دشواری‌های ریشه‌یابی فارسی و روشی برای ریشه‌یابی فعل‌های ساده‌ی فارسی

احمد یوسفان[۱] ، سمیه صالحی زارعی[۲] ، بهروز مینایی بیدگلی[۳]
۱_ کاشان، کیلومتر ۶ بلوار قطب راوندی، دانشگاه کاشان، دانشکده‌ی مهندسی، گروه کامپیوتر ، کدپستی ۸۷۳۱۷-۵۱۱۶۷،
تلفن ۰۳۶۱۵۵۵۵۳۳۳ داخلی ۲۷۵۶ ، usofan@kashanu.ac.ir
۲_ کاشان، کیلومتر ۶ بلوار قطب راوندی، دانشگاه کاشان، دانشکده‌ی مهندسی، گروه کامپیوتر ، کدپستی ۸۷۳۱۷-۵۱۱۶۷،
zsalehi.somaye@gmail.com
۳_ دانشگاه علم و صنعت ایران ، نارمک تهران b_minaei@iust.ac.ir

چکیده

در این مقاله به دشواری‌های نوشتاری فارسی و برخی از ناهماهنگی‌های آن پرداخته می‌شود. همچنین به دشواری‌های نوشتار رایانه‌ای زبان فارسی و ریشه‌یابی واژه در فارسی پرداخته می‌شود. همزمان راهکارهایی برای برخی از آن‌ها داده می‌شود. سپس روشی برای ریشه‌یابی مشتقات فعل‌های فارسی به کمکِ ساخت مشقات پیشنهاد و پیاده سازی شده است.

۱ـ پیشگفتار

زبان فارسی دربردارنده‌ی گنجینه‌ی بزرگی از زیباترین سروده‌ها و داستان‌ها است. زبان فارسی یکی از پربارترین زبان‌های دنیا است. کتاب‌هایی چون مثنوی معنوی ، دیوان حافظ ، رباعیات خیام و ... به زبان‌های گوناگون گیتی برگردانده شده و بارها چاپ شده‌اند. برترین ویژگی این نوشته‌ها ، انسانی بودن آنها است به گونه‌ای که همه‌ی انسان‌ها گرایشی درونی به این نوشته‌ها دارند.

این زبان که از نیمه‌ی سده‌ی سوم ، آثاری از آن در دست است و از آن تاریخ به بعد ، روز به روز گسترش یافته و آثار بی‌شماری در آن آفریده شده ، در حیطه‌ی فرهنگ آسیایی و بلکه جهانی ، تبدیل به زبانی شده است که شاهکارهای جهانی آفریده است؛ چنان که کمتر زبانی از زبان‌های زنده‌ی موجود می‌توان سراغ داشت که در این پهنه با زبان فارسی کوس برابری زند. از جمله ویژگی‌های این زبان ، دیرندگی آن است در طول بیش از هزار سال ؛ بدین معنی که دگرگونی‌ها در آن ، در مقام سنجش با بسیاری از زبان‌های جهان ، به نسبت کمتر بوده است؛ به طوری که فارسی زبانان امروز به آسانی می‌توانند شعر فردوسی را بخوانند و بفهمند و از آن بالاتر این‌که در آثار بسیاری از گویندگان و نویسندگان سده‌های اخیر و حتی معاصر می‌توان کاربردهای کهن هزار سال پیش را دید؛ حتی در کتاب‌های دوره‌ی دبستانی می‌توان شعر رودکی و فردوسی را گنجانید[احمدی گیوی،۱۳۸۰] .

متأسفانه این درخت تنومند نیاز به توجه بیشتری دارد زیرا برای دنیای نوین آماده نشده است. پیرایش و ویرایش بر روی دیگر زبان‌های دنیا خیلی پیشتر از این آغاز شده است. ساده کردن قاعده‌ها ، کم کردن قاعده‌های پیچیده و استثناها در زبان روزمره (نه زبان ادبی)، یکسان کردن گفتار و نوشتار روزمره ، به کارگیری تعداد کمی واژه و اصطلاح ، گسترش استانداردهای آماده شده برای زبان از کارهایی است که بر روی بسیاری از زبان‌ها انجام شده است. استادان زبان انگلیسی و

زبان‌شناسان، بسیاری از قاعده‌های این زبان را پیراسته‌اند و یادگیری و به کارگیری این زبان را ساده نموده‌اند. بـرای نمونـه در نوشتار امروزی انگلیسی کمتر حرفها به هم چسبیده نوشته می‌شوند و واژه‌ها و اصطلاح‌های کمـی ، بـه ویـژه در نوشـته‌هـای علمی ، به کار گرفته می‌شود. ویراش‌های انجام شده در زبان انگلیسی بسیار بر کارهای رایانه‌ای ، که بر پایـه‌ی زبـان انگلیـسی هستند ، اثر داشته است و به پیشرفت نرم افزارهای رایانه‌ای کمک نموده است. پیرایش‌هایی که در زبان انگلیسی انجـام شـده است ، بسیاری از پیچیدگی‌های ساختِ نرم افزار، برای این زبان را کاسته است. بـه نوبـه‌ی خـود سـاختِ نـرم افـزار رایانـه‌ای گسترش استاندارد آن زبان را در پی داشته است.

۲_ قالب‌های گوناگون پرونده‌های رایانه‌ای

امروزه پرونده‌های نوشتاری فارسی با نرم‌افزارهای گوناگون نوشته می‌شـوند. بـرای ریـشه یـابی واژه‌هـای درون ایـن نوشتارها به کمک رایانه باید قالب این پرونده‌ها خوانده شود. قالب این پرونده‌ها اغلب هم‌خوانی کمی با هم دارند. نرم‌افزارهـای گوناگونی همچون pe2 ، زرنگار ، کِلک ، نشر الف ، word microsoft (که نسخه‌های گونـاگون آن فارسـی را بـه یـک شـکل پشتیبانی نمی‌کنند) pdf ، latex ... برای نوشتن در رایانه به کار گرفته می‌شود. که قالب پرونـده نوشـته شـده در هـر کـدام ویژه‌ی خود آن نرم افزار است. کسانی نیز این نوشته‌ها را به تصویر تبدیل می‌کنند تا خواننده بتواند به سادگی آن‌ها را بر روی هر رایانه‌ای بخواند. آماده کردن یک برنامه‌ی رایانه‌ای که همه‌ی این قالب‌ها را بخواند اگر ناممکن نباشد بسیار سـخت خواهـد بود. پرونده‌های با قالب xhtml (صفحه‌های شبکه جهانی) از جنبه‌های گوناگون بهتر هستند. نخست آن که این پرونده‌ها قالب استاندارد دارند که به سادگی می‌توان واژه‌های درون آنها را با برنامه خواند. دوم، به خوبی از سوی مجمـع جهـانی وب (w3c) پشتیبانی و به روز می‌شود. سوم، کاربرانی بسیاری از آن بهره می‌برند و روز به روز به دامنـه‌ی آنهـا افـزوده مـی‌شـود. چهـارم، توانایی‌ها و امکانات xhtml روز به روز در حال گسترش است و همزمان می‌توان هم برای نمایش و هم برای چاپ از آن کمـک گرفت. البته باید به خوبی با قانون‌هـای آن و صـفحه‌هـای سَـبک (Cascade Style Sheet) آشـنا بـود؛ تـا بتـوان از همـه‌ی توانایی‌های آن سود برد. پنجم، قابلیت حمل (portability) بالایی دارد و به خـوبی بـر روی رایانـه‌هـای گونـاگون و سیـستم عامل‌های گوناگون از آن بهره برد. به‌روزترین روش در این زمینـه بـه کـارگیری xml (eXtensible Markup Language) و تبدیل آن به xhtml به کمک xsl(eXtensible Stylesheet Language) است. که البته برای قالب دهی می‌توان از css یا از xsl-fo کمک گرفت.

