
بسم الله الرحمن الرحيم

دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی ارشد مهندسی هوش مصنوعی

کاوش مقیاس‌پذیر نظرات کاربران با استفاده از محاسبات خوشه‌ای

نگارنده

مجتبی اسداللهی

استاد راهنما

دکتر هدی مشایخی

شهریور ۹۷

شماره: ۵۸/ق/د
تاریخ: ۹۷/۷/۱

باسمه‌تعالی

مدیریت تحصیلات تکمیلی

فرم شماره (۳) صورتجلسه نهایی دفاع از پایان نامه دوره کارشناسی ارشد

با نام و یاد خداوند متعال، ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد آقای **مجتبی اسداللهی** با شماره دانشجویی **۹۴۰۱۹۴۴** رشته مهندسی کامپیوتر گرایش هوش مصنوعی تحت عنوان **کاوش مقیاس‌پذیر نظرات کاربران با استفاده از محاسبات خوشه‌ای** که در تاریخ **۱۳۹۷/۶/۱۹** با حضور هیأت محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح ذیل اعلام می گردد:

قبول (با درجه: خوب .....) ☒ **مردود** ☐

نوع تحقیق: نظری ☒ **عملی** ☐

عضو هیأت داوران	نام ونام خانوادگی	مرتبهٔ علمی	امضاء
۱ـ استادراهنمای اول	دکتر هدی مشایخی	استادیار	
۲- استادراهنمای دوم			
۳- استاد مشاور			
۴- نماینده تحصیلات تکمیلی	دکتر منصور فاتح	استادیار	
۵- استاد ممتحن اول	دکتر فاطمه جعفری نژاد	استادیار	
۶ـ استاد ممتحن دوم	دکتر بهروز مینایی	دانشیار	

نام و نام خانوادگی رئیس دانشکده:

تاریخ و امضاء و مهر دانشکده:

تبصره: در صورتی که کسی مردود شود حداکثر یکبار دیگر (در مدت مجاز تحصیل) می تواند از پایان نامه خود دفاع نماید (دفاع مجدد نباید زودتر از ۴ ماه برگزار شود).

سپاسگزاری

با سپاس از پروردگار،

از استاد گرامیم سرکار خانم دکتر مهدی مشایخی بسیار سپاسگزارم چرا که بدون راهنمایی‌های ایشان تا مین این پایان نامه بسیار مشکل می نمود. همچنین از تمامی اساتید گرامی که تا این مقطع درسی مرا راهنمود فرمودند مراتب سپاسگزاری را دارم.

و در انتها با سپاس از دو وجود مقدس:

آنان که ناتوان شدند تا ما به توانایی برسیم...

آنان که موهایشان سپید شد تا ما روسفید شویم...

و عاشقانه سوختند تا گرما بخش وجود ما و روشنگر راهمان باشند...

پدرانمان

مادرانمان

تعهد نامه

اینجانب **مجتبی اسداللهی** دانشجوی دوره کارشناسی ارشد رشته **هوش مصنوعی و رباتیک** دانشکده کامپیوتر و فناوری اطلاعات دانشگاه صنعتی شاهرود نویسنده پایان‌نامه **کاوش مقیاس‌پذیر نظرات کاربران با استفاده از محاسبات خوشه‌ای** تحت راهنمائی **خانم دکتر هدی مشایخی** متعهد می‌شوم.

- تحقیقات در این پایان‌نامه توسط اینجانب انجام‌شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورداستفاده استناد شده است.
- مطالب مندرج در پایان‌نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود است و مقالات مستخرج با نام «دانشگاه صنعتی شاهرود» و یا «Shahrood University of Technology» به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان‌نامه تأثیرگذار بوده‌اند در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که از موجود زنده (یا بافتهای آن‌ها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری، ضوابط و اصول اخلاق انسانی رعایت شده است.

امضای دانشجو

تاریخ

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود است. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در پایان‌نامه بدون ذکر مرجع مجاز نمی‌باشد.

چکیده:

در دنیای امروز، گسترش رسانه‌ها و شبکه‌های اجتماعی و همچنین افزایش تارنماهای تحلیلی تجاری، خبری، باعث استقبال و حضور گسترده مردم در این مجموعه‌ها شده است. این امر موجب انتشار و اشتراک داده‌ها و نظرات بسیار زیادی می‌شود. با توجه به این حجم انبوه از داده، با تحلیل این داده‌ها، می‌توان اطلاعات بسیار ارزشمندی استخراج نمود. اما در کنار این مزیت، پردازش این حجم از اطلاعات خود یک معضل را پدید می‌آورد. سرعت پردازش روش‌های گذشته و سنتی در مقابل رشد سریع جریان داده‌های ورودی کند می‌باشد. برای حل این مشکل، می‌توان از روش‌های مقیاس‌پذیر و همچنین ابزارهای نوین در حوزه پردازش کلان داده، برای تسریع روند پردازش استفاده نمود. پژوهش‌هایی در این حوزه انجام شده است اما همچنان این حوزه با توجه به پتانسیل بالای خود، نیازمند پژوهش‌های بیشتری می‌باشد. در پژوهش‌های گذشته علاوه بر کندی سرعت پردازش مشکلاتی از جمله پایین بودن دقت، استفاده از ساختار پیچیده روش پیشنهادی و وابستگی به دامنه موضوع وجود دارد.

اسپارک چارچوبی نسبتا نوین برای پردازش کلان‌داده‌ها می‌باشد که پردازش را در حافظه انجام می‌دهد و این موضوع باعث افزایش سرعت پردازش الگوریتم می‌شود. در این پژوهش از رده‌بند ماشین بردار پشتیبان(SVM) و ابزار اسپارک به صورت همزمان جهت رده‌بندی نظرات کاربران استفاده شده است. روش پیشنهادی در کنار حفظ دقت مطلوب، وابسته به دامنه خاصی نمی‌باشد و در پیاده‌سازی نیز از پیچیدگی به‌دور است. برای بهره‌مندی از توانایی اسپارک در پردازش توزیع‌شده از رده‌بند SVM موازی به صورت آبشاری، آبشاری بهبودیافته و گروهی و رأی‌گیری بین چند SVM بهره گرفته شده است. برای مقایسه کارایی روش‌های فوق، SVM استاندارد نیز پیاده‌سازی شده است. برای ارزیابی از مجموعه-داده IMDB استفاده شده است. این مجموعه‌داده شامل نظرات مربوط به فیلم است و به زبان انگلیسی می‌باشد. در نتایج بدست آمده برای رده‌بندی، بهترین دقت حاصل شده ۸۵,۹۴۶٪ است. همچنین زمان سپری شده برای اجرای الگوریتم SVM استاندارد در محیط اسپارک در مقابل پژوهش دیگری که زمان

را گزارش داده است، بیش از ۸ مرتبه کاهش یافته است. زمان برای SVMهای موازی در مقابل SVM استاندارد نیز بهبود قابل توجهی داشته است. در بهترین حالت زمان آموزش روش پیشنهادی نسبت به زمان آموزش SVM استاندارد حدود ۶۰ مرتبه کاهش یافته است.

کلمات کلیدی: نظرکاوی، ماشین بردار پشتیبان، پردازش موازی، اسپارک

فهرست مطالب

فهرست جدول‌ها

فهرست شکل‌ها

فهرست واژگان و اختصارات

Accuracy	ACC	دقت
Conventional Neural Network	CNN	شبکه عصبی کانولوشن
International Movie Database	IMDB	پایگاه‌داده بین‌المللی فیلم
Karash-Kuhn-Tucker	KKT	کاراش-کوهن-تاکر
Logistic Regression	LR	رگرسیون منطقی
Maximum Entropy	ME	بیشینه آنتروپی
Naïve Bayes	NB	بیز ساده
Natural Language Proccesing	NLP	پردازش زبان طبیعی
Opinion Mining	OM	نظرکاوی
Part of Speech	POS	اقسام جمله
Quadratic Programming	QP	برنامه‌ریزی مرتبه دوم
Radial Basic Function	RBF	تابع پایه شعاعی
Resilient Distributed Dataset	RDD	مجموعه‌داده توزیع‌شده انعطاف‌پذیر
Sensitivity (True Positive Rate)	TPR	نرخ پیش‌بینی مثبت صحیح
Sentiment Analysis	SA	تحلیل احساسات
Sequential Minimal Optimization	SMO	بهینه‌سازی متوالی کمینه
Stochastic Gradient Descent	SGD	گرادیان نزولی تصادفی
Support Vector	SV	بردار پشتیبان
Support Vector Machine	SVM	ماشین بردار پشتیبان
Specificity (True Negative Rate)	TNR	نرخ پیش‌بینی منفی صحیح

فصل اول

مقدمه

۱ـ۱ مقدمه

امروزه با گسترش اینترنت، روزانه اطلاعات زیادی در این فضا ایجاد می‌شود که پردازش این حجم انبوه از اطلاعات نیازمند ابزارها و روش‌هایی می‌باشد که در پردازش، سریع و در عملکرد، دقیق هستند. با توسعه طراحی و معماری وب ۲/۰ تعامل کاربران با فضای اینترنت تسهیل شد و باعث ظهور شبکه‌های اجتماعی[۱]، وب‌نوشت‌های کوچک[۲]، انجمن‌ها[۳] و سایت‌های تحلیلی مختلفی گردید. روزانه نظرات و دیدگاه‌های بسیاری در موضوعات مختلف، توسط کاربران در این محیط‌ها ارسال می‌شود. این نظرات می‌تواند حاوی اطلاعات ارزشمندی برای دولت‌ها، سازمان‌ها، شرکت‌ها و مردم باشد. بعنوان نمونه از گذشته تاکنون، افراد جهت خرید یک محصول یا سرویس، از نظرات دیگران برای تصمیم‌گیری استفاده می‌نمایند. لذا با بررسی نظرات دیگران می‌تواند تصمیم بهتری در این مورد اخذ نمایند. سازمان‌ها و شرکت‌ها نیز با بررسی همین نظرات می‌توانند نقاط قوت و ضعف محصولات و خدمات خود را شناخته و با سلیقه مردم آشنا شوند و در نقشه توسعه خود از آن بهره ببرند. دولت‌ها نیز می‌توانند در زمینه‌های مختلف با بررسی نظرات مردم تفکر کلی جامعه را بدون استفاده از نظرسنجی رصد کنند و این موضوع در تصمیم‌گیری‌ها حائز اهمیت خواهد بود. همین موضوع باعث ایجاد شاخه‌ای جدید به نام نظرکاوی[۴] در متن‌کاوی[۵] شده است.

۱ـ۲ بیان مسئله و تعاریف

روزانه حجم عظیمی از نظرات کاربران در شبکه‌های اجتماعی مختلف، انجمن‌ها و وب‌نوشت‌های کوچک منتشر می‌شود و به همین دلیل بررسی انسانی آن عملا غیر ممکن است. با توجه به پتانسیل و اهمیت این موضوع، نظرکاوی به صورت خودکار موردنیاز می‌باشد. تاکنون پژوهش‌هایی در این زمینه انجام شده

[1] Social Networks
[2] Microblogs
[3] Forums
[4] Opinion Mining
[5] Text Mining

است، اما همچنان ارائه روش‌هایی با سرعت و کارایی بیشتر موردنیاز می‌باشد.

در ادامه نیز تعاریف نظر، نظرکاوی و همچنین سطوح نظرکاوی آورده شده است.

نظر: بر اساس فرهنگ لغت آکسفورد «دیدگاه یا قضاوت شکل گرفته در مورد یک چیز، که لزوما براساس واقعیت یا دانش نمی‌باشد[1] ». همچنین بینگ لیو این تعریف را ارائه داده است: «نظر، احساسات نظردهنده در مورد یک وجود[2](محصول یا شئ) یا ویژگی‌های یک وجود می‌باشد»[۱].

جمله ذهنی[3]: این نوع جملات شامل نظر و احساس نظردهنده در مورد وجود و یا ویژگی‌های آن می‌باشد. مثال: «باتری گوشی‌های برند سونی دارای طول عمر بیشتری نسبت به سایر برندها می‌باشد.»

جمله عینی[4]: این نوع جملات همراه با احساس نمی‌باشد و بیشتر شامل یک حقیقت یا جمله خبری می‌باشند. مثال: «گوشی جدید برند سونی امروز معرفی شد.»

نظرکاوی: نظرکاوی (و یا تجزیه و تحلیل احساس[5]) زیرشاخه‌ای از متن‌کاوی و بخشی از حوزه پردازش زبان طبیعی[6] می‌باشد که به تشخیص و استخراج اطلاعات ذهنی[7] از نظر می‌پردازد و قطبیت[8] نظر شامل مثبت و منفی، یا حالت سوم یعنی طبیعی (نرمال) را تعیین می‌نماید.

نظرکاوی در سه سطح مختلف انجام می‌شود که در ادامه به آن اشاره شده است:

سطح سند[9]: در این سطح به تجزیه و تحلیل کل سند و متن پرداخته می‌شود و قطبیت آن شامل مثبت، منفی و یا حالت طبیعی تعیین می‌شود. چالش مهم این قسمت تعیین جملات و قسمت‌های ذهنی متن می‌باشد. در این سطح معمولا قطبیت کل سند در مورد یک ویژگی بیان می‌شود و این یکی

[1] https://en.oxforddictionaries.com/definition/opinion
[2] Entity
[3] Subjectivity
[4] Objectivity
[5] Sentiment Analysis
[6] Natural Language Processing (NLP)
[7] Subjective Information
[8] Polarity
[9] Document level

از مشکلات آن می‌باشد.

سطح جمله[1]: در این سطح از پردازش، متن به تک تک جملات تقسیم‌بندی شده و قطبیت هر جمله به طور جداگانه تعیین می‌شود. در این سطح، در مرحله پیش پردازش جملات عینی یا ذهنی[2] تفکیک شده و پس از آن قطبیت جملات ذهنی تعیین می‌گردد. این سطح نسبت به سطح سند به طور جزئی‌تر به تعیین قطبیت می‌پردازد. شناسایی قطبیت در این سطح از دو روش تعیین می‌گردد: رویکرد نحوی دستوری[3] و رویکرد معنایی[4]. در رویکرد نحوی دستوری با استفاده از ساختار دستوری جمله مانند اسم و صفت و ... قطبیت جمله تعیین می‌گردد، اما در رویکرد معنایی با استفاده از تعداد تکرار کلمات با قطبیت مثبت و منفی در جمله، قطبیت کلی جمله مشخص می‌گردد.

سطح جنبه یا ویژگی[5]: در این سطح نیز ویژگی‌های وجود استخراج و همچنین کلمات و یا عبارات نظری[6] مربوط به هر ویژگی استخراج می‌شود و در نهایت قطبیت نظر نسبت به هر ویژگی تعیین می‌شود.

۱ـ۳ هدف پایان‌نامه

با توجه به حجم عظیم نظرات منتشر شده در فضای اینترنت، پردازش آن نیازمند ابزار و روشی با سرعت پردازش و دقت عملکرد بالا می‌باشد. روش‌های ارائه شده پیشین نیز سعی در بهبود سرعت و دقت نظرکاوی داشته‌اند اما همچنان از معضل زمان زیاد پردازش و کاهش دقت رنج می‌برند. در برخی از پژوهش‌ها به دلیل استفاده از روش‌های پیچیده و دارای محاسبات زیاد، به دقت بالایی دست یافته‌اند اما همین موضوع سبب زمان‌بر شدن روش‌پیشنهادی شده است. در برخی دیگر از پژوهش‌ها سرعت الگوریتم پیشنهادی بالاست و زمان کمی برای اجرا مصرف می‌نماید اما در زمینه دقت عملکرد، قابل

[1] Sentence Level
[2] Objective or Subjective Sentences
[3] Grammatical Syntactic Approach
[4] Semantic Approach
[5] Aspect or Feature Level
[6] Opinion Words or Phrases

قبول نمی‌باشند و در این مورد ضعف دارند. لذا با این تفاسیر، فقدان روشی با دقت مطلوب در کنار هزینه زمانی معقول و کم، محسوس است. با وجود این محدودیت‌ها، با بهره‌مندی هوشمندانه از امکانات در دسترس می‌توان بر این مشکلات تا حدودی غلبه کرد.

استفاده مناسب از ابزاهای پردازش کلان‌داده و ارائه روش‌های مقیاس‌پذیر[1] می‌تواند در هزینه زمانی روش پیشنهادی موثر باشد و باعث بهبود و کاهش زمان اجرای الگوریتم شود. همچنین استفاده از الگوریتم‌های مناسب حوزه موردنظر که دارای عملکرد دقیق می‌باشند و سازگار با پردازش موازی هستند نیز می‌تواند دقت روش پیشنهادی را تضمین نماید.

در این پژوهش روشی مقیاس‌پذیر ارائه شده است و از ابزار پردازش کلان‌داده اسپارک استفاده شده است. این امر باعث کاهش زمان آموزش و بهبود در سرعت تشخیص قطبیت نظر می‌شود. همچنین در این پژوهش با بهره‌گیری از الگوریتم و رده‌بند SVM دقت بدست آمده مطلوب و قابل قبول می‌باشد.

۱ـ۴ روش پیشنهادی

در این پژوهش، روشی مقیاس‌پذیر برای رده‌بندی نظرات کاربران ارائه شده است. به این منظور دو روش SVM مبتنی بر رأی‌گیری و SVM با پالایش بردارهای پشتیبان پیشنهاد شده و در محیط اسپارک پیاده‌سازی شده است. همچنین برخی از روش‌های موجود مانند SVM آبشاری، SVM آبشاری بهبودیافته و SVM گروهی که در محیطی خارج از اسپارک پیاده‌سازی شده‌اند و بعضا برای کاربردهایی خارج از حوزه نظرکاوی پیشنهاد شده بودند، در محیط مذکور پیاده سازی و برای رده بندی نظرات کاربران مورد استفاده قرار گرفت.

۱ـ۵ روش ارزیابی

برای ارزیابی نتایج مدل‌ها با یکدیگر و همچنین با سایر پژوهش‌ها از چهار معیار استفاده شده است.

[1] Scalable

اولین معیار، زمان آموزش و آزمون مدل رده‌بند می‌باشد که بر حسب ثانیه است. معیار دیگر نیز دقت[1]، تشخیص[2] و حساسیت[3] می‌باشد که در فصل چهارم به طور کامل معرفی شده‌اند.

۱ـ۶ ساختار پایان‌نامه

این پایان نامه در شش فصل تدوین گردیده است. در ادامه، فصل دوم شامل ادبیات موضوع می‌باشد و فصل سوم مربوط به پژوهش‌های گذشته می‌باشد. در فصل چهارم روش ارائه شده در این پایان‌نامه توضیح داده شده است. نتایج و ارزیابی روش موردنظر در فصل پنجم آورده شده است. فصل ششم نیز شامل پیشنهادات و نتیجه‌گیری می‌باشد.

[1] Accuracy
[2] Specificity
[3] Sensivity

فصل دوم

ادبیات موضوع

۲ـ۱ مراحل اولیه نظرکاوی

روند نظرکاوی شامل مراحل مختلفی می‌شود که از مراحل مقدماتی آن می‌تواند به مرحله جمع‌آوری داده و پیش پردازش متن اشاره کرد. در ادامه به توضیح بیشتر این مراحل می‌پردازیم.

۲ـ۱ـ۱ جمع‌آوری داده[1]

اولین مرحله و قدم برای نظرکاوی ایجاد مجموعه داده برای این عمل می‌باشد. دو روش برای این عمل وجود دارد:

- استفاده از رابط برنامه‌نویسی کاربردی[2] تارنما[3]هایی نظیر آمازون[4]، توییتر[5]، فیسبوک[6] برای جمع‌آوری داده، که بعنوان مثال Twitter4J API[7] برای توییتر می‌باشد و از دیگر APIهای شناخته شده برای توییتر می‌توان [۲-۴] را نام برد.
- ایجاد و استفاده از خزنده‌های وب[8] که در وبسایت‌ها به جمع‌آوری نظرات و داده می‌پردازد. الستون و ناجورک یک بررسی دقیق و قوی روی خزنده‌های وب انجام داده‌اند[۵].

هر کدام از روش‌های فوق دارای مزایا و معایبی می‌باشند، برای مثال رابط‌های برنامه‌نویسی کاربردی دارای محدودیت‌هایی از طرف میزبان در هنگام جمع‌آوری داده می‌باشند و در مقابل خزنده‌ها نیز دارای مشکلاتی در پیاده‌سازی می‌باشند. همچنین تعدادی از پژوهش‌ها از برخی مجموعه‌داده‌های معرفی شده و استاندارد استفاده می‌کنند که این موضوع باعث ارزیابی بهتر روش ارائه شده نسبت به سایر پژوهش-های پیشین می‌شود. کایجی گئو[9] و همکاران[۶] به طور خلاصه به مقایسه این دو روش پرداخته اند.

---

[1] Data Acquisition
[2] Application Programming Interface
[3] Website
[4] Amazon
[5] Twitter
[6] Facebook
[7] http://twitter4j.org/en/index.html
[8] Web Crawlers
[9] Kaijie Guo

۲ـ۱ـ۲ پیش‌پردازش متن

مرحله دیگر پیش‌پردازش متن می‌باشد که با استفاده از عمل‌های معمول پردازش زبان طبیعی و استفاده از تجزیه و تحلیل واژه‌ها انجام می‌شود. مهمترین تکنیک‌های معمول در ادامه اشاره شده است:

تقسیم‌بندی کلمات[۱]: به عمل جداسازی و تفکیک جملات یک سند به لیستی از کلمات مجزا گفته می‌شود. این تکنیک برای زبان‌هایی که کلمات با استفاده از فاصله جدا می‌شوند، به راحتی قابل پیاده-سازی است اما برای زبان‌هایی مانند چینی و ژاپنی که کلمات با فاصله از هم جدا نمی‌شوند با مشکلاتی روبرو است.

ریشه‌یابی[۲]: به فرآیند حذف پسوند و پیشوندهای کلمات و به طور کلی تبدیل آن‌ها به شکل ریشه و اصلی آن، ریشه‌یابی گفته می‌شود. بعنوان مثال کلماتی نظیر شخص، اشخاص، شخصیت، شخصی، همگی پس از ریشه‌یابی به کلمه شخص تبدیل می‌شوند. یکی از معروفترین ریشه‌یاب‌ها برای زبان انگلیسی Porter's stemmer می‌باشد [۷].

حذف ایست‌واژه‌ها[۳]: این تکنیک شامل حذف ایست‌واژه‌ها می‌باشد که جزء ساختار زبانی می‌باشند اما کمکی به تحلیل نظر نمی‌کنند. و، یا، با، تا، از، به، است و .... تعدادی از این نوع کلمات می‌باشند.

تقسیم‌بندی جمله[۴]: این تکنیک برای تفکیک جملات از یکدیگر استفاده می‌شود. این روش مشکلات مربوط به خود را دارد. یکی از راه‌های جداسازی جملات از یکدیگر استفاده از نقطه انتهای جمله می‌باشد، اما اعداد اعشاری و همچنین کلمات مخفف شده باعث ایجاد مشکل در این حالت می‌شوند.

برچسب‌گذاری اقسام جمله[۵]: این روش شامل برچسب‌گذاری نقش هر کلمه در جمله مانند اسم،

[1] Word Segmentation
[2] Stemming
[3] Stopword Removal
[4] Sentence Segmentation
[5] Part-of-Speech (POS) Tagging

فعل، صفت، قید، حرف اضافه و... می‌باشد. از این روش در پردازش‌های مراحل بعدی استفاده می‌شود. برای مثال می‌توان از آن به عنوان ویژگی ورودی در روش‌های مبتنی بر یادگیری ماشین[1] استفاده کرد.

پیش‌پردازش‌های دیگری نیز برای حالت‌های خاص انجام می‌گیرد. برای مثال هنگامی که از خزنده‌های وب برای جمع‌آوری داده استفاده می‌شود، مراحلی برای حذف لینک‌ها و همچنین موارد غیرمتنی (مانند عکس‌ها و...) انجام می‌شود و یا برای مطالب موجود در توییتر نیز پیش‌پردازش برای حذف هشتگ[2]ها، اشاره[3]ها، شکلک[4]ها، کلمات مختصرشده و خنده‌های نوشته شده می‌باشد.

