شماره مستند: ۱۹۰/۲۵۳۷/۱/۲

جمهوری اسلامی ایران

دبیرخانه شورای عالی اطلاع رسانی

بررسی ابعاد تعیین قلمرو «کلمه» برای پیاده‌سازی ریخت‌شناسی برای پیکره متنی زبان فارسی

نسخه ۱.۰

دانشگاه علم و صنعت ایران

فروردین ۸۸

فهرست مطالب

عنوان **صفحه**

مقدمه

۱-۱. مقدمه

از آن جایی که جمله، یک واحد متنی پایه است و بلافاصله بعد از کلمه و عبارت قرار می‌گیرد، تعیین محدوده جمله یک مسئله اساسی برای بسیاری از کاربردهای پردازش زبان طبیعی[۱]، مانند تجزیه[۲]، استخراج‌اطلاعات[۳]، ماشین‌ترجمه[۴]، خلاصه‌سازی[۵] و خلاصه‌گیری[۶]، ساخت پیکره‌های متنی[۷] زبان و برچسب-گذاری[۸] نحوی و معنایی و ... است و علاوه بر این، برچسب‌گذاری اجزای کلام[۹] شدیداً نیازمند تعیین دقیق حدود جملات است[۱]. دقت این سیستم به طور مستقیم روی کارآیی برنامه‌های کاربردی اثر می‌گذارد. اگرچه قطعه‌بندی جمله می‌تواند از روی علائم نشانه‌گذاری مثل نقطه یا کوتیشن به دست آید، اما ابهام‌های زیادی هنوز در متون واقعی باقی می‌ماند.

جمله مهمترین واحد در بسیاری از کارهای پردازش زبان طبیعی است. برای مثال، تنظیم جملات[۱۰] در اسناد چند زبانه موازی[۱۱] نیاز دارد که اول محدوده جملات به روشی برچسب‌گذاری شود[۲].

بیشتر برچسب‌های اجزای کلام، به حذف ابهام محدوده جملات در متن ورودی نیاز دارند، معمولاً این کار با درج یک رشته کاراکتر منحصر به فرد در پایان هر جمله انجام می‌گیرد. بدین‌گونه است که ابزار[۱۲] تحلیل پردازش زبان طبیعی جملات منحصر به فرد را به راحتی می‌توانند تشخیص دهند.

تعیین حدود جمله در زمره مسائل پیش‌پردازش زبان فارسی قرار می‌گیرد که کاری روی آن انجام نشده است و در جاهایی که نیاز به این کار بوده، از علائم نشانه‌گذاری استفاده شده است. در این پایان‌نامه چند روش جهت شناسایی حدود جمله در متن فارسی ارائه شده و این روش‌های پیشنهادی مورد بررسی و تجزیه و تحلیل قرار داده شده است. این روشها عبارتند از بررسی ساختاری جمله و تعیین حدود جمله با استفاده

---

[1] Natural Language Processing (NLP)
[2] Parsing
[3] Information Extraction
[4] Machine Translation
[5] Abstraction
[6] Summarization
[7] Corpus
[8] Tagging
[9] Part-of-Speech tag (POS tag)
[10] Alignment
[11] Parallel Multi-Lingual Corpora
[12] Tools

از یک روش کارای تشخیص فعل در جمله، استفاده از مدل چند-تایی در تعیین حدود جمله، استخراج خصیصه از جملات و به کاربردن روشهای رده‌بندی، چون شبکه‌های عصبی ، برای تعیین حدود جمله.

مفاهیم اولیه مورد نیاز و مشکلات و چالشهای زبان فارسی در فصل دوم مورد بررسی قرار گرفته و در ادامه آن مسئله تعیین حدود جمله به‌طور کامل تشریح شده و مورد بررسی قرار می‌گیرد.

تعیین حدود جمله، یکی از مهم‌ترین مراحل پردازش لغوی که از جمله مراحل پیش‌پردازش در اکثر کارهای متن‌کاوی و پردازش زبان طبیعی است، می‌باشد. در این زمینه تابه‌حال در مورد زبان فارسی هیچ کاری صورت نگرفته است، اما در مورد زبان‌های دیگر، مثل انگلیسی، پرتغالی و ژاپنی و ...، کارهای زیادی با استفاده از روشهای مختلف صورت گرفته است. در این کارها مسئله تعیین حدود جمله تبدیل به مسئله رفع ابهام علائم نشانه‌گذاری شده است و با آن به صورت یک مسئله رده‌بندی پایه برخورد شده است. چندی از کارهای مهم انجام شده در زبان‌های مختلف در فصل سوم مورد بررسی قرار داده شده و در فصل چهارم یک سری روش‌های پیشنهادی جهت تعیین حدود جمله در متن فارسی ارائه و مورد بررسی قرار گرفته است. فصل پنجم به پیاده سازی و ارزیابی روش‌های ارائه شده و مقایسه‌ای بین آنها پرداخته است. در فصل آخر نتیجه‌گیری کارهای انجام شده به همراه راه‌کارهای آینده آن مورد بررسی قرار گرفته است

مفاهیم اولیه و زمینه‌ی تحقیق

۱-۲. پردازش متون فارسی

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

متأسفانه این درخت تنومند امروزه 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. برای نمونه در نوشتار امروزی انگلیسی کمتر حرفها به هم چسبیده نوشته می‌شوند و واژه‌ها و اصطلاحهای کمی، بویژه در نوشته‌های علمی، به کار گرفته می‌شود. ویراشهای انجام شده در زبان انگلیسی بسیار بر کارهای رایانه‌ای، که بر پایه‌ی زبان انگلیسی هستند، اثر داشته است و به پیشرفت نرم افزارهای رایانه‌ای کمک نموده است. پیرایش‌هایی که در زبان انگلیسی انجام شده است، بسیاری از پیچیدگی‌های ساخت نرم افزارهایی برای این زبان را کاسته است و به نوبه‌ی خود ساخت نرم افزار رایانه‌ای گسترش استاندارد آن زبان را در پی داشته است.

در پردازش متون زبان طبیعی با زبان نوشتاری سروکار داریم. این مسئله باعث می‌شود گرچه به جهت از دست دادن اطلاعات گویشی مانند لحن گوینده، آهنگ صدا، تاکید و مکث، با مشکلات و ابهاماتی مواجه شویم، ولی در مقابل با شکل

محدودتری از زبان کار می‌کنیم. بسیاری از بی‌ترتیبی‌های زبان، متعلق به زبان گفتاری است و در زبان نوشتاری بیشتر قالب‌های دستوری رعایت می‌شوند و لذا تهیه دستور زبان پوشاننده‌ی تمام متن، ساده‌تر است.

در تلاش برای ساخت یک سیستم پردازش و درک متون فارسی با مسائل و مشکلاتی مواجه می‌شویم که بعضی در بیشتر زبان‌ها بروز کرده و برخی خاص زبان فارسی می‌باشند. همچنین برخی از این پیچیدگی‌ها به طبیعت زبان و نارسایی‌های قواعد زبان شناسی مربوط و برخی دیگر برخاسته از مشکلات ایجاد سیستم‌های هوش مصنوعی است [۳]. در بخش بعد به برخی از این مسائل اشاره می‌شود [۴].

۱-۲-۱. پیچیدگی‌های پردازش متون فارسی

با توجه به بحث اخیر می‌توان در کل اهم مشکلات فعلی پردازش متون فارسی را در چند دسته زیر خلاصه نمود[۴]:

۱) عدم وجود منابع زبانی مناسب و کافی برای زبان فارسی مانند واژگان‌های تک زبانه و چند زبانه محاسباتی، واژگان‌های معنایی و متصل به هستان شناسی (هستان شناسی‌های لغوی)، هستان شناسی جامع عمومی و تخصصی، پیکره‌های عمومی و تخصصی ساده یا برچسب خورده (با برچسب‌های اجزای کلام، کسره اضافه، نقش‌های موضوعی، مفاهیم و روابط مفهومی و غیره)، مجموعه مدون قوانین ساختواژی و دستوری پوشا، عدم وجود استانداردهای شیوه نگارش، فاصله‌گذاری و رمزگزاری حروف و علائم.

۲) مشکل تشخیص مرز کلمات (مسئله شیوه‌های نگارش متفاوت)

۳) مشکل تشخیص مرز گروه‌های اسمی (مسئله کسره اضافه نامرئی)

۴) از دست دادن اطلاعات گویشی

۵) مسئله ابهام

۶) افعال مرکب و اصطلاحات

۷) مسئله هم نگاره‌ها و تحت آن مسئله حذف مصوت‌های کوتاه (اعراب) از نوشتار

۸) معناشناسی و مشکلات تحلیل معنایی.