۳_ استاندارد خط در رایانه

روند فارسی سازی و استاندارد نمودن خط فارسی برای رایانه فراز و نشیب‌های زیـادی داشـته اسـت. کوچـک‌تـرین واحد نوشته نویسه (character) نامیده می‌شود. نویسه یک حرف ، اِعراب ، علامت نقطه گذاری ، نشانه‌ی بریل یا نماد ریاضی می‌تواند باشد. هر حرف دارای یک یا چند شکلِ نمایش است که شکل (glyph) نامیده می‌شـود. بـرای نمونـه نویـسه‌ی «ی» دارای شکل‌های نمایشی «ی» ، «ی» ، «ی» ، «ی» است. مجموعه‌ی کد به دو گونه تعریف شده است:

الف_ نگاشت میان هر شکل با یک بایت (یا چند بایت پیاپی)

ب_ نگاشت میان هر نویسه با یک بایت (یا چند بایت پیاپی)

مؤسسه‌ی استاندارد و تحقیقات صنعتی ایران در استاندارد ۲۹۰۰ روش الف را برگزید و برای هر شکل یـک نویـسه یـک کـد یک بایتی قرار داد. این مؤسسه در استاندارد ۳۳۴۲ روش ب را برگزید و برای هر نویسه یک کد یک بایتی قرار داد. ایـن روش را روش تک نمادی نیز می‌نامند. شکل نمایش یک نویسه در واژه بستگی به جـای آن نویـسه در واژه و پیونـد پـذیر یـا پیونـد ناپذیر بودنِ حرف دارد. برای نمونه «ی» پیوند پذیر است. شکلهای گونـاگون «ی» در واژه‌هـای «یـک» ، «میـان» ، «یکـی» ، «برای» بستگی به جای آن دارد. بنابراین می‌توان با دسته‌بندی حروف فارسی و به کارگیری الگوریتم با توجه به جای حرف در واژه شکل نمایش آن را شناسایی کرد. ولی به واژه‌ی «خانه‌ها» دقت کنید که در آن می‌خواهیم «ها» در کنارِ (بدون فاصله بـا)

«خانه» باشد و «ه» در پایان خانه به شکل «خانهها» تبدیل نگردد. بنابراین نویسه‌ی فاصله‌ی مجازی (Zero Width Non joiner) پیشنهاد شد. این نویسه پس از «خانه» و پیش از «ها» گذاشته شده است. همچنین در «ه .ش» می‌خواهیم که «ه» به شکل «ه . ش» نوشته نشود. بنابراین نویسه‌ی اتصال مجازی(zero width joiner) پیشنهاد گردید. این نویسه پس از «ه» در «ه .ش» گذاشته شده است تا شکل دلخواه ما به دست آید. امروزه بیشتر روش تک نمادی به کار گرفته می‌شود. شرکت‌های بزرگ دنیا به جای پذیرشِ استانداردِ ایران مجموعه‌ی کد دیگری را به کار گرفتند که بزرگ‌ترین تفاوت آن با استاندارد ۳۳۴۲ مؤسسه‌ی استانداردِ ایران رعایت نکردن ترتیب چهار حرف «پ» ، «چ» ، «ژ» ، «گ» در این مجموعه‌ی کد است. البته با توجه به همه‌گیر شدن این کد به کمک نرم‌افزارهای خارجی در ایران ، استاندارد ایران(حتی در درون کشور) به فراموشی سپرده شد. به همین ترتیب استاندارد ۲۹۰۱ برای صفحه‌ی کلید نیز تا اندازه‌ای به دست فراموشی سپرده شد. البته در برخی از سیستم‌های عامل(مانندِ linux) و برخی نرم‌افزارها(مانندِ unipad) استاندارد صفحه کلید ایران رعایت شده است. چون یک بایت گنجایش همه‌ی نویسه‌های زبان‌های گوناگون را ندارد به هر مجموعه‌ی کد برای یک زبان نامی داده شد. مجموعه‌ی کد عربی (و فارسی) را cp1256 یا windows1256 یا Arabic windows نام نهادند. یکی از دردسرهایِ دیگرِ این مجموعه‌ی کد، گذاشتنِ حرفِ «ی» با دو کد ۲۳۶ و ۲۳۷ در آن است؛ استاندارد به روشنی میان این دو تفاوت گذاشته است. «ي» برای عربی و «ی» برای فارسی در نظر گرفته شده است. با این همه سیستم‌های عامل گوناگون و نرم افزارهای گوناگون بدون توجه به زبان، یکی از این دو را به کار می‌برند و در پردازش نوشته‌های رایانه‌ای فارسی باید دقت نمود. کدهای ۱۵۲ و ۲۲۳ نیز برای «ک» به کار رفته است ولی اغلب برای فارسی ۱۵۲ به کار می‌رود.

با توجه به این‌که یک بایت برای همه‌ی زبان‌های دنیا بسنده نیست ؛ پس به جای یک بایت پیشنهاد شد که دو بایت برای کد کردن نویسه‌ها به کار گرفته شود. این روش کد گذاری(مجموعه‌ی کد) را یونی‌کد(unicode) نامیدند. البته در این کد نیز ترتیب چهار حرف فارسی رعایت نشده است. همچنین مشکلِ حرف‌هایی با چندین کد(و رعایت نکردن فارسی یا عربی بودن آن در نرم‌افزارهای ویرایشگر)، نیز وجود دارد. یونی‌کد با طول دو بایت (ucs2) نامیده شد. گسترش یونی‌کد به چهار بایت، (ucs4) نامیده شد.

