تشخیص هوشمند و خودکار غلط‌های تایپی در پایگاه‌داده‌های بزرگ بدون استفاده از لغت‌نامه

الناز زعفرانی معطر[۱]، مربی؛ محمدرضا فیضی درخشی[۲]، استادیار؛ آزاده روحانی[۳]، مربی

۱- گروه مهندسی کامپیوتر - واحد تبریز - دانشگاه آزاد اسلامی - تبریز - ایران - e.zafarani@iaut.ac.ir
۲- گروه مهندسی کامپیوتر - دانشکده مهندسی برق و کامپیوتر - دانشگاه تبریز - تبریز - ایران - mfeizi@tabrizu.ac.ir
۳- گروه مهندسی کامپیوتر - واحد خسروشاه - دانشگاه آزاد اسلامی - خسروشاه - ایران - roohany@iaukhosh.ac.ir

چکیده: غلط‌های تایپی یکی از مشکلات مهم در سیستم‌های کامپیوتری و سیستم‌های پایگاه‌داده‌ای است. وجود غلط‌های تایپی در پایگاه‌داده‌ها نه تنها از نظر صحت پایگاه‌داده مشکل‌ساز هستند، بلکه باعث می‌شوند به هنگام ضرورت نتوان رکورد واردشده را بازیابی کرد. همین امر گاه باعث می‌شود که کاربر مجدداً همان رکورد را وارد پایگاه‌داده نماید که باعث به وجود آمدن افزونگی می‌گردد. روش‌های موجود تشخیص غلط‌ها، مبتنی بر لغت‌نامه هستند. بدین معنی که از یک لغت‌نامه بزرگ که همه لغات آن صحیح فرض می‌شوند استفاده می‌کنند و اگر کلمه‌ای در لغت‌نامه نباشد به‌عنوان غلط تایپی شناخته می‌شود. تهیه لغت‌نامه‌ای بزرگ و با دقت بالا بسیار پرهزینه و زمان‌بر است. به‌علاوه چنین لغت‌نامه‌ای مختص یک زمینه خاص (مثلاً محیط پزشکی) است و قابل‌استفاده و در زمینه‌های دیگر (مثلاً جامعه‌شناسی) نیست. در این مقاله روشی ارائه می‌شود که بدون نیاز به لغت‌نامه می‌تواند غلط‌های تایپی را تشخیص دهد. روش پیشنهادی با چند معیار مرسوم ارزیابی شده است. نتایج آزمایش‌ها نشان‌دهنده دقت ۹۳/۵ درصدی برای این روش است. علاوه‌بر دقت بالای روش پیشنهادی، عدم‌نیاز به لغت‌نامه یک ویژگی منحصر به فرد برای آن به‌شمار می‌رود.

واژه‌های کلیدی: تشخیص غلط‌های تایپی، غلط‌های املایی، فازی.

The Intelligent and Automatic Detection of Type Errors in Large Databases without using Dictionary

E. Zafarani-Moattar[1], Instructor; M. R. Feizi-Derakhshi[2], Assistant Professor; A. Roohany[3], Instructor

1- Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran, Email: e.zafarani@iaut.ac.ir
2- Department of Computer Engineering, University of Tabriz, Tabriz, Iran, Email: mfeizi@tabrizu.ac.ir
3- Department of Computer Engineering, Khosroshah Branch, Islamic Azad University, Khosroshah, Iran, Email: roohany@iaukhosh.ac.ir

Abstract: Type errors are one of the main problems in computer systems and database systems. Existence of type errors within databases, not only causes accuracy problem for database, but also leads user to re-enter the record into database because the entered record could not be found. It results in redundancy. The existing error detection methods are based on dictionary. It means that they use a large dictionary whose all words are assumed true and if a word is not in the dictionary, it is detected as a type error. Providing a large dictionary with high precision is expensive and time consuming. In addition, such a dictionary belongs to a special field (for example, medical environment) and is not applicable in other fields (such as sociology). In this paper, a method is presented that could detect type errors without requiring a dictionary. The proposed method has been evaluated with some common criteria. The experimental results show 93.5 percent precision for this method. In addition to the high precision of the proposed method, not requiring a dictionary is considered as its unique feature.

Keywords: Detection of type errors, spelling errors, fuzzy.

تاریخ ارسال مقاله: ۱۳۹۴/۰۷/۱۹
تاریخ اصلاح مقاله: ۱۳۹۴/۱۰/۲۲ و ۱۳۹۵/۰۲/۱۳
تاریخ پذیرش مقاله: ۱۳۹۵/۰۴/۰۵
نام نویسنده مسئول: الناز زعفرانی معطر
نشانی نویسنده مسئول: ایران - تبریز - اتوبان پاسداران - دانشگاه آزاد اسلامی واحد تبریز - دانشکده فنی - گروه مهندسی کامپیوتر.

۱- مقدمه

با گسترش سریع علم و دانش و لزوم دسترسی سریع به اطلاعات نیاز به ذخیره صحیح آن‌ها در پایگاه‌داده‌ها است تا به‌راحتی بازیابی شود. اشکالات تایپی در ورود داده‌ها از مواردی است که باعث نادرستی داده‌های داخل پایگاه‌داده‌ها می‌شود و مسلماً بازیابی اطلاعات با مشکلاتی مواجه می‌شود از جمله پیدا نکردن رکورد موردنظر و نیاز به جستجوهای پیشرفته که باعث اتلاف زمان می‌شود و اگر رکورد موردنظر پیدا نشد دوباره وارد پایگاه‌داده می‌شود که باعث به وجود آمدن رکوردهای تکراری می‌شود. رکوردهای تکراری باعث افزونگی در پایگاه‌داده‌ها می‌شود [۱].

رایج‌ترین روش برای تشخیص غلط‌های تایپی به این صورت است که کلمه در یک لغت‌نامه جستجو می‌شود اگر این کلمه در آن لغت‌نامه یافت شد، صحیح و در غیر این صورت آن کلمه را دارای غلط املایی فرض می‌کند. در ضمن این فرض هم در نظر گرفته می‌شود که تمام کلمات لغت‌نامه صحیح و بدون اشکال می‌باشند. البته لزوماً این فرض همواره صحیح نیست. بدیهی است که هرقدر تعداد کلمه‌های لغت‌نامه زیادتر بوده و با دقت بیش‌تری جمع شده باشد، پیدا کردن کلمه‌های غلط با اعتبار بیش‌تری انجام می‌گیرد. اما برای ساختن لغت‌نامه با چنین حجم و دقت بالایی زمان و هزینه بالایی نیاز است. به‌علاوه برای هر حوزه علمی، لغت‌نامه مختص آن حوزه لازم است. مثلاً لغت‌نامه حوزه آب شهری در زمینه تأسیسات برق قابل‌استفاده نیست. از این‌رو باید نیروی انسانی صاحب علمی در آن حوزه خاص، وجود داشته باشد تا لغت‌نامه‌ای مناسب ایجاد گردد. همچنین با وجود لهجه‌ها و گویش‌های مختلف که در سراسر کشور وجود دارد اصطلاح‌های گوناگونی هم وجود دارد که باید درست تشخیص داده شود. اگر در چندین زمینه و علوم مختلف داده‌ها وارد پایگاه‌داده شود نیاز به لغت‌نامه‌های مخصوص به هر یک از آن‌ها می‌باشد که وجود این لغت‌نامه‌ها حجم زیادی را اشغال خواهد نمود.

