Iranian Association of Information and Communication Technology
IKT 2013
Shiraz University
Regular Paper

تعیین احساس از روی متن فارسی

محمد مهدی محصولی[1]، محمد مهدی همایون‌پور[1]

[1] آزمایشگاه پردازش هوشمند داده‌های چندرسانه‌ای، دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر
دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)، {mahsuli, homayoun}@aut.ac.ir

چکیده

در بسیاری از کاربردهای تبدیل متن به گفتار بهتر است تا مشخصات گفتار تولید شده هرچه بیشتر شبیه به انسان باشد. برای این کار باید متنی که توسط سیستم ادا می‌شود، از لحاظ معنایی بررسی شود. یکی از مهم‌ترین این ویژگی‌های معنایی، احساس حاکم بر متن است. در زمینه تعیین احساس از روی متن، کارهای مختلفی در زبان انگلیسی صورت گرفته است؛ اما کمتر کسی اقدام به کار بر روی پیکره‌های فارسی کرده است. در این مقاله، پیکره ای شامل ۳۷۰۲ جمله از ۶ کلاس احساس خوشحالی، عصبانیت، خنثی، ناراحتی، تنفر و ترس تهیه شده است و روش‌های گوناگونی جهت تعیین احساس از روی یک جمله متنی به زبان فارسی به کار گرفته شده است. با بررسی نتایج بدست آمده متوجه می‌شویم که عملکرد برنامه در صورت استفاده از رویکرد مبتنی بر پیکره مطلوب است و دارای حداکثر دقت ۷۸/۸۵٪ و زمان بسیار کوتاه آموزش می‌باشد.

واژه‌های کلیدی

پردازش زبان طبیعی، مدل زبانی، تحلیل معنایی، یادگیری ماشین، Complement Naïve Bayes.

۱- مقدمه

امروزه کاربردهای پردازش متن در زندگی روزمره انسان‌ها به حدی مورد استفاده قرار می‌گیرد که بدون این امر، بسیاری از کارها مختل می‌شود. به طور مثال می‌توان به ترجمه ماشینی، استخراج اطلاعات، تشخیص گفتار و خلاصه‌سازی خودکار متون اشاره کرد. یکی از مواردی که کاربردهای خاص خود را دارد، تبدیل متن به گفتار می‌باشد. در بسیاری از کاربردهای تبدیل متن به گفتار بهتر است تا مشخصات گفتار تولید شده هرچه بیشتر شبیه به انسان باشد. برای این کار باید متنی که توسط سیستم ادا می‌شود، از لحاظ معنایی بررسی شود. یکی از مهم‌ترین این ویژگی‌های معنایی، احساس حاکم بر متن است.

در زمینه تعیین احساس از روی متن، کارهای مختلفی در زبان انگلیسی صورت گرفته است؛ اما کمتر کسی اقدام به کار در زبان فارسی کرده است که عدم وجود پیکره مناسب، از مهم‌ترین دلایل آن می‌باشد. در این مقاله، پیکره ای شامل ۳۷۰۲ جمله از ۶ کلاس احساس خوشحالی، عصبانیت، خنثی، ناراحتی، تنفر و ترس تهیه شده است و روش‌های گوناگونی جهت تعیین احساس از روی یک جمله متنی به زبان فارسی به کار گرفته شده است. ایده کلی این است که مجموعه‌ای از جملات در دست داریم و هرکدام از آنها دارای برچسب یکی از احساسات تعریف شده هستند. حال سیستم باید از این مجموعه جملات، ویژگی‌هایی را استخراج کند و از آنها در تصمیم‌گیری خود استفاده نماید.

در نحوه استخراج ویژگی‌ها و دسته‌بندی، روش‌های مختلفی ارائه شده است که بیشتر این روش‌ها به صورت آماری کار می‌کنند. این در حالی است که بعضی از روش‌ها از یک سری قوانین که به صورت دست‌ساز توسط یک فرد خبره تولید شده‌اند نیز استفاده می‌کنند. بعضی از روش‌ها از یک دسته کلمات[1] جهت انتخاب ویژگی‌های کاندید استفاده می‌کنند [۱]. در بعضی روش‌های دیگر، از مدل‌های مخلوط[2] مانند GMM استفاده می‌شود [۲]. محققان دانشگاه ایلینویز در [۳]، تعیین احساس را بر روی متن یک داستان انجام داده‌اند. این در حالی است که تعیین احساس از روی یک جمله که موضوع این مقاله است، به دلیل کمبود اطلاعات موجود در جمله، دشوارتر است و کمتر صورت پذیرفته است. همچنین کار صورت گرفته توسط محققان دانشگاه کُرنل در [۴]، بر روی دو کلاس احساس مثبت و منفی است؛ اما ما در این مقاله، اقدام به شناسایی ۶ احساس مختلف کرده‌ایم.

