

ارائه یک روش مبتنی بر شبکه عصبی پشته‌ای حلقوی برای تشخیص شایعه در شبکه‌های اجتماعی

حبیب ایزدخواه^{۱*}، دانشیار، رشید بهزادی دوست^۲، کاندید دکتری

^۱ دانشکده ریاضی، آمار و علوم کامپیوتر، دانشگاه تبریز، تبریز، ایران izadkhah@tabrizu.ac.ir

^۲ دانشکده ریاضی، آمار و علوم کامپیوتر، دانشگاه تبریز، تبریز، ایران r.behzadi@tabrizu.ac.ir

چکیده: امروزه شبکه‌های اجتماعی به عنوان یک ابزار مفید و پر استفاده شناخته می‌شوند. اگرچه جنبه‌های مثبت زیادی همانند تجارت الکترونیک، سرگرمی، افزایش آگاهی و ارتباط جهانی از طریق شبکه‌های اجتماعی وجود دارد، اما وجود شایعات جنبه منفی شبکه‌های اجتماعی است. با کمترین تلاش، یک شایعه می‌تواند به سرعت برای بسیاری از کاربران منتشر شود و منجر به پدیده‌های ناخوشایند اجتماعی شود. در کارهای قبلی، مدل‌هایی مبتنی بر یادگیری ماشین و یادگیری عمیق برای تشخیص خودکار شایعات ارائه شده‌است. در این مقاله، ما یک مدل چند شبکه عصبی پشته‌ای حلقوی را برای تشخیص شایعات در شبکه‌های اجتماعی ارائه می‌دهیم. از تمامی شبکه عصبی پیچشی پشته شده-شبکه عصبی حافظه طولانی کوتاه مدت دوجهته یک ماتریس پیچشی ایجاد می‌شود و سپس از پیچش ایجاد شده، حالت حلقوی ایجاد می‌شود. این مدل یادگیری عمیق فقط متن کاربران شبکه‌های اجتماعی را در نظر می‌گیرد و تشخیص می‌دهد آیا آن متن شایعه است یا خیر. ما مدل پیشنهادی را بر روی مجموعه داده‌های واقعی PHEME و Liar ارزیابی کرده‌ایم. مدل پیشنهادی نسبت به دوازده مدل یادگیری ماشین و یادگیری عمیق از لحاظ دقت بهتر عمل می‌کند.

واژه‌های کلیدی: پردازش زبان طبیعی؛ یادگیری عمیق؛ تشخیص شایعه.

* حبیب ایزدخواه، izadkhah@tabrizu.ac.ir

Circular multi-stacked CNN-BILSTM for rumor detection in social networks

Habib Izadkhah ^{1*}, Associate Professor, Rashid Behzadidoost ², Ph.D. Candidate

¹ Faculty of Mathematics, Statistics, and Computer Science, University of Tabriz, Tabriz, Iran, izadkhah@tabrizu.ac.ir

² Faculty of Mathematics, Statistics, and Computer Science, University of Tabriz, Tabriz, Iran, r.behzadi@tabrizu.ac.ir

Abstract: Nowadays social networks have emerged as a popular platform for communication, entertainment, news and event updates, and more. While social networks offer several benefits such as e-commerce and easy communication, the presence of rumors is a significant drawback. Rumors can spread quickly to many users with minimal effort and lead to unpleasant social phenomena. In previous works, researchers have proposed machine learning and deep learning approaches for automatic rumor detection. This paper proposes a circular multi-layered CNN-BILSTM model for rumor detection in social networks. The model transforms the multi CNN-BILSTMs into a convolutional matrix and then applies a circular state to the matrix. The model only uses the text of social network users and classifies it as rumor or non-rumor. We tested the model on the two real public datasets PHEME and Liar. The model outperforms twelve state-of-the-art models in terms of accuracy.

Keywords: *Natural Language Processing; Deep Learning; Rumor Detection.*

* **Habib Izadkhah**, izadkhah@tabrizu.ac.ir

۱. مقدمه

سخت باشد، زیرا مجموعه کلمات شایعه و غیر شایعه ممکن است بسیار شبیه به هم باشند.

ویژگی‌های مبتنی بر زمینه، اطلاعات مربوط به پست‌ها را در نظر می‌گیرند. این ویژگی‌ها می‌توانند بر اساس کاربر (مانند تعداد دنبال‌کننده‌ها، کشور، دوستان و تعداد کلمات یک پست) و یا شبکه (مانند ساختار انتشار پست‌ها) باشند. به‌طور کلی، ویژگی‌های مبتنی بر زمینه می‌توانند به تشخیص شایعه کمک کنند، اما باید با دقت به کار گرفته شوند [۳].

ویژگی‌های زمینه، اطلاعاتی هستند که مرتبط با یک پست هستند و به‌صورت مستقیم به ساختار شایعه اشاره نمی‌کنند. در حالی که ویژگی‌های محتوا (مانند ویژگی‌های واژگانی) به‌صورت مستقیم به ساختار شایعه اشاره می‌کنند. در روش مبتنی بر واژگان، بسیاری از ویژگی‌های شهودی مانند سبک نوشتن، کلمات نفی، کلمات با حالت برجسته، عبارات و کلمات دشمن وجود دارند. به همین دلیل در این مقاله، قصد داریم از ویژگی‌های واژگانی برای تشخیص شایعه استفاده کنیم. از آنجا که کلمه‌ها مستقیماً برای انتشار یک شایعه استفاده می‌شوند، ویژگی‌های محتوا بهترین راه‌حل برای تشخیص شایعه از غیرشایعه هستند. به‌علاوه، ویژگی‌های واژگانی می‌توانند به شناسایی الگوهای خاص در متن کمک کنند. به‌طور مثال، کلمات دشنام، کلمات نفی و ساختار جملات می‌توانند به شناسایی شایعه کمک کنند [۴].

نوآوری اصلی این مقاله در پیشنهاد چند شبکه عصبی پیچشی^۱ پشته شده^۲ شبکه عصبی حافظه طولانی کوتاه مدت دو جهته^۳، با ساختار حلقوی برای تشخیص شایعه به شکلی کارآمد است. شایعه‌سازان از کلمات و جملاتی با ساختار فریبده استفاده می‌کنند که حتی برای انسان‌ها هم تشخیص دادن آن‌ها دشوار است. به همین دلیل، استفاده از مدلی که قابلیت استخراج ویژگی‌های متمایز زیاد برای تمایز بین شایعات و غیرشایعات را داشته باشد ضروری است. حلقوی یک نوع روش ترکیبی است که با استفاده از عملیات جمع و ضرب، می‌تواند وزن بردارهای بدست آمده از شبکه‌های عصبی پیچشی و شبکه عصبی حافظه کوتاه مدت دو جهته را با هدف بهبود تمایز بین ویژگی‌های دسته‌های مختلف اصلاح کند. مدل ارائه شده از اطلاعات قبلی به‌خاطر سپرده شده استفاده می‌کند و با تکرار عملیات جمع و ضرب، یک بردار

شبکه‌های اجتماعی به عنوان یک منبع اصلی اطلاعات و تبادل نظر در مورد مسائل جاری، نقش مهمی در زندگی روزمره میلیون‌ها نفر از افراد را دارند. در این شبکه‌ها، روزانه حجم عظیمی از داده‌های متنوع و بروز انتشار داده می‌شود که می‌توانند برای تحلیل و پژوهش مورد استفاده قرار گیرند. به همین دلیل، شبکه‌های اجتماعی به‌عنوان یکی از بهترین منابع برای دانشمندان داده و محققان در زمینه پردازش زبان طبیعی شناخته می‌شوند [۱]. علی‌رغم مزایای زیاد شبکه‌های اجتماعی [۲] آنها دارای معایبی نیز هستند. یکی از این معایب، انتشار اخبار جعلی و شایعات نادرست است. تمامی شبکه‌های اجتماعی این مشکل را دارند و شبکه‌ای مستثنی نیست. شایعات اطلاعاتی هستند که بدون تأیید شدن، معمولاً از طریق زبان یا از طریق شبکه‌های اجتماعی منتشر می‌شوند. آنها ممکن است درست یا نادرست باشند، اما آنچه که آن‌ها را متمایز می‌کند، عدم وجود منابع معتبر یا شواهدی است که آن‌ها را پشتیبانی کند. برخلاف شایعات، اخبار جعلی به عمد ساخته و پخش می‌شوند و هدفشان گمراه کردن عموم است. اخبار جعلی اغلب از عناوین گمراه‌کننده‌ای همچون تصاویر دستکاری شده یا داستان‌های ساختگی استفاده می‌کنند تا یک داستان نادرست را ارائه دهند. از آنجایی که شبکه‌های اجتماعی به بخشی از زندگی مردم تبدیل شده‌اند، اخبار جعلی و شایعات می‌توانند نقش بسیار مخربی در زندگی و کسب و کار افراد داشته باشند. به‌عنوان مثال، در اپیدمی ویروس کرونا، شایعات مردم را تشویق می‌کردند که واکسینه نشوند یا به مصرف مواد شوینده شیمیایی برای جلوگیری از آلوده شدن به ویروس کرونا بپردازند. این موارد نشان می‌دهند که شایعات تا چه اندازه می‌توانند خطرناک باشند. برای جلوگیری و به حداقل رساندن چنین آسیب‌هایی، نیاز به مدل‌هایی است که به‌طور خودکار و کارآمد شایعات را شناسایی کنند. در این مقاله، یک مدل یادگیری عمیق جدید ارائه شده است که می‌تواند شایعات را از غیر شایعات به‌صورت کارآمد تشخیص دهد.

در کل، برای تشخیص شایعه می‌توان از ویژگی‌های مبتنی بر محتوا یا مبتنی بر زمینه استفاده کرد. ویژگی‌های محتوایی از مبانی واژگانی، نحوی و معنایی پیروی می‌کنند. برای مثال، ویژگی‌های نحوی شامل اسم‌ها، افعال، صفت‌ها و الگوها هستند و ویژگی‌های واژگانی شامل کلمات یک متن هستند. ویژگی‌های مبتنی بر معنایی هم شامل تجزیه و تحلیل احساسات، نظر کاوی و غیره هستند. با این حال، تشخیص شایعه و غیرشایعه با استفاده از ویژگی‌های مبتنی بر محتوا ممکن است

^۱Convolutional Neural Network (CNN)

^۲Stacked

^۳Bidirectional long short-term memory network (BiLSTM)

نهایی از ویژگی‌ها تولید می‌کند. بردار حلقوی در کنار دیگر بردارهای بدست آمده، به یک لایه کاملاً متصل داده می‌شود. این ساختار حلقوی برای افزایش قدرت تمایز بین دسته‌های مختلف استفاده می‌شود. ساختار حلقوی شرح داده شده، برای اولین بار به ترکیب ساختار شبکه‌های عصبی پیچشی و حافظه کوتاه مدت دو جهته اضافه شده است و تحت دو مجموعه داده کارایی برتر از خود نشان داده شده است.