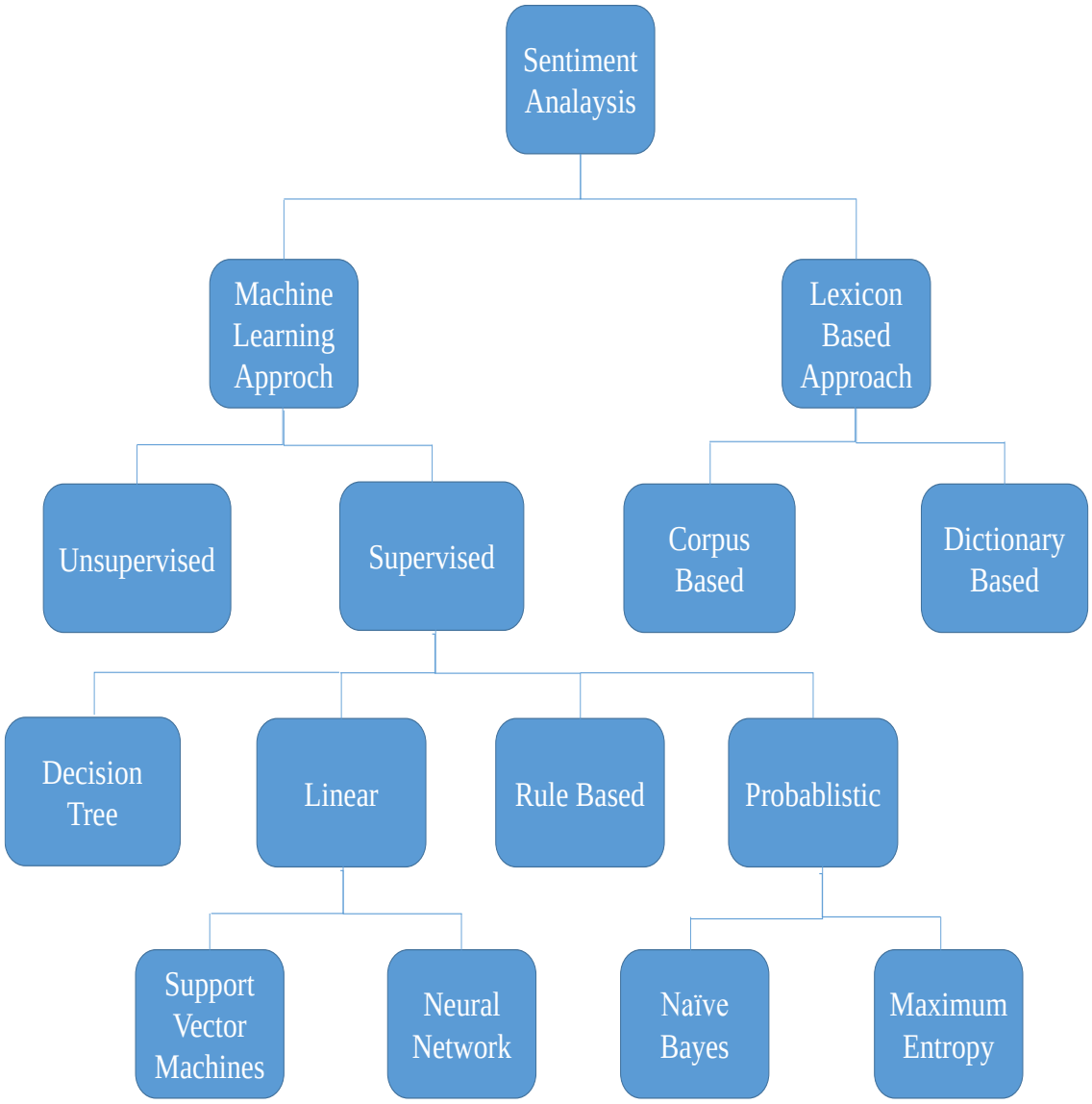
۲ـ۲ روش‌های نظرکاوی

روش‌های نظرکاوی دارای دو قسمت اصلی می‌باشند: روش‌های مبتنی بر یادگیری ماشین و روش‌های مبتنی بر واژه نامه[5] که در ادامه به آن‌ها پرداخته شده است. در شکل ۲-۱ شمایی کلی از روش‌های موجود نشان داده شده است.

۲ـ۲ـ۱ روش‌های مبتنی بر یادگیری ماشین

این دسته از روش‌ها، در بخش روش‌های با ناظر قرار می‌گیرند و در بعضی از منابع با این عنوان نیز مطرح می‌گردند. همان‌طور که انسان‌ها از تجربیات گذشته خود یاد می‌گیرند، رایانه‌ها نیز می‌توانند از داده‌ها آموزش ببینند که این داده‌ها مربوط به کاربردهای گذشته می‌باشد. در این روش نیاز به دو دسته از داده‌ها می‌باشد، مجموعه داده‌های آموزش که دارای برچسب کلاس می‌باشند و از آن برای ایجاد و آموزش رده‌بند[6] استفاده می‌شود و مجموعه داده‌های آموزش که با توجه به رده‌بند آموزش دیده، رده‌بندی روی آن‌ها انجام می‌شود. نکته‌ای که در این روش‌ها حائز اهمیت است، ویژگی‌هایی می‌باشد که از داده‌های مجموعه آموزش استخراج می‌شود تا برای یادگیری سیستم از آن‌ها استفاده شود.

[1] Machine Learning Based Methods
[2] Hashtag
[3] Mention
[4] Emoji
[5] Lexicon Based Methods
[6] Classifier



شکل ۲-۱: روش‌های موجود در تحلیل احساسات و نظرکاوی

در ادامه به تعدادی از مهمترین ویژگی‌ها مورد استفاده، اشاره می‌شود. پس از آن نیز تعدادی از روش‌های مبتنی بر یادگیری ماشین معرفی می‌شود.

وجود و تعداد تکرار کلمات[1]: این ویژگی به کلمات موجود در متن و یا کلمات n-gram متن توجه و به بررسی حضور و تعداد تکرار آن می‌پردازد. این ویژگی بسیار زیاد در زمینه نظرکاوی استفاده می‌شود و کاملا موثر واقع شده است. در پژوهشی، بو پانگ[2] و همکاران [۸] از n-gram برای بررسی نظرات

[1] Terms presence and frequency
[2] Bo Pang

مربوط به فیلم استفاده شده است و باعث افزایش بازده روش شده است. دیو و همکارانش نیز [۹] از bi-gram و tri-gram برای بررسی نظرات یک محصول استفاده نموده‌اند.

اطلاعات مربوط اقسام جمله[1]: این ویژگی نیز در تحلیل احساسات و نظرکاوی بسیار مهم می‌باشد. با استفاده از برچسب‌گذاری، به شناسایی نقش کلمات در جمله می‌پردازیم. کلماتی مانند صفت‌ها و قیدها تاثیر بسیاری زیادی در تجزیه و تحلیل احساسات صاحب‌نظر دارند.

نفی[2]: کلمات نفی (مانند نا، پیشوند ن و ...) نیز در نظرکاوی و تحلیل احساسات بسیار مهم است. زیرا این نوع از کلمات قطبیت جمله را معکوس می‌نمایند.

عبارات و کلمات نظری[3]: کلمات و عبارات نظری، کلماتی هستند که قطبیت جمله و احساس کاربر را بیان می‌کنند. اکثرا این کلمات شامل صفت‌ها و قیدها می‌شوند اما بعضی از کلمات در نقش اسم مانند آشغال و یا به صورت فعل مانند دوست داشتن و متنفر بودن در متن استفاده می‌شوند.

روش‌های یادگیری ماشین شامل روش‌های زیادی می‌باشد که از معروف‌ترین و کاربردی‌ترین آن‌ها بخصوص در نظرکاوی می‌توان به بیز ساده، ماشین بردار پشتیبان، بیشینه آنتروپی اشاره کرد. در ادامه به معرفی جزئی‌تر این برخی از این روش‌ها می‌پردازیم.

۲ـ۲ـ۱ـ۱ رده‌بندی کننده بیز ساده[4]:

رده‌بند بیز ساده، معمول‌ترین و ساده‌ترین روش مورد استفاده در رده‌بندی متن می‌باشد. این روش مبتنی بر محاسبه احتمال پیشین یک کلاس می‌باشد و بر مبنای توزیع کلمات در متن می‌باشد. این تکنیک به نقش و مکان کلمات در جمله توجهی نمی‌کند. در این روش به طور ساده و خلاصه، ویژگی-های مربوط به یک کلاس را درنظر گرفته و پس از آن با بررسی احتمال تعلق یک متن به همراه

[1] Part of speech information
[2] Negation
[3] Opinion Words and Phrases
[4] Naive Bayes Classifier (NB)

ویژگی‌هایش در هر کلاس، رده‌بندی را انجام می‌دهد.

۲ـ۲ـ۱ـ۲ ماشین بردار پشتیبان[1]:

این روش یکی از موثرترین و کارآترین روش‌ها برای رده‌بندی متن بوده و می‌باشد. ایده اصلی این روش، تعیین یک جداساز خطی برای تفکیک هرچه بهتر فضای جستجو می‌باشد. به طور خلاصه در این روش، هدف ایجاد یک ابرصفحه (بردار) می‌باشد که دارای بیشینه اطمینان برای تفکیک یک کلاس از سایر کلاس‌ها می‌باشد. داده‌های متنی برای این روش بسیار ایده‌آل می‌باشند و این موضوع به دلیل ماهیت پراکندگی (ویژگی‌های) متن می‌باشد که جداسازی را تقریبا تسهیل می‌نماید. در فصل چهارم، به توضیح بیشتر این روش که مدنظر ماست خواهیم پرداخت.

۲ـ۲ـ۱ـ۳ بیشینه آنتروپی[2]:

این روش نیز از روش‌های کارآمد در زمینه پردازش زبان طبیعی می‌باشد و در اکثر موارد بهتر از روش بیز ساده عمل می‌نماید. در این تکنیک با استفاده از کدگذاری، مجموعه ویژگی‌ها به یک بردار تبدیل می‌شود. سپس از این بردار کدگذاری شده برای تعیین وزن هر ویژگی استفاده می‌شود و در نهایت با توجه به وزن و اهمیت ویژگی‌ها برای کلاس‌ها، رده‌بندی صورت می‌گیرد.

۲ـ۲ـ۱ـ۴ درخت تصمیم[3]:

درخت تصمیم، یک تقسیم‌بندی سلسله مراتبی را از مجموعه داده‌های آموزش با توجه به محدودیت‌ها و شرایط خاصی انجام می‌دهد. این شرایط می‌تواند وجود یا عدم وجود یک یا چند کلمه باشد. این تجزیه و تقسیم به صورت بازگشتی انجام می‌شود تا زمانی که گره‌های برگ به حداقل تعداد مشخصی داده (رکورد) برسد.

[1] Support Vector Machine (SVM)
[2] Maximum Entropy (ME)
[3] Decision Tree

۲ـ۲ـ۱ـ۵ رده‌بندی کننده‌های مبتنی بر قاعده[1]:

در این روش فضای داده با تعدادی قوانین مدل می‌شود. در سمت چپ قوانین، ویژگی‌ها قرار دارند و در سمت راست آن‌ها، برچسب کلاس قرار دارد. قوانین با استفاده از معیارهای زیادی تولید می‌شود که مرحله ساخت و آموزش قوانین به همین معیارها بستگی دارد. دو نوع از مهمترین این معیارها، پشتیبان[2] و اطمینان[3] می‌باشد[۱۰].

۲ـ۲ـ۱ـ۶ شبکه عصبی مصنوعی[4]:

در این روش با استفاده از شبکه عصبی مصنوعی، رده‌بندی انجام می‌شود. شبکه عصبی مصنوعی از تعدادی نرون[5] تشکیل شده که هر نرون دارای یک وزن مشخص می‌باشد. این شبکه و نرون‌ها با استفاده از مجموعه داده آموزش، آموزش می‌بینند و وزن نرون‌ها تعیین می‌شود. پس از آن با اعمال داده‌های آموزش، شبکه اقدام به رده‌بندی داده‌ها می‌نماید.

۲ـ۲ـ۲ روش‌های مبتنی بر واژه‌نامه

در بسیاری از موارد ایجاد یک مجموعه داده برچسب‌گذاری شده برای آموزش مشکل می‌باشد و در مقابل ایجاد یه مجموعه داده بدون برچسب نسبتا آسان‌تر می‌باشد. بنابراین برای غلبه بر این مشکل روش‌های مبتنی بر واژه‌نامه ایجاد و مورد استفاده قرار گرفتند. در این روش‌ها با استفاده از کلمات و عبارات نظری موجود در متن به تعیین قطبیت هر جمله و متن پرداخته می‌شود که نیازی به داشتن مجموعه داده برچسب‌گذاری شده پیشین نمی‌باشد. مزیت این روش‌ها همان‌طور که اشاره شد عدم نیاز به مجموعه آموزشی می‌باشد و همچنین یکی از مهمترین معایب آن مشکل در ایجاد واژه‌نامه می‌باشد که یا باید به صورت انسانی باشد که بسیار وقت‌گیر و هزینه‌بر می‌باشد و یا باید به صورت اتوماتیک و با استفاده از روش‌های مختلف انجام شود که آن‌ها نیز مشکلات مربوط به خود را دارا می‌باشند. روش‌های

[1] Rule Based Classifier
[2] Support
[3] Confidence
[4] Artificial Neural Network (ANN)
[5] Neuron

مبتنی بر واژه نامه به دو دسته‌ی روش‌های مبتنی بر فرهنگ‌لغت[1] و روش‌های مبتنی بر پیکره[2] تقسیم می‌شوند.

۲ـ۲ـ۲ـ۱ روش‌های مبتنی بر فرهنگ لغت:

در این روش از یک فرهنگ‌لغت از پیش تعیین شده که شامل عبارات و کلمات نظری می‌باشد استفاده می‌شود که قطبیت هر کلمه (مثبت یا منفی) و شدت این قطبیت مشخص است. برای تعیین قطبیت یک جمله با این روش، ابتدا کلمات و عبارات نظری جمله استخراج و با توجه قطبیت آن‌ها در فرهنگ-لغت، قطبیت کل جمله تعیین می‌شود. همچنین در این روش از مترادف و متضاد کلمات نیز برای تعیین احساس استفاده می‌شود. فرهنگ‌لغت ابتدا با تعداد محدودی کلمه ایجاد می‌شود و در ادامه با توجه به مترادف و متضاد کلمات گسترش می‌یابد. مترادف و متضاد کلمات نیز با استفاده از وردنت[3] [۱۱] قابل دستیابی می‌باشد. از معایب این روش می‌توان به ناتوانی آن در تعیین قطبیت کلمات در زمینه‌های خاص اشاره کرد و از مزایای آن عدم نیاز به دانش قبلی (مجموعه داده آموزش) با جایگزین کردن فرهنگ‌لغت اشاره نمود.

۲ـ۲ـ۲ـ۲ روش‌های مبتنی بر پیکره:

در این روش یک پیکره عظیم از متون و کلمات مربوط به زمینه‌ای خاص ایجاد می‌شود و پس از آن الگوها و کلمات نظری متن و جمله مورد بررسی، در آن جستجو و با توجه به آن، قطبیت جمله تعیین می‌گردد. این جستجو و تحلیل احساس می‌تواند برحسب روش‌های آماری و یا استفاده از ساختار معنایی و نحوی الگو باشد. برای نمونه و تاثیر ساختار نحوی، هنگامی که دو کلمه (مثلا صفت) با حرف اضافه «و» در یک الگو قرار می‌گیرند، در اکثر موارد دارای قطبیت یکسان می‌باشند و در هنگامی که از حرف اضافه «اما» استفاده می‌شود، قطبیت جمله تغییر می‌نماید. از جمله نقاط قوت این روش، تعیین قطبیت

[1] Dictionary Based Approach
[2] Corpus Based Approach
[3] WordNet

متون و جملات در زمینه‌های خاص می‌باشد که به طور صحیح این عمل را انجام می‌دهد و از جمله معایب آن نیز می‌توان به مشکل بودن ایجاد یک تنه عظیم متن در زمینه‌ای خاص اشاره کرد.

۲ـ۲ـ۳ روش‌های ترکیبی[1]

در این روش از ترکیبی از هر دو روش مبتنی بر یادگیری ماشین و روش مبتنی بر فرهنگ‌لغت استفاده می‌شود. در [۱۲-۱۴] از این نوع روش استفاده شده است.

۲ـ۳ آپاچی اسپارک

اسپارک یک چارچوب پردازش داده‌های بزرگ به‌صورت توزیع شده و متن‌باز است که اولین بار در سال ۲۰۰۹ به عنوان یک پروژه در آزمایشگاه ای ام پی[2] دانشگاه کالیفرنیا، برکلی[3] طراحی شد و بعدها به بنیاد نرم‌افزاری آپاچی[4] هدیه گردید. در سال ۲۰۱۳ به یک پروژه انحصاری از بنیاد نرم‌افزاری آپاچی تبدیل شد و در اوایل سال ۲۰۱۴ میلادی به یکی از پروژه‌های برتر بنیاد ارتقا یافت.

یکی از ویژگی‌های اصلی و بزرگترین برتری اسپارک حفظ مجموعه‌داده در حین اجرای پردازش‌ها در حافظه می‌باشد، همین مورد بازدهی اسپارک را در مقابل سایر چارچوب‌ها مانند آپاچی هدوپ[5] افزایش می‌دهد. بعبارتی دیگر اسپارک به گونه‌ای طراحی و بهینه شده است که عملیات پردازش را در حافظه انجام دهد و این مزیت نسبت به برنامه نگاشت-کاهش که داده‌ها را بر روی دیسک‌سخت[6] نوشته و از روی دیسک‌سخت نیز برای پردازش فراخوانی می‌کند، موجب سرعت فوق‌العاده بالاتری شده است. در آپاچی هدوپ مدل نگاشت-کاهش بر روی دیسک‌سخت انجام می‌گیرد. در نتیجه، همان‌طور که اشاره شد مقداری از زمان صرف خواندن و نوشتن اطلاعات بر روی دیسک‌سخت و انتقال آن به حافظه یا

[1] Hybrid Methods
[2] AMPLab
[3] University of California, Berkeley
[4] Apache Software Foundation
[5] Apache Hadoop
[6] Hard Disk

بالعکس می‌شود.

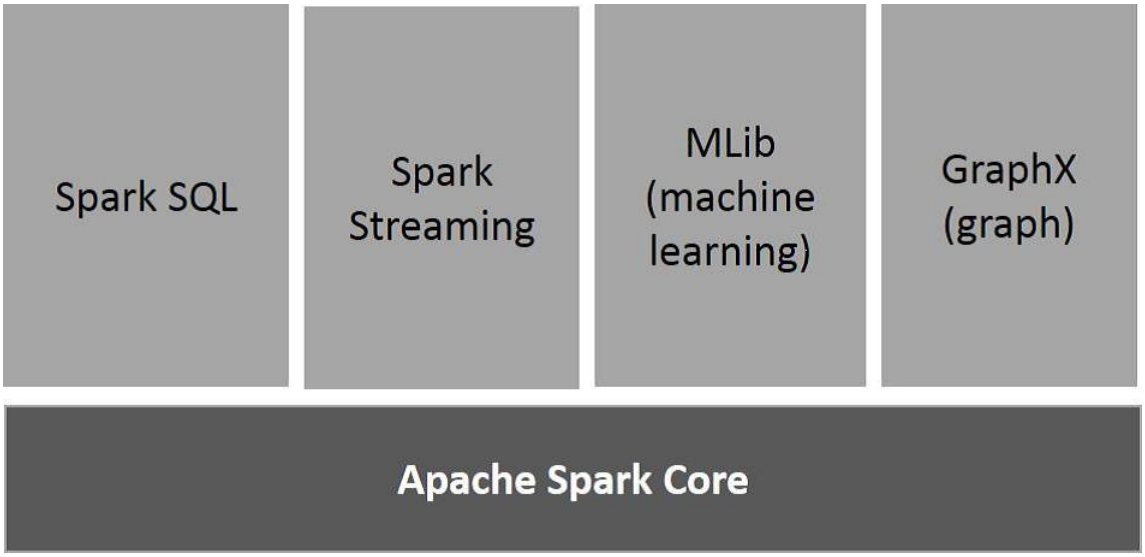
اسپارک دارای اجزا و مولفه‌های مختلفی‌ست که در شکل ۲-۲ مشخص می‌باشد. از دیگر مزایای اسپارک انسجام اجزای آن است. این مزیت، توانایی نوشتن برنامه‌هایی را می‌دهد که در آن از مدل‌های مختلف پردازش به طور یکپارچه استفاده می‌شود. برای مثال در اسپارک می‌توان برنامه‌ای نوشت که رده‌بندی داده‌ها از طریق یادگیری ماشین صورت می‌پذیرد و داده‌های ورودی نیز به صورت همزمان از منابع جریانی[1] دریافت می‌شوند. در آپاچی هدوپ برای هر کدام از این مولفه‌ها موتوری مجزا تعریف شده است که یکپارچه‌سازی و برنامه‌نویسی را مشکل‌تر می‌نماید. در ادامه به توضیح مختصر این اجزا پرداخته می‌شود.

هسته اسپارک[2]: مسئولیت هسته اسپارک انجام عملیات‌های اولیه اسپارک همانند مدیریت حافظه، زمان‌بندی وظایف، مقابله و رفع خطا و همچنین تعامل سیستم ذخیره‌سازی با دیگر اجزای اسپارک می‌باشد. مفهوم اصلی برنامه‌نویسی اسپارک یعنی مجموعه‌داده توزیع شده انعطاف‌پذیر[3] در هسته تعریف می‌شوند. RDDها مجموعه‌داده‌هایی هستند که توانایی اجرای توزیع شده بر روی چندین گره[4] محاسباتی متعدد را دارند و بدین ترتیب می‌توان آن‌ها را به صورت موازی پردازش نمود.

مولفه Spark SQL: این مولفه یک بخش بر روی هسته اسپارک می‌باشد و پشتیبانی از داده‌های ساخت‌یافته[5] و پرس و جو روی آن‌ها را فراهم می‌آورد.

مولفه Spark Streaming: مولفه پردازش داده‌های جریانی اسپارک یکی از اجزای اسپارک است که

[1] Streaming
[2] Apache Spark Core
[3] Resilient Distributed Dataset (RDD)
[4] Node
[5] Structured-Data



شکل ۲-۲: اجزای Apache Spark [۱۵]

از قابلیت زمان‌بندی سریع هسته‌های اسپارک برای فراهم کردن آنالیزهای جریانی[1] استفاده می‌کند و پردازش جریان داده‌ها را فراهم می‌آورد. یکی دیگر از دلایل محبوبیت اسپارک همین مولفه می‌باشد که امکان پردازش همزمان جریان داده همراه با ورود آن را در اختیار قرار می‌دهد.

مولفه MLlib: این مولفه همان‌طور که از نام آن مشخص است، کتابخانه یادگیری ماشین، برای اسپارک است و توانایی پردازش الگوریتم های مختلف داده کاوی و یادگیری ماشین را بر روی بستر اسپارک فراهم می آورد. MLlib انواع مختلفی از الگوریتم‌های یادگیری ماشین از جمله رده‌بندی، رگرسیون[2]، خوشه‌بندی[3] را ارائه می‌دهد و همچنین از قابلیت‌های مثل ارزیابی مدل و ورود داده‌ها پشتیبانی می‌کند.

مولفه GraphX: GraphX، یک مولفه و کتابخانه دیگر از اسپارک است. این کتابخانه، توانایی پردازش و تحلیل داده های گراف (نظیر گراف دوستی شبکه‌های اجتماعی) و یا داده هایی که به صورت گراف نمایش داده می شوند را به صورت استاندارد و موازی دارد. GraphX همچنین عملگرهای مختلفی را برای تغییر گراف‌ها (نظیر subgraph و mapVertices) و کتابخانه‌ای از الگوریتم‌های گراف خاص (نظیر

[1] Streaming Analytics
[2] Regression
[3] Clustering

PageRank و شمارش مثلث‌های گراف) فراهم آورده است.

اسپارک همچنین از چندین زبان برنامه‌نویسی مانند جاوا[1]، اسکالا[2]، پایتون[3]، آر[4] پشتیبانی می‌کند و SQL نیز از زبان هایی است که میتوانید به وسیله آن، با داده های موجود در اسپارک کار کنید.

۲ـ۴ ماشین بردار پشتیبان

۲ـ۴ـ۱ مقدمه

ماشین بردار پشتیبان، یکی از روش‌های یادگیری با نظارت[5] می‌باشد که در حال حاضر به صورت کارآمد و گسترده برای پاسخ به مسائل رده‌بندی مورد استفاده قرار می‌گیرد. الگوریتم اصلی ماشین بردار پشتیبان توسط ولادیمیر واپنیک[6] و الکسی چروونکسیس[7] در سال ۱۹۶۳ اختراع شد. در سال ۱۹۹۲ ولادیمیر واپنیک و همکاران یک روش برای ایجاد رده‌بندی‌های غیرخطی بوسیله استفاده از ترفند هسته[8] ارائه دادند [۱۶]. روش استاندارد رایج نیز توسط واپنیک و کورینا کورتس[9] در سال ۱۹۹۳ مطرح و در ۱۹۹۵ به چاپ رسید[10] [۱۷]. مبنای کاری رده‌بندی کنندهٔ SVM پیدا کردن خط (ابرصفحه) با حداکثر حاشیه برای رده‌بندی خطی داده‌ها است. به بیان دیگر در تقسیم خطی داده‌ها سعی می‌کند خطی را انتخاب کند که حاشیه اطمینان بیشتری داشته باشد. می‌توان پیدا کردن خط بهینه برای داده‌ها را به یک مسئله بهینه‌سازی درجه دوم محدب[11] تبدیل کرد و به وسیله روش‌های QP[12] که روش‌های شناخته شده‌ای در حل مسائل محدودیت‌دار هستند، حل نمود.