[1] Bag of Word (BoW)
[2] Mixture Model

۲- روش ارائه شده

ابتدا لازم است تا بعضی پیش‌پردازش‌ها بر روی داده آموزشی صورت گیرد. به طور مثال، ما کاراکترهای خاص مانند ~!@#$%^&*( ) را از جملات حذف کردیم. همچنین یک کاراکتر خاص در نوشتار فارسی موجود است که باعث کشیده شدن فاصله بین حروف می‌گردد و توسط ترکیب Shift+J قابل تایپ است. به طور مثال کلمه "کشـــــــیده" دارای این کاراکتر می‌باشد. اگر این کاراکترهای خاص حذف نشوند، ممکن است یک کلمه که به صورت‌های گوناگون در جملات آموزشی آمده است، به صورت چند ویژگی مختلف در نظر گرفته شود. همچنین وجود اعداد در دادگان[3] نیز می‌تواند تخمین سیستم را دچار مشکل کند. برای جلوگیری از بروز این قبیل مشکلات، تابعی به نام StripPunctuations را پیاده‌سازی کرده‌ایم و قبل از استفاده از داده‌ها آن را به کار می‌بریم.

برای انجام یک دسته‌بندی از داده‌ها، ما هر کلمه را به عنوان یک ویژگی در نظر گرفتیم. یعنی فرکانس تکرار هر کلمه در جمله آموزشی، مقدار یک ویژگی را تشکیل می‌دهد. نام این روش، TF[4] است. بدین منظور ابتدا یک دیکشنری درست کردیم و با دیدن هر کلمه در مجموعه آموزشی، در صورت جدید بودن کلمه، آن را به دیکشنری اضافه کردیم و در غیر این صورت، یک واحد به فرکانس مربوط به آن کلمه افزودیم. پس از بدست آمدن دیکشنری فرکانس‌ها که در اصل با درنظر گرفتن مدل زبانی unigram بدست می‌آید، تعداد تکرار هر کلمه از دیکشنری در هر جمله آموزشی را شمردیم و مجموعه داده‌های آموزشی را با پسوند csv. تولید کردیم. فرمت csv. این امکان را فراهم می‌کند تا به راحتی بتوان داده‌ها را در نرم‌افزارهای کاربردی دیگری نظیر Weka و Matlab دسته‌بندی کرد.

به منظور دسته‌بندی داده‌ها روش‌های گوناگونی را بررسی کردیم. داده-ها شامل ۷ کلاس عصبانیت، خوشحالی، خنثی، ناراحتی، تنفر، ترس و غم بودند که به دلیل شباهت زیاد دو کلاس ناراحتی و غم، آنها را با یکدیگر ادغام کردیم.

اما هدف اصلی در این مقاله، بدست آوردن نتایج به صورت ترکیب احتمالاتی از کلاس‌های مختلف بوده است. بدین منظور به جای استفاده از فرکانس هر کلمه، میزان احتمال تعلق کلمه مورد نظر به هر کلاس را محاسبه می‌کنیم. به طور مثال، فرض کنید کلمه "تیره" در ۷۵٪ مواقع به کلاس ترس، و در ۲۵٪ به کلاس ناراحتی تعلق دارد و تعلق آن به کلاس-های دیگر صفر است.[5] سپس با استفاده از مدل unigram، احتمال تعلق هر جمله به یک احساس را می‌توان توسط ضرب احتمال تعلق کلمات درون جمله به آن احساس نمایش داد.

$$P(emotion_j \mid Sentence) \propto \prod_{i:word_i \in Sentence} P(emotion_j \mid word_i) \quad (۱)$$

پس از محاسبه این احمالات، آنها را نرمالیزه می‌کنیم؛ بدین صورت که مقدار هر کدام را بر جمع تعلقات جمله به کلاس‌های مختلف تقسیم می‌کنیم تا مجموع احتمالات یک شود.