مدل پیشنهادی شامل چندین لایه‌ی شبکه عصبی پیچشی پشت‌پشته شده-شبکه عصبی حافظه طولانی کوتاه مدت دوجهته^۴ با ساختار حلقوی است که می‌تواند ویژگی‌های متمایز برای دسته‌های متفاوت را به شکلی کارآمد استخراج کند. برای دو مجموعه داده، مدل پیشنهاد شده به بهبود دقت امیدوار کننده‌ای دست یافته است.

فرضیه این مقاله به شرح زیر است:

(۱) آیا مدل چند SCNN-BILSTM با ساختار حلقوی می‌تواند عملکرد تشخیص شایعه را بهبود ببخشد؟

ادامه این مقاله به شرح زیر ساختاردهی شده است: بخش ۲ به مطالعات مرتبط پرداخته است، بخش ۳ روش پیشنهادی را توصیف می‌کند، بخش ۴ نتایج آزمایشی را ارائه می‌دهد و بخش ۵ نتیجه‌گیری را ارائه می‌دهد.

۲. کارهای مرتبط

تحقیقات زیادی برای طبقه‌بندی متن عمدتاً با الگوریتم‌های یادگیری ماشین بردار پشتیبان، K نزدیک‌ترین همسایه، جنگل تصادفی، درخت تصمیم و روش بیز انجام شده است. چندین دهه است که الگوریتم‌های یادگیری ماشین معرفی شده‌اند به گونه‌ای که آنها عملکرد خوبی در اکثر زمینه‌ها مانند طبقه بندی متن نشان داده‌اند. الگوریتم‌های یادگیری ماشین دارای معایبی هستند:

۱. وابستگی به داده‌ها [۵]: عملکرد الگوریتم‌های یادگیری ماشین به کیفیت داده‌ها بستگی دارد، به عبارت دیگر، اگر داده‌ها تمیز نباشند و مشکلاتی مانند ناقص بودن و نویز داشته باشند، پس این امکان وجود دارد که عملکرد مدل یادگیری ماشین تحت تأثیر قرار بگیرد.

۲. جعبه سیاه [۶]: اکثر الگوریتم‌های یادگیری ماشین مانند جعبه سیاه هستند و فرآیند درک آنها دشوار است. از این نظر، کار یک متخصص یادگیری ماشین برای توسعه مدل ساده نخواهد بود. اگرچه این ویژگی جعبه سیاه در مدل‌های یادگیری عمیق نیز وجود دارد و به مراتب درک آن دشوارتر است.

۳. پیچیدگی بالا [۷]: بسیاری از الگوریتم‌های یادگیری ماشین از نظر زمان و حافظه پیچیدگی بالایی دارند. این ویژگی در مدل‌های مبتنی بر یادگیری عمیق بسیار بیشتر است.

۴. بیش‌برازش [۸]: در الگوریتم‌های یادگیری ماشین، به دلیل استخراج ویژگی‌ها توسط متخصصان و انتخاب دستی فرآیندها بیش‌برازش ممکن است رخ دهد.

اخیراً مدل‌هایی که محققان برای یادگیری ماشین ارائه می‌دهند، حالت خاصی از این الگوریتم‌ها هستند. از آنجایی که الگوریتم K نزدیک‌ترین همسایه، دارای پیچیدگی زمانی بالایی است، تان و یوکان [۹] K -RS نزدیک‌ترین همسایه را ارائه کردند که نسبت به الگوریتم K نزدیک‌ترین همسایه معمول پیچیدگی زمانی کمتری داشت. آنها در نتایج‌شان نشان دادند که K -RS نزدیک‌ترین همسایه از K نزدیک‌ترین همسایه اولیه بهتر عمل می‌کند. گودجیل و همکاران [۱۰] یک مدل یادگیر فعال جدید ارائه کردند که هدف اصلی آن برای کاهش زمان زیاد برچسب‌گذاری بود به گونه‌ای که به جای انتخاب دستی، برچسب‌ها به صورت هوشمند انتخاب می‌شدند و با توجه به این نکته که دقت طبقه‌بندی کاهش پیدا نکند. برای در نظر گرفتن رابطه بین کلمات، نویسندگان در [۱۱] جنگل تصادفی معنایی را ارائه کردند و نشان دادند که جنگل تصادفی معنایی از الگوریتم اولیه جنگل تصادفی بهتر عمل کرده است. در کار دیگری برای الگوریتم‌های یادگیری ماشین، وانگ و همکاران، نی‌تری^۵ [۱۲] را ارائه دادند. نی‌تری یک الگوریتم ترکیبی بود که از یک طبقه‌بندی کننده ساده بیز در هر گره برگ درخت تصمیم ساخته شده استفاده می‌کرد. با افزایش داده‌ها و پیچیدگی مسائل، الگوریتم‌های یادگیری ماشین عملکرد ضعیفی می‌توانند داشته باشند، و به دلیل انتخاب دستی ویژگی‌ها و فرآیندها، معمولاً دچار بیش‌برازش می‌شوند. به منظور افزایش بهره‌وری الگوریتم‌های یادگیری ماشین، معمولاً روابط متفاوتی برای مجموعه داده‌های مختلف در نظر گرفته می‌شوند، که اینکه کدام

^۵NBTREE

^۴SCNN-BILSTM

بهترین رابطه انتخاب شود خود یک چالش است. به عنوان مثال، در الگوریتم ماشین بردار پشتیبان، هسته‌های زیادی وجود دارد و ممکن است انتخاب هر کدام یک بر کارایی مدل تأثیر مستقیم بگذارد. انتخاب یک هسته مناسب امری دشوار و معمولاً نیاز به آزمایش‌های تجربی زیادی را به همراه دارد. برای این منظور، یک رهیافت مناسب برای دوری کردن از انتخاب رابطه‌های متعدد برای یک مدل یادگیر، استفاده از یادگیری عمیق است. اگرچه آنها نیازمند داده‌های زیادی برای آموزش هستند و پیچیدگی زمانی و حافظه‌ای بالایی دارند. در مقاله [۱۳]، پیشنهاد شده است که از شبکه‌های عصبی پیچشی و شبکه عصبی حافظه طولانی کوتاه مدت دوجته با لایه توجه برای طبقه‌بندی متون فارسی استفاده شود. نویسندگان در [۱۴]، با استفاده از معیارهای مسدود کردن متن و شباهت کسینوسی، کپی‌برداری در متون علمی را شناسایی کردند. در یک کار مقایسه‌ای و کاربردی [۱۵]، روش‌های یادگیری ماشین و یادگیری عمیق در تحلیل احساسات پیام‌های تلگرام به زبان فارسی مورد ارزیابی قرار گرفته است. در یک مقاله اخیر [۱۶]، یک مدل یادگیری عمیق مبتنی بر محاسبات دانه‌ای برای طبقه‌بندی متن ارائه شده است.

تشخیص شایعه یک حوزه فعال تحقیقاتی است. ما و همکاران [۱۷] از مدل شبکه‌های عصبی تکراری برای شناسایی شایعات استفاده کردند. آنها در کار خود نشان داده‌اند که روش آنها به ترتیب در مجموعه داده‌های توییت و ویبو بهتر عمل می‌کند. کار دیگر در زمینه تشخیص شایعه مربوط به کار کومار و همکاران بود [۱۸]. آنها در ارزیابی‌های تجربی خود نشان دادند که بهترین دقت بر روی مجموعه داده PHEME، ۷۸٫۴۰٪ بود. روچانسکی و همکاران [۱۹] یک مدل عمیق ترکیبی^۱ برای تشخیص اخبار جعلی ارائه کردند که در آن مدل، آنها از ویژگی‌های ترکیبی استفاده می‌کردند. مدل بعدی برای تشخیص شایعه، طبقه‌بندی‌کننده شبکه عصبی حافظه طولانی کوتاه مدت^۲ بود [۲۰]. آنها قبل از طبقه‌بندی، نرمال‌سازی هشتگ و خوشه‌بندی هشتگ را انجام دادند. برخلاف کارهای قبلی، نوین [۲۱] در مقاله‌ای روی زمان تشخیص یک شایعه تمرکز داشت. خو و همکاران در سال ۲۰۲۰ [۲۲]، از یک مدل تشخیص شایعه مبتنی بر موضوع استفاده کردند که در آن از موضوع توییت‌های داده شده برای تشخیص شایعه استفاده شده است. آنها دقت ۸۲٫۶۶٪ را برای مجموعه داده PHEME و

^۱Hybrid

^۲Long Short Term Memory (LSTM)

^۳Liar

دقت ۶۲٫۲۴٪ را برای مجموعه داده لیار^۳ در مقایسه با ۱۱ روش مطالعه شده تحت اعتبارسنجی متقاطع ۵ مرحله‌ای به دست آوردند. در کار دیگری، یوی و همکاران [۲۳] با استفاده از شبکه عصبی نمودار انتشار با مکانیسم توجه، یک طبقه‌بندی چهار کلاس برای تشخیص شایعه در مجموعه داده‌های PHEME را انجام دادند. ژانگ و همکاران [۲۴] از یک شبکه جمع‌آوری مسیر انتشار سبک برای تشخیص شایعه استفاده کردند. برای مجموعه داده PHEME، آنها به دقت ۸۰٪ دست یافتند. نویسندگان در [۲۵]، بر خلاف دیگر کارها یک شبکه عصبی چند-مدل برای تشخیص اولیه‌ی شایعه را ارائه دادند که هم ویژگی‌های متنی و هم ویژگی‌های تصویری را در شبکه‌های اجتماعی در نظر می‌گرفت. آنها نشان دادند که مدل آنها از لحاظ معیار بازیابی بر روی مجموعه داده‌های توییت و ویبو کارآمد است. به خاطر در نظر گرفتن چندین مدل یادگیری عمیق و استخراج‌کننده‌های ویژگی مجزا برای متن و تصویر، پیچیدگی این مدل بالا است. در مقاله [۲۶]، تلاش شده است تا درک جامعی از ساختار شایعه به دست بیاید که از این نظر آنها یک شبکه عصبی خود توجه بر اساس ساختار بازتوییت‌ها تشکیل دادند که هم از اطلاعات کاربران و هم از داده‌های متنی کاربران، در این شبکه استفاده می‌کرد. در مطالعه [۲۷]، نویسندگان یک مدل شبکه عصبی توجه گراف چند کانالی با ماژول اشتراک‌گذاری رویداد برای تشخیص شایعه ارائه دادند که همانند مطالعه [۲۵]، پیچیدگی این مدل بالا بود. در تحقیق [۲۸]، چن و همکاران مدلی برای تشخیص شایعه با استفاده از چندین زمینه اطلاعاتی مرتبط با کاربران شبکه‌های اجتماعی برای یادگیری بهینه بازنمایی مختلف کاربرانی که در گسترش پست شرکت کردند، ارائه دادند. آنها برای ارزیابی مدل ارائه شده، مجموعه داده‌های ویبو، توییت ۱۵ و ۱۶ را در نظر گرفتند. مطالعات در زمینه انتشار اطلاعات نشان داده‌اند که دایره اجتماعی یک نقش قابل توجه در سرعت، محدوده و محتوای شایعه پخش شده ایفا می‌کند. با توجه به این نکته مهم، الگوریتم تشخیص شایعه جدیدی در [۲۹] پیشنهاد شده است. این الگوریتم با استخراج دایره‌های اجتماعی با همگنی بالا از محیط کاربر و ترکیب آن با تعامل اجتماعی به طور خودکار شایعه‌ها را شناسایی می‌کند. نتایج آزمایشی بر روی سه مجموعه داده واقعی (توییت ۱۵، توییت ۱۶ و PHEME) انجام شده است. این آزمایش‌ها نشان می‌دهند که رویکرد پیشنهادی بهتر از روش‌های پیشین عمل کرده و کارآمدی برتری در تشخیص شایعه را نشان می‌دهد. شایان ذکر است که آنها فقط بخشی از داده‌های

