



دانشگاه کاشان  
University of Kashan

مجله محاسبات نرم

## SOFT COMPUTING JOURNAL

تارنمای مجله: [scj.kashanu.ac.ir](http://scj.kashanu.ac.ir)



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

حیب ایزدخواه<sup>1\*</sup>، دانشیار، رشید بهزادی دوست<sup>1</sup>، کاندید دکتری

<sup>1</sup> دانشکده ریاضی، آمار و علوم کامپیوتر، دانشگاه تبریز، تبریز، ایران.

#### اطلاعات مقاله

#### چکیده

تاریخچه مقاله:

دریافت 3 مرداد ماه 1402

پذیرش 28 آذر ماه 1402

کلمات کلیدی:

پردازش زبان طبیعی

یادگیری عمیق

تشخیص شایعه

شبکه عصبی

شبکه اجتماعی

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 شده است. 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 پشته‌ای-شبکه عصبی حافظه طولانی کوتاه مدت دوجته 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.

© 1402 نویسندگان. مقاله با دسترسی آزاد تحت مجوز CC-BY

#### 1. مقدمه

دانشمندان داده و محققان در زمینه پردازش زبان طبیعی شناخته می‌شوند [1]. علی‌رغم مزایای زیاد شبکه‌های اجتماعی [2]، 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 تایید شدن، به طور معمول 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. بر خلاف شایعات، اخبار جعلی به عمد ساخته و پخش می‌شوند و هدف آنها گمراه کردن عموم است. اخبار جعلی اغلب از عناوین

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 می‌تواند برای تحلیل و پژوهش مورد استفاده قرار گیرند. به همین دلیل، شبکه‌های اجتماعی به عنوان یکی از بهترین منابع برای

\* نوع مقاله: پژوهشی

\* نویسنده مسئول

پست(های) الکترونیک: [izadkhah@tabrizu.ac.ir](mailto:izadkhah@tabrizu.ac.ir) (ایزدخواه)

[r.behzadi@tabrizu.ac.ir](mailto:r.behzadi@tabrizu.ac.ir) (بهزادی دوست)

صورت 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 دشنام 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. به علاوه، ویژگی‌های واژگانی می‌توانند به شناسایی الگوهای خاص در متن کمک کنند. به طور مثال، کلمات دشنام، کلمات نفی و ساختار جملات می‌توانند به شناسایی شایعه کمک کنند [4].

نوآوری اصلی این مقاله در پیشنهاد چند شبکه عصبی پیچشی<sup>1</sup> پشته شده<sup>2</sup>-شبکه عصبی حافظه طولانی کوتاه مدت دو جهته<sup>3</sup>، با ساختار حلقوی برای تشخیص شایعه به شکلی کارآمد است. شایعه‌سازان از کلمات و جملاتی با ساختار فریبنده 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 به دست 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 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.

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. به عنوان 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، نظرکاوی و غیره هستند. با این حال، تشخیص شایعه و غیرشایعه با استفاده از ویژگی‌های مبتنی بر محتوا ممکن است سخت باشد، زیرا مجموعه کلمات شایعه و غیرشایعه ممکن است بسیار شبیه بهم 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. به طور کلی، ویژگی‌های مبتنی بر زمینه می‌توانند به تشخیص شایعه کمک کنند، اما باید با دقت به کار گرفته شوند [3].

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) به

<sup>1</sup> Convolutional Neural Network (CNN)

<sup>2</sup> Stacked

<sup>3</sup> Bidirectional long short-term memory network (BiLSTM)

3. پیچیدگی بالا [7]: بسیاری از الگوریتم‌های یادگیری ماشین از نظر زمان و حافظه پیچیدگی بالایی دارند. این ویژگی در مدل‌های مبتنی بر یادگیری عمیق بسیار بیشتر است.

4. بیش‌برازش [8]: 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 [9] K-RS نزدیک‌ترین همسایه را ارائه کردند که نسبت به الگوریتم  $k$  نزدیک‌ترین همسایه معمول پیچیدگی زمانی کمتری داشت. آنها در نتایج خود نشان دادند که K-RS نزدیک‌ترین همسایه از  $k$  نزدیک‌ترین همسایه اولیه بهتر عمل می‌کند. گودجیل و همکاران [10] یک مدل یادگیر فعال جدید ارائه کردند که هدف اصلی آن برای کاهش زمان زیاد برچسب‌گذاری بود به گونه‌ای که بجای 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 [11] 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، نبی‌تری<sup>2</sup> [12] را ارائه دادند. نبی‌تری 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 بهترین رابطه انتخاب شود خود یک چالش است. به عنوان مثال، در الگوریتم ماشین بردار پشتیبان، هسته‌های زیادی وجود دارد و

مدل پیشنهادی شامل چندین لایه‌ی شبکه عصبی پیچشی پشت‌پشته شده-شبکه عصبی حافظه طولانی کوتاه مدت دوجهته<sup>1</sup> 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:

- 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 2 به مطالعات مرتبط پرداخته است، بخش 3 روش پیشنهادی را توصیف می‌کند، بخش 4 نتایج آزمایشی را ارائه می‌دهد و بخش 5 نتیجه‌گیری را ارائه می‌دهد.

## 2. 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 معنایی هستند:

1. وابستگی به داده‌ها [5]: 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 تحت‌تاثیر قرار بگیرد.
2. جعبه سیاه [6]: 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.

