ارائهٔ یک رتبه‌بند برای خطایاب معنایی با استفاده از ویژگی‌های حساس به متن

بهزاد میرزابابایی و هشام فیلی
دانشکدهٔ مهندسی برق و کامپیوتر، پردیس دانشکده‌های فنی، دانشگاه تهران، تهران، ایران

چکیده

در عصر فناوری، روزانه حجم زیادی از سندهای الکترونیکی تولید می‌شود. ازآنجاکه این سندها توسط افراد مختلف تولید می‌شود، دارای خطاهایی هستند. وجود خطاها باعث کاهش کیفیت سندها می‌شود؛ بنابراین وجود ابزارهای خطایاب باعث افزایش کیفیت می‌شود. یکی از انواع خطاها، خطای معنایی حساس به متن است. همان‌طور که از نام آن برمی‌آید، برای تشخیص و تصحیح آن، نیاز به تحلیل اطلاعات موجود در متن است. در این مقاله، یک رتبه‌بند متمایزگر مستقل از زبان برای خطایاب‌های معنایی حساس به متن ارائه دادیم و از اطلاعات کل متن برای رتبه‌بندی استفاده کردیم. موجود بودن جمله‌های قبلی و بعدی جملهٔ خطادار یکی از پیش‌نیازهای روش ارائه شده است. این رتبه‌بندی توسط ویژگی‌های حساس به متن و یک مدل لگاریتم خطی انجام شده است. برای ارزیابی روش، از دو روش مبنای مختلف که یکی بر اساس مترجم ماشینی آماری و دیگری بر اساس مدل زبانی است، استفاده کرده‌ایم. به‌منظور ارزیابی سامانه از دو دادهٔ آزمون مختلف در زبان فارسی استفاده شده است. این روش باعث بهبود ۱۷٪ در بازخوانی تشخیص و تصحیح نسبت به روش مبنای مترجم ماشینی آماری شده است.

واژگان کلیدی: خطایاب، خطای حساس به متن، مترجم ماشینی آماری، رتبه‌بندی آگاه به متن

۱- مقدّمه

با پیشرفت فناوری اطلاعات، روزانه حجم زیادی از سندهای الکترونیکی تولید می‌شود. این سندها شامل روزنامه‌ها، مجله‌ها، کتاب‌های الکترونیکی و رایانامه‌ها هستند. از آنجایی که این سندها توسط انسان تولید شده‌اند، همواره شامل خطاهای نوشتاری هستند و وجود این خطاها در متن باعث کاهش کیفیت آنها می‌شود. با افزایش حجم تولید سندها و افزایش سرعت تولید آن‌ها، خطاهای تولیدشده نیز افزایش یافته و باعث کاهش کیفیت سندها می‌شود.

به‌منظور کاهش خطاهای ایجادشده در متن و افزایش کیفیت سندها می‌توان از ابزارهای مربوط به پردازش زبان طبیعی به‌خصوص خطایاب‌های خودکار استفاده کرد. خطایاب‌ها یکی از ابزارهای پرکاربرد در پردازش زبان طبیعی هستند. هر خطایاب خودکار متشکل از سه مرحله اصلی است. این سه مرحله عبارت‌اند از، تشخیص خطا در متن، تولید نماینده‌های مناسب برای هر خطا و در آخر، مرتب‌کردن نامزدها به‌منظور یافتن محتمل‌ترین نامزد. خطاهای موجود در متن به پنج دسته تقسیم شده است (کوکیچ، ۱۹۹۲):

- خطای املایی[1]: این نوع خطا، به نوعی از خطای نوشتاری گفته می‌شود که لغت خطا در لغت‌نامه یافت نشود. برای مثال، لغت اشتباه "دسن" که به‌جای لغت درست "دست" نوشته شده است، یک خطای املایی است..
- خطای نحوی[2]: این نوع خطا که به آن خطای گرامری[3] هم گفته می‌شود، شامل لغات و یا

[1] Isolated error
[2] Syntactic error
[3] Grammatical error

عباراتی هستند که قوانین نحوی و ساخت‌واژی[1] زبان را نقض می‌کنند. برای مثال، عدم تطابق فعل با فاعل، "شما رفت."، جابه‌جایی صفت و موصوف، "زیبا درخت" و جمع بستن کلمه‌ها به‌صورت اشتباه، "سبزیات"، از این نوع خطا هستند.

- خطای معنایی[2]: خطای معنایی یا خطای حساس به متن[3] نوعی خاص و پیچیده‌تر از خطاهای املایی هستند. در این نوع خطا، لغتی که به‌اشتباه نوشته شده یک لغت معنادار است. برای مثال، اگر لغت "دست" به‌اشتباه به‌صورت "دشت" نوشته شود، یک خطای حساس به متن رخ داده است.
- خطاهای ناشی از ساختار گفتمان[4]: این نوع خطا به عدم هماهنگی در جمله مربوط می‌شود. برای مثال، اشتباه در شمارش یکی از این موردهاست. در جملهٔ "قرمز، سبز و آبی چهار رنگ اصلی هستند."، یک خطای ناشی از ساختار رخ داده است.
- خطای مربوط به واقعیت[5]: این نوع از خطا مربوط به عدم دانش نویسنده است. به‌عنوان مثال در جملهٔ "پایتخت کشور ایران شهر شیراز است." یک خطای مربوط به واقعیت رخ داده است.

در این مقاله، تمرکز ما روی خطای نوع سوم (خطای معنایی) است. نام دیگر این نوع خطا، خطای حساس به متن است. همان‌طور که از نام آن مشخص است، برای تشخیص و تصحیح این نوع خطا تحلیل وابسته به متن نیاز است.

روش‌های مختلفی را که روی این نوع خطا کار کردند، می‌توان از نظر میزان متنی که برای تحلیل در نظر گرفته‌اند و نحوهٔ تحلیل آن متن، به دسته‌های مختلف تقسیم کرد. در بسیاری از کارهای پیشین، یک پنجرهٔ متنی[6] محدودی اطراف کلمه‌ای که خطا تشخیص داده شده است در نظر می‌گیرند و سپس متنی را که در این پنجره قرار می‌گیرد با ویژگی‌هایی آماری و زبانی مختلفی مثل چندکلمه‌ای[7] (بسیل و الوانی، ۲۰۱۲) و برچسب اجزای کلام[8] (گلدینک و شبز، ۱۹۹۶) بررسی و تحلیل می‌کنند.

[1] Morphological
[2] Semantic error
[3] Context-sensitive error
[4] Discourse structure error
[5] Pragmatic error
[6] Context window
[7] N-gram
[8] Part of speech tag

در این مقاله، میزان متنی که برای تحلیل در نظر گرفته‌ایم برابر است با کل جملهٔ ورودی به همراه تمام کلمه‌های کلیدی متنی که جملهٔ ورودی از آن استخراج شده است.