اکثر مدل‌های یادگیری ماشین و یادگیری عمیق برای تفسیر کلمات باید آنها را به یک نمایش عددی تبدیل کنند [۳۳]. تعبیه کلمه نمایشی از کلمات در فضاهای برداری پیوسته است که در آن رابطه معنایی بین کلمات در نظر گرفته می‌شود و کلماتی که دارای معنای مشابه هستند بازنمایی مشابهی دارند. رمزگذاری یک داغ^{۱۱} [۳۴] یکی از ساده‌ترین نمایش‌های واژگان است که از ابعاد بالا، ماتریس پراکنده بزرگ و عدم تعبیه اطلاعات معنایی رنج می‌برد. رویکردهای تعبیه کلمه محدودیت‌های رمزگذاری یک داغ را برطرف می‌کنند. برای نگاشت هر کلمه به یک فضای برداری، تعبیه‌های از پیش آموزش دیده و ثابتی وجود دارد. تعبیه‌های از پیش آموزش دیده شده معروف واژه به بردار^{۱۲} [۳۵] و بردارهای جهانی برای نمایش کلمه^{۱۳} [۳۶] هستند که می‌توانند اطلاعات معنایی را در بین کلمات در نظر بگیرند. به عبارتی هرچه کلمات شباهت بیشتری داشته باشند، نمایش مشابه بیشتری خواهند داشت.

۲-۳. شبکه‌های عصبی پیچشی

شبکه‌های عصبی پیچشی یک شبکه بازخورد روبه‌جلو^{۱۴} هستند که برای استخراج الگوهای محلی منحصر به فرد ممکن در توالی‌های بزرگ در اندازه‌های مختلف استفاده می‌شوند. آنها از یک تابع آموخته شده غیرخطی با پنجره‌ای به اندازه k استفاده می‌کنند. در هر مرحله زمانی، برای رسیدن به کل توالی، پنجره لغزنده می‌شود. در نتیجه این کار، مقداری برگردانده می‌شود که نشان دهنده داده‌های آن پنجره زمانی است. با استفاده از یک فیلتر برای هر پنجره، یک بردار n بعدی به دست می‌آید. در ادامه، با استفاده از عملیات ادغام، بردارهای به دست آمده در یک بردار واحد ادغام می‌شوند به نحوی که این بردار نمایانگر نمایی از کل توالی باشد. متأسفانه، شبکه‌های عصبی پیچشی نمی‌توانند یک الگوی سراسری از توالی‌های به دست آمده را داشته باشند [۳۷].

۳-۳. حافظه‌ی کوتاه‌مدت طولانی

از آنجایی که شبکه‌های عصبی پیچشی، شبکه عصبی بازخورد رو به جلو هستند، نمی‌توانند وابستگی بین توکن‌های یک دنباله را در نظر بگیرند. شبکه‌های عصبی بازگشتی^{۱۵} برای حل این مشکل پیشنهاد

PHEME را استفاده کردند. در مقاله [۳۰]، یک مدل شبکه عصبی ارائه داده شده است که از دانش خارجی موجود در گراف دانش برای تشخیص شایعه استفاده می‌کند. با توجه به معنای کم و مبهم کلمات درون هر پست، از شناسایی موجودیت‌ها در محتوای پست استفاده می‌شود و آنها با موجودیت‌ها و مفاهیم مرتبط در گراف دانش ارتباط داده می‌شوند تا به عنوان اطلاعات مکمل معنایی برای هر متن پست در نظر گرفته شوند. این مدل به منظور بهینه‌سازی استفاده از دانش خارجی در نمایش متنی، یک مکانیزم توجه را توسعه می‌دهد تا دانش محلی را به متن پست ادغام کند. آزمایش‌های بر روی چهار مجموعه داده واقعی (توییت^{۱۵}، توییت^{۱۶} و واقعیت سیاسی و PHEME) کارایی این مدل را نشان می‌دهد. آنها فقط بخشی از داده‌های PHEME را استفاده کردند. در مقاله [۳۱] یک مدل تشخیص محتوای جعلی بر اساس شبکه‌های عصبی گراف ترکیب شده^{۱۷} برای شبکه‌های اجتماعی پیشنهاد شده است. در این مدل، شبکه عصبی گراف به عنوان ساختار پایه و همچنین شبکه‌های عصبی کانولوشن و شبکه‌های عصبی بازگشتی جهت حفظ دو جنبه از معنا استفاده شده است. این چارچوب تشخیص ترکیب شده، بر روی مجموعه داده‌های ویبو و لیار ارزیابی شده است. آنها فقط بخشی از داده‌های لیار را استفاده کردند. شایعات برخط با چند حالت به طور همزمان احتمالاً می‌توانند کاربران را گمراه کنند. بنابراین تشخیص شایعات چندحالتی نباید نادیده گرفته شوند. در [۳۲]، یک شبکه عصبی ادغام چندحالتی برای تشخیص شایعات پیشنهاد شده است که هم بر روی تصاویر و هم متن توجه می‌کند. آزمایش‌های مقایسه‌ای روی دو مجموعه ویبو و توییت انجام شده است.

۳. مدل پیشنهادی

در این بخش اجزای مدل پیشنهادی توصیف می‌شوند. مدل پیشنهادی را به اختصار CMSCB^{۱۸} می‌نامیم.

برای تشخیص شایعات، از مدل‌های یادگیری عمیق استفاده می‌کنیم. تشخیص شایعه در این مطالعه، یک کار طبقه‌بندی متن است. به‌طور کلی، در طبقه‌بندی متن با داده‌های با ابعاد بالا مواجه هستیم و توسعه و عملکرد یک مدل بهتر می‌شود، اگر از یادگیری عمیق استفاده کنیم زیرا ویژگی‌ها از داده‌های با ابعاد بالا به‌سادگی و بهتر استخراج می‌شوند.

۱-۳. تعبیه کلمه

^{۱۱}One-hot

^{۱۲}Word2vec

^{۱۳}Glove

^{۱۴}feed-forward

^{۱۵}RNN

^{۱۶}GNN

^{۱۷}Circular Multi Stacked CNN-BILSTM

منظور، عملیات متفاوتی همچون حالت‌های مختلف از شبکه‌های عصبی پیچشی پشته شده و شبکه عصبی حافظه طولانی کوتاه مدت دو جهته و ساختار حلقوی در نظر گرفته می‌شوند. به صورت رسمی، مجموعه داده D بعنوان ورودی به چندین شبکه عصبی پیچشی پشته شده $CB_{K,T}$ ، $K = 1, 2, \dots, P$ ، $T = 1, 2, \dots, Q$ داده می‌شود که P تعداد شبکه‌های عصبی پیچشی شامل شده در پشته و Q تعداد کل شبکه‌های عصبی پشته شده است. حاصل ادغام $CB_{K,T}$ به یک شبکه عصبی حافظه طولانی کوتاه مدت دو جهته B_Q متصل می‌شود. سپس تمامی B ها با یکدیگر دو به دو جمع می‌شوند و به صورت حلقوی در یکدیگر ضرب می‌شوند که تعداد حالت دوبه‌دوی آن‌ها با استفاده از ترکیب بدست می‌آید که در رابطه ۱ نشان داده شده است. حالت‌های جمع و حلقوی در رابطه ۲ و ۳ نشان داده شده‌اند، به‌طوری- که a_0 تا a_{y-1} و c_0 تا c_{y-1} بردارهایی به طول m هستند که هر کدام یک از آنها حاصل ترکیب شبکه عصبی پیچشی پشته شده با شبکه عصبی حافظه طولانی کوتاه مدت دو جهته است.

$$f(Q, r) = \frac{Q!}{(Q-r)!r!} = y \quad (1)$$

$$a_0 = B_1 + B_2, a_0 \in \mathbb{R}^m$$

$$a_1 = B_2 + B_3, a_1 \in \mathbb{R}^m$$

$$\dots$$

$$a_{y-1} = B_Q + B_{Q-1}, a_{y-1} \in \mathbb{R}^m$$

$$c_0 = a_0 \times a_{y-1}, c_0 \in \mathbb{R}^m$$

$$c_1 = c_0 \times a_1, c_1 \in \mathbb{R}^m$$

$$\dots$$

$$c_{y-1} = c_Q \times a_{y-1}, c_{y-1} \in \mathbb{R}^m \quad (3)$$

بعد از آن، در مرحله‌ی چهارم میانگین از عبارات‌های رابطه سوم گرفته می‌شود:

شده‌اند. در شبکه‌های عصبی بازگشتی، توسط برخی از گیت‌های داخلی، حالت‌های قبلی به‌خاطر سپرده می‌شوند. با استفاده از یک شبکه عصبی بازگشتی، سیگنال‌های خطای انتشار به عقب ممکن است ناپدید شوند. به اصطلاح محوشدگی گرادیان^{۱۶} رخ دهد. شبکه‌های عصبی حافظه طولانی کوتاه مدت راه‌حلی برای رفع محوشدگی گرادیان ارائه می‌دهند. آن‌ها می‌توانند ثابت جریان خطا را توسط گیت‌های ورودی و فراموشی ذخیره کنند. این گیت‌ها تعیین می‌کنند که چه مقدار داده می‌تواند ذخیره شود. شبکه عصبی حافظه طولانی کوتاه مدت دو جهته، نوعی شبکه عصبی است که برخلاف شبکه عصبی حافظه طولانی کوتاه مدت که فقط یک جهت را برای آموزش در نظر می‌گیرند، نه تنها جهت رو به جلو بلکه جهت عقب را نیز در نظر می‌گیرد [۳۸].

