.N. Saha, A. Senapati, and A. Mahajan, "LSTM based Deep RNN Architecture for Election Sentiment Analysis from Bengali Newspaper," in International Conference on Computational Performance Evaluation (ComPE), Shillong, India, 2020, pp. 564-569, doi: 10.1109/ComPE49325.2020.9200062.

[25] M.E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U.R. Acharya, "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis," Future Gener. Comput. Syst., vol. 115, pp. 279–294, 2021, doi: 10.1016/j.future.2020.08.005.

[26] P.F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment analysis using Word2Vec and long short-term memory (LSTM) for Indonesian hotel reviews," Procedia Comput. Sci., vol. 179, pp. 728–735, 2021, doi: 10.1016/j.procs.2021.01.061.

[27] S. Poria, E. Cambria, and A. Gelbukh, "Aspect extraction for opinion mining with a deep convolutional neural network," Knowl. Based Syst., vol. 108, pp. 42–49, 2016, doi: 10.1016/j.knosys.2016.06.009.

[28] T. Singh, A. Nayyar, and A. Solanki, "Multilingual opinion mining movie recommendation system using RNN," in Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019), Springer, Singapore,2020, pp. 589–605, doi: 10.1007/978-981-15-3369-3_44.

[29] T. Mikolov, S. Kombrink, L. Burget, J. Cernocky, and S. Khudanpur, "Extensions of recurrent neural network language model," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague Congress Center, Prague, Czech Republic, 2011, pp. 5528–5531, doi: 10.1109/ICASSP.2011.5947611.

[30] A. Mnih and G.E. Hinton, "A scalable hierarchical distributed language model," in Advances in neural information processing systems, British Columbia, Canada, 2008, pp. 1081–1088.

[31] 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, Helsinki, Finland, 2008, pp. 160–167, doi: 10.1145/1390156.1390177.

[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in 1st International Conference on Learning Representations, ICLR, Scottsdale, Arizona, USA, 2013.

[33] J.Pennington, R. Socher, and C.D. Manning, "Glove: Global vectors for word representation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Doha, Qatar, 2014, pp. 1532–1543, doi: 10.3115/v1/d14-1162.

[34] W. Zhang, X. Tang, and T. Yoshida, "Tesc: An approach to text classification using semi-supervised clustering," Knowl. Based Syst., vol. 75, pp. 152–160, 2015, doi: 10.1016/j.knosys.2014.11.028.

[35] M. Pavlinek and V. Podgorelec, "Text classification method based on self-training and LDA topic models," Expert Syst. Appl., vol. 80, pp. 83–93, 2017, doi: 10.1016/j.eswa.2017.03.020.

[36] B. Agarwal and N. Mittal, "Text classification using machine learning methods-a survey," in Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS), Jaipur, India,2014, pp. 701–709, New Delhi, doi: 10.1007/978-81-322-1602-5_75.

[37] S.H. Mohammed and S. Al-augby, "LSA & LDA Topic Modeling Classification: Comparison study on E-books," Indonesian J. Electr. Eng. Comput. Sci., vol. 19, no. 1, pp. 353–362, 2020, doi: 10.11591/ijeecs.v19.i1.pp353-362.

[38] W. Wang, B. Guo, Y. Shen, H. Yang, Y. Chen, and X. Suo, "Twin labeled LDA: a supervised topic model for document classification", Appl. Intell., vol. 50, no. 12, pp. 4602–4615, 2020, doi: 10.1007/s10489-020-01798-x.

[39] L. Xu, Z. Xue, and H. Huang, "Short text semantic feature extension and classification based on LDA," in IOP Conf. Ser.: Mater. Sci. Eng., Shanghai, China, 2020, vol. 715, no. 1, p. 12110, doi: 10.1088/1757-899X/715/1/012110.

[40] P. Zhang and Y. Fang, "Research on Text Classification Algorithm Based on Machine Learning," in Journal of Physics: Conference Series, vol. 1624, no. 4, p. 42010, 2020, doi: 10.1088/1742-6596/1624/4/042010.

