تصحیح خودکار غلط‌های تایپی فارسی به‌کمک شبکهٔ عصبی مصنوعی ترکیبی

امیرشهاب شاهمیری[۱] **رضا صفابخش**[۲] **رسول دژکام**[۳]

۱- دانش‌آموخته کارشناسی ارشد دانشکده مهندسی کامپیوتر- دانشگاه صنعتی امیرکبیر - تهران- ایران
amir@shahmiri.ir

۲- استاد دانشکده مهندسی کامپیوتر- دانشگاه صنعتی امیرکبیر- تهران- ایران
safa@ce.aut.ac.ir

۳- دانش‌آموخته کارشناسی ارشد دانشکده مهندسی کامپیوتر- دانشگاه صنعتی امیرکبیر- تهران- ایران
dezhkam@ce.aut.ac.ir

چکیده: ارایهٔ راهی برای تصحیح غلط‌های املایی نگاشته شده توسط انسان یکی از اهداف مورد توجه در دانش هوش مصنوعی، متن کاوی و پردازش زبان طبیعی است. بیشتر روش‌های موجود برای تصحیح غلط‌های املایی بر پایهٔ الگوریتم‌های جست‌وجو در فرهنگ واژگان و تعیین نسبت شباهت واژگان درست موجود در فرهنگ واژگان با واژهٔ نادرست مورد نظر کار می‌کنند.

در این پژوهش طراحی، پیاده‌سازی و ارزیابی یک مصحح املایی به‌کمک شبکه‌های عصبی مصنوعی هاپفیلد و پرسپترون چند لایه با رویکرد ویژه به غلط‌های تایپی کاربر ارایه می‌شود. نتایج به‌دست آمده نشان می‌دهند که برای یادگیری واژه‌نامه‌ای مشتمل بر ۴ تا ۲۵۶ واژهٔ ۴ تا ۶ حرفی و تصحیح غلط‌های مربوط به آنها، شبکهٔ هاپفیلد به دقتی بین ۵۵٪ و ۱۰۰٪ درستی و شبکهٔ پرسپترون چندلایه - که در این تحقیق عمل یادگیری را در قالب دسته‌بندی انجام می‌دهد - به دقتی بین۸۰٪ و ۱۰۰٪ درستی دست یافته، که این مقدار با به‌کارگیری شبکه‌های ترکیبی به نزدیک به۸۰٪ دقت درستی برای بیش از ۳۰۰۰ واژه افزایش یافته است.

واژه های کلیدی: صحیح خودکار غلط تایپی فارسی، غلط املایی، شبکهٔ عصبی مصنوعی هاپفیلد، پرسپترون چند لایه، فاصله کلید.

تاریخ ارسال: مقاله : ۱۳۸۶/۸/۴

تاریخ پذیرش مقاله : ۱۳۸۷/۱۰/۲

نام نویسنده‌ی مسئول : رضا صفابخش

نشانی نویسنده‌ی مسئول : ایران – تهران – خیابان حافظ – پلاک ۴۲۴ – دانشگاه صنعتی امیر کبیر – دانشکده‌ی مهندسی کامپیوتر و فناوری اطلاعات

۱- مقدمه

گفت‌وگوی معنی‌دار میان انسان و ماشین یکی از آرزوهای دانشمندان علوم رایانه و یکی از مباحث مورد توجه در زمینهٔ هوش مصنوعی است. در این زمینه، درک جملهٔ بیان شده توسط انسان به‌زبان طبیعی از سوی ماشین و نیز تولید جملهٔ درست و با معنی توسط ماشین، از مهم‌ترین اهداف علوم پردازش و فهم زبان طبیعی[۱] و هوش مصنوعی است که هرچند تاکنون به موفقیت‌های چشمگیری دست نیافته، اما افق‌های روشنی را پیش روی پژوهشگران این عرصه قرار داده است.

مسالهٔ مشابه دیگری که محققان هوش مصنوعی و متن‌کاوی[۲] بدان توجه دارند، ویرایش یا درک متن نوشته شده به‌دست انسان، توسط ماشین است که معمولا سه گام زیر را در بر می‌گیرد[۱]:

ویرایش لغوی[۳]: که در آن کوشش می‌شود تا در صورت اشتباه بودن نگارش املایی یک واژه، جایگزین مناسبی به‌کمک یک پایگاه دادهٔ جامع از واژگان برای آن پیدا شود. برای نمونه به‌جای واژهٔ "ویزایش" واژهٔ "ویرایش" پیشنهاد گردد.

ویرایش دستوری[۴]: که در آن کوشش می‌شود تا اشتباهات دستوری جملات شناسایی و تصحیح گردد. برای نمونه، رخداد زمانی میان فعل و قید جملهٔ "دیروز من و او یکدیگر را دید." کشف گردد و جمله‌ای مانند "دیروز من و او یکدیگر را دیدیم." پیشنهاد گردد. این امر به‌کمک مجموعهٔ قواعد دستورزبانی و روش‌های گوناگون تجزیهٔ واژگان جمله در مبحث پردازش و فهم زبان طبیعی به نتایجی رضایت‌بخش رسیده است.

ویرایش مفهومی[۵]: که در آن کوشش می‌شود تا به‌رغم درستی ساختار جمله از نظر دستورزبان، ناهماهنگی‌های مفهومی آن دست‌کم شناخته و تا جای ممکن تصحیح گردد. برای نمونه جملهٔ "میز هواپیما را خورد." از دیدگاه املایی و دستوری اشتباهی ندارد، اما مفهوم درستی از آن بر نمی‌آید.

روشن است که در هر گونه پردازش متن نخستین گام، تصحیح غلط‌های واژگان متن است. غلط‌های واژه‌ای یک متن به دو دسته تقسیم می‌شوند: یکی آن‌که واژهٔ مورد نظر در واژه‌نامه موجود است، اما به‌دلیل ضعف دانش زبانی نگارنده یا اشتباه وی، با توجه به مفهوم جمله نابه‌جا نوشته شده است (مانند واژهٔ "*غریب*" در جملهٔ "*من در این شهر قریب هستم*"). این‌گونه خطاها در زبان فارسی کمیاب است و بیشتر در واژه‌های دخیل از زبان‌های دیگر، به‌ویژه زبان عربی، رخ می‌دهد و در زبان‌های غیرآوایی مانند انگلیسی نیز بسیار دیده می‌شود (مانند "*piece*" در "*a peace of cake*")[۲]. این دسته از اشتباهات از جمله خطاهای ابهام گرامری[۶] و تشابه آوایی[۷] هستند[۳] و در محدودهٔ این پژوهش نمی‌گنجند، زیرا تنها با تحلیل معنا و مفهوم جمله می‌توان آنها را کشف کرد. دستهٔ دوم غلط‌ها نیز واژگانی هستند که در واژه‌نامه موجود نیستند و از این‌رو غلط به‌شمار می‌آیند.

به‌طور کلی خطاها و غلط‌های متون و مستندات را می‌توان به سه دسته تقسیم کرد[۴]:

غلط تایپی[۸]، که چهار گروه اصلی درج حرف اضافه[۹]، حذف حرف[۱۰]، نگارش یک حرف اشتباه به‌جای حرف اصلی[۱۱] و جابه‌جایی دو حرف همسایه[۱۲] را در بر دارد[۳و۵]. این چهار گونه، نزدیک به ۸۰٪ خطاهای تایپی را در بر می‌گیرند[۶].

غلط املایی[۱۳]، که ناشی از ضعف دانش زبانی نویسنده در تشخیص اصوات واژه و شیوهٔ نگارش آن است.

خطای انتقال[۱۴]، که در هنگام انتقال اطلاعات بر شبکه یا دیسک و بازشناسی نوری حروف[۱۵] رخ می‌دهد.

برای پیاده‌سازی یک سیستم مصحح املایی، دو کار اصلی باید انجام پذیرد: نخست باید واژهٔ مورد نظر در فرهنگ واژگان جست‌وجو شود تا در صورت موجود نبودن غلط قلمداد گردد؛ سپس با استفاده از روش‌های گوناگون در حالت خودکار[۱۶] یک، یا در حالت محاوره‌ای[۱۷] چند واژه به‌عنوان جایگزین پیشنهاد گردد[۷]. از آنجا که بسیاری از واژگان با تغییر یک حرف به واژه‌ای دیگر تبدیل می‌شوند، هیچگاه نمی‌توان انتظار داشت که دقت تصحیح لغوی[۱۸] به صد درصد برسد و دستیابی بهترین روش‌های مصحح به این دقت نیز در عمل - حتی با توجه به درک مفهوم جملات بر پایهٔ روش‌های پردازش زبان طبیعی - برای تصحیح تمامی واژگان یک زبان امکان‌ناپذیر خواهد بود. گذشته از این، با توجه به این‌که بیشتر روش‌های تصحیح غلط بر روی تنها یک خطا در واژه خوب کار می‌کنند، تا زمانی که روش‌های تصحیح خطاهای ترکیبی مانند جابه‌جایی رشته[۱۹] و وارونه نویسی[۲۰] به‌میان نیایند، نمی‌توان به آنها اطمینان چندانی داشت[۴]. از این‌رو به‌نظر می‌رسد که دقت درستی ۹۰٪ برای یک مصحح خودکار مطلوب باشد؛ در حالی‌که این مقدار برای انسان ۷۵٪ است[۸]. از این‌رو این‌گونه غلط‌ها به‌ندرت در گروه واژگان آزمایشی جای می‌گیرند[۵و۹].

اغلب روش‌هایی که تاکنون برای تصحیح لغوی ارایه شده‌اند، تطبیق رشته حروف[۲۱] واژهٔ غلط[۲۲] را - که در فرهنگ واژگان موجود نیست - با نزدیک‌ترین واژه در فرهنگ واژگان، بر پایهٔ فاصلهٔ ویرایشی[۲۳]، فاصلهٔ همینگ[۲۴] یا فاصلهٔ لونشتین[۲۵] به‌کار می‌گیرند و یک یا چند واژهٔ جایگزین را پیشنهاد می‌دهند[۳-۹]. این فاصلهٔ ویرایشی بنا بر اختلافات تک‌تک حروف واژهٔ غلط با کلمات فرهنگ واژگان به‌دست می‌آید و از این‌رو اغلب - به‌ویژه در هنگام ویرایش واژگان کم‌حرف - نتیجهٔ دلخواهی به‌همراه ندارد. برای نمونه، اگر به‌جای واژهٔ "*word*" رشتهٔ "*wprd*" تایپ شود، این روش ۷۱۰ واژهٔ جایگزین هم‌سطح را پیشنهاد می‌دهد[۵].