۳-۴. مدل پیشنهاد شده

برای تمایز بین ساختار شایعات و غیر شایعات، ویژگی‌های محلی و وابستگی بین آنها در نظر گرفته می‌شود. برای استخراج الگوهای خاص محلی، از شبکه‌های عصبی پیچشی با هسته‌های مختلف استفاده می‌شود و برای استخراج ویژگی‌های متمایز، وابستگی ویژگی‌های محلی به‌دست آمده توسط شبکه‌های عصبی پیچشی، توسط شبکه عصبی حافظه طولانی کوتاه مدت دو جهته گرفته می‌شود. از آنجایی که مسائل طبقه‌بندی متن ماهیتاً ابعاد بالایی دارند و برای آنکه ویژگی‌های مناسب و خاص را تا حد قابل ممکن پیدا کنیم، در این مطالعه، ما چند CNN-BILSTM را در نظر گرفته‌ایم. این روش کمک می‌کند تا کارایی افزایش یابد. در واقع، این چند ماتریس CNN-BILSTM به‌صورت دوبه‌دو با یکدیگر جمع می‌شوند و سپس دوبه‌دو آنها به‌صورت حلقوی با یکدیگر ضرب می‌شوند. بعد از میانگین‌گیری و نرمال‌سازی، بر روی ماتریس‌ها در مرحله قبل، مدل ارائه شده یادگیری ساختار شایعات و غیر شایعات بر روی تک ماتریس به‌دست آمده را انجام می‌دهد.

در یادگیری عمیق، ویژگی‌ها برای یک مسئله داده شده به‌صورت خودکار استخراج می‌شوند و همچون یادگیری ماشین نیازی به فراهم کردن فرضیه خاصی برای استخراج ویژگی‌ها نیست. در مدل پیشنهادی، هدف افزایش خودکار ویژگی‌های متفاوت از طریق افزودن بردارهای قبلی یادگرفته شده و اصلاح وزن‌های بردارها با عملیات مختلف جهت متمایزسازی بین دسته‌های مختلف است. به همین

$$\eta = \frac{1}{y} \times \sum_{i=1}^y \sum_{j=1}^m c_{ij}, \eta \in \mathbb{R}^m \quad (4)$$

شبکه عصبی پیچشی پشته شده، پولینگ گرفته می‌شود. در این مطالعه، ما چهار شبکه عصبی پیچشی پشته شده را در نظر گرفته‌ایم که در واقع، شبکه عصبی پیچشی یک تا چهار هر کدام چهار هسته متفاوت دارند که در هربار با آن هسته به‌روزرسانی می‌شوند. در ادامه، شبکه عصبی پیچشی پشته شده با یکدیگر ادغام می‌شوند و برای کاهش فرآپارامترها از آن یک پولینگ گرفته می‌شود. در مرحله بعد، خروجی هر شبکه عصبی پشته شده به یک شبکه عصبی حافظه طولانی کوتاه مدت دو جهته وصل می‌شود، جایی که شبکه‌های عصبی حافظه طولانی کوتاه مدت دو جهته در نظر گرفته شده، دارای فرآپارامترهای یکسان هستند. در مرحله ۴، اعمال جمع ماتریس برای هر جفت خروجی ماتریس به‌دست آمده از حاصل اتصال هر شبکه عصبی پیچشی پشته شده و شبکه عصبی حافظه طولانی کوتاه مدت دو جهته انجام می‌شود. اگر X_1 تا X_4 خروجی‌های بدست آمده از حاصل اتصال هر شبکه عصبی پیچشی پشته شده و شبکه عصبی حافظه طولانی کوتاه مدت دو جهته باشند، آنها بر اساس رابطه ۲ دو به دو با یکدیگر جمع می‌شوند و بر اساس رابطه ۳ به صورت حلقوی در $CO(norm, B, \eta) = Concat(\eta, norm, B), C_1 \in \mathbb{R}^{3 \times m}$ یکدیگر ضرب می‌شوند.

$$\begin{aligned} a_0 &= x_1 + x_2 \\ a_1 &= x_1 + x_3 \\ a_2 &= x_1 + x_4 \\ a_3 &= x_2 + x_3 \\ a_4 &= x_2 + x_4 \\ a_5 &= x_3 + x_4 \end{aligned}$$

$$\begin{aligned} c_0 &= a_0 \times a_5 \\ c_1 &= c_0 \times a_1 \\ c_2 &= c_1 \times a_2 \\ c_3 &= c_2 \times a_3 \\ c_4 &= c_3 \times a_4 \\ c_5 &= c_4 \times a_5 \end{aligned}$$

بر اساس رابطه ۴، میانگین C_0 تا C_5 گرفته می‌شود و در ماتریس η ذخیره می‌شود و سپس نرمال می‌شود، به نحوی که این نرمال‌سازی داده‌ها را در مرکز تقریباً صفر با انحراف استاندارد ۱ توزیع می‌کند. سپس، برای یادگیری اطلاعات بخاطر سپرده شده در مرحله آموزش، بردارهای در مرحله قبل با یکدیگر ادغام می‌شوند. در مرحله پایانی، بردارهای ادغام شده به یک لایه کاملاً متصل با ۱۲۸ گره و یک تابع فعال سازی

در مرحله بعد، بردار بدست آمده از رابطه ۴ با رابطه ۵ نرمال‌سازی می‌شود:

$$norm(\eta, \mu, v) = \frac{\eta - \mu}{\sqrt{v}}, norm \in \mathbb{R}^m \quad (5)$$

η برداری است که از میانگین تمامی بردارها در ساختار حلقوی بدست آمده است، μ میانگین ریاضی و v واریانس η هستند.

در نهایت، برای در نظر گرفتن ویژگی‌های بدست آمده از بردارهای مختلف، بردارهای بدست آمده از رابطه ۴، رابطه ۵ و B_1 (اولین شبکه عصبی پیچشی پشته شده وصل شده به شبکه عصبی حافظه طولانی کوتاه مدت دو جهته) پیوند داده می‌شوند:

ممکن است در اثر انتقال داده‌ها در بخش‌های مختلف (عملیات مختلف صورت گرفته توسط ماتریس‌ها)، تمایز قوی بین ساختار شایعات و غیر شایعات ایجاد نشود. در این مقاله، برای ایجاد تمایز قوی بین ساختار شایعات و غیر شایعات، از حالت‌های قبلی به‌خاطر سپرده شده استفاده می‌شود، یعنی η و B_1 . در نهایت، CO ، به یک لایه کاملاً متصل، اتصال داده می‌شود.

شکل ۱ مدل پیشنهاد شده در این مقاله را نشان می‌دهد. در مرحله ۱ از بلوک لایه ورودی، پست‌های شبکه اجتماعی (فقط متن) استخراج می‌شوند و این داده‌های متنی توسط تعبیه کلمه از پیش آموزش دیده شده "بردارهای جهانی برای نمایش کلمه" به فضاها برداری پیوسته عددی تبدیل می‌شوند. در مرحله ۳، چند CNN-BILSTM ایجاد می‌شود که در آن چهار شبکه عصبی پیچشی پشته شده با فرآپارامترهای مختلف و چهار شبکه عصبی حافظه طولانی کوتاه مدت دو جهته با فرآپارامترهای یکسان در نظر گرفته می‌شوند.

برای هر شبکه عصبی پیچشی پشته شده، شبکه‌های عصبی پیچشی متفاوتی در نظر گرفته می‌شود. به نحوی که هر کدام اندازه هسته، پولینگ^{۱۷} و فیلتر متفاوتی دارند. برای کاهش تعداد پارامترها در هر

^{۱۷}Pooling

مختلف مانند اندازه هسته، تعداد فیلترها و اندازه پولینگ در خط ۱ تا ۱۱ از شبکه کد ۱ ایجاد می‌شود. در این تابع، در هر تکرار از حلقه، یک لایه پیچش یک‌بعدی با هسته‌ها و فیلترهای مختلف ایجاد شده (شبکه عصبی پیچشی پشته شده) و به لیست خروجی افزوده می‌شود. سپس لایه‌های به دست آمده از لیست خروجی، ادغام شده و به یک لایه پیچش یک‌بعدی دیگر ارسال می‌شوند. در نهایت، یک لایه پولینگ از لایه پیچش یک‌بعدی گرفته می‌شود و به یک لایه شبکه عصبی حافظه طولانی کوتاه مدت ایجاد شده ارسال می‌شود و این لایه‌ی آخر به عنوان نتیجه تابع بازگردانده می‌شود. در ادامه شبه کد ۱، تنظیمات اولیه مانند تعداد گره‌ها، لایه ورودی و ماتریس تعبیه کلمات (تعبیه کلمه از پیش آموزش دیده "بردارهای جهانی برای نمایش کلمه" ۳۰۰ بعدی) تعریف می‌شوند. سپس لایه‌های f1 تا f4 نیز با استفاده از تابع ایجاد شده در خطوط ۱ تا ۱۱، در خطوط ۱۵ تا ۲۲ ایجاد می‌شوند. در خطوط ۲۳ و ۲۴ ساختار حلقوی به شبکه عصبی ایجاد شده، اعمال می‌شود. در عمل، در خط ۲۳ بصورت دو به دو لایه‌های f1 تا f4 با یکدیگر جمع می‌شوند و سپس حاصل جمع آنها بصورت حلقوی در خط ۲۴ بصورت دو به دو با یکدیگر ضرب می‌شوند. در خطوط بعدی، لایه‌های میانگین‌گیری، نرمال‌سازی، ادغام و متراکم ایجاد شده و به مدل اضافه می‌شوند. مدل با استفاده از الگوریتم بهینه‌ساز Adam و تابع هزینه آنتروپی متقاطع طبقه‌ای پراکنده کامپایل می‌شود.

