تولید مجموعه ای برای

مقایسه اثر طول واژه و محل رخداد خطا در تصحیح خطای لغوی برای واژگان فارسی

چکیده:

با افزایش روز افزون کاربرد کامپیوتر و دستگاه‌های هوشمند در زندگی روزمره و افزایش متون الکترونیکی غلط‌های املایی در متون الکترونیکی افزایش می‌یابد. انسان‌ها غالباً به دلیل بی‌توجهی و یا عدم دانش کافی خطای تایپی تولید می‌کنند. خطاهای لغوی دسته‌ی بزرگی از خطاهای معمول در متون الکترونیکی و تایپی هستند که توسط انسان و یا نرم افزارها ایجاد می شوند.پژوهش در حوزه‌ی پردازش زبان طبیعی درباره‌ی انواع الگوهای خطای لغوی و ساختار لغوی زبان می‌تواند موجب کارآیی بهتر سامانه‌های غلط‌یاب و شناسایی نوری نویسه‌ها گردد. در همین راستا طول واژه و محل بروز خطای لغوی می‌تواند دقت تشخیص و تصحیح خطا را متاثر کند. در این پژوهش تاثیر طول واژه و محل بروز خطا بر فرآیند تشخیص و تصحیح خطای لغوی مورد مطالعه قرار گرفته است. یک پیکره استاندارد شامل بیش از ۱۰۰۰ مدرک دارای خطای لغوی و مقدار صحیح واژه از چکیده مقالات فارسی و پیکره‌ی خبری پرسیکا تولید شده است. نتایج آزمایش‌ها نشان داد که طول واژه‌ها و محل بروز خطا می‌تواند دقت تشخیص و تصحیح خطایاب‌های شناخته شده در زبان فارسی را تحت تاثیر قرار دهد.

کلیدواژه‌ها:غلط‌یابی لغوی، اثر طول بر دقت غلط‌یاب لغوی، اثر محل رخداد خطا بر غلط‌یاب لغوی، پیکره‌ی برچسب دار لغوی زبان فارسی.

۱

فهرست جداول

فهرست اشکال

فهرست مطالب

فصل اول

مقدمه

۱-مقدمه

در دنیای امروز، هر لحظه اطلاعات باارزشی تولید می‌شود. این اطلاعات با ارزش را باید طوری ذخیره نمود که با کمترین هزینه قابل بازیابی باشد [۱]. برای حفظ کارآیی سامانه های ذخیره و بازیابی، باید محتوای تولید شده از نظر صحّت متن، مورد بررسی قرار گیرد. بدین منظور اکثر سامانه‌هایی که محتوای متنی تولید می‌کنند دارای اجزای ویراستاری و بررسی لغوی متن هستند [۲].

برای مثال می‌توان به ویکیپدیا اشاره نمود. این مجموعه طبق آمار ارائه شده در سال ۲۰۱۷ دارای بیش از ۵ میلیون مقاله انگلیسی متشکل از بیش از ۴۱ میلیون صفحه برای حدود ۳۰ میلیون کاربر است. مسلماً مدیریت و کنترل لغوی چنین حجمی از صفحات دیجیتال بصورت دستی امکان پذیر نیست و باید راهکارهای خودکار برای چنین حجمی از داده‌ها فراهم نمود.

تولید غلط‌یاب‌ها یکی از جمله راه‌کارهایی است که برای بررسی اولیه و یافتن مشکلات تایپی و نگارشی متن بکار می‌رود. امروزه غلط‌یاب‌های متعددی برای زبان‌های متداول مانند انگلیسی وجود دارد. زبان فارسی مشکلات متعددی که از نظر نگارش در رایانه دارد [۳]. بنابر این مشکلات، همانند زبان‌های دیگر غلط‌یاب‌های با کیفیت که بتوان بصورت فراگیر از آن استفاده نمود، وجود ندارد و یا در دسترس عموم نیست [۱], [۴]–[۶].

برای آنکه بتوان در رسته‌ی پردازش متن فارسی، غلط یابی متن، شباهت سنجی متون، شناسایی نوری نویسه‌ها و بطور کلی پردازش زبان طبیعی برای زبان فارسی گام‌های موثری برداشت، نیاز است پیکره‌ها و

مجموعه‌های آزمون مناسبی در دسترس پژوهشگران باشد. در کنار این منابع، نیاز است ابزارهای پردازشی نیز برای زبان فارسی تولید شود تا بتوان از آن منابع به نحو شایسته‌ای در پژوهش‌ها استفاده نمود. بدیهی است بکارگیری ابزارهای استاندارد در پردازش زبانی موجب افزایش کیفیت جستجو و پردازش می شود [۱].

در حال حاضر عدم دسترسی به منابع زبانی برای پژوهش در حوزه‌ی زبان فارسی، موجب شده است که پژوهش‌های این حوزه نسبت به زبان‌های رایج دیگر مانند انگلیسی، آلمانی یا فرانسه با مشکلات متعددی روبرو شود. بنابراین تولید و ساخت منابع و ابزارهای پردازش زبان برای زابان فارسی بسیار مهم و از جمله پیش‌نیازهای پژوهش در این حوزه است [۳]. هر چند در سال‌های اخیر تلاش‌های بسیاری در این حوزه صورت پذیرفته است [۱], [۵], [۷]–[۹] ولی پیشرفت در این حوزه مستلزم تلاش صاحبنظران، پژوهشگران، دانشجویان و علاقه‌مندان برای تولید منابع کیفی به میزان کافی در تمامی رسته‌های پردازش زبان طبیعی است.

هدف از این پژوهش تولید پیکره‌ی (مجموعه‌ی) آزمون برای غلط یابی لغوی و بررسی آثار تاثیر طول واژه ها بر شناسایی و تصحیح خطا است. چنین مجموعه‌هایی برای زبان‌های متداول مانند انگلیسی، آلمانی و فرانسه تولید شده است [۲] و نتایج پژوهش آن در اختیار علاقه‌مندان است. وجود چنین پژوهش‌های زیربنایی موجب می شود اصول غلط‌یابی و تصحیح واژگان براساس قوانین مستخرج از چنین پژوهش‌هایی بتواند عمل بازیافت خطا را با درصد توفیق بیشتری به سرانجام برساند.

پردازش واژگانی کلیه‌ی زبان‌های طبیعی امری دشوار است. ترکیب واژگان، منجر به تشکیل واژگانی می‌شود، که ممکن است در اثر بی‌دقتی کاربران، از دید رایانه به دو یا چند شکل مختلف خوانده شوند. مثلاً در جایی که منظور نویسنده «سیب‌زمینی» است، اگردر اثر بیدقتی «سیب زمینی» نوشته شود، رایانه قادر به تشخیص واژه‌ی اصلی نخواهد بود .دومین دلیل پیچیدگی پردازش واژگانی زبان‌های طبیعی، ترکیب واژها با یکدیگر و تولید واژه‌هایی است که حاوی اطلاعاتی مانند مالکیت، جمع یا مفرد بودن واژه هستند (به عنوان نمونه می‌توان به واژه‌ی«کتاب‌هایشان» اشاره نمود). این واژه‌های جدید در واژه‌نامه‌ها وجود ندارد اما معنای آن‌ها همان معنایی است که در واژه‌ی اولیه نهفته بوده است. از دید رایانه تنها در صورتی دو واژه با هم یکسان هستند، که به یک شکل نوشته شده باشند. سومین دلیل مشکل بودن تفسیر واژه از دید رایانه، آن است که برخی از قاعده‌های تولید واژه در زبان‌های طبیعی، میتوانند واژه‌هایی به وجود آورند که در واژه‌نامه‌ها وجود ندارند (مانند «بازنگریسته»)[۹]. از سوی دیگر، اگر رایانه بتواند تمام قاعده‌های ساخت واژه‌ها را در خود جای دهد، در آن صورت واژگانی که امکان تولید آن‌ها در زبان وجود دارد اما گویشوران زبان تاکنون آن‌ها را به کار نبرده‌اند نیز در زمره‌ی واژه‌های مورد تایید رایانه قرار خواهند گرفت. بنابراین، پردازش واژه‌های یک متن به خودی خود رایانه را با مشکلاتی در تشخیص واژه‌ها مواجه می‌کند. اینها همگی در صورتی هستند که اشتباهات دستور خط فارسی و رایانه‌ای املایی کاربران در نظر گرفته نشود و نیز تمام کاربران قواعد و اصول نسبتاً یکسانی را در نگارش خود به کار برند. ولی عملاً چنین فرضی

امکان‌پذیر نیست و کاربران رایانه با توجه به تعدد آموزش‌ها و قواعد فارسی، روش‌های گوناگونی را بکار خواهند برد.

با توجه به مشکلات متعددی که خط فارسی دارد و غلط‌یابی و پردازش متن را نیز با مشکلات متعددی روبرو خواهد کرد، انجام تحقیقات در خصوص ذات زبان فارسی در رایانه بسیار مفید است و موجب تولید قوانینی خواهد شد که عملاً کارآیی سامانه‌های پردازش متن را بالا می‌برد.

این پژوهش با مقایسه اثر طول واژگان بر محل وقوع خطا، شناسایی خطا و تصحیح خطا، پیکره‌ای استاندارد برای آزمون غلط‌یابی فارسی ارائه می نماید. این پیکره می تواند در بسیاری سامانه‌های دیگر مانند شناسایی نوری نویسه‌ها نیز کاربرد داشته باشد.

در ادامه پژوهش‌های انجام شده در خصوص غلط‌یابی مورد بررسی قرار خواهد گرفت و در این میان نگاه ویژه‌ای به خط فارسی خواهیم داشت. سپس خصوصیات ویژه‌ی خط فارسی در رایانه مورد بررسی قرار خواهد گرفت و پس از آن، روش تهیه پیکره مورد بررسی خواهد گرفت و نتایج آزمودن پیکره نیز در انتها به بحث گذاشته خواهد شد.

فصل دوم

پیشینه پژوهش

۲-۱-تعاریف و مبانی غلط‌یابی متن

وجود ارتباط متقابل میان زبان‌شناسان و متخصصین رایانه، یکی از نیازهای اصلی جامعه‌ی اطلاعاتی امروز و رشد صنعت پردازش الکترونیکی متن در کشور ایران است . اگر در ساختار خط فارسی تغییری برای همگام شدن با این رشد روی ندهد، میزان عقب ماندگی کشور ایران در فنآوری اطلاعات از سایر کشورها جبران ناپذیر خواهد شد. زمانی فرا خواهد رسید که روزنامه‌ها و مقالات خارجی با سرعت بالایی تولید شده و خلاصه‌ی آن‌ها در کسری از ثانیه استخراج می‌شود، در حالی که در ایران حتی کار خطایابی واژگانی را نیز با خطای بالا و به کندی انجام می‌گیرد [۹]. زمانی که موتورهای جستجوی دیگر زبان‌ها می‌توانند هنگام جستجوی یک واژه، مترادف‌ها، ریشه‌ها و سایر مشتقات آن را برای زبان‌های غیرفارسی نیز بازیابی کنند، ما همچنان در حال رفع مشکل چندگانگی کدکاراکتر "ی" و یا مشکل اجزای واژه بوده و انجام یک جستجوی ساده در وبگاه‌های رسمی کشور به درستی انجام صورت نمی‌گیرد. از این رو اهمیت موضوع یکسان‌سازی نحوه‌ی نگارش واژه‌ها و نیز موارد ابهام‌زای موجود در دستور خط فارسی دوچندان می‌شود و باید تصمیمی جدی جهت رفع اشکالات و چالش‌های این بخش اندیشید [۹].

بنابراین یکی از چالش‌های تحقیق در حوزه‌ی پردازش زبان طبیعی کمبود منابع و ابزارها است. زیرا برای زبان فارسی ابزارهای پردازش زبان طبیعی در دسترس نیست و خود محقق می باید این مهم را تهیه نماید. این ابزارها پیش‌نیاز شروع تحقیق در حوزه‌ی پردازش زبان طبیعی است [۱]. ساخت پیکره‌ای برای کنترل اثر

طول واژه و محل رخداد خطای لغوی برای آزمودن سامانه‌های غلط‌یاب و همچنین سامانه‌های شناسایی نوری نویسه‌ها کاربرد دارد.

تشخیص و تصحیح خطای لغوی در متون فارسی بسیار مهم و حائز اهمیت است. زیرا بسیاری از پردازش‌های خودکار و اتوماسیون سیستم‌ها در گرو پردازش صحیح متن فارسی است. به عنوان مثال اگر سامانه های شناسایی نوری نویسه‌ها بتوانند متن را بطور دقیق شناسایی کنند می توان با شناسایی دقیق متن از رایانه برای دسته‌بندی با دقت و سرعت بالا استفاده نمود. بنابراین کاربرد تصحیح متن به تنهایی برای پردازش‌های متنی نیست و بسیاری از کاربردهای دیگر، در گرو پیشرفت در حوزه‌ی پردازش صحیح متن فارسی است.

امروزه با گسترش دسترسی الکترونیکی به اطلاعات در سطح وب، این دسترسی از گستره‌ای از میزبان‌ها با سخت‌افزار و سیستم‌عامل‌های متفاوت صورت می‌پذیرد. این تنوع در ساختار، موجب می‌شود نوشتن و تایپ نویسه‌ها بعضاً با خطا مواجه شود. در بسیاری از دستگاه‌ها از جمله تلفن‌های همراه هوشمند، تبلت‌ها و ... قدرت پردازشی مناسبی برای پردازش متن در کنار کامپیوترهای قدرتمند امروزی وجود دارد. این ویژگی امکان بکارگیری غلط‌یاب و پردازش متن را در تمامی دستگاه‌های الکترونیکی محیّا می‌نماید.

بکارگیری روش‌ها و الگوریتم‌های تشخیص و تصحیح املایی خودکار متن، از چالش‌های حوزه‌ی غلط‌یابی ماشینی است. پژوهش در حوزه‌ی غلط‌یابی ماشینی از حدود ۱۹۶۰ میلادی آغاز شده است. بسیاری از حوزه‌های این حیطه علیرغم انجام پژوهش‌های گسترده، هنوز عملیاتی نشده و تحت مطالعه و پژوهش است.

دلایل کافی برای ادامه‌ی پژوهش‌های آکادمیک درباره‌ی غلط‌یابی و تصحیح خودکار متن وجود دارد، زیرا برترین غلط‌یاب‌های تحقیقاتی و تجاری، کاملاً عملیاتی نشده‌اند، زیرا از نظر حوزه و دامنه‌ی پوشش و همچنین دقّت مشکلات عدیده‌ای دارند. غلط‌یاب‌های ماشینی می‌تواند به عنوان نرم افزار کمکی در کنار بسیاری از سیستم‌ها مانند شناسایی نوری نویسه‌ها (نویسه‌خوان نوری)[1] استفاده شود. زیرا نویسه‌خوان نوری در بهترین شرایط معمولاً دقتی بین ۹۵٪ الی ۹۹٪ دارند [۱۰]، و این بدین معنی است که در بهترین حالت از هر ۱۰۰ کلمه یک کلمه غلط ایجاد می‌کنند. در این مرحله غلط‌یاب ماشینی می‌تواند واژه‌های مشکل‌دار را شناسایی نماید. ولی تمامی کاربردها در این حوزه تا زمانی که شناسایی و تصحیح واژگان بصورت کارا بهبود نیابد، عملیاتی نخواهد شد. دیگر کاربردهایی که از این بهبود سودمند خواهند شد عبارتند از: ماشین ترجمه، ویراستاری ماشینی، ابزارهای ویراستاری متن، ابزارهای‌یادگیری زبان، تبدیل صوت به متن، آموزش با رایانه، تبدیل فکس به صوت و ...

۲-۱-۱-روش‌های غلط‌یابی متن

غلط‌یابی ماشینی از سال‌های نخست بکارگیری متن در رایانه مورد توجه پژوهشگران بوده است، بنابراین پژوهش‌های گسترده‌ای در این حوزه برای زبان‌های رایج مانند انگلیسی صورت پذیرفته است. متاسفانه زبان فارسی به دلیل مشکلاتی که دارد و اهم آن در قسمت جداگانه‌ای بررسی می‌شود، در زمینه پردازش متن و

[1] OCR (Optical Character Recognition)

غلط‌یابی مغفول واقع شده است. پژوهشگران این حوزه به دلیل مشکلات و کمبود منابع نیاز دارند اقدامات زیر بنایی و گسترده‌ای در این خصوص انجام پذیرد.

در حوزه‌ی غلط‌یاب ماشینی شناسایی خطا[1] و تصحیح خطا[2] معانی و عملکرد متمایزی دارند. الگوریتم‌ها و روش‌های کارآمدی برای تشخیص واژگان نادرست در متن که در یک فهرست یا دیکشنری موجود نیستند، پیشنهاد شده است، ولی تصحیح یک واژه غلط به مراتب مساله‌ی مشکل‌تری است. برای تصحیح واژه نه تنها تهیه فهرستی از واژه‌های کاندید جایگزینی واژه غلط و ارزش‌گذاری[3] آن فرآیند پرچالشی است، بلکه تهیه و دسترسی به قواعد و قوانین زبان مورد نظر نیز فرآیند مشکلی است.

پژوهش در خصوص غلط‌یاب و تصحیح واژه‌های نادرست را می‌توان به سه دسته‌ی کوچکتر تقسیم نمود:

تشخیص غیرواژه‌ها (کلمه‌هایی که در دیکشنری یا فهرست کلمه‌های درست موجود نیست): پژوهش درباره‌ی این حوزه از ۱۹۷۰ میلادی به مدت یک دهه در جریان بوده است.

تصحیح واژه‌ها بصورت منفرد[4]

تصحیح واژه‌ها مبتنی بر متن[1]

[1] Error Detection
[2] Error Correction
[3] Ranking
[4] Isolated word error correction

از سال ۱۹۷۰ الی ۱۹۸۰ پژوهش‌ها درباره‌ی شیوه‌های جستجو و تطبیق متن و تطبیق الگو در درون فهرست و یا دیکشنری در جریان بوده است. پژوهش درباره‌ی دسته‌ی دوم (تصحیح منفرد واژه‌ها) از ۱۹۶۰ تا کنون در جریان بوده است. پژوهش درباره‌ی تصحیح واژه‌ها مبتنی بر متن، از ۱۹۸۰ میلادی تا کنون در جریان است و ساخت مدل‌های خودکار پردازش زبان طبیعی با ساخت مدل سازی آماری زبان[۲] گره خورده است.