هدف این پژوهش ارایهٔ روشی جدید برای ویرایش لغوی متون با رویکرد ویژه به غلط‌های تایپی است. دلیل این انتخاب آن بوده که متونی که توسط کاربر تایپ می‌شوند، خود پیشتر توسط نویسنده (که ممکن است خود کاربر باشد) و با آگاهی وی از ساختار جملات و به‌ویژه مفهوم آنها شکل گرفته‌اند و بنابراین نیاز چندانی به ویرایش دستوری و مفهومی ندارند و حتی ممکن است به‌دلایل ویژه، به‌عمد

نکاتی در آنها درج شده باشد که از دیدگاه دستوری یا مفهومی اشتباه به‌نظر برسد و بنابراین اعمال تغییرات بر آنها خود موجب پیش آمدن اشتباهات بیشتر گردد. از سوی دیگر، با افزایش سواد و دانش عمومی و نیز با گسترش کاربرد رایانه در جهان، امروزه دیگر مشکل غلط املایی رو به‌کاهش و در عوض غلط تایپی رو به افزایش است.

۱-۱- پیشینهٔ پژوهش

"فوروگوری" در سال ۱۹۹۰ با توجه به تفاوت ماهیت غلط‌های املایی انگلیسی غیرانگلیسی‌زبانان و با بررسی تفاوت‌های آوایی زبان ژاپنی و انگلیسی، سیستمی کمکی برای تصحیح غلط‌های املایی ژاپنی‌ها در هنگام نگارش به‌زبان انگلیسی پیشنهاد کرد که می‌توانست دقت نرم‌افزار تصحیح املایی کرکت‌استار[۲۶] را از ۶۰٪ به ۷۵٪ برساند[۱۰]. در سال ۱۹۹۶ "شانگ" و "مرتال" با بررسی چند الگوریتم دیگر، روش ترای[۲۷] را برای تخمین نزدیکی دو رشته حروف بهبود و گسترش دادند[۱۱]. "لاونیه" در سال ۱۹۹۲ توانست تراشه‌ای در قالب آرایه‌ای ۲-بعدی از ۶۹ پردازنده را به‌همراه یک تکنیک برنامه‌نویسی پویا برای تصحیح مقایسه‌ای رشته حروف بسازد که می‌توانست ۲۰۰ هزار واژه را در هر ثانیه تصحیح کند[۶]. "چاودوری" در سال ۲۰۰۲ با توجه به ویژگی‌های آوایی و غیرآوایی زبان هندی و با به‌کارگیری برخی از تکنیک‌ها - مانند روش n-گرام[۲۸] - توانست سیستم مصحح املایی برای زبان هندی و بنگلایی[۲۹] ارایه کند که به دقت درستی ۹۵٪ دست یافت[۱۲]. در سال ۲۰۰۰ "هاج" و "آوستین" توانستند با ترکیب روش n-گرام و رویکرد فاصلهٔ همینگ سیستمی را فراهم آورند که برای یک پایگاه با تعداد محدودی از واژگان و با سرعتی قابل قبول، هر چهار گونه از غلط املایی را با دقتی تا ۹۷/۵٪ تصحیح کند که به‌طور متوسط ۸ واژهٔ جایگزین را پیشنهاد می‌کرد[۵]. در سال ۲۰۰۰ "لی" و همکارانش توانستند یک روش تطابق تخمینی واژه فازی[۳۰] را در قالب مصحح املایی اختصاصی برای زبان چینی ارایه کنند که علاوه بر غلط‌های رایج، جابه‌جایی رشته را نیز تصحیح می‌کرد[۱۳]. "راش" و همکارانش در سال ۲۰۰۱ توانستند با ترکیبی از روش‌ها - مانند تطابق رشته به رشته و مدل مخفی مارکف - در تصحیح واژگان متون پزشکی به دقت درستی ۹۸٪ دست یابند[۲و۳]. "هوانگ" و "پاورز" در سال ۲۰۰۱ با ترکیبی از روش‌ها و با در نظر گرفتن برخی غلط‌های تایپی توانستند در متون حجیم به دقت تصحیح ۷۴٪ دست یابند[۱۴].

"چرکاسکی" و همکارانش در سال ۱۹۹۰ توانستند با ترکیب برخی از شبکه‌های عصبی یادآور[۳۱] مانند شبکهٔ هاپفیلد[۳۲] و شبکه‌های پس‌انتشار[۳۳] با دیگر روش‌ها، سیستم غلط‌یاب املایی برای واژگان کوتاه (۵ تا ۷ حرفی) و بلند (۱۰ تا ۱۲ حرفی) با تعداد گره‌های ورودی برابر با توانی از ۲۶ (به‌تعداد حروف الفبای انگلیسی) به مقدار n در الگوریتم n-گرام و گره‌های خروجی به‌تعداد واژه‌های ذخیره شده، بسازند و به دقت ۱۵ تا ۱۰۰ درصد برای انواع خطا و مقادیر n دست یابند[۴]. "گارفینکل" و همکارانش در سال ۲۰۰۲ کوشیدند تا با شناخت و ترکیب روش‌های مبتنی بر فاصلهٔ واژگان برای انواع خطاهای احتمالی در متن، تعاریفی سیستماتیک را همراه با الگوریتم‌های تصحیح ارایه دهند[۷].

۱-۲- ویژگی‌های مساله

در مسالهٔ غلط‌های املایی، احتمال پیش آمدن غلط تایپی ناشی از اشتباه کاربر، بیشتر از غلط‌های املایی متاثر از ضعف دانش زبانی نویسندهٔ اصلی است؛ زیرا معقول‌تر آن است که فرض کنیم، کسی که خود متنی را به‌کمک رایانه تایپ می‌کند یا آن‌را به حروفچین می‌سپارد، به اندازه‌ای تسلط بر واژگان دارد که غلط‌های املایی متن وی ناچیز باشد، اما از سوی دیگر رخداد غلط تایپی امری رایج در هنگام ماشینی کردن متون است. این دو گونه اشتباه، ماهیتی متفاوت دارند و شناخت و استفاده از ویژگی‌های آنها می‌تواند به ما در تصحیح املایی متون کمک کند. این نکته مساله‌ای است که در روش‌های کلاسیک و نرم‌افزارهای تجاری تصحیح غلط املایی - که اغلب بر پایهٔ الگوریتم‌های جست‌وجو استوارند - بدان توجه چندانی نشده و از این‌رو یکی از امتیازات این تحقیق به‌شمار می‌رود. البته باید در نظر داشت که نرم‌افزارهای تجاری از آن‌رو بر پایهٔ روش‌های جست‌وجو در فرهنگ واژگان[۳۴] کار می‌کنند که هدف آنها دستیابی سریع به پاسخ با کمترین حافظهٔ مصرفی است؛ در حالی که روش‌های آکادمیک بیشتر به دستیابی به بهترین پاسخ می‌اندیشند[۴].

از آنجا که جست‌وجو در پایگاه داده‌ای با چند صد هزار واژه کاری وقت‌گیر است، روش‌های تصحیح غلط املایی چهار راهبرد اساسی را برای کاهش تعداد جست‌وجوها به‌کار می‌گیرند که آنها را روش‌های تطبیق کامل[۳۵] می‌نامند[۴]:

بر پایهٔ تعداد دفعات تکرار: بدین ترتیب که واژگانی که بیشترین مورد استفاده را در متون دارند، در یک فرهنگ واژگان کوچکتر گردآوری شده، مصحح در هنگام یافتن غلط، نخست این پایگاه را می‌جوید و در صورتی که واژه یافته نشد، پایگاه‌های واژگان کم‌استفاده‌تر در اولویت‌های بعدی را نیز ارزیابی می‌کند.

بر پایهٔ طول واژه: فرهنگ واژگان به فرهنگ‌های ۲ تا n حرفی بخش می‌شود و در هنگام برخورد با واژهٔ غلط k حرفی، نخست فرهنگ k حرفی و سپس در صورت نیافتن واژهٔ صحیح، فرهنگ‌های k+1 و k-1 حرفی جست‌وجو می‌گردد.

بر پایهٔ نخستین حرف واژه‌نامه با ساختار درختی: در این ساختار، ریشه به تعداد حروف خط زبان مورد نظر (انگلیسی ۲۶ و فارسی ۳۲) گره دارد و هر گره نیز فرزندانی تا همین تعداد دارد. بنابراین برای یافتن واژهٔ k حرفی به k جست‌وجو نیاز است. اما روشن است که این روش زمانی خوب کار می‌کند که حروف آغازین واژه درست نگاشته شده باشند.

بر پایهٔ فشرده‌سازی فرهنگ واژگان: با منظور کردن این نکته که بسیاری از واژگان یک زبان ریشه‌ای مشترک دارند، می‌توان فرهنگ را به این واژگان ریشه‌ای محدود ساخت و در عوض قوانینی برای تولید واژگان

ممکن بدان افزود. روش فشرده‌سازی فرهنگ، به‌دلیل ویژگی‌های واژه‌سازی زبانی، در زبان عربی بهتر از فارسی و انگلیسی کار خواهد کرد.

در مجموع می‌توان ویژگی‌های روش‌های تصحیح کلاسیک مبتنی بر جست‌وجو و مبتنی بر شبکهٔ عصبی را چنین بیان کرد:

- با وجود این‌که شبکهٔ عصبی در فاز یادگیری به زمان بالایی نیاز دارد، اما در فاز آزمایش به‌سرعت نتیجه می‌گیرد و بنابراین از هر روش دیگری سریع‌تر است.
- شبکه‌های عصبی مصنوعی ظرفیتی محدود دارند؛ در نتیجه برای یادگیری تعداد زیادی واژه که بتواند حداقل نیاز برای مصحح املایی را برآورده سازد، با مشکل کاهش دقت روبرو می‌شوند.
- یافتن بهترین شبکه با تعداد لایه و نرون، خود مساله‌ای نسبی و تجربی است و تصمیم‌گیری در مورد آن دقیق نیست. از این‌رو مساله باید با شبکه‌های گوناگون آزموده شود، تا بهترین ترکیب به‌دست آید.
- با بهتر شدن و پیدایش انواع تازه و پرتوان‌تر شبکه‌های عصبی در آینده، نتایج کار نیز بهبود خواهند یافت.
- در شبکه‌های عصبی با ساختار و طراحی ساده، با ورود یک واژه (درست یا نادرست) خروجی در نهایت یک واژه است، اما در روش‌های کلاسیک می‌توان بنا بر نسبت شباهت واژهٔ نادرست به واژگان درست در فرهنگ واژگان، تعدادی واژهٔ جایگزین را پیشنهاد کرد.
- در صورتی که واژه‌ای بدون بررسی درست یا نادرست بودنش (بنا بر بود یا نبود آن واژه در فرهنگ واژگان) به شبکهٔ عصبی وارد گردد، به‌دلیل ماهیت شبکه‌های عصبی که همواره درصدی از خطا را به‌همراه دارد، ممکن است که واژهٔ درست هم به واژه‌ای نادرست نگاشت شود.

در ادامهٔ این مقاله نخست در بخش ۲ به تعریف و شناخت ماهیت غلط تایپی می‌پردازیم و تفاوت‌های آن با غلط املایی در زبان فارسی را بررسی خواهیم کرد. سپس در بخش ۳ به روش‌های ارایه شده برای تصحیح این اشتباهات خواهیم پرداخت و پس از آن در بخش ۴ نتایج تصحیح را بررسی خواهیم نمود و سرانجام در بخش ۵ مزایا و معایب جمع‌بندی و اهداف آینده بیان خواهد شد.