شبه کد ۱: شبکه عصبی پشته‌ای حلقوی

Function Circular_Multi_Stacked_CNNBILSTM(MAX_SEQ_LEN, EMBEDDING_DIM)

1. **function** Convolution_and_BILSTM_layers(embedding, kernel_sizes, filters, pool_size, lstm_nodes)
2. convs = []
3. for kernel_size in kernel_sizes
4. l_conv = create_conv1d_layer(filters=filters, kernel_size=kernel_size, padding='same', activation='relu')(embedding)
5. l_pool = create_maxpool1d_layer(kernel_size)(l_conv)
6. convs.append(l_pool)
7. l_merge = concatenate(convs, axis=1)
8. l_cov = create_conv1d_layer(filters=filters, kernel_size=4, activation='relu')(l_merge)
9. l_pool = create_maxpool1d_layer(pool_size)(l_cov)
10. lstm_layer = create_bidirectional_lstm_layer(lstm_nodes)(l_pool)
11. return lstm_layer
12. Nodes=64
13. input = create_input_layer(MAX_SEQ_LEN)
14. embedding_matrix = create_matrix(EMBEDDING_DIM)(input)

رلیو^{۱۸} داده می‌شود و بعد از آن این ۱۲۸ گره به یک لایه کاملاً متصل دیگر با ۲ گره و با تابع فعال‌ساز بیشینه هموار^{۱۹} متصل می‌شود. فرآیند آنها برای شبکه‌های عصبی پیچشی پشته شده در جدول ۱ نشان داده شده است، به طوری که برای هر شبکه‌های عصبی پیچشی، یک فیلتر ثابت (اندازه پنجره) و هسته و پولینگ وجود دارد. در جدول ۱، پولینگ ۱ برای شبکه‌های عصبی پیچشی پشته شده استفاده می‌شود در حالی که پولینگ ۲ برای شبکه‌های عصبی پیچشی ادغام شده استفاده می‌شود. حافظه طولانی کوتاه مدت دو جهت‌دار ۱۲۸ گره و مقدار رهاشدگی^{۲۰} ۰.۲۰ است. حداکثر تعداد کلمات و حداکثر طول دنباله‌ها به ترتیب ۱۰۰۰۰ و ۵۰۰ است. در مسائل پیچیده، برای تمایز بین نمونه‌های مثبت و منفی (یعنی غیرشایعه و شایعه) توابع خطی به خوبی کار نمی‌کنند. به جز آخرین لایه پنهان که از تابع فعال‌ساز تابع بیشینه هموار استفاده می‌کند، سایر فعال‌سازها از تابع رلیو استفاده می‌کنند. برای مدل ارائه شده، از تعبیه کلمه از پیش آموزش دیده "بردارهای جهانی برای نمایش کلمه" ۳۰۰ بعدی استفاده شده است. برای بهینه‌سازی، از بهینه‌ساز معروف آدام^{۲۱} استفاده شده است. برای پیاده‌سازی مدل‌های یادگیری ماشین و یادگیری عمیق، به ترتیب از سایکیت‌لرن^{۲۲} [۳۹] و کراس^{۲۳} [۴۰] استفاده کرده‌ایم. پیاده‌سازی‌ها در زبان برنامه‌نویسی پایتون و محیط گوگل کولب^{۲۴} انجام شده است.

برای درک بهتر از مدل پیشنهادی، شبه کد شبکه عصبی حلقوی در ادامه آورده شده است. شبه کد ۱، پیاده‌سازی یک معماری شبکه عصبی پشته‌ای حلقوی را برای طبقه‌بندی متن نشان می‌دهد. این معماری از ترکیب لایه‌های مختلف از جمله لایه‌های شبکه عصبی پیچشی و شبکه عصبی حافظه طولانی کوتاه مدت تشکیل شده است. در ادامه توضیحات هر بخش را بیان خواهیم کرد.

در ابتدا، تابعی برای ایجاد لایه‌های شبکه عصبی پیچشی متصل به شبکه عصبی حافظه طولانی کوتاه مدت با ویژگی‌های

^{۱۸}Relu

^{۱۹}Softmax

^{۲۰}Dropout

^{۲۱}ADAM

^{۲۲}Skit-Learn

^{۲۳}Keras

^{۲۴}Google Colab

داده شش دسته "نادرست"، "به سختی درست"، "شلوار آتش"، "نیمه درست"، "بیشتر درست" و "درست" را دارد. از آنجا که کار ما در این مقال طبقه‌بندی دو کلاسه است، ما به "سختی درست" "نیمه درست"، "بیشتر درست" و "درست" را بعنوان دسته یک و "شلوار آتش" و "نادرست" را به عنوان دسته صفر در نظر می‌گیریم. این مجموعه داده حاوی ۱۲۸۳۶ نمونه است.

۴-۲. معیارهای ارزیابی

برای ارزیابی مدل‌ها از معیارهای استاندارد در معادله‌های ۶ تا ۱۱ استفاده شده است.

$$\text{صحت (شایعه)} = \frac{TP}{TP + FP} \quad (6)$$

$$\text{پوشش (شایعه)} = \frac{TP}{TP + FN} \quad (7)$$

$$\text{صحت (غیرشایعه)} = \frac{TN}{TN + FN} \quad (8)$$

$$\text{بازیابی (غیرشایعه)} = \frac{TN}{TP + FN} \quad (9)$$

$$\text{دقت} = 2 \times \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

TP نشان دهنده مثبت واقعی، TN نشان دهنده منفی واقعی، FP نشان دهنده مثبت کاذب و FN نشان دهنده منفی کاذب است. FN به تعداد نمونه‌هایی اشاره می‌کند که به‌طور صحیح و اشتباه به عنوان غیر شایعه برچسب‌گذاری شده‌اند. همچنین TN و FP به تعداد نمونه‌هایی اشاره می‌کنند که به‌طور صحیح و اشتباه بعنوان شایعه برچسب‌گذاری شده‌اند. برای کلاس شایعه / غیر شایعه، بازیابی^{۲۵} به درصدی از کل پیش بینی‌ها اشاره دارد که به‌درستی طبقه‌بندی شده‌اند، در حالی که

```

15. k1=[9,7,5,3]
16. f1 = Convolution_and_BILSTM_layers(embedding, k1, 128, 96, Nodes)
17. k2=[8,6,4,2]
18. f2 = Convolution_and_BILSTM_layers(embedding, k2, 256, 128, Nodes)
19. k3=[8,6,4,2]
20. f3 = Convolution_and_BILSTM_layers(embedding, k3, 64, 20, Nodes)
21. k4=[9,7,5,3]
22. f4 = Convolution_and_BILSTM_layers(embedding1, k4, 32, 10, Nodes)
23. a0, a1, a2, a3, a4, a5 = create_add_layer([(f1, f2), (f1, f3), (f1, f4), (f2, f3), (f2, f4), (f3, f4)])
24. c0, c1, c2, c3, c4, c5 = create_multiply_layers([(a5, a0), (c0, a1), (c1, a2), (c2, a3), (c3, a4), (c4, a5)])
25. merge0 = create_average_layer()([c0, c1, c2, c3, c4, c5])
26. merge1 = create_normalization_layer()(merge0)
27. mergedOut = concatenate([merge1, bi1, merge0])
28. mergedOut = create_flatten_layer()(merge1)
29. merge1 = create_dense_layer(128, activation='relu')(mergedOut)
30. merge1 = create_dense_layer(2, activation='softmax')(merge1)
31. model = create_model([input], merge1)
32. model.compile(loss='sparse_categorical_crossentropy', optimizer='adam')
return model

```

جدول ۱: مقادیر فرآپارامترها برای CNNهای پشته شده

موارد	شبکه عصبی پشته‌ای ۱	شبکه عصبی پشته‌ای ۲	شبکه عصبی پشته‌ای ۳	شبکه عصبی پشته‌ای ۴
فیلتر	۱۲۸	۲۵۶	۶۴	۳۲
هسته	۳, ۵, ۷, ۹	۲, ۴, ۶, ۸	۲, ۴, ۶, ۸	۳, ۵, ۷, ۹
پولینگ ۱	۳, ۵, ۷, ۹	۲, ۴, ۶, ۸	۲, ۴, ۶, ۸	۳, ۵, ۷, ۹
پولینگ ۲	۹۶	۱۲۸	۲۰	۱۰

۴. نتایج تجربی

در این بخش، مجموعه داده‌های استفاده شده، و نتایج تجربی به‌دست آمده از مدل پیشنهادی نشان داده می‌شوند.

۴-۱. مجموعه داده

برای تمامی ارزیابی‌ها، مجموعه داده شناخته شده PHEME و لیار را در نظر گرفته‌ایم. این مجموعه داده دارای ۵۸۰۲ توییت است که ۳۸۳۰ و ۱۹۷۲ تا نمونه از این مجموعه داده به ترتیب غیر شایعه (دسته صفر در طبقه‌بندی) و شایعه (دسته یک در طبقه‌بندی) هستند. مجموعه داده دوم استفاده شده، داده لیار [۴۱] است. این مجموعه

^{۲۵}Recall

صحت^{۲۶}، تعداد کل پیش‌بینی‌های انجام‌شده برای کلاس مورد نظر است. $F1$ میانگین وزنی هارمونیک صحت و بازیابی است. دقت^{۲۷} نسبت تعداد کل پیش‌بینی‌های صحیح است. برای انجام تمام آزمایش‌ها، ما از اعتبارسنجی متقاطع به‌ازای $k=5$ استفاده کرده‌ایم، به‌نحوی که ۸۰ درصد داده‌ها برای اعتبارسنجی متقاطع و ۲۰ درصد از آنها برای آزمایش استفاده می‌شوند.

۳-۴. نتایج

ما مدل پیشنهادی را با مدل‌های یادگیری عمیق و یادگیری ماشین که برای تشخیص شایعه یا طبقه‌بندی متن ارائه شده‌اند، مقایسه کرده‌ایم. همچنین، برای بررسی قوی بودن مدل‌های پیشنهادی، برخی از الگوریتم‌های یادگیری ماشین محبوب را برای تشخیص شایعه پیاده‌سازی کرده‌ایم. نتایج به‌دست آمده تشخیص شایعه هر مدل تحت مجموعه داده PHEME در جدول ۲ نشان داده شده است، در حالی که نتایج به‌دست آمده هر مدل تحت مجموعه داده لیار در جدول ۳ نشان داده شده است. این جداول دو نوع نتایج را نشان می‌دهد: نتایج تحت داده‌های آزمایش و نتایج تحت اعتبارسنجی متقاطع که به‌ازای $k=5$ انجام شده است. هر دو نوع نتایج شامل چهار معیار ارزیابی برای مجموعه داده‌های PHEME و لیار است که شامل دو دسته صفر و یک است. هر سطر از نتایج به‌دست آمده از جدول‌های ۲ و ۳ تحت مجموعه داده‌های شرح داده شده، یا حاصل پیاده‌سازی طبقه‌بندی ارائه شده از مقالات قبلی یا با استفاده از کتابخانه‌های موجود همانند کراس و سایکیت-لرن است. ما از همان پارامترها و روش‌های (برای مثال استخراج‌گر ویژگی) پیش‌فرض در مقالات و کتابخانه‌ها استفاده کرده‌ایم. شیوه‌ی آموزش و آزمایش این طبقه‌بندها دقیقاً همانند آموزش و آزمایش در مدل پیشنهادی است.