<sup>2</sup> NBTREE

<sup>1</sup> SCNN-BILSTM

استفاده کردند که در آن از موضوع توییت‌های داده شده برای تشخیص شایعه استفاده شده است. آنها دقت 82/66% را برای مجموعه داده PHEME و دقت 62/24% را برای مجموعه داده لیار<sup>3</sup> در مقایسه با 11 روش مطالعه شده تحت اعتبارسنجی متقاطع 5 مرحله‌ای به دست آوردند. در کار دیگری، یوی و همکاران [23] با استفاده از شبکه عصبی نمودار انتشار با مکانیسم توجه، یک طبقه‌بندی چهار کلاسه برای تشخیص شایعه در مجموعه داده‌های PHEME را انجام دادند. ژانگ و همکاران [24] از یک شبکه جمع‌آوری مسیر انتشار سبک برای تشخیص شایعه استفاده کردند. برای مجموعه داده PHEME، آنها به دقت 80% دست یافتند. نویسندگان در [25] بر خلاف دیگر کارها یک شبکه عصبی چند-مدل برای تشخیص اولیه 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 ویو کارآمد است. به‌خاطر در نظر گرفتن چندین مدل یادگیری عمیق و استخراج کننده‌های ویژگی مجزا برای متن و تصویر، پیچیدگی این مدل بالا است. در مقاله [26]، 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 [27]، نویسندگان یک مدل شبکه عصبی توجه گراف چند کانالی با ماژول اشتراک‌گذاری رویداد برای تشخیص شایعه ارائه دادند که همانند مطالعه [25]، پیچیدگی این مدل بالا بود. در تحقیق [28]، 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 ویو، توییت 15 و 16 را در نظر گرفتند. مطالعات در زمینه انتشار اطلاعات نشان داده‌اند که دایره اجتماعی یک نقش قابل توجه در سرعت، محدوده و

ممکن است انتخاب هر کدام یک بر کارایی مدل تاثیر 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 [13]، پیشنهاد شده است که از شبکه‌های عصبی پیچشی و شبکه عصبی حافظه طولانی کوتاه مدت دو جهته با لایه توجه برای طبقه‌بندی متون فارسی استفاده شود. نویسندگان در [14]، با استفاده از معیارهای مسدود کردن متن و شباهت کسینوسی، کپی‌برداری در متون علمی را شناسایی کردند. در یک کار مقایسه‌ای و کاربردی [15]، روش‌های یادگیری ماشین و یادگیری عمیق در تحلیل احساسات پیام‌های تلگرام به زبان فارسی مورد ارزیابی قرار گرفته است. در یک مقاله اخیر [16]، یک مدل یادگیری عمیق مبتنی بر محاسبات دانه‌ای برای طبقه‌بندی متن ارائه شده است. تشخیص شایعه یک حوزه فعال تحقیقاتی است. ما و همکاران [17] 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 ویو بهتر عمل می‌کند. کار دیگر در زمینه تشخیص شایعه مربوط به کار کومار و همکاران بود [18]. آنها در ارزیابی‌های تجربی خود نشان دادند که بهترین دقت بر روی مجموعه داده PHEME، 78/40% بود. روچانسکی و همکاران [19] یک مدل عمیق ترکیبی<sup>1</sup> برای تشخیص اخبار جعلی ارائه کردند که در آن مدل، آنها از ویژگی‌های ترکیبی استفاده می‌کردند. مدل بعدی برای تشخیص شایعه، طبقه‌بندی کننده شبکه عصبی حافظه طولانی کوتاه مدت<sup>2</sup> بود [20]. آنها قبل از طبقه‌بندی، نرمال‌سازی هشتگ و خوشه‌بندی هشتگ را انجام دادند. برخلاف کارهای قبلی، نوین [21] در مقاله‌ای روی زمان تشخیص یک شایعه تمرکز داشت. خو و همکاران در سال 2020 [22]، از یک مدل تشخیص شایعه مبتنی بر موضوع

<sup>1</sup> Hybrid

<sup>2</sup> Long Short Term Memory (LSTM)

<sup>3</sup> Liar

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 توئیتر انجام شده است.

### 3. مدل پیشنهادی

در این بخش اجزای مدل پیشنهادی توصیف می‌شوند. مدل پیشنهادی را به اختصار CMSCB<sup>2</sup> می‌نامیم. برای تشخیص شایعات، از مدل‌های یادگیری عمیق استفاده می‌کنیم. تشخیص شایعه در این مطالعه، یک کار طبقه‌بندی متن است. به طور 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.

#### 1.3. تعبیه کلمه

اکثر مدل‌های یادگیری ماشین و یادگیری عمیق برای تفسیر کلمات باید آنها را به یک نمایش عددی تبدیل کنند [33]. 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<sup>3</sup> [34] 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 معروف واژه به بردار<sup>4</sup> [35] و بردارهای جهانی برای نمایش کلمه<sup>5</sup> [36] هستند که می‌توانند اطلاعات معنایی را در بین کلمات در نظر بگیرند. به عبارتی هرچه کلمات شباهت بیشتری داشته باشند، نمایش مشابه

محتوای شایعه پخش شده ایفا می‌کند. با توجه به این نکته مهم، الگوریتم تشخیص شایعه جدیدی در [29] 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 (توییتر 15، توئیتر 16 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 PHEME را استفاده کردند. در مقاله [30]، یک مدل شبکه عصبی ارائه داده شده است که از دانش خارجی موجود در گراف دانش برای تشخیص شایعه استفاده می‌کرد. 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 (توییتر 15، توئیتر 16 و واقعیت سیاسی و PHEME) کارایی این مدل را نشان می‌دهد. آنها فقط بخشی از داده‌های PHEME را استفاده کردند. در مقاله [31]، یک مدل تشخیص محتوای جعلی بر اساس شبکه‌های عصبی گراف ترکیب شده<sup>1</sup> 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 [32]، یک شبکه

<sup>2</sup> Circular Multi Stacked CNN-BILSTM

<sup>3</sup> One-hot

<sup>4</sup> Word2vec

<sup>5</sup> Glove

<sup>1</sup> GNN

بیشتری خواهند داشت.