مشکل اصلی روش لغت‌نامه اختصاصی بودن آن است. بدین معنی که معمولاً لغت‌نامه مختص یک کاربرد خاص است و با تغییر محیط کاربری برنامه، می‌بایست لغت‌نامه دیگری برای محیط جدید تهیه شود. مثلاً لغت‌نامه ثبت املاک قابل‌استفاده در محیط پزشکی نیست. با توجه به حجم بالای لغت‌نامه و دقت زیاد موردنیاز، تهیه آن بسیار زمان‌بر و پرهزینه خواهد بود که باعث می‌شود بسیاری از برنامه‌های کاربردی رایج نتوانند از این روش استفاده کنند و درنتیجه اکثر برنامه‌های کاربردی رایج در بازار، عمل تشخیص غلط‌های تایپی را انجام نمی‌دهند.

در این مقاله ما به‌دنبال روشی هستیم که بدون استفاده از لغت‌نامه و بدون نیاز به صرف هزینه برای تولید آن بتوانیم غلط‌های تایپی را تشخیص دهیم. ایده اصلی برای این کار، استفاده از اطلاعات موجود در پایگاه‌داده‌های بزرگ است که امروزه معمولاً در تمامی سازمان‌ها وجود دارند. بدین معنی که اگر پایگاه‌داده به‌اندازه کافی بزرگ باشد، کلمات پرتکرار موجود در پایگاه‌داده می‌تواند نشان‌دهنده کلمات تخصصی حوزه کاربرد آن پایگاه‌داده باشد. مزیت این روش آن است که بدون نیاز به لغت‌نامه تخصصی می‌توان به لغات آن حوزه دسترسی داشت. همچنین با توجه به اینکه پایگاه‌داده همواره به‌روز می‌شود، درصورتی‌که احیاناً لغت جدیدی به حوزه کاربرد اضافه شود (با توجه به خاصیت زایش زبان)، این روش به‌خوبی آن را پوشش می‌دهد. البته بدیهی است که پایگاه‌داده را نمی‌توان عاری از خطا دانست به همین دلیل در این مقاله از منطق فازی کمک گرفته شده است تا لغاتی که دارای تکرار کم هستند و احیاناً غلط می‌باشند از چرخه تأثیرگذاری خارج شوند. اصولاً ایده این مقاله، یک ایده heuristic و مبتنی بر تجربه انسانی است که در آزمایش‌ها نشان داده که بسیار خوب و کارا عمل می‌کند. برای پیاده‌سازی تجربه انسان در کامپیوتر نیز از منطق فازی استفاده شده است. به‌طورکلی می‌توان گفت که هرچه یک تصمیم‌گیری بیش‌تر درگیر نیروی انسانی و تجربه انسان باشد و پیچیدگی سیستم نیز بالا رود، پدیده فازی بیش‌تر مسلط بر توضیح این سیستم‌ها می‌گردد [۲].

۲- پیشینه تحقیق

غلط‌های تایپی یکی از مشکلات رایج در کامپیوتر و پایگاه‌داده است که دارای انواع مختلفی است. طبق تحقیق انجام‌شده در [۳] انواع غلط‌های تایپی عبارت‌اند از:

درج: در هنگام نگارش، حرف دیگری اضافه بر حروف کلمه تایپ می‌شود.

حذف: در هنگام نگارش، حرفی جاافتاده باشد.

جابجایی: در هنگام نگارش، به‌جای حرف موردنظر حرف دیگری تایپ شده باشد.

انتقال: در هنگام نگارش، حرف در مکان دیگری به غیر از جایگاه اصلی خود تایپ شده باشد.

ترکیبی: بقیه حالات اشتباه تایپی در این دسته قرار داده شده است.

جدول ۱ انواع غلط‌های تایپی را به‌همراه نسبت رخداد آن‌ها در زبان نشان می‌دهد.

جدول ۱: انواع غلط‌های تایپی و نسبت رخداد آن‌ها در زبان فارسی [۳]

ردیف	نوع غلط	گونه اصلی	درصد
۱	درج حرف	درج	۸
۲	تکرار حرف	درج	۱۴/۵
۳	حذف حرف	حذف	۱۹
۴	جابجایی کلید	جابجایی	۳۹/۵
۵	انتقال کلید	انتقال	۵
۶	فاصله افزوده	درج	۲/۵
۷	فاصله کم	حذف	۵
۸	موارد دیگر	ترکیبی	۶/۵

به‌علت اهمیت موضوع غلط‌های املایی و تایپی، کارهای تحقیقاتی زیادی در زمینه تشخیص و رفع آن انجام شده است که از جمله آن‌ها می‌توان به فاصله ویرایشی[1] اشاره کرد. این الگوریتم برای یافتن جایگزین مناسب برای کلمه‌ای که دارای غلط تایپی است مناسب می‌باشد. این الگوریتم با مقایسه دو رشته S و T کم‌ترین عملیاتی که نیاز است تا رشته اول به رشته دوم تبدیل شود را با استفاده از فرمول‌های (۱) و (۲) محاسبه می‌کند.

$$\mathrm{Sim}(S,T)=\frac{D(|S|,|T|)}{\mathrm{Max}(|S|,|T|)} \quad (۱)$$

$$D(i,j)=\min\begin{cases}D(i-1,j-1)+d(S_i,T_j) & \text{کپی و تعویض}\\ D(i-1,j)+1 & \text{واردکردن}\\ D(i,j-1)+1 & \text{حذف}\end{cases} \quad (۲)$$

که در آن S و T رشته‌های موردنظر است همچنین |S| و |T| طول رشته‌های موردنظر می‌باشد. $d(S_i,T_j)$ هزینه موردنیاز برای تعویض و یا کپی است [۴].

برای تشخیص غلط‌های تایپی تلاش‌های زیادی شده است در تحقیقات اولیه مانند [۵]، از روش چک‌کردن کلمه با کلمات داخل لغت‌نامه استفاده می‌شود. در تحقیقات آتی هم همه برای تشخیص کلمه غلط از یک لغت‌نامه استفاده می‌کنند که این لغت‌نامه‌ها بنا به زبان و زمینه موردنظر تغییر می‌کند و به این دلیل روی زبان‌های مختلف از جمله عربی، چینی، اردو، هندی و فارسی و علوم مختلف، تحقیقات مختلفی شده است.

غلط‌های تایپی همه‌گیر است و در زبان‌های هندی و چینی و ... نیز وجود دارد که برخی مقالات مانند [۱۰-۶]، بر روی غلط‌های تایپی و املایی زبان چینی کار کرده‌اند. در جدول ۲ خلاصه‌ای از کارهای انجام‌شده برای زبان چینی آورده شده است. توجه به ستون F_1 در این جدول نشان می‌دهد که نتایج به‌دست‌آمده، نتایج نسبتاً ضعیفی می‌باشند. این ضعف در مقایسه با زبان‌های دیگر (به‌عنوان‌مثال جدول ۶ و جدول ۸ را ببینید) بیش‌تر خودنمایی می‌کند. دلیل این امر را شاید بتوان در طرز نگارشی زبان چینی جدا نبودن نویسه‌های آن جستجو کرد. به‌علاوه می‌توان دید که اختلاف زبان می‌تواند اختلاف معنی‌داری در نتایج ایجاد کند. در [۱۱] بر روی تشخیص و اصلاح کلمات اشتباه در زبان هندی کار شده است. نویسنده در مقاله خود خاطرنشان کرده است که کارهای زیادی بر روی زبان انگلیسی انجام شده است ولی بر روی زبان هندی کار چندانی انجام نشده است و در زبان هندی هم به‌طور مستقیم از روش‌های موجود برای زبان انگلیسی و همچنین لغت‌نامه‌های آن‌ها نمی‌توان استفاده کرد و با اشاره به اینکه در زبان انگلیسی ۲۱ حرف صامت و ۵ حرف صدادار وجود دارد درحالی‌که در زبان هندی ۴۰ حرف صامت و ۱۰ حرف صدادار وجود دارد، سختی کار را برجسته‌تر کرده است.