۲- غلط تایپی

همان‌گونه که در مقدمه گفته شد، در هنگام مواجهه با غلط‌های املایی، بهتر است فرض کنیم که غلط موجود ناشی از اشتباه تایپی بوده است و نه ضعف دانش زبانی نویسنده؛ زیرا کسی که متنی را به‌کمک رایانه تایپ می‌کند یا آن‌را به حروف‌چین می‌سپارد، به اندازهٔ کافی بر زبان و واژگانش آشنایی دارد که غلط‌های املایی متن وی ناچیز باشد، اما از سوی دیگر رخداد غلط تایپی در هنگام تایپ متن بسیار رایج است.

۲-۱- غلط املایی در زبان فارسی

غلط املایی در زبان فارسی معمولا به دو شکل پیش می‌آید: یکی این‌که نگارنده یکی از حروف هم صدا را به‌جای دیگری به‌کار برد. این‌گونه اشتباه معمولا در مورد واژه‌های عربی دخیل در فارسی رخ می‌دهد. برای نمونه نویسنده ممکن است در واژهٔ "اضطرار حرف "ض" را با "ز"، "ذ" یا "ظ" و حرف "ط" را با "ت" اشتباه بگیرد و واژهٔ "تقریظ" را "تغریظ"، "تغریض" یا به‌شیوه‌های دیگر بنگارد. این اشتباه عموما در مورد واژه‌های هم آوا در خط فارسی مانند "ت : ط"، "ح : ه"، "ز : ذ : ض : ظ"، "ق : غ" و گاهی نیز "ا : ع" (مانند مواخذه : معاخذه) رخ می‌دهد.

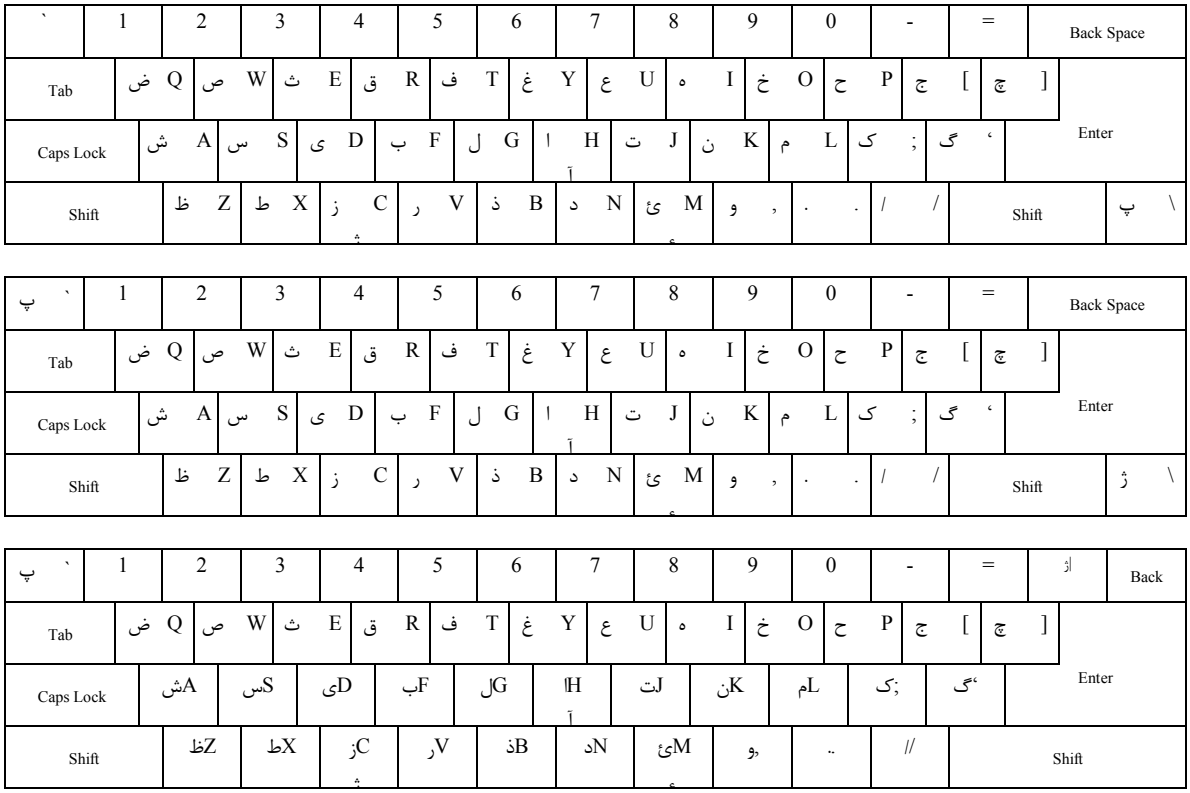
دوم اشتباه در نگارش حروف صدادار (مصوت) یا آوای واژگان است که آن هم بیشتر در مورد واژگان دخیل از زبان‌های اروپایی - مانند انگلیسی و فرانسه - و نیز زبان عربی رخ می‌رهد. برای نمونه برای واژگان "*Flout*"، "*Float*" و "*Flat*" دو نگارش "فلوت" و "فلت" را به‌کار می‌برند یا واژهٔ "*Robot*" را "ربات"، "روبات" و یا "ربوت" و یا "مسئله" از عربی را "مساله"، "مسأله" یا به‌شیوه‌های دیگر می‌نویسند. هر دوی این اشتباهات اغلب برای نویسندگان کم سن و سال (تا مقطع دبستان و راهنمایی) پیش می‌آید و با افزایش سن و تجربه کم‌کم از بین می‌رود.

۲-۲- غلط تایپی در زبان فارسی

اما غلط تایپی به سواد نویسنده بستگی ندارد، بلکه وابسته به مهارت و دقت حروف‌چین و موقعیت کلیدهای صفحه‌کلید است. شکل ۱، سه نمونه از موقعیت‌های حروف بر انواع صفحه‌کلید را نشان می‌دهد. بخشی از این ناهمگونی‌ها به‌دلیل تغییرات پیاپی در استانداردها و تعداد دکمه‌های صفحه‌کلید رخ داده و بخشی دیگر، به‌علت نبود اتفاق‌نظر میان استانداردگذاران درون و حتی بیرون از کشور بر سر تعیین جایگاه حروف فارسی بر دکمه‌ها پدید آمده و این ناهماهنگی‌ها موجب افزایش خطاهای تایپی میان کاربران فارسی‌زبان گشته است.

در این زمینه برای شناخت بهتر ماهیت غلط تایپی فارسی با چند فرد خبره در حروف‌چینی گفت‌وگو شد و اطلاعات آنان به شکل زیر طبقه‌بندی گردید:

- اشتباه در تایپ یک واژه، معمولا بین کلیدهای همسایه رخ می‌دهد. مانند: "ج" و "ح"، "ن" و "ت" یا "ی" و "س".
- اشتباه تایپی اغلب بین کلیدهای یک سطر از صفحه کلید رخ می‌دهد و اشتباه با کلیدهای سطر بالا یا پایین کم پیش می‌آید. برای نمونه حرف "ه" با "خ" یا "ع" اشتباه گرفته می‌شود، اما اشتباه آن با "ت" یا "ن" که در همسایگی سطر پایین‌تر قرار دارد، بسیار کمیاب است.
- به‌دلیل استاندارد نبودن جای برخی از حروف بر صفحه کلید، برخی از حروف که نه شباهت آوایی با هم دارند و نه همسایه نزدیک یکدیگر هستند، با هم اشتباه می‌شوند. مانند "پ" و "ژ".



شکل (۱): سه نمونه از جانشانی حروف فارسی بر صفحه کلید.

- ارتباطی ناچیز هم در اشتباه بین حروف هم‌صدا وجود دارد.۳۶ مانند: "ض" و "ظ" یا "غ" و "ق".
- گاه در حروفی که نیاز به فشردن کلید شیفت دارند، آن حرف با حرف اصلی جابه‌جا می‌شود. مانند "ا" به‌جای "آ" (که البته این مورد غلط املایی به‌شمار نمی‌رود)، یا "ی" و "ط"، "ز" به‌جای "ژ" (در برخی از صفحه کلیدها) و "ی"، "س" و "ش" به‌جای هر یک از اعراب.
- گاه دو حرف از یک واژه جابه‌جا تایپ می‌شود، مانند: "زیبا : زبیا". این اتفاق نیز بیشتر در مورد حروف همسایه رخ می‌دهد.
- گاه غلط املایی بدین شکل رخ می‌دهد که یکی از حروف واژه زده نمی‌شود یا یکی از کلیدهای جانبی یا خود کلید یکی از حروف واژه، اضافه زده می‌شود. مانند: "هشدار : هششدار : هشار".

از این‌رو انواع غلط‌های تایپی زبان فارسی به‌شرح زیر دسته‌بندی شد:

درج حرف: در هنگام نگارش، یک حرف اضافه به‌اشتباه تایپ می‌شود. مانند: "کیومرث : کیونمرث".

تکرار حرف (درج حرف همسایه): یعنی در هنگام نگارش، یکی از حروف اصلی به‌اشتباه دو بار تایپ می‌شود. مانند: "کیومرث : کیوممرث". این گونه غلط خود نوعی درج حرف است که بیش از گونه‌های دیگر رخ می‌دهد.

حذف حرف: در هنگام نگارش، یکی از حروف اصلی تایپ نمی‌شود. مانند: "کیومرث : کیمرث".

جابه‌جایی کلید: به‌جای یکی از حروف اصلی، یکی از کلیدهای دیگر بر صفحه کلید فشرده می‌شود. کلید اشتباه اغلب نزدیک به کلید اصلی است. مانند: "کیومرث : کبومرث / گیومرث".

انتقال کلید: یکی از حروف اصلی، با حرف همسایه یا چند حرف قبل یا بعدی جابه‌جا تایپ می‌شود. مانند: "کیومرث : کویمرث / کومریث".

غلط املایی (جابه‌جایی ویژه): به‌جای یکی از حروف اصلی، یکی از دیگر کلیدهای صفحه‌کلید فشرده می‌شود؛ به‌گونه‌ای که در زبان فارسی آن حرف با حرف اصلی هم‌صدا است مانند: "کیومرث : کیومرس". این گونه غلط ناشی از ضعف دانش زبانی حروف‌چین است و بسیار کم رخ می‌دهد، زیرا متن اصلی پیش روی حروف‌چین است.

فاصلهٔ اضافه: یک کاراکتر فاصله نابه‌جا درج می‌گردد. مانند: "کیومرث : کی ومرث". این غلط به‌ویژه در برخی حالت‌های ویژه مانند "ها"ی جمع و "می" استمرار و اغلب به‌جای فاصلهٔ مجازی[۳۷] (کوتاه) رخ می‌دهد. مانند: "واژه‌ها : واژه ها" و "می‌رود : می رود". این غلط گونه‌ای ویژه از غلط‌های درج است.