### 2.3. شبکه‌های عصبی پیچشی

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 [38].

### 4.3. 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 به دست آمده توسط شبکه‌های عصبی پیچشی، توسط شبکه عصبی حافظه طولانی کوتاه مدت دو جهته گرفته می‌شود. از آنجایی که مسائل طبقه‌بندی متن ماهیتا ابعاد بالایی دارند و برای آنکه ویژگی‌های مناسب و خاص را تا حداقل 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 به صورت دوجه دو با یکدیگر جمع می‌شوند و سپس دوجه دو آنها به صورت حلقوی با یکدیگر ضرب می‌شوند. بعد از میانگین‌گیری و نرمال‌سازی، بر روی ماتریس‌ها در مرحله قبل، مدل ارائه شده یادگیری ساختار شایعات و غیرشایعات بر روی تک ماتریس به دست آمده را انجام می‌دهد.

در یادگیری عمیق، ویژگی‌ها برای یک مسئله داده شده به صورت 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 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

شبکه‌های عصبی پیچشی یک شبکه بازخورد روبه‌جلو<sup>1</sup> 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 [37].

### 3.3. حافظه کوتاه مدت 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<sup>2</sup> برای حل این مشکل پیشنهاد شده‌اند. در شبکه‌های عصبی بازگشتی، توسط برخی از گیت‌های داخلی، حالت‌های قبلی به خاطر سپرده می‌شوند. با استفاده از یک شبکه عصبی بازگشتی، سیگنال‌های خطای انتشار به عقب ممکن است ناپدید شوند. به اصطلاح محوشدگی گرادیان<sup>3</sup> 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

<sup>1</sup> Feed-forward

<sup>2</sup> RNN

<sup>3</sup> Vanishing gradient

$\eta$  برداری است که از میانگین تمامی بردارها در ساختار حلقوی به دست آمده است،  $\mu$  میانگین ریاضی و  $v$  و واریانس  $\eta$  هستند. در نهایت، برای در نظر گرفتن ویژگی‌های به دست آمده از بردارهای مختلف، بردارهای به دست آمده از رابطه (4)، رابطه (5) و  $B_1$  (اولین شبکه عصبی پیچشی پشته شده وصل شده به شبکه عصبی حافظه طولانی کوتاه مدت دوجته) پیوند داده می‌شوند  $CO(norm, B, \eta) = Concat(\eta, norm, B)$  (وقتی  $C_1 \in \mathbb{R}^{3 \times m}$ ) 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  $\eta$  و  $B_1$ . در نهایت،  $CO$ ، به یک لایه کاملاً متصل، اتصال داده می‌شود.

شکل (1) مدل پیشنهاد شده در این مقاله را نشان می‌دهد. در مرحله 1 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 فضاهای برداری پیوسته عددی تبدیل می‌شوند. در مرحله 3، چند CNN-BILSTM ایجاد می‌شود که در آن چهار شبکه عصبی پیچشی پشته شده با فرآیندهای مختلف و چهار شبکه عصبی حافظه طولانی کوتاه مدت دوجته با فرآیندهای یکسان در نظر گرفته می‌شوند.

برای هر شبکه عصبی پیچشی پشته شده، شبکه‌های عصبی پیچشی متفاوتی در نظر گرفته می‌شود. به‌نحوی که هر کدام اندازه هسته، پولینگ<sup>1</sup> و فیلتر متفاوتی دارند. برای کاهش تعداد پارامترها در هر 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 هر بار با آن هسته به‌روزرسانی می‌شوند. در ادامه، شبکه عصبی پیچشی پشته شده با یکدیگر ادغام می‌شوند و برای کاهش فرآیندها

پشته شده  $CB_{K,T}$ ،  $K = 1, 2, \dots, Q$ ،  $T = 1, 2, \dots, P$  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 به دست می‌آید که در رابطه (1) نشان داده شده است. حالت‌های جمع و حلقوی در رابطه (2) و (3) نشان داده شده‌اند، به طوری 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.

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

$$\begin{aligned} a_0 &= B_1 + B_2, \quad a_0 \in \mathbb{R}^m \\ a_1 &= B_2 + B_3, \quad a_1 \in \mathbb{R}^m \\ &\dots \end{aligned} \quad (2)$$

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

$$\begin{aligned} c_0 &= a_0 \times a_{y-1}, \quad c_0 \in \mathbb{R}^m \\ c_1 &= c_0 \times a_1, \quad c_1 \in \mathbb{R}^m \\ &\dots \end{aligned} \quad (3)$$

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

بعد از آن، در مرحله چهارم میانگین از عبارت‌های رابطه سوم گرفته می‌شود:

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

در مرحله بعد، بردار بدست آمده از رابطه (4) با رابطه (5) نرمال‌سازی می‌شود:

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

<sup>1</sup> Pooling

$$a_0 = x_1 + x_2$$

$$a_1 = x_1 + x_3$$

$$a_2 = x_1 + x_4$$

$$a_3 = x_2 + x_3$$

$$a_4 = x_2 + x_4$$

$$a_5 = x_3 + x_4$$

$$c_0 = a_0 \times a_5$$

$$c_1 = c_0 \times a_1$$

$$c_2 = c_1 \times a_2$$

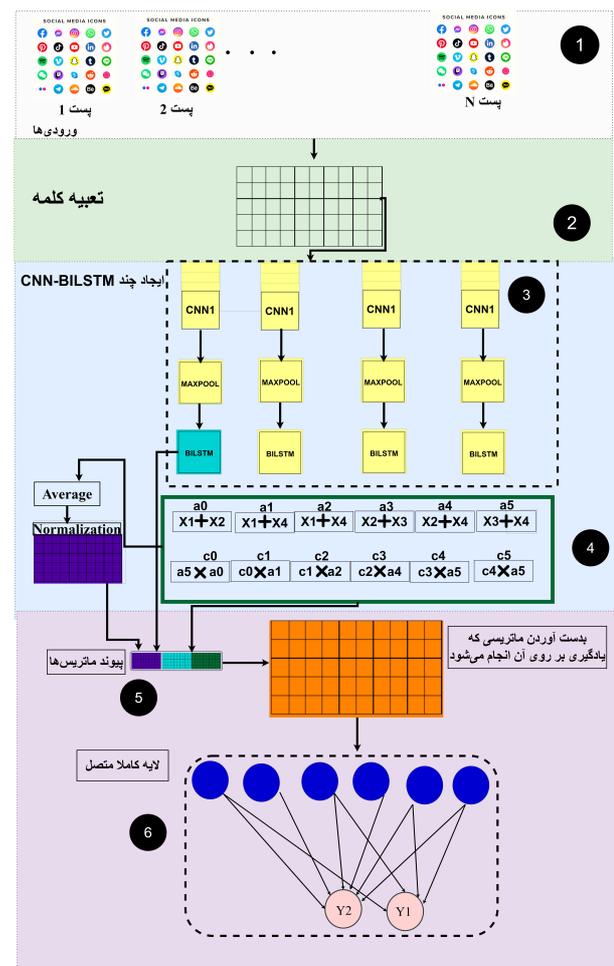
$$c_3 = c_2 \times a_3$$

$$c_4 = c_3 \times a_4$$

$$c_5 = c_4 \times a_5$$

بر اساس رابطه (4)، میانگین  $c_0$  تا  $c_5$  گرفته می‌شود و در ماتریس  $\eta$  ذخیره می‌شود و سپس نرمال می‌شود، به نحوی که این نرمال‌سازی داده‌ها را در مرکز تقریباً صفر با انحراف استاندارد 1 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 128 گره و یک تابع فعال‌سازی رلیو<sup>1</sup> داده می‌شود و بعد از آن این 128 گره به یک لایه کاملاً متصل دیگر با 2 گره و با تابع فعال‌سازی بیشینه هموار<sup>2</sup> متصل می‌شود. فرآیند برای شبکه‌های عصبی پیچشی پشته شده در جدول (1) 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 (1)، پولینگ 1 برای شبکه‌های عصبی پیچشی پشته شده استفاده می‌شود در حالی که پولینگ 2 برای شبکه‌های عصبی پیچشی ادغام شده استفاده می‌شود. حافظه طولانی کوتاه مدت دو جهت‌دار 128 گره و مقدار رهاشدگی<sup>3</sup>  $0/20$  است. حداکثر تعداد کلمات و حداکثر طول دنباله‌ها به ترتیب 10000 و 500 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 فعال‌سازی تابع بیشینه هموار استفاده می‌کند، سایر فعال سازها از تابع رلیو استفاده می‌کنند. برای مدل ارائه شده، از تعبیه

از آن یک پولینگ گرفته می‌شود. در مرحله بعد، خروجی هر شبکه عصبی پشته شده به یک شبکه عصبی حافظه طولانی کوتاه مدت دو جهت‌دار وصل می‌شود، جایی که شبکه‌های عصبی حافظه طولانی کوتاه مدت دو جهت‌دار در نظر گرفته شده، دارای فرآیندهای یکسان هستند. در مرحله 4، اعمال جمع ماتریس برای هر جفت خروجی ماتریس به دست آمده از حاصل اتصال هر شبکه عصبی پیچشی پشته شده و شبکه عصبی حافظه طولانی کوتاه مدت دو جهت‌دار انجام می‌شود. اگر  $x_1$  تا  $x_4$  خروجی‌های به دست آمده از حاصل اتصال هر شبکه عصبی پیچشی پشته شده و شبکه عصبی حافظه طولانی کوتاه مدت دو جهت‌دار باشند، آنها بر اساس رابطه (2) دو به دو با یکدیگر جمع می‌شوند و بر اساس رابطه (3) به صورت حلقوی در یکدیگر ضرب می‌شوند.



شکل (1): مدل شبکه عصبی پشته‌ای حلقوی پیشنهادی

<sup>1</sup> Relu

<sup>2</sup> Softmax

<sup>3</sup> Dropout

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 الگوریتم (1)، تنظیمات اولیه مانند تعداد گره‌ها، لایه ورودی و ماتریس تعبیه کلمات (تعبیه کلمه از پیش آموزش دیده «بردارهای جهانی برای نمایش کلمه» 300 بعدی) تعریف می‌شوند. سپس لایه‌های  $f1$  تا  $f4$  نیز با استفاده از تابع ایجاد شده در خطوط 1 تا 11، در خطوط 15 تا 22 ایجاد می‌شوند. در خطوط 23 و 24 ساختار حلقوی به شبکه عصبی ایجاد شده، اعمال می‌شود. در عمل، در خط 23 بصورت دو به دو لایه‌های  $f1$  تا  $f4$  با یکدیگر جمع می‌شوند و سپس حاصل جمع آنها بصورت حلقوی در خط 24 به صورت 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.