غالب روش‌های تصحیح خطا به روش تصحیح واژه‌ها بصورت منفرد تکیه دارد و تصحیح واژه را بدون در نظر گرفتن اطلاعات زبان‌شناسی و همچنین متنی که واژه در آن قرار گرفته است انجام می‌دهد. این نوع تصحیح خطا نمی‌تواند کسری از خطاها را شامل خطاهای آوایی[۳]، شناختی[۴]، گرامری و چاپی[۵] را شناسایی و تصحیح نماید. خطاهای آوایی معمولاً در استفاده نادرست از حروف هم‌آوا پدید می‌آید و خطای شناختی نیز در اثر عدم تسلط کاربر به زبان پدید می‌آید که معمولاً کاربر معنا و املای صحیح واژه را نمی‌داند. خطای تایپی زمانی رخ می‌دهد که کاربر املای صحیح لغت را می داند، ولی در زمان تایپ واژه دچار مشکل شده و واژه را غلط تایپ می‌کند. در بسیاری از موارد تشخیص و تصحیح خطا بدون در نظر گرفتن متن امکان‌پذیر نمی‌باشد. استفاده از متن واژگان در شناسایی و تصحیح خطا علاوه بر امکان شناسایی و تصحیح خطاهای معنایی[۶] در تصحیح لغات غیر واژه نیز کاربرد دارد [۱۱].

---

[1] Context dependent word correction
[2] Statistical Language Model
[3] Phonetic
[4] Cognitive
[5] Typographic
[6] Real-word error

با توجه به اینکه تمرکز این پژوهش بر غلط‌یابی و تصحیح غیرواژه‌ها است، در ادامه روش‌های غلط‌یابی و تصحیح غیرواژه‌ها مورد بررسی قرار می‌گیرد.

دو روش اصلی برای شناسایی غیرواژه‌ها مورد استفاده قرار گرفته است. این دو روش عبارتند از:

- تحلیل N-gram
- فهرست کنترل شده یا جستجو در دیکشنری[1]

تحلیل N-gram، عبارت از در نظر گرفتن زیر رشته‌هایی بطول n-کاراکتر از کلمه‌ها یا رشته‌ها است. معمولاً مقدار n برابر یک[2]، دو[3] و یا سه[4] است. به طور کلی در روش تحلیل N-gram، هر N-gram در رشته متن ورودی بررسی شده و موجود بودن و فرکانس آن دریک جدول (که بصورت دسته‌ای از انبوهی از داده‌ها تهیه شده) است، کنترل می‌شود. رشته‌هایی که در جدول موجود نباشند و یا براساس فرکانس موجود در جدول احتمال آن‌ها نادر تشخیص داده شود به عنوان غلط‌های احتمالی املایی علامت گذاری می‌شوند [۱۲]. روش تحلیل N-gram نیازمند حجم انبوهی از داده‌ی متنی برای تحلیل و ساخت جدول N-gram است. در دسترس بودن چنین حجمی از متن، یکی از چالش‌های این روش است. معمولاً سیستم‌های تشخیص متن[5] از روش N-gram استفاده می‌کنند. یکی از مشکلات

[1] Dictionary lookup
[2] Uni-gram
[3] Bi-gram
[4] Tri-gram
[5] Text recognition systems

تشخیص نوری نویسه‌ها[1] تشخیص حروف شبیه بهم مانند O و D است. برای چنین مواردی، استفاده از روش N-gram برای مشخص کردن نویسه‌ی درست بسیار مفید است زیرا N-gramهای نامحتمل را مشخص می‌نماید [۱۳].

هارمون [۱۴] گزارش داده است که ۴۲٪ از ترکیب‌های ممکن هیچ‌گاه در متن انتشارات انگلیسی دیده نشده‌اند و جایگزینی یک حرف از کلمه با یکی از حروف دیگر، در بیش از ۷۰٪ مواقع ترکیبی با احتمال صفر تولید می‌کند. با تکیه بر این یافته، اکثر سیستم‌های تشخیص نوری نویسه‌ها، با استفاده از این روش، ترکیب‌های نامحتمل و تشخیص شناسایی کلمه‌ها را شناسایی می‌نمایند.

جداول N-gram به صورت‌های گوناگون پیاده سازی می‌شوند، که ساده ترین حالت آن آرایه‌ی دودویی[2] Bi-gram است. در این روش یک آرایه دو بعدی که هر بعد آن به تعداد حروف زبان مورد نظر است (۲۶*۲۶ برای زبان انگلیسی) تشکیل می‌شود. در این جدول تمام ترکیب‌های دو حرفی ممکن شکل می‌گیرد. اگر هر ترکیب دو حرفی در دیکشنری و یا گنجینه لغت مورد استفاده برای تشکیل جدول حداقل یک بار دیده شود، مقدار این خانه از جدول "یک" و در غیر اینصورت "صفر" خواهد بود. به همین ترتیب، جدول tri-gram دودویی، سه بعدی خواهد بود. این نوع جداول محل وقوع N-gram در درون کلمه را مشخص نمی‌کنند،

[1] OCR
[2] Binary bigram array

بنابراین این نوع جداول غیرموضعی[1] نامیده می‌شوند. نوع دیگر جداول N-gram، جداول موضعی[2] است. در این حالت برای پوشش دقیق‌تر ساختار گنجینه لغت، مکان وقوع N-gram نیز ذخیره می‌شود. برای مثال اگر موقعیت I,J,K در یک جدول tri-gram دودویی یک باشد، نشان دهنده این است که حداقل یک کلمه در گنجینه لغت، که در محل I,J,K حروف L,M,N را داشته است مشاهده شده است. ذخیره ساختار گنجینه لغت نیاز به منابع بیشتر و حجم بیشتری از فضای ذخیره سازی دارد. بدین صورت هر کلمه بسادگی در جدول N-gram کنترل می‌شود. برای مقایسه کارآیی روش‌های غیرموضعی و موضعی پژوهش‌های بسیاری انجام شده است.

هنسون و همکارانش [۱۵] با بررسی روش موضعی و غیرموضعی گزارش دادند که روش موضعی با تشخیص ۹۸٪ خطاهایی که یک حرف با حرف دیگری جایگزین شده بود، بهترین نتیجه را بدست آورده است. برای تشـــــخیص غلط‌های املایی موریس و همکارش [۱۶] یک جدول tri-gram که بر اساس فرکانس کلمات مشاهده شده در همان مدرک کار می‌کرد، تشکیل دادند. سپس برای هر کلمه‌ی درون مدرک، شاخص غرابت[3] را محاسبه نمودند که تابعی از فرکانس‌های جدول بود. آن‌ها ثابت کردند که لغات دارای غلط املایی در بالای لیست قرار می‌گیرند و به عبارت دیگر غرابت بیشتری دارند.

[1] Non-positional
[2] positional
[3] peculiarity

در روش جستجو در دیکشنری، برای شناسایی غیرواژه‌ها، وجود هر واژه در متن ورودی دریک دیکشنری و یا به عبارت دیگر فهرستی از کلمات درست، کنترل می‌شود. اگر واژه در فهرست موجود نباشد، به عنوان واژه غلط علامت گذاری می‌شود. این روش بسادگی قابل استفاده است ولی زمان پاسخ سیستم یکی از چالش‌های مهم این روش است، بخصوص زمانی که اندازه‌ی دیکشنری بیشتر از چند صد واژه باشد. معمولاً در کاربردهایی مانند پردازش اسناد و بازیابی اطلاعات، تعداد واژه‌های دیکشنری از حدود ۲۵ هزار الی بیش از ۲۵۰ هزار واژه در نوسان است. سه راه حل برای غلبه بر زمان پاسخ دیکشنری استفاده شده است:

۱-استفاده از فرهنگ لغت یا الگوریتم‌های تطبیق الگو

۲-استفاده از روش‌های تقسیم‌بندی دیکشنری

۳-استفاده از تکنیک‌های زبان شناسی

مرسوم ترین روش برای ساخت دیکشنری‌ها، روش جدول درهم‌کرد[1] است. در این روش برای هر رشته ورودی، تابع درهم‌کرد محاسبه می‌شود و با کمک آن محل واژه در جدول درهم‌کرد مشخص می‌شود. ممکن است دو یا چند واژه در زمان ساخت جدول درهم‌کرد با هم تداخل داشته باشند، بنابراین باید مکانیزمی برای حل مشکل تداخل در نظر گرفته شود. اگر واژه‌ی ورودی در جدول درهم‌کرد مدخلی نداشته باشد، به عنوان واژه‌ی غلط علامت گذاری می‌شود.

[1] Hash table

توربا [۱۷]، مروری بر ساخت دیکشنری با کمک جدول درهم‌کرد انجام داده است. مزیت اصلی روش درهم‌کرد، امکان دسترسی تصادفی[1] به هر واژه در مقایسه با روش جستجوی ترتیبی[2] و حتی درخت‌های جستجو[3] است. عیب عمده‌ی این روش این است که نیاز به تابع درهم‌کرد هوشمند است که بتواند با حداقل تداخل جدول درهم‌کرد بزرگی را ساخته و جستجو کند. در برخی پژوهش‌های اخیر نیز برای صرفه جویی در فضای حافظه، خود واژه را در جدول درهم‌کرد ذخیره نمی‌کنند و بجای آن از یک بیت که نشان دهنده‌ی معتبر بودن واژه است، استفاده می‌شود.

برنامه‌ی spell دریونیکس، یکی از مثال‌های کاربرد جدول درهم‌کرد برای ساخت دیکشنری برای کنترل واژه‌ها است. در پژوهش‌های دیگری آهو و همکارش و کنات [۱۸], [۱۹] از درخت دودیی جستجو و ماشین متناهی برای کاهش زمان جستجوی دیکشنری استفاده کردند.

پیترسون [۲۰] از تقسیم‌بندی دیکشنری به سه سطح برای جستجو و عملیات غلط‌یابی استفاده کرده است. سطح اول در حافظه‌ی سریع[4] ذخیره می‌شود، این بخش شامل چند صد واژه‌ای است که بیشترین استفاده را در دیکشنری دارند و بیش از ۵۰٪ اوقات در متن ظاهر می‌شوند. سطح دوم در حافظه‌ی اصلی ذخیره می‌شود و شامل چند هزار واژه‌ای است که غالباً مورد استفاده قرار می‌گیرد (۴۵٪) و در نهایت سطح سوم در حافظه‌ی جانبی ذخیره می‌شود و شامل چندین هزار واژه است که البته به ندرت کاربرد دارند (۵٪).

[1] Random access
[2] Sequential search
[3] Search tree
[4] Cache memory

مشکلات حافظه موجب می‌شود که غلط‌یاب‌های اخیر، تمامی اشکال واژه را ذخیره نکنند (همانند زمان گذشته، حال، آینده فعل و اسامی جمع و ...) و فقط ریشه‌ی واژه‌ها را در دیکشنری ذخیره شود. حال اگر کلمه‌ای در دیکشنری پیدا نشود، یک تحلیلگر زبان شناسی بکار می‌افتد و کلمه را با بکار گیری حالت‌های مختلف کلمه از نظر زمان و پسوندها و پیشوندهای شناخته شده کنترل می‌کند، این محاسبات معمولاً زمان بیشتری از پردازشگر را طلب می‌کند. ولی در نظر گرفتن تمام حالات زبان شناسی که مردم در متن و گفتار استفاده می‌کنند، عملاً ممکن نیست و بخشی از آن که قوانین مدونّی برای آن وجود دارد را می‌توان کنترل کرد.

ساخت دیکشنری برای کاربرد غلط‌یابی واژه‌ها و یا شناسایی واژه‌ها باید با دقت بسیار برای دامنه‌ی مورد نظر تنظیم و تهیه شود. یک گنجینه لغت خیلی کوچک می‌تواند کاربر را با واژه‌های بسیاری که اشتباهاً غلط تشخیص داده شده است، مواجه نماید. همچنین یک گنجینه لغت بسیار بزرگ نیز می‌تواند کاربر را با واژه‌های صحیح دور از ذهن، که در زبان رایج نیست، مواجه نماید. این واژگان معمولاً فرکانس بسیار پایینی دارند، ولی رابطه‌ی میان فرکانس واژه‌ها و غلط‌یابی آن‌ها شفاف و واضح نیست. پیترسون [۲۱] در تحقیقاتش نشان داده است که حدود نیمی از خطاهای تک حرفی که بصورت تبدیل از حرفی به حرف دیگر رخ می‌دهند، در یک فهرست با ۳۵۰ هزار واژه، به واژه‌ی درست دیگری تبدیل می‌شوند. نرخ این نوع خطا با بزرگ شدن اندازه‌ی گنجینه لغت افزایش می‌یابد. بنابراین پیترسون توصیه کرده است که بهتر است حتی المقدور اندازه‌ی گنجینه لغت کوچک باشد. در این زمینه نظرات متفاوتی نیز مطرح شده است.

دامرائو و میز [۲۲]، این توصیه به چالش کشیده شده‌اند و یک گنجینه لغت از تمامی موضوع‌ها با بیش از ۲۲ میلیون واژه تشکیل داده‌اند. در این پژوهش مشخص شد که با افزایش اندازه‌ی گنجینه لغت، و افزایش فهرست مرتب بر اساس ارزش گذاری[1]، از ۵۰ هزار به ۶۰ هزار واژه، تشخیص ۱۳۴۸ واژه که اشتباهاً غلط اعلام شده بود، در مقابل فقط ۲۳ واژه غلط، که به اشتباه درست علامت‌گذاری شده بود، امکان پذیر گردید. با توجه به بهبود نتایج بدست آمده، آن‌ها پیشنهاد دادند که اندازه‌ی گنجینه لغت بزرگتر باشد.

دیکشنری‌ها نیز منبع مناسبی برای تهیه‌ی گنجینه‌های لغت نیستند، در پژوهش‌های انجام شده مشخص گردید حدود ۶۱٪ از واژه‌های دیکشنری[2] هیچ گاه در متن ۸ میلیون واژه‌ای نیویورک تایمز مشاهده نشده‌اند و حدود ۶۴٪ از واژه‌های متن در دیکشنری وجود ندارند [۲۳]. میتون [۲]، یک دیکشنری قابل استفاده برای کامپیوتر، با استفاده از دیکشنری پیشرفته آکسفورد با بیش از ۳۸۰۰۰ مدخل اصلی تهیه کرده است. در پژوهش دیگری کاشفی و همکارانش [۹] یک گنجینه لغت برای زبان فارسی تهیه کرده‌اند. اندازه‌ی این گنجینه لغت از ۷۰۰ هزار الی ۸۵۰ هزار واژه در نوسان بوده است. صادقی و همکارانش نیز کتابی با حدود ۳۳۰۰۰ مدخل از واژگان املایی زبان فارسی تهیه کرده‌اند [۸۶].

تشخیص محدوده‌ی واژه‌ها از چالش‌های پردازش زبان طبیعی است. این مشکل برای اکثر زبان‌ها وجود دارد و ابزارهایی برای آن ارائه شده است. شاید تشخیص محدوده‌ی کلمه‌ها بسیار راحت به نظر برسد ولی در

[1] Ranked list
[2] Merriam-webster Seventh collegiate

نظر گرفتن کاراکتر فاصله، کاراکتر بازگشت به سرخط و ... کافی نیست. این فرض برای زمانی که واژه‌ها بیش ازیک کلمه باشند و یا زمانی که خطای پیوستگی[1] اتفاق افتد، با مشکل مواجه می‌شود. برای مثال می‌توان به این موارد در زبان انگلیسی (inthebox, thereare, iam) و یا (سلامتبدن، ابرهایبارانزا، سیستمعامل) به زبان فارسی اشاره نمود. همچنین اضافه شدن کاراکتر فاصله نیز می‌تواند مشکل خطای انفصال[2] ایجاد کند، برای مثال می‌توان به (th ebook, val ue) و یا (سی ستم، مش خص) اشاره نمود. کوکیچ [۲۵]، گزارش داد ۱۵٪ تمامی خطاهای غیرواژه، دریک پیکره‌ی ۴۰ هزار واژه‌ای از این نوع دو خطای انفصال و پیوستگی (۲٪ خطای انفصال و ۱۳٪ خطای پیوستگی) بوده است.

میتون [۲] در پژوهشی دیگر مشاهده نمود که خطای پیوستگی و انفصال در اغلب موارد می‌تواند منجر به تولید یک واژه‌ی معتبر در گنجینه لغت شود. برای مثال واژه‌ی inform می‌تواند به دو واژه‌ی معتبر "in" و "form" تبدیل شود و یا "سلامتی" می‌تواند یک واژه‌ی معتبر "سلام" ویک واژه‌ی نامعتبر "تی" تبدیل شود. خطای پیوستگی و انفصال نشان می‌دهد که تشخیص محدوده‌ی واژه‌ها بسیار مهم است. معمولاً غلط‌یاب‌ها تشخیص محدوده‌ی واژه‌ها را به عنوان یک خطای جداگانه در نظر نمی‌گیرند. معمولاً در زمان تصحیح خطا اگر پیشنهادها با در نظر گرفتن یک آستانه از مقداری کمتر باشد، خطای پیوستگی و انفصال کنترل می‌شود.

[1] Run-on Error
[2] Split Error

۲-۱-۲-تصحیح خطای غیرواژه بصورت منفرد

برای برخی از کاربردها شناسایی خطاهای واژه‌ای ممکن است کافی باشد، ولی برای کاربردهایی نظیر غلط‌یاب‌ها، تشخیص متن و یا گفتار، تشخیص خطا به تنهایی کافی نیست، و باید مکانیزم‌هایی برای تصحیح خطا نیز داشته باشیم. زیرا کاربران انتظار دارند که غلط‌یاب‌ها، پس از شناسایی خطا، واژه‌های صحیح را نیز پیشنهاد دهند. برای پیشنهاد فهرست واژه‌های جایگزین واژه‌ی غلط، خصوصیات کاربردهای مختلف و برخی محدودیت‌ها موجب می‌شود این مکانیزم اختصاصاً برای هر کاربرد تنظیم شود. دو کاربرد عمده‌ی تصحیح واژه‌ها در ویرایش متن و شناسایی متن است، ولی کاربردهای دیگری را نیز برای تصحیح خطا در سیستم‌های برنامه‌نویسی، رابط کاربری خط فرمان، بازیابی اطلاعات و ... می‌توان برشمرد.

معمولاً معیارهایی که در طراحی سیستم تصحیح خطا در نظر گرفته می‌شود عبارت است از اندازه‌ی گنجینه لغت که از نظر دامنه پوشش و تعداد لغات برای کاربرد مورد نظر تنظیم می‌شود. معیار دیگر زمان پاسخ است که با توجه به نوع رابط کاربری و ارتباط سیستم با کاربر بصورت تراکنشی و برخط و یا آفلاین تنظیم می‌شود. الگوی خطا[1] نیز باید برای کاربرد مورد نظر در نظر گرفته شده و با توجه به آن تصحیح خطا انجام شود.