فاصلهٔ کم: کاراکتر فاصله میان دو واژهٔ جداگانه درج نمی‌گردد. مانند: "کیومرث سیامک : کیومرثسیامک". این غلط گونه‌ای ویژه از غلط‌های حذف است.

بر این پایه با نمونه‌برداری و تخمین آماری غلط‌های تایپی زبان فارسی در هنگام کار حروف‌چین خبره به شرح جدول ۱ به‌دست آمد. برای بررسی صحت و سقم هر یک از موارد این جدول از هر یک از افراد خواسته شد تا متنی را بدون استفاده از کلید بازگشت[۳۸] در هنگام

رخداد خطا، تایپ کنند. نتایج به‌دست آمده در تایید موارد فوق بود و خلاف آن‌را نشان نمی‌داد.

جدول (۱): انواع غلط‌های تایپی و نسبت رخداد آنها در زبان فارسی.

ردیف	نوع غلط	گونهٔ اصلی	درصد
۱	درج حرف	درج	۸
۲	تکرار حرف	درج	۱۴/۵
۳	حذف حرف	حذف	۱۹
۴	جابه‌جایی کلید	جابه‌جایی	۳۹/۵
۵	انتقال کلید	انتقال	۵
۶	غلط املایی	جابه‌جایی	۰/۵
۷	فاصلهٔ افزوده	درج	۲/۵
۸	فاصلهٔ کم	حذف	۵
۹	موارد دیگر	ترکیبی	۶/۵
۱۰	نسبت حروف غلط به حروف درست	-	۱/۹۵
۱۱	نسبت واژگان غلط به واژگان درست	-	۵/۳۱

۲-۳- تصحیح املایی بر پایهٔ فاصلهٔ

امروزه بیشتر نرم‌افزارهای مصحح غلط املایی که به‌صورت تجاری در بازار ارایه می‌شوند، بر پایهٔ فاصله (ویرایشی، همینگ، لونشتین و ...) کار می‌کنند، که البته در بسیاری از موارد - به‌ویژه در هنگام رخداد غلط‌های جابه‌جایی - نتیجهٔ دلچسبی به‌همراه ندارند. در این میان، مصحح املایی نرم‌افزار ورد از شرکت مایکروسافت[۳۹] یکی از کارآمدترین و پرطرفدارترین تصحیح‌کننده‌های املایی متن را ارایه کرده است. با وجود آن‌که این نرم‌افزار خود برای تایپ و صفحه‌بندی طراحی شده، به‌غلط تایپی توجهی ندارد، بلکه احتمالا با توجه به پایگاه داده، تک‌تک واژگان تایپ شده را با آن پایگاه داده می‌سنجد و در صورتی که آن واژه در پایگاه موجود نباشد، آن‌را مشخص کرده و به‌کمک برخی قوانین کلاسیک برای یافتن نزدیکترین واژه، چند واژه را به‌ترتیب اولویت از روی سازگاری به کاربر پیشنهاد می‌دهد.

برای نمونه، در صورتی که کاربر بخواهد واژهٔ "*golf*" را تایپ کند، اما به اشتباه به‌جای حرف "*g*"، حرف کناری آن "*h*" و بنابراین "*holf*" را تایپ کند، نرم‌افزار ورد واژگان زیر را به‌ترتیب اولویت پیشنهاد می‌دهد:

"*half*": در اینجا "*a*" به‌جای "*o*" آمده در صورتی که احتمال اشتباه تایپی این دو به دلیل دوری از هم بسیار ناچیز است.

"*hoof*": در اینجا "*o*" به‌جای "*l*" آمده در صورتی که این دو حرف در دو سطر جدا قرار دارند و احتمال اشتباه تایپی این دو به دلیل دوری از هم بسیار ناچیز است.

"*hold*": در اینجا "*d*" به‌جای "*f*" آمده و این دو حرف بر کلیدهای مجاور قرار دارند و این انتخاب معقول‌تر به‌نظر می‌رسد.

"*hole*": در این مورد هم "*e*" به‌جای "*f*" آمده، در صورتی که این دو حرف در دو سطر جدا قراردارند و احتمال اشتباه تایپی این دو ناچیز است.

"*holy*" در اینجا "*y*" به‌جای "*f*" آمده در صورتی که این دو حرف در دو سطر جدا قرار دارند و احتمال اشتباه تایپی این دو به دلیل دوری از هم بسیار ناچیز است.

همان‌گونه که مشاهده می‌شود، در این نرم‌افزار واژهٔ "*golf*" اصلا پیشنهاد نشده و بهترین پیشنهاد در اولویت سوم قرار دارد.

به‌عنوان نمونه‌ای دیگر، در صورتی که کاربر بخواهد واژهٔ "*sear*" را تایپ کند، اما به اشتباه به‌جای حرف "*s*"، حرف کناری آن یعنی "*a*" و بنابراین "*aear*" را تایپ کند، نرم‌افزار ورد واژگان زیر را به‌ترتیب اولویت پیشنهاد می‌دهد:

"*area*": در اینجا نرم‌افزار، حرف "*r*" دو خانه به‌عقب برده و فرض کرده که کاربر جای آن را دو خانه جابه‌جا تایپ کرده که البته این اشتباه چندان محتمل نیست.

"*air*": در اینجا فرض شده که کاربر به‌جای حرف "*i*" دو حرف "*ea*" را به‌اشتباه تایپ کرده و البته این اشتباه هم چندان منطقی نیست.

"*afar*": در اینجا "*f*" به‌جای "*e*" آمده در صورتی که این دو حرف در دو سطر جدا قراردارند و احتمال اشتباه تایپی این دو، به‌دلیل دوری از هم، ناچیز است.

"*ajar*": در اینجا هم مانند مورد قبلی حرف "*j*" به‌جای "*e*" آمده در صورتی که این دو حرف در دو سطر جدا قرار دارند و احتمال اشتباه تایپی این دو اندک است.

"*agar*" باز هم در اینجا حرف "*g*" به‌جای "*e*" آمده در صورتی که این دو حرف در دو سطر جدا قراردارند و احتمال اشتباه تایپی این دو کم است.

همان‌گونه که دیده می‌شود، در این نرم‌افزار واژهٔ "*sear*" که منطقی‌ترین گزینه است، اصلا پیشنهاد نشده و بهترین پیشنهاد در ردهٔ سوم قرار دارد.

اکنون نمونه‌ای دیگر را در نظر گیرید: اگر کاربر بخواهد واژهٔ "*that*" را تایپ کند، اما به اشتباه به‌جای حرف "*t*" دوم، حرف کناری آن یعنی "*y*" و بنابراین "*thay*" را تایپ کند، نرم‌افزار ورد واژگان زیر را به‌ترتیب اولویت پیشنهاد می‌دهد:

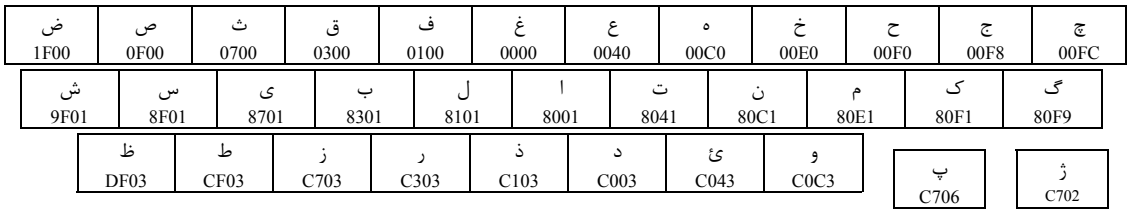
"*they*": در اینجا "*e*" به‌جای "*a*" آمده در صورتی که این دو حرف در دو سطر جدا قراردارند و احتمال اشتباه تایپی این دو ناچیز است.

"*the*": در این مورد فرض شده که کاربر به‌جای حرف "*e*"، به‌اشتباه دو حرف "*ay*" را تایپ کرده که امری بسیار بعید است.

"*tray*": در اینجا فرض شده که به‌جای حرف "*r*"، حرف "*h*" تایپ شده در صورتی که این دو حرف در دو سطر جدا قراردارند و احتمال اشتباه تایپی این دو اندک است.

"*that*": در اینجا "*t*" به‌جای "*y*" آمده و این دو حرف بر کلیدهای مجاور قرار دارند و این انتخاب معقول‌ترین انتخاب به‌نظر می‌رسد.

"*thaw*": در این مورد هم "*w*" به‌جای "*y*" آمده، در صورتی که این دو حرف در همسایگی دوری نسبت به‌هم جای دارند و احتمال اشتباه تایپی این دو کم است.



شکل (۲): کد اختصاص داده شده به هر یک از حروف فارسی بنا بر فاصلهٔ همسایگی.

همان‌گونه که دیده می‌شود، در این مثال گزینهٔ درست در اولویت چهارم پیشنهاد شده است و برخی از گزینه‌ها غیر منطقی به‌نظر می‌رسند.

۳- تصحیح غلط تایپی به‌کمک شبکهٔ عصبی

شبکهٔ عصبی مصنوعی روش و ابزاری محاسباتی است که بر روی مقادیر عددی کار می‌کند. بنابراین نخستین کار در چنین مساله‌ای این است که مفاهیم کیفی مانند حروف الفبا و دوری یا نزدیکی آنها به یکدیگر، به مقادیر کمّی و عددی قابل محاسبه تبدیل شود. از این‌رو به هر یک از حروف صفحه‌کلید یک کد عددی به‌گونه‌ای اختصاص می‌یابد که مفهوم دوری / نزدیکی کلیدها را نیز در بر داشته باشد.

۳-۱- تعریف فاصله (شعاع همسایگی) کلیدها

برای داشتن معیاری برای سنجش دوری یا نزدیکی دو حرف بر صفحه کلید، فاصله یا شعاع همسایگی دو کلید را چنین تعیین می‌کنیم:

- فاصلهٔ هر کلید با کلید همسایهٔ چپ یا راستش برابر با یک است.
- فاصلهٔ هر کلید با کلید kام کنارش برابر با k است.
- فاصلهٔ هر کلید با کلید همسایه‌اش در سطر بالا یا پایین برابر با ۲ است.

با توجه به این‌که در طولانی‌ترین سطر صفحه‌کلید ۱۲ حرف فارسی گنجانیده شده، یک کد ۱۲ بیتی می‌تواند فاصله کلیدهای هر ردیف را نشان دهد و از آنجا که ۳ ردیف حروف بر صفحه‌کلید موجود است و هر دو ردیف باید دست‌کم فاصلهٔ ۲ با هم داشته باشند، طول این کد به ۱۶ بیت افزایش می‌یابد. بنابراین با اختصاص این کدها (شکل ۲ و جدول ۲) می‌توان به هر یک از کلیدهای صفحه‌کلید یک کد ۱۶ بیتی منحصر به‌فرد اختصاص داد.