جدول ۲: نمونه‌ای از نتایج کارهای انجام‌شده در زبان چینی

روش	استفاده از لغت‌نامه	خلاصه نتایج				توضیحات
		Pr	Re	F_1	Ac	
Shuiyuan Zhany-2015	بله	۰/۸	۰/۵۳	۰/۶۴	۰/۷	
		۰/۸۳	۰/۵۳	۰/۶۴	۰/۷	بهترین
Weijan-2015	بله	۰/۶۱۴	۰/۳۱	۰/۳۹	۰/۵۴	میانگین
Yuen-Hsien Tseng-2015	بله	۰/۵	۰/۶۷	۰/۵۷	۰/۶	
Shaohra yang-2012	بله	۰/۳۶	۰/۷۲	۰/۴۸	NR	
Patrick-2005	بله	۰/۶۹~۰/۸۵	NR	NR	NR	

NR گزارش‌نشده

به‌عنوان یک نمونه کار در زبان سوئدی، [۱۲] اعلام می‌دارد که از آن‌جایی که یادداشت‌های پزشکی[2] با عجله نوشته می‌شوند دارای غلط‌های املایی است که ممکن است ضررهای زیادی را در تشخیص و درمان بیماری به دنبال داشته باشد [۱۲]. طبق بررسی‌های انجام‌یافته [۶] حدوداً ۳۰٪ توکن‌ها در این نوشته‌ها غیرقابل تشخیص هستند. برای پیدا کردن غلط‌های املایی در یادداشت‌های پزشکی سوئدی در [۱۲] از ابزارهای پردازش متن استفاده شده است. این الگوریتم با استفاده از فهرستی از واژه‌هایی که از یک فرهنگ لغت پزشکی و یک لیست از اختصارات به‌دست می‌آورد، به تشخیص غلط‌های املایی می‌پردازد. این الگوریتم توانسته است غلط‌های املایی را به ۷/۶٪ نسبت به روش کنترل دستی کاهش دهد.

بر روی زبان‌های مختلف از جمله زبان فارسی نیز مطالعاتی برای غلط‌های تایپی انجام گرفته است. روش ارائه‌شده در [۱۳] غلط‌های املایی را تشخیص و تصحیح کرده و رتبه‌بندی در پاسخ‌های پیشنهادی را انجام می‌دهد. همچنین در [۱۴] یک سیستم خبره جهت تصحیح خودکار خطاهای املایی ارائه شده است. که برای تشخیص کلمات غلط از یک لغت‌نامه فارسی استفاده می‌کند و پس از آن، مناسب‌ترین کلمه صحیح با استفاده از توابع ابتکاری مختلفی انتخاب می‌گردد. لغت‌نامه آن‌ها بر اساس مجموعه‌ای از لغات ایرنا ساخته شده است. در [۱۵] با چک‌کردن گرامر و اشتباهات دستوری به تشخیص غلط‌های املایی می‌پردازد که با این تمهیدات دقت[3] ۷۰٪ و یادآوری[4] ۸۳٪ را به‌دست آورده است. همچنین در [۱۶] یک روش آماری بر اساس اطلاعات

متقابل کلمات فارسی برای مقابله با غلط‌های املایی ارائه شده است که توانسته یادآوری ۸۰/۵٪ و دقت ۸۷٪ را به‌دست آورد. در [۳] نیز یک تصحیح خودکار غلط‌های تایپی فارسی به‌کمک شبکه عصبی پرسپترون چندلایه ارائه شده است و این شبکه را با شبکه هاپفید مقایسه نموده است ولی خاطرنشان کرده که دقت عملکرد هر دو شبکه در هنگام افزایش واژگان کاهش می‌یابد.

در زبان عربی هم بر روی غلط‌های املایی و گرامرها و حروف آن‌ها تحقیقاتی انجام گرفته است [۲۵-۱۷] و ازآنجایی‌که کارهای انجام‌گرفته بر روی زبان فارسی زیاد نیستند، به بررسی کارهای زبان عربی می‌پردازیم چون نویسه‌های مشابهی با زبان فارسی دارند. در [۲۶] یک کار بر روی زبان عربی ارائه شده است که در دو مرحله به غلط‌گیری املایی می‌پردازد. مرحله اول تشخیص و مرحله دوم تصحیح خطاهای املایی است. در تشخیص خطاهای املایی، دو نوع خطا معرفی شده است: ۱: خطاهای بدشکل[۵]، ۲: خطاهایی با معنای نادرست[۶]. که تشخیص نوع اول راحت‌تر است. در روش پیشنهادی برای این عمل از BAMA[۷] که شامل ۳۸۶۰۰ اصل موضوع می‌باشد، استفاده شده است. در [۲۷] برای اصلاح غلط‌های املایی در زبان عربی، الگوریتم معرفی کردند که از روش لونشتین[۸] به‌همراه یک لغت‌نامه استفاده می‌کند. در این روش کلمات غلط با استفاده از لغت‌نامه‌ای تشخیص داده می‌شود. سپس از bi-gram در الگوریتم Levenshtein کمک گرفته و روشی ارائه کرده است که با استفاده از آن به تصحیح کلمات اشتباه می‌پردازد به‌طوری‌که با نسبتی که با این روش برای تمام پیشنهادهای موجود در لغت‌نامه به دست می‌آورد، تصمیم می‌گیرد که کدام کلمه صحیح‌تر است و آن کلمه را برای تصحیح‌شده کلمه غلط پیشنهاد می‌دهد. در [۲۸] بر روی تشخیص و تصحیح خطاهای املایی زبان عربی تحقیقاتی انجام داده‌اند آن‌ها از یک لغت‌نامه با ۹/۲ میلیون کلمه استفاده می‌کنند و با بهبود مدل زبان به‌وسیله تحلیل خطاها در منابع مختلف یک زیرمجموعه بهینه ایجاد کرده‌اند که این زیرمجموعه در تشخیص خطاها بهتر عمل می‌کند. در [۲۹] یک رویکرد مبتنی بر احتمال برای تصحیح غلط املایی متن عربی ارائه شده است. در این روش یک سیستم دولایه مطرح شده است به‌طوری‌که به‌طور خودکار کلمات غلط را در یک پایگاه‌داده بزرگ اصلاح می‌کند. در لایه اول یک لیست شامل جایگزین‌های ممکن برای هر کلمه غلط که با استفاده از لونشتین و برخی دیگر از روش‌ها تهیه می‌شود. در مرحله بعد کلمه صحیح با یک احتمالی که با روش n-gram داده می‌شود، مشخص می‌گردد. در این روش یک پایگاه‌داده بزرگ برای ساخت کلمات معادل و نیز ارزیابی سیستم وجود دارد. در آزمایش‌ها مشخص شد که هرقدر مجموعه آموزشی بزرگ‌تر باشد به همان میزان دقت عمل بالا می‌رود.

تمامی روش‌هایی که تاکنون بررسی کردیم، همگی مبتنی بر لغت‌نامه می‌باشند. (جداول ۲، ۶ و ۸ را ببینید) نگاهی به جدیدترین مقالات موجود در این زمینه مانند [۳۲ - ۳۰] نیز نشان می‌دهد که همگی آن‌ها نیز مبتنی بر لغت‌نامه می‌باشند. درحالی‌که استفاده از لغت‌نامه سه مشکل اصلی را شامل است: نخست آنکه ساخت لغت‌نامه، زمان‌بر و پرهزینه است، دوم اینکه لغت‌نامه ساخته‌شده در یک زمینه یا کاربرد قابل‌استفاده برای زمینه‌ها یا کاربردهای دیگر نیست و سوم اینکه لغات و اصطلاحات جدید در آن زمینه قابل‌شناسایی نخواهد بود (مگر آنکه لغت‌نامه به‌روزرسانی شود). ایده و نوآوری اصلی این مقاله ارائه روشی است که بدون استفاده از لغت‌نامه بتوان غلط‌های تایپی را شناسایی کرد.