۱-۲-۲. مروری بر فعالیتها و دستاوردهای گذشته

در دو دهه اخیر فعالیتهای در زمینه پردازش زبان فارسی در ایران و سایر کشورهای جهان انجام شده است. این فعالیت‌ها در زمینه صرفی، نحوی، معنایی و کاربرد پردازش طبیعی صورت گرفته است. در این بخش مروری بر فعالیتها به تفکیک حوزه اصلی تمرکز (صرف، نحو، معنا و کاربرد) خواهیم داشت[۴].

◘ پردازش لغوی

منظوراز پردازش لغوی شناسایی مرز لغات و جملات در یک متن است. این مرز ممکن است به شکل ساده توسط جداکننده‌هایی مانند فاصله، کاما، نقطه، علامت سوال و ... تعیین شود و یا نیاز به پردازش‌های پیچیده‌تر داشته باشد مانند زمانی که میان بخش‌های یک کلمه از فاصله استفاده می‌کنیم، مثل کلمه«می توان» و یا وقتی که دو کلمه مجزا را بدون فاصله و پی در پی می‌نویسیم، مثل عبارت «دربرابرباد». تعیین مرز کلمات در زبان فارسی بدلیل گوناگونی رسم‌الخط و عدم وجود استانداردهای نگارشی و همچنین به دلیل وجود شکل‌های مختلف حروف، اول – وسط – آخر و چسبان و غیرچسبان، بیش از زبان انگلیسی مشکل‌ساز است. این مشکل در زبان انگلیسی تنها برای کلمات مرکب ممکن است رخ دهد که آنها را می‌توان در مراحل بعدی پردازش مثل پردازش نحوی تشخیص داد. اما در زبان فارسی علاوه بر کلمات مرکب که مشکلی مشابه با انگلیسی ایجاد می‌کنند، مرز کلمات غیرمرکب نیز ممکن است بدرستی تشخیص داده نشود. از فعالیت‌های انجام شده در این زمینه می‌توان به [۶] اشاره نمود که به تشخیص انتهای کلمات و فاصله گذاری میان آنها می‌پردازد. همچنین [۷] در مطالعه‌ای به بررسی نحوه تشخیص کسره اضافه در متن با استفاده از روشهای آماری مبتنی بر پیکره‌های زبانی پرداخته است. تشخیص کسره اضافه محذوف کمک بسیاری به حل مشکل ابهام در شناسایی مرز گروههای اسمی می‌نماید. از سوی دیگر در بعضی کارها این مرحله با مراحل دیگر ادغام و یا در طی مراحل دیگر انجام می‌شود.مثلا در برخی تحلیلگر ساختواژی معرفی شده در بخش بعد در حین تحلیل ساختواژی، مرز کلمات نیز تعیین و فاصله‌های زائد درون کلمات حذف می‌شوند.

◘ پردازش ساختواژی

مرحله دیگر در پردازش متن تحلیل ساختواژی می‌باشد. این مرحله به تجزیه و ترکیب اجزاء کلمات می‌پردازد. به عبارت دیگر در تحلیل ساختواژی در هنگام درک یا تجزیه متن به تشخیص اجزاء کلمه -تک‌کلمه‌ها- و استخراج ریشه و وندهای متصل به آن و در هنگام تولید متن به ساخت کلمه از روی اجزاء سازنده‌اش توجه داریم. تحلیل ساختواژی بر دو نوع است: تصریفی و اشتقاقی. تحلیل ساختواژی تصریفی به تجزیه کلماتی می‌پردازد که با تصریف ساخته شده اند. تصریف افزودن وند به کلمه‌ای برای ساخت کلمه دیگر است به گونه‌ای که معمولاً منجر به تغییر مقوله (طبقه نحوی) و معنی کلمه نشود مانند صرف افعال برای شخص‌ها و زمان‌های مختلف، جمع بستن یا نکره کردن اسامی، افزودن ضمیر ملکی به اسم یا ضمیرمفعولی به فعل و امثالهم. نوع دوم تحلیل ساختواژی مربوط به ساختواژی اشتقاقی است. در اشتقاق، کلمه جدید حاصل از افزودن وند با کلمه قبل معمولاً از جهت مقوله نحوی و معنا متفاوت می‌شود. این بخش گستره وسیعی از ساختواژی فارسی را می‌پوشاند و تاکنون نه مجموعه قواعد ساختواژی کاملی برای آن تدوین شده و نه تحلیلگر جامعی برای پوشش این گستره ایجاد گشته است.

زبان فارسی از جهت ساختواژی به خصوص از دیدگاه تصریفی زبانی قانونمند و ساخت یافته است. در تولید تحلیلگر برای یک زبان دو مسئله وجود دارد (۱): تهیه مجموعه مدونی از قواعد تحلیل (۲): طراحی و ساخت تحلیلگری که با استفاده از این مجموعه قواعد قادر به تحلیل عناصر زبان باشد. در بسیاری موارد الگوریتم‌ها و روش‌های تحلیل، مستقل از زبان هستند و یا با تغییرات اندک قابل انطباق با زبان خاص می‌باشند. در این حالت مسئله اصلی، پیاده سازی این الگوریتم‌های شناخته شده، بهبود کارایی آنها و افزودن قواعد وابسته به زبان به آنهاست. این نکته برای تحلیل ساختواژی به تهیه قواعد ساختواژی تصریفی و اشتقاقی زبان و ایجاد الگوریتم‌هایی که بتوانند بر اساس قواعد فوق کلمات زبان را تجزیه و تحلیل کنند تبدیل می‌شود. تاکنون تحلیلگرهای ساختواژی تصریفی مختلفی برای زبان فارسی ساخته شده اند که عمدتاً در بخش تجزیه و برای استخراج اجزاء کلمات کار می‌کنند. از جمله می‌توان به تحلیلگر ساختواژی ساخته شده در پروژه مترجم شیراز [۸] و تحلیلگرهای معرفی شده در [۹،۱۰،۱۱،۱۲]، اشاره نمود. در تحلیلگر شیراز مسئله تصریف در زبان فارسی تا حد زیادی حل شده است. اما نرم افزار آن بصورت آزاد در دسترس محققین برای استفاده قرار ندارد. از سوی دیگر برخی تحلیلگرهای عمومی برای زبان انگلیسی ایجاد و آزموده شده اند که قابل انطباق برای زبان فارسی نیز

هستند. به عنوان نمونه [۹] با جمع آوری مجموعه قواعد ساختواژی تصریفی (و چند نمونه برای ساختواژی اشتقاقی) تحلیلگر Ample را برای تحلیل کلمات زبان فارسی منطبق ساخته است. کار [۹] تمام موارد لحاظ شده در پروژه شیراز بعلاوه چند نکته جدید از جمله در نظرگرفتن «ه صامت» و همچنین افعال متصل را دربردارد.

همانطور که گفته شد بیشتر فعالیت‌های انجام شده در زمینه ساخت تحلیلگرهای ساختواژی بر تصریف متکی بوده اند. علت این امر آن است که قوانین تصریف محدود و تعریف شده هستند ولی قوانین اشتقاق بسیار زیاد، غیر مدون و مملو از استثنائات و حالات خاص می‌باشند. به عبارت دیگر گلوگاه اصلی ایجاد یک سیستم تحلیل‌گر ساختواژی جامع اکتساب وتهیه مجموعه مدون قوانین ساختواژی زبان است و نه طراحی و پیاده سازی الگوریتم‌های تحلیل ساختواژی. بدلیل کثرت، تنوع و استثناپذیری این قوانین، اکتساب دستی آنها کاری زمان بر و پرهزینه است. به همین جهت دسته دیگری از فعالیت‌های تحقیقاتی بر اکتساب خودکار آنها متمرکز شده است. از جمله‌این تحقیقات می‌توان به [۱۳] اشاره نمود که در آن واژک‌های زبان فارسی به صورت بدون نظارت از روی پیکره‌هایی از لغات فارسی استخراج می‌شوند. در این کار سیستم با دیدن لغات مختلف نحوه شکستن لغات به اجزاء سازنده را بتدریج می‌آموزد و می‌تواند بدون داشتن قوانین صریح عمل تجزیه ساختواژی تصریفی و اشتقاقی را انجام دهد.