#### 4. نتایج تجربی

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

##### 1.4. مجموعه داده

برای تمامی ارزیابی‌ها، مجموعه داده شناخته شده PHEME و لیار را در نظر گرفته‌ایم. این مجموعه داده دارای 5802 توییت است که 3830 و 1972 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 [41] است. این مجموعه داده شش دسته «نادرست»، «به سختی درست»، «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 12836 نمونه است.

کلمه از پیش آموزش دیده «بردارهای جهانی برای نمایش کلمه» 300 بعدی استفاده شده است. برای بهینه‌سازی، از بهینه‌ساز معروف آدم<sup>1</sup> استفاده شده است. برای پیاده‌سازی مدل‌های یادگیری ماشین و یادگیری عمیق، به ترتیب از سائیکیت‌لرن<sup>2</sup> [39] و کراس<sup>3</sup> [40] استفاده کرده‌ایم. پیاده‌سازی‌ها در زبان برنامه‌نویسی پایتون و محیط گوگل کولب<sup>4</sup> انجام شده است.

جدول (1): مقادیر فرارامترها برای CNN های پشته شده

موارد	شبکه عصبی پشته‌ای 1	شبکه عصبی پشته‌ای 2	شبکه عصبی پشته‌ای 3	شبکه عصبی پشته‌ای 4
فیلتر	128	256	64	32
هسته	3, 5, 7, 9	2, 4, 6, 8	2, 4, 6, 8	3, 5, 7, 9
پولینگ 1	3, 5, 7, 9	2, 4, 6, 8	2, 4, 6, 8	3, 5, 7, 9
پولینگ 2	96	128	20	10

برای درک بهتر از مدل پیشنهادی، شبه کد شبکه عصبی حلقوی در ادامه آورده شده است. الگوریتم (1)، 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 مختلف مانند اندازه هسته، تعداد فیلترها و اندازه پولینگ در خط 1 تا 11 از الگوریتم (1) 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