۳-۲- کدهای ورودی شبکهٔ عصبی

برای تبدیل هر یک از حروف الفبای فارسی به یک مقدار عددی که به‌عنوان ورودی برای شبکهٔ عصبی مصنوعی مناسب باشد، به‌جای این‌که هر یک از حروف با یکی از اعداد ۰ تا ۳۲ متناظر شود، به‌هر یک از آنها یک کد ۱۶ بیتی اختصاص یافته است. بدین منظور حرف "غ" مرکز کلیدهای فارسی با کد دودویی $0000000000000000_{(2)}$ و شانزده‌شانزدهی $0000_{(H)}$ در نظر گرفته شده است و کد کلیدهای کناری آن به‌گونه‌ای تنظیم شده است که قواعد شعاع همسایگی رعایت شود. برای نمونه، کلید کناری آن حرف "ع" کد $0000000010000000_{(2)}$ یا $0040_{(H)}$ را گرفته که با حرف "غ" یک بیت اختلاف دارد و حرف "ه" نیز کد $0000000011000000_{(2)}$ یا $00C0_{(H)}$ را می‌گیرد که با حرف "ع" یک اختلاف و با حرف "غ" دو اختلاف دارد و این اختلافات نشانگر فاصلهٔ همسایگی میان آنها است.

شکل ۲ حروف فارسی و جایگاه آن بر صفحه‌کلید را به‌همراه کدهای شانزده‌شانزدهی مربوطه نشان می‌دهد. ورودی شبکهٔ عصبی هر یک از حروف فارسی با شرح بیشتر در جدول ۲ درج گردیده است.

جدول (۲): حروف فارسی و کلید و کد آنها بنا بر فاصلهٔ همسایگی.

ردیف	حرف فارسی	کلید	کد هگز	کد دودویی (ورودی شبکهٔ عصبی)
۱	ا	h	8001	۱۰۰۰۰۰۰۰۰۰۰۰۰۰۱
۲	ب	f	8301	۱۰۰۰۰۰۱۱۰۰۰۰۰۰۰۱
۳	پ	` یا \	C706	۱۱۰۰۰۱۱۱۰۰۰۰۱۱۰
۴	ت	j	8041	۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱
۵	ث	e	0700	۰۰۰۰۰۱۱۱۰۰۰۰۰۰۰
۶	چ	[	00F8	۰۰۰۰۰۰۰۰۱۱۱۱۱۰۰۰
۷	چ	]	00FC	۰۰۰۰۰۰۰۰۱۱۱۱۱۱۰۰
۸	ح	p	00F0	۰۰۰۰۰۰۰۰۱۱۱۱۰۰۰۰
۹	خ	o	00E0	۰۰۰۰۰۰۰۰۱۱۱۰۰۰۰۰
۱۰	د	n	C003	۱۱۰۰۰۰۰۰۰۰۰۰۰۱۱
۱۱	ذ	b	C103	۱۱۰۰۰۰۰۱۰۰۰۰۰۰۱۱
۱۲	ر	v	C303	۱۱۰۰۰۰۱۱۰۰۰۰۰۰۱۱
۱۳	ز	c	C703	۱۱۰۰۰۱۱۱۰۰۰۰۰۰۱۱
۱۴	ژ	C یا \	C702	۱۱۰۰۰۱۱۱۰۰۰۰۰۰۱۰
۱۵	س	s	8F01	۱۰۰۰۱۱۱۱۰۰۰۰۰۰۰۱
۱۶	ش	a	9F01	۱۰۰۱۱۱۱۱۰۰۰۰۰۰۰۱
۱۷	ص	w	0F00	۰۰۰۰۱۱۱۱۰۰۰۰۰۰۰۰
۱۸	ض	q	1F00	۰۰۰۱۱۱۱۱۰۰۰۰۰۰۰۰
۱۹	ط	x	CF03	۱۱۰۰۱۱۱۱۰۰۰۰۰۰۱۱
۲۰	ظ	z	DF03	۱۱۰۱۱۱۱۰۰۰۰۰۰۱۱
۲۱	ع	u	0040	۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰
۲۲	غ	y	0000	۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰
۲۳	ف	t	0100	۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰
۲۴	ق	r	0300	۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰
۲۵	ک	;	80F1	۱۰۰۰۰۰۰۰۱۱۱۱۰۰۰۱
۲۶	گ	‘	80F9	۱۰۰۰۰۰۰۰۱۱۱۱۱۰۰۱
۲۷	ل	g	8101	۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۱
۲۸	م	l	80E1	۱۰۰۰۰۰۰۰۱۱۱۰۰۰۰۱
۲۹	ن	k	80C1	۱۰۰۰۰۰۰۰۱۱۰۰۰۰۰۱
۳۰	و	,	C0C3	۱۱۰۰۰۰۰۰۱۱۰۰۰۰۱۱
۳۱	ه	i	00C0	۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۰
۳۲	ی	Z و d	8701	۱۰۰۰۰۱۱۱۰۰۰۰۰۰۰۱
۳۳	ئ	m	C043	۱۱۰۰۰۰۰۰۱۰۰۰۰۰۱۱

البته در عمل، مقادیر ورودی شبکه‌های عصبی نه به‌طور باینری، بلکه به‌صورت دوقطبی در نظر گرفته شده‌اند تا بازدهی شبکه بهتر باشد. پس هر یک از حروف واژگان، یک کد با ۱۶ ورودی ۱ یا ۱- را به‌خود

اختصاص می‌دهد و از این‌رو در لایهٔ ورودی شبکهٔ عصبی، ۱۶ نرون به هر یک از آنها اختصاص خواهد یافت.

۳-۳- فرهنگ واژگان به‌کار رفته در شبیه‌سازی

در شبیه‌سازی این پژوهش، ۶۰۰ واژهٔ ۴، ۵ و ۶ حرفی به تعداد مساوی، برای آموزش شبکه‌ها و ۳۰۰ واژه نیز برای آزمایش به‌کار گرفته شده‌اند. بیشتر واژگان از اسامی اعلام (کسان و جای‌ها) یا کلمات پرکاربرد در زبان فارسی و از فرهنگ واژه‌های معتبر فارسی، مانند "لغت‌نامه دهخدا" و "فرهنگ معین"، برگزیده شده‌اند.

۴- شبیه‌سازی

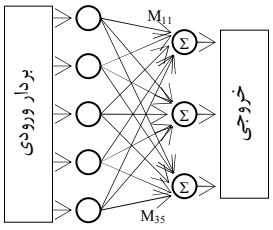
شبیه‌سازی و آزمون مقادیر و واژگان توسط نرم‌افزار متلب[۴۰]، به‌همراه نرم‌افزار کمکی تحت زبان ویژوال بیسیک[۴۱] برای تهیهٔ مقادیر ورودی دو نوع شبکهٔ عصبی هاپفیلد و پرسپترون چند لایه[۴۲] انجام شد. الگوهای ورودی در دسته‌های ۴، ۵ و ۶ حرفی به شبکهٔ عصبی وارد شد و پس از آموزش شبکه، هر بار الگوهای ورودی به‌دفعات و با تغییر تصادفی یکی از حروف واژه آزموده شد.

۴-۱- شبیه‌سازی با شبکهٔ عصبی هاپفیلد

هنگامی که دربارهٔ تصحیح املایی واژگان سخن می‌گوییم، شبکه‌های گونهٔ حافظهٔ یادآور[۴۳] نخستین شبکه‌های مناسب به‌نظر می‌رسند. شبکه حافظهٔ یادآور سیستمی است که می‌تواند داده‌های ذخیره شده (الگوها[۴۴]) را حتی با دیدن ورودهای همراه با غلط بازیابی (یادآوری) کند[۴]. به‌عنوان نمونه‌ای کوچک برای آشنایی با روش کار این شبکه خط فارسی را به ۵ حرف ("ا"، "ب"، "ر"، "ش" و "ی") محدود می‌کنیم و می‌خواهیم سه واژهٔ "ابر"، "آرش" و "بیش" را یاد گرفته، یادآوری نماییم. بر این پایه هر واژه را می‌توان بنا بر حروف مورد استفاده در آن به‌صورت یک ماتریس ۵ عنصری مانند شکل ۳-الف نمایش داد. این مساله به شبکه‌ای با ۵ گره ورودی و ۳ گره خروجی مانند شکل ۳-ب نیاز دارد. پاسخ آزمون واژه‌ای مانند "اید" - که همان واژهٔ "یاد" با رخداد خطای انتقال است - از ضرب دو ماتریس فرهنگ واژگان در دادهٔ آزمایشی به‌دست آمده، سطر خروجی بزرگتر، پاسخ سیستم است. ماتریسی که فرهنگ واژگان را در بر دارد، ماتریس همبستگی[۴۵] می‌نامند.

[ی ر د ب ا]
[1 1 0 1 0] = "ابر"
[0 1 1 1 0] = "برد"
[1 0 1 0 1] = "یاد"

شکل ـ۳-(الف): کدگذاری و ذخیرهٔ واژگان به‌عنوان ورودی شبکه



شکل (۳-ب): شبکهٔ عصبی حافظهٔ یادآور

$$R = M.S = \begin{bmatrix} 11010 \\ 01110 \\ 10101 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix}$$

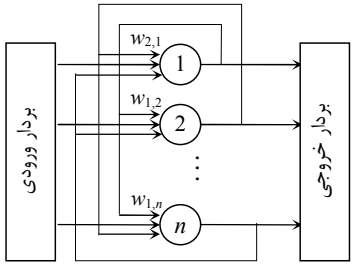
شکل (۳-پ): بازیابی واژگان در شبکهٔ حافظهٔ یادآور به‌کمک ماتریس همبستگی

شبکهٔ عصبی مصنوعی هاپفیلد شاخص‌ترین نوع شبکه‌های یادآور و از گونهٔ حافظهٔ خودیادآور[۴۶] است که کاربردهای بسیاری در دانش‌های گوناگون دارد. این شبکه نخستین بار در سال ۱۹۸۲ میلادی توسط جان هاپفیلد[۴۷] ارایه شد[۱۵]. وی در سال ۱۹۸۵ این شبکه را به‌کمک تنک[۴۸] گسترش داد و با آن مساله فروشندهٔ دوره‌گرد[۴۹] را با در نظر گرفتن ده شهر و صد نرون با کارایی بهتری حل کرد. فروشندهٔ دوره‌گرد یک مسالهٔ بهینه‌سازی معروف است که در زمرهٔ مسایل بسیار مشکل قرار می‌گیرد و با روش‌های معمولی نمی‌توانند در زمانی معقول پاسخی بهینه را به‌دست آورد. هاپفیلد و تنک مسالهٔ خود را تا ۳۰ شهر با موفقیت گسترش دادند و پس از گذشت ۲۰ سال هنوز هم روش آنها جزو بهترین الگوریتم‌های شبکهٔ عصبی برای حل مسالهٔ فروشندهٔ دوره‌گرد است[۱۶].