◘ منابع زبانی

یکی از گلوگاه‌های پردازش زبان فارسی دردسترس نبودن منابع زبانی کافی و معتبر برای فارسی است. منابع مورد نیاز شامل واژگان محاسباتی، دستورزبان محاسباتی، پیکره‌های خام و برچسب خورده، هستان شناسی‌های عمومی و تخصصی، قواعد صرفی و الگوهای معنایی می‌باشد. در رابطه با تهیه این منابع نیز فعالیت‌هایی صورت گرفته است. اکثر تلاش‌های صورت گرفته برای ساخت واژگان محاسباتی مانند [۹،۱۰،۱۴] به طراحی ساختار واژگان و تهیه مجموعه محدودی از اطلاعات واژی در یک ساختار تعریف شده انجامیده‌اند. هدف این فعالیت‌ها بیشتر تحقیق بر مبانی نظری تهیه واژگان‌های محاسباتی و طراحی ساختار مناسب برای آنها بوده است و در نهایت محصول خاصی که قابل استفاده برای عموم باشد به دست نیامده است [۱۵] در کار خود علاوه بر طراحی ساختار، به ورود دانش مورد نیاز در این ساختار نیز پرداخته است. این واژگان قرار است بر روی وب و بصورت آزاد

برای استفاده در فعالیت‌های دیگر ارائه شود.

پیکره‌های متنی منابع مهم بعدی هستند که به صورت خام و برچسب خورده مورد استفاده قرار می‌گیرند. در [۱۶] پیکره بی‌جن‌خان[1] معرفی شده است که یک پیکره برچسب خورده با متون دسته-بندی شده در ۴۳۰۰ دسته مختلف و حاوی ۲.۶ میلیون کلمه برچسب خورده می‌باشد. ۴۰ نوع برچسب اجزای کلامی فارسی در این پیکره مورد استفاده قرار گرفته است. همچنین پیکره همشهری[2] [۱۷] پیکره دیگری حاوی ۳۴۵ مگابایت از اخبار و مقالات روزنامه همشهری می‌باشد. در [۱۸] پیکره‌ی « محک» معرفی شده است. این مجموعه که از خبرگزاری ها جمع‌آوری شده است شامل اخبار و مقالاتی در اندازه‌ی نیم صفحه تا چندین صفحه می‌باشد. « محک» شامل ۳۰۰۷ سند، ۲۱۶ پرس و جو در مورد آنها و لیست اسناد مرتبط با این پرس و جوها می‌باشد. از منابع تهیه شده دیگر می‌توان به لیست کلمات غیرمفید فارسی [۱۹] اشاره نمود.

◘ فعالیت‌های کاربردی

در بخش‌های قبل به فعالیت‌های پایه‌ای اشاره شد که در کاربردهای بزرگ و واقعی پردازش زبان‌های طبیعی مورد استفاده قرار می‌گیرند. این بخش به بررسی فعالیت‌های کاربردی انجام شده مرتبط با پردازش متون فارسی می‌پردازد.

از فعالیت‌های مرتبط با پردازش متون فارسی می‌توان به خلاصه سازی متون فارسی [۲۰]، سیستم‌های پرسش و پاسخ به زبان فارسی[۲۱]، پیش بینی رایانه‌ای کلمه[۲۲]، ویرایش ادبی جملات فارسی[۲۳]، اعراب گذاری متون فارسی[۲۴]، استخراج اطلاعات از مستندات متنی[۲۵] و کارهای دیگری که در [۴] به تفصیل اشاره شده است.

اولین و مهمترین مسئله در حوزه پردازش واژه که یکی از مسائل مهم در دامنه پردازش زبان طبیعی است، تقسیم کردن متن به کلمات و جملات بنظر می‌رسد. این مسئله به این سادگی که بنظر می‌رسد نیست. بسیاری از متون موجود خصوصاً متونی که از اسناد وب بدست می‌آید ساختار درستی

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

از نظر علائم نشانه‌گذاری ندارند و حتی آنهایی نیز که ساختار درستی دارند، این علائم دارای ابهامات زیادی برای تشخیص حدود جمله است که مفصلاً در بخش‌های بعدی مورد بررسی قرار خواهد گرفت. همچنین تعیین حدود کلمه و توکنایز کردن[1] متن صرفاً از روی فاصله‌گذاری بصورت دقیق بدست نمی‌آید و در بسیاری از زبانها مثل چینی، ژاپنی و تایلندی متن می‌تواند بدون فاصله نوشته شود و این کار را سخت‌تر می‌کند.

بعلت اینکه مسئله تعیین حدود و کلمه[2] در متن یک کار پایه‌ای برای بسیار از کارها در حوزه پردازش زبان طبیعی است باید از دقت بسیار بالایی برخوردار باشد زیرا وجود درصد خطای کم نیز تاثیر زیادی روی درستی اعمال انجام شده در مرحله بعد می‌گذارد. در بخشهای بعد به بررسی متدهای مختلفی که در این زمینه‌ها وجود دارد و نقاط قوت و ضعف آنها و مشکلات کلی که در این زمینه وجود دارد، بیشتر روی زبان انگلیسی و فارسی، می‌پردازیم.

[1] Tokenization
[2] Word and Sentence Boundary

تعیین حدود کلمه

۱-۳. مقدمه

در این فصل یک الگوریتم برپایه قوانین یادگیر برای انجام قطعه‌بندی کلمه ارائه می‌شود. این الگوریتم هم بعنوان یک قطعه‌بندی کننده خودکفا[1] با دقت بالا و هم بعنوان یک پس‌پردازنده[2] که خروجی الگوریتمهای موجود خروجی الگوریتمهای قطعه‌بندی کلمه موجود را بهبود می‌بخشد، کار می‌کند. در سیستم نوشتار بسیاری از زبانها شامل چینی، ژاپنی و تایلندی کلمه بوسیله فاصله جدا نمی‌شود. معمولاً تعیین حدود کلمه یکی از مراحل اولیه لازم برای پیش‌پردازش است. برای انجام کارهایی مثل برچسب‌گذاری نحوی و تجزیه کردن متدهای متنوعی اخیراً برای انجام قطعه‌بندی کلمه تولید شده‌اند و نتایج بطور گسترده‌ای چاپ شده‌اند.

یک مشکل اصلی در ارزیابی الگوریتم قطعه‌بندی این است که هیچ دستورالعمل کاملاً مورد قبولی درباره آنچه یک کلمه را تشخیص می‌دهد وجود ندارد، بنابراین هیچ قراردادی روی چگونگی صحت قطعه‌بندی یک متن در یک زبان قطعه‌بندی نشده وجود ندارد. در مقالات بارها راجع به اینکه گویندگان بومی یک زبان همیشه در مورد قطعه‌بندی درست موافق نیستند و همان متن می‌تواند به چندین مجموعه مختلف از کلمات بوسیله گویندگان بومی تقسیم شود، و همه بطور مساوی درست است، بحث شده است. نتایج تجربی نشان داده است که نرخ توافق بین گوینده‌های محلی بطور معمول ۷۵٪ است [۱۳]. در نتیجه یک الگوریتم که با یک قطعه‌بندی کننده محلی برابری کند، ممکن است در مقایسه با دیگر قطعه‌بندی کننده‌های درست بد نتیجه بگیرد. برخی دیگر از نتایج را در ارزیابی قطعه‌بندی کلمات در ۳-۳-۱ مورد بحث قرار گرفته است.