نکته‌ای که وجود دارد، این است که ممکن است تمامی احتمالات برابر صفر شوند. اگر برای تمامی احساس ها، به ازای حداقل یک کلمه از جمله آزمایشی، احتمال صفر داشته باشیم، این اتفاق می‌افتد. به طور مثال، فرض کنید می‌خواهیم احساس جمله "من رفتم" را از میان دو کلاس خوشحالی و ناراحتی بدست آوریم و تعلقات به صورت زیر باشد.

$$P(\text{خوشحالی} \mid \text{"من"}) = 1, \quad P(\text{ناراحتی} \mid \text{"من"}) = 0$$

$$P(\text{خوشحالی} \mid \text{"رفتم"}) = 0, \quad P(\text{ناراحتی} \mid \text{"رفتم"}) = 1$$

در این صورت با استفاده از رابطه (۱) داریم:

$$P(\text{خوشحالی} \mid \text{"من رفتم"}) = P(\text{ناراحتی} \mid \text{"من رفتم"}) = 0$$

اما ما می‌دانیم که مجموع این تعلقات باید برابر با ۱ باشد. راه حل ساده این مشکل، درنظر گرفتن احتمالات برابر در مواردی نظیر مورد بالا است. یعنی برای مسئله دوکلاسه، احتمال ۵۰٪ را به هر کدام از دو کلاس خوشحالی و ناراحتی نسبت می‌دهیم. اما این راه حل مقداری ساده‌انگارانه است. راه حل بهتری که وجود دارد، نرم کردن[6] مدل احتمالاتی است؛ بدین معنی که به کلمات دیده نشده، یک احتمال کوچک غیر صفر را نسبت می-دهیم. این مقدار را اپسیلون می‌نامیم. همچنین کلماتی که در هیچ کلاسی دیده نشده‌اند، در زمان آزمایش بررسی نمی‌شوند. یعنی به ازای آنها، احتمالات تغییری نمی‌کند یا به عبارت دیگر، در ۱ ضرب می‌شود.

به منظور کاهش بار محاسبات می‌توانیم بعضی کلمات را که بار اطلاعاتی درستی ندارند و یا دارای اهمیت کمی هستند، از ویژگی‌ها حذف کنیم. این کلمات، کلمات توقف[7] نام دارند. انتخاب مناسب کلمات توقف بسیار مهم است. زیرا انتخاب نامناسب آنها باعث از دست رفتن ویژگی‌های مفید خواهد شد. در این تحقیق، لیستی از کلمات توقف شامل ۲۲۷۶ مدخل تهیه شده است که با توجه به عملکرد سیستم و مدل تولید شده

[3] Dataset
[4] Term Frequency
[5] این مقادیر وابسته به داده‌های آموزشی، روش آموزش، مقادیر پارامترهای روش، استفاده یا عدم استفاده از لیست کلمات توقف و ... می‌باشد. در هر صورت، مجموع این تعلقات برابر با ۱ است.
[6] Smoothing
[7] Stop Words

توسط آن، انتخاب شده‌اند. برخی پیشوندها مانند "می" در این لیست لحاظ شده‌اند. با این کار، به طور مثال کلمات "رفتم" و "می‌رفتم" به عنوان یک ویژگی یکسان در نظر گرفته می‌شوند. تا حد ممکن، جدانویسی کلمات رعایت شده است تا بتوان پیشوندها را از کلمات جدا کرد. همچنین می‌توان ویژگی‌هایی را که در آنها میزان نسبی تغییرات از یک حد آستانه کمتر است، حذف کرد. این کار را در تابع RemoveUseless.m در محیط Matlab انجام داده‌ایم و اجرای این تابع باعث کاهش اندازه دادگان می‌شود. اما کمی بر روی دقت سیستم تاثیر منفی می‌گذارد و بنابر این از آنجایی که حجم دادگان در این لحظه برای ما مهم نیست، آن را در آزمایشات نیاورده‌ایم.