شبکه عصبی هاپفیلد در قالب یک سیستم پویا توسط یک تابع انرژی که باید تعادلی میان اهداف تابع مساله - که باید حداقل شود - ایجاد می‌کند[۱۷]. پس از این موفقیت شبکه عصبی هاپفیلد، بسیاری از مسایل مهندسی در قالب تابع انرژی که باید کمینه شود، ارایه گردید. چنین راه حلی بسیار جذاب است، زیرا پردازش موازی را برای حل مسایل امکان پذیر می‌سازد[۱۸].

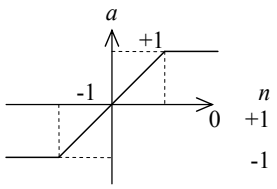
بنابراین این شبکه می‌تواند با یادگیری و حفظ تعدادی واژه در حافظهٔ خود، ورودی همراه با نویز یا همان غلط املایی را به نزدیک‌ترین الگو نگاشت کند و صورت درست واژه را در خروجی تداعی کند. ساختار شبکهٔ هاپفیلد در شکل ۴ و تابع فعالیت[۵۰] آن در شکل ۵

نشان داده شده است.



شکل (۴): نمای کلی شبکهٔ هاپفیلد.

اما دو نقص بزرگ شبکهٔ هاپفیلد ظرفیت پایین آن (در حدود ۱۵٪ اندازهٔ شبکه یا تعداد گره‌ها) و همچنین هزینهٔ محاسباتی بالای آن است[۱۹]. از این‌رو انتظار می‌رود که با افزایش تعداد الگوهای یاد داده شده، دقت شبکه کاهش یابد و بنابراین شبکهٔ هاپفیلد نامزدی مناسب برای سیستم غلط‌یاب املایی نیست[۴].

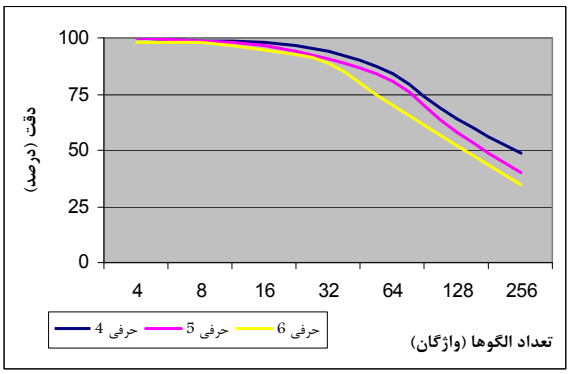


شکل (۵): تابع فعالیت (دو قطبی) شبکهٔ هاپفیلد.

نتایج این آزمون و شبیه‌سازی آن در جدول ۳ و شکل ۶ نمایش داده شده است.

جدول (۳): ارزیابی نتایج شبکهٔ هاپفیلد در تصحیح غلط تایپی واژگان.

ردیف	تعداد الگوها (واژگان)	تعداد حروف واژه	حداقل اپک لازم برای همگرا شدن شبکه	دقت (درصد)
۱	۴	۴	۲۰	۱۰۰
۲	۴	۵	۵۰	۱۰۰
۳	۴	۶	۶۰	۹۸
۴	۸	۴	۱۰۰	۹۹
۵	۸	۵	۱۵۰	۹۹
۶	۸	۶	۲۰۰	۹۸
۷	۱۶	۴	۳۰۰	۹۸
۸	۱۶	۵	۵۰۰	۹۷
۹	۱۶	۶	۸۰۰	۹۵
۱۰	۳۲	۴	۱۰۰۰	۹۴
۱۱	۳۲	۵	۲۵۰۰	۹۱
۱۲	۳۲	۶	۴۰۰۰	۸۹
۱۳	۶۴	۴	۹۰۰۰	۸۴
۱۴	۶۴	۵	۱۲۰۰۰	۸۱
۱۵	۶۴	۶	۱۲۰۰۰	۷۰
۱۶	۱۲۸	۴	۲۰۰۰۰	۶۴
۱۷	۱۲۸	۵	۲۰۰۰۰	۵۸
۱۸	۱۲۸	۶	۲۰۰۰۰	۵۲
۱۹	۲۵۶	۴	۲۰۰۰۰	۴۹
۲۰	۲۵۶	۵	۲۰۰۰۰	۴۰
۲۱	۲۵۶	۶	۲۵۰۰۰	۳۵



شکل (۶): نتایج شبکهٔ هاپفیلد در تصحیح غلط تایپی واژگان ۴، ۵ و ۶ حرفی

همان‌گونه که در این جدول مشاهده می‌شود، با افزایش تعداد الگوها، دقت شبکه کاهش یافته است. همچنین افزایش تعداد حروف واژگان مورد آزمایش اندکی از دقت شبکه کاسته و برای یادگیری و تصحیح تنها ۲۵۶ واژه به بیش از ۲۵۰۰۰ دور گردش شبکه (اپک[۵۱]) در هنگام آموزش نیاز است در صورتی که دقت تصحیح شبکه از ۵۵٪ فراتر نمی‌رود.

۴-۲- شبیه‌سازی با شبکهٔ عصبی پرسپترون چندلایه

شبکهٔ عصبی پرسپترون چند لایه، شبکه‌ای است که در اصل برای مسایل دسته‌بندی و تخمین تابع طراحی شده و در این کارها از موفق‌ترین شبکه‌های عصبی بوده و توانسته است با استفاده از قانون انتشار خطا به‌عقب[۵۲] بسیاری از مسایل غیرقابل حل توسط شبکه‌های دیگر را حل کند[۱۹]. اما با توجه به این‌که کار اصلی این شبکه دسته‌بندی و تخمین تابع است، بنابراین به‌نظر نمی‌رسد که در مسالهٔ تصحیح غلط تایپی - که می‌توان گفت مسالهٔ یادآوری است - به‌کار آید.

نکتهٔ ساده‌ای که باید برای کارآمدی شبکهٔ پرسپترون چند لایه در مسالهٔ تصحیح واژهٔ غلط به‌کار گرفت، این است که: اگر هر یک از الگوهایی که باید یاد گرفته شود را با یک دسته در فضای n بعدی متناظر کنیم، برای یادگیری 2^n واژه، مساله به مسالهٔ دسته‌بندی 2^n دسته‌ای با n نرون در لایهٔ خروجی شبکه متناظر می‌شود. پس برای نمونه، برای یادگیری مجموعه‌ای از واژگان با حدود ۶۵ هزار واژه، به شبکه‌ای با ۱۶ گره خروجی نیاز است. جدول ۴ نمونه‌ای از این روش را برای دسته‌بندی ۸ واژهٔ چهار حرفی نشان می‌دهد. واژه‌های مورد آزمایش به‌عمد به‌گونه‌ای برگزیده شده‌اند که به‌هم نزدیک باشند و با دگرگونی یکی-دو حرف به واژه‌ای دیگر در واژه‌نامه تبدیل شوند، تا نتایج تصحیح بر روی آنها بهتر نمایان شود.

جدول (۴): نمونه‌ای از ۸ واژهٔ آزموده با شبکهٔ عصبی و کد ویژهٔ آنها.

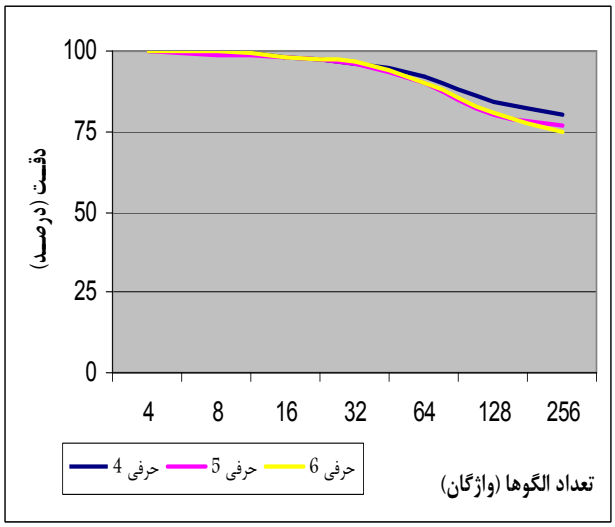
ردیف	واژه	کد باینری (دسته)
۱	نامی	۰۰۰
۲	مانی	۰۰۱
۳	مینا	۰۱۰
۴	امین	۰۱۱
۵	میان	۱۰۰
۶	ایمن	۱۰۱
۷	نیام	۱۱۰
۸	نیما	۱۱۱

روشن است که این مجموعه نیاز به شبکه‌ای با ۳ نرون خروجی دارد. نتایج آزمایش با این روش با تعداد واژگان مختلف در جدول ۵ و شکل ۷ آمده است. در این آزمون‌ها، تعداد اپک‌ها به‌گونه‌ای تنظیم شده که خطای شبکه به صفر برسد، اما از آنجا که گاه این امر میسر نمی‌شود و با توجه به این‌که کار شبکه با تعداد دسته، لایه و نرون بسیار سنگین می‌شود (آموزش مورد ردیف ۱۹ بیش از ۳ روز زمان برده است!) کار با اپک‌های کمتر خاتمه یافته است.

همان‌گونه که مشاهده می‌شود، با افزایش تعداد الگوها، دقت شبکه کاهش می‌یابد. همچنین افزایش تعداد حروف واژگان مورد آزمایش اندکی از دقت شبکه کاسته است.

جدول (۵): ارزیابی نتایج شبکهٔ پرسپترون چندلایه در تصحیح غلط تایپی واژگان.

ردیف	تعداد دسته‌ها	تعداد حروف	حداقل اپک (لازم یا	تعداد لایهٔ	تعداد نرون‌های	دقت (درصد)
۱	۴	۴	۱۰	۱	۲	۱۰۰
۲	۴	۵	۱۵	۱	۲	۱۰۰
۳	۴	۶	۲۰	۱	۲	۱۰۰
۴	۸	۴	۵۰	۱	۳	۱۰۰
۵	۸	۵	۶۰	۱	۳	۹۸
۶	۸	۶	۸۰	۱	۳	۱۰۰
۷	۱۶	۴	۱۰۰	۲	۴	۹۸
۸	۱۶	۵	۱۴۰	۲	۴	۹۸
۹	۱۶	۶	۲۰۰	۲	۴	۹۸
۱۰	۳۲	۴	۵۰۰	۲	۵	۹۶
۱۱	۳۲	۵	۸۰۰	۲	۵	۹۶
۱۲	۳۲	۶	۱۳۰۰	۲	۵	۹۸
۱۳	۶۴	۴	۲۰۰۰	۲	۶	۹۲
۱۴	۶۴	۵	۳۰۰۰	۲	۶	۹۰
۱۵	۶۴	۶	۴۰۰۰	۲	۶	۹۰
۱۶	۱۲۸	۴	۱۰۰۰۰	۲	۷	۸۴
۱۷	۱۲۸	۵	۱۲۰۰۰	۲	۷	۸۰
۱۸	۱۲۸	۶	۱۵۰۰۰	۲	۷	۸۱
۱۹	۲۵۶	۴	۱۰۰۰۰	۳	۸	۸۰
۲۰	۲۵۶	۵	۱۲۵۰۰	۳	۸	۷۷
۲۱	۲۵۶	۶	۱۵۰۰۰	۳	۸	۷۵