یک راه حل ساده برای مسئله قطعه‌بندی چندگانه درست ممکن است دستورالعمل‌های خاصی را برای آنچه که یک کلمه در زبان قطعه‌بندی نشده، هست یا نیست بسازد. این دستورالعمل‌ها می‌توانند بصورت تئوری همه مجموعه‌ها را بطور یکنواخت با همان قرارداد قطعه‌بندی کنند و می‌توانیم بطور

[1] Stand Alone
[2] PostProcessing

مستقیم متدهای موجود را روی همان مجموعه مقایسه کنیم. از آنجاییکه این روش در جلوبردن پیشرفت کارهای پردازش زبان طبیعی مانند برچسب‌گذاری نحوی و تجزیه موفق است، آرگومانهای قابل‌قبولی در برابر پذیرش آن برای قطعه‌بندی کلمه وجود دارد. برای مثال از آنجاییکه قطعه‌بندی کلمه صرفاً یک کار پیش‌پردازش است برای کارهای گوناگون بعدی مثل تجزیه و استخراج اطلاعات می‌تواند مفید یا حتی حیاتی باشد. در این مفهوم، قطه‌بندی کلمه مانند تشخیص گفتار است، که این سیستم باید مستقل از گویندگان مختلف کلمات را درست تشخیص دهد و تطبیق پیداکند. در برخی موارد ممکن است همچنین چندین قطعه‌بندی درست برای همان متن، بسته به احتاجات پردازش‌های بعدی، لازم به پذیرش باشد. به هر حال بسیاری از الگوریتم‌ها لیست کلمات دامنه خاص گسترده و روتین‌های تشخیص نام پیچیده را علاوه بر ماژول‌های ریخت‌شناسی کد شده برای فراهم کردن یک خروجی قطعه‌بندی از پیش تعیین شده بکار می‌برند. اصلاح الگوریتم قطعه‌بندی موجود برای ایجاد یک قطعه‌بندی متفاوت، بویژه اگر چگونگی و محل تفاوتهای موجود سیستماتیک در قطعه‌بندی واضح نباشد، می‌تواند سخت باشد.

این بطور گسترده‌ای در مقالات قطعه‌بندی کلمه گزارش شده است که مرز پیشتاز قطعه‌بندی درست کلمه در تشخیص کلمه است، که در لغت‌نامه قطعه‌بندی کننده‌ها وجود ندارد. مثلاً یک مسئله هم به منبع لغت و هم به وابستگی‌های بین متن مورد بحث و فرهنگ لغت وابسته است. نشان داده شده است که دقت قطعه‌بندی، وقتی‌که فرهنگ لغت با استفاده از همان مجموعه‌ای که بعنوان تست استفاده می‌شود ساخته شده باشد، بطور معنی‌داری بیشتر است. استدلالی مبنی بر اینکه به جای ساخت یک فرهنگ لغت مجزا یا حتی یک سری فرهنگ لغات خاص دامنه، استفاده از یک متد یادگیر قوی، برای جبران نارسایی‌های فرهنگ لغت عملی‌تر به نظر می‌رسد، انجام شده است، بعلاوه ساخت چنین الگوریتمی اجازه می‌دهد که بسیاری از زبانها را بدون نیاز به منابع ریخت‌شناسی گسترده و فرهنگ لغات خاص دامنه در هر زبان واحد قطعه‌بندی کنیم.

به این دلیل، مسئله قطعه‌بندی کلمه از یک مسیر متفاوت نشان داده شده است. یک الگوریتم برپایه قانون که می‌تواند بدقت متنی را که یک تخمین اولیه تقریبی برای قطعه‌بندی دارد را قطعه‌بندی کند. این الگوریتم با خروجی‌های تعداد زیادی از الگوریتم‌های قطعه‌بندی موجود با شماهای قطعه‌بندی مختلف وفق پیدا می‌کند، که این مزایای قطعه‌بندی صحیح چندگانه متن را تصدیق می‌کند. بعلاوه الگوریتم برپایه قوانین می‌تواند برای تکمیل الگوریتم‌های قطعه‌بندی موجود

بمنظور جبران نارسایی فرهنگ لغات بکار رود. این الگوریتم همچنین یادگیر و مستقل از زبان است و می‌تواند برای زبانهای قطعه‌بندی نشده بکار رود.

۱-۴. قطعه‌بندی برپایه تبدیل

جزء کلیدی الگوریتم قطعه‌بندی یادگیر مورد بحث، آموزش خطاگرای[1] برپایه تبدیل است، متد پردازش زبان برپایه مجموعه در ۱۹۹۳ معرفی شده است [۱۴]. این تکنیک یک الگوریتم ساده برای آموزش یک سری قوانین که می‌تواند برای کارهای مختلف پردازش زبان بکار رود را ارائه می‌کند، و از چند جهت با متدهای برپایه مجموعه معمولی متفاوت است. آن بطور ضعیفی آماری است اما احتمالی نیست، روشهای برپایه تبدیل داده‌های آموزشی کمتری از بیشتر روشهای آماری نیاز دارد. آن برپایه قانون است اما برای بدست آوردن قوانین به جای مهندسی دانش پرهزینه به یادگیری ماشین استناد می‌کند. نشان داده شده‌است که قوانین بدست آمده برای بدست آوردن بینش در طبیعت سری قوانین و بهبود دستی و اشکال‌زدایی سری مفید هستند. در تمام مراحل آموزش الگوریتم یادگیری مجموعه آموزشی کامل را در نظر می‌گیرد به جای کاهش دادن اندازه داده آموزشی بعنوان پیشرفت یادگیری، مانند استنتاج در درخت تصمیم‌گیری[۱۵].

کارهایی صورت گرفته است که اثباتی بر امکانپذیری تکنیک‌های برپایه تبدیل به شکل یک تعدادی از پردازنده‌ها شامل برچسب‌زننده‌های نحوی (توزیع شده بطور گسترده) [۱۶]، الصاق عبارات با حرف اضافه[2][۱۷] و یک تجزیه‌کننده قلابی[3] [۱۸]. تمام اینها با کارآیی قابل قیاس یا بهتر از قبل ایجاد شده است. آموزش برپایه تبدیل همچنین با موفقیت به چانکینگ متن[4] [۱۹]، رفع ابهام ریخت‌شناسی[5] [۲۰] و تجزیه اصطلاح[6] [۲۱] اعمال می‌شود.

[1] Error Driven
[2] Prepositional Phrase Attachment
[3] Bracketing Parser
[4] Text Chunking
[5] Morphological Disambiguation
[6] Phrase Parsing

۱-۴-۱. آموزش

قطعه‌بندی کلمه به راحتی می‌تواند درقالب یک مسئله برپایه تبدیل قرار گیرد، که یک مدل اولیه، یک وضعیت هدفی که ما می‌خواهیم مدل اولیه را به آن تبدیل کنیم و یک سری تبدیلاتی برای عملی کردن این بهبود، نیاز دارد. الگوریتم برپایه تبدیل شامل اعمال و امتیازبندی همه قوانین ممکن برای آموزش داده‌ها و تعیین اینکه چگونه قوانین بیشتر مدل را بهبود می‌بخشند، است. این قوانین سپس به همه جملات قابل اجرا اعمال می‌شود و فرآیند تکرار می‌شود تا دیگر هیچ قانونی ارزش داده‌های آموزش را بهبود نبخشد. بنابراین یک سری قوانین برای بهبود چرخشی مدل اولیه ساخته می‌شوند. ارزیابی سری قوانین روی یک مجموعه تست از داده‌ها که مستقل از داده‌های آموزش هستند صورت می‌گیرد.

اگر ما خروجی یک الگوریتم قطعه‌بندی موجود را بعنوان قطعه‌بندی اولیه و قطعه‌بندی مطلوب را بعنوان وضعیت هدف تلقی کنیم، می‌توانیم روی وضعیت اولیه یک سری تبدیلات را انجام دهیم (حذف حدود نادرست و اضافه کردن حدود جدید) برای اینکه از وضعیت هدف یک تقریب صحیح‌تر بدست آورد. بنابراین نیاز داریم فقط یک گرامر قانون مناسب برای تبدیل این تقریب اولیه تعریف کنیم و داده آموزشی مناسبی را آماده کنیم.

برای آزمایشات انجام شده یک مجموعه‌ای که بطور دستی توسط گویندگان بومی چینی و تایلندی تقسیم‌بندی شده بود استفاده شده است. بطور تصادفی این مجموعه به مجموعه تست و آموزش تقسیم شد. تقریباً ۸۰٪ داده‌ها برای آموزش الگوریتم قطعه‌بندی بکار رفت و ۲۰٪ بعنوان یک مجموعه تست برای ارزیابی قوانینی که از داده‌های آموزش یادگرفته شده‌اند. علاوه بر چینی و تایلندی، همچنین آزمایشات قطعه‌بندی با استفاده از یک مجموعه بزرگ که همه فاصله‌ها از متن حذف شده بودند روی انگلیسی نیز انجام شد. بیشتر آزمایشات روی انگلیسی نیز با همان نسبت ۸۰-۲۰ انجام شد، اما نتایج آزمایشات انگلیسی با مقادیر متفاوت داده‌ها آموزش را جلوتر بحث می‌کنیم.