<sup>1</sup> ADAM

<sup>2</sup> Skit-Learn

<sup>3</sup> Keras

<sup>4</sup> Google Colab

### الگوریتم (1): 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)
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')
33. return model
```

## 2.4. معیارهای ارزیابی

شده‌اند. همچنین TN و FP به تعداد نمونه‌هایی اشاره می‌کنند که به طور صحیح و اشتباه به عنوان شایعه برچسب‌گذاری شده‌اند. برای کلاس شایعه / غیرشایعه، بازیابی<sup>1</sup> به درصدی از کل پیش‌بینی‌ها اشاره دارد که به درستی طبقه‌بندی شده‌اند، در حالی که صحت<sup>2</sup>، تعداد کل پیش‌بینی‌های انجام‌شده برای کلاس

برای ارزیابی مدل‌ها از معیارهای استاندارد در روابط (6) تا (11) استفاده شده است. در این روابط، TP نشان‌دهنده مثبت واقعی، TN نشان‌دهنده منفی واقعی، FP نشان‌دهنده مثبت کاذب و FN بیانگر منفی کاذب است. FN به تعداد نمونه‌هایی اشاره می‌کند که به طور صحیح و اشتباه به عنوان غیرشایعه برچسب‌گذاری

<sup>1</sup> Recall

<sup>2</sup> Precision

ارزیابی برای مجموعه داده‌های PHEME و لیار است که شامل دو دسته صفر و یک است. هر سطر از نتایج بدست آمده از جدول‌های (2) و (3) تحت مجموعه داده‌های شرح داده شده، یا حاصل پیاده‌سازی طبقه‌بندی‌های 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 (2) 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 (2) نشان می‌دهد که دقت مدل شبکه عصبی حلقوی 84/31 است در حالی که ماشین بردار پشتیبان مبتنی بر سری زمانی 83/53 است. در مدل پیشنهادی، پیش‌بینی‌های صحیح (بازیابی) توییت‌های غیرشایعه 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 (2) نشان می‌دهد که مدل ترکیبی، شبکه عصبی حافظه طولانی کوتاه مدت دو جهته و شبکه عصبی پیچشی بازگشتی به نتایج بسیار ضعیفی دست یافته‌اند. علاوه بر جدول (2)، جدول (3) نتایج مدل پیشنهادی شبکه عصبی حلقوی (CMSCB) را با مدل‌های پایه مرز علم تحت مجموعه داده لیار مقایسه می‌کند. با توجه به نتایج به دست آمده از جدول (3)، می‌توان چند نتیجه زیر را بیان کرد:

- بین مدل‌های پایه، شبکه عصبی حافظه طولانی کوتاه مدت دو جهته و مدل ترکیبی عملکرد بسیار ضعیفی دارند.

مورد نظر است. F1 میانگین وزنی هارمونیک صحت و بازیابی است. دقت<sup>3</sup> نسبت تعداد کل پیش‌بینی‌های صحیح است. برای انجام تمام آزمایش‌ها، ما از اعتبارسنجی متقاطع به ازای  $k=5$  استفاده کرده‌ایم، به نحوی که 80 درصد داده‌ها برای اعتبارسنجی متقاطع و 20 درصد از آنها برای آزمایش استفاده می‌شوند.

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

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

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

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

$$F1 = 2 \times \frac{\text{بازیابی} \times \text{صحت}}{\text{بازیابی} + \text{صحت}} \quad (10)$$

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

### 3.4. 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 به دست آمده تشخیص شایعه هر مدل تحت مجموعه داده PHEME در جدول (2) نشان داده شده است، در حالی که نتایج به دست آمده هر مدل تحت مجموعه داده لیار در جدول (3) نشان داده شده است. این جداول دو نوع نتایج را نشان می‌دهد: نتایج تحت داده‌های آزمایش و نتایج تحت اعتبارسنجی متقاطع که به ازای  $k=5$  انجام شده است. هر دو نوع نتایج شامل چهار معیار

<sup>3</sup> Accuracy

جدول (2): مقایسه نتایج CMSCB با مدل‌های پایه

Data PHEME	نتایج تحت اعتبارسنجی متقابل به‌ازای k=5					نتایج تحت داده‌های آزمایش				
	مدل	کلاس	دقت	صحت	بازیابی	F1	دقت	صحت	بازیابی	F1
RNN [17]	0		45,01	30,07	62,86	40,67	48,87	48,95	52,58	50,70
	1			70,16	37,38	48,77		48,78	45,17	46,91
LSTM [20]	0		84,49	76,45	73,42	73,95	82,24	88,79	73,79	80,60
	1			88,71	89,23	88,94		77,58	90,69	83,62
CNN [42]	0		82,91	77,17	70,73	73,27	78,16	92,36	61,40	73,76
	1			85,53	89,27	87,33		71,09	94,92	81,29
BILSTM [18]	0		29,99	29,99	100	46,14	50	50	100	66,67
	1			0	0	0		0	0	0
CSI <sup>1</sup> [19]	0		29,99	29,99	100	46,14	50	50	100	66,67
	1			0	0	0		0	0	0
RCNN <sup>2</sup> [43]	0		78,66	0	0	0	50	0	0	0
	1			78,66	100	88,05		50	50	100
AdaBoost	0		81,16	78,54	61,40	68,90	71,79	91,02	48,35	63,16
	1			82,13	91,33	86,48		64,84	95,23	77,15
SVM-TS [44] <sup>3</sup>	0		85,46	76,41	74,57	75,46	83,53	91,47	73,96	81,79
	1			89,22	90,12	89,67		78,14	93,10	84,97
Random-forest <sup>4</sup>	0		86,86	85,80	67,39	75,42	81,03	94,33	66,03	77,68
	1			87,22	95,19	91,03		73,87	96,03	83,50
SVM <sup>5</sup>	0		86,28	84,78	66,16	74,29	80,94	93,25	66,72	77,78
	1			86,76	94,89	96,64		74,09	95,17	83,32
KNN <sup>6</sup>	0		80,31	83,27	42,96	56,61	69,22	91,44	42,41	57,95
	1			79,78	96,31	87,26		62,51	96,03	75,73
SGD <sup>7</sup>	0		85,48	77,82	72,56	74,94	83,01	91,36	72,93	81,11
	1			88,61	91,02	89,77		77,47	93,10	84,57
MSCB بدون حلقوی	0		84,68	72,72	75,51	75,51	82,24	90,47	72,06	80,23
	1			91,09	86,86	88,81		76,79	92,41	83,88
CMSCB مدل پیشنهادی	0		85,05	78,42	70,62	73,79	84,31	89,64	77,58	83,17
	1			88,02	91,23	89,50		80,24	91,03	85,29

<sup>1</sup> مدل ترکیبی

<sup>2</sup> شبکه عصبی پیچشی بازگشتی

<sup>3</sup> ماشین بردار پشتیبان مبتنی بر سری زمانی

<sup>4</sup> جنگل تصادفی

<sup>5</sup> ماشین بردار پشتیبان

<sup>6</sup> K نزدیک‌ترین همسایه

<sup>7</sup> کاهش گرادیان تصادفی

جدول (3): 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
RNN [17]	0		27,73	49,18	35,44		52,70	53,20	52,95