[1] Java
[2] Scala
[3] Python
[4] R
[5] Supervised Learning
[6] Vladimir N. Vapnik
[7] Alexey Yakovlevich Chervonenkis
[8] Kernel Trick
[9] Corinna Cortes
[10] https://en.wikipedia.org/wiki/Support_vector_machine#History
[11] Convex Quadratic Optimization Problem
[12] Quadratic Programming

۲ـ۴ـ۲ ماشین بردار پشتیبان

ابتدا برای توضیح بهتر به ماشین بردار پشتیبان خطی پرداخته می‌شود. فرض کنید تعداد N نقطه داده $\{(x_1,y_1),(x_2,y_2),...,(x_N,y_N)\}$ در مجموعه‌داده آموزش قرار دارد که در آن $x_i \in R^d$ و $y_i \in \{\pm 1\}$ می‌باشد. می‌خواهیم یک رده‌بند خطی آموزش دهیم:

$$f(x) = \mathrm{sgn}(w.x - b) \qquad (۱-۲)$$

همچنین می‌خواهیم خطی داشته باشیم که بیشترین حاشیه جداسازی با توجه به دو دسته را داشته باشد. به بیانی دیگر خط $H: y = w.x - b = 0$ که دو خط $H_1: y = w.x - b = +1$ و $H_2: y = w.x - b = -1$ موازی و با فاصله برابر نسبت به آن قرار دارند را تعیین نماییم با این شرط که هیچ داده ورودی بین H_1 و H_2 قرار نگیرد و همچنین فاصله بین H_1 و H_2 بیشینه گردد.

برای انتخاب حاشیه دو حالت در نظر گرفته می‌شود. برای داده‌هایی که براحتی قابل جداسازی می‌باشند، حالتی به نام سخت‌حاشیه[1] استفاده می‌شود. در این حالت می‌توان همیشه برای هر خط جداساز H و خطوط H_1 و H_2، با نرمال‌سازی[2] ضرایب بردار w خطوط H_1 و H_2 را به ترتیب به صورت $y = w.x - b = +1$ و $y = w.x - b = -1$ تعیین کرد. ما می‌خواهیم فاصله بین H_1 و H_2 بیشینه شوند. تعدادی داده نمونه مثبت بر روی خط H_1 و تعدادی داده نمونه منفی بر روی خط H_2 قرار دارند. به این نمونه‌ها بردارهای پشتیبان می‌گویند، زیرا تنها این نقاط در تعریف خط جداساز تاثیر گذار می‌باشند و سایر نمونه‌ها را می‌توان نادیده گرفت.

به یاد داریم که در فضای دو بعدی، فاصله یک نقطه (x_0, y_0) از خط $Ax + By + C = 0$ برابر با $\frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}}$ می‌باشد. به طور مشابه فاصله یک نقطه روی خط H_1 تا خط $H: y = w.x - b = 0$

[1] Hard-Margin
[2] Normalize

برابر با $\frac{|\mathrm{w.x-b}|}{\|\mathrm{w}\|}=\frac{1}{\|\mathrm{w}\|}$ می‌باشد و فاصله بین H_1 و H_2 برابر با $\frac{2}{\|\mathrm{w}\|}$ می‌باشد. بنابراین برای بیشینه کردن فاصله، باید $\|\mathrm{w}\|=\mathrm{w}^{\mathrm{T}}\mathrm{w}$ را کمینه نمود با این شرط که داده‌ای بین H_1 و H_2 وجود نداشته باشد:

$\mathrm{w.x-b}\geq+1$ برای داده‌های مثبت $y_i=+1$

$\mathrm{w.x-b}\leq-1$ برای داده‌های منفی $y_i=-1$

می‌توان این دو شرط را با یک‌دیگر ترکیب نمود: $y_i(\mathrm{w.x}_i-\mathrm{b})\geq 1$

بنابرین می‌توانیم مسئله‌مان را به صورت معادله‌ی بهینه‌سازی مقید زیر بیان نماییم:

$$y_i(\mathrm{w.x}_i-\mathrm{b})\geq 1 \quad \text{s.t.} \quad \min_{w,b}\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w} \qquad (۲\text{-}۲)$$

و این یک مسئله بهینه‌سازی درجه دوم محدب است.

برای حل این مسئله ساده‌تر آن است که مسئله به صورت دوگان[1] مطرح و حل شود. ضرایب لاگرانژ یک استراتژی برای پیدا کردن کمینه یا بیشینه یک تابع با توجه به محدودیت‌ها است. برای بدست آوردن شکل دوگان مسئله ضرایب لاگرانژ مثبت $\alpha_i\geq 0$ در قیود ضرب شده و از تابع هدف کسر می‌شود و در نهایت به شکل مسئله اولیه[2] زیر می‌رسیم:

$$L(\mathrm{w},\mathrm{b},\alpha)\equiv\frac{1}{2}\mathrm{w}^{\mathrm{T}}\mathrm{w}-\sum_{i=1}^{N}\alpha_i\,\mathrm{y}_i(\mathrm{w.x}_i-\mathrm{b})-\sum_{i=1}^{N}\alpha_i \qquad (۳\text{-}۲)$$

شرایط KKT[3] نقش مهمی را در مسائل بهینه‌سازی مقید، بازی می‌کنند. این شرایط، شرایط لازم و کافی برای داشتن پاسخ بهینه برای معادلات مقید را بیان می‌دارند و باید مشتق نسبت به متغیرها صفر باشد. با اعمال شرایط KKT روی معادله L و همچنین از L نسبت به w و b مشتق گرفته و برابر صفر

[1] Dual
[2] Primal Problem
[3] Karush-Kuhn-Tucker

قرار دهیم به روابط زیر می‌رسیم:

$$\frac{\partial L}{\partial w} = 0 \Rightarrow \mathrm{w} = \sum_{i=1}^{N} \alpha_i y_i x_i$$
$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (۴-۲)$$

و با جایگذاری این روابط در رابطه $L(\mathrm{w}, \mathrm{b}, \alpha)$ ، رابطه زیر را داریم:

$$L_D \equiv \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j \, \mathrm{y}_i \mathrm{y}_j (\mathrm{x}_i \, . x_j) \qquad (۵-۲)$$

که معادله دوگان[1] نامیده می‌شود. L و L_D هر دو از شرایط یکسانی بدست آمده‌اند و لذا با محاسبه کمینه L و یا بیشینه L_D با شرط $\alpha_i \geq 0$ می‌توان به جواب رسید.

برای هر نمونه آموزشی یک ضریب لاگرانژ $\alpha_i \geq 0$ وجود دارد. به نمونه‌هایی که برای آنها $\alpha_i > 0$ می‌باشد، بردارهای پشتیبان گفته می‌شود و همان نقاطی هستند که بر روی خط جداساز قرار می‌گیرند.

پس از حل معادله فوق و بدست آوردن مقادیر α_i، می‌توان $\mathrm{w} = \sum_{i=1}^{N} \alpha_i y_i x_i$ را محاسبه و پس از آن می‌توان داده x جدید را رده‌بندی نمود:

$$\begin{aligned} f(\mathrm{x}) &= \mathrm{sgn}(\mathrm{w}.\mathrm{x} + \mathrm{b}) \\ &= \mathrm{sgn}((\sum_{i=1}^{N} \alpha_i y_i x_i).\mathrm{x} + \mathrm{b}) \\ &= \mathrm{sgn}((\sum_{i=1}^{N} \alpha_i y_i (x_i.\mathrm{x}) + \mathrm{b}) \end{aligned} \qquad (۶-۲)$$

۲ـ۴ـ۳ بهینه‌سازی متوالی کمینه

بهینه‌سازی متوالی کمینه(SMO[2])، یک الگوریتم برای حل مسائل QP منتج شده در ضمن فرآیند

[1] Dual Problem
[2] Sequential Minimal Optimization

آموزش ماشین بردار پشتیبان می‌باشد. این روش در سال ۱۹۹۸ توسط جان پلات[1] در مرکز تحقیقات مایکروسافت[2] ابداع گردید[۱۸]. SMO برای تعلیم ماشین بردار پشتیبان به طور گسترده‌ای مورد استفاده قرار گرفت و سبب موج استقبال وسیعی توسط انجمن SVM گردید. روش‌های موجود تا قبل از ارائه SMO برای حل SVM بسیار پیچیده و هزینه‌بر بودند.

SMO الگوریتم ساده‌ایست که به سرعت مسئله QP در SVM را بدون استفاده از راه‌حل عددی بهینه-سازی درجه دوم، حل می‌کند. SMO اقدام به تجزیه مسئله‌ی اصلی به تعدادی زیرمسئله می‌نماید و به این ترتیب همگرایی روش را نیز تضمین می‌نماید. در هر مرحله SMO دو ضریب لاگرانژ را برای بهینه کردن انتخاب می‌نماید و مقدار بهینه را برای این ضرائب پیدا نموده و SVM را برای یافتن مقدار بهینه به‌روز می‌نماید. مزیت روش SMO در این است که راه‌حل مسئله برای دو ضریب لاگرانژ می‌تواند به صورت تحلیلی حاصل شود. بنابراین در سراسر راه‌حل از بهینه‌سازی QP به صورت عددی پرهیز می-گردد.

میزان فضای مورد نیاز برای SMO نسبت به اندازه‌ی مجموعه‌ی آموزش خطی می‌باشد که این خود به SMO اجازه‌ی مدیریت مجموعه‌های آموزشی بزرگ را می‌دهد. با توجه به اینکه در این روش از محاسبات ماتریسی اجتناب شده است. لذا SMO بین مرتبه‌های خطی و دوم نسبت به اندازه‌ی مجموعه-ی آموزشی قرار می‌گیرد، در حالی که در SVM استاندارد در میان مرتبه‌های دوم و سوم قرار می‌گیرند. در دنیای واقعی SMO می‌تواند هزار بار از الگوریتم قطعه‌بندی استاندارد برای SVM سریع‌تر باشد[۱۸].

الگوریتم ساده شده SMO:

الگوریتم SMO دو ضریب α، یعنی α_i و α_j را در هر مرحله تکرار انتخاب می‌نماید و ارزش عینی هر دو ضریب را به طور مشترک بهینه می‌کند. پس از آن پارامتر b بر اساس αهای جدید تنظیم و محاسبه

[1] John Platt
[2] Microsoft Research

می‌شود. در ادامه این مراحل بیشتر توضیح داده می‌شود [۱۹].

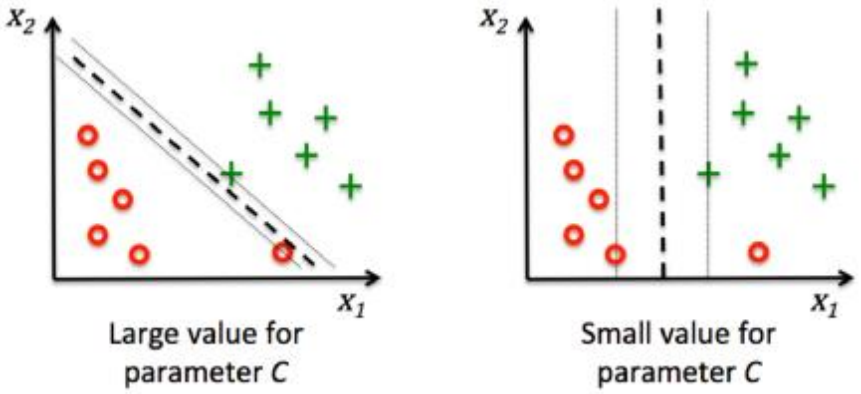
انتخاب دو ضریب α: بخش عمده‌ای از الگوریتم SMO روش اکتشافی[1] برای انتخاب ضرایب α_i و α_j می‌باشد تا آنها را بهینه نماید. برای مجموعه داده‌های بزرگ، این نکته برای سرعت انجام الگوریتم حیاتی می‌باشد. زیرا $N(N-1)$ تعداد انتخاب‌های ممکن برای ضرایب α_i و α_j می‌باشد و برای برخی از این انتخاب‌ها، بهبود بسیار کمتر نسبت به سایر ضرایب می‌باشد.

برای این کار از دو انتخاب اکتشافی مجزا برای ضرائب استفاده می‌شود. در الگوریتم ارائه شده [۱۸] از دو حلقه‌ی متداخل استفاده شده است. در حلقه‌ی خارجی در اولین اجرا، تمام مجموعه‌ی داده‌های آموزشی را برای یافتن نمونه‌ای که شرایط KKT را نقض نماید، پویش می‌نماید. اگر نمونه‌ای شرایط KKT را نقض نماید، مناسب برای بهینه‌سازی خواهد بود. بعد از اولین تکرار حلقه‌ی خارجی تکرار را بر روی نمونه‌هایی از ضرایب لاگرانژ که بین ۰ و C قرار دارند ($0 < \alpha_i < C$)، ادامه می‌دهد.

- پارامتر C برای اجتناب از داده‌های آموزشی اشتباه رده‌بندی شده در بهینه‌سازی SVM می‌باشد. برای مقادیر بزرگ C، حاشیه کم برای ابرصفحه[2] انتخاب می‌شود تا رده‌بندی داده‌ها درست انجام شود. برای مقادیر کوچک C نیز حاشیه زیاد برای ابرصفحه درنظر گرفته می‌شود. در این حالت داده‌های بیشتری اشتباه رده‌بندی می‌شود. شکل ۲-۳ تاثیر مقادیر مختلف C نمایش می‌دهد. سپس مجدداً، هر نمونه با شرایط KKT مقایسه شده و نمونه‌ای که آنرا نقض نماید، مناسب برای بهینه‌سازی خواهد بود. این بهینه‌سازی در حلقه‌ی خارجی تا زمانی اجرا می‌شود که تمام این نمونه‌های مذکور شرایط KKT را با خطای ε برآورده سازند. در ادامه حلقه خارجی بازگشت نموده و تکرار را روی تمام مجموعه‌ی آموزشی انجام میدهد. حلقه خارجی به طور مداوم تا زمانی که تمامی نمونه‌های آزمایشی شرایط KKT را با خطای ε برآورده سازند، بین این دو انتخاب این جابجا می‌شود. بعد از انتخاب اولین ضریب توسط حلقه‌ی

[1] Heurist
[2] Hyperplane

خارجی، دومین ضریب توسط حلقه داخلی و با تکرار بر روی ضرائب باقیمانده، برای انجام بهینه‌سازی انتخاب می‌شود.



شکل ۲-۳: پارامتر C در SVM

محاسبه مقادیر دو ضریب: برای ضرایب لاگرانژ انتخاب بهینه کردن α_i و α_j، ابتدا قیود روی مقادیر این پارامترها محاسبه می‌کنیم و سپس مسئله بیشینه‌سازی مقید مربوط به آن را حل می کنیم.

ابتدا مرزهای L و H را بدست می‌آوریم به طوری که $L < \alpha_i < H$ و به شرطی قید $0 < \alpha_i < \mathrm{C}$ برآورده شود. این مرزها توسط روابط زیر بدست می‌آید:

$$\begin{aligned} &\text{if } y_i \neq y_j, \ \mathrm{L}=\max(0, \alpha_j - \alpha_j), \ \mathrm{H}=\min(\mathrm{C}, \mathrm{C} + \alpha_j - \alpha_j) \\ &\text{if } y_i = y_j, \ \mathrm{L}=\max(0, \alpha_j + \alpha_j - \mathrm{C}), \ \mathrm{H}=\min(\mathrm{C}, \alpha_j + \alpha_j) \end{aligned} \qquad (۷-۲)$$

حال می‌خواهیم مقدار α_j را بدست آوریم تا تابع هدف را بیشینه نماییم. اگر این مقدار از مرزهای L و H عبور کند، ما مقدار α_j در این محدوده نگه می‌داریم. مقدار بهینه α_j با استفاده از رابطه زیر بدست می‌آید:

$$\alpha_j := \alpha_j - \frac{y_j(\mathrm{E}_i - \mathrm{E}_j)}{\eta} \qquad (۸-۲)$$

که در آن E_k و η از روابط زیر بدست می‌آید:

$$E_k = f(\mathrm{x}_k) - \mathrm{y}_k \qquad (۹-۲)$$
$$\eta = 2(\mathrm{x}_i . \mathrm{x}_j) - (\mathrm{x}_i . \mathrm{x}_i) - (\mathrm{x}_j . \mathrm{x}_j)$$

E_k را می‌توان به عنوان خطایی بین خروجی SVM برای نمونه‌ی kام و برچسب درست آن دانست که $f(\mathrm{x}_k)$ در آن رابطه زیر می‌باشد:

$$f(\mathrm{x}) = \sum_{i=1}^{N} \alpha_i \ \mathrm{y}_i (\mathrm{x}_i . \mathrm{x}) + \mathrm{b} \qquad (۱۰-۲)$$

همچنین هنگام محاسبه پارامتر η می‌توان در صورت استفاده از هسته ، از آن بجای ضرب داخلی به صورت دلخواه استفاده نمود.

پس از آن ما مقدار α_j را با استفاده از رابطه زیر برش می‌دهیم تا در بازه [L,H] قرار گیرد:

$$\alpha_j := \begin{cases} \mathrm{H} & \text{if } \alpha_j > \mathrm{H} \\ \alpha_j & \text{if } \mathrm{L} < \alpha_j < \mathrm{H} \\ \mathrm{L} & \text{if } \alpha_j < \mathrm{L} \end{cases} \qquad (۱۱-۲)$$

در نهایت پس از اینکه مقدار α_j را بدست آوردیم، مقدار α_i را با توجه به α_i و با استفاده از رابطه زیر بدست می‌آوریم:

$$\alpha_i := \alpha_i + \mathrm{y}_i \mathrm{y}_j (\alpha_j^{(\mathrm{old})} - \alpha_j) \qquad (۱۲-۲)$$

$\alpha_j^{(\mathrm{old})}$ مقدار قبل از بروزرسانی در این مرحله می‌باشد. همچنین اگر $\eta = 0$ حالتی‌ست که ما از این جفت α نمی‌توانیم بهبودی داشته باشیم و به سراغ یک جفت دیگر می‌رویم.

محاسبه مقدار b: پس از بهینه‌سازی مقادیر α_i و α_j مقدار آستانه b را انتخاب می‌نماییم به قسمی که شرایط KKT را برای i ام و j امین نمونه برآورده نماید. برای بدست آوردن آستانه b جدید، اگر مقدار α_i پس از بهینه‌سازی در محدوده معین شده قرار گیرد (برای مثال $0 < \alpha_i < \mathrm{C}$) بنابراین مقدار آستانه b_1 از رابطه زیر بدست می‌آید:

$$b_1 = b - E_i - \mathrm{y}_i(\alpha_i - \alpha_i^{(\mathrm{old})})(\mathrm{x}_i \,.\, \mathrm{x}_i) - \mathrm{y}_j(\alpha_j - \alpha_j^{(\mathrm{old})})(\mathrm{x}_i \,.\, \mathrm{x}_j) \qquad (۱۳-۲)$$

به طور مشابه، b_2 از رابطه زیر بدست می‌آید اگر $0 < \alpha_j < \mathrm{C}$ قرار گیرد:

$$b_2 = b - E_j - \mathrm{y}_i(\alpha_i - \alpha_i^{(\mathrm{old})})(\mathrm{x}_i \,.\, \mathrm{x}_j) - \mathrm{y}_j(\alpha_j - \alpha_j^{(\mathrm{old})})(\mathrm{x}_j \,.\, \mathrm{x}_j) \qquad (۱۴-۲)$$

اگر هر دو ضریب در محدوده $0 < \alpha_i < \mathrm{C}$ و $0 < \alpha_j < \mathrm{C}$ قرار گیرند، مقادیر آستانه فوق معتبر می‌باشند و آن‌ها با هم برابر خواهند شد. اگر مقادیر جدید دو ضریب α_i و α_j روی مرزها قرار بگیرد (برای مثال $\alpha_i = 0 \text{ or } \alpha_i = C \text{ and } \alpha_j = 0 \text{ or } \alpha_j = C$) تمام مقادیر آستانه بین دو مقدار b_1 و b_2، شرایط KKT را برآورده می‌نماید. بنابراین می‌توان مقدار b جدید را بر طبق رابطه زیر بدست آورد:

$$\mathrm{b} := \begin{cases} \mathrm{b}_1 & \text{if } 0 < \alpha_i < \mathrm{C} \\ \mathrm{b}_2 & \text{if } 0 < \alpha_j < \mathrm{C} \\ (\mathrm{b}_1 + \mathrm{b}_2)/2 & \text{otherwise} \end{cases} \qquad (۱۵-۲)$$

در شکل ۲-۳ نیز شبه‌کد این روش آمده است.

۲ـ۴ـ۴ هسته در ماشین بردار پشتیبان

ماشین بردار پشتیبان با تشکیل خطی جداساز بین دو دسته‌ی داده‌های آموزش، اقدام به رده‌بندی داده‌های آزمون می‌نماید. این مدل پایه را ماشین بردار پشتیبان خطی[1] می‌گویند. در برخی از مسائل داده‌ها به راحتی قابل تفکیک از یکدیگر نمی‌باشند و در این حالت یک ماشین بردار پشتیبان خطی موثر باشد. در این حالت اگر داده‌ها را به فضایی با ابعاد دیگر انتقال دهیم می‌توان راه‌حلی برای جداسازی داده‌ها بدست آورد. برای این عمل می‌توان از توابع مختلفی از جمله چندجمله‌ای[2]، گوسین[3] یا تابع پایه شعاعی[4] (RBF) استفاده نمود که آن را هسته می‌نماند. شکل ۲-۵ نمونه‌ای از کاربرد هسته را نشان می‌دهد.

[1] Linear SVM
[2] Polynomial
[3] Gussian
[4] Radial Basic Function

Algorithm: Simplified SMO

Input:

C: regularization parameter

tol: numerical tolerance

max_passes: max # of times to iterate over α 's without changing

$(x_1, y_1), ..., (x_N, y_N)$: training data

Output:

$\alpha \in \mathbb{R}^N$: Lagrange multipliers for solution

$b \in \mathbb{R}$: threshold for solution

Pseudo-Code:

Initialize $\alpha_i = 0, \forall i,\ b = 0$

Initialize passes = 0

while (passes < max_passes)

num_changed_alphas = 0

for i = 1,...,m

Calculate $E_i = f(x_i) - y_i$ using (9-2) and (10-2)

if $((y_i E_i < -tol \ \&\& \ \alpha_i < C) \,||\, (y_i E_i > tol \ \&\& \ \alpha_i > 0))$

Select j $\neq$ i randomly

Calculate $E_j = f(x_j) - y_j$ using (9-2) and (10-2)

Save old α 's: $\alpha_i^{(old)} = \alpha_i$, $\alpha_j^{(old)} = \alpha_j$

Compute L and H by (7-2)

if (L == H)

continue to next i

Compute η by (9-2)

If(η >= 0)

continue to next i

Compute and clip new value for α_i using (8-2) and (11-2)

If $(|\alpha_j - \alpha_j^{(old)}| < 10^{-5})$

continue to next i

Determine value for α_i using (12-2)

Compute b_1 and b_2 using (13-2) and (14-2) respectively

Compute b by (15-2)

num_changed_alphas := num_changed_alphas +1

end if

end for

if (num_changed_alphas == 0)

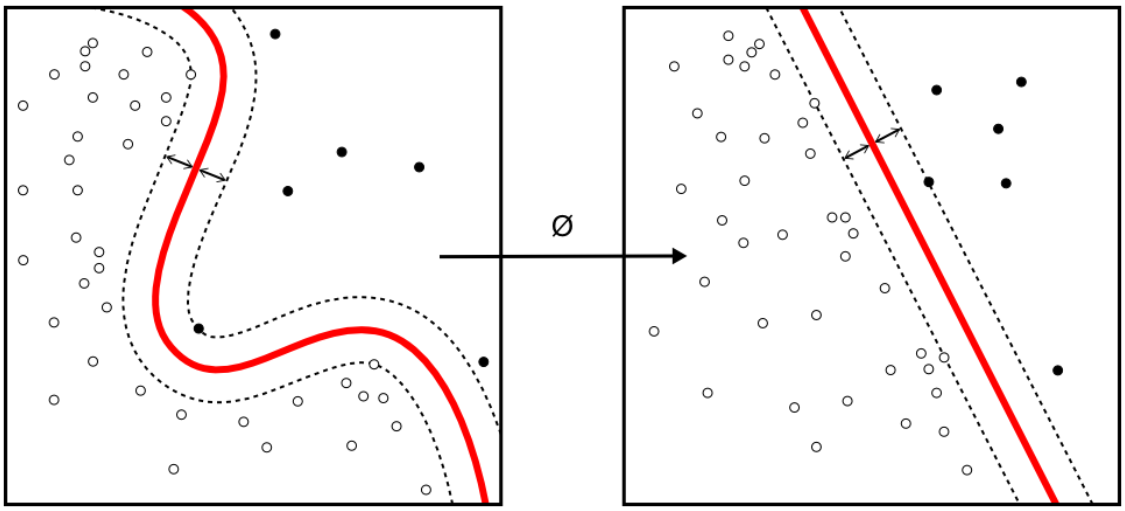
passes := passes +1

else

passes := 0

end while

شکل ۲-۳: الگوریتم SMO ساده شده[۱۹]



شکل ۲-۵: استفاده از هسته برای جداسازی داده‌ها

۲ـ۵ مشکلات مرتبط با نظرکاوی و چالش‌ها

در حوزه نظرکاوی و تجزیه و تحلیل احساسات، تعدادی مشکل وجود دارد که به آن اشاره می‌شود و می‌توان برای کارهای آینده بر روی این موضوعات تحقیق نمود. اکثرا این مشکلات مربوط به زمینه پردازش زبان طبیعی می‌باشند.