با توجه به اینکه این پژوهش در خصوص غلط‌یاب فارسی انجام شده است، تمامی معیارها برای این کاربرد تنظیم و بررسی شده است. الگوهای خطا شدیداً به کاربرد مورد استفاده وابسته است. به عنوان مثال در

[1] Error patterns

سامانه‌های ویراستاری متن، الگوهای خطا به صفحه کلید، محل قرار گرفتن کلیدها، ملیت کاربر و ... وابســـته است. انواع خطاهای غیرواژه به سه دسته عمده تقسیم می‌شوند [۶]:

خطاهای چاپی[1] (تایپ)

خطاهای شناختی[2]

خطاهای هم آوایی[3]

خطاهای چاپی (تایپ) زمانی رخ می‌دهد که کاربر املای صحیح واژه را می‌داند، ولی در زمان تایپ واژه دچار مشکل شده و واژه را غلط تایپ می‌کند. برای مثال می‌توان به تبدیل "صبحانه" به "ضبحانه" و یا "the" به "thw" اشاره نمود.

خطاهای شناختی به دلیل عدم آگاهی کاربر از هجی و یا املای صحیح واژه نشات می‌گیرد. برای مثال می‌توان به خطای تبدیل "غلطک" به "غلتک" اشاره نمود. خطاهای هم آوایی نیز زیر شاخه‌ای از خطاهای شناختی هستند و در این دسته از خطا، یک حرف با حرف هم صدا یا هم‌آوا جایگزین می‌شود. برای مثال می‌توان به جایگزینی "طلب" و "تلب" اشاره نمود.

[1] Typographic errors
[2] Cognitive errors
[3] Phonetic errors

بیشتر مطالعات در حوزه‌ی الگوهای خطا برای ساخت الگوی تصحیح خطا برای غلط‌های غیرواژه صورت پذیرفته است. امروزه با افزایش غلط‌یاب‌ها، معمولاً خطاهای غیرواژه در متن‌ها کمتر شده است ولی خطاهای معنایی و وابسته به متن با تصحیح خودکار واژه می‌تواند افزایش یابد. بنابراین نیاز است خطاهای معنایی نیز در غلط‌یاب‌ها لحاظ شود.

دامرائو [۱۱] در پژوهشی که، پایه‌ی بسیاری از سیستم‌های تصحیح خطا تا به امروز بوده است گزارش داد که بیش از ۸۰٪ از تمامی خطاها، توسط یکی از چهار عملگر زیر ساخته شده‌اند (مثال در جدول ۲-۱ آورده شده است):

درج[1]: درج یک حرف در میان، ابتدا و یا انتهای کلمه

حذف[2]: حذف یک حرف از میان، ابتدا و یا انتهای کلمه

جابجایی[3]: جابجایی یک حرف از کلمه با حرفی دیگر

جایگزینی[4]: جایگزینی یک حرف از کلمه با حرفی دیگر

[1] Insertion
[2] Deletion
[3] Transposition
[4] Substitution

جدول ۲- ۱-مثال الگوهای خطای غیر واژه

واژه صحیح	واژه غلط	الگوی خطا
سیب	سیبب	درج
سیب	سب	حذف
سیب	سیپ	جایگزینی
سیب	بیس	جابجایی

وجود بیش از ۸۰٪ خطاهای تکی[1] براساس الگوهای‌یاد شده بستگی به کاربرد دارد. پولاک و همکارانش [۲۶] گزارش دادند از بین ۵۰ هزار غیرواژه، فقط ۶٪ چند خطای همزمان[2] داشتند، به عبارت دیگر این ۶٪ از غیر واژه‌ها، چند خطا از الگوهای یاد شده را همزمان دریک واژه داشتند. همچنین میتون نیز [۲] دریافت که ۳۱٪ از حدود ۱۷۰ هزار غیر واژه در متن‌های دست نویس مقالات، شامل چندین خطای همزمان بودند. همچنین با مطالعه‌ی غیر واژه‌ها مشخص گردید که طول واژه‌ها در کلمات دارای خطا موثر است. معمولاً

[1] Single error
[2] Multi Error

واژه‌های دو حرفی بیشتر در فهرست غیرواژه‌ها مشاهده گردید. ولی متاسفانه تصحیح خطا در واژه‌هایی با طول کوتاه‌تر مشکل‌تر است، زیرا اطلاعات کمتری در اختیار تصحیح‌گر قرار می‌دهد.

قانون زیف[1] [۲۷] بیان می‌کند که کلمات با طول کمتر، بیشتر از کلمات با طول زیاد، در متن ظاهر می‌شوند. همچنین کلمات با فرکانس بالاتر (به عنوان مثال کلمات کوتاه)، بیشتر در کنار یک واژه با یک خطا قرار می‌گیرند، بنابراین تصحیح آن‌ها با کمک همسایه‌ها کمی مشکل‌تر است.

پولاک و همکارانش [۲۸] ، با بررسی حدود ۵۰ هزار غیرواژه، گزارش دادند واژه‌هایی با طول کم، در تصحیح خطا مشکل‌زا هستند، حتی اگر فرکانس تکرار کمی داشته باشند. با بررسی واژه‌ها مشخص شد در حالی که واژه‌هایی با طول ۴-۳ کاراکتر، تنها ۹.۲٪ از کل خطاها را تشکیل می‌دهند، این واژه‌ها ۴۲٪ از خطاهای تصحیح را شامل می‌شوند. در پژوهشی مشابه، کوکیچ [۲۹] با بررسی حدود ۲۰۰۰ خطا از انواع مختلف، گزارش داد حدود ۶۳٪ از خطاها، در کلماتی با طول دو، سه و یا چهار اتفاق می‌افتد.

محل وقوع خطا نیز مورد مطالعه بوده است. معمولاً بر این باور هستیم که تعداد کمی از خطاها در ابتدای کلمه اتفاق می‌افتد. پولاک و همکارش [۲۸] گزارش دادند تنها ۳.۳٪ خطاها از بررسی بیش از ۵۰ هزار خطای غیر واژه در ابتدای کلمه واقع شده‌اند. میتون [۲] نیز دریافت که ۷٪ از خطاهای غیرواژه در ابتدای واژه رخ داده‌اند.

[1] Zipf

با توجه به اینکه وقوع خطا در ابتدای واژه‌ها کمتر اتفاق می‌افتد، می‌توان از این مزیت برای تقسیم‌بندی و پارتیشن کردن گنجینه لغت بر اساس حرف ابتدایی کلمه استفاده کرد. بسیاری از سیستم‌های غلط‌یاب و مصحح خطا بر این اصل تاکید کرده‌اند. البته به طورکلی بین زمان پاسخ و دقت باید توازن را براساس کاربرد برقرار کرد.

محل قرار گرفتن کاراکترها روی صفحه کلید در الگوی ایجاد خطا، و به تبع آن در تصحیح خطا موثر است. برای انجام تحقیقات بر الگوی تایپ کاربر با صفحه کلید در سال ۱۹۸۳ یک شبیه‌ساز تایپ توسط گروه تحقیقاتی LNR تولید گردید، هدف این سامانه، شبیه سازی مدل تایپ کاربر بود. با استفاده از این سامانه عملکرد تایپ کاربران خبره آنالیز شده است. در این تحقیق مشخص گردید بیش از ۵۸٪ از خطاهای جایگزینی، بایک کاراکتر همسایه در صفحه کلید بوده است.

در پژوهشی دیگر کاشفی و همکارانش [۳] از یک معیار جدید برای محاسبه‌ی فاصله‌ی دو رشته براساس محل قرارگیری کاراکترها روی صفحه کلید استفاده کردند. این معیار که برای زبان فارسی نیز طراحی و محاسبه شده است و با در نظر گرفتن همجواری کاراکترها در صفحه کلید، فاصله دو رشته را محاسبه می‌کند. بنابراین از هم جواری کاراکترها در صفحه کلید می‌توان برای تولید جایگزین‌های کلمه‌ی غلط استفاده کرد.

پژوهش‌های انجام شده در این حوزه نشان می‌دهد فرکانس خطا در متن به عواملی از قبیل سایز پیکره، نحوه‌ی ساخت متن که ممکن است دست نویس، تایپی و یا بویسله‌ی تشخیص نوری نویسه‌ها باشد، بستگی

دارد. حتی تاریخ تولید متن نیز می‌تواند در فرکانس تولید بعضی واژه‌ها موثر باشد. پژوهش‌های انجام شده توسط شرکت LNR نشان داد که نرخ خطاهای املایی تایپی برای کاربران خبره حدود ۱٪ و برای کاربران مبتدی حدود ۳.۲٪ است. کوکیچ [۲۹] گزارش داد حدود ۶٪ از خطاهای املایی دریک پیکره ۴۰ هزار واژه‌ای غیرواژه بوده‌اند.

برکل و همکارانش [۳۰]، گزارش دادند، حدود ۳۸٪ از خطاهای املایی تولید شده، مربوط به خطاهای هم آوایی حروف می باشند. در پژوهشی دیگر میتون[۲] دریافت که حدود ۴۴٪ از خطاهای تولید شده در مقالات ۹۲۵ دانشجو، در استفاده از حروف هم آوا بوده است. این پژوهش‌ها نشان می‌دهد که هم آوایی نیز فاکتور موثری در تصحیح خطا است.

تصحیح خطا بصورت منفرد باید طی سه مرحله انجام شود:

شناسایی خطا

تولید فهرستی از واژه‌های کاندید برای تصحیح خطا

ارزش گذاری[1] فهرست واژه‌های کاندید و نمایش فهرست نهایی به کاربر

با بررسی روش‌های شناسایی خطا در پژوهش‌های فوق مشخص شد که روش‌های برپایه‌ی دیکشنری از جمله روش‌های کارا برای تشخیص خطا می باشد. برای ارزش‌گذاری[1] معمولاً شباهت رشته‌ای مابین رشته دارای

[1] Ranked list

غلط املایی و رشته صحیح (از دیکشنری) سنجیده می‌شود و یا اینکه شباهت بین دو رشته با روش‌های احتمالی[۲] تخمین زده می‌شود. همچنین از شبکه عصبی نیز برای ارزش گذاری فهرست واژه‌ها استفاده می‌شود.

به طور معمول روش‌های تصحیح خطا به ۶ دسته‌ی عمده تقسیم می‌شوند :

روش‌های برپایه‌ی کمترین فاصله‌ی ویرایشی[۳]

روش‌های برپایه‌ی شباهت کلید[۴]

روش‌های مبتنی بر قانون[۵]

روش‌های مبتنی بر N-gram

روش‌های احتمالی[۶]

روش‌های مبتنی بر شبکه عصبی

روش کمترین فاصله‌ی ویرایشی بیشترین کاربرد را در تحقیقات داشته است. این دسته از الگوریتم‌ها، مینیمم فاصله‌ی بین رشته‌ی دارای غلط املایی و رشته‌های درون دیکشـــنری را محاســـبه می‌کنند. نام کمترین فاصله‌ی ویرایشی توسط واگنر[۳۱] مطرح شد و به این صورت تعریف می‌شود:

---

[1] Rank
[2] Probabilistic methods
[3] Minimum edit distance
[4] Key
[5] Rule-Based methods
[6] Probabilistic

"تعداد کمترین عملگرهایی (درج-حذف-جایگزینی-جابجایی) که لازم است یک رشته‌ی غلط را به رشته‌ی درست تبدیل کند."

اولین الگوریتم بر پایه‌ی معیار کمترین فاصله‌ی ویرایشی توسط دامرائو [۱۱] تولید گردید و تقریباً در همان زمان، لونشتین [32] نیز یک الگوریتم مشابه ارائه نمود. بنابراین این معیار بنام هر دو[1] نام‌گذاری شده است. در ادامه واگنر و همکارانـــــش[۳۱] الگوریتم ارائه شده توسط لونشتین را جامعیت بخشیدند تا غلط‌های املایی چندگانه[2] را نیز پوشش دهد.

معمولاً تمامی الگوریتم‌های این حوزه معیار Damerau-Levenshtein را محاسبه می‌کنند. برخی از این الگوریتم‌ها کمترین فاصله‌ی ویرایشی را بصورت عدد صحیح[3] و برخی دیگر بصورت اعداد غیر صحیح و یا حتی با استفاده از همجواری صفحه کلید [۳] محاسبه می‌کنند. ورانیس [۳۳] از الگوریتم برنامه‌نویسی پویا برپایه‌ی کاراکترهای هم‌آوا برای تصحیح خطا استفاده کرده است. تاکید بر هم‌آوایی به دلیل اینکه این نوع خطا بسیار شایع است، روشی موثر و کارا در تصحیح خطا است.

درحالت کلی اگر m تعداد عناصر دیکشنری باشد، الگوریتم‌های تصحیح خطا برپایه‌ی کمترین فاصله‌ی ویرایشی، نیاز به m مقایسه بین واژه‌ی کاندید جایگزینی غلط و رشته‌های درون دیکشنری دارند. روش‌هایی نیز برای کاهش زمان جســـتجو در درون دیکشنری ارائه شده است. با توجه به این یافته که حذف از

[1] Damerau-Levenshtein Metric
[2] Multi-Error
[3] integer

عملـــگـــرهای شـــایع برای تولید خطای غیرواژه است، در این پژوهش [۳۴]، مور و همکارانش، دیکشنری را بر اساس طول واژه‌ها مرتب کردند و جستجو بر اساس واژه‌هایی با طول حداکثر یکی بیشتر از طول واژه‌ی غلط در دیکشنری انجام می‌شود و بدین ترتیب زمان جستجو کاهش می‌یابد.

کمترین فاصله‌ی ویرایشـــی معکوس[1]، توسط گارین [۳۵] مطرح شد. در روش کمترین فاصله‌ی ویرایشی معکوس، یک مجموعه از واژه‌های کاندید جایگزینی واژه‌ی غلط، با تغییر رشته‌ی غلط توسط چهار عملگر (حذف- درج- جایگزینی- جابجایی) تولید شده و رشته‌ی حاصل در دیکشنری کنترل می‌شود، در صورتی که نتیجه حاصل از بکارگیری عملگرها، رشـــته‌ای معتبر در درون دیکشـــــنری باشد، به مجموعه‌ی واژه‌های کاندید اضافه می‌شـــود. چرچ و گیل [۳۶] از روش کمترین فاصله‌ی ویرایشی معکوس برای ساخت فهرست واژه‌های کاندید جایگزینی واژه‌ی غلط استفاده کرده‌اند.

با توجه به مشخص بودن تعداد حروف در زبان مقصـــد (مثال: ۲۶ حرف در زبان انگلیسی)، اگر طول واژه‌ی غلط را n فرض کنیم، حداکثر (n+1)*26 درج، n حذف، 25n جایگزینی و n-1 جابجایی خواهیم داشت. در مجموع، 53n+25 رشته باید تولید و کنترل شود. این عدد در مقایسه با سایز دیکشنری مطمعناً بسیار ناچیز بوده و زمان ساخت فهرست واژه‌های کاندید را بهبود می‌بخشد.

[1] Reverse minimum edit distance

دان‌لاوری و همکارش [۳۷]، الگوریتمی برپایه‌ی یک دیکشنری با ساختار ماشین متناهی ارائه نمودند. ماشین متناهی تمام رشته‌های درون دیکشنری را تشخیص می‌دهد و از یک ساختار درختی بر اساس کمترین فاصله‌ی ویرایشی استفاده می‌کند که برگ‌های درخت کاندیدهای جایگزینی واژه‌ی مورد نظر هستند.

دامرائو [۱۱] گزارش داد که با استفاده از کمترین فاصله‌ی ویرایشی، ۹۵٪ از واژه‌های دارای فاصله‌ی ویرایشی یک (دارای یک خطا)، از یک مجموعه‌ی دارای ۹۶۴ خطا، شناسایی و تصحیح گردیدند. با درنظر گرفتن خطاهای چندگانه نیز ۸۴٪ از خطاها بطور موفقیت آمیز تصحیح شدند. در پژوهشی دیگر موث و همکارش [۳۸] گزارش دادند که ۹۷٪ از خطاها توسط الگوریتمی برپایه‌ی کمترین فاصله‌ی ویرایشی تصحیح شده‌اند.

روش دوم تصحیح خطا، روش شباهت برپایه‌ی کلید است. ایده‌ی این روش عبارت است از تطبیق هر رشته به یک کلید، به صورتی که رشته‌هایی با املای مشابه، دارای کلید یکسان یا شبیه باشند. بنابراین وقتی کلید برای یک واژه‌ی غلط املایی محاسبه می‌شود، یک اشاره‌گر به تمام واژگانی که دارای املای صحیح هستند و از نظر واژه‌ای نیز شبیه به واژه‌ی غلط هستند (فهرست کاندید جایگزینی واژه) تولید می‌شود. مزّیت این روش سرعت تولید فهرست کاندید جایگزینی واژه‌ی غلط است، در این روش دیگر نیازی به مقایسه با دیکشنری نمی‌باشد.

این روش در کدهای سامانه‌ی Soundex، در سال ۱۹۱۸ استفاده شده بود [۳۹]. این روش در سامانه‌ی رزرو خطوط هوایی نیز استفاده شده است که هر رشته را به یک کلید منطبق می‌کند. پولاک و همکارش [۲۶]، با استفاده از روش تطبیق کلید، ۵۰ هزار واژه‌ی دارای غلط املایی بررسی کردند. آن‌ها با استفاده از روش تطبیق کلید روشی برای تصحیح واژه‌های دارای یک خطا تولید کردند. در این پژوهش از یافته‌های روش تطبیق کلید استفاده شده است و ۵۰ هزار از واژه‌ها بصورت کلید برای علم شیمی مورد بررسی قرار گرفته است. با استفاده از روش شباهت کلید، الگوریتمی بنام Speed Cop استفاده شده و واژه‌های دارای خطای یگانه تصحیح شده‌اند. در این الگوریتم، تصحیح هر غلط املایی با تولید کلید و یافتن آن در شاخص مرتّب کلیدها انجام می‌شود. پس از یافتن کلید واژه‌ی غلط، با کمک کلیدهای مجاور، فهرست کلمات کاندید جایگزین واژه‌ی غلط استخراج می‌شود. دقت این روش در تصحیح خطا، در یک پیکره با حدود ۴۰ هزار واژه، به طور متوسط حدود ۹۴٪ گزارش شده است.