اغلب سخت‌افزارها و نرم‌افزارهای موجود برپایه‌ی یک بایت کار می‌کردند؛ همچنین یونی‌کد یک استاندار دو بایتی (یا چهار بایتی ucs4) است؛ پس باید همه‌ی سیستم‌ها باید جایگزین سیستم‌هایی می‌شدند که بتوانند با دو یا چهار بایت کار کنند. تبدیل ناگهانی سیستم‌ها هزینه‌ی سنگینی را دربرداشت. بنابراین تصمیم گرفته شد به گونه کد گذاری شود که سخت‌افزارها و نرم‌افزارهای موجود هم بتوانند دستِ کم با حروف زبان انگلیسی (که برای آن هم ساخته شده بودند) کار کنند. پس باید مجموعه‌ی کدی ساخته می‌شد که برای نویسه‌های زبان انگلیسی (زیر ۱۲۸) یک بایتی می‌بود. با توجه به این محدودیت، تنها راه چاره به کار بردن مجموعه‌ی کدی با طول متغیر بود. ای روش کد گذاری با تعدادِ متغیر بایت، utf-8 نامیده شد. دو بایتِ(یا چهار بایت) یک نویسه در یونی‌کد در utf-8 به کدی با تعداد بایت‌های متغیر(از یک تا حداکثر ۶ بایت) نگاشته می‌شود. تعداد بایت در این نگاشت بستگی به نویسه دارد. اغلب برایِ پردازشِ پرونده‌ای با کد utf-8، کدِ پرونده به یونی‌کد تبدیل می‌شود.

در اینجا تعدادی از قالب‌های استاندارد کد گذاری فارسی آورده شد. تعدادِ زیادی از سندهای رایانه‌ای کدهای ویژه‌ی خود را دارند. برای ریشه‌یابی(یا هر پردازش نوشتار) کد گذاری‌های گوناگون باید به یک کد تبدیل شوند تا بتوان ریشه‌یابی را بر روی واژه‌های آن انجام داد.

۴_ دستورِ خط فارسی

گرچه با کوشش فرهنگستان زبان و ادب فارسی استاندارد یکسانی برای دستورِ خط فارسی آماده شده است ولی هنوز به خوبی بسیاری از این دشواری‌ها در نوشته‌های درسی دیده می‌شود. همچنین ناهماهنگی‌هایی در خود استاندارد دیده می‌شود. نمونه‌هایی که در این نوشتار آورده شده است. بیشتر از کتابهای درسی دوره‌های دبستان، راهنمایی و دبیرستان و همچنین نشریات رسمی فرهنگستان است. چه، اگر بنا بود از کتاب‌های دیگر و یا از سایت‌های شبکه‌ی جهانی نمونه آورده

شود؛ دامنه‌ی بسیار گسترده‌تری از این ناهماهنگی‌ها دیده می‌شد. در [یوسفان،۱۳۸۲] نمونه‌های بیشتری در این زمینه، در دوره‌ی زمانی طولانی‌تری آورده شده است و همچنین دربردارنده‌ی برخی از ناهماهنگی‌های دیگری نیز است که چون اکنون در کتاب‌های رسمی کمتر دیده می‌شود؛ آنها در این نوشتار آورده نشده‌اند. برای برطرف شدن این ناهماهنگی‌ها در کار برنامه‌نویسی، باید همه‌ی حالت‌های ممکن پوشش داده شود. همچنین برای کمک به بهبود این ناهماهنگی‌ها در نوشتار رایانه‌ای فارسی پیشنهادهایی داده شده است.

۴_۱_ «ی» پس از «ه»

یکی از تغییرهایی که در این چند سال در نگارش فارسی به وجود آمده است ، تغییر شکل کسره‌ی اضافه پس از «ه» است. در گذشته با گذاشتن (ٔ) بر سرِ «ه» این کار انجام می‌شد؛ ولی امروزه برای نشان دادنِ کسره‌ی اضافه پس از «ه» ، «ی» به کار گرفته می‌شود.

«زبان فارسی به اندیشه‌ی ما شکل داده است»[شریعت،۱۳۸۲].

«همهٔ تصمیمها یا گزینشهایی که در قسمت خلّاقهٔ مغز فرستنده به عمل آمده»[خبرنامه،۱۰۹-۱۳۸۴]

در [دستورِ خط،۱۳۸۰] صفحه‌ی۲۰، همزه پیشنهاد شده است که در کتاب‌های کنونی درسی به صورت «ی» نوشته می‌شود و همین باعث سردرگمی نویسندگان خواهد شد.

ناهماهنگی دیگری در نوشتن (هٔ) دیده می‌شود که به دلیل به کار بردن کدهای گوناگون در رایانه برای آن به وجود می‌آید. نویسه «ۀ» و دیگری با دو نویسه‌ی «ه» و« ٔ» است. که بهتر است که یکی از این دو روش برای نوشتن پیشنهاد گردد. گرچه به نظر می‌رسد که به کار بردن «ی» در «لانه‌ی گنجنشک» خواناتر و زیباتر باشد و این مشکلات را در برندارد.

۴_۲_ «ها»ی نشانه‌ی جمع

«ها (نشانهٔ جمع) همواره به کلمهٔ پیش از خود می‌چسبد، مانند کتابها، باغها، چاهها، کوهها، گرهها، مگر هنگامی که:

۱.خواهیم صورتِ مفردِ کلمه را مشخص کنیم:کتاب‌ها، درس‌ها، باغ‌ها

۲.کلمه به های غیرملفوظ (بیان حرکت) و یا های ملفوظی که حرفِ قبل از آن حرف متّصل باشد، ختم شود:میو ه‌ها، خانه‌ها ، سفیه‌ها، فقیه‌ها، پیه‌ها، به‌ها»[دستورِ خط،۱۳۸۰].

این سفارش فرهنگستان که مبهم است. در کتاب‌های دوره‌های دبستان ، راهنمایی و دبیرستان «ها» بیشتر جدا نوشته می‌شود. «تم وقتی در طویله کار داشت، این لباسها را به تن می‌کرد. ارباب آهن‌ها را از دست غلام بازکرد.» [رستگارفسایی،۱۳۸۳].

«با زبان‌های بیگانه‌ای که در کار تحقیقی با آن سر و کار دارند»[سمیعی گیلانی،۱۳۸۴].

«روزنامه‌نگاران که غالب کتابهای آن دوره را نوشته‌اند»[خبرنامه،۱۱۰_۱۳۸۴].