۲ـ۵ـ۱ مشکلات مربوط به منبع داده

در زمینه نظرکاوی فقدان مجموعه داده‌ای معیار برای ارزیابی نتایج محسوس می‌باشد. همچنین برای بعضی از زمینه‌های نظرکاوی و برخی زبان‌ها مانند زبان‌های فارسی و عربی و ... مشکل تهیه مجموعه داده وجود دارد. میکالای[1] [۲۰] تعداد کمی از معروف‌ترین مجموعه داده‌ها را برای نظرکاوی معرفی کرده است که تا حدودی می‌توان از آن‌ها بعنوان معیار استفاده نمود (به دلیل استفاده فراوان). همچنین وب‌سایت‌های IMDB[2]، آمازون و توییتر از مهم‌ترین منابع برای نظرکاوی می‌باشند.

[1] Mikalai Tsytsarau
[2] Intenational Movie Database

۲ـ۵ـ۲ مشکلات مربوط به پردازش زبان طبیعی

مشکلاتی در پردازش زبان طبیعی وجود دارد که در این بخش تعدادی از این مشکلات را مرور می‌نماییم.

نفی: یکی از بخش‌های مهم تشخیص احساسات توجه و اداره کردن کلمات نفی می‌باشد. برای مثال نظر «رستوران خوبی بود.» دارای قطبیت مثبت می‌باشد اما نظر «رستوران خوبی نبود.» دارای قطبیت منفی می‌باشد، علی‌رقم استفاده از صفتی مانند «خوب» که بار مثبت دارد. همچنین به این نکته نیز باید توجه شود که همیشه با دیدن کلمات نفی نباید بار معنایی جمله را معکوس نمود.

برای مثال دو نظر «جای تعجب ندارد که رستوران خوبی است.» و «نه تنها فضای رستوران جالب بود، بلکه سرویس‌دهی نیز عالی بود.» دارای کلمات نفی می‌باشند اما قطبیت نظرات کاملا مثبت می‌باشد و همین موضوع باعث ایجاد مشکل در این بخش می‌شود.

وابستگی به دامنه: برخی از کلمات با توجه به دامنه و جمله‌ای که در آن قرار می‌گیرند دارای قطبیت و بار معنایی متفاوتی می‌باشند، بنابراین توجه به این موضوع نیز حائز اهمیت می‌باشد. برای مثال «طول عمر باتری گوشی طولانی است.» کلمه طولانی دارای بار مثبت می‌باشد اما در «زمان بارگذاری سیستم عامل گوشی، طولانی است.» کلمه طولانی، دارای بار منفی می‌باشد.

استفاده از ضمیر: در برخی از نظرات از ضمیر استفاده می‌شود و یا اشاره به موجودیتی دارد که مشخص نیست و اداره کردن این موضوع مشکل می‌باشد. برای مثال، تعیین قطبیت نظر «این فیلم کارگردان نسبت به فیلم قبلی بهتر است.» وابسته به قطبیت فیلم قبلی کارگردان دارد.

استفاده از چند زبان: در برخی از تارنماها، نظرات با زبان‌های متفاوتی بیان می‌شود که طبیعتا با استفاده از روش‌های متکی به یک زبان نمی‌توان به نظرکاوی آن‌ها پرداخت و یا اینکه در یک نظر از چند زبان استفاده می‌شود. برای مثال «این گوشی موبایل از کابل رابط OTG پشتیبانی می‌کند.»

ابهام فهم جمله برای رایانه: برخی نظرات دارای بار مثبت و منفی به طور همزمان می‌باشند و با این حال برای انسان‌ها به راحتی قابل فهم می‌باشد اما درک و تحلیل آن برای رایانه مشکل می‌باشد. برای مثال نظر«فیلم مانند بمب ترکید با وجود اینکه بازیگر اصلی آن را لرزاند (و خوب کار نکرد).»

فصل سوم

کارهای پیشین

در این فصل به مرور کارهای پیشین پرداخته می‌شود.

۳ـ۱ روش‌های نظرکاوی

همانطور که در فصل قبل توضیح داده شد، روش‌های نظرکاوی دارای دو قسمت اصلی روش‌های مبتنی بر یادگیری ماشین و روش‌های مبتنی بر واژه نامه[1] می‌باشند. در این قسمت پژوهش‌های مرتبط پیشین به تفکیک روش‌های استفاده شده آورده شده است.

۳ـ۱ـ۱ روش‌های مبتنی بر یادگیری ماشین

۳ـ۱ـ۱ـ۱ بیز ساده، ماشین بردار پشتیبان و بیشینه آنتروپی

پانگ و همکاران برای سه رده‌بندی کننده فوق مقایسه‌ای را با استفاده ویژگی‌های متفاوت انجام داده‌اند [۸]. در نتایج این پژوهش، هنگامی که مجموعه ویژگی‌ها کوچک باشد، رده‌بند بیز ساده بهتر عمل می‌نماید. همچنین هنگامی که مجموعه ویژگی‌ها بزرگ باشد، ماشین بردار پشتیبان نسبت به دو روش دیگر بهتر عمل می‌کند و همچنین در حالت کلی با افزایش اندازه مجموعه ویژگی، روش بیشینه آنتروپی بهتر از بیز ساده عمل می‌نماید.

۳ـ۱ـ۱ـ۲ درخت تصمیم:

زینیا سینگلا[2] و همکاران[۲۱] برای رده‌بندی برای تحلیل نظرات کاربران از سه روش رده‌بند SVM، Naïve Bayes(NB) و همچنین درخت تصمیم استفاده کرده است. آنوجا پی جین[3] و همکارش از دو رده‌بند NB و درخت تصمیم برای تحلیل و رده‌بندی نظرات کاربران استفاده نموده‌اند[۲۲].

۳ـ۱ـ۱ـ۳ رده‌بندی کننده‌های مبتنی بر قاعده:

یولیا بیدولیا[4] و برونوا[5][۲۳] یک رده‌بند مبتنی بر قانون را برای تحلیل احساسات در حوزه کیفیت

[1] Lexicon Based Methods
[2] Zeenia Singla
[3] Anuja P Jain
[4] Yuliya Bidulya
[5] Brunova

خدمات بانکی انجام داده است. آیمن سیدیکوآ[1] و همکارانش از ترکیب رده‌بند مبتنی بر قانون و رده‌بند NB تحت نظارت ضعیف برای تحلیل نظرات کاربران در توییتر استفاده نموده است[۲۴].

۳ـ۱ـ۱ـ۴ شبکه عصبی مصنوعی:

رودریگو مورائز[2] و همکاران در پژوهشی به مقایسه تکنیک شبکه عصبی و ماشین بردار پشتیبان پرداخته‌اند و همچنین از رده‌بند NB نیز استفاده نموده‌اند [۲۵]. در این مقایسه که بر روی مجموعه داده‌هایی از جمله نظرات مربوط به فیلم، دوربین و کتاب انجام شده است، شبکه عصبی نتایج بهتری را نسبت به ماشین بردار پشتیبان بدست آورده است. همچنین هر کدام از تکنیک‌های فوق دارای محدودیت‌هایی می‌باشند. برای مثال هزینه زمانی و محاسباتی مرحله آموزش برای شبکه عصبی مصنوعی و همچنین هزینه محاسباتی برای ماشین بردار پشتیبان از محدودیت‌ها و معایب این روش‌ها می‌باشند.

در پژوهش دیگری، هائو لیندوگ[3] و ژائو هوئی[4] [۲۶] از شبکه عصبی کانولوشن استفاده کرده‌اند و سعی در بهبود آن داشته‌اند. همچنین در این پژوهش از سایر روش‌های یادگیری از جمله SVM، NB، MaxEntropy و CNN استاندارد استفاده شده است.

تانجیم الهق[5] و همکاران [۲۷] به تحلیل نظرات کاربران بر روی مجموعه‌داده آمازون پرداخته‌اند. در این پژوهش از چندین روش مبتنی بر یادگیری ماشین استفاده شده است. SVM خطی، Mutlinimoial NB، SGD[6]، Random Forest، Logistic Regression و درخت تصمیم روش‌های استفاده در این پژوهش می‌باشد. در ادامه به بررسی سایر پژوهش‌هایی که بر روی مجموعه‌داده آمازون اجرا شده، پرداخته شده است.

[1] Umme Aymun Siddiqua
[2] Rodrigo Moraes
[3] Hao Lidong
[4] Zhao Hui
[5] Tanjim Ul Haque
[6] Stochastic Gradient Descent

۳ـ۱ـ۲ روش‌های مبتنی بر فرهنگ لغت

سحر سوهان‌گیر و همکاران در پژوهشی از روش‌های مبتنی بر فرهنگ لغت برای تشخیص احساسات استفاده نموده‌اند [۲۸]. در پژوهش انجام شده از SentiWordNet و TextBlob برای رده‌بندی استفاده کرده است. همچنین در این پژوهش از سه روش مبتنی بر یادگیری ماشین از جمله Linear SVM، NB و Logistic Regression برای مقایسه استفاده شده است.

۳ـ۲ مقیاس‌پذیری[۱]

«مقیاس‌پذیری توانایی یک سامانه[۲]، شبکه[۳] و یا فرآیند[۴] برای رسیدگی (پاسخ‌گویی) به مقداری کار می‌باشد که در حال افزایش است و یا آمادگی آن برای بزرگ شدن (توسعه و افزایش عملکرد) در مقابل رشد کار است [۲۹].»

امروزه با توجه به گسترش شبکه‌های اجتماعی، حجم زیادی داده در فضای مجازی تولید می‌شود. این داده‌ها در بسیاری از زمینه‌ها می‌توانند مفید باشند و اطلاعات ارزشمندی را شامل شوند. اما پردازش این حجم عظیم از داده، بخصوص در مواردی که نیازمند به پردازش بلادرنگ[۵] می‌باشد، روش‌های نوینی را نسبت به روش‌های گذشته می‌طلبد. پژوهش‌گران راه‌کارها و روش‌های گوناگونی را برای غلبه بر این مشکل ارائه داده‌اند. برای نمونه پژوهش‌گران در حوزه نظرکاوی به ارائه چارچوب‌های مقیاس پذیر[۶] پرداخته‌اند که درون خود از روش‌های مختلفی از جمله روش‌های مبتنی بر یادگیری ماشین و یا روش‌های مبتنی بر پردازش زبان طبیعی (NLP) بهره می‌برند. در ادامه به برخی از این پژوهش‌ها اشاره و تا حدودی به کلیت روش ارائه شده می‌پردازیم.

[1] Scalability
[2] System
[3] Network
[4] Process
[5] Real-Time
[6] Scalable

ماریا کاراناسو[1] و همکاران [۳۰] روشی مقیاس‌پذیر برای غلبه بر مشکل پردازش عظیم داده‌ها در زمینه نظرکاوی ارائه داده‌اند. روشی که در این مرجع بیان شده دارای دو قسمت کلی می‌باشد. در قسمت اول که مرحله برون‌خطی[2] نامیده می‌شود، رده‌بند با استفاده از مجموعه داده آموزشی برچسب‌گذاری شده، آموزش داده می‌شود. در مرحله دوم که مرحله درون‌خطی[3] نامیده می‌شود، داده‌ها به صورت بلادرنگ دریافت می‌شوند و پیش‌پردازش و انتخاب ویژگی بر رویشان اعمال می‌شود و درنهایت از همان رده‌بند آموزش دیده برای رده‌بندی داده‌های بلادرنگ استفاده می‌شود. در روش پیشنهادی از یادگیری گروهی استفاده شده است. استفاده از رأی اکثریت[4] برای بهبود عملکرد رده‌بندها استفاده شده است. همچنین در مرحله دوم از بازخورد[5] برای ارزیابی استفاده می‌شود و درصورت عدم دقت کافی مرحله اول تکرار می‌شود. این روش بر روی چارچوب استورم[6] پیاده‌سازی شده است.

آسو حمزه‌ای و همکاران، با ارائه روشی به نام SSSA[7] سعی در بهبود مشکل پردازش عظیم داده‌ها در زمینه نظرکاوی داشته‌اند[۳۱]. در روش پیشنهادی پس از مراحل پیش‌پردازش، به هر کلمه امتیازی با استفاده از ابزارهای وردنت و SentiWordNet تعلق می‌گیرد و در نهایت با استفاده از مجموع امتیازات بدست آمده، قطبیت جمله یا نظر تعیین می‌گرد. در روش ارائه شده سعی شده کندی روش‌های مبتنی بر معنا[8]، با استفاده از ویژگی‌های NLP و روش‌های مقیاس‌پذیر بهبود یابد.

باستوری باتاچارجی[9] و لیندا پتزولد[10]، روشی مقیاس‌پذیر روی مجموعه نظرات مشتریان رستوران‌ها مطرح کرده‌اند[۳۲]. در روش بیان شده با ارائه تعریفی جدید به نام فراویژگی[11] سعی در ایجاد ویژگی جدیدی نموده‌اند تا باعث کاهش بردار ویژگی و افزایش سرعت و دقت شوند. فراویژگی‌ها با توجه به هر

[1] Maria Karanasou
[2] Offline
[3] Online
[4] Maxvote
[5] Feedback
[6] Storm Framwork
[7] Semantic Scoring Sentiment Analysis
[8] Semantic-based
[9] Kasturi Bhattacharjee
[10] Linda Petzold
[11] Meta-Feature

مجموعه داده متفاوت هست و وابسته به نوع مجموعه نظر می‌باشد و در روش مورد نظر ابتدا جنبه[۱]ها و توصیف‌گر[۲]های نظرات استخراج و رده‌بندی می‌شوند و سپس فراویژگی‌ها با ترکیب جنبه‌ها و توصیف-گرها ایجاد می‌شود و در نهایت با توجه به فراویژگی‌های موجود در نظرات، قطبیت نظر تعیین می‌گردد. برای مثال نظر«غذاهای اینجا عالی‌ست.» فراویژگی (غذا، عالی) وجود دارد که غذا خود شامل یک دسته و عالی نیز خود دارای یک زیرمجموعه هم معنی می‌باشند.

کاوالین[۳] و همکاران[۳۳] یک معماری مقیاس‌پذیر برای تحلیل بلادرنگ داده‌های وب‌نوشت‌های کوچک[۴] ارائه کرده‌اند. این معماری بر روی زیرساخت جریانی IBM InfoSphere[۵] پیاده‌سازی شده است و توانایی مقابله با حجم زیاد نظر را داراست. این روش به تحلیل نظرات کاربران توییتر در طی برگزاری بازی نهایی[۶] تیم فوتبال برزیل در جام کنفدراسیون‌ها ۲۰۱۳ [۷] به صورت درون‌خطی پرداخته است. در روش ارائه شده رده‌بند با استفاده از نظرات کاربران در بازی‌های پیشین آموزش دیده است و از آن برای رده‌بندی نظرات کاربران در بازی نهایی استفاده می‌شود.

۳ـ۳ ماشین بردار پشتیبان موازی[۸]

یکی از راهکارهای اخیر که برای مقابله با هزینه پردازش داده‌های حجیم و کاهش آن مورد توجه قرار گرفته استفاده از چندین گره برای پردازش سریع‌تر و موازی است. در این راه‌کار از برنامه‌نویسی نگاشت-کاهش[۹] استفاده می‌شود. مراحلی مانند پیش‌پردازش، استخراج و انتخاب ویژگی به راحتی به صورت موازی قابل پیاده‌سازی می‌باشد اما آموزش رده‌بند و استفاده از آن به صورت موازی یکی از دغدغه‌های این راه‌کار می‌باشد.

---

[1] Aspect
[2] Descriptor
[3] P. R. Cavalin
[4] Microblogging data
[5] IBM InfoSphere Streams platform
[6] Final
[7] 2013 FIFA Confederations Cup
[8] Parallel SVM
[9] MapReduce

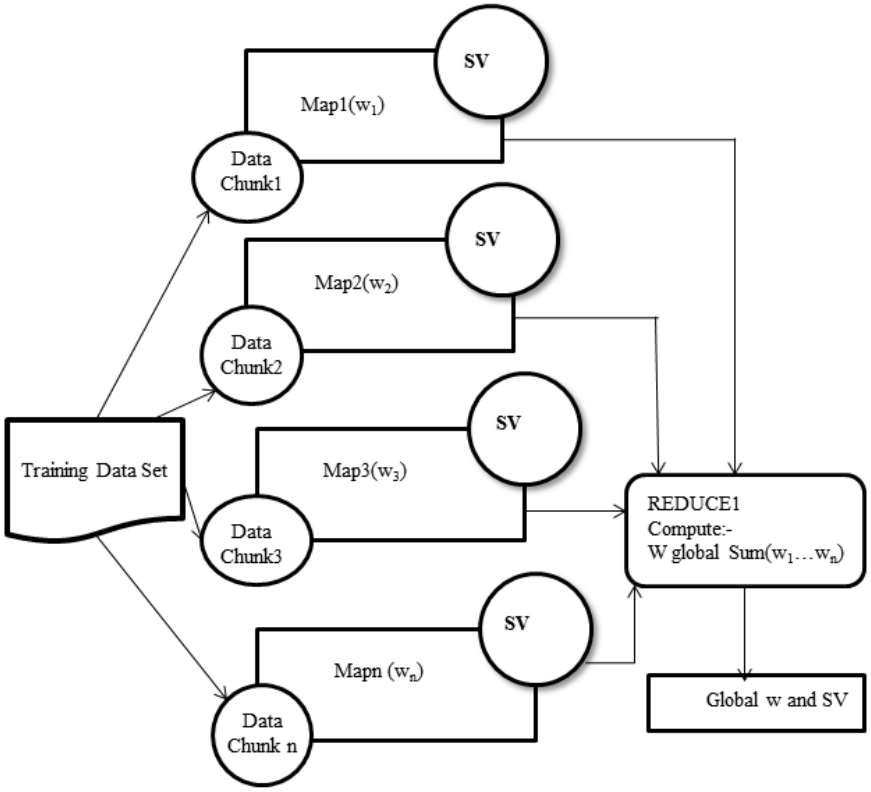
از طرفی رده‌بند SVM یکی از رده‌بندهای پرکاربرد و دارای کارایی بالا می‌باشد و با توجه به اینکه رده‌بند SVM در زمینه نظرکاوی نتایج و دقت بالاتری را داشته است، در پژوهش‌ها بیشتر مورد توجه قرار گرفته است.. در بسیاری از پژوهش‌های انجام شده، پژوهش‌گران برای افزایش سرعت و مقیاس-پذیری روش پیشنهادی، سعی در بهبود عملکرد رده‌بند SVM با استفاده از برنامه‌نویسی نگاشت-کاهش دارند. در ادامه به معرفی تعدادی از این پژوهش‌ها می‌پردازیم.

بو یان[1] و همکارانش به رده‌بندی احساسات نظرات کاربران بر روی چارچوب اسپارک[2] پرداخته‌اند[۳۴]. آنها ابتدا نظرات را پیش‌پردازش نموده و با استفاده از TF-IDF بردار ویژگی را ایجاد کرده و بعنوان ورودی رده‌بند در نظر گرفته‌اند. در ادامه نیز به آموزش رده‌بند SVM موازی همراه با هسته RBF با استفاده از برنامه‌نویسی نگاشت-کاهش نموده‌اند. در این مرجع موازی‌سازی در قسمت ساخت هسته RBF انجام شده و سایر قسمت‌ها همانند SVM استاندارد می‌باشد. در نهایت روش پیشنهادی را بر روی چند مجموعه‌داده متفاوت و با استفاده از چند گره پردازشی پیاده‌سازی کرده‌اند.

پوجاسومن تریپسی[3] و همکاران به تجزیه و تحلیل ریسک[4] با استفاده از SVM موازی بر روی چارچوب هدوپ[5] پرداخته‌اند [۳۵]. در روش پیشنهادی مجموعه‌داده به چند قسمت تقسیم شده و برای پردازش-های اولیه در اختیار گره‌ها قرار می‌گیرد. در ادامه هر قسمت از زیرمجموعه‌داده اولیه برای آموزش رده‌بند SVM استفاده و بردارهای پشتیبان آموزش دیده[6]، توسط نگارنده‌ها[7] به کاهنده‌ها[8] فرستاده می‌شود. در ادامه تمامی بردارهای پشتیبان به کاهنده‌ها ارسال می‌شود و توسط آنها ترکیب و بردارهای پشتیبان نهایی و رده‌بند نهایی آماده می‌شود. در پژوهش دیگری [۳۶] نیز ابتدا توضیحی کوتاه در مورد SVM آورده شده، در ادامه اشاره‌ای به ساختار SVM موازی، مروری بر برنامه‌نویسی نگاشت-کاهش و نگاهی

[1] Bo Yan
[2] Spark
[3] Pujasuman Tripathy
[4] Risk analysis
[5] Hadoop
[6] Trained Support Vectors
[7] Mappers
[8] Reducers

به چارچوب هدوپ داشته است. در نهایت به ارائه شبه‌کد SVM موازی مبتنی بر نگاشت-کاهش مشابه مرجع قبل داشته است. در شکل ۳-۱ ساختار روش ارائه شده در مراجع [۳۵, ۳۶] نشان داده شده است.