در جدول ۲ نتایج برای اعتبارسنجی متقاطع و داده‌های آزمایش گزارش شده است. از این نظر، مشاهده می‌شود که مدل پیشنهادی از نظر معیارهای دقت و $F1$ نسبت به دوازده مدل پایه پیش‌رو مورد مطالعه برتری دارد. به جز مدل پیشنهادی ارائه شده، ماشین بردار پشتیبان مبتنی بر سری زمانی بهترین عملکرد را دارد، جایی که جدول ۲ نشان می‌دهد که دقت مدل شبکه عصبی حلقوی ۸۴٫۳۱ است در حالی که ماشین بردار پشتیبان مبتنی بر سری زمانی ۸۳٫۵۳ است. در مدل پیشنهادی، پیش‌بینی‌های صحیح (بازیابی) توییت‌های

غیر شایعه بیشتر از توییت‌های شایعه‌دار است. این نتیجه منطقی است زیرا تعداد توییت‌های غیر شایعه بیش از شایعات است. از کارهای ارائه شده قبلی، ماشین بردار پشتیبان مبتنی بر سری زمانی برای مدل‌های یادگیری ماشین و شبکه عصبی حافظه طولانی کوتاه مدت برای مدل‌های یادگیری عمیق بهترین عملکرد را دارند، همچنین جدول ۲ نشان می‌دهد که مدل ترکیبی، شبکه عصبی حافظه طولانی کوتاه مدت دوجته و شبکه عصبی پیچشی بازگشتی به نتایج بسیار ضعیفی دست یافته‌اند. علاوه بر جدول ۲، جدول ۳ نتایج مدل پیشنهادی شبکه عصبی حلقوی (CMSCB) را با مدل‌های پایه مرز علم تحت مجموعه داده لیار مقایسه می‌کند. با توجه به نتایج بدست آمده از جدول ۳، می‌توان چند نتیجه زیر را بیان کرد:

- بین مدل‌های پایه، شبکه عصبی حافظه طولانی کوتاه مدت دو جهت و مدل ترکیبی عملکرد بسیار ضعیفی دارند.
- با توجه به جدول ۳، می‌توان گفت که مدل شبکه عصبی حلقوی یک روش قوی و کارآمد برای تشخیص شایعه است.
- با توجه جداول ۲ و ۳، اگرچه مدل پیشنهادی در هر دو مجموعه داده نسبت به مدل‌های پایه برتری دارد اما واضح است که دقت مدل تحت مجموعه داده PHEME بالاتر از لیار است و هنوز جای کار برای آینده باقی می‌ماند. البته حجم زیاد داده‌های مجموعه داده لیار نسبت به PHEME می‌تواند یکی از عوامل کاهش دقت باشد.

نتایج تجربی فرضیه در نظر گرفته شده در این مقاله را تأیید می‌کند. برای این فرضیه، شبکه عصبی پشته‌ای حلقوی از نظر دقت تحت دو مجموعه داده از مدل‌های پایه بهتر عمل کرده است که این نتیجه خوب نشان‌دهنده تأثیر در نظر گرفتن ویژگی‌های محلی و سراسری استخراج‌شده و ساختار حلقوی در نظر گرفته شده است. ما همچنین برای ارزیابی تأثیر حالت حلقوی پیشنهادی، مدل خود را بدون حالت حلقوی آزموده‌ایم.

^{۲۶}Precision

^{۲۷}Accuracy

جدول ۲: مقایسه نتایج CMSCB با مدل‌های پایه

PHEME Data	نتایج تحت اعتبارسنجی متقابل به‌ازای $k=5$					نتایج تحت داده‌های آزمایش				
	مدل	کلاس	دقت	صحت	بازیابی	F1	دقت	صحت	بازیابی	F1
RNN [17]	۰		۴۵,۰۱	۳۰,۰۷	۶۲,۸۶	۴۰,۶۷	۴۸,۸۷	۴۸,۹۵	۵۲,۵۸	۵۰,۷۰
	۱			۷۰,۱۶	۳۷,۳۸	۴۸,۷۷		۴۸,۷۸	۴۵,۱۷	۴۶,۹۱
LSTM [20]	۰		۸۴,۴۹	۷۶,۴۵	۷۳,۴۲	۷۳,۹۵	۸۲,۲۴	۸۸,۷۹	۷۳,۷۹	۸۰,۶۰
	۱			۸۸,۷۱	۸۹,۲۳	۸۸,۹۴		۷۷,۵۸	۹۰,۶۹	۸۳,۶۲
CNN [42]	۰		۸۲,۹۱	۷۷,۱۷	۷۰,۷۳	۷۳,۲۷	۷۸,۱۶	۹۲,۳۶	۶۱,۴۰	۷۳,۷۶
	۱			۸۵,۵۳	۸۹,۲۷	۸۷,۳۳		۷۱,۰۹	۹۴,۹۲	۸۱,۲۹
BILSTM [18]	۰		۲۹,۹۹	۲۹,۹۹	۱۰۰	۴۶,۱۴	۵۰	۵۰	۱۰۰	۶۶,۶۷
	۱		
CSI ^{۱۸} [19]	۰		۲۹,۹۹	۲۹,۹۹	۱۰۰	۴۶,۱۴	۵۰	۵۰	۱۰۰	۶۶,۶۷
	۱		
RCNN ^{۱۹} [43]	۰		۷۸,۶۶	.	.	.	۵۰	.	.	.
	۱			۷۸,۶۶	۱۰۰	۸۸,۰۵		۵۰	۱۰۰	۶۶,۶۷
AdaBoost ^{۲۰}	۰		۸۱,۱۶	۷۸,۵۴	۶۱,۴۰	۶۸,۹۰	۷۱,۷۹	۹۱,۰۲	۴۸,۳۵	۶۳,۱۶
	۱			۸۲,۱۳	۹۱,۳۳	۸۶,۴۸		۶۴,۸۴	۹۵,۲۳	۷۷,۱۵
SVM-TS [44] ^{۲۱}	۰		۸۵,۴۶	۷۶,۴۱	۷۴,۵۷	۷۵,۴۶	۸۳,۵۳	۹۱,۴۷	۷۳,۹۶	۸۱,۷۹
	۱			۸۹,۲۲	۹۰,۱۲	۸۹,۶۷		۷۸,۱۴	۹۳,۱۰	۸۴,۹۷
Random-forest ^{۲۲}	۰		۸۶,۸۶	۸۵,۸۰	۶۷,۳۹	۷۵,۴۲	۸۱,۰۳	۹۴,۳۳	۶۶,۰۳	۷۷,۶۸
	۱			۸۷,۲۲	۹۵,۱۹	۹۱,۰۳		۷۳,۸۷	۹۶,۰۳	۸۳,۵۰
SVM ^{۲۳}	۰		۸۶,۲۸	۸۴,۷۸	۶۶,۱۶	۷۴,۲۹	۸۰,۹۴	۹۳,۲۵	۶۶,۷۲	۷۷,۷۸
	۱			۸۶,۷۶	۹۴,۸۹	۹۶,۶۴		۷۴,۰۹	۹۵,۱۷	۸۳,۳۲
KNN ^{۲۴}	۰		۸۰,۳۱	۸۳,۲۷	۴۲,۹۶	۵۶,۶۱	۶۹,۲۲	۹۱,۴۴	۴۲,۴۱	۵۷,۹۵
	۱			۷۹,۷۸	۹۶,۳۱	۸۷,۲۶		۶۲,۵۱	۹۶,۰۳	۷۵,۷۳
SGD ^{۲۵}	۰		۸۵,۴۸	۷۷,۸۲	۷۲,۵۶	۷۴,۹۴	۸۳,۰۱	۹۱,۳۶	۷۲,۹۳	۸۱,۱۱
	۱			۸۸,۶۱	۹۱,۰۲	۸۹,۷۷		۷۷,۴۷	۹۳,۱۰	۸۴,۵۷
MSCB بدون حلقوی	۰		۸۴,۶۸	۷۲,۷۲	۷۵,۵۱	۷۵,۵۱	۸۲,۲۴	۹۰,۴۷	۷۲,۰۶	۸۰,۲۳
	۱			۹۱,۰۹	۸۶,۸۶	۸۸,۸۱		۷۶,۷۹	۹۲,۴۱	۸۳,۸۸
CMSCB مدل پیشنهادی	۰		۸۵,۰۵	۷۸,۴۲	۷۰,۶۲	۷۳,۷۹	۸۴,۳۱	۸۹,۶۴	۷۷,۵۸	۸۳,۱۷
	۱			۸۸,۰۲	۹۱,۲۳	۸۹,۵۰		۸۰,۲۴	۹۱,۰۳	۸۵,۲۹