۴_۳_ فاصله‌گذاری

«فاصله‌گذاری میان کلمات، خواه بسیط و خواه مرکّب، امری ضروری است که اگر رعایت نشود طبعاً سبب بدخوانی و ابهام معنایی می‌شود.»[دستورِخط،۱۳۸۰]

خوشبختانه امروزه در نوشتار رایانه‌ای این امر تا اندازه‌ای رعایت می‌شود؛ زیرا مفهوم فاصله به خوبی روشن است. بدونِ رعایت کردن فاصله ریشه‌یابی سخت‌تر می‌شود. در واژه‌هایی مانندِ «می‌نویسم» می‌خواهیم که «می» به «نویسم» چسبیده نباشد(مینویسم) و در عین حال بهتر است با آن، فاصله نیز نداشته باشد(می نویسم). در واژه‌هایِ دیگری مانند «علاقه‌مند» , «خانه‌ها» و .... همین نیاز را داریم. یا به طورِ روشن‌تر در بسیاری از واژه‌ها می‌خواهیم میان بخش‌های واژه فاصله گذاشته نشود و در عین حال حروف پیوندپذیر یک بخش از واژه به بخش پس از آن نچسبد. «همانطور که متصل کردن

دو حرف نشانه‌ی اتصال است (مانند حرف ب در کلمه‌ی با) فاصله‌ی کوتاه میان دو حرف هم نشانه‌ی اتصال است. چنان که در این کلمات می‌بینیم: روزنامه ، یادداشت ، هم‌منزل»[وحیدیان کامیار،۱۳۸۴].

در این حالت‌ها میان این دو واژه فاصله‌ی مجازی گذاشته می‌شود. همین مشکل در «مند»، ... نیز وجود دارد. پیشنهاد می‌شود که در دستورِ خط فارسی بخش‌هایی برای رایانه گذاشته شود یا اینکه جزوه‌ی جداگانه‌ای در این زمینه آماده شود. این جزوه (یا بخش‌های درون دستورِ خط) دربردارنده‌ی قانون‌هایی از این دست باشد. به این ترتیب در نوشتار رایانه‌ای نیز یکسان سازی انجام خواهد شد.

اگر میان واژه‌های درونِ یک واژه‌ی مرکب فاصله‌ی معمولی نباشد، پردازش رایانه‌ای نوشتار ساده‌تر خواهد شد(به ویژه در پردازش ساختاری جمله). اگر حرفِ پایانی یکی از واژه‌های میانی پیوندپذیر باشد؛ می‌توان میان این دو واژه فاصله‌ی مجازی گذاشت. برای نمونه در «گیله‌مرد» میان دو بخش یک فاصله‌ی مجازی گذاشته شده است. همچنین در حالت‌هایی که کنارِ هم گذاشتن واژه‌های درون واژه‌ی مرکب باعث ابهام شود؛ فاصله‌ی مجازی میان بخش‌های واژه‌ی مرکب گذاشته شود. به عبارت دیگر، اگر بتوان یک واژه‌ی مرکب را به چند گونه تجزیه کرد، باید میان واژه‌های آن (که موردِ نظرِ نویسنده است) فاصله‌ی مجازی گذاشت. در این‌باره نیز بهتر است، قانونِ مشخصی نوشته شود. همچنین اگر برای نوشتن عددها نیز این قاعده رعایت شود؛ بهتر است. برای نمونه به جایِ «سی و سه»، «سی‌وسه» نوشته شود. با وجود پیشنهادهایی که داده می‌شود؛ بر این باور هستم که متخصصان رایانه نباید باعث دگرگونی دستورِ خط شوند و چیزی را به زبان تحمیل کنند. وظیفه‌ی متخصصِ رایانه برآورده کردن خواسته‌های زبان است و نه دگرگونی در آن. پس استادانِ ادبیات و دستورِ زبان فارسی و همچنین زبان‌شناسان فارسی در این‌باره تصمیم‌گیری می‌کنند.

۴_۴_ واژه‌های مرکب

درباره‌ی چگونگی نگارشِ واژه‌های مرکبِ فارسی، دیدگاه‌های گوناگونی وجود دارد. حتی در این زمینه که واژه‌هایی را با هم در نظر بگیریم و آن‌ها را مرکب بنامیم، نظراتِ گوناگونی وجود دارد (به ویژه در فعل مرکب). با همه‌ی کوششی که برای یکسان سازی در نوشتن این واژه‌ها انجام شده است؛ هنوز ابهام‌ها و ناهماهنگی‌های فراوانی دیده می‌شود.

صفحه‌ی ۱۲۶ از [حق‌شناس،۱۳۸۴] : «در همان چاپخانه گرد آمدند».

در صفحه‌های ۱۱۵ و ۱۵۸ همان «کتاب‌شناسی، کتاب‌نامه، کتاب‌خوان، کتاب‌نویس، کتاب‌فروشی»

صفحه‌ی ۸۷ از [سنگری،۱۳۸۴] : «آن‌جا هوای خوب و دلپذیری داشت.»

صفحه‌ی ۱۱۹ همان: «از پدر و مادرِ دل‌سوز خود، سپاس گزاریم»

صفحه‌ی ۱۳۳ از [حاجیان۱۳۸۳]: «اسم این بزغاله‌ها را به دلخواه انتخاب کن»

این نمونه‌ها فقط از کتاب‌های درسی فارسی آورده شد. تفاوت‌ها در دیگر کتاب‌های درسی (ریاضی، جغرافی، ...) آشکار است. همچنین در کتاب‌های غیر درسی بیشتر و در روزنامه‌ها و مجلات بسیار بیشتر است؛ در صفحه‌های فارسی شبکه‌ی جهانی نیز که هیچ هماهنگی دیده نمی‌شود. آماده کردن یک فهرست کامل از این واژه‌های مرکب در فرهنگستان و شکل پیشنهادی آن بسیار شایسته است. در این صورت روش نوشتن آن‌ها سلیقه‌ای نخواهد بود.

۴_۵_ حرکت گذاری در نوشتار فارسی

«در خط فارسی افزون بر حرف‌های الفبا نُه نشانه خطی دیگر نیز به کار می‌رود. این نشانه‌ها ـَ ـُ ـِ ـً ـٌ ـٍ ـّ آ هستند. کاربرد این نشانه‌ها کم است؛ زیرا در خط فارسی حرکت گذاری به کار برده نمی‌شود. در نوشتن واژه‌ها از میان نشانه‌های نه‌گانه بالا مد، تشدید، تنوین نصب (اً) بیشتر کاربرد دارند . تنوین رفع و جر (ٍ) تنها در واژه‌های عربی رایج در فارسی به کار می‌رود و دیگر نشانه‌ها را در جاهایی به کار می‌بریم که رعایت نکردن آنها ابهام و بدفهمی به وجود می‌آورد» [حق شناس،۱۳۷۷].