۱-۴-۲. گرامر قانون

سه دسته اصلی از تبدیلاتی که می‌توانند روی وضعیت فعلی یک قطعه‌بندی ناقص عمل کنند:

- درج[1]: قراردادن یک محدود جدید بین دو کاراکتر
- حذف[2]: حذف یک محدوده موجود بین دو کاراکتر
- لغزش[3]: انتقال موقعیت فعلی یک محدوده موجود بین دو کاراکتر به یک موقعیت ۱، ۲ یا ۳ کاراکتر به چپ یا راست

در این گرامر، تبدیلات حذف و درج می‌تواند روی هر دو کاراکتر همسایه (یک ۲-تایی) و یک کاراکتر چپ یا راست ۲-تایی کارکند. تبدیل لغزش می‌تواند روی یک سری از ۱، ۲ یا ۳ کاراکتر اطراف که محدوده می‌تواند حرکت داده شود، کارکند. شکل ۳-۱، ۲۲ تبدیل قطعه‌بندی تعریف‌شده را برمی‌شمرد.

Rule			Boundary Action	Triggering Context
AB	⟺	A B	Insert (delete) between A and B	any
xB	⟺	x B	Insert (delete) before any B	any
Ay	⟺	A y	Insert (delete) after any A	any
ABC	⟺	A B C	Insert (delete) between A and B AND Insert (delete) between B and C	any
JAB	⟺	JA B	Insert (delete) between A and B	J to left of A
¬JAB	⟺	¬JA B	Insert (delete) between A and B	no J to left of A
ABK	⟺	A BK	Insert (delete) between A and B	K to right of B
AB¬K	⟺	A B¬K	Insert (delete) between A and B	no K to right of B
xA y	⟺	x Ay	Move from after A to before A	any
xAB y	⟺	x ABy	Move from after bigram AB to before AB	any
xABC y	⟺	x ABCy	Move from after trigram ABC to before ABC	any

شکل۱-۱. تبدیلات ممکن. ABCJ و k کاراکترهای خاص هستند، x و y می‌تواند هر کاراکتری باشد. ¬j و ¬k می‌تواند هر کاراکتری بجز j و k باشد.

[1] Insert
[2] Delete
[3] Slide

۱-۵. نتایج

با الگوریتم بالا در محل مناسب، ما می‌توانیم داده‌های آموزش را برای تولید یک سری قوانین برای تکمیل‌کردن یک قطعه‌بندی اولیه، بمنظور بدست آوردن قطعه‌بندی مناسب استفاده کنیم. بعلاوه از آنجاییکه همه قوانین کاملاً برپایه کاراکتر هستند، یک‌سری می‌تواند برای هر مجموعه کاراکتر و بنابراین هر زبانی یادگرفته شود. این الگوریتم برپایه قانون برای بهبود نرخ قطعه‌بندی کلمات چندین الگوریتم قطعه‌بندی در سه زبان استفاده شده است.

۱-۵-۱. ارزیابی قطعه‌بندی

علیرغم تعداد مقالات موجود در این موضوع، ارزیابی و مقایسه الگوریتم قطعه‌بندی موجود واقعاً غیر ممکن است. بعلاوه برای مسئله قطعه‌بندی صحیح چندگانه یک متن، مقایسه بخاطر فقدان معیار پایه الگوریتم مشکل است. دو معیار عمومی کارآیی فراخوانی و دقت وجود دارد، که فراخوانی بعنوان درصد کلماتی که در متن قطعه‌بندی شده دستی بوسیله الگوریتم قطعه‌بندی مشخص می‌شوند، تعریف می‌شود، و پیش‌بینی بعنوان درصد کلمات بگردانده شده بوسیله الگوریتم که در متن قطعه‌بندی شده دستی در همان موقعیت واقع شده‌اند، تعریف می‌شود. اجزاء امتیازات فراخوانی و پیش‌بینی سپس برای محاسبه معیار F [۲۲]، که $F = (1+\beta)PR/(\beta P + R)$ است، بکار می‌روند. در اینکار تمام نتایج بعنوان یک معیار F متعادل (فراخوانی و پیش‌بینی با وزن برابر) با $\beta = 1$ گزارش شده است، یعنی $F = 2PR/(P+R)$.

۱-۵-۲. چینی

برای آزمایشات چینی، مجموعه آموزش شامل ۲۰۰۰ جمله (۶۰۱۸۷ کلمه) از یک مجموعه آژانس خبری زینهو[1] است، مجموعه تست ۵۶۰ جمله (۱۸۷۸۳ کلمه) از همان مجموعه است. چهار آزمایش با استفاده از این مجموعه انجام شده است، با چهار الگوریتم مختلف نقطه شروع را برای یادگیری تبدیلات قطعه‌بندی فراهم شده است. در هر مورد سری قوانین یادگرفته شده از مجموعه آموزش، یک بهبود معنی‌داری را روی مجموعه تست نتیجه داده است.

۱-۵-۲-۱. کاراکتر بعنوان کلمه[2]

یک قطعه‌بندی اولیه خیلی ساده برای چینی درنظر گرفتن هر کاراکتر بعنوان یک کلمه است. از آنجاییکه متوسط طول کلمات در چینی کاملاً کوتاه است، بیشتر کلمات شامل ۱ یا ۲ کاراکتر هستند، این قطعه‌بندی کاراکتر بعنوان کلمه بسیاری کلمات یک کاراکتری را مشخص می‌کند و یک قطعه‌بندی اولیه را فراهم می‌کند با امتیاز F=40.3. در حالیکه یک امتیاز قطعه‌بندی کم است، این الگوریتم قطعه‌بندی کلمات کافی را برای تولید یک تقریب قطعه‌بندی قابل‌قبول مشخص می‌کند. در واقع الگوریتم کاراکتر بعنوان کلمه به‌تنهایی نشان داده شده [۲۳،۲۴] که برای استفاده در بازیابی اطلاعات چینی مناسب است.

الگوریتم مذکور ۵۹۰۳ فرم تبدیل از ۲۰۰۰ جمله مجموعه آموزش یادگرفت. ۵۹۰۳ تبدیل به مجموعه تست اعمال شد و امتیاز را از F=40.3 به 78.1 بهبود بخشید، یک کاهش ۶۳.۳٪ در نرخ خطا. این یک نتیجه شگفت‌آور و دلگرم‌کننده است، از این حیث که از یک تخمین اولیه خیلی ساده بدون استفاده از هیچ فرهنگ لغتی جز اینکه بطور ضمنی از داده‌های آموزش بدست آمده باشد، این

[1] Xinhun news agency corpus
[2] Character-as-word (CAW)

الگوریتم برپایه قانون توانایی دارد یک سری از تبدیلات با یک دقت قطعه‌بندی بالا تولید کند.

۱-۵-۲-۲. الگوریتم (حریصانه) حداکثر تطبیق[1]

یک روش معمولی برای قطعه‌بندی کلمات بکاربردن یک نوع از الگوریتم حداکثر تطبیق است، که بارها بعنوان الگوریتم حریصانه مورد توجه قرار گرفته است. الگوریتم حریصانه از کاراکتر اول در یک متن شروع می‌شود و با استفاده از یک لیست از کلمات قطعه‌بندی شده زبان، سعی می‌کند طولانی‌ترین کلمه با این کاراکتر را از لیست پیدا کند. اگر کلمه‌ای پیدا شد، الگوریتم حداکثر تطابق یک محدوده را در پایان طولانی‌ترین کلمه نشانه‌گذاری می‌کند، سپس همان جستجوی تطبیق طولانی را برای حرف کاراکتر بعدی شروع می‌کند. اگر هیچ تطابقی پیدا نشد الگوریتم حریصانه به سادگی آن کاراکتر را در نظر نمی‌گیرد و جستجو را برای کاراکتر بعدی شروع می‌کند. در این روش یک قطعه‌بندی اولیه می‌تواند بدست آید که بیشتر از روش کاراکتر بعنوان کلمه قوی‌تر است. الگوریتم حداکثر تطابق برای مجموعه تست با استفاده از ۵۷۴۷۲ کلمه چینی از قطعه‌بندی کننده NMSUCHSEG (که در بخش بعد توضیح داده شده است) بکار برده شده است. این الگوریتم حریصانه یک امتیاز اولیه F=64.4 بدست آورده است.

یک سری از ۲۸۹۷ تبدیل از مجموعه آموزش یاد گرفته شده است، با اعمال به مجموعه تست امتیاز از F=64.4 به 84.9 بهبود بخشیده شد، یک کاهش خطای ۵۷.۸٪. از یک لیست کلمات چینی ساده، الگوریتم برپایه قانون یک امتیاز قطعه‌بندی قابل مقایسه را با الگوریتم‌هایی که با دامنه دانش خیلی بزرگ تولید شده‌اند ایجاد می‌کند.