بیتزر و همکارانش [۴۰]، برای تصحیح خطا، برای هر واژه یک کلید تولید کردند. گنجینه لغت نیز براساس کلید مرتّب شده بود. شباهت کلید در این پژوهش براساس معیارهای شناختی مانند طول واژه، حرف اول واژه، حروف واژه، ترتیب حروف واژه و هجاهای واژه محاسبه شده است. هر معیار به صورت یک بیت در کلید تعریف شده بود. این ساختار اضافه کردن بیت‌های دیگر را برای معیارهای جدید براحتی ممکن می‌سازد. برای مشخص کردن شاخص هر واژه، کلید آن واژه محاسبه شده و با استفاده از تطبیق دودویی نزدیکترین تطبیق

برای آن کلید محاسبه شده و فهرست جایگزینی واژگان استخراج می‌گردد. این روش بروی یک دیکشنری با خطاهای معادل آزمایش گردیده بود و گزارش شد حدود ۹۵٪ از خطاها را می‌تواند تصحیح نماید.

در الگوریتم دیگری که توسط بوکست (۱۹۹۱) معرفی گردید و تشخیص کلمه[1] نامیده می‌شود، از تطبیق فازی برای تطبیق کلید استفاده شده است. در این الگوریتم با ورود رشته دارای غلط املایی، الگوریتم یک معیار شباهت که میانگین وزنی چهار فاکتور مستخرج از واژه است را استخراج می‌کند. دیکشنری واژه‌ها نیز بر اساس طول واژه‌ها و حرف شروع واژه تقسیم‌بندی[2] می‌شود و کلید آن‌ها محاسبه شده و مقایسه فازی انجام می‌شود. در این پژوهش گزارش شد که الگوریتم، علیرغم سادگی آن، بسیار کارا و سریع است و می‌تواند ۷۸٪ از خطاها را تصحیح نماید. این نتایج بدست آمده در آن زمان از غلط‌یاب‌های تجاری بهتر بوده است .

دسته ســـــوم از روش‌های تصحیح خطا، روش‌های مبتنی بر قانون[3] و یا برنامه‌هایی با الگوریتم‌های ابتکاری[4] هستند. این روش‌ها دانش غلط‌یابی و تصحیح خطا را به صورت قوانین پیاده سازی می‌کنند. برای بدست آوردن فهرست واژه‌های کاندید جایگزینی واژه‌ی غلط، قوانین بروی رشته‌ی غلط اعمال می‌شود و واژه‌های تولید شده جهت اعتبارسنجی در یک دیکشنری جستجو می‌شود. ارزش‌گذاری فهرست نیز براساس انتساب یک عدد برپایه‌ی اینکه احتمال تولید این خطا توسط قانون اعمال شده چقدر است، انجام می‌شود.

[1] Token recognition
[2] Partition
[3] Rule-Based methods
[4] heuristic

در دو پژوهش، یاناکوداکیس و همکارش [۴۱]، یک روش تصحیح خطا برپایه‌ی مجموعه‌ای از قوانین ارائه کردند. هدف آن‌ها ارائه‌ی یک غلط‌یاب عمومی بوده است، بنابراین الگوریتم و قوانین بروی مجموعه‌ای از ۵۳۴ خطا از یک دیکشنری با ۹۳۷۶۹ کلمه آزمایش گردید. با توجه به اینکه قوانین از طول کلمه استفاده می‌کردند، دیکشنری براساس طول کلمات و حرف اول واژه‌ها، تقسیم‌بندی[1] گردید. تولید فهرست کاندیداها نیز در دیکشنری تقسیم‌بندی شده به دنبال واژه‌هایی که یک یا دو خطا با واژه‌ی مورد نظر فاصله دارند انجام می‌شود. پس از تهیه‌ی فهرست، ارزش‌گذاری آن با استفاده از تخمین احتمال رخداد قوانین انجام می‌پذیرد. آن‌ها گزارش کردند که برپایه‌ی آزمون‌های انجام شده، واژه‌ی درست در ۷۵٪ از موقع در تقسیم‌بندی دیکشنری مشاهده می‌شود و در ۹۰٪ از مواقع واژه‌ی صحیح در جایگاه اول فهرست قرار دارد.

در پژوهشی دیگر مینز (۱۹۸۸)، یک سامانه‌ی خطایاب برپایه‌ی قوانین تولید کرده است [۴۲]. در این پژوهش با تکیه بر میزان کوتاه نوشته‌ها، مخفف‌ها و اصطلاحات فنی یک سامانه برای پردازش زبان طبیعی ایجاد شده است. این سامانه ابتدا یک سری از قوانین زبان‌شناسی را برای تصحیح خطا کنترل می‌کند، سپس کنترل می‌شود که ممکن است واژه خلاصه نویسی شده‌ی یک واژه دیگر باشد. در نهایت تمامی خطاهای تکی برای مشخص کردن کلمه‌ی درست کنترل می‌شود. همچنین کاراکتر فاصله نیز در نظر گرفته می‌شود و احتمال درج آن در نظر گرفته می‌شود.

[1] Partition

روش سوم روش‌های مبتنی بر N-gram هستند. در این روش N-gramهای حرفی شامل uni-gram، bi-gram و tri-gram هستند. این N-gramها در سیستم‌های تشخیص متن و غلط‌یاب‌ها استفاده می‌شوند. کاربرد N-gram در تصحیح خطا، تسهیل دسترسی به دیکشنری برای تهیه‌ی فهرست واژه‌های جایگزین واژه‌ی غلط است. همچنین از تشابه لغوی آن‌ها برای محاسبه‌ی شباهت رشته‌ها نیز استفاده می‌شود. از N-gram برای پیاده سازی واژه‌ها به عنوان برداری از ویژگی‌های لغوی استفاده شده است و در این فضای برداری برای ارزش گذاری فهرست از معیارهای فاصله میان بردارها استفاده شده است. برخی از سیستم‌های غلط‌یاب که بر اساس روش N-gram کار می‌کنند، فازهای شناسایی خطا، بازیابی فهرست کاندید و ارزش گذاری فهرست را در سه مرحله جداگانه انجام می‌دهند، اما برخی دیگر هر سه مرحله را به یکباره به انجام می‌رسانند.

رایزمن و هنسون [۴۳]، شرح کاملی از استفاده‌ی N-gram در تصحیح خطا برای کاربرد شناسایی نوری نویسه‌ها ارائه کرده‌اند. در این روش پس از تقسیم‌بندی دیکشنری بر اساس طول کاراکتر واژه‌ها، N-gram دودویی مکان‌دار برای هر قسمت از دیکشنری تشکیل می‌شود. این محاسبه موجب می‌شود که پاسخ به این سوال که "آیا واژه‌ای که با حرف a در محل i و حرف b در محل j در این قسمت از دیکشنری موجود است‌یا خیر؟" امکان‌پذیر شود. در کاربرد تشخیص نوری نویسه‌ها براحتی می‌توان کنترل کرد که تمام

N-gramهای رشته‌ی خروجی دارای مقدار یک باشند. اگر رشته خروجی دارای یک خطای یگانه باشد، یک عدد صفر در N-gramهای آن مشاهده می‌شود و ماتریس N-gram نیز مکان خطا را نشان می‌دهد.

فهرست کاندیداهای جایگزینی واژه‌ی غلط با استفاده از سطر و ستون‌های N-gramهای دارای مقدار صفر در ماتریس بدست می‌آید. در این پژوهش روش N-garm بروی داده‌های آزمون با واژه‌هایی بطول ۶ کاراکتر آزمایش گردید و مشخص گردید روش tri-gram مکان‌دار از روش‌های دیگر N-gram، بهتر عمل می‌کند و می‌تواند ۹۸.۶٪ از خطاها را تشخیص داده و ۶۲.۴٪ آن‌ها را تصحیح نماید. مزیت این روش نسبت به روش جستجو در دیکشنری[1]، این است که نیاز به تعداد زیادی مقایسه در دیکشنری ندارد.

آنجل و همکارانش [۴۴]، از tri-gram برای تصحیح خطای املائی استفاده کردند. این روش شامل محاسبه‌ی معیار شباهت بر اساس tri-gramهای بدون مکان مشترک میان رشته‌ی غلط و واژه‌های درون دیکشنری است. معیار شباهت با یک تابع بسیار ساده بصورت فرمول ۲-۳۹ محاسبه می‌شود. در این فرمول، c تعداد tri-gramهای مشترک بین واژه‌ی دیکشنری و رشته‌ی غلط است و n و m طول دو رشته را نشان می‌دهد. در حقیقت این فرمول شباهت را بر اساس فرمول Dice پیاده سازی کرده است.

[1] Dictionary lookup

$$\text{Similarity} = \frac{2.\,c}{n + m} \qquad (۲-۱)$$

با استفاده از tri-gramها نیز یک شاخص معکوس به دیکشنری تولید گردیده است و با استفاده از این شاخص می‌توان واژه‌هایی از دیکشنری که حداقل‌یک tri-gram مشترک با واژه‌ی غلط املایی دارند، بازیابی نمود. در نهایت با کمک تابع شباهت (فرمول ۲-۳۹)، برای این مجموعه شباهت محاسبه می‌شود. این روش بروی مجموعه‌ی آزمون، با ۱۵۴۴ رشته‌ی غلط و دیکشنری با ۶۴۶۳۶ واژه آزمایش گردید و دقت حدود ۷۶٪ را بدست آورده است.

مشکل ضریب شباهت Dice این بود که وقتی رشته‌ی غلط کلاً در درون‌یک واژه‌ی درست مشاهده می‌گردید، خطا ایجاد می شد. به عنوان مثال اگر واژه‌ی نادرست concider را در نظر بگیریم، تابع Dice مقدار شباهت ۰.۷۱ و ۰.۷۰ را به واژه‌های cider و consider انتساب می‌دهد. برای رفع این مشکل، تابع Dice بصورت فرمول ۲-۴۰ تغییر داده شد. با آزمایش مشخص گردید این تابع مشکل تابع قبلی را ندارد و حتی بروی خطاهای چندگانه نیز تاثیر منفی ندارد.

$$\text{Similarity} = \frac{c}{\max(m, n)} \qquad (۲-۲)$$

در پژوهشی دیگر کوهنن با استفاده از tri-gramها [۴۵]، برای کاربردهای ویرایش متن و بازیابی اطلاعات استفاده کرده است. در این پژوهش بایک نوآوری، از N-gramها برای پیاده‌سازی واژه‌ها بصورت بردار

استفاده گردید و مقایسه‌ها در فضای برداری انجام پذیرفت. این روش هر سه مرحله‌ی شناسایی، تهیه فهرست و ارزش‌گذاری فهرست را در یک مرحله انجام می‌دهد [۴۴], [۴۵].

کوکیچ [۲۵] روش برداری بر پایه‌ی N-gram را برای یک مجموعه آزمون با ۱۷۰ خطای‌یگانه و چندگانه مورد آزمایش قرار داده است و مشاهده شد که دقت ۵۴٪ برای ضرب داخلی[1]، ۶۸٪ برای روش فاصله‌ی همینگ و ۷۵٪ برای شباهت کسینوسی بردارها در تصحیح خطا بدست آمده است.

روش ماتریس همبستگی حافظه[2] (CMM)، بسیار شبیه به روش فضای برداری است. در این روش گنجینه لغت (دارای m واژه) به‌یک ماتریس n در m، (n معیار لغوی) تبدیل می‌کند. بنابراین در این حالت، بردارهای بطول n از معیارهای لغوی برای هر واژه تشکیل می‌شود. در فرآیند تصحیح خطا، بردار حاصل از رشته‌ی غلط، در ماتریس همبستگی حافظه ضرب می‌شود، حاصل ضرب یک بردار بطول m است که عضو iام آن واژه‌ی iام را در گنجینه لغت مدل می‌کند. عنصری که شـــباهت بیشـــتری دارد، همبستگی بیشتری با رشته‌ی غلط دارد، بنابراین کاندیدای جایگزینی واژه‌ی غلط است.

چرکاسکی و همکارانش [۴۶]، مقایســـــه‌ای بر اسـاس بردارها ، بین bi-gram و tri-gram انجام دادند. برای انجام مقایسه، دو مجموعه آزمون از خطاهای یگانه بصورت تصادفی تشکیل شده است. یکی از مجموعه‌ها واژه‌های با طول متوسط (۳ تا ۵ کاراکتر) و دیگری لغات بلند (۱۰ تا ۱۲ کاراکتر) را مدل می‌کند.

[1] Dot product
[2] Correlation memory matrix

مجموعه‌ها دارای ۵۰۰ الی ۱۱۰۰۰ واژه بودند. درنهایت دقت بسیار خوب ۹۰٪ برای تصحیح خطا برای روش tri-gram گزارش شد.

روش دیگری که برای مدل سازی دریک فضا براساس معیارها استفاده شده است، روش تجزیه‌ی مقادیر منفرد[1] (SVD) است. این روش توسط کوکیچ [29] استفاده شده است. روش تجزیه‌ی مقادیر منفرد می‌تواند برای تجزیه‌ی ماتریس یک گنجینه لغت، به حاصلضرب سه ماتریس استفاده شود. ماتریس اول پیاده سازی n-gram حروف منفرد به عنوان برداری از معیارها برای هرواژه است، ماتریس دوم که یک ماتریس قطری است، مجموعه‌ای از مقادیر منفرد را نشان می‌دهد و در نهایت هر عنصر ماتریس سوم، گنجینه لغت را بصورت برداری از معیارها مدل می‌کند. هدف از این تجزیه این است که معیارهای مهم برای تشخیص ارتباطات شباهت واژه‌ها در فضای برداری مشخص شود. به عبارت دیگر با حذف نویز داده‌ها، ارتباط میان داده‌ها بهتر مشخص می‌شود.

از روش تجزیه‌ی مقادیر منفرد در کاربرد غلط‌یابی نیز استفاده شده است [۴۷]. همانند روش استفاده شده در بازیابی اطلاعات، برای کاربرد تصحیح خطا، یک ماتریس که در آن واژه‌ها بصورت بردارهایی از bi-gram و یا uni-gram مدل شده‌اند، تشکیل می‌شود. این ماتریس به سه جزء تشکیل دهنده تجزیه می‌شود. یک واژه‌ی غلط به وسیله‌ی حاصل‌ضرب مجموع بردارهای n-gram هر کاراکتر منفرد در رشته‌ی غلط (که در ماتریس اول پیاده سازی شده است) در ماتریس مقادیر منفرد که وزن‌ها را نشان می‌دهد، تصحیح می‌شود. بردار حاصل، محل واژه‌ی غلط را در فضای n بعدی نشان می‌دهد. هر معیار استاندارد فاصله

[1] Singular value decomposition

(مانند ضرب یا کسینوس زاویه بین دو بردار) می‌تواند برای مشخص کردن فاصله‌ی بردار واژه‌های درست (ماتریس سوم) استفاده شود تا بتوان کاندیداهای جایگزینی را ارزش گذاری نمود. در پژوهش انجام شده توسط کوکیچ (۱۹۹۰)، دقت تصحیح ۷۶٪ الی ۸۱٪ برای مجموعه‌ای از ۵۲۱ واژه بدست آمده است [۲۹]. ولی برای مجموعه‌های بزرگتر بهبود مهم و با ارزشی بدست نیامده است.

بایکل [۴۸]، از یک روش ترکیبی بر اساس uni-gram و بردارهایی برپایه‌ی معیارهای ابتکاری استفاده کرده است. کاربرد مورد استفاده در یک پایگاه داده شامل نام کارمندان به عنوان شاخص بوده است. در این پایگاه حدود ۱۰۰۰ نام کارمند ذخیره شده بود. در این روش نام صحیح هر کارمند به عنوان یک بردار با روش uni-gram پیاده‌سازی شده است. مقدار برداری هر uni-gram اگر حرف در نام مشاهده نشود صفر است. مقدار عددی هر حرف با توجه به فرکانس آنها در پایگاه داده‌ی نام‌ها از قبل مشخص شده بود. در این حالت حروفی که فرکانس پایینی داشتند، وزن بالایی می‌گرفتند زیرا فرض بر این است که حروفی که کمتر مشاهده شده‌اند، در جستجو وجه تمایز بالاتری دارند و ارزشمندتر هستند. نام‌هایی که غلط وارد سیستم شده‌اند نیز بصورت بردارهای uni-gram مدل می‌شوند. فرض می‌شود که حرف اول نام صحیح است. براساس آن بخشی از گنجینه لغت جستجو می‌شود. با استفاده از ضرب داخلی بردارها، شباهت میان رشته ورودی و آن بخش از گنجینه لغت محاسبه می‌شود. محاسبه شباهت در ۹۵٪ از مواقع برای تصحیح خطا موثر گزارش شده است و توانسته است نام صحیح را استخراج نماید.

استفاده از N-gram در کاربردهایی مانند تصحیح غلط‌های املایی و شناسایی متن، پژوهش‌ها را به سمت روش‌های احتمالی[1] سوق داده است. در روش احتمالی، احتمال انتقال[2] و احتمال اشتباه[3] مورد استفاده قرار می‌گیرد. احتمال انتقال، احتمال اینکه یک حرف یا یک رشته، با حرف دیگری ادامه یابد را نشان می‌دهد. احتمال انتقال وابسته به زبان است و بعضی اوقات با این فرض که زبان یک منبع مارکوف است، این احتمال بر اساس احتمال‌های مارکوف محاسبه می‌شود. در روش احتمالی، تخمین احتمالات با جمع آوری N-gramها بروی یک مجموعه‌ی بزرگ از متن صورت می‌پذیرد. مدل احتمال‌های اشتباه برای یک کاربرد و دستگاه خاص با استفاده از مثال‌های غلط املایی که توسط آن کاربرد، تولید شده است، انجام می‌شود. این فرآیند برای تصحیح خطا فاز یادگیری نامیده می‌شود. به عنوان مثال، برای زبان انگلیسی می‌توان یک بردار با ۲۶ عنصر (به تعداد حروف انگلیسی) تولید کرد که میزان شباهت برای هر حرف را نشان دهد.

احتمال‌های اشتباه براساس خطاهای انسانی، احتمال‌های خطا نامیده می‌شود. این احتمال‌ها از مجموعه‌های بزرگی از متن‌های تولید شده توسط انسان استخراج می‌شود. تحقیقات انجام شده درباره‌ی کاربرد این روش در خصوص تشخیص متن نشان داد، که استفاده از این احتمال‌ها نمی‌تواند به دقت قابل قبولی در تصحیح خطا برسد. ولی ترکیب این احتمالات با روش‌های مبتنی بر دیکشنری، می‌تواند به نتایج بهتری در تصحیح خطا برسد.

[1] Probabilistic methods
[2] Transition probability
[3] Confusion probability

بلدسو و همکارانش در کتابشان [۴۹] صفحات ۲۳۲-۲۲۵، در استفاده از تخمین شباهت برای کاربرد تشخیص متن پیشگام شدند. در این روش عملیات شناسایی و تصحیح در دو مرحله انجام شده است. در مرحله‌ی اول با استفاده از بردارهایی بطول ۲۶ (به تعداد حروف)، شباهت هر حرف بطور مجزا تخمین زده می‌شود. در ادامه، با کمک دیکشنری، ترکیبی از حروف شناسایی شده به صورتی انتخاب می‌شود که احتمال ماکزیمم شود و کلمه‌ی بدست آمده معتبر باشد. به عبارت دیگر با تاکید بر اطلاعات دیکشنری، کلمه شناسایی می‌شود.