این نشانه‌ها نیز دشواری دیگری در ناهماهنگی در نگارش فارسی به وجود آورده‌اند، مانند «رَفت» و «رُفت» که تنها فرق آنها در ـَ و ـُ است که بر سر «ر» گذاشته شده است . ولی رُفتگر را بیشتر در نگارش بدون ـُ می‌گذارند و این کار ریشه یابی را سخت‌تر می‌کند. البته می‌دانیم که «رَفتگر» نداریم که باید به صورت استثنا به رایانه داده شود. بنابراین با توجه به زبان فارسی تعداد این استثناها بسیار زیاد خواهد بود.

خوشبختانه با گسترش رایانه‌ها و در دسترس بودن نرم‌افزارهای توانمند نگارش و ویرایش و دامنه‌ی بزرگی از نویسه‌ها که این نرم افزارها پشتیبانی می‌کنند از این دشواری کمی کاسته شده است و گذاشتن این نشانه‌ها نیز ساده‌تر گشته است. گرچه هنوز نمی‌توان به درستی گفت که کجا باید این نشانه‌ها رعایت شوند.

۵_ دگرگونی در واژه‌ها هنگامِ پیوند

در هنگامِ پیوند واژه‌ها یا پیشوند به واژه یا پسوند به واژه، نیز تغییرهای گوناگونی رخ می‌دهد. در زیر نمونه‌هایی از این تغییرها نشان داده شده است.

«زنده»+«ان»←«زندگان» ؛ «زنده»+«م»←«زنده‌ام»؛ «گو»+«م»←«گویم»

«ن»+«افتاد»←«نیفتد»؛ «ن»+«آزما»←«نیازما»؛ «زنده»+«ها»←«زنده‌ها»

که متخصص رایانه در هنگام ریشه‌یابی باید این‌ها را در نظر بگیرد. همین مشکل در واژه‌های پربسامد(سراسری،عمومی،stopword) پیش می‌آید که در [یوسفان،۱۳۸۴] توضیح داده شده است.

۶_ واژه‌های زبان‌های دیگر در فارسی

وجود دامنه‌ی گسترده‌ای از واژه‌های زبان‌های دیگر در زبان فارسی (مانند واژه‌های زبان‌های عربی، انگلیسی، ترکی، مغولی و فرانسوی یا واژه‌های دیگر زبان‌های بیگانه‌ی آورده شده به زبان فارسی) ریشه‌یابی رایانه‌ای را بسیار سخت می‌کند. برخی از این واژه‌ها به ساختارهای واژه‌های زبان فارسی (برای نمونه یک کلمه‌ی مرکبِ فارسی) نزدیک هستند، بنابراین این احتمال وجود دارد که ریشه‌یاب به نادرستی آن‌ها را ریشه یا مشتق یک واژه‌ی فارسی در نظر بگیرد. برای نمونه «ایدئالیست» یا «تایپیست» را می‌توان ترکیب «ایدئالی»(او آدمِ ایدئالی است) و «ست» (است) پنداشت و آن را با این روش به دو بخش شکست. برای بسیاری دیگر از واژه‌ها نیز مشابه این مشکل پیش می‌آید. می‌توان به کسانی که با رایانه کار می‌کنند ؛ پیشنهاد داد که به گونه‌ای واژه‌های زبان‌های دیگر را مشخص نمایند. برای نمونه در هنگام ساختن صفحه‌ی وب که با قالب html است می‌توان از

تایپیست لاتُعَدُّولاتُحْصی

یا روش دیگری برای مشخص نمودن واژه‌های زبان‌های دیگر در زبانِ فارسی به کار گرفته شود.

۷_ لغت‌نامه

در هنگام انجام ریشه‌یابی به یک لغت‌نامه‌ی شایسته نیاز است. ولی پایگاه واژه‌های یک لغت‌نامه‌ی فارسی در دسترس نبود. به کارگیری پایگاه واژه‌ی نرم افزارهای در دسترس (یا روشن‌تر ، شکستن پایگاه داده و برداشت واژه) نیز به دلیل تجاری، نادرست و غیرقانونی بود. که در آن زمان، خرید پایگاه واژه‌ی این نرم افزارها نیز به دلیلِ هزینه‌ی آن امکان‌پذیر نبود. نوشتن (تایپ) یک لغت‌نامه‌ی کوچک نیز در دستور کار قرار گرفت ولی به پایان رساندن آن ناممکن و بسیار زمان‌گیر می‌نمود. پس در [یوسفان۱۳۸۲] روشی به کار گرفته شد که فشرده‌ای از آن آورده می‌شود.

۷_۱_ گردآوری واژه

نخست باید به گونه‌ای فهرستِ کاملی از واژه‌های فارسی روزمره گردآوری می‌شد. پس از کوشش‌های فراوان بهترین منبع برای گردآوری واژه‌های فارسی روزمره، روزنامه‌های ایرانی در دسترس روی شبکه جهانی تشخیص داده شد. (شاید نقل نوشته‌ای از روزنامه‌ای بدون آوردن نام منبع غیرقانونی باشد ولی احتمالاً به کارگیری واژه‌های درون آنها کار نادرستی نیست). به کمك نرم‌افزار رایگان httrack هزاران صفحه‌ی روزنامه گردآوری شد (نزدیك به یک GB نوشته)، کُد نوشتاری این صفحه‌ها CP1256 (یا Windows1256) بود. واژه‌های فارسی درون این نوشته‌ها و فراوانی آنها درآورده شد. چون واژه‌هایی که فراوانی بسیار کمی دارند به احتمال زیاد نادرست (اشکال تایپی یا نگارشی) هستند یا واژه‌های بسیار تخصصی هستند؛ بنابراین واژه‌های با فراوانی بسیار کم دور ریخته شدند.

۷_۲_ شناسایی ریشه‌ی فعل‌ها

به خاطر در دسترس نبودن لغت‌نامه تصمیم گرفته شد که برپایه‌ی برخی قاعده‌ها رده‌ی همه‌ی واژه‌ها شناسایی شود. در زبان فارسی شناسایی رده‌ی یک واژه (فعل ، اسم ، ...) به سادگی امکان‌پذیر نیست. برای نمونه به کارگرفتن گزاره‌ی ساده‌ی «فعل کلمه‌ای است که شناسه می‌پذیرد.» , به هیچ روی نمی‌تواند فعل‌ها را شناسایی کند. زیرا بسیاری از اسم‌ها و صفت‌ها نیز شناسه می‌گیرند. برای نمونه اغلب «خوبیم» به جای «خوب هستیم» و «عبارتند» به جای «عبارت هستند» به کار گرفته می‌شود. فعل‌ها در زبان فارسی رده‌ی بسیار بزرگی هستند که به کمک آن‌ها بسیاری از اسم‌ها و صفت‌ها ، ... ساخته می‌شوند. بنابراین با شناسایی این رده ، می‌توان بسیاری از واژه‌های فارسی را ریشه‌یابی نمود.