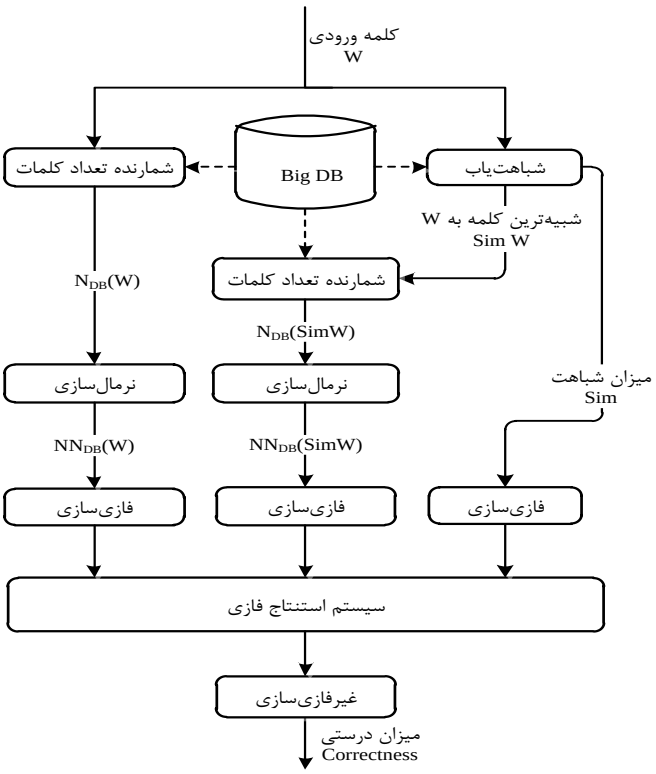
۳- روش پیشنهادی

همان‌گونه که گفتیم هدف این مقاله ارائه روشی است که بتواند غلط‌های تایپی را در ورودی (و نیز داخل) پایگاه‌داده‌های بزرگ بیابد و این کار را بدون استفاده از لغت‌نامه انجام دهد.

ایده اصلی مورداستفاده در این مقاله این است که یک پایگاه‌داده به‌اندازه کافی بزرگ حاوی لغات تخصصی حیطه‌ای است که پایگاه‌داده روی آن ساخته شده است. لذا از خود این پایگاه‌داده می‌توان برای ارزیابی کلمات جدید استفاده کرد. بدیهی است که این ایده زمانی صحیح است که پایگاه‌داده مورداستفاده به‌اندازه کافی بزرگ باشد. امروزه اکثر سازمان‌ها دارای پایگاه‌داده‌هایی حاوی چندین میلیون رکورد می‌باشند. ازاین‌رو پیش‌فرض وجود پایگاه‌داده به‌اندازه کافی بزرگ ارضا می‌شود. با این فرض، می‌توان با استناد به داده‌های درون چنین پایگاه‌داده‌ای و پایه قرار دادن آن‌ها، در خصوص داده‌های جدید تصمیم‌گیری کرد. البته باید در نظر گرفت که داده‌های موجود در پایگاه‌داده‌های موجود در پایگاه‌داده لزوماً صحیح نیستند. اما باید توجه کرد که داده‌های غلط معمولاً دارای دو ویژگی هستند: اول آنکه تعداد تکرار آن‌ها کم است و دوم اینکه معمولاً کلمه دیگری شبیه به آن‌ها در پایگاه‌داده وجود دارد. لذا در روش پیشنهادی هم به تکرار کلمات توجه شده است و هم به وجود کلمات مشابه.

شکل **۱** معماری سیستم پیشنهادی را نشان می‌دهد. در این شکل منظور از BigDB همان پایگاه‌داده بزرگی است که معمولاً در سازمان‌ها وجود دارد. همان‌گونه که دیده می‌شود، روش پیشنهادی نه تنها بسامد کلمه را موردتوجه قرار می‌دهد بلکه با استفاده از یک شباهت‌یاب، شبیه‌ترین کلمه به کلمه ورودی را نیز یافته و میزان شباهت و بسامد شبیه‌ترین کلمه را نیز مدنظر قرار می‌دهد.

بسته به ویژگی‌های موردنیاز سیستم، از کلیه روش‌های شباهت‌یابی مطرح مانند Edit Distance, Jaro, Q-gram, Smith waterman, Soundex و غیره (با توجه به ویژگی‌های هر یک) می‌توان برای شباهت‌یابی استفاده کرد. در تحقیق حاضر از روش Edit Distance [۴] به این منظور استفاده کرده‌ایم.



شکل ۱: معماری سیستم پیشنهادی

همچنین میزان بسامد کلمات از یک ماژول نرمال‌ساز عبور داده می‌شوند تا تأثیر اندازه پایگاه‌داده بر بسامد کنترل و حذف شود. زیرا بسامد کلمات وابسته به بزرگی پایگاه‌داده است و نیز با توجه به اضافه و حذف از پایگاه‌داده در طول زمان نیز می‌تواند تغییر کند. نرمال‌سازی با استفاده از فرمول (۳) و (۴) انجام می‌شود.

$$\mathrm{RecMean}=\frac{N_{\mathrm{Rec}}(DB)}{N_{\mathrm{Word\ Type}}(DB)} \qquad (۳)$$

$$NN_{DB}(W)=\frac{N_{DB}(W)}{\mathrm{RecMean}}=\frac{N_{DB}(W)\times N_{\mathrm{Word\ Type}}(DB)}{N_{\mathrm{Rec}}(DB)} \qquad (۴)$$

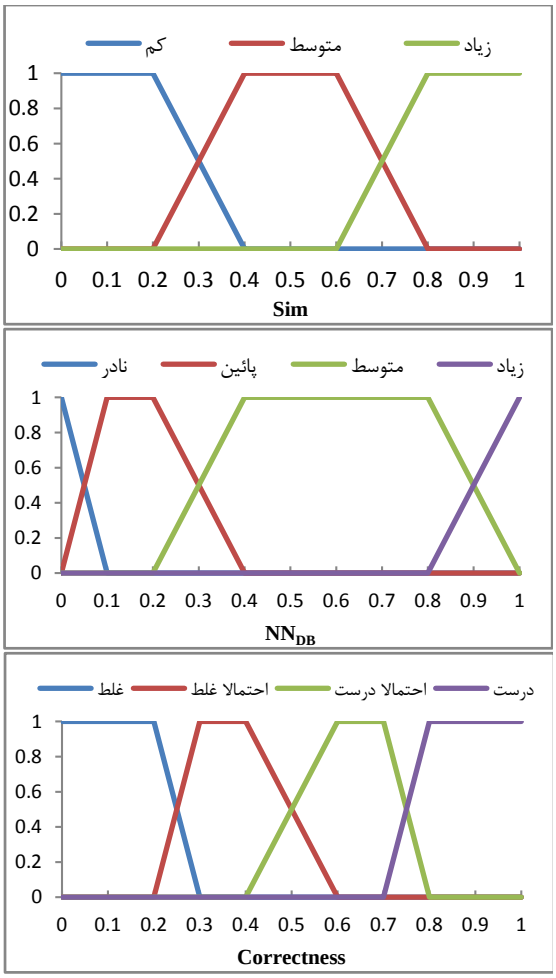
که در آن $NN_{DB}(w)$ تعداد تکرار نرمال‌شده کلمه W، $N_{DB}(W)$ تعداد تکرار کلمه W در پایگاه‌داده، $N_{Rec}(DB)$ تعداد کل رکوردهای پایگاه‌داده و $N_{WordType}(DB)$ تعداد «نوع کلمه» در پایگاه‌داده را نشان می‌دهد. منظور از «نوع کلمه» (Word Type) تعداد کلمات، بدون در نظر گرفتن تکرار است.