در اینجا ما یک روش رتبه‌بندی متمایزگر مستقل از زبان برای خطایابی حساس به متن ارائه داده‌ایم. این رتبه‌بندی توسط یک روش لگاریتم خطی انجام می‌شود. ما به ازای هر جملهٔ ورودی یک فهرست که شامل n عدد از بهترین نامزد[9] درست احتمالی است، تولید و سپس فهرست مورد نظر را به‌کمک ویژگی‌های آگاه به متن[10] رتبه‌بندی مجدد کردیم. روشی که ما ارائه دادیم، یک روش مستقل از زبان است و علاوه‌بر یک خطایاب مستقل، می‌توان به‌عنوان یک پس‌پردازش روی خروجی هر خطایاب حساس به متن، به آن نگاه کرد. از این رو این روش را می‌توان روی هر زبان و هر خطایابی که یک فهرست به‌ازای هر جمله ورودی بازمی‌گرداند، اعمال کرد و نتیجهٔ آن را بهبود داد. به‌منظور ارزیابی روش مورد نظر، پس‌پردازش را روی دو سامانهٔ خطایاب معنایی مبتنی بر مترجم ماشینی آماری (SMT)[11] (احسان و فیلی، ۲۰۱۳) و روش[12] MDM (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) و همچنین روی دو دادهٔ آزمون مختلف در زبان فارسی، شامل خطاهای واقعی[13] و خطاهای مصنوعی ارزیابی کردیم.

۲-کارهای مشابه

خطای واقعی یا حساس به متن در دستهٔ سوم طبقه‌بندی مطرح شده در (کوکیچ، ۱۹۹۲) قرار می‌گیرد. تحقیق‌های زیادی در مورد اهمیت این نوع خطا و میزان رخداد آن انجام شده است. به‌عنوان مثال (میلتون، ۱۹۸۷) در مطالعه‌ای روی ۹۲۵ مقاله نوشته‌شده توسط دانش آموزان دبیرستانی متوجه شد که چهل درصد خطاهای نوشتاری آن‌ها خطای حساس به متن است.

تاکنون روش‌های مختلفی برای تشخیص و تصحیح این نوع خطا انجام شده است که اغلب آنها روش‌های آماری بوده‌اند (احسان و فیلی، ۲۰۱۳؛ ویلکاگس-اُهرن و همکاران، ۲۰۰۸). روش‌های آماری از ویژگی‌های زبانی مختلف و همچنین از طبقه‌بندهای مختلفی استفاده می‌کنند. به‌عنوان

[9] N-best list
[10] Discourse-aware
[11] Statistical machine translation (SMT)
[12] Mays, Damerau and Mercer
[13] Real-world

مثال برخی روش‌ها از چندکلمه‌ای‌ها (بسیل و الوانی، ۲۰۱۲) و برخی از برچسب اجزای کلام (گلدینک و شبز، ۱۹۹۶) استفاده می‌کنند. روش‌هایی هم هستند که از طبقه‌بند بیزی[1] (گیل و همکاران، ۱۹۹۲)، فهرست‌های تصمیم[2] (یاروفسکی، ۱۹۹۴) و مترجم ماشینی آماری (احسان و فیلی، ۲۰۱۳) استفاده می‌کنند.

در (بسیل و الوانی، ۲۰۱۲) از چندکلمه‌ای گوگل[3] به‌عنوان یک منبع آماری برای تشخیص و تصحیح خطای حساس به متن استفاده کردند. این مجموعه داده شامل چندکلمه‌ای‌های استخراج‌شده از اینترنت است. طول پنجرهٔ متن در این روش حداکثر پنج لغت است. در این روش، از ۳-گرام برای تشخیص و از ۵-گرام برای تصحیح استفاده شده است.

در (گلدینک و شبز، ۱۹۹۶) از چندکلمه‌ای برچسب اجزای کلام و طبقه‌بند بیز برای تشخیص و تصحیح خطا استفاده کرده‌اند. طول پنجره متن در این روش سه لغت است. این روش بیست درصد از پیکره Brown را به‌عنوان دادهٔ آزمون و مابقی را به‌عنوان داده آموزشی در نظر گرفته است و همچنین تعداد کمی لغت را برای تشخیص و تصحیح پوشش داده‌اند.

پژوهش (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) یک بازنگری و تحلیل از (مِی و همکاران، ۱۹۹۱) انجام داده است. آنها اندازه‌های متفاوتی برای پنجرهٔ متن در نظر گرفتند که عبارت‌اند از ۶، ۱۰ و ۱۴ کلمه اطراف کلمهٔ مشکوک. به‌منظور تحلیل لغت‌های درون پنجره متن از مجموعهٔ ۳-گرام لغت‌های درون پنجره استفاده کردند. از آنجا که این روش یکی از روش‌های مبنای ماست، در زیربخش ۳-۳ با جزییات بیشتر به آن می‌پردازیم.

در سال‌های اخیر خطایاب‌هایی هم برای زبان فارسی ارائه شده است، مانند (احسان و فیلی، ۲۰۱۳؛ کاشفی و همکاران، ۲۰۱۳؛ میانگاه، ۲۰۱۳). در (کاشفی و همکاران، ۲۰۱۳) یک معیار جدید بر اساس فاصلهٔ نوشتاری لغت‌ها در زبان فارسی ارائه شده است. این فاصله بین کلمه‌ها بر اساس فاصلهٔ حرف‌ها در صفحه‌کلید و اشتباه‌های املایی ناشی از تایپ، محاسبه می‌شود. از این معیار فاصله برای رتبه‌بندی گزینه‌های پیشنهادی خطایاب استفاده شده است.

[1] Bayes
[2] Decision lists
[3] Google web 1T n-gram

خطایاب املایی دیگری برای زبان فارسی توسط (میانگاه، ۲۰۱۳) ارائه‌شده است. این خطایاب از سه مرحلهٔ مجزای تشخیص خطا، تولید نامزدهای مناسب هر کلمهٔ اشتباه و در آخر، رتبه‌بندی نامزدها تشکیل شده است. در این مقاله سعی شده است تأثیر پیکرهٔ بزرگ بر روی کیفیت خطایاب املایی بررسی شود.

یکی دیگر از روش‌هایی که برای خطایابی استفاده شده، روش مترجم ماشینی آماری است. این روش یک روش مستقل از زبان است و می‌تواند در خطایابی بر روی زبان‌های مختلف استفاده شود. در (احسان و فیلی، ۲۰۱۳) از این روش برای تشخیص و تصحیح خطای حساس به متن و نحوی در زبان انگلیسی و فارسی استفاده شده است. این روش نیز، مانند روش (ویلکاگس-اُهرن و همکاران، ۲۰۰۸)، به‌عنوان روش مبنا در نظر گرفته شده است و در زیربخش ۳-۳ با جزییات بیشتر توضیح داده خواهد شد.

۳- مدل رتبه‌بند متمایز کننده

در این مقاله، به‌منظور تشخیص و تصحیح خطاهای حساس به متن یک روش رتبه‌بندی بر اساس ویژگی‌های آگاه به متن ارائه شده است. روند کلی کار به این صورت است که در ابتدا برای هر جملهٔ مشکوک در دادگان آزمون یک فهرست مرتب‌شده از نامزدها مانند روش‌های مطرح‌شده در (احسان و فیلی، ۲۰۱۳؛ ویلکاگس-اُهرن و همکاران، ۲۰۰۸) تولید می‌شود؛ سپس جمله‌ای که در رتبه یک این فهرست‌ها قرار می‌گیرد، به‌عنوان جواب پیشنهادی روش مبنا در نظر گرفته می‌شود. در ادامه به‌منظور بهبود فهرست‌ها و قراردادن جواب صحیح در رتبه‌های بهتر، نامزدهای فهرست مورد نظر توسط سه ویژگی حساس به متن رتبه‌بندی مجدد می‌شوند.