« در فعل‌های ساده پس از حذف « ـَن » از مصدر ، بُن ماضی باقی می‌ماند و از جهت تغییری که از بُن ماضی به بُن مضارع انجام می‌گیرد ، آنها را می‌توان در هشت گروه جای داد. شکل زیر تغییرهای فعل‌ها را در گروه‌های هشت‌گانه نشان می‌دهد. » [انوری،۱۳۸۰]

شماره‌ی گروهها	حروف پایانی بن‌ماضی+پسوند	حروف پایانی بن‌ماضی	بن مضارع	مثال	
۱	یدن	ـ ید	پس از حذف «ـید» باقی‌مانده بن مضارع است.	نال+ـید+ـَن	نالیدن نال/نالید
۲	دن	د	پس از حذف «د» باقی‌مانده بُن مضارع است.	خور+د+ـَن	خوردن خور/خورد
۳	ودن	ـود	پس از حذف «و» ، «و» به «ا» تبدیل می‌شود.	آزمو+د+ـَن	آزمودن آزما/آزمود
۴	ادن	ـاد	پس از حذف «اد» باقی‌مانده بن مضارع است.	افت+اد+ـَن	افتادن افت/افتاد
۵	ختن	ـخت	«ت» حذف و «خ» تبدیل به «ز» می‌شود.	ساخ+ت+ـَن	ساختن ساز/ساخت
۶	ستن	ـست	پس از حذف «ست» باقی‌مانده بن مضارع است.	آرا+ست+ـَن	آراستن آرا/آراست
۷	شتن	ـشت	پس از حذف «ت» ، «ش» تبدیل به «ر» می‌شود.	کاش+ت+ـَن	کاشتن کار/کاشت
۸	فتن	ـفت	پس از حذف «ت» ، «ف» تبدیل به «ب» می‌شود.	تاف+ت+ـَن	تافتن تاب/تافت

شکل ۱ـ هشت گروه فعلهای فارسی

تعداد کمی از فعل‌های فارسی از قاعده‌های این هشت گروه پیروی نمی‌کنند. بُن گذشته و غیر گذشته‌ی این فعل‌ها جداگانه نوشته شدند. این روش بر روی گردایه‌ی بزرگ واژه‌ها به کار گرفته شد. بُن گذشته و غیر گذشته‌ی بیشترِ فعل‌های ساده‌ی فارسی به خوبی شناسایی شدند.

۸_ روشِ ریشه‌یابی

در این‌جا به روش‌های موجود ریشه‌یابی و دشواری‌های آن‌ها پرداخته می‌شود. همچنین روشی پیشنهاد شده و پیاده سازی شده است. این روش به صورت وارون کار می‌کند به این معنا که مشتق فعل را می‌سازد و در صورت درست بودن آن مشتق ، از آن پس ریشه‌ی آن مشتق بُن فعل خواهد بود.

۸_۱_ ریشه‌یابی به کمک روش‌های آماری

در این روش (یا دسته از روش‌ها) یک گردایه‌ی بزرگ از واژه‌ها با ساخت‌های گوناگون گردآوری می‌شود. هر چه این گردایه بزرگ‌تر و کامل‌تر باشد این ریشه‌یاب‌ها بهتر کار می‌کنند. در این روش تحلیل آماری به کار گرفته می‌شود . با روش آماری وندهایی که در واژه‌ها تکرار شده‌اند، شناسایی می‌گردند. این روش در [Manning,1999] آورده شده است. این روش به زبان بستگی ندارد و این بزرگ‌ترین برتری این روش می‌باشد. در بیشترِ زبان‌های هند و اروپایی، اغلب بر پایه‌ی وند اشتقاق انجام می‌شود. اگر این روش بتواند برای زبان انگلیسی پاسخ شایسته‌ای بدهد ؛ گسترش آن به دیگران زبان‌های دسته‌ی هند و اروپایی ساده خواهد بود. این روش با سه مشکل بزرگ روبروست :

الف ـ در این روش به یک گردایه‌ی بزرگ از واژه‌ها نیاز است. این گردایه باید کامل باشد و واژه‌های درون آن نیز درست باشند. وجود واژه‌های نادرست در گردایه بر کارآیی این ریشه‌یاب اثر بسیار بد می‌گذارد و آن را گمراه می‌کند. گردآوری گردایه‌ی بزرگی از واژه‌های صد در صد درستِ فارسی نیز، ناممکن می‌نماید.

ب ـ هنوز این روش‌ها در حال آزمایش هستند و کارآیی آنها چشم‌گیر نیست.

ج ـ این روش‌ها نیاز به رایانه‌های با سرعت زیاد و حافظه بزرگ دارند و اجرای برنامه‌های نوشته شده بر پایه‌ی این روش‌ها بسیار زمانبر است. برای اجرای این روش‌ها با رایانه‌های در دسترس باید تعدادی از آن‌ها با هم موازی شوند و شاید برای یک بار اجرا ، چند روز زمان گرفته شود. گرچه در پیاده سازی این روش‌ها بهتر می‌توان به نیازهای آنها پی‌برد.

۸_۲_ ریشه‌یابی به کمک روش porter یا شبیه به آن

روش porter یک روشِ توانمند و در عینِ حال یکی از قدیمی‌ترین روش‌های ریشه‌یابی در زبان انگلیسی است. این روش بر پایه‌ی زبان‌شناسی و دسته‌بندی واژه‌ها به کمک واج‌ها و هجاها بنا نهاده شده است. پس از آن وندهای واژه‌های درون گردایه به طور خودکار برداشته می‌شوند. این روش در [porter,1980] توضیح داده شده است.

به طور کلی می‌توان روش‌های ریشه‌یابی به کمک قاعده‌های زبان را در ادامه‌ی کارِ همین روش دانست. برای نمونه می‌دانیم که «گفت» ریشه‌ی گذشته‌ی یک فعل است و ریشه غیرِگذشته آن «گو» می‌باشد ؛ بنابراین با نگاهی به دستور زبان در می‌یابیم که ریشه همه‌ی واژه‌های زیر «گفت» می‌باشد.

«گفتم» ، «می‌گفتم» ، «نگفتم» ، «گفته‌ام» ، «می‌گوید» ، «گوینده» ، «گفتار» و ...