روش دیگری که به منظور تخمین تعلق یک جمله به احساس‌های گوناگون به کار برده‌ایم، استفاده از گنجواژه[8] است. گنجواژه یا تزاروس شامل مدخل‌های مختلفی می‌باشد که در هر مدخل آن، کلمات مترادف و مرتبط با آن مدخل قرار گرفته‌اند. برای هر یک از احساس‌های مورد نظر، یک لیست از کلمات مترادف درون گنجواژه تهیه شده است. در هنگام تست، با دیدن هر یک از کلمات درون گنجواژه هر احساس، یک امتیاز به آن احساس اضافه می‌کنیم. سپس در پایان، امتیازات بدست آمده را به مجموع امتیازات تقسیم می‌کنیم تا بتوان به آنها به دید احتمالاتی نگاه کرد. نمای کلی برنامه تعیین احساس در شکل ۱ آمده است.

به منظور آموزش مدل، از درخت فازی FID3 [۵] پیاده‌سازی شده در محیط Matlab نیز استفاده شد. برای فازی‌سازی[9] هر ویژگی از ۲ خوشه استفاده شد. پس از تست مدل توسط چند جمله، نتایج منطقی به نظر می‌رسیدند. اما به دلیل ابعاد بالای دادگان و محاسبات سنگین فازی، استفاده از این روش معقول نیست و بار محاسباتی زیادی دارد. بنابر این از درج نتایج این آزمایش خودداری گردید.

همچنین ذکر این نکته لازم است که به دلیل کوچک بودن نمونه‌های آموزشی (در حد یک جمله)، استفاده از روش‌هایی نظیر Tf–idf[10] توجیهی نداشت. چرا که این روش‌ها نیازمند نمونه‌هایی در اندازه چند پاراگراف هستند.

۳- آزمایش‌ها

داده‌های مورد استفاده در این تحقیق، دارای ۶ کلاس هستند که از پیکره PersEmotionalSentV1 تهیه شده در آزمایشگاه پردازش هوشمند داده‌های چندرسانه‌ای، دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر استفاده شده است. مشخصات این داده‌ها در جدول ۱ ارائه شده است.

تعدادی از آزمایش‌های صورت گرفته به همراه نتایج در جدول ۲ آمده است. این آزمایش‌ها با استفاده از تمامی داده‌ها، عدم استفاده از لیست کلمات توقف (درنظر گرفتن تمامی ویژگی‌ها و عدم حذف ویژگی‌های توقف)، و استفاده از هموارسازی ۰/۰۱ برای روش مبتنی بر پیکره انجام شده است. به ازای هر نمونه، محتمل ترین احساس به صورت رنگی نمایش داده شده است. در مواردی که احساس دارای بیشترین احتمال به درستی تعیین شده باشد، رنگ سبز، و در غیر این موارد، رنگ قرمز لحاظ شده است.