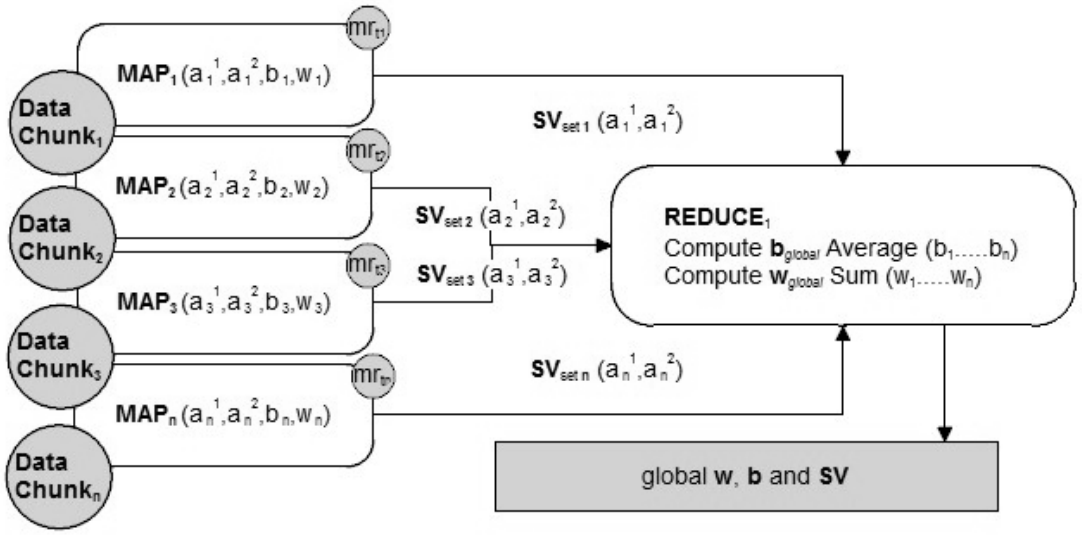


شکل ۳-۱: ساختار SVM موازی ارائه شده در مراجع [۳۵, ۳۶]

گادوین کاروآنا[1] و همکارانش [۳۷] روشی برای فیلتر کردن هرزنامه[2] با استفاده از SVM موازی مبتنی بر مدل نگاشت-کاهش ارائه کرده‌اند که پیاده‌سازی آن بر روی هدوپ انجام شده است. پس از تقسیم داده‌ها به چند بخش، هر بخش به یک نگارنده ارسال می‌شود و با استفاده از الگوریتم SMO، بردارهای پشتیبان (مقادیر b و w) برای هر بخش محاسبه می‌شود و این بردارهای پشتیبان برای بدست آوردن بردار پشتیبان کلی به کاهنده(ها) فرستاده می‌شود. در کاهنده میانگین مقادیر b بعنوان b کلی و

[1] Godwin Caruana
[2] Spam

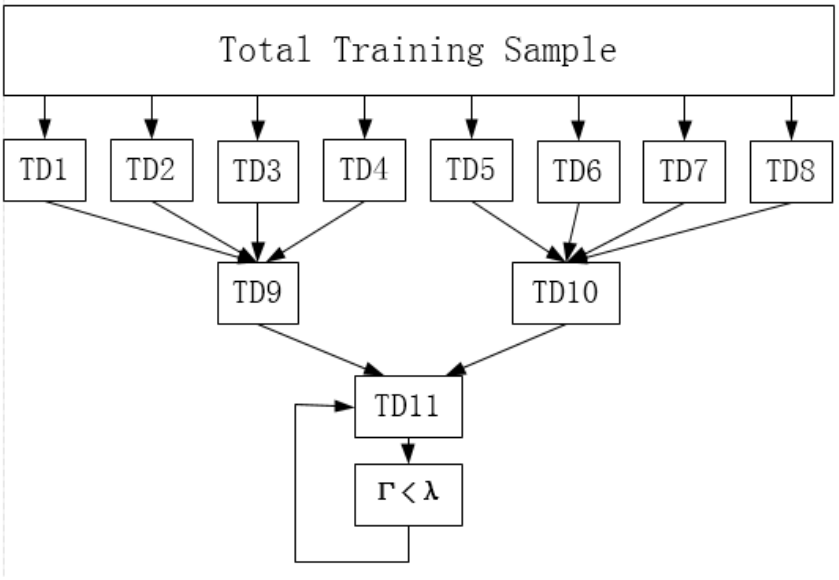
مجموع مقادیر w بعنوان w کلی در نظر گرفته شده است. پس از بدست آوردن مدل رده‌بند از آن برای تشخیص هرزنامه استفاده می‌گردد. ساختار و معماری روش ارائه شده در مرجع [۳۷] در شکل ۳-۲ نمایش داده شده است.



شکل ۳-۲: ساختار SVM موازی استفاده شده در مرجع [۳۷]

وی نی[1] و همکارانش [۳۸] سعی در بهینه سازی رده‌بند SVM موازی با استفاده از چند تابع هسته داشته‌اند. ابتدا اشاره ای به SVM موازی آبشاری[2] و SVM گروهی[3] داشته‌اند و در روش خود، ترکیبی از این دو مدل استفاده کرده‌اند. همچنین برای تابع هسته مدل پیشنهادی خود از ترکیب تابع هسته چندجمله‌ای و RBF بهره برده‌اند. در SVM استفاده شده در لایه آخر از شرط توقف λ (تعداد بردارهای پشتیبان مرحله کنونی تعداد بردارهای پشتیبان مرحله قبلی) برای اطمینان از همگرایی الگوریتم بهره گرفته شده است. ساختار SVM پیشنهادی مرجع [۳۸] را در شکل ۳-۳ می‌توانید مشاهده نمایید.

[1] Wei Nie
[2] Cascade-PSVM
[3] Grouped-SVM



شکل ۳-۳: ساختار SVM موازی ارائه شده در مرجع [۳۸]

پس از معرفی پژوهش‌هایی که برای موازی‌سازی SVM از نگاشت-کاهش استفاده نموده‌اند، حال به معرفی تعداد دیگری از پژوهش‌ها می‌پردازیم. در این پژوهش‌ها سعی شده است تا ساختار و معماری جدیدی برای رده‌بند SVM معرفی گردد.

۳ـ۳ـ۱ ماشین بردار پشتیبان آبشاری[1]

در این روش مجموعه‌داده آموزشی به چند (توانی از دو) بخش تقسیم می‌شود و بر روی هر قسمت یک SVM مجزا اعمال می‌شود. خروجی SVMها (بردارهای پشتیبان) دو به دو با هم ترکیب شده و بعنوان ورودی SVM مرحله بعد در نظر گرفته می‌شوند. این روش از مرجع [۳۹] اقتباس شده است که برای رده‌بندی تصاویر و خارج از محیط اسپارک پیاده‌سازی شده است. در این مرجع از نظر ریاضی نیز همگرایی روش اثبات شده است. برای فرموله کردن این روش ابتدا پارامترهای این روش معرفی می‌شود. n تعداد بخش‌های مجموعه‌داده، j سطح j ام در SVM آبشاری، m مدل آموزش‌دیده برای هر بخش مجموعه‌داده، sv مجموعه‌ی برداهای پشتیبان و L نیز سطح پایانی در SVM آبشاری می‌باشد.

[1] Cascade SVM

روش ماشین بردار پشتیبان آبشاری به صورت زیر فرموله می‌شود:

مدل‌های آموزش‌دیده سطح j
$$m_1^j, m_2^j, \ldots, m_i^j, \ldots m_n^j$$
بردارهای پشتیبان خروجی مدل‌های آموزش‌دیده سطح j
$$\Rightarrow sv_1^j, sv_2^j, \ldots, sv_i^j, \ldots sv_n^j$$
بردارهای پشتیبان سطح j، ورودی مدل‌های سطح j+1
$$\{sv_1^j, sv_2^j\}, \ldots, \{sv_{2*i-1}^j, sv_{2*i}^j\}, \ldots, \{sv_{n-1}^j, sv_n^j\} \rightarrow$$
مدل‌های آموزش‌دیده سطح j+1
$$m_1^{j+1}, m_2^{j+1}, \ldots, m_i^{j+1}, \ldots m_{n/2}^{j+1}$$
بردارهای پشتیبان خروجی مدل‌های آموزش‌دیده سطح j+1
$$\Rightarrow sv_1^{j+1}, sv_2^{j+1}, \ldots, sv_i^{j+1}, \ldots sv_{n/2}^{j+1}$$
...
در نهایت مدل آموزش‌دیده نهایی
$$M = m_1^{n+1}$$

۳ـ۳ـ۲ ماشین بردار پشتیبان آبشاری بهبودیافته[1]

در این روش مجموعه‌داده آموزشی به چند بخش تقسیم می‌شود و بر روی هر قسمت یک SVM مجزا اعمال می‌شود. تفاوت روش SVM آبشاری بهبودیافته با روش SVM آبشاری در این است که خروجی SVMهای مرحله اول، دو به دو ترکیب نمی‌شود بلکه خروجی تمام SVMهای مرحله اول باهم ترکیب شده و به تنها SVM مرحله بعدی (آخر) فرستاده می‌شود. این روش اقتباسی نیز در مرجع [۴۰] ارائه شده است و روش پیشنهادی در مرجع [۳۹] را بهبود داده و از این روش برای تشخیص نفوذ[2] استفاده کرده است. فرموله کردن این روش نیز به صورت زیر انجام می‌گیرد:

مدل‌های آموزش‌دیده سطح یک
$$m_1^1, m_2^1, \ldots, m_i^1, \ldots m_n^1$$
بردارهای پشتیبان خروجی مدل‌های آموزش‌دیده سطح یک
$$\Rightarrow sv_1^1, sv_2^1, \ldots, sv_i^1, \ldots sv_n^1$$
بردارهای پشتیبان سطح یک، ورودی مدل سطح دو
$$sv_1^1, sv_2^1, \ldots, sv_i^1, \ldots sv_n^1 \rightarrow m_1^2$$
در نهایت مدل آموزش‌دیده سطح دو، مدل نهایی
$$M = m_1^2$$

۳ـ۳ـ۳ ماشین بردار پشتیبان گروهی[3]

روش SVM گروهی تا حدودی مشابه با روش SVM آبشاری بهبود یافته می‌باشد، با این تفاوت که

[1] Improved Cascade SVM
[2] Intrusion Detection
[3] Grouped-SVM

SVM در مرحله بعدی وجود ندارد و خروجی SVMهای مرحله اول با یک‌دیگر ترکیب می‌شوند. خروجی مرحله آموزش برای هر SVM بردار w(ضرایب لاگرانژ) و ثابت b می‌باشد. w کلی ترکیب wهای مرحله قبل و b کلی، میانگین bهای مرحله قبل می‌باشد. این روش در [۴۱] معرفی گردیده است. در مرجع [۴۲] از این روش با استفاده از MapReduce برای تشخیص و رده بندی نوع فایل (فرمت فایل) استفاده کرده بود و در مرجع [۴۳] نیز از این روش برای تفسیر تصویر (Image Annotation) استفاده کرده بود.

فرموله بندی روش SVM گروهی به صورت زیر می‌باشد:

مدل‌های آموزش‌دیده سطح یک $m_1^1, m_2^1, \ldots, m_i^1, \ldots m_n^1$

بردارهای پشتیبان خروجی مدل‌های آموزش‌دیده سطح یک $Þ\ sv_1^1, sv_2^1, \ldots, sv_i^1, \ldots sv_n^1$

در نهایت مدل نهایی، اجتماع بردارهای پشتیبان سطح یک $M = \bigcup_{i=1}^{n} sv_i^1$

در رابطه بالا n تعداد بخش‌های مجموعه‌داده، m مدل آموزش‌دیده برای هر بخش مجموعه‌داده، sv مجموعه‌ی برداهای پشتیبان می‌باشد.

۳ـ۴ تجمیع پژوهش‌های پیشین

در این قسمت به بررسی و رده‌بندی کارهای مرتبط و پیشین می‌پردازیم. در جدول ۳-۱ نتایج این قسمت آورده شده است. در تمامی پژوهش‌ها از روش‌های مبتنی بر یادگیری ماشین استفاده شده است. معیارهایی همچون مقیاس‌پذیری روش، استفاده از محاسبات خوشه‌ای و همچنین کاربرد روش پیشنهادی در تحلیل احساسات در بررسی مراجع و پژوهش‌های پیشین مورد توجه قرار گرفته است. همچنین خلاصه‌ای از روش پیشنهادی نیز آورده شده است.

جدول ۳-۱ : بررسی و تجمیع کارهای مرتبط پیشین

	مقیاس‌پذیری	محاسبات خوشه‌ای	تحلیل احساسات	روش مورد استفاده
مرجع [۲۵]	خیر	خیر	بله	استفاده و مقایسه SVM و ANN
مرجع [۱۲-۱۴]	خیر	خیر	بله	ترکیب روش های مبتنی یادگیری ماشین و واژه نامه
مرجع [۳۰]	بله	بله	بله	بهره‌گیری از دو فاز برون و دورن-خطی برای غلبه بر پردازش داده‌های کلان و پیاده‌سازی در استورم[1]
مرجع [۳۱]	بله	خیر	بله	ارائه روشی به نام SSSA و استفاده از SentiWN و SVM
مرجع [۳۲]	بله	خیر	بله	ارائه تعریف جدیدی به نام فراویژگی و امتیازدهی به کلمات و جملات
مرجع [۳۳]	بله	بله	بله	استفاده از رده‌بند آموزش دیده برای تحلیل نظرات کرابران در طی مسابقه فوتبال، پیاده‌سازی در IBM InfoSphere
مرجع [۳۴]	بله	بله	بله	موازی‌سازی فرآیند تشکیل هسته در مرحله آموزش SVM، پیاده‌سازی در Spark
مرجع [۳۵-۳۷]	بله	بله	خیر	موازی‌سازی SVM مبتنی بر MapReduce و پیاده‌سازی در محیط Hadoop

[1] Apache Storm

مرجع [۳۸]	بله	بله	خیر	استفاده از ترکیب چند هسته برای آموزش SVM و ا ستفاده از SVM آبشاری و گروهی، پیاده‌سازی در Hadoop
مرجع [۳۹]	خیر	خیر	خیر	معرفی و استفاده از SVM آبشاری
مرجع [۴۰]	خیر	خیر	خیر	بهبود روش SVM آبشاری
مرجع [۴۱]	خیر	خیر	خیر	معرفی و استفاده از SVM گروهی
مرجع [۴۲]	بله	بله	خیر	استفاده از SVM گروهی برای رده‌بندی نوع داده
مرجع [۴۳]	بله	بله	خیر	استفاده از SVM گروهی برای تفسیر تصویر
مرجع [۴۴]	بله	بله	بله	موازی‌سازی SVM و پیاده‌سازی در محیط Cloudera
مرجع [۴۵]	بله	بله	بله	ارا ئه الگوریتمی به نام STING و پیاده‌سازی موازی آن در Cloudera
مرجع [۴۶]	بله	بله	بله	استفاده از گرامر وابسته به مجموعه‌داده برای ا ستخراج ویژگی و استفاده از SVM، NB و Logistic Regression، پیاده-سازی در Spark
مرجع [۴۷]	بله	خیر	بله	استفاده از شبکه عصبی CNN و ترکیب با روش‌های دیگر
مرجع [۴۸]	بله	بله	بله	موازی‌سازی الگوریتم Fuzzy C-Mean، پیاده‌سازی در محیط

				Cloudera
مرجع [۴۹]	خیر	خیر	بله	ارائـه روشـــی برای پیش‌پردازش و مقایســه آن با اســتفاده از SVM، NBو ANN
مرجع [۵۰]	خیر	خیر	بله	استفاده از شبکه عصبی بازگشتی و همچنین ترکیـب مجموعـه‌داده‌هـا برای مرحله آموزش
مرجع [۵۱]	خیر	خیر	بله	اســتفاده از یادگیری عمیق و شــبکه عصبی کانولوشن
مرجع [۵۲]	خیر	خیر	بله	اســتفاده از Unigram و Bigram برای اســـتخراج ویژگی، همچنین اســـتفاده از رده‌بند های SVM و NB و ترکیب هر دو

فصل چهارم

روش پیشنهادی

۴ـ۱ مقدمه

حجم زیاد اطلاعات در دنیای امروز برای پردازش و تحلیل، نیازمند روش‌های سریع و کارآمد می‌باشد. نظرات کاربرانی که در فضای اینترنت منتشر می‌شود نیز از این قاعده مستثنی نیست و رده‌بندی آنها محتاج روش‌های پرسرعت و در عین حال دقیق می‌باشد.

هدف این پژوهش نیز ارائه روشی جهت بهبود سرعت رده‌بندی نظرات کاربران در کنار حفظ دقت بالا می‌باشد. در این پژوهش چندین روش مبتنی بر SVM پیشنهاد شده است. این روش‌ها بر روی چارچوب اسپارک پیاده‌سازی شده اند. در ادامه به توضیح بیشتر این روش‌ها می‌پردازیم.

۴ـ۲ روش پیشنهادی

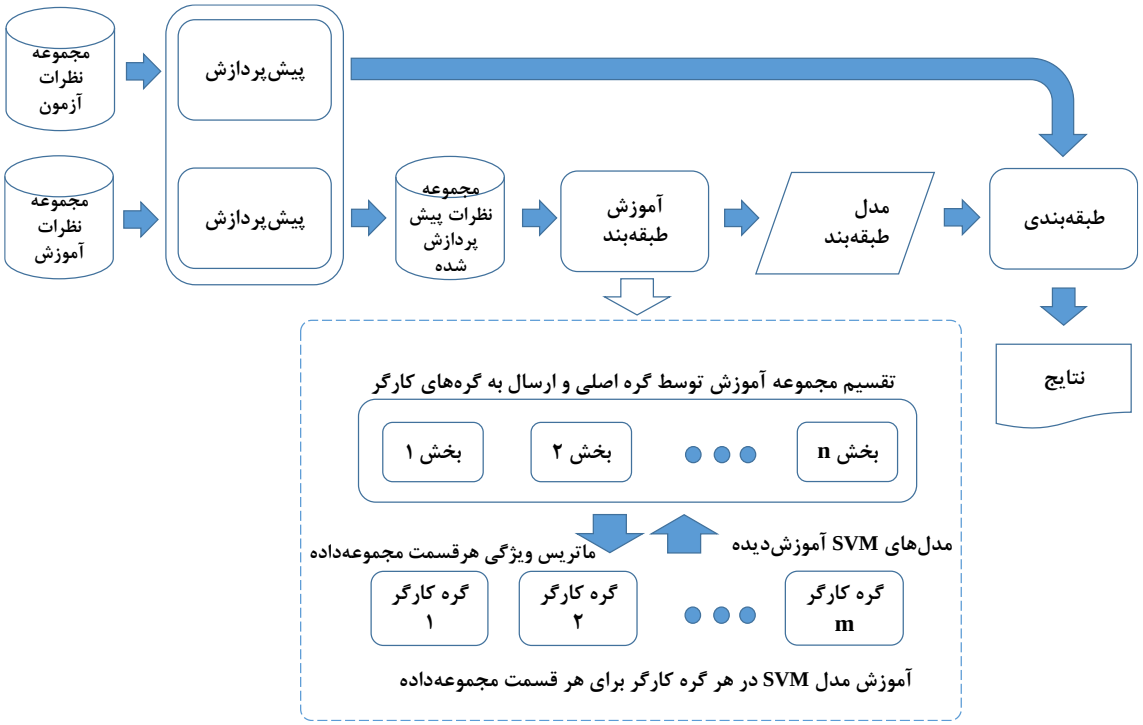
در این پژوهش روشی برای رده‌بندی نظرات کاربران به دو دسته مثبت و منفی بر روی مجموعه‌داده IMDB ارائه شده است. پس از انجام مراحل پیش‌پردازش به آموزش مدل رده‌بند پرداخته می‌شود و در ادامه از مدل آموزش‌دیده برای رده‌بندی استفاده می‌شود. پیاده‌سازی روش پیشنهادی در محیط اسپارک انجام شده است. در شکل ۴-۱ معماری و ساختار روش پیاده‌سازی شده آورده شده است.

۴ـ۲ـ۱ پیش‌پردازش

در این پژوهش مراحل پیش‌پردازشی زیادی انجام نشده است که این موضوع به سادگی پیاده‌سازی و سرعت آن کمک می‌نماید. ابتدا ایست‌واژه‌ها[1] از نظرات حذف می‌شوند. پس از آن برداری شامل تمام کلمات مجموعه‌داده آموزش تشکیل می‌شود. کلماتی که تعداد تکرار آن‌ها در تمام مجموعه نظرات کمتر از ۷۰ مرتبه می‌باشد نیز در نظر گرفته نمی‌شوند. در نهایت با توجه به کلمات موجود در هر نظر و تعداد تکرار آن، بردار ویژگی برای هر نظر، تشکیل می‌شود. تقریبا این نوع پیش‌پردازش به دلیل عدم استفاده از نقش کلمات در جمله (و در صورتی که کلمات پرتکرار نادیده گرفته شوند) می‌تواند برای زبان‌های

[1] Stopword

نیز استفاده شود. این قسمت در نرم‌افزار متلب[1] پیاده‌سازی شده است.



شکل ۴-۱: معماری کلی روش پیشنهادی

۴ـ۲ـ۲ آموزش مدل رده‌بند

در این مرحله با استفاده از ماتریس ویژگی به آموزش مدل رده‌بند پرداخته می‌شود. در این پژوهش، از رده‌بند SVM برای رده‌بندی دسته نظرات کاربران استفاده کرده‌ایم. این رده‌بند در حوزه متن‌کاوی و نظرکاوی بسیار موثر و کارا عمل می‌نماید. بنابراین در این پژوهش از رده‌بند SVM استفاده گردد. برای آموزش رده‌بند از روش‌های SVM استاندارد، آبشاری، آبشاری بهبودیافته، گروهی و SVM مبتنی بر رأی‌گیری برای آموزش مدل به‌صورت مستقل استفاده شده است. در ادامه به طور مفصل‌تر به معرفی ویژگی‌های استفاده شده در هر یک از روش‌ها می‌پردازیم.

برای آموزش مدل رده‌بند SVM از روش آموزشی SMO استفاده کرده‌ایم و به دلیل زیاد بودن مولفه‌های بردار ویژگی، از هسته RBF استفاده نموده‌ایم. در مجموعه‌داده‌های بزرگ و در صورت استفاده از هسته،

[1] Matlab

قبل از اجرای SMO نیاز به تشکیل ماتریس هسته[1] می‌باشد که زمان زیادی از مرحله آموزش را نیازمند است. این ماتریس متقارن می‌باشد و می‌توان به راحتی با نیمی از زمان مورد نیاز، آن را ایجاد کرد. در مقالات مطالعه شده، مقاله‌ای که به این نکته اشاره نماید، مشاهده نشد.

۴ـ۲ـ۳ رده‌بندی

پس از مرحله آموزش مدل رده‌بند، مرحله آخر یعنی رده‌بندی قرار دارد. در این حالت نظر برای رده‌بندی و تعیین قطبیت به مدل آموزش‌دیده فرستاده می‌شود و خروجی پیش‌بینی بدست می‌آید. این قسمت برای تمام روش‌های SVM موازی به‌جز روش SVM مبتنی بر رأی‌گیری، توسط یک مدل آموزش‌دیده انجام می‌شود. در روش SVM مبتنی بر رأی‌گیری، چند مدل آموزش‌دیده وجود دارد و نظر توسط هر مدل پیش‌بینی و رده‌بندی شده و درنهایت رأی‌گیری بین پیش‌بینی مدل‌ها انجام می‌شود.

۴ـ۳ آموزش مدل رده‌بندی

برای رده‌بندی و آموزش مدل رده‌بند از روش SVM مبتنی بر رأی‌گیری و روش‌های SVM موازی آبشاری، آبشاری بهبودیافته، گروهی استفاده شده است که در ادامه به توضیح بیشتر این روش‌ها می‌پردازیم.