در این روش با کمک قانون بیز[1]، احتمال پسین برای هر کلمه با کمک دیکشنری و احتمال‌های شباهت حروف محاسبه می‌شود. اگر X نماینده‌ی یک کلمه در دیکشنری و Y نماینده‌ی خروجی سامانه تشخیص نوری نویسه‌ها باشد، با کمک قانون بیز می‌توان فرمول ۲-۴۱ را بدست آورد.

$$P(X|Y) = \frac{P(Y|X).P(X)}{P(Y)} \qquad (۳-۲)$$

$$G(Y|X) = \log P(Y|X) + \log P(X) \qquad (۴-۲)$$

در این فرمول، P(X|Y) احتمال اینکه واژه معتبر باشد را نشان می‌دهد، احتمال شرطی P(Y|X)، احتمال مشاهده‌ی Y وقتی X یک کلمه معتبر است، را نشان می‌دهد. P(X) و P(Y) نیز احتمال‌های ناوابسته‌ی کلمه‌های X وY را نشان می‌دهد. برای بدست آوردن محتمل ترین کلمه‌ی دیکشنری، باید تابع ۲-۴۲ ماکزیمم شود. در

[1] Bayes

فرمول ۲-۷، P(X) احتمال‌های uni-gram برای کلمه‌ی X را نشان می‌دهد. احتمال پسین برای هر کلمه‌ی دیکشنری با کمک احتمال‌های شباهت برای هر کاراکتر منفرد با استفاده از فرمول ۲-۴۳ محاسبه می‌شود.

$$\log P(Y|X) = \sum_{i=1}^{l=n} \log P(Y_l|X_i) \qquad (۵ -۲)$$

در این معادله، n طول واژه و i شاخصی است که هر کاراکتر را در کلمه نشان می‌دهد. برای مثال فرض می‌کنیم خروجی تشخیص نوری نویسه‌ها واژه‌ی "Doq" باشد و کلمه‌ی دیکشنری "Dog" باشد، آنگاه خواهیم داشت:

$$G(Doq|Dog) \qquad -۲)$$
$$= \log P(D|D) + \log P(O|O) + \log P(q|g) \qquad (۶$$
$$+ \log P(Dog)$$

در این روش تخمین شباهت برای هر کاراکتر در رشته‌ی خروجی، با استفاده از سامانه تشخیص نوری نویسه‌ها پشتیبانی می‌شود و با استفاده از فرکانس uni-gram، واژه‌های نادیده گرفته شده، کلمه‌ای که بیشترین احتمال را دارد انتخاب می‌شود.

کاهن و همکارانش [۵۰]، با استفاده از ترکیب احتمال‌های اشتباه با روش جستجو در دیکشنری[1]، روشی برای تشخیص نوری نویسه‌ها ابداع کردند. وقتی کلمه‌ی خروجی تشخیص نوری نویسه‌ها توسط غلط‌یاب رد می‌شود، کلمه‌های جایگزین بر پایه‌ی احتمال اشتباه‌ها که در فاز یادگیری بدست آمده است، تولید می‌شود. این پروسه تکرار می‌شود تا کلمه‌ی معتبری در دیکشنری مشاهده شود و یا اینکه آستانه توقف حاصل شود. در نهایت دقت ۹۷٪ برای تصحیح واژه‌های غلط بدست آمده است.

اطلاعات احتمالی در کاربرد تصحیح خطا نیز بصورت کاربردی و کارا استفاده شده است. اوشیکا و همکارانش [۵۱]، با استفاده از مدل مخفی مارکوف[2] (HMM)، یک سامانه برای تشخیص نام خانوادگی در پنج زبان با استفاده از اطلاعات قومی و نژادی و مثال‌هایی از نام‌های خانوادگی تولید کردند. از مدل مخفی مارکوف برای دسته‌بندی[3] نام خانوادگی قبل از بکارگیری وابستگی‌های مختص زبان استفاده شده است. در این پژوهش گزارش شد که با بکارگیری مدل مخفی مارکوف، دقّت برای تشخیص نام خانوادگی از ۶۹٪ به ۸۸٪ ارتقا یافته است. کارآیی بهتر در تصحیح خطا برای سامانه‌هایی که از ترکیب اطلاعات احتمالی (روش پایین به بالا) با اطلاعات دیکشنری (روش بالا به پایین) استفاده می‌کنند، گزارش شده است.

سین‌ها [۵۲]، از روش ترکیبی با استفاده از احتمالات، اطلاعات دیکشنری و روش‌های ابتکاری استفاده کرده است. هدف این پژوهش مقابله با لغات خارج از دیکشنری (واژه‌هایی که در متن مشاهده می‌شوند ولی در

[1] Dictionary lookup
[2] Hidden Markov Model
[3] Classification

دیکشنری موجود نیستند) است. بنابراین او بخشی از دیکشنری را که پرکاربرد ترین کلمات (۱۰۰۰۰ فرکانس برتر) را دارد، استفاده کرده است. الگوریتم شامل دو مرحله بود: ابتدا واژه‌هایی تصحیح می‌شد که احتمال اشتباه بودن یک واژه‌ی صحیح در دیکشنری را مشخص می‌کرد. سپس یک الگوریتم جستجوی توسعه یافته بکار گرفته می‌شد که واژه‌ی صحیحی که در دیکشنری موجود نیست را پیدا کند. پس از جستجوی دیکشنری، با استفاده از روش‌های ابتکاری، فهرست واژه‌ها ارزش گذاری شده است. این روش بروی مجموعه‌ای با بیش از ۵۰۰۰ واژه آزمایش شده و کارآیی میانگین حدود ۹۸٪ بدست آورده است.

جونز و همکارانش [۵۳]، یک روش پس‌پردازش برای شناسایی نوری نویسه‌ها را پیاده سازی کردند. با استفاده از قوانین بیزین در تخمین واژه‌ها، یک منبع دانش تهیه گردید. برای مرحله‌ی یادگیری آن‌ها مدل احتمالی را برای سیستم شناسایی نوری نویســـــه‌ها، و فایل مورد پردازش تهیه کردند. ساخت مدل آماری در این پژوهش با استفاده از N-gramها برای حروف و واژه‌ها انجام شد. تصحیح واژه‌ها در این پژوهش در سه مرحله انجام شده است:

ابتدا فهرستی از واژه‌های جایگزین براساس احتمال‌های اشتباه و یک دیکشنری تولید می‌شود.

ترکیب کلمات کنترل می‌شود تا از خطای گسستگی جلوگیری شود.

براساس دیاگرام آماری واژه‌ها، فهرست مجدداً ارزش گذاری می‌شود.

در آزمایش‌های انجام شده بروی نرم افزارهای کاربردی دقت ۸۹.۴٪ برای تصحیح خطا بدست آمده است.

در پژوهش‌های اخیر نیز از روش‌های احتمالی برای تصحیح خطا اســـتفاده شده است. کاشیاپ و همکارش [۵۴]، از روش احتمالی برای بهبود کارآیی تصحیح خطای واژه‌های کوتاه (کمتر از ۶ حرف) استفاده کردند. با بررسی مشخص شد که اغلب غلط‌های تایپی بایک حرف نادرست که همسایه حرف صحیح در صفحه کلید هستند، جایگزین شده‌اند. سپس احتمال‌های جایگزینی محاسبه گردید و علاوه بر این، احتمال درج و حذف یک کاراکتر در درون کلمه محاسبه شد. با استفاده از یک الگوریتم بازگشتی و با در نظر گرفتن عملگرهای درج، خذف و جایگزینی، رشته‌ی غلط به عنوان ورودی دریافت گردیده و در دیکشنری با واژه‌های صحیح مقایسه و برای هر واژه ارزش گذاری در مقایسه با واژه‌ی غلط انجام شده است. در نهایت بروی مجموعه‌ی آزمون این روش دقتی بین ۳۰٪ الی ۹۲٪ کسب کرده است. دقت به دست آمده به طول کلمه و تعداد خطا در هر کلمه وابسته است.

در دو پژوهش مستقل کرنیگان و گیل و همکارش [۵۵], [۵۶]، الگوریتمی برای تصحیح واژه‌های دارای خطای تکی ارائه کردند. آن‌ها پایگاهی از خطاهای واقعی از بین ۴۴ میلیون واژه از متون یک پایگاه خبری[1]، استخراج کردند و با استفاده از این داده‌ها ماتریس احتمالی اشتباه را برای عملگرهای درج، حذف، جایگزینی و

[1] Associated Press

جابجایی ایجاد کردند. سپس با استفاده از روش کمترین فاصله‌ی ویرایشی معکوس، فهرست کاندیداها تولید شد و با استفاده از محاسبات بیزین ارزش گذاری گردید. دقت این روش ۸۷٪ در تصحیح خطا گزارش شده است.

تروی [۵۷]، ترکیبی از روش‌های احتمالی را با روش فاصله‌ی بردارها استفاده کرده است. او در این پژوهش، از معیار کسینوسی بردارها و احتمال‌های uni-gram واژه‌ها استفاده کرده است. از معیار کسینوسی بردارها برای تولید فهرست کاندیداها استفاده شده است. با استفاده از تکنیک برنامه نویسی پویا و با بکارگیری احتمال‌های uni-gram، فهرست کاندیداها ارزش گذاری شده و می‌تواند خطاهای تکی و چندگانه را تصحیح نماید. استفاده از روش احتمالی در تصحیح واژه‌ها دقت معیار کسینوسی بردارها را از ۷۵٪ به ۷۸٪ ارتقا داده است.

روش ششم که برای تصحیح خطا استفاده شده است، شبکه عصبی است. با توجه به اینکه شبکه عصبی برای کاربردهایی که با داده‌های نویزی کار می‌کند بسیار مناسب است، از این روش در تصحیح خطا نیز استفاده شده است. آموزش شبکه عصبی با داده‌های واقعی برای الگوهای خاص خطا انجام می‌شود و در این صورت نتایج خوبی برای الگوهای آموزش داده شده بدست می‌آید.

الگوریتم انتشار به عقب[1]، یکی از پرکاربردترین روش‌های آموزش شبکه عصبی است. معمولاً شبکه‌های عصبی از سه (نوع) لایه تشکیل شده‌اند: یک لایه ورودی، یک لایه‌ی میانی (لایه مخفی) و یک لایه خروجی. هر

[1] Back-Propagation

گره در لایه‌ی ورودی بایک اتصال وزن‌دار به همه گره‌های دیگر در لایه‌ی مخفی متصل می‌شود. به همین ترتیب گره‌های لایه‌ی مخفی نیز به همه گره‌های لایه‌ی خروجی متصل می‌شوند. بر اساس الگوها، گره‌های ورودی/خروجی خاموش یا روشن می‌شوند. معمولاً یک نشان دهنده‌ی روشن بودن و صفر نشان دهنده‌ی خاموش بودن گره است. از اعداد حقیقی نیز می‌توان برای وزن گره‌ها استفاده کرد و گره‌ای را در حالت نیمه روشن و یا کاملاً روشن قرار داد. پردازش الگوریتم آموزش انتشار به عقب عبارتست از اعمال الگو به گره‌های ورودی، اعمال نتیجه الگو به گره‌های وزن دار لایه‌ی مخفی و محاسبه‌ی الگوی مخفی و در نهایت الگوی بدست آمده به گره‌های خروجی اعمال می‌شود. وزن‌ها نیز شدت اتصال بین گره‌ها را نشان می‌دهد. این اتصال‌ها همانند مقاومت‌ها در مدار الکتریکی عمل می‌کنند و میزان شارژ شدن گره بعدی توسط گره قبلی را تعیین می‌کنند. معمولاً میزان فعالیت هر گره، مجموع میزان فعالیتهای گره‌های سطح پایین‌تری است که به آن گره منتهی می‌شود. این مجموع در شدت اتصال‌ها (وزن اتصال‌ها) ضرب می‌شود. این مجموع قاعدتاً با یک آستانه به صورت دودویی (صفر/یک) تفسیر شده و یا به یک عدد حقیقی بین ۰.۱ الی ۰.۹ تبدیل می‌شود.

الگوریتم انتشار به عقب با استفاده از مجموعه‌ی وزن‌ها که از مثال‌های داده شده بدست آمده، امکان تولید نتایج درست و یا نزدیک به درست را برای الگوی خروجی فراهم می‌سازد. معمولاً در ابتدا وزن‌ها به عدد کوچکی نزدیک صفر به صورت تصادفی مقداردهی اولیه می‌شود. سپس برای هر زوج مثال یادگیری که شامل ورودی و خروجی است، الگوی ورودی به گره‌های ورودی اعمال می‌شود و با اعمال نتایج به لایه‌های بعدی

خروجی بدست می‌آید. خروجی بدست آمده با خروجی دلخواه مقایسه شده و اختلاف آن (خطا) برای گره‌های خروجی بدست می‌آید. با توجه به خطای بدست آمده از خروجی واقعی، با انتشار به عقب مجدداً وزن اتصال‌ها تنظیم می‌شود. این عملیات برای مثال‌ها تکرار می‌شود تا وزن‌های شبکه همگرا شود.

در سامانه غلط‌یاب و تصحیح خطا، می‌توان واژه‌ی غلط را به صورت بردار N-gram دودویی پیاده سازی، و به ورودی شبکه عصبی اعمال کرد. خروجی می‌تواند برداری از m عنصر باشد، اگر m تعداد عناصر گنجینه لغت باشد، فقط گره‌هایی روشن می‌شود که به کلمات صحیح وابسته به کلمه‌ی غلط اشاره دارد. این نوع شبکه عصبی گاهاً یک طبقه‌بند[1]یک به m نیز نامیده می‌شود، زیرا هدف شبکه عصبی این است که گره‌های وابسته به واژه‌ی غلط را روشن و بقیه گره‌ها را خاموش نماید. در نهایت مقدار گره‌های خروجی شباهت میان واژه‌ی غلط و واژه‌های صحیح درون گنجینه لغت را نشان می‌دهد.

در پژوهش‌ها شبکه عصبی برای تصحیح خطا در کاربرد تشخیص نوری نویسه‌ها استفاده شده است. بور [۵۸]، ازیک روش دو مرحله‌ای برای تصحیح خطا استفاده کرده است. در مرحله‌ی اول ازیک شبکه عصبی با ۲۶ خروجی به تعداد حروف انگلیسی و ۱۳ معیار ورودی مستخرج از واژه‌ها استفاده شده است. خروجی شبکه عصبی توزیع شباهت میان واژه و حروف انگلیسی را نشان می‌دهد. در مرحله‌ی دوم با استفاده از روش احتمالی، و قوانین بیزین، شباهت میان واژه و واژه‌های درون دیکشنری محدود شده توسط مرحله‌ی اول تخمین زده می‌شود. با استفاده از این روش دقت ۹۴٪ در تصحیح خطا گزارش شده است.

[1] Classifier

در پژوهش‌های دیگری کوکیچ، [60] ,[59]، از شبکه عصبی برای تصحیح نام استفاده کرد. این پژوهش‌ها از الگوریتم انتشار به عقب استفاده کرده‌اند. برای تصحیح نام‌ها ازیک گنجینه لغت با ۱۸۳ نام خانوادگی استفاده شده است. لایه خروجی شبکه عصبی نیز ۱۸۳ گره داشت که هر گره‌یکی از نام‌ها را مدل می‌کرد. لایه ورودی دارای ۴۵۰ گره در ۱۵ بلاک ترتیبی بود، در این حالت هر بلاک شامل ۳۰ گره بوده که می‌توانست نام‌هایی با طول بیشتر از ۱۵ کاراکتر را نیز مدل کند. هر بلاک با ۳۰ گره شامل یک گره برای هر کاراکتر می باشد (الفبای مورد استفاده ۳۰ حرف داشته است). بنابراین به عنوان مثال اگر حرف A به عنوان ورودی داده می‌شد، گره‌ای که این حرف را مدل می‌کرد روشن می‌شد. شبکه عصبی به صدها مثال مصنوعی از نام‌هایی که‌یک خطا دارند آموزش داده شده بود. آموزش شبکه عصبی زمانی مشتمل بر ده‌ها ساعت نیاز داشت. نتایج حاصل از این آزمایش‌ها به صورت زیر است:

شبکه عصبی بروی نام‌هایی که خطا دارند، بهتر از نام‌هایی که خطا ندارند، آموزش می‌بینند.

وقتی تعداد گره‌های لایه‌ی مخفی تا ۱۸۳ گره افزایش می‌یابد، کارآیی اضافه می‌شود و بیشتر از ۱۸۳ گره، کارآیی ثابت می‌ماند.

شبکه‌هایی که از ترتیب کدگذاری شده برای آموزش استفاده می‌کنند، سریعتر از شبکه‌هایی که بصورت توزیع نرمال داده‌ها برای آموزش استفاده می‌کنند، همگرا می‌شوند.

چرکاسکی (۱۹۸۹) ، یافته‌های کوکیچ را مورد تایید قرار داد. در این پژوهش uni-gram و bi-gram به عنوان روش کدگذاری برای ورودی شبکه عصبی استفاده شده است. کدگذاری خروجی نیزیک گره به ازای هر عضو مجموعه‌ی گنجینه لغت است. بنابراین ساختار شبکه عصبی یک دسته بندیک به m است. شبکه عصبی با نام‌های صحیح آموزش داده شده است و سپس شبکه عصبی با نام‌هایی که بصورت مصنوعی یک خطای حذف- جایگزینی در آن ایجاد شده است مورد آزمایش قرار گرفته است. اندازه‌ی گنجینه لغت نیز از ۲۴ الی ۱۰۰ نام متفاوت بوده است. برای گنجینه لغت‌های کوچک دقت ۱۰۰٪ گزارش شده است. همچنین گزارش شد که نرخ آموزش[1] و تعداد گره‌های لایه‌ی مخفی در کارآیی شبکه عصبی موثر است. مقدار بهینه گره‌های لایه‌ی مخفی به تعداد عناصر گنجینه لغت است. این بدان معنی نیست که شبکه عصبی عمومیت ایجاد نکرده است زیرا با دیدن نام‌هایی که در مرحله آموزش ندیده است می‌تواند بخوبی آن‌ها را تصحیح نماید.