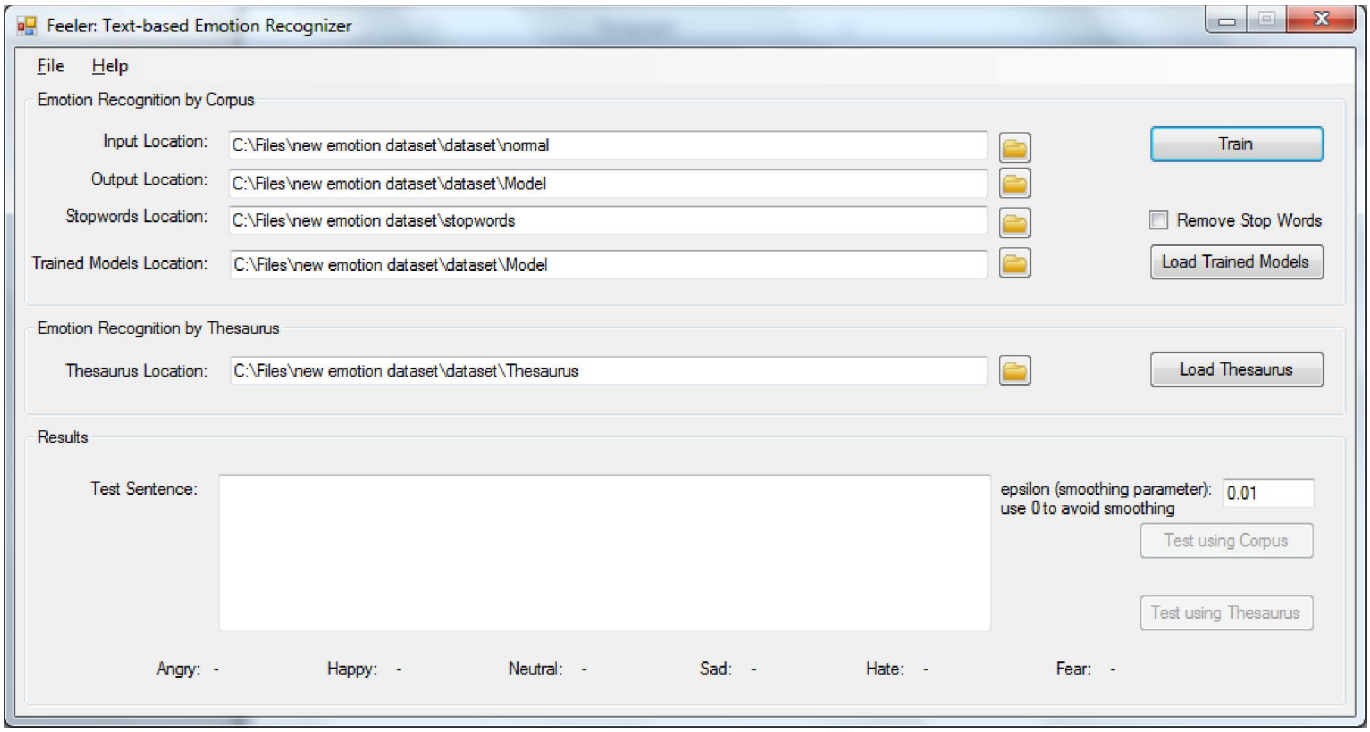
جدول ۱: مشخصات داده‌ها

نام کلاس	تعداد نمونه ها
خوشحالی	۵۹۵
عصبانیت	۷۴۲
خنثی	۴۱۱
ناراحتی	۱۱۲۰
تنفر	۲۶۴
ترس	۵۷۰
مجموع	۳۷۰۲
میانگین	۶۱۷
انحراف معیار (با بایاس)[11]	۲۷۰/۳

در جدول ۳، همین تنظیمات وجود دارد؛ با این تفاوت که از لیست کلمات توقف در فاز آموزش استفاده شده است. مشاهده می‌شود که استفاده از لیست کلمات توقف، در اکثر موارد منجر به کاهش دقت سیستم شده است. با این حال، این کار باعث می‌شود تا حجم داده ذخیره شده در حدود ۳۰ درصد کاهش یابد. این امر، در جاهایی که میزان فضای مورد نیاز برای برنامه مهم است (مانند ریزپردازنده‌های نهفته[12])، به کار می‌آید. البته کاهش دقت که در اینجا مشاهده شده، به ازای داده‌های آزمایشی مورد نظر است و محتمل‌ترین کلاس نیز در تمامی موارد تغییر نکرده است؛ بلکه احتمال آن کاهش یافته است. نتیجه آزمایش در محیط Weka نشان می‌دهد که انتخاب غیر احتمالاتی کلاس، برای حالتی که از لیست کلمات توقف استفاده شود، چند درصد بالاتر از حالتی است که از تمامی ویژگی‌ها استفاده گردد.

در جدول ۴، نتایج پس از استفاده از روش مبتنی بر تزاروس آمده است. مشاهده می‌شود که دقت روش مبتنی بر تزاروس، به مراتب پایین‌تر از روش مبتنی بر پیکره می‌باشد و در اکثر موارد حرفی برای گفتن ندارد. چرا که از کلمات محدودی در تولید قوانین خود استفاده می‌کند. در مقایسه این دو روش، می‌توان روش مبتنی بر پیکره را یک روش احتمالاتی، و روش مبتنی بر تزاروس را یک روش نسبتا قانونمند[13] در نظر گرفت. البته روش

[8] Thesaurus
[9] Fuzzification
[10] Term Frequency – Inverse Document Frequency
[11] Biased Standard Deviation
[12] Embedded Microprocessors
[13] Rule-based



شکل ۱: نمای کلی برنامه تعیین احساس

مبتنی بر تزاروس مزایایی هم دارد. به طور مثال، در این روش، مدل زبانی فراتر از unigram است و term های مورد مطابقت می‌توانند بزرگتر از یک کلمه باشند. همچنین مزیت دیگر، سرعت بالای این روش در فاز آموزش است. چرا که عملا این کار (آموزش) در زمان تهیه تزاروس انجام شده است و کافی است که کلمات تزاروس در یک لیست از رشته‌ها بارگذاری شود.