اولین ویژگی مدل زبانی[4] (LM) است که یک ویژگی مناسب برای نشان‌دادن وابستگی‌های کوتاه[5] است. دو ویژگی‌های دیگری که برای باز رتبه‌بندی در نظر گرفتیم، به‌نوعی وابستگی یا هم‌رخدادی[6] بین لغت‌های جمله و هم‌رخدادی بین لغت‌های جمله و کلمه‌های کلیدی متن را نشان می‌دهد و همچنین مناسب برای تحلیل وابستگی دور[7] بین کلمات است. این ویژگی‌ها در زیربخش ۳-۱ توضیح داده می‌شوند.

[4] Language model
[5] Short distance dependency
[6] Co-occurrence
[7] Long distance dependency

۳-۱- ویژگی‌های آگاه به متن

برای تحلیل حساس به متن ابتدا بایستی به‌ازای هر کلمه مشکوک، یک پنجرهٔ متنی از لغت‌های اطراف آن را به‌عنوان یک محتوا تعریف کنیم که در مقاله دو نوع پنجره با ابعاد مختلف به‌عنوان محتوا در نظر گرفته شده است. اندازهٔ پنجرهٔ اوّل کل جملهٔ ورودی است و با ویژگی‌های مدل زبانی و اطلاعات متقابل نقطه‌ای (PMI)[1] بررسی و تحلیل می‌شود و پنجرهٔ دیگر کلمه‌های کلیدی کل متنی است که جملهٔ ورودی از آن استخراج شده است و توسط PMI بررسی می‌شود. PMI دو لغت A و B طبق فرمول (۱) محاسبه می‌شود:

$$\mathrm{PMI(A,B)}=\frac{\mathrm{co_occurrence(A,B)}}{\mathrm{Doc.Count(A)} \times \mathrm{Doc.Count(B)}} \qquad (۱)$$

در فرمول (۱)، Doc.Count(A) برابر با تعداد سندهایی هست که لغت A در آن آمده باشد. co_occurrence(A,B) برابر است با تعداد سندهایی که هر دو لغت A و B در آن آمده باشد. بر اساس فرمول (۱) دو ویژگی مختلف بر اساس طول پنجرهٔ محتوا تعریف می‌کنیم. ویژگی اوّل بر اساس پنجرهٔ محتوایی است که شامل لغت‌های کلیدی کل متن است. این ویژگی را به‌خاطر اندازهٔ پنجرهٔ محتوا، $\mathrm{PMI_{doc}}$[2] می‌نامیم. $\mathrm{PMI_{doc}}$ برای هر جملهٔ نامزد محاسبه می‌شود و برابر است با میانگین PMI بین تمام لغت‌های جمله و تمام کلمه‌های کلیدی متن. برای محاسبهٔ $\mathrm{PMI_{doc}}$ ، ابتدا باید کلمه‌های کلیدی متنی را که جملهٔ ورودی از آن استخراج شده است، یافت. بدین منظور از روش فراوانی لغت[3] و معکوس فراوانی سند[4] استفاده شده است. معکوس فراوانی سند از روی تمام سندهای جمله‌های دادهٔ آزمون محاسبه شده است. به‌ازای هر سند، فراوانی لغت و معکوس فراوانی سند را برای تمام لغت‌های غیردستوری محاسبه بر اساس اهمیت مرتب شده است. در آخر برای هر سند، پنجاه لغت پراهمیت‌تر را به‌عنوان کلمه‌های کلیدی انتخاب کرده‌ایم. برای نشان‌دادن اهمیت کلمه‌های کلیدی، فرض کنید اگر جملهٔ سادهٔ "برق آمد." یک نامزد برای جملهٔ ورودی "برف آمد." باشد، با استفاده از کلمه‌های کلیدی متنی که جملهٔ ورودی از آن استخراج شده است، می‌توان نامزد مناسب را انتخاب کرد.

اگر W یک جملهٔ n کلمه‌ای از سند D باشد و سند D شامل ۵۰ کلمه کلیدی $KeyWords_D = \{k_{D1}, k_{D2}, \dots, k_{D50}\}$ باشد، آنگاه $\mathrm{PMI_{doc}}$ برای جملهٔ $S_j = \{w_{j1}, w_{j2}, \dots, w_{jn}\}$، که برابر با j-امین نامزد تولیدشده برای جملهٔ W توسط روش مبنا است، از رابطه (۲) محاسبه می‌شود:

$$PMI_{doc}(S_j) = \frac{\sum_{i=1}^{n}\sum_{m=1}^{50} PMI(w_{ji};k_{Dm})}{n*50} \qquad (۲)$$

درواقع این رابطه برابر میانگین مجموع PMI لغت‌های جمله با کلمه‌های کلیدی است. در رابطهٔ (۲)، n برابر تعداد کلمه‌های نامزد و k_{Dm} برابر است با کلمهٔ کلیدی m-ام از سند D و w_{ji} برابر با کلمهٔ i-ام از j-امین نامزدی جملهٔ ورودی W است. اگر k_{Dm} و w_{ji} هر دو یکسان باشند، مقدار $PMI(w_{ji};k_{Dm})$ کم می‌شود؛ درصورتی‌که وجود یکی از کلمه‌های کلیدی در جملهٔ نامزد باید ارزش آن نامزد را بیشتر از مابقی نامزدها کند. بنابراین اگر k_{Dm} و w_{ji} هر دو یکسان باشند یا به عبارت دیگر w_{ji} یکی از لغت‌های کلیدی باشد، مقدار $PMI(w_{ji};k_{Dm})$ آن‌ها را برابر با بیشترین مقدار PMI در جملهٔ نامزد در نظر می‌گیریم.

همان‌طور که پیش‌تر گفته شده، PMI_{doc} برای تحلیل پنجرهٔ محتوایی به اندازه کل متن مناسب است و ارتباطی بین لغت‌های کلیدی متن و جملهٔ ورودی برقرار می‌کند. در بعضی مواقع جمله‌هایی در متن است که وابستگی کمتری با کل متن دارند و یا کلمه‌هایی در یک جمله هستند که وابستگی زیادی با کلمهٔ مورد نظر دارند، درصورتی‌که در فاصلهٔ دور از یکدیگر قرار می‌گیرند و با مدل زبانی قابل تشخیص نیست. بنابراین ویژگی دیگری بر اساس PMI تعریف کردیم که برای تحلیل پنجرهٔ محتوای به اندازهٔ طول جمله ورودی مناسب است. نام این ویژگی PMI_{sen}[5] است و برای هر جمله محاسبه می‌شود. PMI_{sen} برابر است با میانگین مجموع PMIها بین هر دو لغت در جملهٔ نامزد. اگر W یک جملهٔ n کلمه‌ای از داده آزمون باشد آنگاه PMI_{sen} برای جملهٔ $S_j = \{w_{j1}, w_{j2}, \dots, w_{jn}\}$ که برابر با j-امین نامزد تولیدشده برای جملهٔ W توسط روش مبنا است از رابطهٔ (۳) محاسبه می‌شود:

$$\mathrm{PMI_{sen}}(S_j)= \frac{\sum_{k=1}^{n}\sum_{m=k}^{n} \mathrm{PMI}(w_{jk};w_{jm})}{n*\frac{(n-1)}{2}} \qquad (۳)$$

[1] Point-wise mutual information
[2] $\mathrm{PMI}_{\text{document}}$
[3] Term frequency
[4] Inverse document frequency
[5] $\mathrm{PMI}_{\text{sentence}}$

درواقع این رابطه، میانگین PMI لغت‌های جمله با یکدیگر را محاسبه می‌کند. در رابطهٔ (۳)، n برابر تعداد کلمه‌های نامزد و w_{jk} برابر با کلمه‌های k-ام از j-امین نامزد جملهٔ ورودی W است.