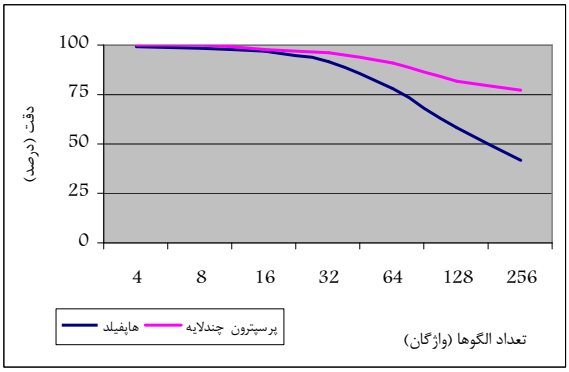
شکل (۷): نتایج شبکهٔ پرسپترون چند لایه در تصحیح واژگان ۴، ۵ و ۶ حرفی

جدول (۶): مقایسهٔ میانگین بازدهی دو شبکهٔ هاپفیلد و پرسپترون چند لایه

ردیف	تعداد الگوها	هاپفیلد	پرسپترون چندلایه
۱	۴	۹۹/۳	۱۰۰
۲	۸	۹۸/۷	۹۹/۷
۳	۱۶	۹۶/۷	۹۸
۴	۳۲	۹۱/۳	۹۶/۳
۵	۶۴	۷۸/۳	۹۰/۷
۶	۱۲۸	۵۸	۸۱/۷
۷	۲۵۶	۴۱/۳	۷۷/۳

۴-۳- ارزیابی نتایج

جدول‌های ۳ و ۵ به‌خوبی نشان می‌دهند که کارکرد شبکهٔ پرسپترون چند لایه بسیار بهتر از شبکهٔ هاپفیلد بوده است. جدول ۶ و شکل ۸ نیز این تفاوت را به‌صورت میانگین برای واژگان ۴ تا ۶ حرفی نشان می‌دهند.



شکل (۸): بازدهی دو شبکهٔ هاپفیلد و پرسپترون چند لایه.

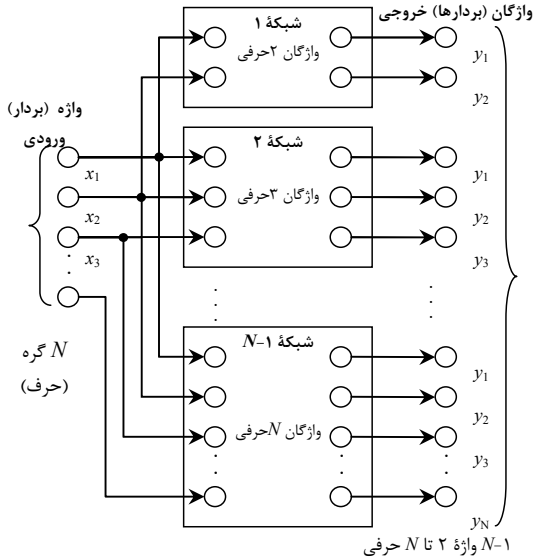
۴-۴- ترمیم مشکل ظرفیت شبکه

همان‌گونه که از نتایج شبیه‌سازی بر می‌آید، پایین بودن ظرفیت این دو شبکهٔ عصبی بزرگترین مشکل بر راه اجرایی شدن طرح است و برطرف ساختن آن ساده به‌نظر نمی‌رسد؛ زیرا شبکهٔ هاپفیلد اصولا ظرفیتی پایین دارد و تنها با اعمال برخی تغییرات ساختاری بر آن می‌توان ظرفیت شبکه را تا چند برابر افزایش داد، که آن نیز از نزدیکترین مرز نیاز دور است. شبکهٔ پرسپترون چند لایه نیز - به‌رغم عملکرد ممتازش نسبت به دیگر شبکه‌ها - با توجه به ماهیت مسالهٔ این پژوهش، امکان افزایش چشمگیر ظرفیت با تنظیمات کنونی را ندارد.

البته هر چند که مشکل ظرفیت مانعی اصولی بر راه انجام و پیشبرد پژوهش در به‌کارگیری شبکه‌های عصبی مصنوعی برای اجرای طرح تصحیح غلط املایی و تایپی نخواهد بود و با پیدایش شبکه‌های پرتوان‌تر یا استفادهٔ بهینه‌تر از شبکه‌های موجود، این مساله حل خواهد شد، اما به‌هر حال ارایهٔ روش‌هایی مستقل از نوع شبکه‌ها برای افزایش ظرفیت سیستم سودمند خواهد بود که در ذیل به دو مورد اشاره می‌شود:

تقسیم‌بندی برپایهٔ طول واژه: با توجه به ماهیت شبکه که ورودی‌هایی گسسته متشکل از واژگان ۲ تا N حرفی (حدود ۱۰) با ۳۳ نوع حرف ("آ" تا "ی" و "ء") دارد، می‌توان آن را مانند شکل ۹ به N-۱ شبکهٔ مستقل برای پردازش واژگان ۲ تا N حرفی تقسیم کرد. روشن است که هر یک از گره‌های لایهٔ ورودی یا خروجی این شبکه برای نمایش حروف، خود از شانزده نرون باینری یا دو قطبی تشکیل می‌شود. این تدبیر ظرفیت کل سیستم را به‌طور متوسط N-۱ برابر افزایش می‌دهد. دیگر مزیت این کار، رفع مشکل درج و حذف حرف در واژگان است که بسته به نیاز، با آزمون واژگان i حرفی در شبکه‌های i+۱ و i-۱ حرفی انجام خواهد شد.

در این شبکه، بردار واژهٔ k حرفی به شبکهٔ k و در صورت نیاز به شبکه‌های k+۱ یا k-۱ وارد می‌شود و بنابراین پاسخ تنها در خروجی شبکه‌های k، k+۱ و k-۱ ظاهر می‌گردد.



شکل (۹): به‌کارگیری شبکه‌های موازی در پردازش واژگان ۲ تا N حرفی

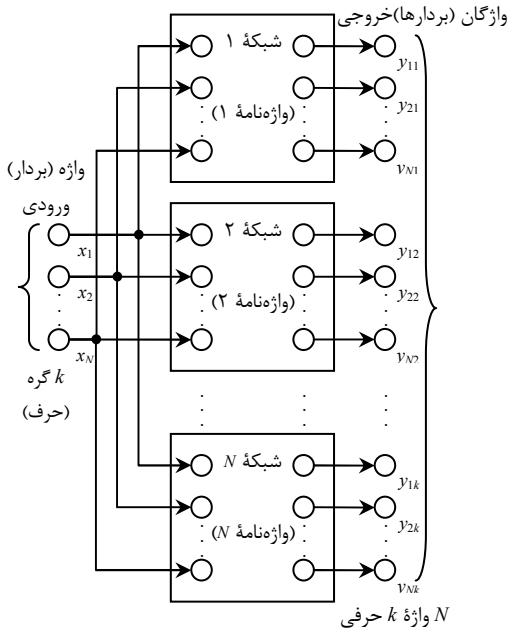
جدول ۷ نتایج آزمون روش تقسیم‌بندی بر پایهٔ طول واژه را بر واژگان ۴ تا ۶ حرفی با ۳ زیرشبکه که هر یک از آنها ۴ تا ۲۵۶ واژه را یاد گرفته‌اند، نشان می‌دهد. در صورتی که واژهٔ درست مرتبط با واژهٔ غلط مورد آزمایش، دست‌کم در یکی از خروجی‌های سیستم ظاهر شده باشد، عملکرد شبکه درست در نظر گرفته شده است.

جدول (۷): افزایش ظرفیت با روش تقسیم‌بندی طول واژه.

ردیف	تعداد الگوهای هر شبکه	تعداد کل واژگان	هاپفیلد	پرسپترون چندلایه
۱	۴	۱۲	۹۹/۳	۱۰۰
۲	۸	۲۴	۹۸/۲	۹۹/۵
۳	۱۶	۴۸	۹۵/۵	۹۷/۲
۴	۳۲	۹۶	۸۹/۱	۹۴/۱
۵	۶۴	۱۹۲	۷۳	۸۸/۷
۶	۱۲۸	۳۸۴	۵۰/۵	۷۹/۸
۷	۲۵۶	۷۶۸	۳۵	۷۲

تقسیم‌بندی برپایهٔ نوع واژه: در هنگام آموزش و دسته‌بندی واژگان k حرفی، می‌توان واژگان را به‌جای یک شبکه، در N شبکه پخش کرد. این ترفند ظرفیت کل سیستم را به‌طور متوسط N برابر افزایش می‌دهد و گذشته از آن، امکان پیشنهاد چندین واژهٔ جایگزین در خروجی را هم پدید می‌آورد. روشن است که گره‌های لایهٔ ورودی و خروجی این شبکه نیز، که در شکل ۱۰ نمایش داده شده است، هر یک

نشانگر شانزده نرون هستند. در این شبکه، بردار واژهٔ k حرفی به تمام شبکه‌ها وارد می‌شود و بنابراین در همهٔ خروجی‌های N شبکه، پاسخ ظاهر می‌گردد.



شکل (۱۰): پخش واژگان در شبکه‌های موازی.

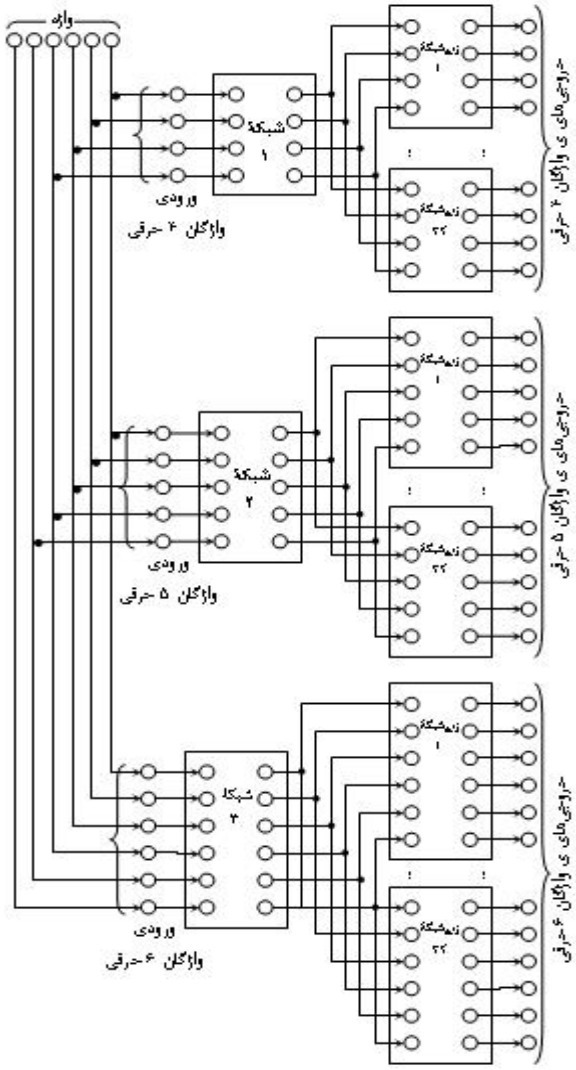
جدول ۸ نتایج آزمون روش تقسیم‌بندی برپایهٔ نوع واژه را بر واژگان ۵ حرفی با شبکه‌هایی متشکل از ۴ تا ۳۲ زیرشبکه که هر یک از آنها ۳۲ واژه را یاد گرفته‌اند، نشان می‌دهد.