۴ـ۳ـ۱ ماشین بردار پشتیبان مبتنی بر رأی‌گیری

در این روش پیشنهادی پژوهش حاضر، مجموعه‌داده آموزش را به چند بخش (تعداد فرد) تقسیم نموده و از هر بخش برای آموزش یک مدل SVM استفاده می‌نماییم. پس از مرحله آموزش، از تمام SVMها برای رده‌بندی مجموعه‌داده آزمون استفاده می‌کنیم. برای این عمل، کل مجموعه‌داده آزمون توسط هر یک از مدل‌های SVM رده‌بندی می‌شود. خروجی رده بندی هر SVM برای هر نظر، دو دسته مثبت و منفی می‌باشد که به ترتیب با ۱ و ۱- مشخص می‌شوند. در نهایت با تجمیع خروجی SVMها برای هر

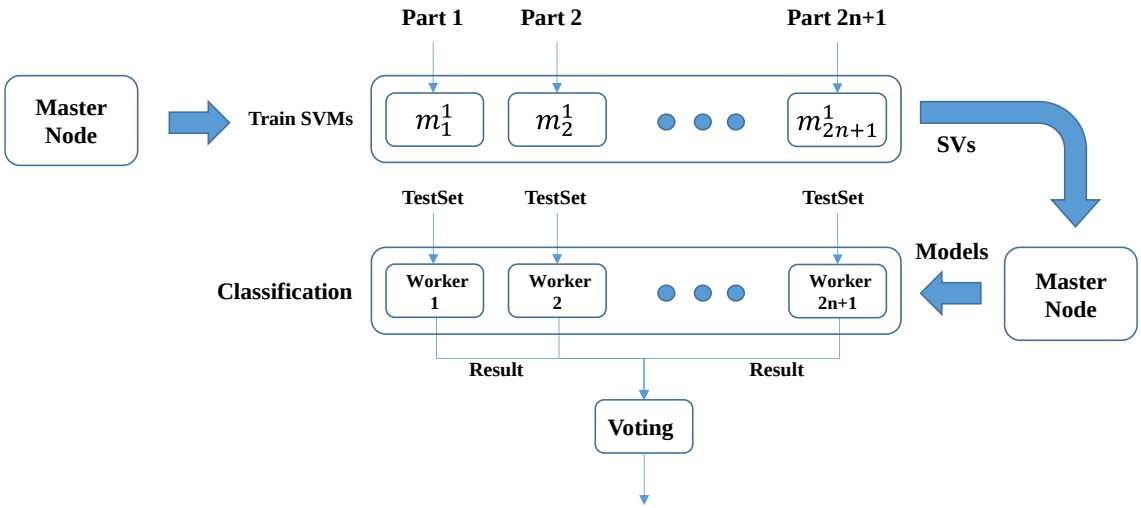
[1] Kernel Matrix

داده‌ی مجموعه آزمون، نتیجه نهایی رده‌بندی بدست می‌آید که رأی‌گیری بین SVMها می‌باشد. مدل پیشنهادی به صورت زیر فرموله می‌شود:

مدل‌های آموزش‌دیده سطح یک $m_1^1, m_2^1, \ldots, m_i^1, \ldots m_n^1$

در نهایت رده‌بندی نظر بر مبنای رأی‌گیری مدل‌های آموزش‌دیده سطح یک $\text{classify}\{M, o\} = \text{argmax}_i \sum_{m \in M} \{\text{classify}(m, o) = i\}$

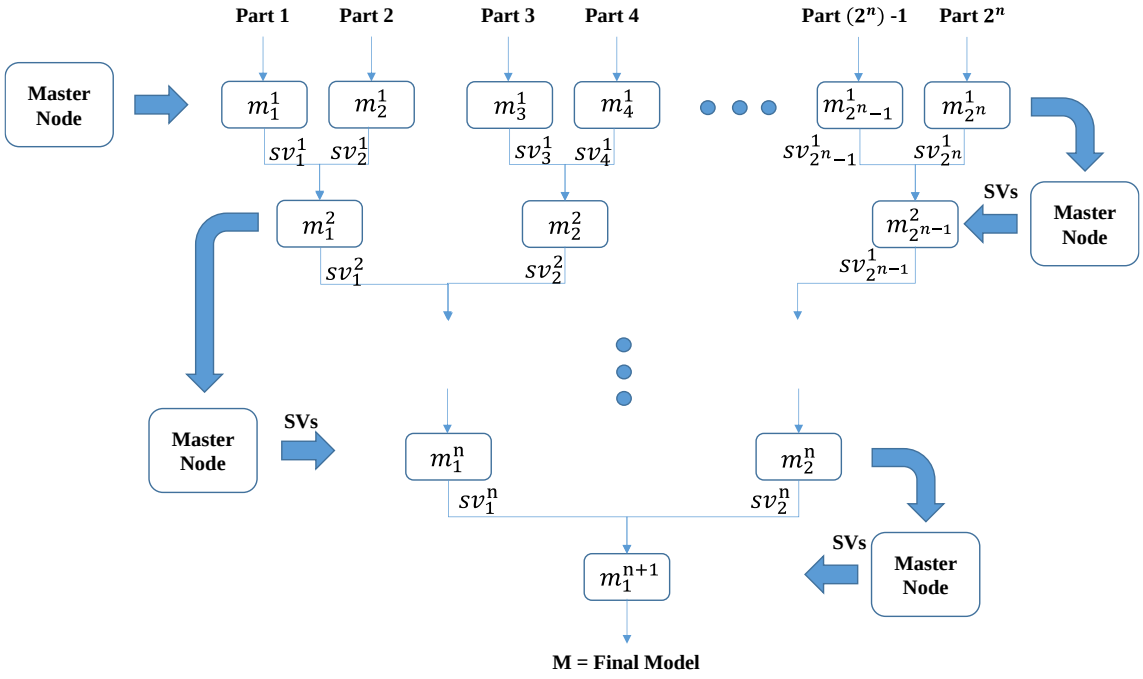
که در آن n تعداد بخش‌های مجموعه‌داده، o نظر ورودی برای رده‌بندی، M مدل آموزش‌دیده و m هر کدام از SVMهای آموزش‌دیده می‌باشند. معماری و ساختار روش SVM مبتنی بر رأی‌گیری در شکل ۴-۲ بر پایه معماری اسپارک، نمایش داده شده است.



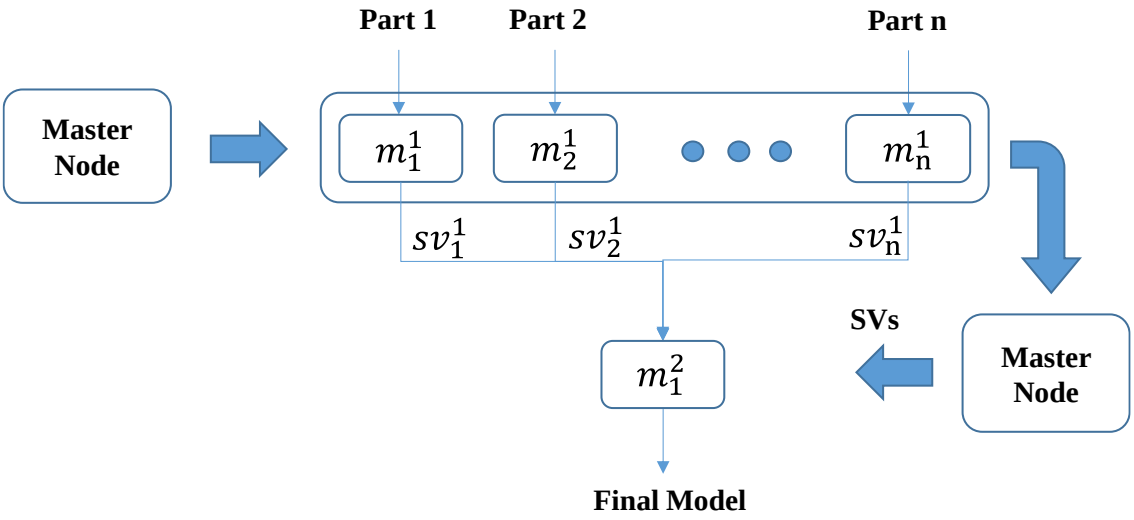
شکل ۴-۲: معماری SVM مبتنی بر رأی‌گیری

۴ـ۳ـ۲ ماشین بردار پشتیبان آبشاری

همانطور که در فصل قبل اشاره شد این روش توسط هانس پیتر گرفت و همکاران [۳۹] معرفی شده است. در پژوهش حاضر از این روش برای استفاده در حوزه نظرکاوی استفاده شده است. پیاده‌سازی این روش نیز در محیط اسپارک انجام شده است. در شکل ۴-۳ معماری و ساختار این روش مبتنی بر اسپارک، نشان داده شده است.



شکل ۴-۳: معماری SVM آبشاری



شکل ۴-۴: معماری SVM آبشاری بهبودیافته

۴ـ۳ـ۳ ماشین بردار پشتیبان آبشاری بهبودیافته

این روش توسط دوهانگلی و همکاران در سال ۲۰۰۹ معرفی شده است[۴۰]. دوهانگلی و همکاران سعی در بهبود روش SVM آبشاری داشته‌اند. در پژوهش حاضر، این روش را در محیط اسپارک پیاده‌سازی و همچنین برای کاربرد در حوزه نظرکاوی استفاده شده است. در شکل ۴-۴ ساختار روش ماشین بردار پشتیبان آبشاری بهبود یافته با توجه به معماری اسپارک را نشان می‌دهد.

۴ـ۳ـ۴ ماشین بردار پشتیبان آبشاری با پالایش بردار پشتیبان

این روش در پژوهش حاضر معرفی و استفاده شده است. ایده این روش سعی در کاهش داده‌های ورودی SVMهای مرحله دوم به بعد می‌باشد. در SVM آبشاری و آبشاری بهبودیافته، در هر مرحله بردارهای پشتیبان جدا شده و به عنوان ورودی SVM مرحله بعدی در نظر گرفته می‌شوند. بردارهای پشتیبان، نمونه‌هایی می‌باشند که برای آن‌ها ضرایب لاگرانژ خروجی از روش آموزشی SMO، بزرگتر از صفر باشند($0 < \alpha_i \le C$). با توجه به این موضوع می‌توانیم با تعیین حدآستانه برای انتخاب ضرایب لاگرانژ، بخشی از ضرایب لاگرانژی که بزرگ‌تر از حدآستانه باشند را به عنوان ورودی مرحله بعد در نظر بگیریم. با این عمل تعداد داده‌های ورودی SVM مرحله بعدی کاهش می‌یابد و در نتیجه باعث افزایش سرعت آموزش می‌شود. از طرفی دیگر ضرایب لاگرانژ با مقادیر بزرگ‌تر دارای اهمیت بیشتری بوده و با توجه به این موضوع از دقت نهایی نیز مقدار زیادی کاسته نخواهد شد. این روش به صورت زیر فرموله می‌شود:

مدل‌های آموزش‌دیده سطح یک
$m_1^1, m_2^1, \ldots, m_i^1, \ldots m_n^1$

بردارهای پشتیبان خروجی مدل‌های آموزش‌دیده سطح یک
$\Rightarrow sv_1^1, sv_2^1, \ldots, sv_i^1, \ldots sv_n^1$

بردارهای پشتیبان سطح یک با ضرایب لاگرانژ بزرگتر از C/2، بعنوان ورودی مدل سطح دو
$\bigcup_{i=1}^{n} \left\{ sv_i^1 \mid \alpha_i > \frac{C}{2} \right\}$

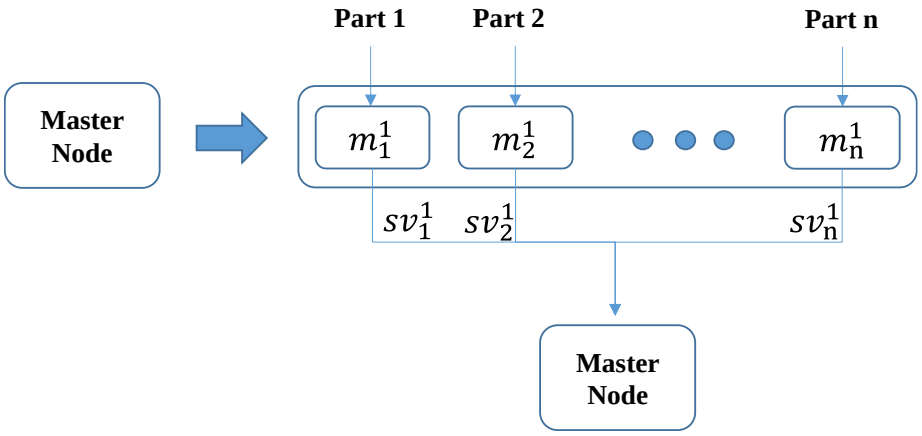
در نهایت مدل آموزش‌دیده سطح دو، مدل نهایی
$\rightarrow m_1^2$

$M = m_1^2$

۴ـ۳ـ۵ ماشین بردار پشتیبان گروهی

همانظور که در فصل سوم اشاره شد، این روش توسط دانگ و همکاران در سال ۲۰۰۳ معرفی گردیده

است[۴۱]. در پژوهش حاضر، از این روش نیز برای استفاده در حوزه نظرکاوی بهره گرفته شده است. پیاده‌سازی روش ذکر شده، در محیط اسپارک انجام شده است. شکل ۴-۵ مربوط به ساختار و معماری این روش مبتنی بر چارچوب اسپارک می‌باشد.



شکل ۴-۵: معماری SVM گروهی

۴ـ۳ـ۶ مرتبه زمانی

مرتبه زمانی رده‌بند SVM غیرخطی (استفاده از هسته) با توجه به کم یا زیاد بودن مقدار پارامتر C، بین دو مقدار $O(N^2)$ و $O(N^3)$ می‌باشد که در آن N تعداد داده‌های ورودی می‌باشد [۵۳]. با این شرایط مرتبه زمانی برای SVM آبشاری (برای حالتی که مجموعه‌داده به چهار بخش تقسیم شده باشد) در کمترین حالت برابر با :

$$O(4*(\frac{N}{4})^2+(|sv_1^1|+|sv_2^1|)^2+(|sv_3^1|+|sv_4^1|)^2+(|sv_1^2|+|sv_2^2|)^2) \quad (۱-۴)$$

و در بیشترین حالت برابر با:

$$O(4*(\frac{N}{4})^3+(|sv_1^1|+|sv_2^1|)^3+(|sv_3^1|+|sv_4^1|)^3+(|sv_1^2|+|sv_2^2|)^3) \quad (۲-۴)$$

می‌باشد. مرتبه زمانی برای SVM آبشاری بهبودیافته در کمترین حالت برابر با:

$$O(n*(\frac{N}{n})^2+(|sv_1^1|+|sv_2^1|+...+|sv_n^1|)^2) \quad (۳-۴)$$

و در بیشترین حالت برابر با رابطه زیر می‌باشد:

$$O(n*(\frac{N}{n})^3 + (|sv_1^1| + |sv_2^1| + ... + |sv_n^1|)^3) \quad (۴-۴)$$

مرتبه زمانی برای SVM گروهی نیز در کمترین حالت $O(n*(\frac{N}{n})^2)$ و بیشترین حالت $O(n*(\frac{N}{n})^3)$ می‌باشد. مرتبه زمانی برای SVM مبتنی بر رأی‌گیری نیز مشابه حالت گروهی می‌باشد.

۴ـ۴ سایر روش‌های انجام شده

در این پژوهش دو روش دیگر نیز مورد بررسی قرار گرفت اما به دلیل نامطلوب بودن نتایج بدست آمده از بررسی بیشتر و ذکر نتایج خودداری شد.

- روش اول: ابتدا مجموعه‌داده به چند بخش تقسیم می‌شود، سپس آموزش SVM توسط بخشی از مجموعه‌داده (برای مثال بخش اول) انجام شده و در ادامه مابقی مجموعه‌داده (برای مثال بخش دوم) با مدل SVM آموزش دیده آزمایش می‌شود، در نهایت بردارهای پشتیبان و همچنین داده‌های اشتباه پیش‌بینی شده برای آموزش SVM مرحله بعد ارسال می‌شود.
- روش دوم: ابتدا مجموعه‌داده به چند بخش تقسیم می‌شود، سپس آموزش SVM توسط بخشی از مجموعه‌داده (برای مثال بخش اول) انجام می‌شود. سپس فاصله (Cosine و RBF) مابقی داده‌ها با بردارهای پشتیبان بدست آمده در مرحله اول تعیین می‌گردد، در نهایت بردارهای پشتیبان و داده‌های دارای کمترین فاصله (بیشترین شباهت) برای آموزش SVM مرحله بعد فرستاده می‌شود.

فصل پنجم

نتایج و ارزیابی

۵ـ۱ مقدمه

در این فصل نتایج حاصل از اجرای الگوریتم‌های SVM موازی در محیط اسپارک آورده شده و با نتایج سایر مقالات مقایسه و ارزیابی می‌شود. در این پژوهش دقت و زمان آموزش برای بررسی مورد استفاده قرار گرفته است. در ادامه نیز نتایج مربوط به هر مدل از SVM به صورت مجزا آمده است.

۵ـ۲ مجموعه داده

در این پژوهش ما از مجموعه‌داده IMDB برای بررسی و ارزیابی نتایج استفاده کرده‌ایم. این مجموعه‌داده شامل ۲۵۰۰۰ نظر برای آموزش و ۲۵۰۰۰ نظر برای آزمون می‌باشد. نظرات در مورد فیلم و به زبان انگلیسی می‌باشند. در هر قسمت مجموعه‌داده آموزش و آزمون ۱۲۵۰۰ نظر مثبت و ۱۲۵۰۰ نظر منفی وجود دارد. این مجموعه‌داده در پژوهش‌های دیگری نیز مورد استفاده قرار گرفته است و به همین دلیل برای مقایسه با سایر مقالات نیز مناسب می‌باشد.

۵ـ۳ مدل پیاده‌سازی

برای پیاده‌سازی این پژوهش از چارچوب اسپارک نسخه ۲,۲,۱ استفاده شده است. مشخصات سیستم رایانه به این شرح می‌باشد:

CPU: Intel(R) Xeon(R) CPU E5-2660 @ 2.20GHz

Cashe: 3MB

RAM: 32GB DDR3

برای پیش‌پردازش نیز از نسخه متلب ۲۰۱۴ استفاده شده است. پارامترهای مربوط به اجرای روش پیشنهادی به شرح جدول ۵-۱ می‌باشد.

جدول ۵-۱: شرح پارامترهای موجود در روش پیشنهادی

پارامتر	مقدار
n	$1-25$
C	$1-48$ (6)
σ	$1-44$ (32)

۵ـ۴ معیارهای ارزیابی

برای ارزیابی روش‌های پیاده‌سازی شده از معیارهای زیر استفاده شده است. اولین مورد معیار زمان آموزش و زمان آزمون بر حسب ثانیه می‌باشد. معیارهای دیگری که برای ارزیابی عملکرد تشخیص قطبیت نظر استفاده شده است در ادامه آورده شده است:

معیار دقت (ACC):

$$\text{Accuracy} = \frac{\text{True Positive Predictions} + \text{True Negative Predictions}}{\text{Total Predictions}} \quad (۱-۵)$$

معیار تشخیص (SPC) یا نرخ پیش‌بینی‌های صحیح منفی[1] (TNR):

$$\text{Specificity} = \frac{\text{Number of True Negatives(TN)}}{\text{Number of True Negatives(TN)} + \text{Number of False Positives(FP)}} \quad (۲-۵)$$

معیار حساسیت یا نرخ پیش‌بینی‌های صحیح مثبت[2] (TPR) یا Recall:

$$\text{Sensitivity} = \frac{\text{Number of True Positives(TP)}}{\text{Number of True Positives(TP)} + \text{Number of False Negatives(FN)}} \quad (۳-۵)$$

[1] True Negative Rate
[2] True Pasitive Rate

۵ـ۵ نتایج

۵ـ۵ـ۱ ماشین بردار پشتیبان مبتنی بر رأی‌گیری

در این بخش نتایج مربوط به اجرای روش SVM مبتنی بر رأی‌گیری آورده و به بررسی آن پرداخته شده است. در جدول ۵-۲ نتایج مربوط به این قسمت آورده شده است. در جدول ۵-۳ مقادیر پارامترهای σ و C برای هر سطر متفاوت می‌باشد و مقادیری که بهترین دقت را بدست آورده‌اند، گزارش شده است. بیشینه زمان برای مرحله آموزش، بیشترین زمان سپری شده برای آموزش بین SVMهای موجود می‌باشد.

جدول ۵-۲: نتایج مربوط به SVM مبتنی بر رأی‌گیری و استاندارد

مجموعه‌داده IMDB	دقت(ACC)	حساسیت(TPR)	تشخیص(TNR)	بیشینه زمان آموزش	زمان آزمون	زمان کل
رأی‌گیری SVM– ۳ بخش	84.872%	0.8386	0.8541	340	4378	5381
رأی‌گیری SVM – ۵ بخش	84.040%	0.8311	0.8502	119	4396	4984
رأی‌گیری SVM– ۷ بخش	83.560%	0.8222	0.8500	63	4381	4797
رأی‌گیری SVM– ۹ بخش	82.964%	0.8158	0.8447	38	4383	4709
رأی‌گیری SVM– ۲۵ بخش	79.928%	0.7830	0.8175	6	4370	4496
SVM استاندارد	85.946%	0.8543	0.8658	3052	4385	7437

همان‌طور که مشخص می‌باشد زمان‌های آموزش نسبت به SVM استاندارد تا حدود ۹۸,۷٪ کاهش داشته است و دقت نیز کاهش داشته است اما تا حدودی با توجه به کاهش زمان، مطلوب است. این زمان گزارش شده برای حالتی‌ست که مجموعه‌داده به نه قسمت تقسیم شده است. در حالتی که مجموعه‌داده به ۲۵ قسمت تقسیم شده است زمان آموزش در حدود ۹۹,۸٪ درصد کاهش داشته است

که رقم قابل توجهی می‌باشد اما با این وجود، دقت ارائه شده دقت مطلوبی نیست و کاهش داشته است. در صورت استفاده از گره‌های کارگر[1] برای پردازش، زمان آموزش به میزان گزارش شده، کاهش می‌یابد. زمان مرحله آزمون نیز با توجه به تعداد گره‌های کارگر کاهش می‌یابد.

جدول ۵-۳: نتایج مربوط به SVM مبتنی بر رأی‌گیری با مقادیر بهینه و متفاوت σ و C

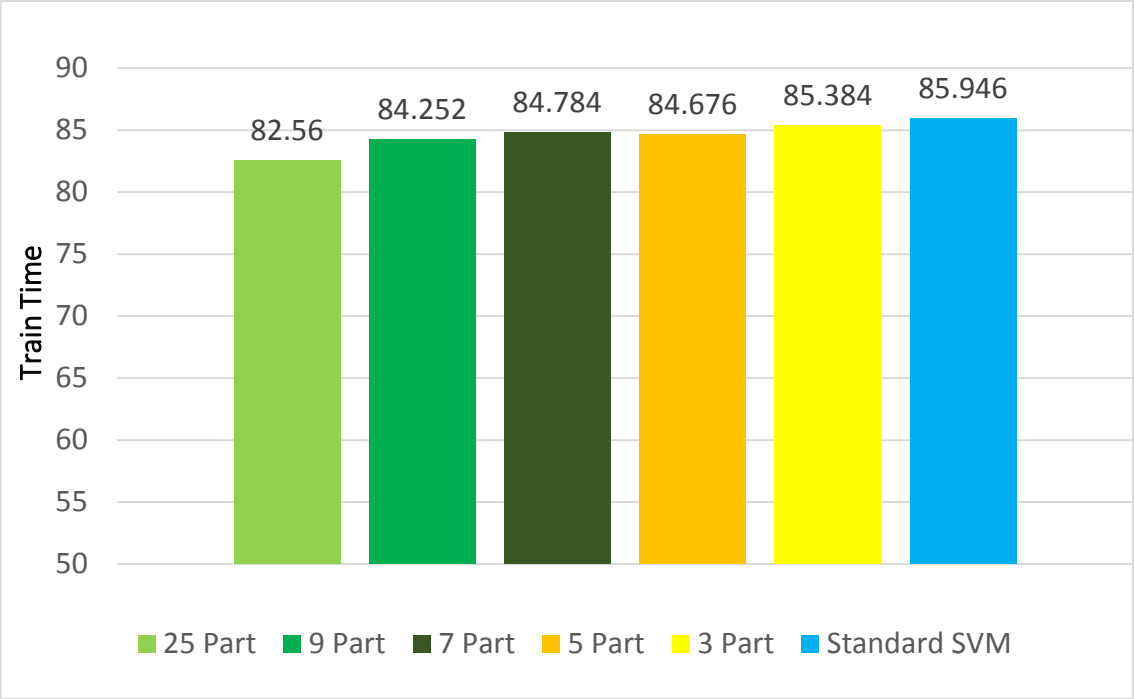
مجموعه‌داده IMDB	دقت(ACC)	حساسیت(TPR)	تشخیص(TNR)	بیشینه زمان آموزش	زمان آزمون	زمان کل
رأی‌گیری SVM– ۳ بخش	85.384%	0.8454	0.8626	433	4382	5663
رأی‌گیری SVM – ۵ بخش	84.676%	0.8381	0.8557	147	4396	5087
رأی‌گیری SVM– ۷ بخش	84.784%	0.8353	0.8613	92	4391	5018
رأی‌گیری SVM– ۹ بخش	84.252%	0.8297	0.8563	51	4373	4823
رأی‌گیری SVM– ۲۵ بخش	82.560%	0.8113	0.8412	9	4388	4579

در نتایج بالا نیز مشاهده می‌شود که زمان آموزش نسبت به نتایج جدول ۵-۲ کمی افزایش یافته است اما نسبت به SVM استاندارد کاهش قابل توجهی داشته است. این کاهش زمان در کنار این مورد که دقت نیز تا حدود زیادی حفظ شده است، ارزش بیشتری پیدا می‌کند.

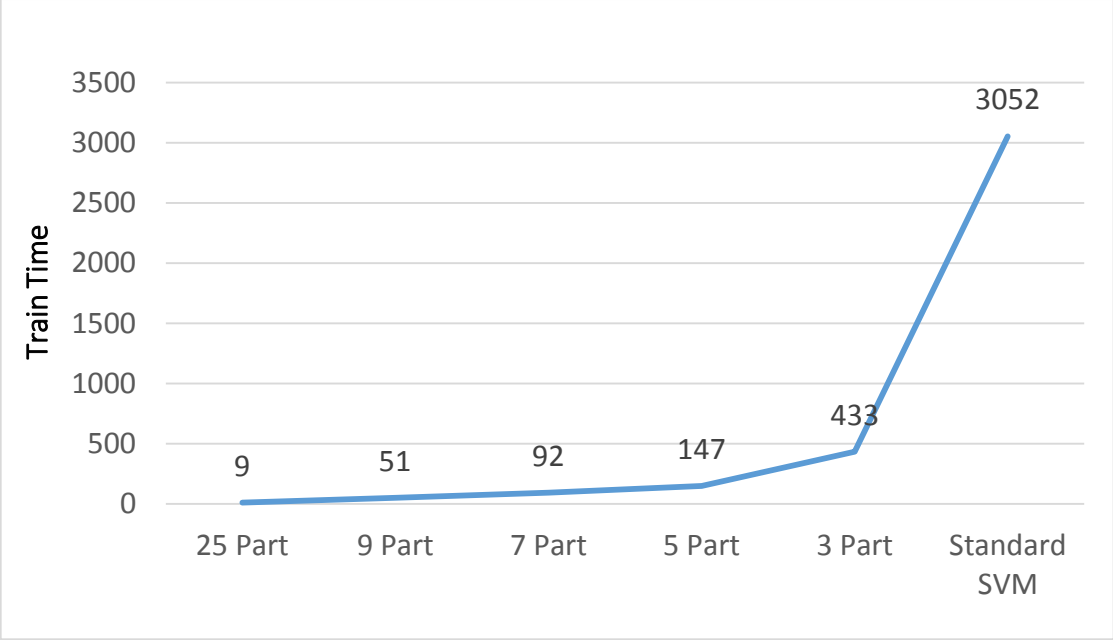
نمودار مربوط به نتایج روش SVM مبتنی بر رأی‌گیری در شکل ۵-۱ و شکل ۵-۲ نشان داده شده است.