به‌منظور نشان‌دادن اهمیت این دو ویژگی، مقدار این دو ویژگی را برای جمله‌های نامزد که توسط روش مبنای SMT تولیدشده در جدول (۱) نشان داده شده است. همان‌طور که از جدول (۱) مشاهده می‌شود، برای جملهٔ ورودی "دندان قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند .". شش نامزد توسط SMT تولید شده است بهترین مقدار PMI_{sen} و PMI_{doc} را پنجمین نامزد دارد که جواب صحیح مورد نظر نیز هست.

(جدول- ۱): شش جواب احتمالی تولیدشده توسط (احسان و فیلی، ۲۰۱۳) برای جملهٔ ورودی "دندان قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند". مقدارهای موجود در جدول به‌صورت لگاریتم است.

PMI_{doc}	PMI_{sen}	نامزد	صحیح/ خطا	رتبهٔ نامزد
-۷/۱۵۳	-۱۰/۸۹۰	*چندان* قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند .	خطا	اول
-۷/۱۵۰	-۱۰/۸۱۰	دندان قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند .	خطا	دوم
-۷/۱۵۴	-۱۰/۸۵۰	*زندان* قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند .	خطا	سوم
-۷/۱۵۲	-۱۰/۹۶۵	*مندان* قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند .	خطا	چهارم
-۷/۰۵۰	**-۹/۹۴۰**	**دزدان قوی هیکل دو متر از ریل راه آهن اوکراین را دزدیدند .**	**صحیح**	**پنجم**
-۷/۱۶۰	-۱۰/۷۶۳	دندان قوی هیکل دو مصر از ریل راه آهن اوکراین را دزدیدند .	خطا	ششم

چنانچه این فهرست را توسط ویژگی‌های حساس به متن (PMI_{sen} وPMI_{doc}) رتبه‌بندی کنیم، جواب صحیح در رتبهٔ اول قرار می‌گیرد.

ویژگی دیگری که برای تحلیل وابستگی کوتاه در نظر گرفتیم، LM است. این ویژگی برای مقایسه احتمال رخداد ۳-گرام‌های موجود جمله‌ها استفاده شده، بنابراین توسط تعداد رخداد ۳-گرام‌ها تخمین زده شده است (استولک و همکاران، ۲۰۰۲).

پس از محاسبهٔ مقدار ویژگی‌ها برای نامزدهای تولیدشده توسط روش‌های مبنا، نوبت به رتبه‌بندی مجدد آن‌ها توسط ویژگی‌های PMI_{sen}، PMI_{doc} و LM است. برای رتبه‌بندی از مدل لگاریتم خطی استفاده کرده‌ایم. برای انجام رتبه‌بندی مجدد توسط یک مدل لگاریتم خطی، وزن یا اهمیت هر ویژگی را باید به‌نوعی تخمین زد. در زیربخش بعدی نحوهٔ به‌دست‌آوردن وزن‌ها گفته‌شده است. به‌منظور استفاده از ویژگی PMI_{doc} علاوه‌بر جملهٔ ورودی اطلاعات کل متن نیز لازم است. وابستگی این ویژگی به متن اطراف جملهٔ ورودی را می‌توان محدودیتی برای این روش دانست.

۳-۲- وزن‌دهی ویژگی‌ها

برای وزن‌دهی ویژگی‌ها از ابزار SVM-light[1] (سُچانداریدیس و همکاران، ۲۰۰۵) استفاده شده است. SVM یک الگوریتم یادگیری ماشین بر اساس نظریهٔ یادگیری آماری است و در سال‌های اخیر به‌خصوص در توابع رگرسیون (جِنگ، ۲۰۰۵) و بازشناسی الگو (سای، ۲۰۰۵) استفادهٔ زیادی از آن شده است. به‌منظور استفاده از SVM، یک دادهٔ آموزشی شامل ۵،۰۰۰ جمله صحیح از پیکره‌های موجود فارسی استخراج شده است و در هر یک از این جمله‌ها یک خطای حساس به متن به‌طور مصنوعی وارد شده است؛ سپس با استفاده از روش (احسان و فیلی، ۲۰۱۳)، به‌ازای هر جمله، حداکثر بیست نامزد تولید شده است و به‌ازای هر نامزد، مقدار ویژگی‌های PMI_{sen}، PMI_{doc} و مدل زبانی محاسبه شده است. توضیح‌های بیشتر در زیربخش ۴-۳ آورده شده است. در کل ۵۶،۳۲۰ نامزد تولید شده است که ۳،۷۲۸ جمله از آنها صحیح است. پس از به‌دست‌آوردن وزن‌های مورد نظر، از یک مدل لگاریتم خطی برای رتبه‌بندی مجدد خروجی روش‌های مبنا استفاده شده است. مدل لگاریتم خطی طبق رابطه (۴) محاسبه می‌شود:

[1] Support vector machine

$$\log P(c|e)=\lambda_1 * \log PMI_{doc} + \lambda_2 * \log PMI_{sen} + \lambda_3 * \log LM \quad (۴)$$

در رابطهٔ (۴)، c جملهٔ نامزدی است که توسط روش مبنا تولید شده است و e جملهٔ ورودی به خطایاب است . λ_iها وزن ویژگی‌هایی هستند که توسط SVM محاسبه شده‌اند. در زیر بخش بعد، روش‌های مبنا را با جزئیات بیشتر توضیح داده خواهد شد.

۳-۳- روش‌های مبنا

دو روش ارائه‌شده در (احسان و فیلی، ۲۰۱۳) و (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) به‌عنوان روش مبنا در نظر گرفته شده است و خروجی آن‌ها را با ویژگی‌های حساس به متن (PMI_{sen} ، PMI_{doc} و LM) رتبه‌بندی مجدد شده‌اند. این دو روش دیدگاه‌های متفاوتی به خطایابی حساس به متن دارند. در یکی از این روش‌ها از مترجم ماشینی استفاده شده است که جملهٔ خطادار را به جملهٔ صحیح ترجمه می‌کند. در روش دیگر از مدل زبانی برای انتخاب نامزد درست استفاده شده است.

یکی از روش‌های مبنا، روش ارائه‌شده در (احسان و فیلی، ۲۰۱۳) است که از SMT برای خطایابی حساس به متن استفاده کرده است. در این روش، با استفاده از مدلی که از دادهٔ آموزشی به‌دست آمده است، جمله‌های خطادار را به جمله‌های صحیح ترجمه می‌کند.