پس از آن از یک سیستم فازی برای اتخاذ تصمیم در مورد میزان درستی کلمه ورودی استفاده می‌شود. سیستم‌های فازی یکی از تثبیت‌شده‌ترین بخش‌های نظریه منطق فازی است. تحقیقات گسترده در این حوزه سودمند بودن این سیستم‌ها را نشان داده است [۳۳].

سیستم فازی مورداستفاده از فازی‌ساز منفرد، موتور استنتاج، حاصل‌ضرب ممدانی، S-norm ماکزیمم، t-norm حاصل‌ضرب، استلزام حاصل‌ضرب ممدانی مبتنی بر ترکیب قواعد جداگانه و غیرفازی‌سازی میانگین مراکز بهره می‌برد.

در شکل ۲ مجموعه‌های فازی مورداستفاده در سیستم نشان داده شده است. مقادیر زبانی و مجموعه‌های فازی معادل آن‌ها بر اساس اجماع آرای ۳ نفر فرد خبره تهیه و تنظیم شده است که مبتنی بر تجربه انسانی و یافته‌های عملی بر روی پایگاه‌داده بزرگ است.

همان‌گونه که از شکل ۲ مشخص است. از توابع تعلق ذوزنقه‌ای برای تعریف توابع تعلق متغیرهای زبانی سیستم استفاده شده است. که توابع مربوطه در پیوست آورده شده است.



شکل ۲: مجموعه‌های فازی مورداستفاده در سیستم استنتاج فازی

پایگاه قواعد مورداستفاده در سیستم استنتاج فازی در شکل ۳ آمده است. این قواعد برای استنتاج از روی ۳ ورودی سیستم به کار می‌روند همان‌گونه که قبلاً گفته شد استنتاج در این سیستم، مبتنی بر استنتاج از قواعد جداگانه و سپس ترکیب آن‌ها است. لذا موتور استنتاج بر اساس استلزام حاصل‌ضرب ممدانی، خروجی فازی را به دست می‌آورد. این خروجی تحویل غیرفازی‌ساز می‌شود تا خروجی نهایی سیستم استخراج گردد.

- If (NN_{DB}(W) is نادر) AND (Sim is زیاد) then (Correctness is غلط)
- If (NN_{DB}(W) is نادر) AND Not (Sim is زیاد) then (Correctness is درست)
- If (NN_{DB}(W) is متوسط) OR (NN_{DB}(W) is زیاد) then (Correctness is درست)
- If (NN_{DB}(W) is پایین) AND (Sim is زیاد) AND (NN_{DB}(SimW) is زیاد) then (Correctness is احتمالاً غلط)
- If (NN_{DB}(W) is پایین) AND Not ((Sim is زیاد) AND (NN_{DB}(SimW) is زیاد)) then (Correctness is احتمالاً درست)

شکل ۳: قوانین فازی مورداستفاده در پایگاه دانش سیستم استنتاج فازی

٤- معیارهای ارزیابی

معیارهای متعددی برای ارزیابی سیستم در کارهای انجام‌شده قبلی مشاهده می‌شود. از بین این معیارها، موارد زیر برای ارزیابی سیستم مورداستفاده قرار گرفتند.

TP: نشان‌دهنده تعداد کلماتی است که سیستم آن‌ها را درست در نظر گرفته و در ارزیابی انسانی نیز درست تشخیص داده شده است (*تعداد کلماتی که سیستم به‌درستی به آن برچسب درست زده است*).

FP: نشان‌دهنده تعداد کلماتی است که سیستم آن‌ها را درست در نظر گرفته اما در ارزیابی انسانی به آن برچسب غلط زده شده است (*تعداد کلماتی که سیستم به‌اشتباه به آن برچسب درست زده است*).

TN: نشان‌دهنده تعداد کلماتی است که سیستم آن‌ها را غلط در نظر گرفته و در ارزیابی انسانی به آن برچسب غلط زده شده است (*تعداد کلماتی که سیستم به‌درستی به آن برچسب غلط زده است*).

FN: نشان‌دهنده تعداد کلماتی است که سیستم آن‌ها را غلط در نظر گرفته اما در ارزیابی انسانی به آن برچسب درست زده شده است (*تعداد کلماتی که سیستم به‌اشتباه به آن برچسب غلط زده است*).

صحت (Accuracy): نشان‌دهنده نسبت تعداد کلماتی است که سیستم به‌درستی آن‌ها را برچسب زده به تعداد کل کلماتی که سیستم برچسب زده است. یعنی:

$$Ac = \frac{TP + TN}{TP + TN + FP + FN} \quad (۵)$$

دقت (Precision): نشان‌دهنده نسبت تعداد مواردی است که به‌درستی برچسب درست داده است به تعداد کل مواردی که برچسب درست داده است. یعنی:

$$Pr = \frac{TP}{TP + FP} \quad (۶)$$

یادآوری (Recall): نشان‌دهنده نسبت تعداد مواردی است که سیستم به‌درستی برچسب درست داده است به تعداد کل مواردی که واقعاً درست بوده است. یعنی:

$$Re = \frac{TP}{TP + FN} \quad (۷)$$

معیار F (F-measure): به‌صورت ترکیب دو معیار دقت و یادآوری تعریف می‌شود. فرمول کلی معیار F که به‌صورت F_β تعریف می‌شود به‌صورت زیر است:

$$F_\beta = \frac{\left(1+\beta^2\right) \times Pr \times Re}{\beta^2 \times Pr + Re} \quad (۸)$$

به‌طورمعمول حالت خاص این معیار با مقدار $\beta = 1$ مورداستفاده قرار می‌گیرد که در این صورت معیار F_1 به‌صورت زیر حاصل می‌شود:

$$F_1 = \frac{2 \times Pr \times Re}{Pr + Re} \quad (۹)$$

F_1 در حقیقت میانگین توافقی (Harmonic Mean) دو معیار Pr و Re است.

٥- مجموعه داده مورداستفاده

برای ارزیابی سیستم در زبان فارسی از دو مجموعه داده شامل یک دادگان کوچک و یک دادگان بزرگ استفاده گردید. مجموعه دادگان کوچک دارای ۱۰۰۰۰ رکورد و دادگان بزرگ حاوی ۳۰۰۰۰ رکورد داده‌ای بوده رکوردهای این دو مجموعه مجزا از یکدیگر و بدون اشتراک بودند. این مجموعه داده از یک پایگاه داده واقعی با حدود ۱۰ میلیون و ۲۰۰ هزار رکورد انتخاب و جداسازی شده و توسط انسان ارزیابی شده بودند. جدول ۳ خلاصه اطلاعات این مجموعه داده‌ها را نشان می‌دهد.

برای ارزیابی سیستم روی زبان عربی نیز از یک دادگان حاوی ۱۰۸۰ رکورد که از طرف انسان ارزیابی شده بود، استفاده شد. مشخصات این مجموعه داده نیز در جدول ۳ آمده است.

٦- ارزیابی روش پیشنهادی

برای ارزیابی روش پیشنهادی آن را روی مجموعه داده‌های ذکرشده در بخش گذشته اجرا نمودیم. همان‌گونه که از پیکربندی سیستم فازی مشخص است، خروجی سیستم عددی بین صفر تا یک است. اما برای

استفاده از معیارهای مورداشاره در بخش ۴ لازم است خروجی سیستم با دو کلاس «درست» و «غلط» برچسب زده شود. برای انجام این تبدیل از یک سیستم FIS با یک ورودی و یک خروجی به‌عنوان سیستم ثانویه یا سیستم مبدل کمک گرفته شد. خروجی سیستم قبلی، ورودی این سیستم را تغذیه می‌کند. سیستم مبدل از فازی‌سازی منفرد، موتور استنتاج ممدانی، S-norm ماکزیمم، t-norm مینیمم، استلزام مینیمم ممدانی و غیرفازی‌سازی ماکزیمم بهره می‌برد. خروجی این سیستم برای انجام آزمایش‌ها مورداستفاده قرار گرفت.