این امتیاز در آینده با ترکیب الگوریتم‌های کاراکتر بعنوان کلمه و حداکثر تطبیق می‌تواند بهبود بخشیده شود. در الگوریتم حداکثر تطبیق توصیف شده در بالا، وقتی یک سری از کاراکترها در متن

[1] Maximum matching (greedy)algorithm

واقع می‌شوند، که در هیچ زیرمجموعه‌ای در لیست کلمات نمایش داده نشده است، کل سری بعنوان یک کلمه واحد در نظر گرفته خواهد شد. این اغلب کلماتی با ۱۰ یا بیشتر کاراکتر نتیجه خواهد داد، که در چینی بسیار غیرمحتمل است. در این آزمایشات وقتی با چنین سری از کاراکترها برخورد می‌کنیم، هر کاراکتر بعنوان یک کلمه مجزا، همانند الگوریتم کلمه بعنوان لغت فوق تلقی خواهد شد. این نوع از الگوریتم حریصانه برای همان لیست ۵۷۴۷۲ کلمات بکار گرفته شد و امتیاز اولیه F=82.9 بدست آمد. یک سری از ۲۴۵۰ تبدیل از مجموعه آموزش یادگرفته شده بود. با اعمال به مجموعه تست امتیاز از F=82.9 به87.7 ، با یک کاهش خطای ۲۸.۱٪ بهبود بخشیده شد. این امتیاز با استفاده از این نوع الگوریتم حداکثر تطابق ترکیبی با یک سری قوانین (۸۷.۷) بسیار به امتیاز قطعه‌بندی کننده NMSU (۸۷.۹) که در بخش بعد بحث می‌شود نزدیک است.

۱-۵-۲-۳. قطعه‌بندی کننده NMSU

سه آزمایش قبلی نشان داد که این الگوریتم سری قوانین می‌تواند نتایج قطعه‌بندی بسیارخوبی را با الگوریتم‌های قطعه‌بندی اولیه بسیار ساده بدهد. به هر حال، کمک در وفق‌پذیری یک الگوریتم موجود با سامانه‌های قطعه‌بندی مختلف، همانطوری که در قسمتهای قبلی توضیح داده شد، با احتمال زیاد با دقت‌های موجود می‌تواند انجام شود. در این آزمایشات مرتبه این الگوریتم که می‌تواند خروجی‌ها را بهبود ببخشد، نشان داده شده است.

قطعه‌بندی کننده چینی CHSEG در آزمایشگاه تحقیقات محاسباتی در دانشگاه ایالت نیومکزیکو[1]، که یک سیستم کامل قطعه‌بندی کننده چینی با دقت بالا است، تولید شده است[۲۵]. بعلاوه برای ماژول قطعه‌بندی کننده اولیه که کلمات را در متن بر اساس لیست کلمات چینی پیدا می‌کند، CHSEG بطور اضافی شامل ماژولهای خاصی برای تشخیص عبارات اصطلاحی چینی[2]، کلمات

[1] Computing Research Laboratory at New Mexico State University
[2] recognizing idiomatic expressions

اشتقاقی[1]، اسامی خاص خارجی[2] است. دقت CHSEG روی یک مجموعه ۸.۶ مگابایتی مستقلانه درحدود F=84.0 گزارش شده است [۲۶]. در این مجموعه تست شده امتیاز قطعه‌بندی F=87.9 بدست آمد. این الگوریتم برپایه قوانین سری ۱۷۵۵ تبدیلات را از مجموعه آموزش یادگرفت. با اعمال به مجموعه تست، امتیاز از ۸۷.۹ به ۸۹.۶ با کاهش نرخ خطای ۱۴٪ بهبود بخشیده شد. این الگوریتم برپایه قانون، بنابراین توانایی دارد سیستم‌های با کارآیی بالای موجود را بهبود بخشد.

جدول ۳-۱ خلاصه وضعیت چهار آزمایش بر روی زبان چینی را نشان می‌دهد.

جدول۱-۱. نتایج بدست آمده روی چینی

Initial algorithm	Initial score	Rules learned	Improved score	Error reduction
Character-as-word	40.3	5903	78.1	63.3%
Maximum matching	64.4	2897	84.9	57.8%
Maximum matching + CAW	82.9	2450	87.7	28.1%
NMSU segmenter	87.9	1755	89.6	14.0%

۱-۵-۳. تایلندی

درحالیکه تایلندی نیز یک زبان جدانشدنی است، سیستم نوشتار تایلندی الفبایی است و متوسط طول کلمه بزرگتر از چینی است. بنابراین انتظار می‌رود که این سیستم برپایه تبدیلات کاراکتر، از آنجاییکه متن بیش از یک کاراکتر در بسیاری از موارد برای تصمیم‌گیری بسیاری از قطعه‌بندی‌ها در زبان الفبایی لازم دارد، بخوبی روی تایلندی کار نکند.

مجموعه تایلندی شامل متنی از آژانس خبری نک‌تک[3] در تایلند است. برای این آزمایشات

[1] derived words
[2] foreign proper names

[3] Thai News Agency via NECTEC

مجموعه آموزش شامل ۳۳۶۷ جمله (۴۰۹۳۷ کلمه) است. مجموعه تست شامل ۱۲۴۵ جمله (۱۳۷۲۴ کلمه) از همان مجموعه است.

قطعه‌بندی اولیه با استفاده از الگوریتم حداکثر تطابق انجام شد، با یک فرهنگ لغت شامل ۹۹۳۳ کلمه تایلندی از فیلتر جداساز کلمه در cttex، که یک بسته نرم‌افزاری در زبان تایلندی است. این الگوریتم حریصانه در قطعه‌بندی اولیه امتیاز F=48.2 را روی مجموعه تست بدست آورد.

الگوریتم برپایه قانون یک سری ۷۳۱ تبدیل را یادگرفت که امتیاز را از ۴۸.۲ به ۶۳.۶ با کاهش خطای ۲۹.۷٪ بهبود بخشید. درحالیکه سیستم الفبایی بطور آشکار برای قطعه‌بندی سخت‌تر هستند، هنوز یک کاهش چشمگیر در نرخ خطای قطعه‌بندی کننده با استفاده از الگوریتم برپایه تبدیل مشاهده می‌شود. با اینحال این جای تردید دارد که یک قطعه‌بندی کننده با امتیاز ۶۳.۶ در بسیاری از کاربردها مفید واقع شود، و این نتیجه نیاز دارد که بطور معنی‌داری بهبود بخشیده شود.

۱-۵-۴. انگلیسی بهم چسبیده

اگرچه انگلیسی یک زبان قطعه‌بندی نشده نیست و سیستم نوشتار الفبایی و متوسط طول کلمه‌ای مثل تایلندی دارد، از آنجایی که متوسط منابع زبان انگلیسی (مثل لیستها و آنالیزگرهای ریخت شناسی) بیشتر در دسترس هستند، آزمایش با مجموعه انگلیسی بهم‌چسبیده، که متون انگلیسی هستند که فاصله بین آنها حذف شده و حدود کلمات مشخص نیست، آموزنده بنظر می‌رسد. در زیر یک مثال از جمله انگلیسی و نوع بهم‌چسبیده آن مشاهده می‌کنید:

About 20,000years ago the last ice age ended .
Abou20,000 yearsagothelasticeageended.

نتایج این آزمایش به تعیین منابع مورد نیاز، که باید بمنظور تولید یک الگوریتم قطعه‌بندی با دقت بالا در زبانهای الفبایی قطعه‌بندی نشده مثل تایلندی کمک می‌کند. بعلاوه می‌توانیم آنالیز خطای با جزئیات بیشتری را در قطعه‌بندی انگلیسی ارائه کنیم.

آزمایشات انگلیسی با استفاده از مجموعه متون مجله وال استریت انجام شد. مجموعه آموزش شامل ۲۶۷۵ جمله (۶۴۶۳۲ کلمه) که تمام فاصله‌های بین آنها حذف شده است می‌باشد، مجموعه تست یک مجموعه جدا ۷۰۰ جمله‌ای (۱۶۳۱۸ کلمه) از همان مجموعه با حذف فواصل است.