[41] K. Shah, H. Patel, D. Sanghvi, and M. Shah, "A comparative analysis of logistic regression, random forest and KNN models for the text classification," Augment. Hum. Res., vol. 5, no. 1, pp. 1–16, 2020, doi: 10.1007/s41133-020-00032-0.

[42] M. Parchami, B. Akhtar, and M. Dezfoulian, "Persian text classification based on K-NN using wordnet," in International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, Dalian, China,2012, pp. 283–291, doi: 10.1007/978-3-642-31087-4_30.

[43] J. Pouramini and B. Minaei-Bidgoli, "A New Synthetic Oversampling Method Using Ontology and Feature Selection in Order to Improve Imbalanced Textual Data Classification in Persian Texts," Bulletin de la Societe Royale des Sciences de Liege, vol. 85, pp. 358–375, 2016, doi: 10.25518/0037-9565.5414.

[44] M. Farhoodi and A. Yari, "Applying machine learning algorithms for automatic Persian text classification," in 6th International Conference on Advanced Information Management and Service (IMS), Seoul, Korea (South), 2010, pp. 318–323.

[45] N. Rezaeian and G. Novikova, "Persian Text Classification using naive Bayes algorithms and Support Vector Machine algorithm," Indonesian J. Electr. Eng. Informatics, vol. 8, no. 1, pp. 178–188, 2020, doi: 10.52549/ijeei.v8i1.1696.

[46] S. Mostafavi, B. Pahlevanzadeh, and M.R. Falahati Qadimi Fumani, "Classification of Persian News Articles using Machine Learning Techniques," Comput. Knowl. Eng., vol. 4, no. 1, pp. 1-10, 2021, doi: 10.22067/cke.2021.69212.1004.

[47] M. Farhoodi, A. Yari, and A. Sayah, "N-gram based text classification for Persian newspaper corpus," in The 7th International Conference on Digital Content, Multimedia Technology and its Applications, Busan, Korea (South), 2011, pp. 55–59.

[48] A.H. Jadidinejad and V. Marza, "Building Semantic Kernel for Persian Text Classification with a Small Amount of Training Data," J. Adv. Comput. Res., vol. 6, no. 1, pp. 125–136, 2015.

[49] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolutional neural networks," in the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, 2015, pp. 103-112, doi: 10.3115/v1/n15-1011.

[50] Y. Kim, "Convolutional neural networks for sentence classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Doha, Qatar, 2014, pp. 1746-1751, doi: 10.3115/v1/d14-1181.

[51] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, Baltimore, MD, USA, 2014, pp. 655-665,doi: 10.3115/v1/p14-1062.

[52] C. Zhou, C. Sun, Z. Liu, and F. Lau, "A C-LSTM neural network for text classification", arXiv Prepr. arXiv1511.08630, 2015.

[53] B. Jang, M. Kim, G. Harerimana, S. Kang, and J.W. Kim, "Bi-LSTM model to increase accuracy in text classification: combining Word2vec CNN and attention mechanism," Appl. Sci., vol. 10, no. 17, p. 5841, 2020, doi: 10.3390/app10175841.

[54] Z. Liu, H. Huang, C. Lu, and S. Lyu, "Multichannel CNN with Attention for Text Classification," arXiv Prepr. arXiv2006.16174, 2020.

[55] P. Wang, J. Xu, B. Xu, C. Liu, H. Zhang, F. Wang, and H. Hao, "Semantic clustering and convolutional neural network for short text categorization," in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 2015, vol. 2, pp. 352–357, doi: 10.3115/v1/p15-2058.

[56] M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H. Daume III, "Deep unordered composition rivals syntactic methods for text classification," in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 2015, vol. 1, pp. 1681–1691, doi: 10.3115/v1/p15-1162.