بدین ترتیب کلیه نمونه‌های آزمون یک‌بار توسط سیستم و بار دیگر توسط انسان ارزیابی شد. سپس نظرات انسان و سیستم به صورتی که در بخش ۴ ذکر نمودیم، با یکدیگر ترکیب و مورداستفاده قرار گرفتند.

برای ارزیابی روش پیشنهادی، ابتدا آن را روی دو مجموعه داده فارسی ذکرشده در بخش ۵ یعنی دادگان فارسی کوچک و دادگان فارسی بزرگ اجرا کردیم که نتایج آن در جدول ۴ آمده است. این آزمایش جهت بررسی اثر اندازه دادگان روی نتایج روش ترتیب داده شده بود. با توجه به نتایج F_1 و صحت در جدول ۴ مشخص می‌شود که همان‌طور که انتظار می‌رفت با بزرگ شدن دادگان عملکرد روش پیشنهادی نیز بهبود می‌یابد.

برای بررسی عملکرد روش پیشنهادی در مقایسه با روش مبتنی لغت‌نامه، آزمایش دیگری ترتیب داده شد. بدین‌صورت که روش مطرح‌شده در [۱۳] پیاده‌سازی و با روش پیشنهادی روی دادگان فارسی بزرگ مقایسه گردید. نتایج این آزمایش در جدول ۵ آمده است. لازم به ذکر است که روش مبتنی بر لغت‌نامه از یک لغت‌نامه حاوی ۲۷۰۰ لغت بهره می‌برد. همان‌گونه که ملاحظه می‌شود روش پیشنهادی بهتر از روش دیگر عمل کرده است. دلیل این امر را باید در وجود لغات تخصصی خارج از لغت‌نامه دانست.

برای مقایسه عملکرد روش پیشنهادی با دیگر روش‌های مبتنی بر لغت‌نامه، نتایج گزارش‌شده توسط سایر محققین بر اساس معیارهای گزارش‌شده توسط آن‌ها در جدول ۶ آورده شده است. نتایج این جدول ضمن تأیید نتایج کسب‌شده از پیاده‌سازی روش مبتنی بر لغت‌نامه، برتری روش پیشنهادی را نشان می‌دهد.

استفاده از داده‌های داخل پایگاه‌داده، این امکان را به سیستم می‌دهد که مستقل از زبان رفتار نماید. برای بررسی این قابلیت، آزمایش دیگری ترتیب داده شد. بدین ترتیب که یک مجموعه داده به شرح قسمت ۵ مقاله، روی زبان عربی تهیه شد. علاوه‌برآن لغت‌نامه‌ای نیز برای زبان عربی تهیه گردید. همچنین دو روش مبتنی بر لغت‌نامه پیاده‌سازی گردید: اول روشی که در بخش تشخیص مقاله [۲۷] به‌کاررفته و دوم روش تشخیصی مقاله [۲۹]. سپس این دو روش (با استفاده از لغت‌نامه عربی تهیه‌شده)، به‌همراه روش پیشنهادی، روی دادگان عربی اعمال شدند. نتایج این آزمایش در جدول ۷ آمده است. همان‌گونه که مشخص است روش پیشنهادی نتایج قابل قبولی در مقایسه با دیگر روش‌ها به دست آورده است.

برای مقایسه عملکرد روش پیشنهادی روی زبان عربی با دیگر روش‌های مبتنی بر لغت‌نامه، نتایج گزارش‌شده توسط سایر محققین (بر اساس نتایج گزارش‌شده توسط آن‌ها) در جدول ۸ آمده است. همان‌گونه که از این جدول مشخص است روش پیشنهادی دارای برتری قابل قبولی است.

جدول ۳: مشخصات مجموعه داده‌های مورداستفاده برای ارزیابی سیستم. مجموعه داده‌ها مجزا از هم و بدون اشتراک می‌باشند.

	تعداد رکوردهای مجموعه آموزشی	تعداد نوع کلمه	تعداد نمونه	RecMean
دادگان فارسی کوچک	۱۰۰۰۰	۲۰۲۱	۲۵	۴/۹۴
دادگان فارسی بزرگ	۳۰۰۰۰	۳۴۷۱	۲۷۰۰	۸/۶۴
دادگان عربی	۱۰۸۰	۸۵۱	۳۵	۱/۲۷

جدول ۴: نتایج حاصل از ارزیابی روش پیشنهادی روی دو مجموعه داده مورداستفاده.

معیارها	TP	TN	FP	FN	Pr	Re	F_1	صحت Ac
دادگان فارسی کوچک	۱۵	۱	۸	۱	۰/۶۵۲	۰/۹۳۷	۰/۷۶۹	۰/۶۴
دادگان فارسی بزرگ	۲۰۶۴	۹۱	۱۴۳	۴۰۲	۰/۹۳۵	۰/۸۳۶	۰/۸۸۸	۰/۷۹۸

جدول ۵: مقایسه نتایج حاصل از روش پیشنهادی با روش مبتنی بر لغت‌نامه (پیاده‌سازی شده توسط مؤلفین از روی [۱۳]) روی دادگان فارسی بزرگ

روش مورداستفاده	TP	TN	FP	FN	Pr	Re	F_1	صحت Ac
روش پیشنهادی	۲۰۶۴	۹۱	۱۴۳	۴۰۲	۰/۹۳۵	**۰/۸۳۶**	**۰/۸۸**	**۰/۷۹۸**
روش مبتنی بر لغت‌نامه Mosavi -2013 [۱۳]	۱۵۹۵	۱۸۳	۰	۹۲۳	**۱**	۰/۶۳۲	۰/۷۷۵	۰/۶۵۷

جدول ۶: مقایسه روش پیشنهادی با سایر روش‌ها روی زبان فارسی

روش	استفاده از لغت‌نامه	خلاصه نتایج			
		Pr	Re	F_1	توضیحات
پیشنهادی	**خیر**	**۰/۹۳۵**	**۰/۸۳۶**	**۰/۸۸**	در ۱۰۷۷۲ کلمه لغت‌نامه با ۲۰۰ بسامد تکرار
Ehsan-2010	**بله**	۰/۷	۰/۸۳	۰/۷۶	
Faili-2010	**بله**	۰/۸۰۵	۰/۸۷	۰/۸۳	
عربی سرخی-۱۳۸۵	**بله**	۰/۵۳	NR	NR	
شاهمیری-۱۳۸۶	**بله**	۰/۷۷	NR	NR	

NR گزارش‌نشده

جدول ۷: مقایسه نتایج روش پیشنهادی با روش مبتنی بر لغت‌نامه (پیاده‌سازی شده توسط مؤلفین از روی [۲۷] و [۲۹]) بر روی زبان عربی

روش مورداستفاده	TP	TN	FP	FN	Pr	Re	F_1	صحت Ac
روش پیشنهادی	۲۸	۰	۱	۶	**۰/۸۲۳۵**	**۰/۹۶۵**	**۰/۸۸**	**۰/۸**
روش مبتنی بر لغت‌نامه [۲۷]	۲۲	۵	۰	۸	۱	۰/۷۳	۰/۸۴	۰/۷۷
روش مبتنی بر لغت‌نامه [۲۹]	۱۸	۵	۴	۸	۰/۸۱	۰/۶۹	۰/۷۵	۰/۶۵