جدول (۸): افزایش ظرفیت با روش تقسیم‌بندی نوع واژه.

ردیف	تعداد زیرشبکه	تعداد کل واژگان	هاپفیلد	پرسپترون چندلایه
۱	۴	۱۲۸	۹۰/۱	۹۵
۲	۸	۲۵۶	۸۸/۱	۹۴/۷
۳	۱۶	۵۱۲	۸۲/۱	۹۰/۹
۴	۳۲	۱۰۲۴	۷۷/۷	۸۶/۱

شبکهٔ ترکیبی: با ترکیب این دو شبکهٔ بالا، در حالتی که سیستم غلط‌یاب از ۳ شبکهٔ ۴ تا ۶ حرفی، هر یک با ۳۲ زیرشبکهٔ تقسیم شده بر پایهٔ ۳۲ واژه (در مجموع ۳۰۷۲ واژه) شکل گرفته است، کارایی سیستم به دقت ۷۰ درصد در شبکهٔ هاپفیلد و ۷۹/۶ درصد در شبکهٔ پرسپترون چند لایهٔ رسید. شکل ۱۱ نمای کلی این شبکه را نشان می‌دهد.

در این سیستم، هر واژهٔ k حرفی تنها به ورودی شبکه متناظرش وارد می‌شود و هر واژه ۳۲ خروجی خواهد داشت.



شکل (۷): افزایش ظرفیت با تکیب شبکه‌ها.

۵- نتیجه‌گیری

در این پژوهش کوشش شد تا به‌کمک شبکهٔ عصبی مصنوعی روشی برای کشف و تصحیح غلط‌های املایی برآمده از خطاهای تایپی کاربر ارایه شود. بدین منظور بنا بر یک قرارداد مبتنی بر فاصلهٔ دکمه‌های صفحه کلید، کدی ۱۶ بیتی به هر یک از حروف اختصاص داده شد تا شبکهٔ عصبی از روی آن همسایگی را تشخیص دهد. سپس نخست واژگان در گروه‌های ۴، ۵ و ۶ حرفی و دسته‌های ۴ تا ۲۵۶ واژه‌ای با دو نوع شبکهٔ عصبی هاپفیلد و پرسپترون چند لایه آزموده شدند. نتایج این آزمون‌ها نشان داد که شبکهٔ هاپفیلد از دقت عملکرد ۱۰۰٪ درستی برای تصحیح غلط‌های تایپی فرهنگ لغتی با ۴ واژه، به دقت درستی کمتر از ۴۲٪ برای تصحیح غلط‌های تایپی فرهنگ لغتی با ۲۵۶ واژه می‌رسد؛ در حالی‌که شبکهٔ عصبی پرسپترون چندلایه چنین فرهنگ واژگانی را با بیش از ۷۷٪ دقت درستی غلط‌یابی می‌کند. از این‌رو با اطمینان می‌توان گفت که شبکهٔ پرسپترون چند لایه، در این مسالهٔ یادآوری بهتر از شبکهٔ هاپفیلد کار می‌کند. افزون بر این، گذشته از آن‌که دقت عملکرد دو شبکه در هنگام افزایش واژگان کاهش

می‌یافت، با افزایش حروف واژگان نیز با کاهشی محسوس روبرو می‌گردید.

همچنین در گام بعدی با تقسیم واژگان واژه‌نامه در گروه‌های کوچکتر بر پایهٔ طول و نوع واژگان و آموزش آنها در شبکه‌های جداگانه و سپس ترکیب این شبکه‌ها، ظرفیت سیستم غلط‌یاب - با حفظ نسبی دقت - به اندازه‌ای چشمگیر افزایش یافت؛ به‌گونه‌ای که بیش از ۳۰۰۰ واژه با دقت درستی ۸۰ درصد تصحیح گردید.

البته روش‌های ارایه شده در این پژوهش، به‌ترتیب در صورتی خوب کار می‌کند که حروف واژهٔ مورد آزمون اشتباه، کم یا اضافه تایپ شده باشد و در مورد خطاهای ناشی از انتقال حروف در واژه، دیگر روش‌های موجود کارآمدتر هستند.

همچنین اهداف زیر دستور کارهای آینده قرار دارد:

- آزمایش تعداد واژگان بیشتر تا دست‌کم ۱۶ هزار کلمه که مقداری مناسب برای فرهنگ واژگان است.
- آزمایش شبکه‌های عصبی دیگر برای دستیابی به نتایج بهتر، به‌ویژه شبکه‌های عصبی فازی[۵۳]
- تنظیم بهتر تعداد لایه‌ها و نرون‌های هر لایه در شبکهٔ پرسپترون چندلایه
- یافتن نقطهٔ بهینه برای حداکثر تعداد اپک‌های شبکه
- ادغام واژگان با تعداد حروف مختلف در یک شبکه
- بهینه‌سازی کدهای حروف صفحه‌کلید و کاهش طول آنها.

مراجع

[1] Allen James, Natural Language Understanding, The Benjamin/Cummings Publishing Co., 2nd Edition, 1994.

[2] Patrick Ruch, Robert Baud and Antoine Geissbuhler, "Toward filling the gap between interactive and fully-automatic spelling correction using the linguistic context", IEEE International Conference on Systems, Man, and Cybernetics 1 (2001) 199-204.

[3] Patrick Ruch, Robert Baud and Antoine Geissbuhler, "Using lexical disambiguation and named-entity recognition to improve spelling correction in the electronic patient record", Artificial Intelligence in Medicine 29 (2003) 169-184.

[4] V. Cherkassky, N. Vassilas and G. L. Brodt, "Conventional and associative memory-based spelling checkers", Proceedings of the 2nd International IEEE Conference on Tools for Artificial Intelligence (1990) 138-144.

[5] V.J. Hodge, J. Austin, "A comparison of a novel neural spell checker and standard spell checking algorithms", Pattern Recognition 35 (11) (2002) 2571–2580.

[6] Dominique Lavenier, "A high performance systolic chip for spelling correction", Euro ASIC '92, Proceedings 1 (1992) 381-384.

[7] R. Garfinkel., E. Fernandez, R. Gopal, "Design of an interactive spell checker: optimizing the list of offered words", Decision Support Systems 35 (2003) 385– 397.

[8] K. Kukich, "Techniques for automatically correcting words in text", ACM Comput Surveys 24 (4) (1992) 377–439.

[9] J.R. Ullman, "A binary n-gram technique for automatic correction of substitution, deletion, insertion and reversal errors in words", The Computer Journal 20 (2) (1977) 141–147.

[10] Teiji Furugori,. "Improving spelling checkers for Japanese users of English". IEEE Transaction on Professional Communication, 33 (3) (1990) 138– 142.

[11] H. Shang and T. H. Merrettal, "Tries for approximate string matching". IEEE Transaction on Knowledge and Data Engineering, 8 (4) (1996) 540– 547.

[12] Bidyut Baran Chaudhuri,. "Towards Indian language spell-checker design". IEEE Proceedings of the Language Engineering Conference (LEC'02), (2002) 139-146.

[13] Z. Lei, Z. Ming, H. Changning and S. Maosong, "Automatic Chinese text error correction approach based-on fast approximate Chinese word-matching algorithm", Proceedings of the 3rd World Congress on Intelligent Control and Automation. (2000) 2739-2743.

[14] Jin Hu Huang and David Powers, "Large scale experiments on correction of confused words". IEEE Proceedings 24th Australasian Computer Science Conference ACSC2001 (2001) 77-82.

[15] V. Parisi, E. Garcia, J. Cabestany, J.Font, , and J. Salas, "A Hopfield neural network to track drifting buoys in the ocean", IEEE OCEANS '98 Conference Proceedings 2 (1998) 1010-1016.

[16] E.M. Cochrane, J.E. Beasley, "The co-adaptive neural network approach to the Euclidean Travelling Salesman Problem", Neural Networks 16 (2003) 1499–1525.

[17] M.A.S. Monfared, M. Etemadi, "The impact of energy function structure on solving generalized assignment problem using Hopfield neural network", European Journal of Operational Research, (2004).

[18] Wenjing Li, Tong Lee, "Projective invariant object recognition by a Hopfield network", Neurocomputing 60 (2004) 1–18.

[19] Robert Hecht-Nielsen, Neurocomputing, Addison-Wesley Publishing Company, 1989.

زیرنویس‌ها

[1] Natural Language Processing / Understanding
[2] Text Mining
[3] Spell Checking
[4] Syntax Checking
[5] Concept Checking
[6] Grammatical Confusion / Grammos
[7] Phonological Similarity / Phonos
[8] Typing Errors / Keyboard Mistyping / Typus
[9] Insertion
[10] Deletion
[11] Substitution
[12] Transposition (Interchange)
[13] Spelling Errors
[14] Transmission and Storage Errors
[15] Optical Character Recognition (OCR)
[16] Automatic Mode
[17] Interactive Mode
[18] Spelling Correction
[19] String Substitution
[20] Reversal Errors

[21] String Matching
[22] Erroneous Word
[23] Edit Distance
[24] Hamming Distance
[25] Levenshtein Distance
[26] CorrectStar
[27] Trie Algorithm
[28] n-gram
[29] Bangla
[30] Fuzzy Approximate Word-Matching
[31] Associative Memory
[32] Hopfield
[33] Backpropagation Networks
[34] Dictionary Look-up Methods
[35] Exact Matching

36 حروف‌چین‌ها به دو دسته تقسیم می‌شوند: یکی آنها که هر چه را که می‌بینند، تایپ می‌کنند و دیگر، آنان که خواسته یا ناخواسته متن را می‌فهمند و سپس تایپ می‌کنند. هر چند که یافتن فردی از گونهٔ دوم برای تایپ موهبتی است و مزایای بسیار دارد، اما احتمالا از آنجا که فرد نخست واژه را در مغز خود پردازش می‌کند و سپس آنرا تایپ می‌کند و گاه سرعت دست از سرعت پردازش برخی از واژگان در مغز بیشتر می‌شود، احتمال خطای املایی نیز در این افراد پیش می‌آید.

[37] Virtual Space
[38] Back Space
[39] Microsoft Word
[40] Matlab
[41] Visual Basic
[42] Multi-Layer Perceptron
[43] Neural Associative Memories
[44] Pattern
[45] Correlation Matrix
[46] Auto-Associative Memory
[47] John J. Hopfield
[48] Tank D. W.
[49] Traveling salesman problem
[50] Activation Function
[51] Epochs
[52] Back-Propagation
[53] Neuro-Fuzzy Networks