برای استفاده از این روش، مجموعه تداخل[1] و دادهٔ آموزشی نیاز است. مجموعه تداخل لغت w_i ، مجموعه‌ای از لغت‌های $\{w_{i1}, w_{i2}, \dots, w_{in}\}$ است که هر یک از w_{ij} می‌تواند به اشتباه به w_i تبدیل شود. دادهٔ آموزشی SMT بر اساس مجموعه تداخل ایجاد می‌شود؛ بنابراین لغت‌هایی که خارج از مجموعه باشند، توسط SMT قابل شناسایی نیستند. اگر در جملهٔ $E = \{w_1, w_2, \dots, w_n\}$، لغت w_i یک خطای حساس به متن باشد و جملهٔ $C = \{w_1, w_2, \dots, w'_i, \dots, w_n\}$ حالت درست جمله E باشد، آنگاه w'_i می‌تواند هر یک از لغت‌هایی باشد که w_i را در مجموعهٔ خود دارند. بنابراین بر اساس جواب‌های ممکن می‌توان نامزدهایی برای هر خطا تولید و بر اساس رابطهٔ بیزی (۵)، بهترین نامزد را انتخاب کرد.

$$\widehat{C}=\operatorname{argmax}_C \frac{P(E|C)P(C)}{P(E)} \quad (۵)$$

در رابطهٔ (۵)، $P(C)$ احتمال وقوع جملهٔ نامزد است که بر اساس LM محاسبه می‌شود. $P(E)$ احتمال وقوع جملهٔ خطاست که برای تمام نامزدهای تولیدشده یکسان است بنابراین می‌توان آن را از رابطه حذف کرد. برای تخمین $P(E|C)$ ابتدا E و C را از سطح کلمه‌ای، رابطهٔ (۶)، به سطح عبارتی $E = \overline{e_1}, \overline{e_2}, \dots, \overline{e_I}$ و $C = \overline{c_1}, \overline{c_2}, \dots, \overline{c_I}$ تبدیل می‌کنیم؛ سپس با استفاده از رابطهٔ (۷)، $P(E|C)$ را تخمین می‌زنیم.

$$P(E|C) = P(w_1, \dots w_i, \dots w_n | w_1, \dots w'_i, \dots w_n) \quad (۶)$$

$$P(E|C) \cong P\left(\overline{e_j} \mid \overline{c_j}\right) = \frac{count(\overline{e_j}, \overline{c_j})}{\sum_{\overline{e_j}} count(\overline{e_j}, \overline{c_j})} \quad (۷)$$

رابطهٔ (۷) درواقع همان مدل ترجمه عبارت در SMT است. در (احسان و فیلی، ۲۰۱۳) از Moses (کوهن و همکاران، ۲۰۰۷) به‌عنوان خطایاب حساس به متن استفاده شده است.

روش دیگری که به‌عنوان روش مبنا در نظر گرفته شده است روش MDM (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) است. این روش برای اولین بار در (مِی و همکاران، ۱۹۹۱) ارائه شد و در (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) مورد تحلیل و بررسی بیشتر قرار گرفت. در این روش برای هر جملهٔ ورودی مانند E، تمام حالت‌های ممکن بر اساس مجموعه تداخل تولید می‌شوند و فرض شده است که در هر جمله حداکثر یک خطای حساس به متن وجود دارد. در آخر بر اساس رابطهٔ (۸) محتمل‌ترین نامزد انتخاب می‌شود.

$$P(C|E) = \frac{P(C)P(E|C)}{P(E)} \quad (۸)$$

در رابطهٔ (۸)، $P(C)$ احتمال رخداد جمله است که با مدل زبانی و بر اساس ۳-گرام‌های موجود در پنجرهٔ محتوا تخمین زده می‌شود. در این مقاله، طول پنجرهٔ محتوا پنج کلمه در نظر گرفته شده است. به‌منظور محاسبهٔ $P(C|E)$، از آنجا که فرض شده است جملهٔ E و C فقط در یک کلمه فرق دارند، احتمال شرطی بین دو جمله را به‌احتمال شرطی بین دو لغت، $P(x|w)$، کاهش می‌دهیم. $P(x|w)$ برابر است با احتمال اینکه لغت نامزد x، حالت درست w باشد.

[1] Confusion set

برای محاسبهٔ $P(x|w)$ از رابطهٔ (۹) استفاده می‌کنیم:

$$P(x|w) \begin{cases} \alpha & \text{if } x=w \\ \frac{(1-\alpha)}{|SV(w)|} & \text{if } x \in SV(w) \\ 0 & \text{otherwise} \end{cases} \quad (۹)$$

در رابطهٔ (۹)، $SV(w)$ برابر است با مجموعه لغت‌هایی که لغت w می‌تواند به آن‌ها تبدیل شود و α برابر است با احتمال اینکه یک لغت توسط نویسنده درست نوشته شود.

(جدول- ۲): اطلاعات پیکره‌های استفاده شده.

پیکره	تعداد سند	تعداد لغت‌های یکتا	تعداد کلی لغت
همشهری	۱۶۶،۷۷۴	۵۷۶،۱۳۷	۷۸،۸۴۱،۰۴۵
ایرنا	۱۷۹،۵۷۴	۳۳۲،۳۴۳	۶۴،۰۸۵،۱۸۱
بیجن‌خان	۸۱۴	۵۶،۲۴۱	۲،۵۳۰،۷۷۲
بیجن‌خان + ایرنا + همشهری	۳۴۷،۱۶۲	۹۲۳،۷۴۴	۱۴۵،۴۵۶،۹۹۸

درواقع این پارامتر، میزان اعتماد سامانه به نویسنده را نشان می‌دهد. در این مقاله مقدار ۰/۹ را برای α در نظر گرفته شده است.

ما در این مقاله خروجی این دو روش را با استفاده از ویژگی‌های آگاه به متن PMI_{sen}، PMI_{doc} و LM، رتبه‌بندی مجدد کردیم و نتیجهٔ آنها را بهبود دادیم. در بخش بعدی پیکره‌های استفاده‌شده و نحوهٔ ساختن مجموعه تداخل، دادهٔ آزمون و آموزش توضیح داده خواهد شد.

۴- داده‌های مورد استفاده

در این بخش، پیکره‌ها و روش‌های ساختن داده‌های آموزش، دادهٔ آزمون و مجموعه تداخل توضیح داده شده است. از پیکره‌های موجود فارسی برای ساختن دادهٔ آموزشی و مجموعهٔ تداخل برای روش‌های مبنا استفاده شده است. این پیکره‌ها عبارت‌اند از همشهری[1]، ایرنا[2] و بیجن‌خان (بی‌جن‌خان، ۲۰۰۴). در جدول (۲) اطلاعاتی در مورد پیکره‌ها آورده شده است.

۴-۱- مجموعه تداخل

برای هر دو روش مبنا، یک مجموعه تداخل ساخته شده است. این مجموعه تداخل بر اساس معیار فاصلهٔ Damerau-Levenshtein (دِمراو، ۱۹۶۴) ساخته‌شده است. بنا به خطاهای املایی جمع‌آوری‌شده در (کاشفی و همکاران، ۲۰۱۳)، ۹۲٪ خطاها با یک تغییر نسبت به لغت درست ایجاد شده است. بنابراین مجموعه تداخل را بر اساس یک تغییر ایجاد نموده‌ایم. این تغییرات که باعث به‌وجودآمدن لغت جدید می‌شود، به چهار دسته تقسیم می‌شوند:

- اضافه‌شدن یک حرف[3]: مانند تبدیل لغت "کسب" به "کاسب"،
- حذف‌شدن یک حرف[4]: مانند تبدیل لغت "خروس" به "خرس"
- تعویض یک حرف[5]: مانند تبدیل "کباب" به "کتاب"
- جابه‌جایی دو حرف کنار هم[6]: مانند تبدیل "کاتب" به "کتاب".