به منظور بهبود روش مبتنی بر تزاروس، به جای مطابقت کامل، برنامه در متن آزمایشی به دنبال کلیدهای تزاروس می‌گردد که متن، حاوی آنها

جدول ۱: چند نمونه از نتایج تست بر روی روش مبتنی بر پیکره بدون استفاده از لیست کلمات توقف

جمله تست	عصبانیت	خوشحالی	خنثی	ناراحتی	تنفر	ترس
من امروز خیلی خوشحال هستم!	۰	۰.۹۹۵	۰	۰.۰۰۵	۰	۰
دیروز به مدرسه رفتم.	۰.۰۰۳	۰.۰۸۸	۰.۹۰۳	۰.۰۰۲	۰.۰۰۱	۰.۰۰۳
این کار ها باعث شد تا از تو دلخور شوم.	۰.۰۲۷	۰	۰	۰.۹۷۲	۰	۰
مگر من صد بار به تو نگفته بودم؟!	۰.۵۹۷	۰.۰۰۱	۰	۰.۴۰۱	۰.۰۰۱	۰
او دارد مرا ترک می کند.	۰	۰	۰	۱	۰	۰
شب قدری دیگه میترکونی ولی یه هفته بعد دوباره...	۰	۰.۰۰۲	۰	۰.۹۹۷	۰	۰
اصلا هیچ استرسی ندارم	۰.۲۳۷	۰.۰۰۳	۰.۲۷۴	۰.۳۷۶	۰.۰۲۵	۰.۰۸۵
دیشب کابوس دیدم.	۰.۰۴۴	۰	۰.۰۰۳	۰.۱۳۱	۰	۰.۸۲۲
دیگر نمی خواهم ببینمت.	۰.۰۸۶	۰.۴۱۴	۰.۰۰۴	۰.۲۲۳	۰.۲۶۹	۰.۰۰۴
ازت بدم میاد.	۰.۰۰۲	۰	۰	۰	۰.۹۹۷	۰.۰۰۱

باشد. به طور مثال، کلمه "غم" در تزاروس مربوط به احساس ناراحتی قرار دارد. اگر در متن آزمایشی، این کلمه عینا ظاهر شود و یا کلمه‌ای ظاهر شود که حاوی این کلید است (مانند "غم بار"، "غمناک" یا "غمگین")، به عنوان کلمه ای از احساس ناراحتی تلقی می‌شود.

به منظور آزمایش پیکره تولید شده به صورت غیر احتمالاتی (اختصاص یک برچسب به نمونه آزمایشی بدون لحاظ کردن میزان تعلق)، تعداد ۲۰۰ نمونه از هر کلاس را انتخاب کردیم و با روش‌های مختلف، توسط نرم‌افزار Weka اقدام به دسته‌بندی آنها نمودیم. انتخاب تنها ۲۰۰ کلمه از هر کلاس، به این دلیل است که انحراف معیار تعداد نمونه‌های کلاس‌ها تا حد ممکن کم شود. با افزایش تعداد نمونه‌ها به ۴۰۰ نمونه در هر کلاس، شاهد افت نتایج در حدود ۲٪ بودیم که به دلیل به هم خوردن نظم بین کلاس‌ها و بالا رفتن انحراف معیار

جدول ۲: چند نمونه از نتایج تست بر روی روش مبتنی بر پیکره با استفاده از لیست کلمات توقف