در ضمن به خاطر رعایت نشدن دستور خط باید «میگفتم» ، «میگفت» و ... را نیز همین گونه ریشه‌یابی کرد. پس باید قاعده‌های بسیاری نوشته شود و این قاعده‌هایِ دستور زبان به کمک برنامه‌نویسی پیاده سازی شود. برای ریشه‌یابی فعل‌های فارسی باید یک زبان برنامه‌نویسی شایسته برگزیده شود. هنگام برنامه‌نویسی، باید حالت‌های گوناگونی که هنگام ترکیب واژه‌های فارسی پیش می‌آید ، پوشش داده شود. همچنین ویرایش‌های پیچیده و بسیار زیادی بر روی قاعده‌هایی از دستور زبان فارسی که به کار گرفته شده بود ، باید انجام شود. به عبارت دیگر در هنگام برنامه‌نویسی ، پیاپی بسیاری از بخش‌ها دگرگون می‌شود. به هیچ روی این امکان وجود ندارد که یک روند خطی برای طراحی و پیاده سازی نرم افزار در نظر گرفته شود. با توجه به این دشواری‌ها یک زبان برنامه‌نویسی بسیار ساده و انعطاف پذیر و توانمند نیاز است. این زبان باید استاندارد یونی‌کد را به خوبی و سادگی پشتیبانی کند. در این زبان باید کار با رشته‌های یونی‌کدی نیز بسیار ساده باشد. دقت کنید که

اگر ماشین پذیرنده‌ی متناهی(Deterministic Finite Automata) نیز طراحی شود و سپس بر پایه‌ی آن برنامه‌نویسی انجام گیرد؛ همواره این امکان وجود دارد که در هنگام طراحی ماشین پذیرنده‌ی متناهی قاعده‌هایی در نظر گرفته نشوند و پس از پایان کار برنامه‌نویسی چنین قاعده‌هایی باید به ماشین پذیرنده افزوده شوند که به این ترتیب باید دوباره کد برنامه تغییر یابد. به این ترتیب روند نگهداری از برنامه‌ی ریشه‌یاب و گسترشِ آن بسیار هزینه بر می‌شود.

۸_۳_ شبه‌زبان برنامه‌نویسی جدید

امروزه با وجود همه‌ی زبان‌های برنامه‌نویسی در دسترس، شرکت‌های بزرگ رایانه‌ای به این نتیجه رسیده‌اند که برای برخی از کارهای بسیار ویژه و گسترده‌ی خود یك زبان ویژه برای آن کار طراحی کنند. گرچه ساختِ یك زبان برنامه‌نویسی جدید زمان و نیروی زیادی می‌گیرد ولی برای کارهای بسیار ویژه که نیاز به ویرایش‌های تخصصی فراوانی در آینده در آن‌ها هست؛ این هزینه‌ی گزاف در هنگام به کار بستن ویرایش‌ها (نگهداری و گسترش سیستم) به خوبی جبران می‌شود.
بنابراین یک شبه زبان برنامه نویسی جدید طراحی شد. این شبه‌زبان بسیار به قالبی که دستور نویسان و زبان‌شناسان برای نوشتن دستور زبان فارسی به کار می‌برند نزدیك است. سادگی این شبه‌زبان به گونه‌ای است که بدون دانستن هیچ زبان برنامه‌نویسی دیگر می‌توان این شبه‌زبان را به کار گرفت. انگیزه‌ی این کار این بوده که شکاف کنونی میان متخصصان زبان فارسی و متخصصان رایانه کاسته شود. این چنین می‌نماید که برای چیره شدن بر پیچیدگی‌های فراوان کارهای رایانه‌ای درباره‌ی زبان فارسی نیاز به همکاری بیشتر میان متخصصان این دو رشته است. نحوِ این شبه‌زبان برنامه‌نویسی و علامت‌ها و عملگرهای تعریف‌شده در آن ، بر پایه‌ی شیوه‌ی نگارش [رحیمیان،۱۳۸۱] است. در [یوسفان،۱۳۸۲] به طور کامل این شبه‌زبان توضیح داده شده است. در [یوسفان،۱۳۸۴] اندکی تغییر یافته است و در شناسایی واژه‌های سراسری به کار گرفته شد. در این‌جا فقط چند خط نوشته شده به این شبه زبان آورده شده است. به خاطر سادگی بیش از اندازه‌ی این شبه‌زبان و نزدیکی آن به روش نوشتن دستورِ زبان، به سادگی می‌توان نحو (syntax) آن را یاد گرفت.

«کجا» + شناسه گذشته/ «ست» / نشانه_ها/∅ .

ضمیر_متصل_مفعولی= «م» / «ت» / «ش» / «مان» / «تان» / «شان».

نشانه_جمع = «یم» / «ید» / «ند». /* همانند بسیاری از زبان‌های برنامه نویسی، می‌توان اینجا توضیح نوشت */

وابسته_ان_جمع= «یست» / «ست» / «ی» + «که» /∅ .

نشانه_ان_جمع= «ان» + شناسه_جمع / ضمیر_متصل_مفعولی / وابسته_ان_جمع/∅.

۸_۴_ ساخت مشتقات فعل‌های فارسی به کمک شبه زبان

به جای روش سر راست برای ریشه‌یابی ، روشی وارون به کار گرفته شد. در ادامه به این روش پرداخته می‌شود.
به کمک فهرست فعل‌های شناخته شده و با داشتن ریشه‌ی گذشته و غیرِ گذشته‌ی آن‌ها دو متغیرِ دیگر به شبه زبان افزوده شد:

الف- «بُن_گذشته» که نشان دهنده‌ی بُن‌های گذشته‌ی فعل‌های شناسایی شده است. به عبارت دیگر در هنگام اجرای برنامه به شبه زبان ، به جای این متغیر , بُن‌های گذشته‌ی همه‌ی فعل‌های شناسایی شده گذاشته می‌شود.

ب- «بُن_غیر_گذشته» که نشان دهنده‌ی فهرست بُن‌های غیر گذشته‌ی فعل‌های شناسایی شده است.

به کمکِ شبه‌زبان و با به کارگیری قاعده‌های دستور زبان فارسی، مشتقات همه‌ی فعل‌های فارسی ساخته شدند. همزمان نگاشت میانِ ریشه‌ی فعل و واژه‌های مشتقِ ساخته شده ، نگهداری شد. برای نمونه دو دستور از دستورهایی که برای ساخت مشتقات نوشته شده است ، در زیر آورده شده است.

بُن_گذشته + «گار» + نشانه_ان_جمع/نشانه_یستکه/نشانه_ها /شناسه_گذشته /ضمیر_متصل_مفعولی/«ا»/∅ .