^{۲۸} مدل ترکیبی ۹
^{۲۹} شبکه عصبی پیچشی بارگشتی

^{۳۰} ادابوست

^{۳۱} ماشین بردار پشتیبان مبتنی بر سری زمانی

^{۳۲} جنگل تصادفی

^{۳۳} ماشین بردار پشتیبان

^{۳۴} K نزدیک‌ترین همسایه

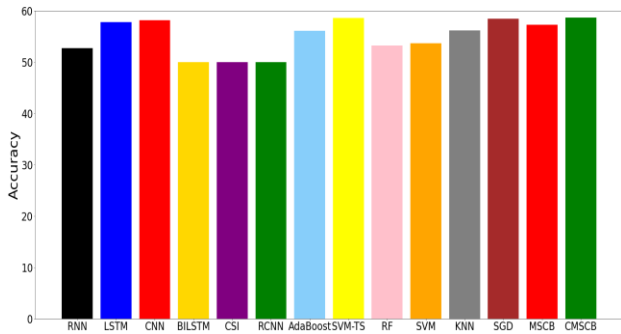
^{۳۵} کاهش گرادیان تصادفی

جدول ۳: مقایسه نتایج CMSCB با مدل‌های پایه

Liar Data	نتایج تحت اعتبارسنجی متقابل به‌ازای $k=5$					نتایج تحت داده‌های آزمایش				
	مدل	کلاس	دقت	صحت	بازیابی	F1	دقت	صحت	بازیابی	F1
RNN [17]	۰		۵۰,۴۷	۲۷,۷۳	۴۹,۱۸	۳۵,۴۴	۵۲,۷۳	۵۲,۷۰	۵۳,۲۰	۵۲,۹۵
	۱			۷۲,۳۹	۵۰,۹۸	۵۹,۸۰		۵۲,۷۶	۵۳,۲۶	۵۲,۵۱
LSTM [20]	۰		۷۰,۱۹	۳۲,۲۶	۳۱	۳۱,۵۹	۵۷,۸۱	۶۱,۹۶	۴۰,۴۶	۴۸,۹۶
	۱			۸۰,۵۰	۸۱,۳۹	۸۰,۹۴		۵۵,۸۰	۷۵,۱۵	۶۴,۰۴
CNN [42]	۰		۷۱,۱۳	۴۶,۸۵	۳۲,۰۱	۳۸,۰۶	۵۸,۱۶	۶۸,۸۲	۲۹,۸۴	۴۱,۶۳
	۱			۷۶,۸۲	۸۶,۰۶	۸۱,۱۷		۵۵,۲۱	۸۶,۴۸	۶۷,۳۹
BILSTM [18]	۰		۲۲,۲۳	۲۲,۲۳	۱۰۰	۳۶,۳۷	۵۰	۵۰	۱۰۰	۶۶,۶۷
	۱		
CSI [19]	۰		۳۳,۹۶	۳۳,۹۶	۱۰۰	۵۰,۶۸	۵۰	۵۰	۱۰۰	۶۶,۶۷
	۱		
RCNN [43]	۰		۷۷,۷۷	.	.	.	۵۰	.	.	.
	۱			۸۷,۵	۱۰۰	۸۷,۵		۵۰	۱۰۰	۶۶,۶۷
AdaBoost	۰		۷۳,۴۸	۵۸,۰۶	۱۶,۶۰	۲۵,۷۹	۵۶,۱۳	۸۴,۸۸	۱۴,۹۲	۲۵,۳۸
	۱			۷۴,۸۲	۹۵,۳۶	۸۳,۸۵		۵۳,۳۶	۹۷,۳۴	۶۸,۹۳
SVM-TS [44]	۰		۶۷,۲۷	۴۰,۸۱	۳۹,۲۷	۴۰,۰۱	۵۸,۶۳	۶۶,۲۲	۳۵,۲۳	۴۵,۹۹
	۱			۷۶,۹۶	۷۸,۰۴	۷۷,۴۹		۵۵,۸۸	۸۲,۰۳	۶۶,۴۷
Random-forest	۰		۷۴,۳۷	۷۴,۳۶	۱۱,۹۰	۲۰,۵۰	۵۳,۲۸	۸۸,۸۸	۷,۵	۱۳,۸۳
	۱			۷۴,۳۸	۹۸,۴۰	۸۴,۷۲		۵۱,۷۱	۹۹,۰۶	۶۷,۹۵
SVM	۰		۷۴,۲۱	۷۳,۴۷	۱۱,۳۹	۱۹,۶۹	۵۳,۷۱	۹۵,۲۳	۷,۸۱	۱۴,۴۴
	۱			۷۴,۲۶	۹۸,۳۸	۸۴,۶۴		۵۱,۹۳	۹۹,۶۰	۶۸,۲۷
KNN	۰		۶۷,۴۶	۸۳,۲۷	۴۲,۹۶	۵۶,۶۱	۵۶,۲۱	۶۵,۴۳	۲۶,۳۲	۳۷,۵۴
	۱			۷۹,۷۸	۹۶,۳۱	۸۷,۲۶		۵۳,۸۸	۸۶,۰۹	۶۶,۲۸
SGD	۰		۶۸,۴۴	۴۲,۳۹	۳۷,۵۳	۳۹,۷۷	۵۸,۴۷	۶۷,۵۲	۳۲,۶۵	۴۴,۰۲
	۱			۷۶,۹۷	۸۰,۳۳	۷۸,۶۱		۵۵,۵۸	۸۴,۲۹	۵۸,۴۷
MSCB بدون حلقوی	۰		۷۳,۳۶	۴۰,۰۷	۳۵,۶۷	۳۶,۵۷	۵۷,۳۰	۷۵,۹۰	۲۱,۴۰	۳۳,۳۹
	۱			۸۲,۱۶	۸۴,۱۳	۸۲,۹۹		۵۴,۲۵	۹۳,۲۰	۶۸,۵۸
CMSCB مدل پیشنهادی	۰		۷۵,۵۷	۴۴,۱۱	۲۵,۵۰	۳۱,۰۱	۵۸,۷۱	۷۲,۷	۲۷,۰۳	۳۹,۵۶
	۱			۸۰,۸۹	۸۹,۸۸	۸۵,۰۸		۷	۹۰,۳۹	۶۸,۶۴
								۵۵,۳		
								۳		

لیار را از لحاظ دقت را نشان می‌دهند. همانطور که مشاهده می‌شود شبکه عصبی حلقوی از تمامی مدل‌های مورد مقایسه کارآمدی بیشتری دارد حتی بیشتر از شبکه عصبی پیشنهادی بدون ساختار حلقوی.

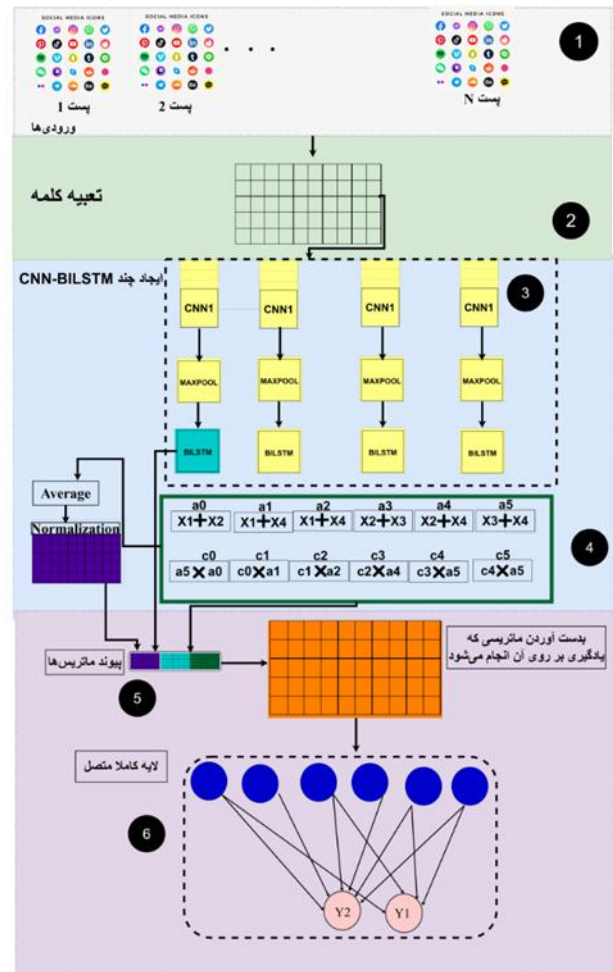
دقت این مدل بدون حالت حلقوی برای مجموعه داده‌های PHEME و لیار به ترتیب برابر با ۸۲,۲۴ و ۵۷,۳۱ است، در حالی که دقت مدل با ساختار حلقوی برابر با ۸۴,۳۱ و ۵۸,۷۱ درصد است که نشان می‌دهد حالت حلقوی برای تشخیص شایعه تأثیر مثبتی دارد. شکل‌های ۲ و ۳ به ترتیب کارایی مدل‌های متفاوت تحت مجموعه داده‌های PHEME و



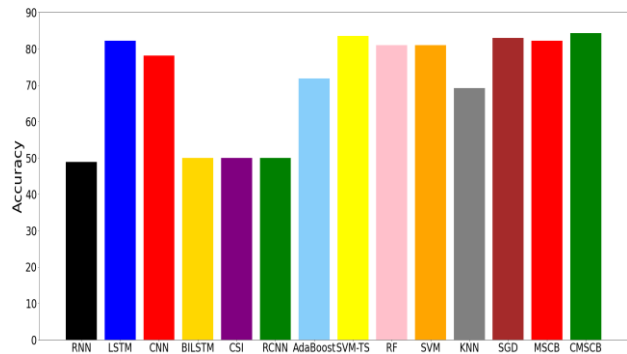
شکل (۳): مقایسه مدل شبکه عصبی پشته‌ای حلقوی با دیگر مدل‌های پایه از لحاظ دقت تحت مجموعه داده لیار

۵. نتیجه‌گیری

شبکه‌های اجتماعی به‌طور گسترده‌ای مورد استفاده قرار می‌گیرند و در زندگی روزانه افراد تأثیر به‌سزایی دارند. این شبکه‌ها مزایای زیادی نظیر سهولت در برقراری ارتباطات جهانی، تجارت الکترونیکی، افزایش آگاهی و سهولت در دسترسی به آخرین رویدادهای جهانی را دارند. اما یکی از عوامل منفی در شبکه‌های اجتماعی وجود شایعات است که می‌توانند منجر به آثار مخرب مالی و جانی شوند. به‌عنوان مثال، شایعات خطرناکی مانند تشویق به مصرف مواد شوینده شیمیایی به عنوان راهی برای جلوگیری از ابتلا به ویروس کرونا می‌تواند برای مردم خطرناک باشد. در این مطالعه، یک شبکه عصبی پشته‌ای حلقوی برای تشخیص شایعات پیشنهاد شده است. معمولاً شایعات به شیوه‌ای فریبکارانه نوشته می‌شوند که حتی متخصصان نیز نمی‌توانند آنها را تشخیص دهند. به همین دلیل، نیاز به مدل‌هایی است که بتوانند ویژگی‌های متمایز را استخراج کنند. برای استخراج ویژگی‌های متمایز شایعات و غیرشایعات، از شبکه‌های عصبی پشته شده برای استخراج ویژگی‌های محلی و از شبکه‌های عصبی حافظه کوتاه مدت دو جهته برای در نظر گرفتن وابستگی بین ویژگی‌های محلی استفاده می‌شود. در این مطالعه، از چندین لایه شبکه عصبی پیچشی پشته شده-شبکه عصبی حافظه طولانی کوتاه مدت دو جهته برای استخراج بیشتر ویژگی‌های متمایز استفاده شده است. علاوه بر این، برای اولین بار ساختار حلقوی به مدل شبکه عصبی پیچشی پشته شده-شبکه عصبی حافظه طولانی کوتاه مدت دو جهته اضافه شده و نشان داده شده است که ساختار حلقوی می‌تواند در بهبود کارایی مدل برای طبقه‌بندی متن مؤثر باشد. در این مطالعه، مدل پیشنهادی با دوازده مدل یادگیری ماشین و یادگیری عمیق مقایسه شده و نتایج نشان می‌دهد که عملکرد مدل پیشنهادی برتری امیدوارکننده‌ای نسبت به مدل‌های مورد مطالعه دارد.