[57] A.K. Sharma, S. Chaurasia, and D.K. Srivastava, "Sentimental Short Sentences Classification by Using CNN Deep Learning Model with Fine Tuned Word2Vec," Procedia Comput. Sci., vol. 167, pp. 1139–1147, 2020, doi: 10.1016/j.procs.2020.03.416.

[58] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in Advances in neural information processing systems, Montreal, Quebec, Canada, 2015, pp. 649–657.

[59] X. Zhang and Y. LeCun, "Text understanding from scratch", arXiv Prepr. arXiv1502.01710, 2015.

[60] A.N. Samatin Njikam and H. Zhao, "Chartec-net: An efficient and lightweight character-based convolutional network for text classification", J. Electr. Comput. Eng., vol. 2020, pp. 9701427:1-9701427:7, 2020, doi: 10.1155/2020/9701427.

[61] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," Science, vol. 344, no.

6191, pp. 1492–1496, 2014, doi: 10.1126/science.124207.

[62] S. Ghasemi and A.H. Jadidinejad, "Persian text classification via character-level convolutional neural networks," in 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium (IRANOPEN), Qazvin, Iran, 2018, pp. 1–6, doi: 10.1109/RIOS.2018.8406623.

[63] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning--based text classification: a comprehensive review," ACM Comput. Surv., vol. 54, no. 3, pp. 1–40, 2021, doi: 10.1145/3439726.

[64] R. Socher, A. Perelygin, J. Wu, J. Chuang, C.D. Manning, A.Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in Proceedings of the 2013 conference on empirical methods in natural language processing, Grand Hyatt Seattle, Seattle, Washington, USA, 2013, pp. 1631–1642.

[65] X. Li and D. Roth, "Learning question classifiers", in 19th International Conference on Computational Linguistics, Howard International House and Academia Sinica, Taipei, Taiwan, 2002.

[66] [66] Phan, X.-H., Nguyen, L.-M., Horiguchi, S., "Learning to classify short and sparse text & web with hidden topics from large-scale data collections", in Proceedings of the 17th international conference on World Wide Web, pp. 91–100, 2008.

[67] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey", Inf., vol. 10, no. 4, p. 150, 2019, doi: 10.3390/info10040150.

[68] A. Qanbari-Sorkhi, H. Hassanpour, M. Fateh, "Regions Proposal Selection in Objects Detection and Recogntion Systems," Soft Comput. J., vol. 5, no. 2, pp. 34-47, 2017, dor: 20.1001.1.23223707.1395.5.2.4.1 [In Persian].

[69] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," Commun. ACM, vol. 60, no. 6, pp. 84-90, 2017, doi: 10.1145/3065386.

[70] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278-2324, 1998, doi: 10.1109/5.726791.

[71] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, 2016, pp. 1480–1489, doi: 10.18653/v1/n16-1174.

[72] S. Hochreiter and J. Schmidhuber, "Long short-term memory", Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[73] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural Networks, vol. 18, no. 5–6, pp. 602–610, 2005, doi: 10.1016/j.neunet.2005.06.042.

[74] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in IEEE international conference on Acoustics, speech and signal processing, Vancouver, BC, Canada, 2013, pp. 6645–6649, doi: 10.1109/ICASSP.2013.6638947.

[75] A. AleAhmad, H. Amiri, E. Darrudi, M. Rahgozar, anf F. Oroumchian, "Hamshahri: A standard Persian text collection", Knowl. Based Syst., vol. 22, no. 5, pp. 382–387, 2009, doi: 10.1016/j.knosys.2009.05.002.

[76] Hamshahri, (2022), [Online]. Available: http://dbrg.ut.ac.ir/Hamshahri/

[77] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in roceedings of the 29th AAAI Conference on Artificial Intelligence, Austin, Texas, USA, 2015, pp. 2267-2273, doi: 10.1609/aaai.v29i1.9513.

[78] S.I. Wang and C.D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, Jeju Island, Korea, 2012, pp. 90–94.