جدول ۸: مقایسه روش پیشنهادی با سایر روش‌ها روی زبان عربی

روش	استفاده از لغت‌نامه	خلاصه نتایج		
		Pr	Re	F1
پیشنهادی	**خیر**	۰/۸۲۳۵	**۰/۹۶۵**	**۰/۸۸**
Shalan-2010	**بله**	**۰/۹ ~ ۰/۹۸**	۰/۸۶ ~۰/۹۴	۰/۸۸ ~۰/ ۹۵
Attia-2015	**بله**	۰/۶۷۵	۰/۷۴۶	۰/۷۱
Bouamor-2015	**بله**	۰/۷۱۴	۰/۶۵	۰/۶۸
Djamel-2015	**بله**	۰/۷۱	۰/۶۵	۰/۶۸
Bougares-2015	**بله**	۰/۷۳	۰/۶۶	۰/۷
Hassan-2014	**بله**	۰/۵۹	۰/۵۷	۰/۵۸
Mubarak-2014	**بله**	۰/۶	۰/۵۹۲	۰/۵۹۹

۷- نتیجه‌گیری

در این مقاله ما روشی برای پیدا کردن غلط‌های تایپی در پایگاه داده‌های بزرگ ارائه کردیم. مزیت بزرگ روش ارائه‌شده این است که برای بررسی درست یا غلط بودن کلمه نیاز به لغت‌نامه ندارد. استقلال از لغت‌نامه یک مزیت بزرگ برای روش محسوب می‌شود زیرا روش‌های وابسته به لغت‌نامه مجبورند برای هر کار خاص یا هر محیط خاص یک لغت‌نامه تهیه کنند و تهیه لغت‌نامه کاری است پرزحمت و زمان‌بر. درحالی‌که روش پیشنهادی را به‌راحتی می‌توان در هر محیطی به کار برد و نیازی به صرف زمان و هزینه برای اختصاصی کردن[۹] آن نیست. تنها پیش‌نیاز روش پیشنهادی، وجود داده کافی در پایگاه‌داده محیطی که قرار است در آن کار کند، است که البته در بیش‌تر سازمان‌ها چنین شرطی مهیا است.

در ارزیابی روش پیشنهادی نشان می‌دهد که با افزایش بزرگی پایگاه‌داده، دقت سیستم نیز افزایش می‌یابد زیرا نرخ قبول غلط کاهش می‌یابد. این امر نشان می‌دهد که به‌مرور زمان که میزان داده‌های سازمان افزایش می‌یابد، سیستم کارکرد بهتری خواهد داشت.

همچنین مقایسه روش پیشنهادی با سایر روش‌ها نشان می‌دهد که علی‌رغم اینکه روش پیشنهادی از یک منبع دانش بزرگ یعنی لغت‌نامه بی‌بهره است، اما به‌خوبی می‌تواند با روش‌های دیگری که در این زمینه وجود دارند رقابت کند.

سپاسگزاری

این مقاله از طرح تحقیقاتی که با بودجه پژوهشی و حمایت مالی دانشگاه آزاد اسلامی واحد تبریز به انجام رسیده است، استخراج شده است. لذا نویسندگان این مقاله از دانشگاه آزاد اسلامی واحد تبریز کمال سپاسگزاری را دارند.

پیوست

توابع تعلق متغیرهای زبانی سیستم همان‌گونه که در شکل ۲ نیز مشخص است برای توضیح بیش‌تر در زیر به آن‌ها اشاره می‌شود. برای متغیر زبانی Sim، سه مقدار زبانی کم (small)، متوسط (medium)، زیاد (large) تعریف شده است که توابع تعلق آن به‌صورت زیر است:

$$\mu_{small}(sim)=\begin{cases}1 & sim \le 0.2\\ \dfrac{0.4-sim}{0.2} & 0.2 \le sim \le 0.4\\ 0 & otherwise\end{cases} \qquad (۱۰)$$

$$\mu_{\text{medium}}(\text{sim}) = \begin{cases} 1 & 0.4 \le \text{sim} \le 0.6 \\ \dfrac{\text{sim}-0.2}{0.2} & 0.2 \le \text{sim} \le 0.4 \\ \dfrac{0.8-\text{sim}}{0.2} & 0.6 \le \text{sim} \le 0.8 \\ 0 & \text{otherwise} \end{cases} \quad (۱۱)$$

$$\mu_{\text{large}}(\text{sim}) = \begin{cases} 1 & 0.8 \le \text{sim} \\ \dfrac{\text{sim}-0.6}{0.2} & 0.6 \le \text{sim} \le 0.8 \\ 0 & \text{otherwise} \end{cases} \quad (۱۲)$$

همچنین برای دو متغیر زبانی NN_{DB} (شامل $NN_{DB}(w)$ و $NN_{DB}(sim)$) چهار مقدار زبانی نادر (rare)، پایین (low)، متوسط (medium) و زیاد (large) به‌شرح زیر تعریف شده است:

$$\mu_{\text{rare}}(NN_{DB}) = \begin{cases} \dfrac{0.1-NN_{DB}}{0.1} & NN_{DB} \le 0.1 \\ 0 & \text{otherwise} \end{cases} \quad (۱۳)$$

$$\mu_{\text{low}}(NN_{DB}) = \begin{cases} 1 & 0.1 \le NN_{DB} \le 0.2 \\ \dfrac{NN_{DB}}{0.1} & 0 \le NN_{DB} \le 0.1 \\ \dfrac{0.4-NN_{DB}}{0.2} & 0.2 \le NN_{DB} \le 0.4 \\ 0 & \text{otherwise} \end{cases} \quad (۱۴)$$

$$\mu_{\text{medium}}(NN_{DB}) = \begin{cases} 1 & 0.4 \le NN_{DB} \le 0.8 \\ \dfrac{NN_{DB}-0.2}{0.2} & 0.2 \le NN_{DB} \le 0.4 \\ \dfrac{0.8-NN_{DB}}{0.2} & 0.8 \le NN_{DB} \le 1 \\ 0 & \text{otherwise} \end{cases} \quad (۱۵)$$

$$\mu_{\text{large}}(NN_{DB}) = \begin{cases} \dfrac{NN_{DB}-0.8}{0.2} & 0.8 \le NN_{DB} \le 1 \\ 0 & \text{otherwise} \end{cases} \quad (۱۶)$$

نهایتاً برای متغیر زبانی Correctness نیز چهار مقدار زبانی غلط (incorrect)، احتمالاً غلط (likely incorrect)، احتمالاً درست (likely correct) و درست (correct) به‌شرح زیر تعریف شده‌اند:

$$\mu_{\text{incorrect}}(\text{cor}) = \begin{cases} 1 & 0 \le \text{cor} \le 0.2 \\ \dfrac{0.3-\text{cor}}{0.1} & 0.2 \le \text{cor} \le 0.3 \\ 0 & \text{otherwise} \end{cases} \quad (۱۷)$$

$$\mu_{\text{likely incorrect}}(\text{cor}) = \begin{cases} 1 & 0.3 \le \text{cor} \le 0.4 \\ \dfrac{\text{cor}-0.2}{0.1} & 0.2 \le \text{cor} \le 0.3 \\ \dfrac{0.6-\text{cor}}{0.2} & 0.4 \le \text{cor} \le 0.6 \\ 0 & \text{otherwise} \end{cases} \quad (۱۸)$$

$$\mu_{\text{likely correct}}(\text{cor}) = \begin{cases} 1 & 0.6 \le \text{cor} \le 0.7 \\ \dfrac{\text{cor}-0.4}{0.2} & 0.4 \le \text{cor} \le 0.6 \\ \dfrac{0.8-\text{cor}}{0.1} & 0.7 \le \text{cor} \le 0.8 \\ 0 & \text{otherwise} \end{cases} \quad (۱۹)$$

$$\mu_{\text{correct}}(\text{cor}) = \begin{cases} 1 & 0.8 \le \text{cor} \le 1 \\ \dfrac{\text{cor}-0.7}{0.1} & 0.7 \le \text{cor} \le 0.8 \\ 0 & \text{otherwise} \end{cases} \quad (۲۰)$$

مراجع

[1] A. K. Elmagarmid, P. G. Ipeirotis, Vassilios, and S. Verykios, "Duplicate record detection: a survey," *IEEE Transactions on Knowledge And Data Engineering*, vol. 19, pp. 1-16, 2007.