کوکیچ [۲۵]، کاربرد شبکه عصبی در تصحیح خطا برای سامانه‌ی تبدیل متن به گفتار را بررسی کرده است. در این پژوهش از گنجینه لغت بزرگتری با ۲۵٪ از خطاهای چندگانه استفاده شده است. شبکه عصبی مورد استفاده سه لایه و الگوریتم آموزش روش انتشار به عقب است. ورودی شبکه عصبی دارای ۴۲۰ گره با المان‌هایی که از uni-gram و bi-gram استفاده می‌کنند، تغذیه می‌شود. تعداد گره‌های لایه‌ی مخفی ۵۰۰ و لایه‌ی خروجی ۱۱۴۲ گره می باشد. با استفاده از پیاده سازی برداری و آزمودن مجموعه‌ای از ۱۷۰ خطا دقت ۷۵٪ در تصحیح خطا بدست آمده است. زمان آموزش شبکه نیز طولانی بوده و مشتمل بر صدها ساعت بوده

[1] Learning rate

است. با استفاده از روش‌هایی که زمان آموزش شبکه عصبی را کاهش می‌دهند، مانند تقسیم‌بندی می‌توان این زمان را کاهش داد.

دفنر و همکارانش [۶۱]، از شبکه عصبی برای تصحیح خطای املایی استفاده کرده‌اند. گنجینه لغت مورد استفاده شامل ۵۰۰۰ واژه بوده است. این سامانه که خود بخشی از سیستم پردازش زبان طبیعی بوده و واژه‌ها را بوسیله‌ی بردارهایی از معیارهای N-gram، معیار تلفظ واژه‌ها، معیار واج و معیارهای معنایی پیاده سازی می‌کند. شباهت بین واژه‌ها بوسیله‌ی معیار همینگ اندازه‌گیری شده است. این روش به دامنه‌ی واژه‌های مورد استفاده در مجموعه وابسته است و شبکه عصبی برای آن مجموعه با توجه به معیارها آموزش داده می‌شود.

باید خاطرنشان کرد که موضوع تصحیح واژه‌ها بصورت منفرد، امروزه برای بسیاری از زبان‌ها منجمله زبان فارسی، یک موضوع تحقیقاتی است و غلط‌یاب کامل در بسیاری از زبان‌ها وجود ندارد و یا اینکه کارآیی آن قابل قبول نیست. یکی از مشکلات مهم در تحقیقات انجام شده این است که مقایسه میان روش‌های مختلف بسیار مشکل است و به غیر از روش مورد مطالعه به مجموعه‌ی آزمون، اندازه‌ی دیکشنری، دامنه‌ی واژه‌ها، طول واژه‌ها و همچنین نوع خطاهای تولید شده بستگی دارد.

در خصوص زبان فارسی تلاش‌هایی در خصوص ساخت غلط‌یاب و مصحح لغوی برای زبان فارسی انجام شده است. موسوی میانگاه [۶۲]، از یک روش ترکیبی دارای سه مرحله برای غلط‌یابی و تصحیح خطا استفاده

کرده است. در مرحله‌ی نخست یک پیکره‌ی تک زبانه برای زبان فارسی تشکیل شده و واژه‌یابی[1] برای آن انجام شده است. سپس تحلیلگر لغوی برای شناسایی خطاهای منفرد غیرواژه بکار گرفته شده است. در مرحله‌ی دوم، برای تصحیح خطا با بکارگیری یک روش ترکیبی، از کمترین فاصله‌ی ویرایشـــــی و اطلاعات N-gram واژه‌ها برای تهیه‌ی فهرست واژه‌های جایگزین استفاده شده است. در نهایت در مرحله‌ی سوم، مجدداً کمترین فاصله‌ی ویرایشی برای ارزش‌گذاری فهرست استفاده شده و به کاربر ارائه می‌گردد. در نهایت گزارش شده است که این سامانه می‌تواند ۹۷٪ از غیرواژه‌ها را شناسایی و حدود ۹۵٪ آنرا تصحیح نماید.

کاشفی و همکارانش [۳]، با یک نوآوری، معیار جدید فاصله‌ی رشته‌ای را برای ارزش گذاری واژه‌های فارسی پیشنهاد دادند. روش پیشنهادی در این پژوهش به ساختار صفحه‌کلید و محل کلیدهای حروف فارسی وابسته است. این معیار جدید فاصله کاشفی[2] نامیده شده است. در نهایت با کمک این معیار دقت ۹۸٪ در تصحیح خطا گزارش شده است. این سامانه به نام غلط‌یاب ویراستیار[3] منتشر شده است.

در پژوهشی دیگر فیلی و همکارانش [۵]، یک سامانه‌ی غلط‌یاب فارسی به نام "وفا" برای زبان فارسی تولید کردند. در این پژوهش از روش ترکیبی براساس روش‌های آماری و مبتنی برقانون برای شناسایی و تصحیح خطا استفاده شده است. برای تهیه فهرست واژه‌های جایگزین و ارزش گذاری آن، ازیک سری معیارهای ترکیبی شامل: کمترین فاصله‌ی ویرایشی، تاثیر صفحه‌کلید، فرکانس واژه‌ها در پیکره و برخی معیارهای ابتکاری استفاده

[1] Tokenization
[2] Kashefi Distance
[3] Virastyar

شده است. در نهایت با بکارگیری مجموعه‌ی آزمون، سامانه‌ی وفا دقت ۹۱٪ در شناسایی خطا و ۷۵٪ در تصحیح خطا را در مقایسه با ویراستیار که دقت ۹۷٪ را در شناسایی خطا و دقت ۷۹٪ را در تصحیح خطا بدست آورده است. البته با بکارگیری معیار میانگین رتبه متقابل[1] (MRR)، (فرمول ۲-۱۰) که میانگین جایگاه قرارگیری واژه‌ی صحیح را در فهرست پیشنهادی نشان می‌دهد، سامانه‌ی وفا جایگاه بهتری از ویراستیار کسب کرده است.

$$\mathrm{MRR} = \frac{1}{|\mathrm{Misspellings}|} \cdot \sum_{i=1}^{N} \frac{1}{\mathrm{Rank}_{\mathrm{Correct-Suggestion}}} \qquad (۲-۷)$$

با استفاده از فرمول ۲-۴۵ می‌توان مشخص کرد کدام سامانه در جایگاه بالاتری توانسته است واژه‌ی جایگزین واژه‌ی غلط را در فهرست پیشنهادی ارائه نماید. این فرمول توسط کانتور و همکارانش [۶۳]، برای کاربرد بازیابی اطلاعات مطرح شد. پس از آن از این معیار برای مقایسه روش‌های ارزش‌گذاری فهرست واژه‌های جایگزین واژه‌ی غلط استفاده شده است. مشخصاً روشی که واژه‌ی صحیح را در بالای فهرست (جایگاه‌های برتر) قرار دهد، روش بهتری است.

[1] Mean Reciprocal Rank

فصل سوم

روش پژوهش

۳-۱-مقدمه

آنالیز آماری الگوهای خطا برای ساخت الگوی تصحیح خطا در بسیاری از کاربردها اهمیت دارد. به عنوان مثال در تصحیح خطاهای لغوی، شناسایی نوری نویسه‌ها و تبدیل متن به گفتار شناسایی الگوی خطا بسیار مهم است و می توان از آن برای بهبود کیفیت و کارآیی کاربرد استفاده کرد.

برای انجام این آنالیز آماری در ابتدا لازم است پیکره‌ای برچسب‌دار[1] از واژگان فارسی بر اساس انواع خطاهای مرسوم تهیه شود. بنابراین با توجه به عدم دسترسی به چنین پیکره‌ای یکی از دستاوردهای این پژوهش یک پیکره‌ی برچسب‌دار از واژگان فارسی است. این پیکره بصورت خودکار (مصنوعی) توسط یک الگوریتم رایانه‌ای که در ادامه به تفضیل شرح داده خواهد شد.

پس از ساخت پیکره، شناسایی و تصحیح خطای لغوی توسط نرم‌افزارهای غلط‌یابی متن که در دسترسند، مورد بررسی قرار می‌گیرد. در این پژوهش از غلط‌یاب‌های ویراستیار [۳]، وفا [۵] و Perspell [۱] استفاده شده است. هر سه غلط‌یاب قابلیت شناسایی و تصحیح خطاهای لغوی را دارند. تاثیر طول واژه بر شناسایی و تصحیح خطا و همچنین تاثیر محل وقوع خطای لغوی (از چهار نوع درج، حذف، جابجایی و جایگزینی) بر شناسایی و تصحیح واژه مورد بررسی قرار می‌گیرد.

الگوهای یافت شده از نظر آماری مهم است و می‌تواند در طراحی کاربردهایی مانند غلط‌یاب استفاده شود.

[1] Labeled Corpus

۳-۲-ساخت پیکره برچسب‌دار لغوی زبان فارسی

پژوهش در حوزه‌ی زبان فارسی همواره با چالش‌هایی روبرو است. یکی از مهمترین این چالش‌ها، کمبود منابع است. با توجه به عدم دسترسی به منابع مورد نیاز، بسیاری از پژوهش‌ها در این حوزه عملاً امکان‌پذیر نیست و یا دارای هزینه‌ی بسیار زیادی در راستای تهیه منابع است.

تهیه منابعی مانند پیکره‌ها بسیار مهم و از زمره‌ی پژوهش‌های بنیادی در حوزه‌ی پردازش زبان طبیعی است. تهیه پیکره‌ها اصولاً به دو صورت کاملاً انسانی و یا خودکار صورت می‌پذیرد. اگر پیکره به صورت انسانی تهیه شود، باید با بکارگیری نیروهای متخصص این مهم انجام شود. این فرآیند دارای هزینه نیروی انسانی زیادی است و معمولاً زمان زیادی را نیز طلب می‌کند. بنابراین روش انسانی برای تهیه پیکره‌های بزرگ عملاً مقرون به صرفه نیست.

روش‌های خودکار تهیه پیکره نیز معمولاً مشکلات زیادی دارد. با توجه به پیچیدگی‌های زبانی، معمولاً تهیه یک الگوریتم همه جانبه بسیار مشکل است. روش‌های بکار رفته در این بخش نیز معمولاً دقت کافی ندارند و یا از نظر رایانه‌ای نیاز به پردازش‌های سنگین دارند. استفاده از روش‌های آماری در کنار الگوریتم‌های هوش مصنوعی (روش ترکیبی) بهترین کارآیی را در این بخش دارد [۶۴], [۶۵]. بنابراین در این پژوهش از روش خودکار برای تهیه پیکره استفاده شده است. مرسوم است که بخش اندکی از پیکره (حدود ۵٪) بصورت انسانی کنترل شود تا از صحت عملکرد الگوریتم اطمینان حاصل گردد.

برای تهیه پیکره ابتدا باید هدف پیکره مشخص باشد. این پیکره مشخصاً برای آزمودن دقت غلط‌یاب لغوی طراحی شده است. بنابراین باید انواع غلط‌های لغوی مرسوم را پشتیبانی کند. غلط‌های لغوی معمولاً به دو دسته اصلی تقسیم می‌شوند. دسته اول شامل خطاهایی است که کاربر املای صحیح واژه را نمی‌داند، بنابراین حتی با دیدن واژه‌ی صحیح قادر به تصحیح واژه‌ی غلط نیست. دسته‌ی دوم خطاهایی هستند که سهواً و یا در اثر تایپ الکترونیکی بوجود می‌آیند. این گروه نیز به قالب صفحه کلید و زبان نویسه‌ها حساس است.

با توجه به اینکه زبان فارسی دارای نویسه‌های هم‌آوا[1]، شبیه به هم از نظر نوشتاری[2]، نویسه‌های متصل و منفصل و صفحه‌کلید چندمنظوره با کلیدهای ترکیبی است، پتانسیل تولید درصد بالایی از خطاهای لغوی وجود دارد. بنابراین الگوریتم تهیه شده باید تمامی حالات فوق را در نظر داشته باشد. به عنوان مثال در زمان تایپ نویسه‌ها احتمال فشردن کلیدی که در مجاورت کلید هدف قرار دارد بیشتر است. بنابراین برای نویسه‌ها معیار مجاورت تعریف شده است [۳]. این معیار، معیار فاصله‌ی کاشفی نام دارد و براساس صفحه کلید فارسی طراحی شده است.

شکل ۳-۱ الگوریتم تولید پیکره را نشان می‌دهد. همانطور که الگوریتم شکل ۳-۱ نشان می‌دهد. برای تولید پیکره لازم است به تصادف برای برخی از واژگان متن با استفاده از یک الگوریتم، خطای لغوی تایپی ایجاد شود. این خطا باید با در نظر گرفتن ویژگی‌های خاص خط و نویسه‌های فارسی، نویسه‌های هم آوا، نویسه‌های

[1] Homophone
[2] Grapheme

هم‌شکل و هم‌جواری نویسه‌ها در صفحه‌کلید تهیه شود. به همین منظور الگوریتم شکل ۳-۱ علاوه بر حالت‌های مرسوم درج، حذف، جابجایی و جایگزینی که با در نظر گرفتن هم‌جواری نویسه‌ها در صفحه‌کلید انجام می‌شود، هم‌آوایی و هم‌شکل بودن نویسه‌ها را نیز در نظر می‌گیرد.

```
List <string> ConfusionSet(string word)
{
   List <string> Con_set=new List<string>();
   Foreach(letter in word)
   {
      all_Persian_letters=neighbors_of(letter);
      Candidate_string=Insert_letter(all_Persian_letters,letter,word);//insertion
       Con_set.Add(Candidate_string);
      Candidate_string=Delete_letter(letter,word);//deletion
       Con_set.Add(Candidate_string);
      Candidate_string=Replace_letter(all_Persian_letters,letter,word);//substitution
       Con_set.Add(Candidate_string);
      Candidate_string=Transmit_letter(letter,all_letter_word,word);//transposition
       Con_set.Add(Candidate_string);
      Candidate_string=Replace_letter(Graphemes,letter,word);//Graphemes
       Con_set.Add(Candidate_string);
      Candidate_string=Replace_letter(Homophone,letter,word);//Homophones
       Con_set.Add(Candidate_string);
   }
    Validate_with_lexicon(Con_set);
    Return(Con_set);
}
```

شکل ۳-۱-الگوریتم تولید واژه‌های پیکره‌ی لغوی فارسی

متن مورد استفاده برای این بخش از دو پیکره استخراج می گردد. برای تهیه پیکره از متن چکیده مقالات علمی پژوهشی فارسی و متن پیکره‌ی خبری پرسیکا [۶۶] استفاده شده است. پایگاه‌داده‌ی مقالات فارسی دارای بیش از ۴۰۰ هزار مقاله فارسی است و در این پژوهش متن ۱۰۰۰ مقاله‌ی فارسی از حوزه‌های موضوعی فنی

مهندسی، کشاورزی، علوم انسانی، علوم پایه و پزشکی به روش نمونه‌گیری طبقه‌بندی شده‌ی تصادفی[1] انتخاب می‌شود. در این پیکره بیش از ۱.۴ میلیون پاراگراف، ۷ میلیون جمله دارد. به همین ترتیب ۱۰۰۰ متن خبری نیز از پیکره‌ی پرسیکا [۶۶] به روش نمونه‌گیری طبقه‌بندی شده‌ی تصادفی انتخاب می‌شود. پیکره‌ی پرسیکا دارای حدود ۱۰ هزار متن خبری در حدود ۱۰۰ حوزه‌ی موضوعی است.

حال دو مجموعه‌ی ۱۰۰۰ عضوی از متون علمی مقالات و متون خبری موجود است. هر متن با یک کد ارجاع مشخص می‌شود. این کد نشان می‌دهد که اصل متن برگرفته کدام متن در پیکره‌ی اصلی است. حال با استفاده از الگوریتم شکل ۳-۱ ، برای هر متن حدود ۲۰٪ از واژه‌ها با طول‌های مختلف به تصادف انتخاب می‌شود و برای هر واژه‌ی انتخاب شده، بصورت مصنوعی خطای لغوی ایجاد می‌شود. سپس محل وقوع خطا، رشته‌ی محتوی خطا و واژه‌ی صحیح در پیکره ذخیره می‌شود. بدین ترتیب پیکره بدست آمده قابل خواندن توسط ماشین[2] خواهد بود. برای آنکه خطای ایجاد شده لغوی باشد باید رشته‌ی بدست آمده واژه‌ی صحیحی در زبان فارسی نباشد در غیر اینصورت، خطا معنایی خواهد بود. به عنوان مثال اگر در تولید خطا واژه‌ی "زمان" نویسه‌ی "ز" به "ر" تبدیل شود، آنگاه واژه‌ی "رمان" بدست می‌آید که خطای معنایی است و خطای لغوی نمی‌باشد زیرا "رمان" خود یک واژه‌ی صحیح فارسی است.

[1] Stratified Random Sampling
[2] Machine readable

شکل ۳-۲ نمونه ای از متن پیکره را نشان می‌دهد. در این بخش مقدمات پژوهش که منبع داده‌های مورد استفاده است به صورت یک پیکره تهیه می‌شود این پیکره از دستاوردهای این پژوهش است و می‌توان از آن برای سنجش دقت غلط‌یاب در خطاهای لغوی استفاده نمود.

مقدمه: گياه سنبل‌الطيب به‌واسطه اثغرات آرام‌بخشي، ضدتشنجي وضددردي از ديرباز در طب سنتي چين، هند وذايران جايگاه خاصي داشته‌ناست. اين گياه باتوجه به داشتن آلكالوييدها-استراسيدهاي آلي –اسيد والريک وايزووالريک ها همواره درجهت كاهش فشارهاي عصبي، درمان افسردگي وبي‌خوابي مزمن مورداستفاده قرارگرفته است. هدف: باتوجه به تهميت موضموع برآن شديم كه اثرات ضداضطرابي وپيش بيهوشي عبصاره اين گياه را باداروي شيمسايي به‌صورت مقديسه‌اي موردمطالعه قرار دهيم.

کد انحصاری متن	تفاوت مکان	واژه صحیح	رشته غلط
349	5	اثرات	اثغرات
349	20	ایران	ذایران
349	23	داشته‌است	داشته‌ناست
349	59	اهمیت	تهمیت
349	60	موضوع	موضموع
349	69	عصاره	عبصاره
349	74	شیمیایی	شیمسایي
349	76	مقایسه‌ای	مقدیسه‌اي

شکل ۳-۲-نمونه‌ی متن دارای خطای لغوی و داده‌های ذخیره شده برای آن

در تهیه پیکره فوق از مجموع متون خبری و علمی استفاده شده است که دایره‌ی واژگان مورد استفاده محدود به متن خاص علمی نباشد و متون روزمره‌ی خبری را نیز شامل شود.