	1	50,47	72,39	50,98	59,80	52,73	52,76	53,26	52,51
LSTM [20]	0		32,26	31	31,59		61,96	40,46	48,96
	1	70,19	80,50	81,39	80,94	57,81	55,80	75,15	64,04
CNN [42]	0		46,85	32,01	38,06		68,82	29,84	41,63
	1	71,13	76,82	86,06	81,17	58,16	55,21	86,48	67,39
BILSTM [18]	0		22,23	100	36,37		50	100	66,67
	1	22,23	0	0	0	50	0	0	0
CSI [19]	0		33,96	100	50,68		50	100	66,67
	1	33,96	0	0	0	50	0	0	0
RCNN [43]	0		0	0	0		0	0	0
	1	77,77	87,5	100	87,5	50	50	100	66,67
AdaBoost	0		58,06	16,60	25,79		84,88	14,92	25,38
	1	73,48	74,82	95,36	83,85	56,13	53,36	97,34	68,93
SVM-TS [44]	0		40,81	39,27	40,01		66,22	35,23	45,99
	1	67,27	76,96	78,04	77,49	58,63	55,88	82,03	66,47
Random-forest	0		74,36	11,90	20,50		88,88	7,5	13,83
	1	74,37	74,38	98,40	84,72	53,28	51,71	99,06	67,95
SVM	0		73,47	11,39	19,69		95,23	7,81	14,44
	1	74,21	74,26	98,38	84,64	53,71	51,93	99,60	68,27
KNN	0		83,27	42,96	56,61		65,43	26,32	37,54
	1	67,46	79,78	96,31	87,26	56,21	53,88	86,09	66,28
SGD	0		42,39	37,53	39,77		67,52	32,65	44,02
	1	68,44	76,97	80,33	78,61	58,47	55,58	84,29	58,47
MSCB بدون حلقوی	0		40,07	35,67	36,57		75,90	21,40	33,39
	1	73,36	82,16	84,13	82,99	57,30	54,25	93,20	68,58
CMSCB مدل پیشنهادی	0		44,11	25,50	31,01		73,77	27,03	39,56
	1	75,57	80,89	89,88	85,08	58,71	55,33	90,39	68,64

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 تایید می‌کند. برای این فرضیه، شبکه عصبی پشته‌ای حلقوی از نظر

- با توجه به جدول (3)، می‌توان گفت که مدل شبکه عصبی حلقوی یک روش قوی و کارآمد برای تشخیص شایعه است.

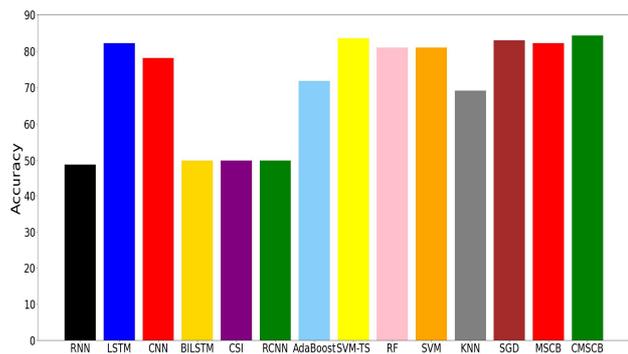
- با توجه جداول (2) و (3)، اگرچه مدل پیشنهادی در هر دو مجموعه داده نسبت به مدل‌های پایه برتری دارد اما واضح است که دقت مدل تحت مجموعه داده

## 5. نتیجه گیری

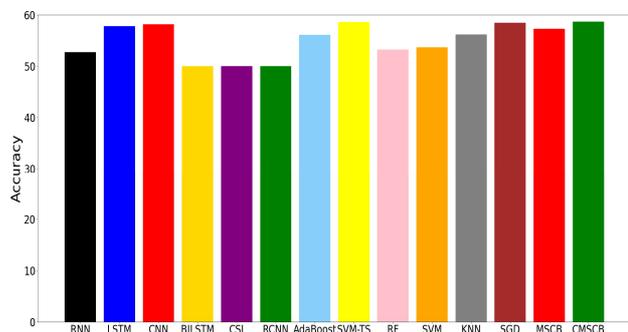
شبکه‌های اجتماعی به طور گسترده‌ای مورد استفاده قرار می‌گیرند و در زندگی روزانه افراد تاثیر به سزایی دارند. این شبکه‌ها مزایای زیادی نظیر سهولت در برقراری ارتباطات جهانی، تجارت الکترونیکی، افزایش آگاهی و سهولت در دسترسی به آخرین رویدادهای جهانی را دارند. اما یکی از عوامل منفی در شبکه‌های اجتماعی وجود شایعات است که می‌تواند منجر به آثار مخرب مالی و جانی شوند. به عنوان 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. به طور معمول 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 جهت 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 جهت 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 جهت اضافه شده و نشان داده شده است که ساختار حلقوی می‌تواند در بهبود کارایی مدل برای طبقه‌بندی متن موثر 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.

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

دقت تحت دو مجموعه داده از مدل‌های پایه بهتر عمل کرده است که این نتیجه خوب بیانگر تاثیر در نظر گرفتن ویژگی‌های محلی و سراسری استخراج شده و ساختار حلقوی در نظر گرفته شده است. ما همچنین برای ارزیابی تاثیر حالت حلقوی پیشنهادی، مدل خود را بدون حالت حلقوی آزموده‌ایم. دقت این مدل بدون حالت حلقوی برای مجموعه داده‌های PHEME و لیار به ترتیب برابر با 82/24 و 57/31 است، در حالی که دقت مدل با ساختار حلقوی برابر با 84/31 و 58/71 درصد است که نشان می‌دهد حالت حلقوی برای تشخیص شایعه تاثیر مثبتی دارد. شکل‌های (2) و (3) به ترتیب کارایی مدل‌های متفاوت تحت مجموعه داده‌های PHEME و 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.



شکل (2): مقایسه مدل شبکه عصبی پشته‌ای حلقوی با دیگر مدل‌های پایه از لحاظ معیار دقت تحت مجموعه داده PHEME



شکل (3): مقایسه مدل شبکه عصبی پشته‌ای حلقوی با دیگر مدل‌های پایه از لحاظ دقت تحت مجموعه داده لیار

- [1] 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," *Qual. Quant.*, pp. 1-32, 2021, doi: 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," *J. Acad. Market. Sci.*, vol. 41, no. 5, pp. 547-566, 2013.
- [3] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques," *Inf. Sci.*, vol. 497, pp. 38-55, 2019, doi: 10.1007/s11747-013-0326-9.
- [4] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdiscipl. Rev.: Data Mining Knowl. Discovery*, vol. 8, no. 4, p. e1249, 2018, doi: 10.1002/widm.1249.
- [5] T. Lehinevych and H. Andii, "Analysis of Deep Metric Learning Approaches," in *Proc. IEEE Int. Conf. Adv. Trends Inf. Theory (ATIT)*, 2019, pp. 369-372, doi: 10.1109/ATIT49449.2019.9030440.
- [6] C. Kuhnert, M. Baruthio, T. Bernard, C. Steinmetz, and J.-M. Weber, "Cloud-based event detection platform for water distribution networks using machine-learning algorithms," *Procedia Eng.*, vol. 119, pp. 901-907, 2015, doi: 10.1016/j.proeng.2015.08.963.
- [7] W.F. Lamberti, "An overview of explainable and interpretable AI," in *AI Assurance*, 2023, pp. 55-123, doi: 10.1016/B978-0-32-391919-7.00015-9.
- [8] T. Dietterich, "Overfitting and undercomputing in machine learning," *ACM Comput. Surv.*, vol. 27, no. 3, pp. 326-327, 1995, doi: 10.1145/212094.212114.
- [9] Y. Tan, "An improved KNN text classification algorithm based on K-medoids and rough set," in *Proc. 10th Int. Conf. Intell. Human-Mach. Syst. Cybern. (IHMSC)*, 2018, vol. 1, pp. 109-113, doi: 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," *Int. J. Autom. Comput.*, vol. 15, pp. 290-298, 2018, doi: 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 Warehous. Knowl. Discov. (DaWaK)*, 2014, pp. 288-299, doi: 10.1007/978-3-319-10160-6\_26.