چنانچه فاصلهٔ Damerau-Levenshtein دو لغت w_i و w_j برابر یک باشد یا به‌عبارت‌دیگر، فقط با یکی از تغییرات گفته‌شده به یکدیگر تبدیل شده باشند، به مجموعه تداخل یکدیگر اضافه می‌شوند. برای ساخت دادهٔ آموزش برای مترجم ماشینی آماری، یک مجموعه تداخل با پنج هزار مجموعه ساخته شده است.

۴-۲- دادهٔ آزمون

برای مقایسهٔ رتبه‌بندی آگاه به متن با روش‌های مبنا از دو دادهٔ آزمون مختلف استفاده شده است. دادهٔ آزمون اول یک

[1] http://ece.ut.ac.ir/DBRG/Hamshahri
[2] http://www.irna.ir.

[3] Insertion
[4] Deletion
[5] Substitution
[6] Transposition

دادهٔ آزمون مصنوعی است. این مجموعهٔ خطا، شامل هزار و پانصد جمله است که به‌طور تصادفی از پیکرهٔ بی‌جن‌خان استخراج‌شده و فرض شده است که هر جمله فقط یک خطای حساس به متن دارد. طول این جمله‌ها بین چهار و بیست کلمه است. برای هر جمله در این مجموعه خطا، به‌طور تصادفی یک کلمه از جمله را که در مجموعه، تداخل باشد، انتخاب کردیم و به‌طور تصادفی با یکی از لغت‌های درون آن مجموعه جابه‌جا کردیم. به عبارت دیگر برای هر جمله یک خطای حساس به متن به‌طور مصنوعی به جمله‌ها اضافه کردیم. این روش ساختن دادهٔ آزمون در (ویلکاگس-اُهرن و همکاران، ۲۰۰۸)، (هیرتس و بُدانیستکی، ۲۰۰۵) و (فُساتی و دی‌یوجینو، ۲۰۰۷) نیز استفاده شده است. از آنجا که این داده به‌طور تصادفی ساخته شده است، این آزمایش سه مرتبه انجام‌شده و نتایج به‌صورت میانگین در بخش بعدی آورده شده است.

دادهٔ آزمون بعدی، یک دادهٔ واقعی[1] است؛ خطاهای این داده توسط انسان در دنیای واقعی ایجاد شده است. این مجموعه شامل هزار و صد خطای حساس به متن است که از وبلاگ‌های فارسی جمع‌آوری شده است (میرزابابایی و فیلی، ۲۰۱۳). میانگین طول جمله‌ها ۱۶/۷ کلمه است. جزئیات بیشتر این مجموعه در جدول (۳) آورده شده است. این مجموعه داده به‌صورت رایگان در دسترس است[2].

(جدول- ۳): جزئیات دادهٔ آزمون واقعی.

تعداد	دادهٔ آزمون واقعی
۱،۱۰۰	کل خطا
۲۷	خطای اضافه
۳۶۶	خطای حذف
۵۲۷	خطای جانشینی
۹۱	خطای جابه‌جایی
۸۹	خطاهای بیشتر از یک تغییر

بنا به خطاهای املایی جمع‌آوری شده در (کاشفی و همکاران، ۲۰۱۳)، ۹۲٪ خطاها با یک تغییر نسبت به لغت درست ایجاد شده است. این آمار در خطاهای موجود در دادهٔ آزمون واقعی هم دیده می‌شود. همان‌طور که در جدول (۳) مشاهده می‌کنید، فقط ۸۹ خطا که حدود ۸٪ از کل خطای دادهٔ آزمون است، نیاز به دو تغییر و ۹۲٪ مابقی خطاها نیاز به یک تغییر دارند.

[1] Real-world
[2] http://ece.ut.ac.ir/nlp/resources/

۴-۳- دادهٔ آموزشی

دادهٔ آموزشی مورد نیاز، دادهٔ آموزشی برای روش‌های مبنای SMT، MDM و همچنین دادهٔ آموزشی برای SVM است. دادهٔ آموزشی SMT بر اساس مجموعه تداخل و از روی پیکره‌های موجود فارسی ساخته شده است. این دادهٔ آموزشی به‌صورت یک پیکرهٔ موازی است که در یک طرف جمله‌های صحیح و در طرف دیگر جمله‌های غلط وجود دارد.

روند ساختن این دادهٔ آموزشی به این صورت است که، به‌ازای هر جملهٔ پیکره‌های همشهری، ایرنا و بی‌جن‌خان، اگر جملهٔ مورد نظر دارای کلمه‌ای بود که در مجموعه تداخل وجود داشت، آن جمله را یک مرتبه به‌صورت صحیح در پیکرهٔ موازی وارد می‌کنیم؛ سپس به‌ازای همهٔ کلمه‌های درون مجموعه تداخل آن لغت، جمله را به‌صورت صحیح و غلط به پیکرهٔ موازی اضافه می‌کنیم. به‌عنوان مثال، اگر مجموعه تداخل لغت "روز"، لغت‌های "روزه"، "روش"، "رود" و "روح" باشد، با دیدن لغت "روز" در جملهٔ "امروز روز سردی است."، جمله‌ها به‌صورتی‌که در جدول (۴) نشان داده شده است به پیکرهٔ موازی اضافه می‌شود. این روش ساختن دادهٔ آموزشی مشابه روش ارائه شده در (احسان و فیلی، ۲۰۱۳) است.

(جدول- ۴): قسمتی از پیکرهٔ موازی از دادهٔ آموزشی برای مترجم ماشینی آماری.

طرف خطادار	طرف صحیح
امروز *روز* سردی است.	امروز *روز* سردی است.
امروز *روزه* سردی است.	امروز *روز* سردی است.
امروز *روش* سردی است.	امروز *روز* سردی است.
امروز *رود* سردی است.	امروز *روز* سردی است.
امروز *روح* سردی است.	امروز *روز* سردی است.

پیکرهٔ موازی ساخته‌شده دارای ۳۸۱،۰۰۷ جفت جمله است و به‌عنوان دادهٔ آموزشی برای SMT استفاده شده است. در این مقاله همانند کار انجام‌شده در (احسان و فیلی، ۲۰۱۳) از مترجم ماشینی Moses استفاده‌شده است و همچنین از ابزار GIZA++ (اُچ و نِی، ۲۰۰۳) برای تطابق کلمه‌ها و از ابزار SRILM (استولک و همکاران، ۲۰۰۲) برای ساختن مدل زبانی استفاده شده است.

به‌منظور ساخت دادهٔ آموزشی برای SVM-rank[1]، ابتدا یک مجموعه تداخل شامل ۲۶،۸۹۱ لغت مختلف از روی همشهری و بی‌جن‌خان ساخته شده است. هر لغت به‌طور میانگین ۴/۶ کلمهٔ نامزد دارد؛ سپس پنج هزار جملهٔ صحیح به‌طور تصادفی از همشهری و بی‌جن‌خان انتخاب شده و خطای حساس به متن به آنها اضافه شده است؛ سپس به‌وسیلهٔ روش SMT برای هر جملهٔ خطادار حداکثر بیست جملهٔ نامزد تولید و مقدار PMI_{doc} ، $PM\ I_{sen}$ و LM برای هر نامزد محاسبه شده است. ۵۶،۳۲۰ جملهٔ صحیح و خطا به همراه مقدارهای PMI_{doc} ، $PM\ I_{sen}$ و LM را به‌عنوان دادهٔ آموزشی برای SVM-rank در نظر گرفتیم.