۳-۳-غلط یابی لغوی

برای آنکه بتوان آزمون درستی از پیکره‌ی تهیه شده بدست آورد، در این پژوهش از سه نرم افزار غلط‌یاب در دسترس استفاده شده است. ویراستیار به صورت افزونه به نرم‌افزار ویراستار ورد اضافه می‌شود و برای تصحیح خطا معیاری بنام فاصله‌ی کاشفی را معرفی کرده است [۳]. این سامانه امکان تشخیص و تصحیح خطاهای لغوی را دارد. نرم افزار غلط‌یاب دیگری که در این پژوهش مورد استفاده قرار گرفته است، نرم افزار وفا [۵] است. این نرم افزار نیز بصورت افزونه بر نرم افزار ویراستار ورد اضافه می‌شود و امکان تشخیص و تصحیح غلط‌های لغوی و معنایی را دارد. نرم افزار دیگری که در این پژوهش به عنوان نرم افزار غلط‌یاب از آن استفاده شده است، نرم افزار پارسی‌اسپل (Perspell) است [۱]. این نرم‌افزار نیز قابلیت تشخیص و تصحیح خطاهای لغوی و معنایی را دارد. پیکره‌ی تهیه شده با هر سه نرم‌افزار به صورت جداگانه مورد آزمایش قرار گرفته و نتایج آن ذکر می‌گردد.

فصل

چهارم

آزمون‌ها و نتایج

۴-آزمون‌ها و نتایج

در این فصل آزمون‌هایی ترتیب داده شده است تا با کمک آن بتوان تاثیر محل رخداد خطا در رشته را بر دقت تشخیص و تصحیح واژه بررسی نمود. همچنین اثر طول بر تشخیص و تصحیح خطا بررسی می‌شود. برای این منظور از پیکره‌ی تولید شده در این پژوهش به عنوان داده‌ی آزمون استفاده می‌شود. ابزار مورد استفاده برای انجام آزمون در فصل ۳-۳ شرح داده شده است.

۴-۱-پیکره آزمون املایی

در فصل ۳-۲ روش تولید پیکره‌ی آزمون شرح داده شد. برای تحلیل اثر طول واژگان فارسی و محل وقوع خطا در رشته بر شناسایی و تصحیح خطا، از سه ابزار غلط‌یاب و پیکره‌ی تولید شده در این پژوهش استفاده می‌شود. متن تولید شده در پیکره املایی که شامل غلط‌های لغوی است، با استفاده از نرم‌افزارهای غلط‌یاب مورد پردازش قرار می‌گیرد. برای هر واژه‌ی غلط، شناسایی شدن و تصحیح شدن آن توسط سه ابزار استفاده شده ثبت می‌گردد.

با توجه به گستره‌ی دایره‌ی واژگان پیکره‌ی املایی، متن شامل متون علمی در تمامی حوزه‌ها و همچنین متون خبری در تمامی حوزه‌های خبری است. خطاهای لغوی که به صورت مصنوعی تولید شده‌اند نیز در تمامی حالات ممکن توزیع شده است. به عبارت دیگر خطای لغوی با جایگزینی، حذف، جابجایی و درج نویسه در ابتدا، میان واژه و یا انتهای واژه تولید شده است.

همانطور که در جدول ۴-۱ نشان داده شده است، میانگین طول متن پیکره‌ی املایی حدود ۱۱۵۷ کاراکتر (به همراه فضای خالی) است. میانگین تعداد خطای لغوی بازای هر مدرک، ۲۰ عدد خطا است. بیشینه طول رشته‌ی خطادار ۱۷ کاراکتر و کمینه‌ی آن ۲ کاراکتر است. میانگین طول رشته‌ی دارای خطای لغوی ۵ کاراکتر است.

جدول ۴-۱- اطلاعات آماری پیکره‌ی املایی

میانگین طول هر متن (کاراکتر)	۱۱۵۷
میانگین تعداد رشته‌های خطادار (بازای هر مدرک)	۲۰
میانگین طول رشته‌ی خطا دار (واژه‌ی دارای خطای لغوی)	۵
بیشینه طول رشته‌ی خطا دار	۱۷
کمینه طول رشته‌ی خطا دار	۲

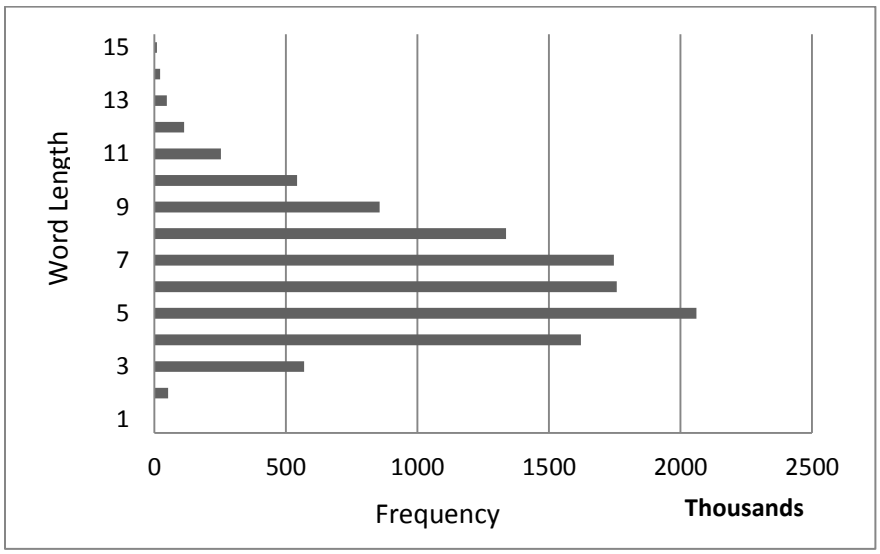
در ابزارهای غلط‌یاب مورد استفاده در این پژوهش از الگوریتم آماری بر پایه‌ی دیکشنری برای شناسایی خطا استفاده می‌شود. برای تصحیح خطای شناسایی شده، هر سه ابزار غلط‌یاب با استفاده از معیارهای مقایسه‌ی رشته‌ها مانند فاصله‌ی ویرایشی، فاصله‌ی کاشفی، اطلاعات متقابل، روشهای ترکیبی و مدل زبان سعی در تصحیح رشته‌ی دارای خطای لغوی دارند [۱]، [۳]، [۵]، [۹]. برای تصحیح خطا، نرم افزارهای غلط‌یاب فهرستی

از واژه‌های پیشنهادی که در فاصله‌ی ویرایشی یک نسبت به واژه‌ی اصلی است به کاربر پیشنهاد می‌دهند تا با استفاده از این فهرست واژه‌ی صحیح انتخاب شود. در این پژوهش اگر واژه‌ی اصلی که در پیکره‌ی برچسب خورده و رشته‌ی خطادار لغوی از آن تولید شده است در این فهرست وجود داشته باشد، فرآیند تصحیح به عنوان موفق علامت زده می‌شود و در غیر اینصورت فرآیند تصحیح به عنوان ناموفق ثبت می‌گردد.

معمولاً در بیش از ۸۰ درصد مواقع واژه‌ی صحیح در فاصله ویرایشی یک نسبت به رشته‌ی خطادار قرار دارد، بنابراین تمامی نرم‌افزارهای غلط‌یاب فاصله‌ی ویرایشی یک را برای تشکیل مجموعه‌ی ابهام استفاده می‌کنند [۱], [۲]. تعداد واژه‌هایی که در فهرست پیشنهادی می‌توان به کاربر نشان داد محدود است بنابراین کارآیی الگوریتم ارزش‌گذاری و انتخاب فهرست نهایی از میان مجموعه‌ی ابهام بسیار مهم است.

شکل ۴-۱ تعداد اعضای مجموعه‌ی ابهام براساس طول را نشان می‌دهد. بر اساس شکل ۴-۱ واژه‌هایی با طول ۴ الی ۷ کاراکتر که از نظر آماری بسامد بیشتری در پیکره‌ها دارند، دارای بیشترین تعداد واژه در مجموعه‌ی ابهام هستند. بزرگترین مجموعه‌ی ابهام دارای ۴۳۵ عضو و کوچترین مجموعه‌ی ابهام تنها دارای یک عضو است. میانگین تعداد اعضای مجموعه‌ی ابهام برای تمامی واژه‌ها با در نظر گرفتن فاصله‌ی ویرایشی یک برابر ۹ عضو، با درنظر گرفتن فاصله‌ی ویرایشی دو برابر ۲۹ عضو و با درنظر گرفتن فاصله ویرایشی صه برابر ۵۳ عضو است. همانطور که نتایج نشان می‌دهد، با افزایش فاصله‌ی ویرایشی اعداد اعضای مجموعه‌ی ابهام به شدت رشد می‌کند و این امر موجب افزایش نویز می‌شود. مسلماً با توجه به محدودیت تعداد اعضایی که می توان به کاربر نمایش

داد، افزایش تعداد اعضای مجموعه‌ی ابهام می تواند چالش بزرگی باشد. بنابراین شناسایی الگوی خطا و اثر طول و محل وقوع خطا می‌تواند الهام بخش پژوهشگران را در تهیه فرمول‌های رتبه‌بندی اعضای مجموعه‌ی ابهام باشد.



شکل۴-۱-بسامد واژگان در مجموعه‌های بهام بر اساس طول واژه

۴-۲-بحث و نتیجه گیری

در این بخش آزمون‌های ترتیب داده شده است تا با استفاده از آن تاثیر طول واژه در دو فرآیند تشخیص و تصحیح خطای لغوی مورد بررسی قرار گیرد. همچنین با بکارگیری واژه‌های دارای خطای لغوی در سه بخش ابتدا، میان و انتهای واژه، تاثیر محل رخداد خطا در رشته‌ی دارای خطای لغوی نیز در دو فرآیند تشخیص و تصحیح خطای لغوی بررسی می‌شود.

برای این منظور از پیکره‌ی تولید شده در این پژوهش و سه غلط‌یاب در دسترس، ویراستیار [۳]، غلط‌یاب وفا [۵] و غلط‌یاب پارسی‌اسپل [۱] استفاده می‌شود. هر سه نرم‌افزار یاد شده دارای قابلیت تشخیص و تصحیح خطای لغوی می‌باشند، بنابراین از این نرم‌افزارها برای دو فرآیند تشخیص و تصحیح خطای لغوی استفاده شده است.

تشخیص خطای لغوی عبارت است از یافتن رشته‌ی دارای خطا توسط نرم‌افزار غلط‌یاب و علامت زدن آن. به همین صورت تصحیح خطا که خود فرآیند جداگانه‌ای محسوب می‌شود عبارت از یافتن واژه‌ی اصلی است که کاربر آنرا به غلط تایپ کرده است. این فرآیند با پیشنهاد تعدادی واژه‌ی جایگزین برای رشته‌ی غلط انجام می‌شود. مسلماً با توجه به محدود بودن تعداد واژه‌هایی که می‌توان پیشنهاد داد، یکی از چالش‌های مهم این حوزه الگوریتم رتبه‌بندی نتایج است [۱], [۲].

برای انجام آزمایش‌های فوق با نظارت انسانی متن به نرم‌افزارها داده شده و نتایج بدست آمده برای هر دو فرآیند تشخیص و تصحیح ثبت می‌شود. برای ارزیابی نتایج بدست آمده در مرحله‌ی تشخیص از فرمول ۴-۱ استفاده شده است.

$$P_i = \frac{TP}{TP+FP} \ , \ P = \frac{\sum_{i=1}^{N} P_i}{N} \qquad ۴\text{-}۱$$

همانطور که در فرمول ۴-۱ نشان داده است، در هر مرحله برای هر مدرک، تعداد تشخیص‌های درست در صورت کسر جزئی و مخرج آن کل تعداد غلط‌های املایی گزارش شده را شامل می‌شود. برای کل مدارک مجموعه‌ی

آزمون که در پیکره وجود دارد، میانگین کل امتیاز مدارک (میانگین دقت[1]) محاسبه می‌شود. به همین ترتیب برای ارزیابی فرآیند تصحیح خطا نیز از فرمول ۴-۱ استفاده می‌شود.

جدول ۴-۲ نتایج آزمون را برای هر سه نرم افزار غلط‌یاب، در فرآیندهای تشخیص و تصحیح خطا نشان می‌دهد. همانطور که در جدول ۴-۲ نشان داده شده است میانگین دقت نرم‌افزارهای غلط‌یاب برای مجموعه‌ی آزمون بر اساس فرمول ۴-۱ در دو فاز تشخیص و تصحیح خطای لغوی محاسبه شده است.

جدول ۴-۲-نتایج ارزیابی نرم افزارهای غلط‌یاب در فاز تشخیص و تصحیح خطای لغوی

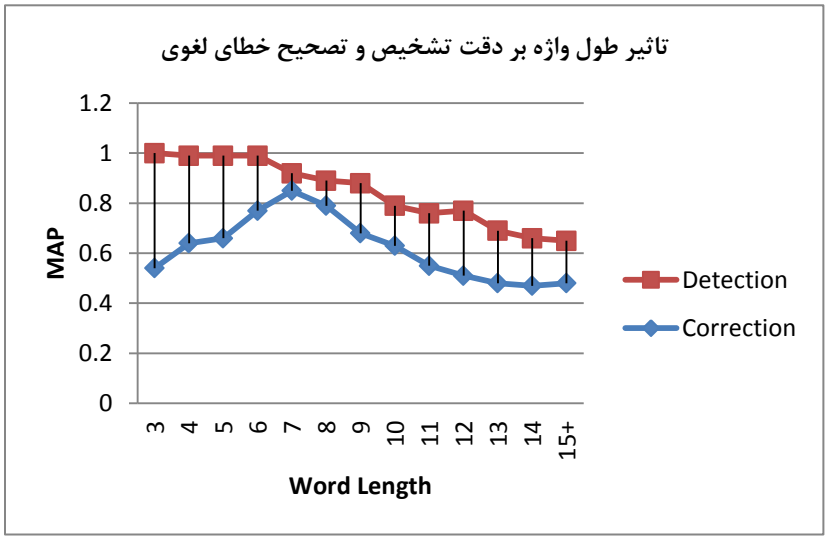
نرم افزار	تشخیص خطا	تصحیح خطا
پارسی‌اسپل	0.96	0.82
وفا	0.91	0.75
ویراستیار	0.96	0.79

برای تحلیل بیشتر میانگین دقت تشخیص و تصحیح خطا برای واژه‌هایی با طول ۳ الی ۱۵ را مطالعه می‌کنیم. در هر مرحله میانگین دقت هر سه غلط‌یاب (MAP[2]) محاسبه می شود. شکل ۴-۲ مقدار میانگین دقت تشخیص و تصحیح خطای لغوی را برای واژه‌هایی به طول ۳ الی ۱۵ را نشان می‌دهد.

همانطور که در شکل ۴-۲ مشاهده می‌شود، تشخیص واژه‌هایی با طول بیشتر برای نرم‌افزارهای غلط‌یاب کمی مشکل‌تر و پیچیده‌تر از واژه‌هایی با طول کمتر است. بیشترین بسامد واژگان در پیکره‌های فارسی برای واژه‌هایی

[1] Average Precision
[2] Mean Average Precision

با طول ۵ الی ۷ نویسه است [۱], [۳], [۴], [۶۷]. به هر ترتیب خطا در واژه‌های کوتاه تا متوسط با دقت بیشتری شناسایی می‌شود.



شکل ۴-۲-تاثیر طول واژه بر دقت تشخیص و تصحیح خطای لغوی

به همین ترتیب اگر با توجه به شکل ۴-۲ به نمودار تصحیح خطای لغوی نگاه کنیم، متوجه می‌شویم که تصحیح واژه‌هایی با طول کم معمولاً برای نرم افزارهای غلط‌یاب از دیگر واژه‌ها مشکل‌تر است. این چالش مشابه برای واژه‌های بلند از نظر تعداد نویسه‌ها نیز وجود دارد.

واژه‌های کوتاه، اطلاعات کمتری را منتقل می‌کنند بنابراین در زمان تصحیح خطا مشکلات بیشتری را برای نرم‌افزار غلط‌یاب ایجاد می‌کنند. همچنین تغییر بخش کوچکی از واژه‌ی کوتاه، در حقیقت درصد بزرگی از واژه را

دستخوش تغییر کرده است که این امر می‌تواند موجب ورود نویز بیشتر در مجموعه‌ی ابهام واژه گردد و بنابراین تصحیح واژگان با چالش جدیدی روبرو خواهد گردید.

به همین ترتیب واژه‌های بلند معمولاً متداول نیستند (قانون زیف) بنابراین این واژه‌ها ممکن است در گنجینه لغت دیکشنری موجود نباشند. بنابراین دقت غلط‌یاب در مواجه شدن با واژه‌های بلندتر کاهش می‌یابد. برای واژه‌های متداول، این مشکل بسیار کم رنگ‌تر است و غلط‌یاب بخوبی می تواند خطا در واژه‌های پر بسامد را تشخیص دهد. ولی تصحیح خطای واژه‌های کوتاه نیز مشکلات خاص خود را دارد.

جدول ۴-۳-تاثیر محل بروز خطا بر دقت تشخیص و تصحیح خطا

محل بروز خطا	میانگین دقت تصحیح	میانگین دقت تشخیص
ابتدای واژه	**0.71**	**0.98**
میان واژه	**0.80**	**0.99**
انتهای واژه	**0.81**	**0.98**

همانطور که نتایج آزمایش‌ها نشان می‌دهد (جدول ۴-۳)، اگر خطای لغوی در ابتدا و انتهای واژه رخ دهد، موجب کاهش دقت تشخیص خطا می‌شود. رخداد خطای لغوی در میان واژه در عمل بسامد کمتری دارد و تاثیر آن بر دقت تشخیص خطای لغوی نیز ناچیز است.

فرآیند تصحیح خطای لغوی با فرآیند تشخیص متفاوت است. در فرآیند تشخیص واژه‌هایی که خارج از دیکشنری باشد به عنوان غلط لغوی علامت زده می‌شود. در فرآیند تصحیح خطا، باید برای رشته‌ی دارای خطای

لغوی یک مجموعه از واژه‌هایی که محتمل برای جایگزینی رشته‌ی غلط هستند تهیه شود. این مجموعه، مجموعه‌ی ابهام نام دارد. برای تهیه‌ی مجموعه‌ی ابهام، مرسوم است که از واژه‌هایی در فاصله‌ی ویرایشی یک از واژه‌ی اصلی به عنوان مجموعه‌ی ابهام انتخاب شود زیرا بیش از ۸۰٪ خطاهای لغوی صرفاً در فاصله ویرایشی یک از واژه‌ی اصلی قرار دارند [۱]–[۳].

رخداد خطای لغوی در ابتدای واژه تاثیر مستقیم بر دقت تصحیح خطا دارد و موجب کاهش دقت تصحیح خطا می‌شود ولی اگر خطای لغوی در میان و یا انتهای واژه رخ دهد، تاثیر کمتری بر دقت تصحیح خطا دارد. دلیل کاهش دقت تصحیح خطای لغوی وابسته به الگوی نویسه‌ای زبان است.