همان‌طور که در این نمودارهای مشاهده می‌گردد زمان آموزش رده‌بند SVM مبتنی بر رأی‌گیری، نسبت به SVM استاندارد کاهش چشم‌گیری دارد؛ این مورد به دلیل تقسیم مجموعه‌داده به بخش‌های کوچک‌تر می‌باشد که همین عامل به دو دلیل باعث افزایش سرعت اجرای روش می‌شود. همان‌طور که

[1] Worker Nodes



شکل ۵-۱ : نتایج معیار دقت روش SVM مبتنی بر رأی‌گیری



شکل ۵-۲ : نتایج زمان روش SVM مبتنی بر رأی‌گیری

در فصل سوم اشاره شد، مرتبه زمانی رده‌بند SVM وابسته به اندازه مجموعه‌داده آموزش است و کاهش

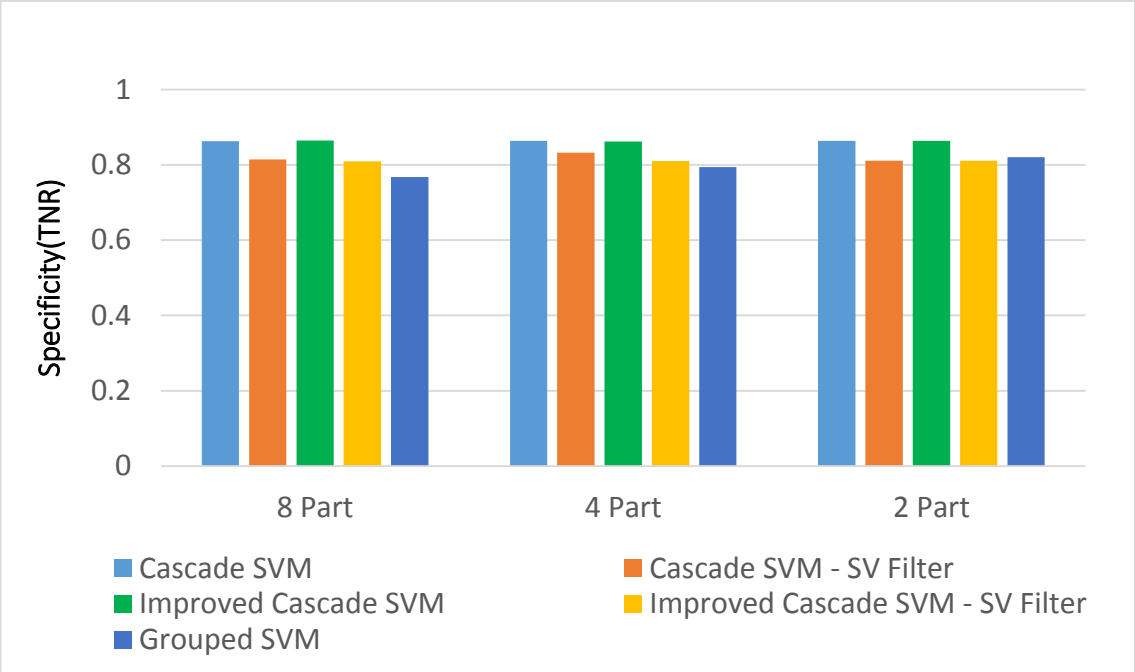
مجموعه‌داده آموزش باعث افزایش سرعت مرحله آموزش رده‌بند می‌گردد. همچنین با تقسیم مجموعه-داده به چند قسمت و استفاده از هر قسمت برای آموزش رده‌بند موجب می‌شود تا بتوان از پردازش موازی بهره برد. مزیت استفاده از الگوریتم SVM استاندارد نسبت به SVM مبتنی بر رأی‌گیری و سایر روش‌های مبتنی بر SVM که در آن مجموعه‌داده تقسیم می‌شود، استفاده از کل مجموعه‌داده برای آموزش رده‌بند می‌باشد. این موضوع باعث افزایش دقت رده‌بند می‌شود. در روش SVM مبتنی بر رأی‌گیری به دلیل این‌که از رأی‌گیری بین SVMها استفاده می‌شود، می‌توان گفت مشابه استفاده از تمام مجموعه‌داده برای آموزش رده‌بند است و به همین دلیل دقت در روش SVM مبتنی بر رأی‌گیری نیز تا حد مطلوبی حفظ شده است.

۵ـ۵ـ۲ مقایسه سایر مدل‌ها

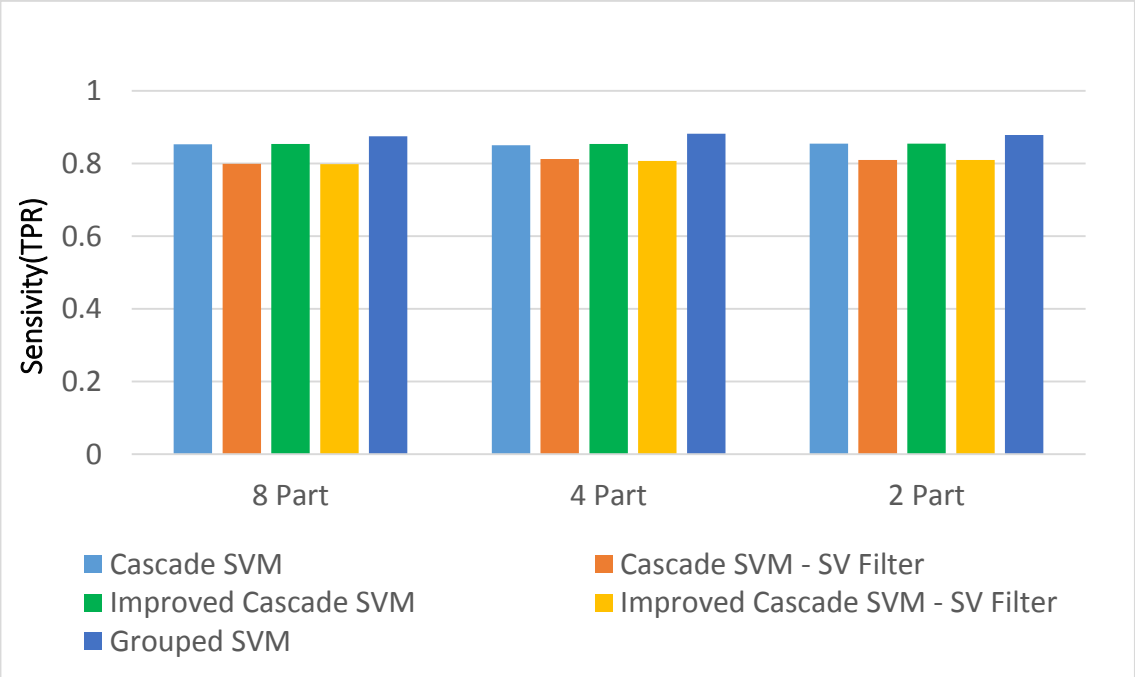
در این قسمت نتایج سایر روش‌های SVM آبشاری، آبشاری بهبودیافته، آبشاری با پالایش بردارهای پشتیبان و گروهی آورده شده و به مقایسه آن‌ها می‌پردازیم. برای ارزیابی، مجموعه‌داده به دو، چهار و هشت قسمت تقسیم شده است. در جدول ۵-۴ معیار دقت برای این روش‌ها گزارش شده است.

جدول ۵-۴: مقایسه معیار دقت(ACC) مدل‌ها

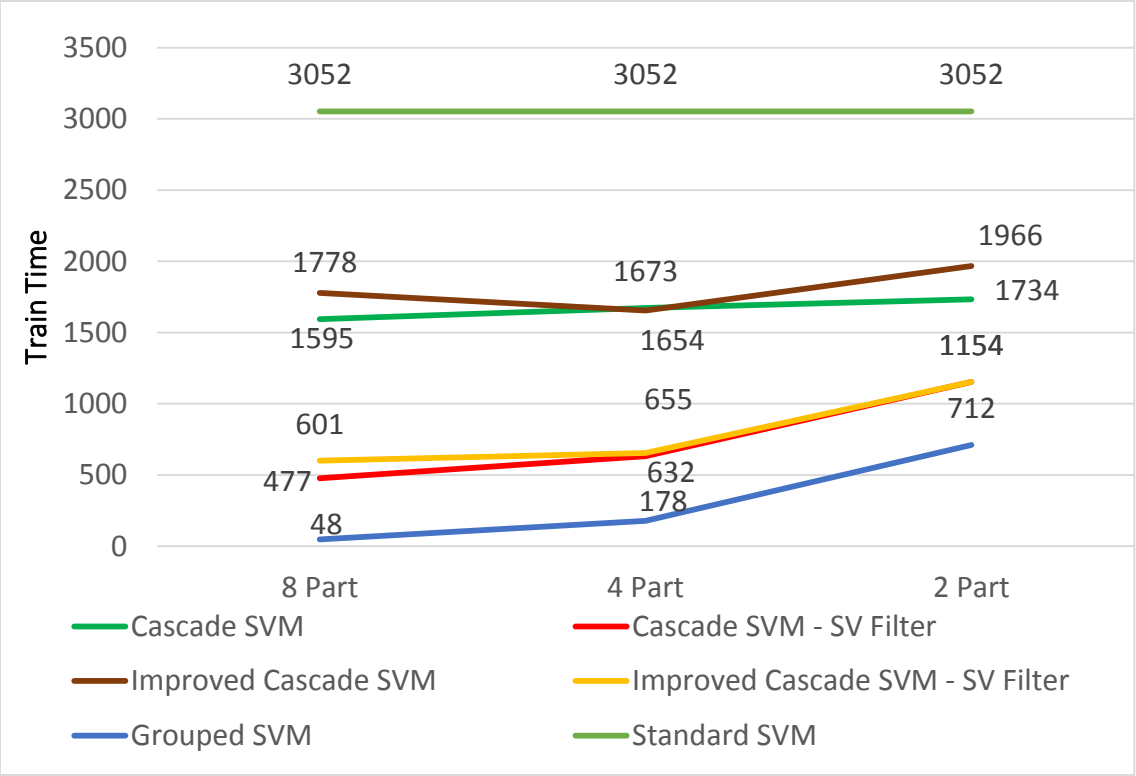
مدل SVM – دقت (ACC)	۲ بخش	۴ بخش	۸ بخش
آبشاری	85.972%	85.844%	85.816%
آبشاری بهبودیافته	85.972%	85.924%	85.916%
آبشاری با پالایش بردارهای پشتیبان	81.08%	82.248%	80.696%
آبشاری بهبودیافته با پالایش بردارهای پشتیبان	81.08%	80.9%	80.396%
گروهی	84.712%	83.276%	81.222%



شکل ۵-۳: مقایسه معیار تشخیص (SPC or TNR) مدل‌ها



شکل ۵-۴: مقایسه معیار حساسیت (TPR) مدل‌ها



شکل ۵-۵: مقایسه معیار زمان آموزش مدل‌ها

شکل ۵-۳ و شکل ۵-۴ به ترتیب مربوط به نتایج معیار تشخیص و حساسیت می‌باشند. همچنین در شکل ۵-۵ نیز زمان آموزش مدل رده‌بند روش‌های مدنظر آمده است. همانطور که مشخص است زمان آموزش برای SVM آبشاری با پالایش بردارهای پشتیبان و آبشاری بهبودیافته با پالایش بردارهای پشتیبان، نسبتا نزدیک است. این مورد برای SVM آبشاری و آبشاری بهبود یافته نیز صدق می‌نماید. زمان آموزش برای SVM گروهی نسبتا به سایرین کمتر می‌باشد زیرا نسبت به سایر روش‌ها تعداد زیررده‌بندهای کمتری مورد استفاده قرار می‌گیرد. پس از آن، روش‌های SVM آبشاری با پالایش بردارهای پشتیبان و آبشاری بهبودیافته با پالایش بردارهای پشتیبان زمان کمتری را نسبت به سایرین سپری کرده‌اند. با توجه به نتایج مشاهده می‌گردد روش‌هایی که زمان آموزش کمتری داشته‌اند، دارای دقت کمتری نیز هستند و بالعکس روش‌هایی که دقت بیشتری داشته‌اند، زمان مرحله آموزش رده‌بند آن‌ها بیشتر بوده است.

۵ـ۵ـ۳ ارزیابی پارامترهای مدل‌ها

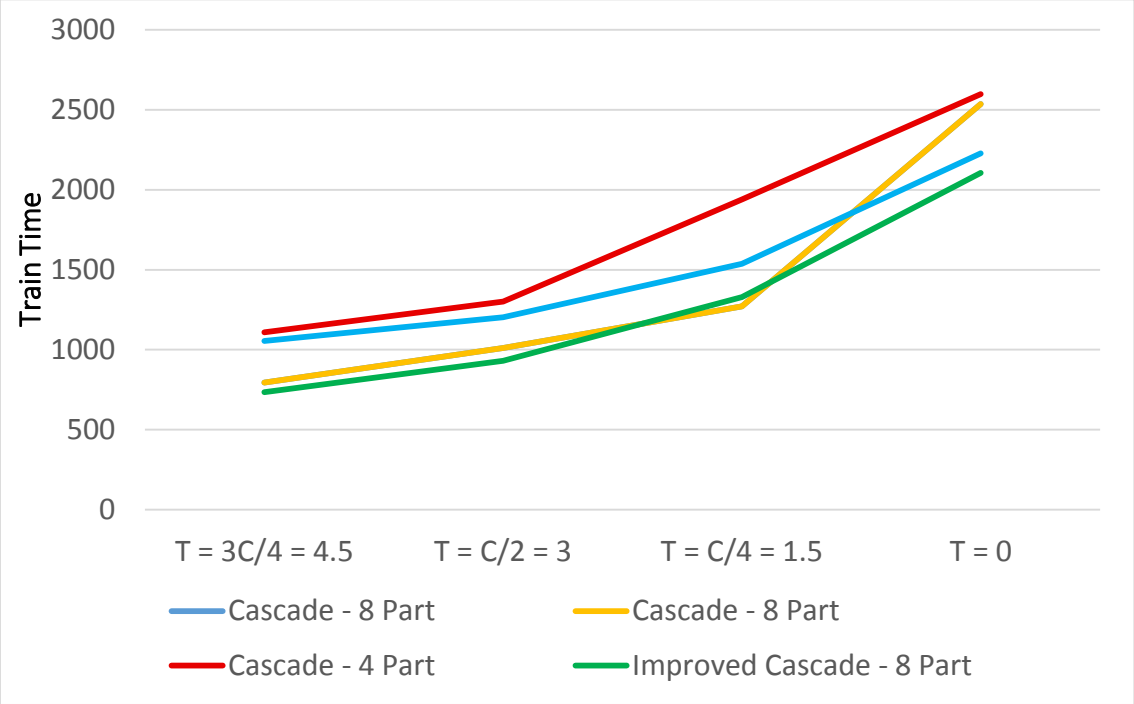
در ارزیابی مدل‌ها با روش‌های مختلف پارامترهای C و σ موثر می‌باشند. برای روش SVM استاندارد، مقادیر C = 6 و σ = 32 دارای بیشترین دقت بودند. برای انجام مقایسه، در سایر روش‌ها نیز نتایج با در نظر گرفتن این مقادیر بدست آورده شده است اما در مقادیر متفاوتی از C و σ، دقت خروجی بهبود می‌یابد. برای نمونه در روش SVM مبتنی بر رأی‌گیری نتایج با تغییر C و σ، بهبود یافت. نتایج نیز در بخش مربوط به همین روش گزارش شده است. در روش SVM آبشاری نیز در C = 4، دقت کمی بهبود داشته است که در جدول ۵-۵ آورده شده است.

جدول ۵-۵: مقایسه معیار دقت و زمان SVM آبشاری برای C = 4,C = 6

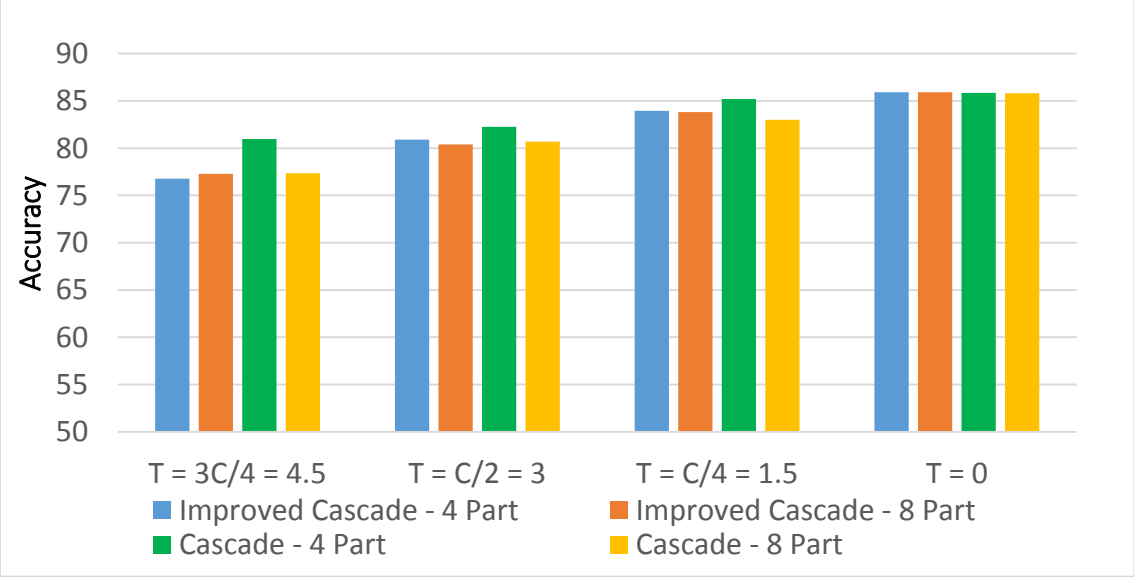
مجموعه‌داده IMDB	دقت(ACC)	تعداد SV	متوسط زمان آموزش	زمان آزمون	زمان کل	C
SVM آبشاری – ۸ بخش	85.816%	12159	1595	4432	6027	6
SVM آبشاری – ۴ بخش	85.844%	12269	1673	4419	6092	6
SVM آبشاری – ۲ بخش	85.972%	12443	1966	5576	7542	6
SVM آبشاری – ۸ بخش	85.832%	12697	1459	4409	5868	4
SVM آبشاری – ۴ بخش	85.904%	12768	1512	4422	5934	4
SVM آبشاری – ۲ بخش	85.848%	12889	1791	5460	7251	4

در ادامه بررسی پارامترها و تاثیر آن‌ها در روند یادگیری مدل‌ها، به تاثیر مقدار C در روش SVM آبشاری با پالایش بردارهای پشتیبان می‌پردازیم. همان‌طور که در معرفی شد، در روش SVM آبشاری با پالایش بردارهای پشتیبان، تمام بردارهای پشتیبان با ضرایب لاگرانژ بزرگتر از صفر به مرحله بعد فرستاده نمی‌شود. ما با تعیین حدآستانه، سعی در پالایش بردارهای پشتیبان داشته و بردارهای پشتیبان موثرتر که مقادیر ضرایب لاگرانژ بیشتر را دارا می‌باشند، برای مرحله بعد انتخاب می‌نماییم. بنابراین با کاهش

مقدار بردارهای پشتیبان ورودی SVM مرحله بعد، زمان آموزش مدل رده‌بند کاهش یافته و دقت نیز تا حد مطلوبی حفظ خواهد شد. حال ما توجه به نمودارهای شکل ۵-۶ و شکل ۵-۷ این حد آستانه را برابر T = C/2 در نظر می‌گیریم. با در نظر گرفتن این مقدار تا حدودی توازنی بین حفظ دقت و کاهش زمان آموزش مدل رده‌بند برقرار می‌شود.



شکل ۵-۶: زمان آموزش SVM آبشاری و آبشاری بهبودیافته با مقادیر مختلف حد آستانه برای ضرایب لاگرانژ



شکل ۵-۷: دقت SVM آبشاری و آبشاری بهبودیافته با مقادیر مختلف حد آستانه برای ضرایب لاگرانژ

۵ـ۵ـ۴ مقایسه مدل‌ها

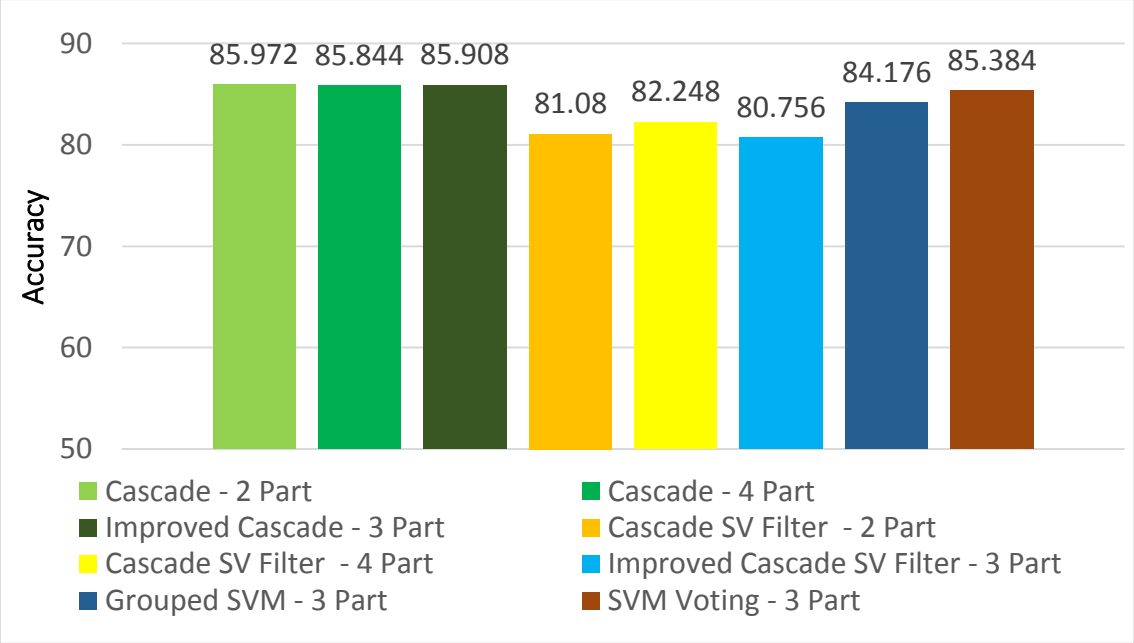
در این قسمت به مقایسه تمام روش‌های پیاده‌سازی شده در این پژوهش می‌پردازیم. برای تمام حالت مجموعه‌داده به سه بخش تقسیم شده است. برای روش آبشاری و آبشاری با پالایش بردارهای پشتیبان که نمی‌توان مجموعه‌داده را به سه بخش تقسیم کرد، نتایج مربوط به دو حالت دو و چهار بخشی مجموعه‌داده آموزش آورده شده است. در جدول ۵-۶ دو معیار تشخیص و حساسیت آورده شده است.