پس از محاسبهٔ وزن هر ویژگی، مقدار ویژگی‌های هر جملهٔ فهرست خروجی روش‌های مبنا محاسبه و به‌وسیلهٔ یک روش لگاریتم خطی رتبه‌بندی مجدد می‌شوند. در بخش بعد، نتایج روش‌های مبنا با نتایج رتبه‌بندی مجدد نشان داده می‌شود.

۵- نتایج آزمایش‌ها

به‌منظور ارزیابی روش، از معیارهای دقت[2] در تشخیص و تصحیح، بازخوانی[3] در تشخیص و تصحیح، معیار F_1[4] در تشخیص و تصحیح و MRR[5] استفاده شده است. هر یک از این معیارها به غیر از MRR توسط منفی و مثبت حقیقی و منفی و مثبت کاذب محاسبه شده است.

- مثبت کاذب[6](FP): لغت‌های خطایی هستند که تشخیص داده نشده‌اند.
- منفی کاذب[7](FN): لغت‌های درستی هستند که خطا تشخیص داده شده‌اند.
- مثبت حقیقی[8](TP): لغت‌های درستی هستند که درست تشخیص داده شده‌اند.
- منفی حقیقی[9](TN): لغت‌های خطایی که به‌درستی تشخیص داده شده‌اند؛ ولی به‌درستی تصحیح نشده‌اند.
- منفی حقیقی با تصحیح[10](TNC): لغت‌های خطایی که به‌درستی تشخیص و تصحیح شده‌اند.

بنا به تعریف‌های گفته‌شده، معیارهای ارزیابی به‌صورت روابط (۱۰-۱۶) محاسبه می‌شوند.

$$\text{Detection Precision}=\frac{\#\text{ of TN}}{\#\text{ of TN + FN}} \quad (۱۰)$$

$$\text{Detection Recall}=\frac{\#\text{ of TN}}{\#\text{ of FP + TN}} \quad (۱۱)$$

$$\text{Detection } F_1=\frac{2*DP*DR}{DP+DR} \quad (۱۲)$$

$$\text{Correction Precision}=\frac{\#\text{ of TNC}}{\#\text{ of TN}} \quad (۱۳)$$

$$\text{Correction Recall}=\frac{\#\text{ of TNC}}{\#\text{ of FP + TN}} \quad (۱۴)$$

$$Correction\ F_1=\frac{2*CP*CR}{CP+CR} \quad (۱۵)$$

$$\text{MRR}=\frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\text{rank}_i} \quad (۱۶)$$

در رابطهٔ (۱۶)، $|Q|$ برابر است با تعداد جمله‌های دادهٔ آزمون و $rank_i$ برابر با رتبهٔ نامزد صحیح در فهرست است. از آنجا که دادهٔ آزمون مصنوعی به‌صورت تصادفی تهیه شده، ارزیابی روی آن سه‌مرتبه انجام شده است. بیشترین بازهٔ اطمینان[11] با احتمال ۹۵٪ در بازهٔ ۰/۰۱۷۳۹± قرار می‌گیرد. همچنین آزمون معناداری[12] T هم روی خروجی‌ها انجام شده است.

خروجی روش مبنای SMT (احسان و فیلی، ۲۰۱۳) را RSMT[13] و خروجی روش مبنای MDM (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) را RMDM[14] می‌نامیم. در جدول (۵)، میزان بهبود روش‌های مبنا، توسط رتبه‌بندی مجدد آگاه به متن نشان داده شده است.

همان‌طور که در جدول (۵) مشاهده می‌کنید، رتبه‌بندی مجدد باعث بهبود قابل ملاحظه‌ای در روش‌های مبنا شده است. این رتبه‌بندی با از دست‌دادن مقدار بسیار کمی از دقت، باعث افزایش چشم‌گیری در بازخوانی تشخیص و تصحیح شده است. چنانچه معیار F_1 را برای ارزیابی در نظر بگیریم، در هر چهار آزمایش بهبود قابل توجهی داشته‌ایم. MRR معیار خوبی برای ارزیابی

[1] http://svmlight.joachims.org
[2] Precision
[3] Recall
[4] F-measure
[5] Mean Reciprocal rank
[6] False positive
[7] False negative
[8] True positive
[9] True negative
[10] True negative with correction
[11] Confidence interval
[12] Significance test
[13] Reranked SMT
[14] Reranked MDM

فهرست‌های خروجی روش مبنا است، همان‌طور که در جدول (۵) مشاهده می‌کنید در ارزیابی روی دادهٔ مصنوعی، مقدار MRR در روش (احسان و فیلی، ۲۰۱۳) و (ویلکاگس-اُهرن و همکاران، ۲۰۰۸) به‌ترتیب ۴٪ و ۲/۸٪ و در ارزیابی روی دادهٔ واقعی، به‌ترتیب ۴٪ و ۵/۳٪ بهبود داشته است.

(جدول- ۵): نتایج ارزیابی رتبه‌بندی آگاه به متن بر روی روش‌های مبنا (میزان بهبود نسبت به روش مبنا در پرانتز آورده شده است.).

نتایج روی دادهٔ آزمون مصنوعی	RSMT		RMDM	
	تشخیص	تصحیح	تشخیص	تصحیح
دقت	۰/۸۵(-٪۰/۱)	۰/۸۳(-٪۰/۱)	۰/۷۵(-٪۲/۲)	۰/۸۵(-٪۱/۳)
بازخوانی	۰/۷۶(+٪۱۷/۸)	۰/۶۴(+٪۱۷/۹)	۰/۶۹(+٪۶/۲)	۰/۵۹(+٪۴/۸)
F_1	۰/۸۰(+٪۹/۱)	۰/۷۲(+٪۱۰/۱)	۰/۷۲(+٪۲/۱)	۰/۷۰(+٪۲/۹)
MRR	۰/۶۷(+٪۴)		۰/۷۰(+٪۲/۸)	
نتایج روی دادهٔ آزمون واقعی	**RSMT**		**RMDM**	
	تشخیص	**تصحیح**	**تشخیص**	**تصحیح**
دقت	۰/۸۹(-٪۰/۲)	۰/۹۷(-٪۱)	۰/۹۰(-٪۲/۱)	۰/۹۷(-٪۱/۶)
بازخوانی	۰/۷۰(+٪۹/۵)	۰/۶۹(+٪۸/۴)	۰/۷۲(+٪۱۲/۵)	۰/۷۰(+٪۱۰/۶)
F_1	۰/۷۹(+٪۴)	۰/۸۰(+٪۸/۴)	۰/۸۰(+٪۵/۹)	۰/۸۱(+٪۵/۵)
MRR	۰/۷۱(+٪۴)		۰/۷۲(+٪۵/۳)	

درکل بر اساس معیارهای تعریف‌شده به جز معیار دقت، رتبه‌بندی مجدد روش‌های مبنا، باعث بهبود خروجی آن‌ها شده است. کاهش مقدار دقت در روش MDM می‌تواند به‌دلیل افزایش تعداد نامزدها و افزایش طول جمله‌های ورودی باشد. در این روش متوسط تعداد نامزدهای تولیدشده برای هر جملهٔ دادهٔ آزمون ۴۲ جمله است. طبق رابطهٔ (۲) و (۳) که برای محاسبهٔ $PM\ I_{sen}$ ، PMI_{doc} تعریف‌شده است، با افزایش طول جمله تأثیر این دو معیار کمتر می‌شود.