۱-۵-۴-۱. آزمایشات حداکثر تطابق

برای یک آزمایش اولیه قطعه‌بندی با استفاده از الگوریتم حداکثر تطابق، با یک فرهنگ لغت بزرگ ۳۴۲۷۲ کلمه انگلیسی بدست آمده از مجله وال استریت، انجام شد. در برابر امتیاز اولیه کم تایلندی الگوریتم حریصانه یک امتیاز قطعه‌بندی انگلیسی اولیه F=73.2 را گرفت. الگوریتم برپایه قوانین یک سری از ۸۰۰ تبدیل را یادگرفت که امتیاز را از ۷۳.۲ به ۷۹.۰ با کاهش خطای ۲۱.۶٪ بهبود بخشید.

تفاوت موجود در امتیاز تایلندی و انگلیسی این است که الگوریتم حریصانه به لیست کلمات وابسته است. برای نمونه نصف کلمات از لیست انگلیسی حذف شد و کارآیی الگوریتم حریصانه از ۷۳.۲ به ۳۲.۳ کاهش یافت. جدول ۳-۲ نتایج استفاده از الگوریتم حریصانه را روی سه زبان نشان می‌دهد.

جدول۱-۲. خلاصه نتایج حداکثر تطابق

Language	Lexicon size	Initial score	Rules learned	Improved score	Error reduction
Chinese	57472	64.4	2897	84.9	57.8%
Chinese (with CAW)	57472	82.9	2450	87.7	28.1%
Thai	9939	48.2	731	63.6	29.7%
English	34272	73.2	800	79.0	21.6%

۱-۵-۴-۲. آزمایشات قطعه‌بندی ریخت‌شناسی پایه

طوریکه در بالا گفته شد منابع انگلیسی در دسترس‌تر از تایلندی است. می‌توان از این منابع برای تقریب قطعه‌بندی اولیه که از الگوریتم حریصانه جدا می‌شوند را فراهم کرد. با دانش محلی انگلیسی یک لیست کوتاهی از پیشوند[1] و پسوندهای[2] بکار گرفته شد و یک الگوریتم ساده برای قطعه‌بندی اولیه انگلیسی که محدوده‌ها بعد از هر پسوند و قبل از هر پیشوند قرار داده شده بعلاوه کاراکترهای

[1] Prefix
[2] Suffix

نشانه‌گذاری قطعه‌بندی تولید شد. در بیشتر موارد، این روش ساده فقط یک یا دو محدوده لازم برای تشخیص کلمات کامل را توانست مشخص کند. و امتیاز اولیه به وضوح کم شد، F=29.8. با اینحال، حتی از این تقریب اولیه معیوب، الگوریتم برپایه قوانین یک سری از ۶۳۲ تبدیل را یادگرفت که فراخوانی کلمات را دوبرابر کرد و امتیاز را از ۲۹.۸ به ۵۳.۳ با کاهش خطای ۳۳.۵٪ بهبود بخشید.

۱-۵-۴-۳. مقدار داده‌های آموزش

از آنجاییکه داده‌های انگلیسی زیادی وجود دارد یک آزمایش کلاسیک برای تعیین تاثیر میزان داده‌های آموزشی روی توانایی سری قانون در بهتر کردن قطعه‌بندی انجام شد. با یک مجموعه آموزش اندکی بزرگتر از مجموعه تست، ۸۷۲ جمله، کار شروع شد و آزمایشات حداکثر تطابق گفته شده تکرار شد. سپس بطور پله‌ای میزان داده‌های آموزش افزایش داده شد و آزمایشات تکرار شد. نتایج خلاصه شده در جدول ۳-۳ نشان می‌دهد که جملات آموزشی بیشتر یک سری قوانین بزرگتر و یک کاهش خطای بیشتر را در داده‌های تست بهمراه خواهد داشت.

جدول۱-۳. اندازه مجموعه‌های آموزش انگلیسی

Training sentences	Rules learned	Improved score	Error reduction
872	436	78.2	18.9%
1731	653	78.9	21.3%
2675	800	79.0	21.6%
3572	902	79.4	23.1%
4522	1015	80.3	26.5%

۱-۶. نتیجه‌گیری

نتایج این آزمایشات مشخص می‌کند که سری قوانین برپایه تبدیل مکمل تقریب اولیه پایه است و می‌تواند قطعه‌بندی دقیقی را تولید کند. بعلاوه توانایی بهبود کارآیی رنج وسیعی از الگوریتمهای

قطعه‌بندی، بدون نیاز به مهندسی دانش گران وجود دارد. یادگیری سری قوانین می‌تواند در زمان کم و بدون نیاز به دانش زبان خاص بدست آید.

این الگوریتمها برای زبانهایی مثل فارسی نیز برای بهینه کردن نتایج الگوریتم‌های قطعه‌بندی کلمه قابل استفاده است اما بعنوان الگوریتم پایه نمی‌توان از آن استفاده کرد بلکه بعنوان یک الگوریتم پس‌پردازنده مکمل می‌تواند استفاده شود.

پردازش متن فارسی

قبل از آنالیز ریخت‌شناسی[۱] یا تجزیه نحوی[۲] یک متن نیاز دارد که به منظور تعیین حدود جمله و کلمه توکنایزشن شود. این بخش توکنایزر بکار رفته در پروژه ماشین ترجمه انگلیسی – فارسی شیراز را که در آزمایشگاه تحقیقاتی محاسبات انجام شده است را توصیف می‌کند[۲۷]. روش و سیستم نوشته‌های فارسی که می‌تواند در تشخیص حدود توکن بکار رود در متن مکتوب وجود دارد. سیستم یک توکنایزر مستقل از زبان سطح پایین را به کار می‌برد، خروجی یک سری نامبهم از توکن‌های پایه است. از آنجاییکه واحدهای معنی‌دار لغوی (واژک‌ها[۳]) قابل تفکیک خاص نیاز دارند قبل از اینکه آنالیز ریخت‌شناسی اتفاق بیافتد به کلمه متصل شوند، در آنالیز متن فارسی مشکلات ظاهر می‌شوند. بعلاوه، کلمات در فرم نوشتار اغلب بهم‌پیوسته هستند. این کارهای پیش‌پردازش بوسیله یک پس‌توکنایزر[۴] که شامل اطلاعات زبان خاص است انجام می‌گیرد.

۱-۷. مقدمه

توکنایزر سطح پایین[۵] بوسیله پس‌توکنایزر که شامل اطلاعات زبان خاص است دنبال می‌شود و روی خروجی توکنایزر سطح پایین عمل می‌کند. پس‌توکنایزر اساساً برای اتصال مجدد المانهای خمش[۶] (تغییری که در ساختار یک کلمه، بخصوص در انتهای آن بر اساس موقعیت گرامری آن ایجاد می‌شود) که بوسیله توکنایزر سطح پایین جدا می‌شوند بکار می‌رود. فارسی از الفبای عربی با چهار کاراکتر اضافی استفاده می‌کند، از راست به چپ نوشته می‌شود. اما کاراکترها برای تمام پردازشها به یونیکد[۷] برگردانده می‌شود. توکنایزر مستقل از زبان خواص کلی کاراکتر برای دسته‌بندی رشته‌ها به انواع توکن پایه بکار می‌رود:

---

[1] morphological analysis
[2] syntactic parsing
[3] Morpheme
[4] Post Tokenizer
[5] Low Level Tokenizer
[6] Inflectional
[7] Unicode

- کلمه: توالی حروف
- عدد: توالی ارقام
- جداکننده: یک جداکننده یونیکد که شامل کاراکترهای نشانه‌گذاری، خط تیره و فضای خالی است
- الفباعددی[1]: یک سری از سمبل‌ها، ارقام و حروف توام با هم.
- سمبل: کاراکترهایی چون ٪، $، # و...
- کنترل: کاراکترهایی چون سطر جدید، جدول‌بندی و کاراکترهای کنترلی مانند متصل-کننده‌ها

توکنایزر سطح پایین نمی‌تواند متن را به واحدهای متنی مثل سرصفحه‌ها، پاراگراف‌ها و جملات تقسیم کند و نمی‌تواند بین کاراکترهای نشانه‌گذاری مختلف تفاوت قائل شود از اینرو نقطه، که معمولاً نشان‌دهنده محدوده یک جمله است را، و کاما همانند هم عمل می‌کنند. تشخیص بین علائم نشانه‌گذاری مختلف باید در مرحله بعد در سیستم یکی شود. بعلاوه، متن مکتوب فارسی الفاظ مرکب و واژک‌های خمش خاص را بوسیله یک جداکننده بطول صفر یا یک کاراکتر کنترلی جدا می‌کند. از آنجایی که یک کاراکتر کنترلی بعنوان یک توکن مجزا بوسیله توکنایزر سطح پایین تلقی می‌شود، دو قسمت یک لفظ مرکب یا کلمه خمش فارسی اغلب جدا شده‌اند. قسمتهای جدا شده یک کلمه خمش نیاز دارد که مجدداً متصل شود قبل از اینکه آنالیز ریخت‌شناسی اعمال شود.