- [12] S. Wang, L. Jiang, and C. Li, "Adapting naive Bayes tree for text classification," *Knowl. Inf. Syst.*, vol. 44, pp. 77-89, 2015, doi: 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 Comput. J.*, vol. 11, no. 1, pp. 120-139, 2022, doi: 10.22052/SCJ.2023.243182.1010 [In Persian].
- [14] N. Majma and S. Bashtin, "Detection of plagiarism in scientific texts based on text blocking and cosine similarity criteria," *Soft Comput. J.*, vol. 11, no. 1, pp. 60-71, 2022, doi: 10.22052/SCJ.2022.243384.1034 [In Persian].
- [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 Comput. J.*, vol. 11, no. 1, pp. 88-105, 2022, doi: 10.22052/SCJ.2023.246553.1077 [In Persian].
- [16] R. Behzadidoost, F. Mahan, and H. Izadkhah, "Granular computing-based deep learning for text classification," *Inf. Sci.*, vol. 652, p. 119746, 2024, doi: 10.1016/j.ins.2023.119746.
- [17] J. Ma et al., "Detecting rumors from microblogs with recurrent neural networks," in *Proc. 25th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2016, pp. 3818-3824.
- [18] A. Kumar, V. Singh, T. Ali, S. Pal, and J. Singh, "Empirical evaluation of shallow and deep classifiers for rumor detection," *Adv. Comput. Intell. Syst.*, pp. 239-252, 2020, doi: 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 *Proc. 2017 ACM Conf. Inf. Knowl. Manage.*, 2017, pp. 797-806, doi: 10.1145/3132847.3132877.
- [20] A.S. Torshizi and A. Ghazikhani, "Automatic Twitter rumor detection based on LSTM classifier," *High-Perform. Comput. Big Data Anal.*, pp. 291-300, 2019, doi: 10.1007/978-3-030-33495-6\_22.
- [21] T.T. Nguyen et al., "Judo: Just-in-time rumour detection in streaming social platforms," *Inf. Sci.*, vol. 570, pp. 70-93, 2021, doi: 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,"

- Knowl.-Based Syst., vol. 207, p. 106391, 2020, doi: 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 Syst. Appl.*, vol. 158, p. 113595, 2020, doi: 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, doi: 10.1016/j.neucom.2021.06.062.
- [25] H. Zhou et al., "MDMN: Multi-task and Domain Adaptation based Multi-modal Network for early rumor detection," *Expert Syst. Appl.*, vol. 195, p. 116517, 2022, doi: 10.1016/j.eswa.2022.116517.
- [26] B. Liu et al., "Nowhere to hide: Online rumor detection based on retweeting graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 4887-4898, 2022, doi: 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," *Inf. Sci.*, vol. 592, pp. 402-416, 2022, doi: 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," *Knowl.-Based Syst.*, vol. 240, p. 108085, 2022, doi: 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," *Inf. Sci.*, vol. 642, p. 119083, 2023, doi: 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," *J. Big Data*, vol. 10, no. 1, p. 45, 2023, doi: 10.1186/s40537-023-00725-4.
- [31] Z. Guo et al., "Mixed graph neural network-based fake news detection for sustainable vehicular social networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15486-15498, 2022, doi: 10.1109/TITS.2022.3185013.
- [32] H. Han, Z. Ke, X. Nie, L. Dai, and W. Slamun, "Multimodal Fusion with Dual-Attention Based on Textual Double-Embedding Networks for Rumor Detection," *Appl. Sci.*, vol. 13, no. 8, p. 4886, 2023, doi: 10.3390/app13084886.
- [33] H. Yuan, Y. Chen, X. Hu, and S. Ji, "Interpreting deep models for text analysis via optimization and regularization methods," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 01, pp. 5717-5724, doi: 10.1609/aaai.v33i01.33015717.
- [34] M. Cassel and F. Lima, "Evaluating one-hot encoding finite state machines for SEU reliability in SRAM-based FPGAs," in *Proc. 12th IEEE Int. On-Line Test. Symp. (IOLTS)*, 2006, p. 6, doi: 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 *Proc. Conf. Empirical Methods Nat. Lang. Process. (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 *Proc. 2020 IEEE Int. Conf. Softw. Test. Verif. Valid. Worksh. (ICSTW)*, 2020, pp. 465-476, doi: 10.1109/ICSTW50294.2020.00083.
- [38] R.J. Williams and D. Zipser, "Gradient-based learning algorithms for recurrent," *Backpropag. Theory Arch. Appl.*, vol. 433, p. 17, 1995.
- [39] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825-2830, 2011.
- [40] N. Ketkar, "Introduction to keras," in *Deep learning with python: a hands-on introduction*, 2017, pp. 97-111, doi: 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, doi: 10.18653/v1/P17-2067.
- [42] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014, doi: 10.3115/v1/D14-1181.
- [43] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. AAAI Conf. Artif. Intell.*, 2015, vol. 29, no. 1.
- [44] 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 *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 1751-1754, doi: 10.1142/9789813223615\_0006.