جمله تست	عصبانیت	خوشحالی	خنثی	ناراحتی	تنفر	ترس	بهبود
من امروز خیلی خوشحال هستم!	۰.۰۰۳	۰.۹۸۷	۰.۰۰۱	۰.۰۰۴	۰.۰۰۱	۰.۰۰۴	خیر
دیروز به مدرسه رفتم.	۰.۰۰۲	۰.۰۹۰	۰.۹۰۱	۰.۰۰۱	۰.۰۰۲	۰.۰۰۴	خیر
این کار ها باعث شد تا از تو دلخور شوم.	۰.۱۴۱	۰.۱۷۰	۰	۰.۷۰۰	۰.۱۴۱	۰	خیر
مگر من صد بار به تو نگفته بودم؟!	۰.۸۸۵	۰.۰۰۸	۰	۰.۰۹۸	۰.۰۰۸	۰	بله
او دارد مرا ترک می کند.	۰.۰۱۰	۰.۰۱۰	۰.۰۱۰	۰.۹۵۰	۰.۰۱۰	۰.۰۱۰	خیر
شب قدری دیگه میترکونی ولی یه هفته بعد دوباره...	۰.۰۰۱	۰.۱۹۸	۰.۰۱۱	۰.۷۹۰	۰	۰	خیر
اصلا هیچ استرسی ندارم	۰.۲۳۷	۰.۰۰۳	۰.۲۷۴	۰.۳۷۶	۰.۰۲۵	۰.۰۸۵	-
دیشب کابوس دیدم.	۰.۰۴۴	۰	۰.۰۰۳	۰.۱۳۱	۰	۰.۸۲۲	-
دیگر نمی خواهم ببینمت.	۰.۰۴۰	۰.۵۴۲	۰.۰۰۴	۰.۰۷۲	۰.۳۳۸	۰.۰۰۵	خیر
ازت بدم میاد.	۰.۰۳۸	۰	۰.۰۰۹	۰.۰۰۲	۰.۹۳۱	۰.۰۱۹	خیر

جدول ۳: چند نمونه از نتایج تست بر روی روش مبتنی بر تزاروس

جمله تست	عصبانیت	خوشحالی	خنثی	ناراحتی	تنفر	ترس	بهبود
من امروز خیلی خوشحال هستم!	۰	۱	۰	۰	۰	۰	بله
دیروز به مدرسه رفتم.	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	خیر
این کار ها باعث شد تا از تو دلخور شوم.	۰	۰	۰	۰.۵۰۰	۰.۵۰۰	۰	خیر
مگر من صد بار به تو نگفته بودم؟!	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	خیر
او دارد مرا ترک می کند.	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	خیر
شب قدری دیگه میترکونی ولی یه هفته بعد دوباره...	۰	۰	۰	۱	۰	۰	بله
اصلا هیچ استرسی ندارم	۰	۰	۰	۰	۰	۱	خیر
دیشب کابوس دیدم.	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	خیر
دیگر نمی خواهم ببینمت.	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	۰.۱۶۷	بله
ازت بدم میاد.	۰	۰	۰	۰.۵۰۰	۰.۵۰۰	۰	خیر

می‌باشد. این مسئله را می‌توان با افزودن نمونه‌های کافی به کلاس‌هایی با داده کمتر مرتفع کرد.

تمامی آزمایش‌ها با استفاده از 10-fold cross validation انجام شده‌اند و در آنها (به جز آزمایش آخر که در آن از لیست کلمات توقف به منظور حذف ویژگی‌ها استفاده شده است) از تمامی ویژگی‌ها به منظور

دسته‌بندی بهره برده‌ایم. بهترین نتیجه مربوط به روش Complement Naïve Bayes Classifier است که دارای دقت ۷۸/۸۵٪ و زمان بسیار کوتاه آموزش می‌باشد. رتبه دوم نیز مربوط به روش SVM است که دارای دقت ۷۶/۶۰٪ و زمان آموزش طولانی‌تر است.

۴- نتیجه‌گیری و روال آتی

در این مقاله یک پیکره فارسی حاوی ۳۷۰۲ جمله از ۶ احساس خوشحالی، عصبانیت، خنثی، ناراحتی، تنفر و ترس را ارائه کردیم و با استفاده از روش‌های پردازش متون و سایر روش‌های گوناگون دسته بندی، اقدام به تعیین احساس حاکم بر جملات فارسی نمودیم. با بررسی نتایج بدست آمده متوجه می‌شویم که دقت برنامه در صورت استفاده از رویکرد مبتنی بر پیکره و به کار بردن هموارسازی در احتمالات، مطلوب است. البته این تنظیمات در شرایط خاصی نیز ممکن است دچار خطا شود. پس از استفاده از کلمات توقف متوجه شدیم که نتایج افت پیدا کردند. این بدین دلیل است که بعضی از ویژگی‌های مناسب نیز درون این کلمات بودند و با حذف آنها، اطلاعات اولیه در مورد کلاس‌های احساس کمتر شده است.