/* صفت لیاقت و فاعلی می‌سازد و برخی مفهوم اسمی دارند .

آفریدگار ، کردگار ، پروردگار ، رستگار ، ماندگار */

بُن_غیر_گذشته + «گار» + نشانه_یستکه/نشانه_ان_جمع/نشانه_ها/نشانه_تر/نشانه_ترین/«ا»/ شناسه_گذشته/∅ .

/* آموزگار ، سازگار ، آمرزگار */

۸_۵_ آزمون درستی واژه‌ها

تا این‌جا فهرست همه‌ی مشتقات فعل‌های فارسی شناسایی شد. ولی در این‌جا بسیاری از واژه‌های ساخته شده ، نادرست هستند. برای نمونه در زبان فارسی «پروردگار» (بُن_گذشته + «گار») وجود دارد ، در حالی که «گفتگار» وجود ندارد و به عبارت دیگر قانون‌های زبان فارسی استثناهای فراوان دارند. به کمک واژه‌های گردآوری شده از شبکه‌ی جهانی، درستیِ واژه‌های ساخته شده آزموده شد. مشتقات ساخته شده‌ای که درون این گردایه‌ی بزرگ از واژه‌ها وجود نداشتند ، دور ریخته شدند. به این ترتیب فهرستِ مشتقات فعل‌های فارسی و نگاشت آن‌ها به ریشه‌ی فعل آماده شد. پس از مرتب سازی این فهرست ، روند ریشه‌یابی در کمترین زمان ممکن انجام می‌شود.

۸_۶_ فهرست مشتقاتِ فعل‌های فارسی به همراه ریشه‌ی آن‌ها

همانگونه که دیده می‌شود فهرست ساخته شده برای کمک به ریشه‌یابی ، جدای از سیستمی که نیاز به ریشه‌یابی دارد، آماده شد. سپس در هنگام نیاز به ریشه‌یابی برای یک واژه در سیستم پایانی ، تنها یک جستجوی دودویی در آرایه (یا در پرونده) انجام می‌شود. در صورت وجود آن واژه در مشتق‌ها , به کمک نگاشت به ریشه‌ی آن واژه ، ریشه‌ی واژه برگردانده می‌شود و در صورتی که آن واژه در فهرست یافت نشود ، خود واژه برگردانده می‌شود.

مراجع

- احمدی گیوی ۱۳۸۰، حسن. دستور تاریخی فعل. تهران: نشر قطره، ۱۳۸۰، جلد اول، ص ۲۷.
- انوری ۱۳۸۰، حسن،و احمدی گیوی، حسن. دستور زبان فارسی ۲. تهران: انتشارات فاطمی ، چاپ بیست و یکم ۱۳۸۰.
- رستگارفسایی ۱۳۸۳،منصور، و حسین داودی، حسن ذوالفقاری، محمدرضا رزسنج،محمدرضا سنگری، عزیز شبانی، غلامرضا عمرانی، حسین قاسم‌پور، افضل مقیمی ، اکبر میرجعفری. ادبیات فارسی ۲، سال دوم دبیرستان. تهران: شرکت چاپ و نشر کتاب‌های درسی ایران، چاپ هفتم ۱۳۸۳.
- حاجیان ۱۳۸۳، فردوس، و دیگران. فارسی اول دبستان، بنویسیم. تهران: شرکت چاپ و نشر کتاب‌های درسی ایران، چاپ پنجم ۱۳۸۳.
- حق شناس ۱۳۷۷، محمد علی ، و دیگران. زبان فارسی (۱) و (۲) ، سال اول نظام جدید آموزش متوسطه . تهران: شرکت نشر و چاپ کتابهای درسی ایران.
- خبرنامه ۱۰۹_۱۳۸۴، گزارش دومین نشست ادب معاصر، خبرنامه شماره ۱۰۹ فرهنگستان زبان و ادب فارسی ، بدون نام نویسنده، صفحه‌ی ۳، شهریور ۱۳۸۴، از سایت http://persianacademy.org/khabarname.htm ، آخرین تاریخ بازدید ۸۵/۱/۲۲.
- خبرنامه ۱۱۰_۱۳۸۴، گزارش دومین نشست ادب معاصر، خبرنامه شماره ۱۰۹ فرهنگستان زبان و ادب فارسی ، بدون نام نویسنده، صفحه‌ی ۱، مهر ۱۳۸۴، از سایت http://persianacademy.org/khabarname.htm ، آخرین تاریخ بازدید ۸۵/۱/۲۲.
- دستور خط ۱۳۸۰، فرهنگستان زبان و ادب فارسی. از سایت http://persianacademy.org ، آخرین تاریخ بازدید ۸۵/۱/۲۲.

- سمیعی گیلانی۱۳۸۴، احمد۱۳۸۴،پژوهشگر نمونه - پژوهشگر هرزه کار، نامه‌ی فرهنگستان، تهران:فرهنگستان زبان و ادب فارسی، دوره‌ی هفتم شماره‌ی دوم، شهریور۱۳۸۴.
- سنگری۱۳۸۴، محمدرضا، و دیگران. فارسی اول دبستان، بخوانیم. تهران: شرکت چاپ و نشر کتاب‌های درسی ایران، چاپ ششم، ۱۳۸۴.
- شریعت۱۳۸۲، محمد جواد، و دیگران. فارسی سال سوم دوره‌ی راهنمایی تحصیلی . تهران: شرکت چاپ ونشر کتابهای درسی ایران . چاپ هشتم ۱۳۸۲.
- وحیدیان کامیار۱۳۸۴، تقی، و دیگران. زبان فارسی۲، سال دوم دبیرستان. تهران: شرکت چاپ و نشر کتاب‌های درسی ایران، چاپ هشتم ۱۳۸۴.
- یوسفان۱۳۸۲، احمد. يك سيستم بازيابى اطلاعات متنى براى زبان فارسى بر پايه نمايه گذارى معانى پنهان. شيراز : دانشگاه شيراز ، شهريور ۱۳۸۲.
- یوسفان۱۳۸۴، احمد، و منصور ذوالقدری جهرمی، مهدی احمدی. روش خودکار شناسایی واژه‌های پربسامد در زبان فارسی. کنفرانس بین المللی اطلاعات و دانش (IKT2005)، تهران:امیرکبیر، خرداد ۱۳۸۴.

References

Manning, Chris, Hinrich Schütze(1999). Foundations of Statistical Natural Language Processing. MIT Press. Cambridge, MA: May 1999.

Porter, M.F.(1980). "An algorithm for suffix stripping". Program, volume 14, number 3, 130-137.