۶- نتیجه‌گیری و ادامه کار

ما در این مقاله یک رتبه‌بند آگاه به متن برای خطایاب‌های حساس به متن ارائه شده است. ما روش‌هایی بر مبنای SMT و MDM را به‌عنوان روش‌های مبنا در نظر گرفتیم و خروجی‌های این دو روش را با یک مدل لگاریتم خطی، رتبه‌بندی مجدد کردیم. برای رتبه‌بندی از سه ویژگی آگاه به متن $PM\ I_{sen}$، PMI_{doc} و LM استفاده شده است و به‌منظور وزن‌دهی این ویژگی‌ها، از الگوریتم یادگیری SVM استفاده شده است با رتبه‌بندی مجدد خروجی روش SMT بر روی داده آزمون واقعی، به بهبود ۸/۴٪ و ۹/۵٪ در بازخوانی تصحیح و تشخیص دست یافته‌ایم. با توجه به کاهش اندک معیار دقت، بهبود قابل توجهی در حدود ۸/۴٪ و ۴٪ در معیار F_1 در تصحیح و تشخیص داشته‌ایم. این بهبود چشم‌گیر در معیارهای بازخوانی، F_1 و MRR و کاهش اندک معیار دقت در دیگر آزمایش‌ها نیز دیده می‌شود.

به‌منظور بهبود این روش، برای حالتی که خطای حساس به متن باعث تغییر نقش کلمه در جمله می‌شود، می‌توان از ویژگی‌های دیگری مثل برچسب اجزای کلام، نقش کلمه‌ها در جمله می‌شود استفاده کرد. همچنین می‌توان از فارس‌نت (شمس‌فرد، ۲۰۰۸) به‌عنوان یک ویژگی آگاه به محتوا استفاده کرد. در آخر، از آنجا که این روش یک روش مستقل از زبان است می‌توان روی زبان‌های پرکاربرد دیگر استفاده کرد. همچنین می‌توان این روش را به‌عنوان یک پس‌پردازش روی خطایاب‌های دیگر اعمال کرد؛ به شرطی که متن اطراف جملهٔ ورودی به خطایاب موجود باشد.

۷- مراجع

Bassil, Y., and Alwani, M. (2012). Context-sensitive Spelling Correction Using Google Web 1T 5-Gram Information. arXiv preprint arXiv:1204.5852.

Bijankhan, M. (2004). The role of the corpus in writing a grammar: An introduction to a software. Iranian Journal of Linguistics, 19(2).

Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. Communications of the ACM, 7(3), 171-176.

Ehsan, N. and Faili, H. (2013). Grammatical and context-sensitive error correction using a statistical machine translation framework. Software: Practice and Experience, 43(2), 187-206.

Gale, W. A., Church, K. W., and Yarowsky, D. (1992). A method for disambiguating word senses in a large corpus. Computers and the Humanities, 26(5-6), 415-439.

Golding, A. R., and Schabes, Y. (1996, June). Combining trigram-based and feature-based methods for context-sensitive spelling correction. In Proceedings of the 34th annual meeting on Association for Computational Linguistics (pp. 71-78). Association for Computational Linguistics.

Hirst, G., and Budanitsky, A. (2005). Correcting real-word spelling errors by restoring lexical cohesion. Natural Language Engineering, 11(01), 87-111.

Jeng, J. T. (2005). Hybrid approach of selecting hyperparameters of support vector machine for regression. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 36(3), 699-709.

Kashefi, O., Sharifi, M., and Minaie, B. (2013). A novel string distance metric for ranking Persian respelling suggestions. Natural Language Engineering, 19(2), 259-284.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., ... and Herbst, E. (2007, June). Moses: Open source toolkit for statistical machine translation. In Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions (pp. 177-180). Association for Computational Linguistics.

Kukich, K. (1992). Techniques for automatically correcting words in text. ACM Computing Surveys (CSUR), 24(4), 377-439.

Mays, E., Damerau, F. J., and Mercer, R. L. (1991). Context based spelling correction. Information Processing & Management, 27(5), 517-522.

Miangah, T. M. (2013). FarsiSpell: A spell-checking system for Persian using a large monolingual corpus. Literary and Linguistic Computing.

Mirzababaei, B., Faili, H. and Ehsan, N. (2013). Discourse-aware Statistical Machine Translation as a Context-Sensitive Spell Checker, In RANLP.

Mitton, R. (1987). Spelling checkers, spelling correctors and the misspellings of poor spellers. Information processing & management, 23(5), 495-505.

Och, F. J., and Ney, H. (2003). A systematic comparison of various statistical alignment models. Computational linguistics, 29(1), 19-51.

Shamsfard, Mehrnoush. (2008). Developing FarsNet: A lexical ontology for Persian. Paper presented at the 4th Global WordNet Conference, Szeged, Hungary.

Stolcke, A. (2002, September). SRILM-an extensible language modeling toolkit. In INTERSPEECH.

Tsai, C. F. (2005). Training support vector machines based on stacked generalization for image classification. Neurocomputing, 64, 497-503.

Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. (2005). Large margin methods for structured and interdependent output variables. In Journal of Machine Learning Research (pp. 1453-1484).

Wilcox-O'Hearn, A., Hirst, G., and Budanitsky, A. (2008). Real-word spelling correction with trigrams: A reconsideration of the Mays, Damerau, and Mercer model. In Computational Linguistics and Intelligent Text Processing (pp. 605-616). Springer Berlin Heidelberg.

Yarowsky, D. (1994, June). Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French. In Proceedings of the 32nd annual meeting on Association for Computational Linguistics (pp. 88-95). Association for Computational Linguistics.

بهـزاد میرزابابـایی کارشناسـی ارشد خود را در رشـته مهندسـی نرم‌افزار از دانشگاه تهران در سـال ۱۳۹۳ به پایان رساند و همچنـین مقطـع کارشناشـی را از دانشـگاه ارومیـه در سـال ۱۳۹۰ در رشـته مهندسی نرم‌افزار اخذ نموده اسـت. زمینـهٔ پژوهشـی ایشـان پردازش زبان طبیعی و به‌طور خاص خطایابی املایی، نحـوی و معنایی است.

نشانی رایانامهٔ ایشان عبارت است از:

b.mirzababaei@ut.ac.ir

هشام فیلی تحصیلات خود را در مقطع کارشناسی مهندسی نرم‌افزار در دانشکدهٔ مهندسی کامپیوتر دانشگاه صنعتی شریف با رتبهٔ یک در سال ۱۳۷۶ به پایان رساند؛ سپس مقاطع کارشناسی ارشد نرم‌افزار و دکترای هوش مصنوعی را به‌ترتیب در سال‌های ۱۳۷۸ و ۱۳۸۵ در همان دانشکده تکمیل کرد. از سال ۱۳۸۷ تاکنون عضو هیأت علمی دانشکدهٔ مهندسی برق و کامپیوتر دانشکدهٔ فنی دانشگاه تهران است. زمینه‌های پژوهشی مورد علاقهٔ ایشان پردازش هوشمند متن و زبان طبیعی، ترجمهٔ ماشینی، داده‌کاوی، بازیابی اطلاعات و شبکه‌های اجتماعی هستند.

نشانی رایانامهٔ ایشان عبارت است از:

hfaili@ut.ac.ir