[۲] رضا اعتماد، محسن کیا و محمدصادق سپاسیان، «پیشنهاد یک روش جدید جهت برنامه‌ریزی توسعه تولید (GEP) بر مبنای تحلیل سلسله‌مراتبی فازی (FAHP)،» *مجله مهندسی برق دانشگاه تبریز*، جلد ۴۴، شماره ۲، صفحات ۱-۱۰، ۱۳۹۳.

[۳] امیر شهاب شاهمیری، رضا صفابخشی و رسول دژکام، «تصحیح خودکار غلط‌های تایپی به‌کمک شبکه عصبی مصنوعی،» *مجله انجمن مهندسین برق الکترونیک ایران*، جلد ۵، شماره ۱، صفحات ۱۶-۲۹، ۱۳۸۷.

[4] W. W. Cohen, P. Ravikumar, and S. E. Fienberg, "A comparison of string distance metrics for name-matching tasks," *American Association for Artificial Intelli gence*, 2003.

[5] J. L. Peterson, "Computer programs for detecting and correcting spelling errors," *Communication of the ACM*, vol. 23, no. 12, pp. 676-687, 1980.

[6] J. Patrick, M. Sabbagh, S. Jain, and H. Zheng, "Spelling correction in clinical notes with emphasis on first suggestion," *2nd Workshop on Building and Evaluating Resources for Biomedical Text Mining*, Valletta, Malta, 2010.

[7] X. Weijian, H. Peijie, Z. Xinrui, H. Kaiduo, H. Qiang, Ch. Bingzhou, and H. Lei, "Chinese spelling check system based on n-gram model," *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 128-136, 2015.

[8] Y. H. Tseng, L. H. Lee, L. P. Chang, and H. H. Chen, "Introduction to sighan 2015 bake-off for chinese spelling check," *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 32-37, 2015.

[9] Z. Sh, X. Jinhua, H. Jianpeng, Z. Qiao, and Ch. Xueqi, "HANSpeller++: A unified framework for chinese spelling correction," *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 38-45, 2015.

[10] T. H. Chang, H. Ch. Chen, and Ch. H. Yang, "Introduction to a proofreading tool for chinese spelling check task of sighan-8," *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 50-55, 2015.

[11] A. Jain, and M. Jain, "Detection and correction of non word spelling errors in hindi language," *IEEE, Data Maining and Intelligent Computing (ICDMIC)*, New Delhi, pp. 1-5, 2014.

[12] N. Uddin, and D. Hercules, "Detection of spelling errors in swedish clinical text," *1st Nordic Workshop on Evaluation of Spellchecking and Proofing Tools (NorWEST2014)*, 2014.

[13] T. Mosavi, "Farsi spell: a spell-checking system for persian using a large monolingual corpus," *Oxford Scholarship, Literary and Linguistic Computing Advance Access*, 2013.

[۱۴] محسن عرب‌سرخی، هشام فیلی و محمد آزادنیا، «ارائه یک سیستم خبره جهت تصحیح خودکار خطاهای املایی زبان فارسی،» *دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران*، دانشگاه شهید بهشتی، تهران، ایران، ۱۳۸۵.

[15] N. Ehsan, and H. Faili, "Towards grammar checker development for persian language," *Presented at the 6th IEEE International Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE'10)*, 2010.

[16] H. Faili, "Detection and correction of real-word spelling errors in Persian languag," *IEEE, Natural Language Processing and Knowledge Engineering (NLP-KE)*, Beijing, 2010.

[17] M. Attia, M. Al-Badrashiny, and M. Diab, "GWU-HASP-2015@QALB- 2015 shared task: priming spelling candidates with probability," *The Second Workshop on Arabic Natural Language Processing*, 2015.

[18] H. Bouamor, H. Sajjad, N. Durrani, and K. Oflazer "QCMUQ@QALB-2015 shared task: combining character level mt and error-tolerant finite-state recognition for arabic spelling correction," *The Second Workshop on Arabic Natural Language Processing*, pp. 144-149, 2015.

[19] M. Djamel, J. Abualasal, O. Asbayou, M. Gzaw, and R. Abbes, "TECHLIMED@QALB-Shared task 2015: A hybrid arabic error correction system," *The Second Workshop on Arabic Natural Language Processing*, pp. 161-165, 2015.

[20] F. Bougares, and H. Bouamor, "UMMU@ QALB- 2015 Shared Task: Character and Word level SMT pipeline for automatic error correction of arabic text," *The Second Workshop on Arabic Natural Language Processing*, pp. 166-172, 2015.

[21] Y. Hassan, M. Aly, and A. Atiya, "Arabic spelling correction using supervised learning," *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Langauge Processing (ANLP)*, pp. 121–126, 2014.

[22] B. Mohit , A. Rozovskaya, N. Habash, W. Zaghouani, and O. Obeid, "The first qalb shared task on automatic text correction for Arabic," *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Langauge Processing (ANLP)*, pp. 39-47, 2014.

[23] M. Attia, M. Al-Badrashiny, and M. Diab, "Hybrid arabic spelling and punctuation corrector," *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Langauge Processing (ANLP)*, pp. 148-154, 2014.

[24] H. Mubarak, and K. Darwish "Automatic correction of arabic text: a cascaded approach," *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Langauge Processing (ANLP)*, pp. 132–136, 2014.

[25] S. Jeblee, H. Bouamor, W. Zaghouani, and K. Oflazer, "An smt-based system for automatic arabic error correction," *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Langauge Processing (ANLP)*, pp. 137–142, 2014.

[26] K. Shaalan, R. Aref, and A. Fahmy, "An approach for analyzing and correcting spelling errors for non-native arabic learners," *7th International Conference on Informatics and Systems*, 2010.

[27] A. S. Lhoussain, G. Hicham ,and Y. Abdellah, "Adaptating the levenshtein distance to contextual speling correction," *International Journal of Computer Science and Applications*, vol. 12, no. 1, pp. 127-133, 2015.

[28] M. Attia, P. Pecina, Y. Samih, K. Shaalan, and J. V. Genabith, "Arabic spelling error detection and correction," *Natural Language Engineering*, 2015.

[29] M. Alkanhal, M. A. Badrashiny, M. Alghamdi, and A. A. Qabbany, "Automatic stochastic arabic spelling correction with emphasis on space insertions and deletions," *Audio, Speech, and Language Processing, IEEE Transactions*, vol. 20, no.7, pp. 2111-2122, 2012.

[30] S. Gupta, and S. Sharma, "A spelling mistake correction (SMC) model for resolving real-word error," *Springer, Computational Intelligence in Data Mining*, vol. 1 and vol. 410 of the *series Advances in Intelligent Systems and Computing*, pp. 429-438, 2015.

[31] B. Siklósi, A. Novák, and G. Prószéky, "Context-aware correction of spelling errors in Hungarian medical documents," *Elsevier, Computer Speech and Language*, pp. 35 219–233, 2016.

[32] P. H. Hema, and C. Sunitha, "Malayalam spell checker using n-gram method," *Computational Intelligence in Data Mining, Series Advances in Intelligent Systems and Computing*, vol. 1, pp. 217-225, 2015.

[۳۳] فرناز صباحی و محمدرضا اکبرزاده توتونچی، «شناسایی سیستم‌های غیرخطی بر اساس منطق فازی توسعه‌یافته،» *مجله مهندسی برق دانشگاه تبریز*، جلد ۴۴، شماره ۱، صفحات ۲۳-۳۲، ۱۳۹۳.

زیرنویس‌ها

[1] Edit Distanced

[2] Clinical Text

[3] Precision

[4] Recall

[5] Ill-formed word errors

[6] Semantically incorrect errors

[7] Buck Arabic Morphological Analyzer

[8] Levenshtein

[9] Customize