جدول ۵-۶: مقایسه معیار تشخیص و حساسیت مدل‌های مختلف مبتنی بر SVM

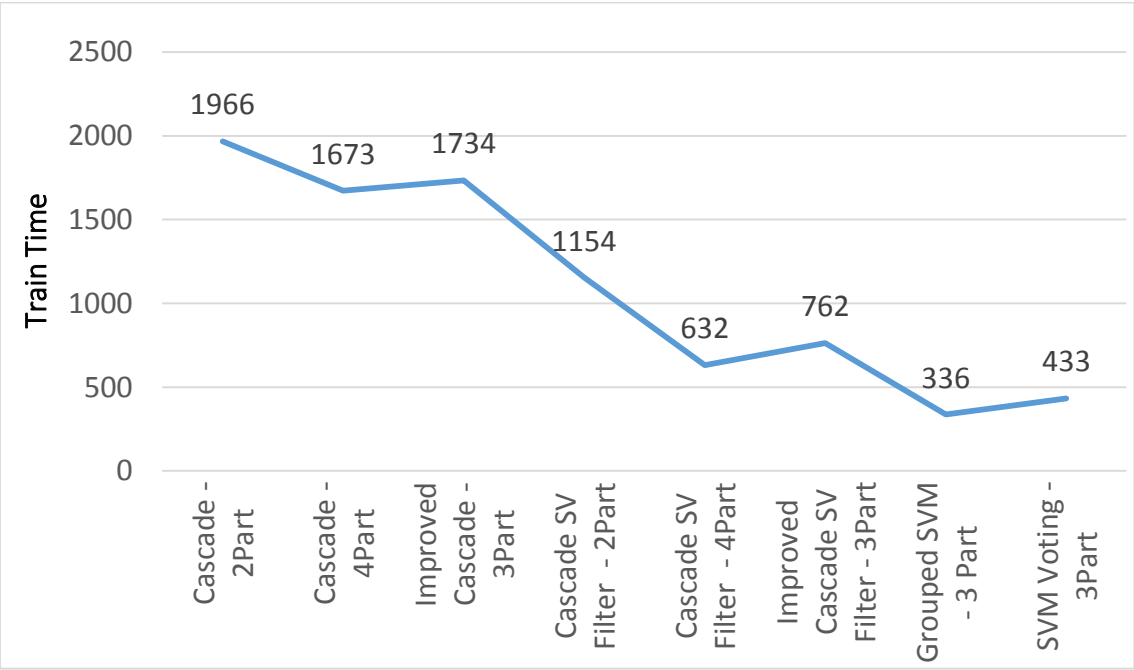
مدل SVM	تشخیص (TNR)	حساسیت (TPR)
آبشاری - ۲ بخش	0.8644	0.8550
آبشاری - ۴ بخش	0.8638	0.8507
آبشاری بهبودیافته – ۳ بخش	0.8647	0.8563
آبشاری با پالایش بردارهای پشتیبان – ۲ بخش	0.8113	0.8102
آبشاری با پالایش بردارهای پشتیبان – ۴ بخش	0.8324	0.8131
آبشاری بهبودیافته با پالایش بردارهای پشتیبان – ۳ بخش	0.8028	0.8124
گروهی – ۳ بخش	0.8035	0.8909
SVM مبتنی بر رأی‌گیری – ۳ بخش	0.8626	0.8454

با توجه به نتایج جدول فوق، بیشترین مقدار معیار حساسیت متعلق به روش SVM گروهی می‌باشد اما مقدار معیار تشخیص این روش نیز جزو کمترین مقادیر می‌باشد. بیشترین مقدار معیار تشخیص نیز به روش SVM آبشاری بهبودیافته می‌باشد که دارای اختلاف کمی با روش SVM آبشاری و SVM مبتنی بر رأی‌گیری می‌باشد. بالا بودن هر دو معیار تشخیص و حساسیت در کنار هم منجر به ایجاد دقت بالا می‌شود.

در ادامه، نتایج مربوط به معیار دقت در شکل ۵-۸ و نتایج مربوط به زمان آموزش در شکل ۵-۹ آورده شده است.



شکل ۵-۸: مقایسه معیار دقت در مدل‌های مختلف مبتنی بر SVM



شکل ۵-۹: مقایسه معیار زمان آموزش در مدل‌های مختلف مبتنی بر SVM

روش‌های SVM آبشاری، آبشاری بهبودیافته و SVM مبتنی بر رأی‌گیری بالاترین دقت را در رده‌بندی

قطبیت نظرات کاربران دارند. کمترین زمان آموزش نیز متعلق به روش‌های SVM گروهی، SVM مبتنی بر رأی‌گیری و آبشاری با پالایش بردارهای پشتیبان دارند. در این بین روش SVM گروهی کمترین دقت را داراست و روش SVM آبشاری با پالایش بردارهای پشتیبان نیز در نسبت به سایر روش‌ها دقت پایین‌تری دارد. روش SVM مبتنی بر رأی‌گیری در بین این روش‌ها دارای بهترین نتیجه می‌باشد. قرار گرفتن این مدل SVM، در بین روش‌های با دقت بالا و همچنین قرار داشتن در بین روش‌هایی با زمان آموزش کم، دلیلی محکم بر این موضوع می‌باشند.

۵ـ۵ـ۵ پیاده‌سازی موازی در سایر پژوهش‌ها بر روی مجموعه‌داده IMDB

در تعدادی از پژوهش‌های دیگر نیز روش‌های موازی و توزیع شده بر روی مجموعه‌داده IMDB انجام شده است. در این قسمت نتایج مربوط به این پژوهش‌ها به همراه نتایج پیاده‌سازی SVM استاندارد در محیط اسپارک توسط ما، آورده شده است. نتایج در جدول ۵-۷ آورده شده است.

جدول ۵-۷: نتیجه SVM استاندارد پیاده‌سازی شده و مقایسه با سایر مراجع آزمایش شده بر روی IMDB

مجموعه‌داده IMDB	دقت	زمان در محیط ترتیبی[1]	زمان در محیط توزیع شده
[48]	60.2%	150,590	37,659
[45]	61.2%	185,000	46,310
[44]	63.7%	222,559	55,633
[46]	56.84% (63.58%)	_	7.94
[47]	82.58%	_	110
استانداردSVM	85.946%	≈18,000	-

در پژوهش اول، روشی برای تحلیل نظرات انگلیسی مبتنی بر رده‌بند SVM در محیط موازی Cloudera ارائه کرده است [۴۴]. در این پژوهش برای آموزش از ۹۰ هزار نظر برای آموزش مدل رده‌بند استفاده

[1] Sequential Environment Time

شده است. پژوهش دوم مربوط به الگوریتم Fuzzy C-Mean به صورت موازی می‌باشد که بر روی IMDB و در Cloudera پیاده‌سازی شده است [۴۸]. مجموعه آموزش در این پژوهش شامل ۶۰ هزار نظر می‌باشد. در پژوهش بعد، الگوریتمی به نام STING معرفی و ارائه شده است و به دو صورت موازی و ترتیبی در محیط Cloudera پیاده‌سازی شده است[۴۵]. مجموعه‌داده آموزشی در این پژوهش نیز شامل ۹۰ هزار نظر انگلیسی می‌باشد. روش‌های SVM و LR در پژوهش چهارم استفاده شده است که نتایج مربوط به SVM در جدول فوق آورده شده است. در این پژوهش از محیط Spark برای پیاده‌سازی استفاده شده است [۴۶]. در آخرین پژوهشی که نتایج آن گزارش شده است، از چند نوع شبکه عصبی برای تحلیل احساسات بهره گرفته شده است. همچنین برای پردازش نیز از واحد گرافیک استفاده شده است[۴۷].

همان‌طور که مشاهده می‌شود روش پیاده‌سازی شده توسط ما از نظر دقت نسبت به سایر مقالات بهتر عمل کرده است و از نظر زمان آموزش نیز بهبود داشته است. در جدول ۵-۸ نتایج جزئی‌تر پیاده‌سازی SVM استاندارد آورده شده است. زمان اجرای کامل الگوریتم SVM تقریبا ۷۵۰۰ ثانیه می‌باشد و ۱۰۵۰۰ ثانیه نیز زمان پیش‌پردازش می‌باشد.

جدول ۵-۸: نتایج SVM استاندارد روش پیاده‌سازی شده

مجموعه‌داده IMDB	دقت	تعداد SV	زمان آموزش	زمان آزمون	زمان کل
SVM استاندارد	85.946%	12740	3052	4385	7437

۵ـ۶ نتایج سایر پژوهش‌ها

در این قسمت نیز نتایج سایر پژوهش‌هایی که بر روی مجموعه‌داده IMDB آزمایش شده‌اند، در جدول ۵-۹ آورده شده است. در تعدادی از این پژوهش‌ها، دقت‌های خوبی گزارش شده است. اما لازم به ذکر است در این مقالات روش‌های استخراج ویژگی متفاوت و پیچیده‌تری استفاده شده است و همچنین در

مورد زمان اجرای روش پیشنهادی اطلاعاتی گزارش نشده است. در ادامه نتایج این روش‌ها آورده شده است.

در مرجع [۴۹] یک روش پیش‌پردازش ارائه شده است که بر روی پنج دیتاست از جمله IMDB اعمال شده است. و برای ارزیابی از سه رده‌بند SVM، Naïve Bayes و ANN استفاده شده است.

در مرجع [۵۰] از شبکه عصبی بازگشتی برای رده‌بندی متن استفاده شده است و بر روی چهار دیتاست انجام شده است. همچنین در این مرجع از چند دیتاست برای آموزش استفاده می‌شود و مرحله تست برای هرکدام از دیتاست‌ها به صورت جداگانه انجام می‌گیرد.

در مرجع [۵۱] نیز از مدلی مبتنی بر شبکه عصبی کانولوشن و یادگیری عمیق استفاده شده است و در قسمت ارزیابی با چند مقاله دیگر مقایسه شده است.

۵ـ۷ جمع‌بندی

در این فصل نتایج حاصل از پیاده‌سازی روش‌های پیشنهادی در محیط اسپارک ارائه شد. با بررسی و ارزیابی نتایج پژوهش و سایر پژوهش‌های موجود، زمان مربوط به روش‌های ارائه‌شده از سرعت بیشتری برخوردار است و دقت نیز در حد مطلوب و قابل قبول حفظ شده است.

جدول ۵-۹: نتایج تعدادی از سایر پژوهش‌های انجام شده بر روی IMDB

	روش مورد استفاده	دقت	سال	استناد
نتایج ارائه شده در [۴۹]	SVM	84.74	[49] 2017	1
	SVM + preprocessing	86.44		
	NB	74.91		
	NB + preprocessing	76.94		
	ANN	78.64		
	ANN + preprocessing	82.03		
نتایج ارائه شده در [۵۰]	Train: IMDB	88.5	[50] 2016	67
	Train: IMDB + SST1	89.5		
	Train: IMDB + SST2	89.8		
	Train: IMDB + SUBJ	89.3		
نتایج ارائه شده در [۵۱]	Multinomial NB-uni [52]	83.55	[52] 2012	480
	Multinomial NB-bi [52]	86.59		
	SVM-unigram [52]	86.95		
	SVM-bigram [52]	89.16		
	NBSVM-unigram [52]	88.29		
	NBSVM-bigram [52]	91.22		
	WEEB + BoW [54]	87.8	[54] 2012	72
	WRBB + BoW(bnc) [54]	89.2		
	BoW(bnc) [55]	87.8	[55] 2011	957
	Full + BoW [55]	88.3		
	Full + Unlabeled + BoW[55]	88.9		
	Paragraph Vector(PV) [56]	92.5	[56] 2014	2382
	PV(LogReg) [57]	94.4	[57] 2015	13
	PV (2-Layer MLP) [57]	94.5		
	Paper’s Approach[51]	93.2	[51] 2018	2

فصل ششم

نتیجه‌گیری

۶ـ۱ جمع‌بندی و نتیجه‌گیری

تحلیل نظرات کاربران در دنیای امروز، بسیار کارآمد می‌باشد و در زمینه‌های مختلفی مورد استفاده قرار می‌گیرد. همین موضوع باعث توجه پژوهش‌گران به این شاخه شده است. اما مشکلاتی در این شاخه وجود دارد از جمله حجم عظیم نظرات که نیازمند روش‌های قدرتمند و سریع‌تر می‌باشد. در این بین رده‌بندهای موازی، برای کاهش زمان آموزش و افزایش سرعت رده‌بندی ارائه شده‌اند. در این پژوهش از SVMهای موازی برای تحلیل نظرات کاربران استفاده شده است. پیاده‌سازی نیز در چارچوب اسپارک انجام شده است که مناسب برای داده‌های عظیم می‌باشد. در قسمت نتایج و ارزیابی روش پیشنهادی، زمان‌های بدست آمده نسبت به سایر پژوهش‌ها، بهبود قابل توجهی داشته است و دقت بدست آمده نیز نسبت به تعدادی از پژوهش‌ها کمتر و نسبت به تعدادی بیشتر می‌باشد، اما در کل با در نظر گرفتن کاهش زمان مصرفی، قابل قبول می‌باشد.

۶ـ۲ کارهای آینده

برای کارهای آینده می‌توان همچنان روی کاهش سرعت در کنار حفظ دقت و یا افزایش آن تحقیق نمود. در پژوهش‌های آینده می‌توان از روش‌های پیش‌پردازش دیگر مانند در نظر گرفتن نقش کلمات در جمله، استفاده نمود تا دقت افزایش یابد و بر روی کاهش سرعت آن‌ها سعی نمود.

در این پژوهش از رده‌بند SVM که بهتر در این زمینه عمل می‌نماید و بیشتر مورد بهره قرار می‌گیرد، استفاده شده است. برای کارهای آتی می‌توان بر روی رده‌بندهای دیگر همانند Naïve Bayes یا Maximum Entropy و سایر روش‌های رده‌بندی مبتنی بر یادگیری ماشین کار کرد و سعی در موازی-سازی آن‌ها نمود. همچنین می‌توان از ترکیب چند رده‌بند برای بهبود عملکرد، بهره برد.

در صورت استفاده از SVM و بهره از تابع هسته در فرایند آموزش، می‌توان از ترکیب توابع هسته مختلف برای بهبود دقت، سود برد.

منابع

[1] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, pp. 1-167, 2012.

[2] A. Bakliwal, J. Foster, J. van der Puil, R. O'Brien, L. Tounsi, and M. Hughes, "Sentiment analysis of political tweets: Towards an accurate classifier," 2013.

[3] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent twitter sentiment classification," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, 2011, pp. 151-160.

[4] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of twitter data," in *Proceedings of the workshop on languages in social media*, 2011, pp. 30-38.

[5] C. Olston and M. Najork, "Web crawling," *Foundations and Trends® in Information Retrieval*, vol. 4, pp. 175-246, 2010.

[6] K. Guo, L. Shi, W. Ye, and X. Li, "A survey of internet public opinion mining," in *Progress in Informatics and Computing (PIC), 2014 International Conference on*, 2014, pp. 173-179.

[7] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, pp. 130-137, 1980.

[8] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, 2002, pp. 79-86.

[9] K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews," in *Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 519-528.

[10] K. Lai and N. Cerpa, "Support vs. confidence in association rule algorithms," in *Proceedings of the OPTIMA Conference, Curicó*, 2001, pp. 1-14.

[11] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to WordNet: An on-line lexical database," *International journal of lexicography*, vol. 3, pp. 235-244, 1990.

[12] A. Z. Khan, M. Atique, and V. Thakare, "Combining lexicon-based and learning-based methods for Twitter sentiment analysis," *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)*, p. 89, 2015.

[13] A. Mudinas, D. Zhang, and M. Levene, "Combining lexicon and learning based approaches for concept-level sentiment analysis," in *Proceedings of the First International Workshop on Issues of Sentiment Discovery and Opinion Mining*, 2012, p. 5.

[14] J. Fang and B. Chen, "Incorporating Lexicon Knowledge into SVM Learning to Improve Sentiment Classification," in *Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP)*, 2011, pp. 94-100.

[15] A. Bhattacharya and S. Bhatnagar, "Big data and apache spark a review," *Int. J. Eng. Res. Sci*, vol. 2, p. 206, 2016.

[16] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, 1992, pp. 144-152.

[17] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20,

pp. 273-297, 1995.
[18] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," 1998.
[19] "*The Simplified SMO Algorithm*." Available: http://cs229.stanford.edu/materials/smo.pdf, 2009 [Sep. 17, 2018].
[20] M. Tsytsarau and T. Palpanas, "Survey on mining subjective data on the web," *Data Mining and Knowledge Discovery*, vol. 24, pp. 478-514, 2012.
[21] Z. Singla, S. Randhawa, and S. Jain, "Sentiment analysis of customer product reviews using machine learning," in *Intelligent Computing and Control (I2C2), 2017 International Conference on*, 2017, pp. 1-5.
[22] A. P. Jain and P. Dandannavar, "Application of machine learning techniques to sentiment analysis," in *Applied and Theoretical Computing and Communication Technology (iCATccT), 2016 2nd International Conference on*, 2016, pp. 628-632.
[23] Y. Bidulya and E. Brunova, "Sentiment analysis for bank service quality: A rule-based classifier," in *Application of Information and Communication Technologies (AICT), 2016 IEEE 10th International Conference on*, 2016, pp. 1-4.
[24] U. A. Siddiqua, T. Ahsan, and A. N. Chy, "Combining a rule-based classifier with weakly supervised learning for twitter sentiment analysis," in *Innovations in Science, Engineering and Technology (ICISET), International Conference on*, 2016, pp. 1-4.
[25] R. Moraes, J. F. Valiati, and W. P. G. Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," *Expert Systems with Applications*, vol. 40, pp. 621-633, 2013.
[26] H. Lidong and Z. Hui, "A new short text sentimental classification method based on multi-mixed convolutional neural network," in *2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, 2018, pp. 93-99.
[27] T. U. Haque, N. N. Saber, and F. M. Shah, "Sentiment analysis on large scale Amazon product reviews," in *Innovative Research and Development (ICIRD), 2018 IEEE International Conference on*, 2018, pp. 1-6.
[28] S. Sohangir, N. Petty, and D. Wang, "Financial Sentiment Lexicon Analysis," in *Semantic Computing (ICSC), 2018 IEEE 12th International Conference on*, 2018, pp. 286-289.
[29] A. B. Bondi, "Characteristics of scalability and their impact on performance," in *Proceedings of the 2nd international workshop on Software and performance*, 2000, pp. 195-203.
[30] M. Karanasou, A. Ampla, C. Doulkeridis, and M. Halkidi, "Scalable and Real-time Sentiment Analysis of Twitter Data," in *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*, 2016, pp. 944-951.
[31] A. Hamzehei, M. Ebrahimi, E. Shafiee, R. K. Wong, and F. Chen, "Scalable Sentiment Analysis for Microblogs Based on Semantic Scoring," in *Services Computing (SCC), 2015 IEEE International Conference on*, 2015, pp. 271-278.
[32] K. Bhattacharjee and L. Petzold, "What Drives Consumer Choices? Mining Aspects and Opinions on Large Scale Review Data Using Distributed Representation of Words," in *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*, 2016, pp. 908-915.
[33] P. R. Cavalin, M. A. d. C. Gatti, T. Moraes, F. S. Oliveira, C. S. Pinhanez, A. Rademaker, *et al.*, "A scalable architecture for real-time analysis of

microblogging data," *IBM Journal of Research and Development*, vol. 59, pp. 16: 1-16: 10, 2015.
[34] B. Yan, Z. Yang, Y. Ren, X. Tan, and E. Liu, "Microblog Sentiment Classification Using Parallel SVM in Apache Spark," in *Big Data (BigData Congress), 2017 IEEE International Congress on*, 2017, pp. 282-288.
[35] P. Tripathy, S. S. Rautaray, and M. Pandey, "Map-reduce based parallel support vector machine for risk analysis," in *I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2017 International Conference on*, 2017, pp. 300-303.
[36] A. Priyadarshini, "A map reduce based support vector machine for big data classification," *International Journal of Database Theory and Application*, vol. 8, pp. 77-98, 2015.
[37] G. Caruana, M. Li, and M. Qi, "A MapReduce based parallel SVM for large scale spam filtering," in *Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on*, 2011, pp. 2659-2662.
[38] W. Nie, B. Fan, X. Kong, and Q. Ma, "Optimization of multi kernel parallel support vector machine based on Hadoop," in *Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2016 IEEE*, 2016, pp. 1602-1606.
[39] H. P. Graf, E. Cosatto, L. Bottou, I. Dourdanovic, and V. Vapnik, "Parallel support vector machines: The cascade svm," in *Advances in neural information processing systems*, 2005, pp. 521-528.
[40] H. Du, S. Teng, X. Fu, W. Zhang, and Y. Pu, "A cooperative intrusion detection system based on improved parallel SVM," in *Pervasive Computing (JCPC), 2009 Joint Conferences on*, 2009, pp. 515-518.
[41] D. Kun, L. Yih, and A. Perera, "Parallel SMO for training support vector machines," *SMA5505 Project Final Report*, 2003.
[42] C. Jiang, T. Wu, J. Xu, N. Zheng, M. Xu, and T. Yang, "A MapReduce-Based Distributed SVM for Scalable Data Type Classification," in *International Conference on Collaborative Computing: Networking, Applications and Worksharing*, 2016, pp. 115-126.
[43] N. K. Alham, M. Li, S. Hammoud, Y. Liu, and M. Ponraj, "A distributed SVM for image annotation," in *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, 2010, pp. 2983-2987.
[44] V. N. Phu, V. T. N. Chau, and V. T. N. Tran, "SVM for English semantic classification in parallel environment," *International Journal of Speech Technology*, vol. 20, pp. 487-508, 2017.
[45] N. D. Dat, V. N. Phu, V. T. N. Tran, V. T. N. Chau, and T. A. Nguyen, "STING algorithm used english sentiment classification in a parallel environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 31, p. 1750021, 2017.
[46] M. S. Bhuvan, V. D. Rao, S. Jain, T. Ashwin, and R. M. R. Guddeti, "Semantic sentiment analysis using context specific grammar," in *Computing, Communication & Automation (ICCCA), 2015 International Conference on*, 2015, pp. 28-35.
[47] X. Xu, H. Ge, and S. Li, "An improvement on recurrent neural network by combining convolution neural network and a simple initialization of the weights," in *Online Analysis and Computing Science (ICOACS), IEEE International Conference of*, 2016, pp. 150-154.
[48] V. N. Phu, N. D. Dat, V. T. N. Tran, V. T. N. Chau, and T. A. Nguyen, "Fuzzy C-

means for english sentiment classification in a distributed system," *Applied Intelligence*, vol. 46, pp. 717-738, 2017.

[49] R. Coughlin, J.-C. Coetsier, and R. Jiamthapthaksin, "Integrating Labeled Latent Dirichlet Allocation into sentiment analysis of movie and general domains," in *Knowledge and Smart Technology (KST), 2017 9th International Conference on*, 2017, pp. 18-22.

[50] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," *arXiv preprint arXiv:1605.05101*, 2016.

[51] A. Hassan and A. Mahmood, "Convolutional Recurrent Deep Learning Model for Sentence Classification," *IEEE Access*, vol. 6, pp. 13949-13957, 2018.

[52] S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, 2012, pp. 90-94.

[53] L. Bottou and C.-J. Lin, "Support vector machine solvers," *Large scale kernel machines*, vol. 3, pp. 301-320, 2007.

[54] G. E. Dahl, R. P. Adams, and H. Larochelle, "Training restricted boltzmann machines on word observations," *arXiv preprint arXiv:1202.5695*, 2012.

[55] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, 2011, pp. 142-150.

[56] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International Conference on Machine Learning*, 2014, pp. 1188-1196.

[57] J. Hong and M. Fang, "Sentiment analysis with deeply learned distributed representations of variable length texts," Technical report, Stanford University2015.

Abstract

The ever increasing growth of media and social networks, alongside the spread of analytical commercial and news webpages, have led to an extensive presence of people all around the world in such environments. This has in turn, resulted in enormous propagation and sharing of data and comments. Valuable information could be extracted by analyzing this massive amount of data. But besides this advantage, processing this huge amount of information encounters several challenges. The processing speed of the conventional methods is too slow against the rapid growth of the input data flow. To solve this problem, scalable methods as well as novel tools in the big data processing area, may be employed to accelerate the processing procedure. Although a lot of researches have been already carried out in this area, there is still room for more improvements. Low processing speed, inadequate accuracy rate, high complexity, and domain dependency are among the issues in similar previous studies.

Spark is a relatively novel framework for processing big data, which carries out processes in main memory to increase the processing speed of the algorithm. In this research, the support vector machine (SVM) and Spark are used simultaneously to classify user opinions. The proposed method, along with maintenance of desirable accuracy, is not dependent on a specific domain and can be simply implemented. To benefit from Spark's ability in distributed processing, some methods of parallel SVMs, including cascade SVM, improved cascade SVM, grouped SVM and voting among multiple SVMs have been utilized. To compare the efficiency of the above-mentioned methods, the standard SVM has also been implemented. In the present study, IMDB dataset is used for the evaluation. This dataset contains various different comments in English on the movies and series. Results for classification indicated the best accuracy to be 85.946%. Moreover, the elapsed time to run the standard SVM algorithm in the Spark environment has been dropped more than 8 times compared to similar research results. The elapsed time for parallel SVMs against the standard SVM has also been remarkably improved. In the best case scenario, the training time of the proposed method is reduced by 60 times compared to the standard SVM training time.

Keywords: opinion mining, Support Vector Machine, parallel processing, Spark

Shahrood University of Technology

Faculty of Computer Engineering

M.Sc. Thesis in Artificial Intelligence Engineering

Scalable opinion mining using cluster computing

By:

Mojtaba Asadollahi

Supervisor:

Dr. Hoda Mashayekhi

Sep 2018