به منظور بهبود بیشتر نتایج می‌توان از برچسب‌های اجزاء کلام[۱۴] استفاده کرد. به طور مثال، وزن اسامی را صفر یا یک عدد کوچک در نظر بگیریم و وزن صفت‌ها را بیشتر کنیم. همچنین می‌توان از یک ریشه‌یاب[۱۵] استفاده کرد و از این طریق، ریشه‌های کلمات را به عنوان ویژگی‌ها به سیستم بدهیم تا با دیدن یک کلمه در دادگان، کلمات شبیه به آن نیز به صورت ضمنی آموزش دیده شوند.

نهایتا به دلیل رویکرد احتمالاتی به کار گرفته شده، هرچه داده‌های آموزشی بیشتر باشد، نتیجه بدست آمده مطلوب‌تر خواهد بود. رویکرد مبتنی بر تزاروس دارای سرعت بالا و دقت پایینی است که استفاده مجزا از آن را ناموجه جلوه می‌دهد. اما می‌توان از کلمات درون آن به عنوان اهدافی در روش مبتنی بر پیکره استفاده کرد که منجر به پاداش به یک احساس می‌شوند. به عبارت دیگر می‌توانیم این رویکرد را با وزن کمی با روش مبتنی بر پیکره ترکیب کنیم. البته به نظر می‌رسد که اگر پیکره به اندازه کافی بزرگ و جامع باشد، نیاز به تزاروس به صورت خودکار برطرف می‌شود.

پراکندگی مناسب بین کلاس‌ها نیز تاثیر زیادی در دقت سیستم دارد. در دادگان این پروژه، پراکندگی زیادی بین تعداد نمونه‌ها در کلاس‌های مختلف وجود دارد که در صورت برطرف شدن، منجر به افزایش دقت سیستم می‌شود.

در آینده قصد داریم تا دادگان را از لحاظ کمی و کیفی بهبود دهیم. این بهبود، شامل افزودن نمونه‌های جدید به دادگان، کاهش واریانس بین تعداد نمونه‌های موجود در کلاس‌های مختلف و برطرف کردن غلط‌ها و ناهمگونی‌های نگارشی در نمونه‌ها می‌باشد. اقدام دیگری که در آینده می‌توان انجام داد، ویرایش و بهبود لیست کلمات توقف است تا استفاده از این لیست، به جدایی‌پذیری بین کلاس‌های احساس کمک کند و موجب بهبود کیفیت کلی سیستم شود.

هدف بعدی برای بهبود نتایج، ارائه روشی جهت انتخاب ویژگی‌های مناسب از میان تمامی کلمات منحصر به فرد درون دادگان به نحوی است که حجم اطلاعات مورد پردازش در حدود ۵۰٪ کاهش یابد شود و دقت سیستم حفظ شود.

مراجع

[1] T. Danisman, A. Alpkocak, "Feeler: Emotion classification of text using vector space model", AISB 2008 Convention Communication, Interaction and Social Intelligence, vol. 1, pp. 53-59, 2008.

[2] C. H. Wu, Z. J. Chuang, et al. (2006). "Emotion recognition from text using semantic labels and separable mixture models", ACM Transactions on Asian Language Information Processing (TALIP) 5(2): 165-183.

[3] C. O. Alm, D. Roth, et al. (2005). "Emotions from text: machine learning for text-based emotion prediction", Association for Computational Linguistics.

[4] J. T. Hancock, C. Landrigan, et al. (2007). "Expressing emotion in text-based communication", ACM.

[5] J. Rives, (1990). "FID3: fuzzy induction decision tree", Uncertainty Modeling and Analysis, Proceedings on First International Symposium, 1990.

[14] Part of Speech (POS) Tags

[15] Stemmer