شکل (۱): مدل شبکه عصبی پشته‌ای حلقوی پیشنهادی



شکل (۲): مقایسه مدل شبکه عصبی پشته‌ای حلقوی با دیگر مدل‌های پایه از لحاظ معیار دقت تحت مجموعه داده PHEME

- Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2018, .vol. 1: IEEE, pp. 109-113
<https://doi.org/10.1109/IHMSC.2018.00032>
- [10] M. Goudjil, M. Koudil, M. Bedda, and N. Ghoggali, "A novel active learning method using SVM for text classification," *International Journal of Automation and Computing*, vol. 15, pp. 290-298, 2018
<https://doi.org/10.1007/s11633-015-0912-z>
- [11] A. Bouaziz, C. Dartigues-Pallez, C. da Costa Pereira, F. Precioso, and P. Lloret, "Short text classification using semantic random forest," in *Data Warehousing and Knowledge Discovery: 16th International Conference, DaWaK 2014, Munich, Germany, September 2-4, 2014. Proceedings 16*, 2014: Springer, pp. 288-299
https://doi.org/10.1007/978-3-319-10160-6_26
- [12] S. Wang, L. Jiang, and C. Li, "Adapting naive Bayes tree for text classification," *Knowledge and Information Systems*, vol. 44, pp. 77-89, 2015
<https://doi.org/10.1007/s10115-014-0746-y>
- [13] M.-R. Feizi-Derakhshi, Z. Mottaghinia, and M. Asgari-Chenaghlu, "Persian Text Classification Based on Deep Neural Networks," *Soft Computing Journal*, vol. 11, no. 1, 2022 [In Persian].
<https://doi.org/10.22052/SCJ.2023.243182.1010>
- [14] N. Majma and S. Bashtin, "Detection of plagiarism in scientific texts based on text blocking and cosine similarity criteria," *Soft Computing Journal*, vol. 11, no. 1, pp. -, 2022 [In Persian].
<https://doi.org/10.22052/SCJ.2022.243384.1034>
- [15] F. Zare Mehrjardi, M. Yazdian-Dehkordi, and A. Latif, "Evaluating Machine learning and Deep-learning methods in Sentiment Analysis on Persian Telegram Message," *Soft Computing Journal*, vol. 11, no. 1, pp. -, 2022 [In Persian].
<https://doi.org/10.22052/SCJ.2023.246553.1077>
- [16] R. Behzadidoost, F. Mahan, and H. Izadkhah, "Granular computing-based deep learning for text classification," *Information Sciences*, vol. 652, p. 119746, 2024
<https://doi.org/10.1016/j.ins.2023.119746>
- R. Behzadidoost, M. Hasheminezhad, M. Farshi, V. Derhami, and F. Alamiyan-Harandi, "A framework for text mining on twitter: a case study on joint comprehensive plan of action (jcpoa)-between 2015 and 2019," *Quality & Quantity*, pp. 1-32, 2021
<https://doi.org/10.1007/s11135-021-01239-y>
- [2] A. Rapp, L. S. Beitelspacher, D. Grewal, and D. E. Hughes, "Understanding social media effects across seller, retailer, and consumer interactions," *Journal of the Academy of Marketing Science*, vol. 41, no. 5, pp. 547-566, 2013
- [3] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques," *Information Sciences*, vol. 497, pp. 38-55, 2019
<https://doi.org/10.1007/s11747-013-0326-9>
- [4] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. e1249, 2018 .8, no. 4, p
<https://doi.org/10.1002/widm.1249>
- [5] T. Lehinevych and H. Andii, "Analysis of Deep Metric Learning Approaches," in *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT)*, 2019: IEEE, pp. 369-372
<https://doi.org/10.1109/ATIT49449.2019.9030440>
- [6] C. Kühnert, M. Baruthio, T. Bernard, C. Steinmetz, and J.-M. Weber, "Cloud-based event detection platform for water distribution networks using machine-learning algorithms," *Procedia Engineering*, vol. 119, pp. 901-907, 2015
<https://doi.org/10.1016/j.proeng.2015.08.963>
- [7] W. F. Lamberti, "An overview of explainable and interpretable AI," *AI Assurance*, pp. 55- and interpretable AI," *AI Assurance*, pp. 55-123, 2023
<https://doi.org/10.1016/B978-0-32-391919-7.00015-9>
- [8] T. Dietterich, "Overfitting and undercomputing in machine learning," *ACM computing surveys (CSUR)*, vol. 27, no. 3, pp. 326-327, 1995
<https://doi.org/10.1145/212094.212114>
- [9] Y. Tan, "An improved KNN text classification algorithm based on K-medoids set," in *2018 10th International and rough*

- Network for early rumor detection," *Expert Systems with Applications*, vol. 195, p. .116517, 2022
<https://doi.org/10.1016/j.eswa.2022.116517>
- [26] B. Liu *et al.*, "Nowhere to hide: Online rumor detection based on retweeting graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, .2022
<https://doi.org/10.1109/TNNLS.2022.3161697>
- [27] H. Ran, C. Jia, P. Zhang, and X. Li, "MGAT-ESM: Multi-channel graph attention neural network with event-sharing module for rumor detection," *Information Sciences*, .pp. 402-416, 2022 ,92vol. 5
<https://doi.org/10.1016/j.ins.2022.01.036>
- [28] X. Chen, F. Zhou, G. Trajcevski, and M. Bonsangue, "Multi-view learning with distinguishable feature fusion for rumor detection," *Knowledge-Based Systems*, vol. .240, p. 108085, 2022
<https://doi.org/10.1016/j.knosys.2021.108085>
- [29] P. Zheng, Z. Huang, Y. Dou, and Y. Yan, "Rumor detection on social media through mining the social circles with high homogeneity," *Information Sciences*, vol. .642, p. 119083, 2023
<https://doi.org/10.1016/j.ins.2023.119083>
- [30] W. Cui and M. Shang, "KAGN: knowledge-powered attention and graph convolutional networks for social media rumor detection," *Journal of big Data*, vol. 10, no. .1, p. 45, 2023
<https://doi.org/10.1186/s40537-023-00725-4>
- [31] Z. Guo, K. Yu, A. Jolfaei, G. Li, F. Ding, and A. Beheshti, "Mixed graph neural network-based fake news detection for sustainable vehicular social networks," *IEEE Transactions on Intelligent Transportation Systems*, 2022
<https://doi.org/10.1109/TITS.2022.3185013>
- [32] H. Han, Z. Ke, X. Nie, L. Dai, and W. Slamu, "Multimodal Fusion with Dual-Attention Based on Textual Double-Embedding Networks for Rumor Detection," *Applied Sciences*, vol. 13, no. 8, p. 4886, 2023
<https://doi.org/10.3390/app13084886>
- [33] Chen, X. Hu, and S. Ji, .H. Yuan, Y "Interpreting deep models for text analysis via optimization and regularization methods," in *Proceedings of the AAAI*
- [17] J. Ma *et al.*, "Detecting rumors from microblogs with recurrent neural .2016 ",networks
- [18] A. Kumar, V. Singh, T. Ali, S. Pal, and J. Singh, "Empirical evaluation of shallow and deep classifiers for rumor detection," in *Advances in Computing and Intelligent Systems: Proceedings of ICACM 2019*, 2020: .Springer, pp. 239-252
https://doi.org/10.1007/978-981-15-0222-4_21
- [19] N. Ruchansky, S. Seo, and Y. Liu, "Csi: A hybrid deep model for fake news detection," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 797-806
<https://doi.org/10.1145/3132847.3132877>
- [20] A. S. Torshizi and A. Ghazikhani, "Automatic Twitter rumor detection based on LSTM classifier," in *High-Performance Computing and Big Data Analysis: Second International Congress, TopHPC 2019, Tehran, Iran, April 23-25, 2019, Revised Selected Papers 2*, .2019: Springer, pp. 291-300
https://doi.org/10.1007/978-3-030-33495-6_22
- [21] T. Nguyen, B. .T. T. Nguyen, T. T. Nguyen, T Vo, J. Jo, and Q. V. H. Nguyen, "Judo: Just-in-time rumour detection in streaming social platforms," *Information Sciences*, vol. .570, pp. 70-93, 2021
<https://doi.org/10.1016/j.ins.2021.04.018>
- [22] F. Xu, V. S. Sheng, and M. Wang, "Near real-time topic-driven rumor detection in source microblogs," *Knowledge-Based Systems*, .vol. 207, p. 106391, 2020
<https://doi.org/10.1016/j.knosys.2020.106391>
- [23] Z. Wu, D. Pi, J. Chen, M. Xie, and J. Cao, "Rumor detection based on propagation graph neural network with attention mechanism," *Expert systems with .2020 ,applications*, vol. 158, p. 113595
<https://doi.org/10.1016/j.eswa.2020.113595>
- [24] P. Zhang, H. Ran, C. Jia, X. Li, and X. Han, "A lightweight propagation path aggregating network with neural topic model for rumor detection," *Neurocomputing*, vol. 458, pp. .468-477, 2021
<https://doi.org/10.1016/j.neucom.2021.06.062>
- [25] H. Zhou, T. Ma, H. Rong, Y. Qian, Y. Tian, Al-Nabhan, "MDMN: Multi-task and .and N Domain Adaptation based Multi-modal

conference on artificial intelligence, 2015,
.vol. 29, no. 1
<https://doi.org/10.1609/aaai.v29i1.9513>
J. Ma, W. Gao, Z. Wei, Y. Lu, and K.-F.
Wong, "Detect rumors using time series of
social context information on
microblogging websites," in *Proceedings of
the 24th ACM international on conference
on information and knowledge
.management*, 2015, pp. 1751-1754
https://doi.org/10.1142/9789813223615_0006

- [44] M. Cassel and F. Lima, "Evaluating one-hot
encoding finite state machines for SEU
reliability in SRAM-based FPGAs," in *12th
IEEE International On-Line Testing
.Symposium (IOLTS'06)*, 2006: IEEE, p. 6 pp
<https://doi.org/10.1109/IOLTS.2006.32>
- [35] Y. Goldberg and O. Levy, "word2vec
Explained: deriving Mikolov et al.'s
negative-sampling word-embedding
method," *arXiv preprint arXiv:1402.3722*,
.2014
- [36] 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)*, 2014, pp.
.1532-1543
- [37] A. Fidalgo, I. Medeiros, P. Antunes, and N.
Neves, "Towards a deep learning model for
vulnerability detection on web application
variants," in *2020 IEEE International
Conference on Software Testing,
and Validation Workshops Verification
(ICSTW)*, 2020: IEEE, pp. 465-476
<https://doi.org/10.1109/ICSTW50294.2020.00083>
- [38] R. J. Williams and D. Zipser, "Gradient-
based learning algorithms for recurrent,"
*Backpropagation: Theory, architectures,
.and applications*, vol. 433, p. 17, 1995
- [39] Scikit-learn: Machine " ,.F. Pedregosa *et al*
learning in Python," *the Journal of machine
Learning research*, vol. 12, pp. 2825-2830,
.2011
- [40] N. Ketkar and N. Ketkar, "Introduction to
keras," *Deep learning with python: a hands-
.on introduction*, pp. 97-111, 2017
https://doi.org/10.1007/978-1-4842-2766-4_7
- [41] W. Y. Wang, "" liar, liar pants on fire": A
new benchmark dataset for fake news
detection," *arXiv preprint
.arXiv:1705.00648*, 2017
10.18653/v1/P17-2067
- [42] Y. Kim, "Convolutional neural networks for
sentence classification," *arXiv preprint
.arXiv:1408.5882*, 2014
10.3115/v1/D14-1181
- [43] Liu, and J. Zhao, "Recurrent .S. Lai, L. Xu, K
convolutional neural networks for text
classification," in *Proceedings of the AAAI*