آزمایش‌های انجام شده در این پژوهش نشان داد که طول واژه و محل وقوع خطای لغوی بر دقت تشخیص و تصحیح خطای لغوی تاثیرگذار است. نتایج بدست آمده در این پژوهش می‌تواند سرلوحه‌ی پژوهشگران برای سامانه‌های غلط‌یاب، شناسایی نوری نویسه‌ها و سامانه‌های تشخیص متن باشد. واژه‌هایی که طبق این پژوهش دارای ابهام بیشتری در زمان تصحیح و تشخیص خطا هستند، می‌توانند با الگوی رفتاری متفاوتی توسط نرم‌افزار غلط‌یاب بررسی شوند، تا میانگین دقت نرم‌افزار قابل قبول باشد.

نتایج این پژوهش نشان داد که فرآیند تشخیص خطای لغوی کمتر به الگوی واژه بستگی دارد ولی تصحیح خطای لغوی به شدت به محل و طول واژه حساس است بنابراین می‌توان از روش‌های وابسته به متن[1] مانند مدل

[1] Context Sensitive

زبان[1]، تخمین مارکف، بردار واژه‌های همجوار و اطلاعات متقابل[2] برای بالا بردن دقت تصحیح خطای لغوی استفاده کرد.

[1] Language Model
[2] Mutual Information

۵-منابع

[1] M. B. Dastgheib, S. M. Fakhrahmad, and M. ZolghadriJahromi, “Perspell: A New Persian Semantic-Based Spelling Correction System,” *Digit. Scholarsh. Humanit.*, vol. 0, no. 0, p. fqw015, 2016.

[2] R. Mitton, “Spelling checkers, spelling correctors and the misspellings of poor spellers,” *Inf. Process. Manag.*, vol. 23, no. 5, pp. 495–505, 1987.

[3] O. Kashefi, M. Sharifi, and B. Minaie, “A novel string distance metric for ranking Persian respelling suggestions,” *Nat. Lang. Eng.*, vol. 19, no. 02, pp. 259–284, 2012.

[4] T. M. Miangah, “FarsiSpell: A spell-checking system for Persian using a large monolingual corpus,” *Lit. Linguist. Comput.*, vol. 29, no. 1, pp. 56–73, 2014.

[5] H. Faili, N. Ehsan, M. Montazery, and M. T. Pilehvar, “Vafa spell-checker for detecting spelling, grammatical, and real-word errors of Persian language,” *Lit. Linguist. Comput.*, p. fqu043, 2014.

[6] T. Naseem and S. Hussain, “A novel approach for ranking spelling error corrections for Urdu,” *Lang. Resour. Eval.*, vol. 41, no. 2, pp. 117–128, 2007.

[7] M. A. Farajian, “PEN: parallel english-persian news corpus,” in *Proceedings of the 2011th World Congress in Computer Science, Computer Engineering and Applied Computing*, 2011.

[8] M. S. Rasooli, O. Kashefi, and B. Minaei-Bidgoli, “Extracting parallel paragraphs and sentences from English-Persian translated documents,” in *Lecture Notes in Computer*

Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 7097 LNCS, Springer, 2011, pp. 574–583.

[9] Kashefi, M. Nasri, and K. Kanani, "Towards Automatic Persian Spell Checking," *Tehran, Iran SCICT*, 2010.

[10] K. Min, W. H. Wilson, and Y.-J. Moon, "Typographical and Orthographical Spelling Error Correction.," in *LREC*, 2000.

[11] F. J. Damerau, "A technique for computer detection and correction of spelling errors," *Commun. ACM*, vol. 7, no. 3, pp. 171–176, 1964.

[12] J. Wu, H. Chiu, and J. S. Chang, "Integrating dictionary and web N-grams for chinese spell checking," *Comput. Linguist. Chinese Lang. Process.*, vol. 18, no. 4, pp. 17–30, 2013.

[13] E. J. Sitar, "Machine recognition of cursive script: The use of context for error detection and correction," *Bell Labs Tech. Mem*, 1961.

[14] L. D. Harmon, "Automatic recognition of print and script," *Proc. IEEE*, vol. 60, no. 10, pp. 1165–1176, 1972.

[15] A. R. Hanson, E. M. Riseman, and E. Fisher, "Context in word recognition," *Pattern Recognit.*, vol. 8, no. 1, pp. 35–45, 1976.

[16] R. Morris and L. L. Cherry, "Computer detection of typographical errors," *Prof. Commun. IEEE Trans.*, no. 1, pp. 54–56, 1975.

[17] T. N. Turba, "Checking for spelling and typographical errors in computer-based text," in

ACM SIGPLAN Notices, 1981, vol. 16, no. 6, pp. 51–60.

[18] D. E. Knuth, *The art of computer programming: sorting and searching*, vol. 3. Pearson Education, 1998.

[19] A. V Aho and M. J. Corasick, "Efficient string matching: an aid to bibliographic search," *Commun. ACM*, vol. 18, no. 6, pp. 333–340, 1975.

[20] J. L. Peterson, "Computer programs for detecting and correcting spelling errors," *Commun. ACM*, vol. 23, no. 12, pp. 676–687, 1980.

[21] J. L. Peterson, "A note on undetected typing errors," *Commun. ACM*, vol. 29, no. 7, pp. 633–637, 1986.

[22] F. J. Damerau and E. Mays, "An examination of undetected typing errors," *Inf. Process. Manag.*, vol. 25, no. 6, pp. 659–664, 1989.

[23] D. Walker and R. Amsler, "The use of machine-readable dictionaries in sublanguage analysis," *Anal. Lang. Restricted Domains*, pp. 69–83, 1986.

[24] ز. زندی مقدم, *فرهنگ املایی خطّ فارسی*. تهران: فرهنگستان زبان و ادب فارسی, ۱۳۸۵ and ع. ا. صادقی.

[25] K. Kukich, "Spelling correction for the telecommunications network for the deaf," *Commun. ACM*, vol. 35, no. 5, pp. 80–90, 1992.

[26] J. J. Pollock and A. Zamora, "Automatic spelling correction in scientific and scholarly text," *Commun. ACM*, vol. 27, no. 4, pp. 358–368, 1984.

[27] G. K. Zipf, "The psycho-biology of language.," 1935.

[28] J. J. Pollock and A. Zamora, “Collection and characterization of spelling errors in scientific and scholarly text,” *J. Am. Soc. Inf. Sci.*, vol. 34, no. 1, pp. 51–58, 1983.

[29] K. Kukich, “A comparison of some novel and traditional lexical distance metrics for for spelling correction,” in *Proceedings of INNC-90-Paris*, 1990, pp. 309–313.

[30] B. van Berkel and K. De Smedt, “Triphone analysis: a combined method for the correction of orthographical and typographical errors,” in *Proceedings of the second conference on Applied natural language processing*, 1988, pp. 77–83.

[31] R. A. Wagner and M. J. Fischer, “The string-to-string correction problem,” *J. ACM*, vol. 21, no. 1, pp. 168–173, 1974.

[32] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” in *Soviet physics doklady*, 1966, vol. 10, no. 8, pp. 707–710.

[33] J. Veronis, “Computerized correction of phonographic errors,” *Comput. Hum.*, vol. 22, no. 1, pp. 43–56, 1988.

[34] M. Mor and A. S. Fraenkel, “A hash code method for detecting and correcting spelling errors,” *Commun. ACM*, vol. 25, no. 12, pp. 935–938, 1982.

[35] R. E. Gorin, “SPELL: A spelling checking and correction program,” *Online Doc. DEC-10 Comput.*, 1971.

[36] K. W. Church and W. A. Gale, “Enhanced Good-Turing and Cat-Cal: Two new methods for estimating probabilities of English bigrams,” in *Proceedings of the workshop on Speech and Natural Language*, 1989, pp. 82–91.

[37] M. R. Dunlavey and L. A. Miller, “Technical corrections: Onspelling correction and beyond,” *Commun. ACM*, vol. 24, no. 9, pp. 608–609, 1981.

[38] F. E. Muth and A. L. Tharp, “Correcting human error in alphanumeric terminal input,” *Inf. Process. Manag.*, vol. 13, no. 6, pp. 329–337, 1977.

[39] R. Russell and M. Odell, “Soundex,” *US Pat.*, vol. 1, 1918.

[40] D. L. Bitzer, B. A. Sherwood, and P. Tenczar, *Computer-based science education*. University of Illinois, Computer-based Education Research Laboratory, 1973.

[41] E. J. Yannakoudakis and D. Fawthrop, “An intelligent spelling error corrector,” *Inf. Process. Manag.*, vol. 19, no. 2, pp. 101–108, 1983.

[42] L. G. Means, “Cn yur cmputr raed ths?,” in *Proceedings of the second conference on Applied natural language processing*, 1988, pp. 93–100.

[43] E. M. Riseman and A. R. Hanson, “A contextual postprocessing system for error correction using binary n-grams,” *Comput. IEEE Trans.*, vol. 100, no. 5, pp. 480–493, 1974.

[44] R. C. Angell, G. E. Freund, and P. Willett, “Automatic spelling correction using a trigram similarity measure,” *Inf. Process. Manag.*, vol. 19, no. 4, pp. 255–261, 1983.

[45] T. Kohonen, “Logic Principles of Content-Addressable Memories,” in *Content-Addressable Memories*, Springer, 1980, pp. 125–189.

[46] V. Cherkassky, N. Vassilas, G. L. Brodt, and H. Wechsler, “Conventional and associative memory approaches to automatic spelling correction,” *Eng. Appl. Artif. Intell.*, vol. 5, no.

3, pp. 223–237, 1992.

[47] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JAsIs*, vol. 41, no. 6, pp. 391–407, 1990.

[48] M. A. Bickel, "Automatic correction to misspelled names: a fourth-generation language approach," *Commun. ACM*, vol. 30, no. 3, pp. 224–228, 1987.

[49] W. W. Bledsoe and I. Browning, *Pattern recognition and reading by machine*. PGEC, 1959.

[50] S. Kahan, T. Pavlidis, and H. S. Baird, "On the recognition of printed characters of any font and size," *Pattern Anal. Mach. Intell. IEEE Trans.*, no. 2, pp. 274–288, 1987.

[51] B. T. Oshika, B. Evans, F. Machi, and J. Tom, "Computational techniques for improved name search," in *Proceedings of the second conference on Applied natural language processing*, 1988, pp. 203–210.

[52] R. M. K. Sinha and B. Prasada, "Visual text recognition through contextual processing," *Pattern Recognit.*, vol. 21, no. 5, pp. 463–479, 1988.

[53] M. A. Jones, G. A. Story, and B. W. Ballard, "Integrating multiple knowledge sources in a Bayesian OCR post-processor," *ICDAR-91*, pp. 925–933, 1991.

[54] R. L. Kashyap and B. J. Oommen, "Spelling correction using probabilistic methods," *Pattern Recognit. Lett.*, vol. 2, no. 3, pp. 147–154, 1984.

[55] M. D. Kernighan, "Specialized spelling correction for a TDD system AT & T Bell Labs Tech," *Mere., August*, vol. 30, 1991.

[56] K. W. Church and W. A. Gale, "Probability scoring for spelling correction," *Stat. Comput.*, vol. 1, no. 2, pp. 93–103, 1991.

[57] P. L. Troy, "Combining probabilistic sources with lexical distance measure for spelling correction," in *Bellcore Tech 1990*, 1990.

[58] D. J. Burr, "Experiments with a connectionist text reader," in *IEEE First International Conference on Neural Networks*, 1987, vol. 4, pp. 717–724.

[59] K. Kukich, "Variations on a back-propagation name recognition net," in *Proc. Advanced Technology Conference, US Postal Service, Wash. DC, USA*, 1988, pp. 722–735.

[60] K. Kukich, "Backpropagation topologies for sequence generation," in *Neural Networks, 1988., IEEE International Conference on*, 1988, pp. 301–308.

[61] R. Deffner, K. Eder, and H. Geiger, "Word recognition as a first step towards natural language processing with artificial neural networks," in *Konnektionismus in Artificial Intelligence und Kognitionsforschung*, Springer, 1990, pp. 221–225.

[62] T. Mosavi Miangah, "Constructing a large-scale english-persian parallel corpus," *Meta J. des traducteursMeta/Translators' J.*, vol. 54, no. 1, pp. 181–188, 2009.

[63] P. B. Kantor and E. M. Voorhees, "The TREC-5 confusion track: Comparing retrieval methods for scanned text," *Inf. Retr. Boston.*, vol. 2, no. 2–3, pp. 165–176, 2000.

[64] H. de Medeiros Caseli and M. das G. V. Nunes, "Evaluation of sentence alignment methods on portuguese-english parallel texts," *Scientia*, vol. 14, no. 2, pp. 1–14, 2003.

[65] A. Fraser, "Improved Unsupervised Sentence Alignment for Symmetrical and

Asymmetrical Parallel Corpora," *Coling*, no. August, pp. 81–89, 2010.

[66] H. Eghbalzadeh, B. Hosseini, S. Khadivi, and A. Khodabakhsh, "Persica: A Persian corpus for multi-purpose text mining and Natural language processing," in *Telecommunications (IST), 2012 Sixth International Symposium on*, 2012, pp. 1207–1214.

[67] M. T. Pilevar, H. Faili, and A. H. Pilevar, "Tep: Tehran english-persian parallel corpus," in *Computational Linguistics and Intelligent Text Processing*, Springer, 2011, pp. 68–79.

[۶۸] ع. صادقی و ز. زندی مقدم، " فرهنگ املایی خط فارسی"، فرهنگستان زبان و ادب فارسی، ۱۳۸۵.

۶-پیوست ۱ –نمونه داده‌های پیکره استاندارد برچسب‌دار املائی فارسی

<table>
<tr><th>داده</th><th>شـــرح</th></tr>
<tr><td>برنامه کارگاه آموزش عملی دانشکده علوم دانشگاه تهران از ۱۲-۱۵ اردیبهشت چجاری به موضوا ژن کلونینگ اختصاص یافت. این بحث با عناوین ترانسفورماسیون در سلمل می‌زبان، استخراج پلاسمید، جددسازی قطعه موردنظر و شناسایی، ارائه می‌شود.</td><td>نمونه داده از بانک خبری (پرسیکا)
شماره سند = ۴۸۲۵</td></tr>
<tr><td><table><tr><th>ERR_word</th><th>Correct_Word</th><th>Index_Wrd</th></tr><tr><td>چجاری</td><td>جاری</td><td>۱۳</td></tr><tr><td>موضوا</td><td>موضوع</td><td>۱۵</td></tr><tr><td>سلمل</td><td>سلول</td><td>۲۷</td></tr></table></td><td>جدول لیبل سند ۴۸۲۵</td></tr>
</table>

<table>
<tr><th>داده</th><th>شـــرح</th></tr>
<tr><td>در اثر دگرگونی‌هایی که درسطح جهانی وئلی در جوامع انسانی رخداده، نقش زئان در نهاد خائواده و سازمان‌های اجتماعی، تغییرات قابل‌توجهی پیدا کرده‌است. در اثر پیدایش ارزش‌ها و نگرش‌های جدید درمیان زنان، مقاومت‌هایی ازسوی آنها، چه در عرصهٔ خانواده وچه در عرصه‌های مختلف اجتماعی، به‌علت گسترش وسایل ارتباط‌جمعی وغفراوانی منابع ارائه‌دهنده اطلاعات، به‌صورت مبارزات اجتماعی پدید</td><td>سند علمی از پیکره چکیده مقالات علمی پژوهشی
شماره سند=۳۸۰</td></tr>
</table>

آمده، که به بطازتولید ارزش‌های معفاوت در باب هویت فغردی وجنسیتی منجر شده‌است. این مقاله در پی بررسی تأثیر عوامل متفاوت اجتمادعی بر هویت جنسیتز زنات است. هدف ازاین بررسی ارزیدبی چگونگی درک زنان از جنسیت وهویتشان، که متأثر از عوامل پوناگون اجتماعی است، باشد. دراین کار از بررسی دیدگاه‌های نظری گونلگون در حوزه‌های جامعه‌شناختی، روان‌شناختی اجتماعی و فمنیستی، دیدگاه محقق چدید آمده، که با اتکاء به‌آن، کار تلحقیق دنبال‌شده است. برای بررسی موضوع از روش‌های اسنادی و پیمایشی ومطالعهٔ میدانی و مصاحیه استفتاده‌شده است. جامعهٔ موردنظر کلیه زنان ۴۰-۲۰ ساله مراجعه‌کننده به کتابخانه‌های عمومی شهر تهران درنظر گرفته‌شده است. حجم نمونه ۳۳۰ نفر وشیوه نمونه‌گیری، طبقه‌ای دو مرحله‌ای تصادفی شده‌است. تجزیه‌وتحلیل اطلاعات با استفاده‌از نرم‌افزار spss ودر سطوح توصیفی وتبیینی است. . نتایج به‌دست آمده نشان می‌دهد که با افزغیش سن، هویت جنسیتی تزنان بیشتر ثتحت‌تأثیر روابط و مناسبخات پدرسالاری است؛ و با کاهش، آن هوثت زنان بیشتر تحت‌تأثیر روابط و مناسبات سرمایه‌داری می‌باشد. در مدل تحلیل مسیر، در میان زنان ۳۰-۲۰ سال، روابط و مناسبات سرمایه‌داری بالاترین تأثیر، ودر میان زنان ۴۰-۳۰ اسال،	

پدرسالاری بیشترین تأثرر را داشته است.

جدول لیبل سند ۳۸۰

ERR_word	Correct_Word	Index_Wrd
ئلی	ملی	۷
زئان	زنان	۱۴
خائواده	خانواده	۱۷
غفراوانی	فراوانی	۶۱
بطازتولید	بازتولید	۷۴
معفاوت	متفاوت	۷۶
فغردی	فردی	۸۰
اجتمادعی	اجتماعی	۹۳
جنسیتز	جنسیتی	۹۶
زنات	زنان	۹۷
ارزیدبی	ارزیابی	۱۰۳
پوناگون	گوناگون	۱۱۶
گونلگون	گوناگون	۱۲۸
چدید	پدید	۱۴۲
تلحقیق	تحقیق	۱۵۱
مصاحیه	مصاحبه	۱۶۷
استفتاده‌شده	استفاده‌شده	۱۶۸
افزغیش	افزایش	۲۲۵
تزنان	زنان	۲۳۰
ثتحت‌تأثیر	تحت‌تأثیر	۲۳۲
مناسبخات	مناسبات	۲۳۵
هوثت	هویت	۲۴۳
اسال	سال	۲۸۱