این گزارش سیستم نوشته‌های فارسی و ابزارهایی که می‌توانند برای تشخیص حدود توکن در متن مکتوب بکار روند را توصیف می‌کند. همچنین روشی را که پس‌توکنایزر برای اتصال واژک‌های جداشده مورد استفاده قرار می‌دهد شرح می‌دهد. یک مسئله رایج در آنالیز متن فارسی وجود دارد و آن این است که کلمات اغلب متصل هستند، این مسئله فعلاً با جزء پیش‌پردازنده حل شده است. فرآیند استفاده شده بوسیله پیش‌پردازنده در جدا کردن کلمات متصل نیز بحث شده است. در قسمت آخر علامت نشانه‌گذاری نقطه از آن جهت که چندین کاربرد در نشانه‌گذاری حدود دارد مورد بحث قرار گرفته است. سرنام[2] (کلمه‌ای که از بهم پیوستن اول کلمات دیگر ساخته می‌شود) و اختصارات

[1] Alphanumeric
[2] Acronym

فارسی بخوبی توصیف شده‌اند، اگرچه این موارد نیز باید درنظر گرفته شود ولی در این نسخه[1] پروژه شیراز در نظر گرفته نشده است. در سرتاسر این فصل اصطلاح توکنایزر برای ارجاع به ماژول شامل توکنایزر سطح پایین و پس‌توکنایزر بکار رفته است.

۱-۸. حروف فارسی

فارسی از الفبای عربی بعلاوه چهار حرف اضافی که در عربی استاندارد وجود ندارد استفاده می‌کند: پ، چ، ژ، گ. یک لیست کامل از الفبای فارسی در جدول ۴-۱ آورده شده است. در سیستم شیراز کاراکترها به یونیکد تبدیل می‌شوند که برای همه پردازشهای داخلی سیستم استفاده شده است. فارسی بین فرم ابتدایی، میانی و انتهایی حروف تفاوت قائل می‌شود که محل وقوع کاراکترها در کلمه را مشخص می‌کند. تشخیص فرم انتهایی برای آنالیز ریخت شناسی بسیار سخت است. در یونیکد، فرم انتهایی کاراکترها در فارسی ارائه شده‌اند بعنوان یک متصل کننده با طول صفر (یک کاراکتر کنترلی) که می‌تواند در طول پردازش توکن‌های ریخت‌شناسی استفاده شود. تعیین حدود کلمات با استفاده از فرم‌های کاراکتر در بخش ۴-۳-۳ مفصلاً بحث شده است. توجه کنید که جدول ۴-۱ کاراکترهای فارسی در فرم مجزا را فقط لیست کرده است و فرمهای میانی و ابتدایی کاراکتر نشان داده نشده‌اند.

بمنظور کار با سیستم نوشته‌های فارسی در یک وضع کارآ، یک رومانیزاسیون[2] خاص برای این پروژه در آزمایشگاه تحقیقات محاسبات ایجاد شده است. این بعنوان رومانیزاسیون شیراز نشان داده شده است. این رومانیزاسیون طراحی شده بود که معقول باشد، برای اینکه متن بتواند به فرمت فارسی عادی بدون از دست دادن هیچ اطلاعاتی برگردانده شود. این تلاش برای حفظ رومانیزاسیون، برای اینکه براحتی قابل خواندن باشد برای کسی که زبان می‌آموزد، صورت گرفته است. از اینرو چهار حرف

[1] Version
[2] romanization

فارسی ز، ذال، ظا و ضاد که همه ز تلفظ می‌شوند همه یک رومانیزاسیون براساس حرف انگیلیسی z دارند. تفاوت بین کاراکترها بوسیله تفکیک کننده‌های مختلف فراهم می‌شود. حرف فارسی الف مانند اکثر حروف صدادار می‌تواند تلفظ شود بسته به متنی که در آن ظاهر می‌شود و از اینرو مانند یک حرف انگلیسی صدادار رومانیز نشده است. بعلاوه از آنجاییکه حروف صدادار کوتاه اغلب در متن فارسی نوشته نمی‌شوند، رومانیزاسیون هیچ حرف صدادار کوتاهی را ارائه نمی‌کند، بنابراین پیچیدگی‌های موجود در متن اصلی باقی می‌ماند.

برای اهداف این سند به هر حال رومانیزاسیونی که برای نمایش حروف فارسی استفاده شده است در جدول ۴-۲ نشان داده شده است. این رومانیزاسیون هیچ حرف صدادار کوتاهی را ارائه نمی‌کند زیرا آنها در متن فارسی وجود ندارند. این جدول همچنین یک راهنمای تلفظ برای خواننده فراهم می‌کند.. توجه کنید به هرحال حروف صدادار کوتاه قابل نوشتن نیستند و از اینرو تلفظ بدست آمده کامل و تلفظ صحیح کلمه فارسی نیست.

جدول۱-۴. الفبای فارسی

Glyph	Name
ط	ta
ظ	za
ع	eyn
غ	gheyn
ف	fe
ق	ghaf
ک	kaf
گ	gaf
ل	lam
م	mim
ن	nun
و	vav
ه	he
ی	ye
ئ	ye hamze
ء	hamze
أ	alef hamze
ؤ	waw hamze
ۀ	he ye
ً	Tanvin

Glyph	Name
آ	alef madd
ا	alef
ب	be
پ	pe
ت	te
ث	se
ج	jim
چ	che
ح	He
خ	khe
د	dal
ذ	zal
ر	re
ز	ze
ژ	zhe
س	sin
ش	shin
ص	sat
ض	zat

توجه کنید آ و ا همه حروف صدادار هستند، حروف ه، و، ی می‌توانند حروف صدادار یا بی‌صدای متن داده شده باشند. بعنوان حرف بی‌صدا آنها ‘h’, ‘v’ و ‘y’ و بعنوان حروف صدادار ‘u’, ‘e’ ، ‘i’ تلفظ می‌شوند.

جدول۱-۵. رومانیزاسیون بکار رفته در پروژه شیراز

Persian Letters	Romanization	Pronunciation
alef with madd	A	father
alef	a	and *or* bed *or* so
be	b	boy
pe	p	pool
te	t	toy
se	s̄	sun
jim	J	Joe
che	ch	church
he	H	horse
khe	x	*similar to* German buch
dal	d	dog
zal	z̄	Zorro
re	r	*similar to* Spanish "r"
ze	z	Zorro
zhe	j	mirage
sin	s	sun
shin	sh	shoe
sat	S	sun
zat	Z	Zorro
ta	T	toy
za	Z̄	Zorro
eyn	e	and *or* bed *or* so *or* uh oh (glottal stop)
gheyn	Q	*similar to* French "r"
fe	f	fun
ghaf	q	*similar to* French "r"
kaf	k	kite
gaf	g	great
lam	l	love
mim	m	Mary
nun	n	nun
vav	v	very *or* food
he	h	horse
ye with hamze	i	you *or* uh oh (glottal stop)
ye	y	you *or* sea
short space (marking a final form character)	~	

۱-۹. حدود کلمه

عمل توکنایزر تشخیص حدود کلمه در یک متن مکتوب و ارائه کردن یک قطعه‌بندی یکنواخت قبل از پردازش متن است. در متن فارسی حدود کلمات می‌تواند بوسیله نشانه‌گذاری و فرم‌های کاراکتری که موقعیتش را در کلمه تعیین می‌کند، مشخص شود. کلمات همچنین می‌توانند متصل نوشته شوند. بطور مشابه برخی واژک‌ها ممکن است به فرم متصل یا غیر متصل ظاهر شوند. در این بخش تمام ترکیبات واژک و کلمه ممکن در متن فارسی مکتوب، بعلاوه علائم مشخص کننده حدود را مورد بحث قرار می‌دهیم. قطعه‌بندی نهایی که بوسیله توکنایزر فراهم شده است با همه کلمات، شامل زیر قسمتهای یک فعل ساده یا مرکب بعنوان توکن‌های مجزا، سروکار خواهد داشت. همه واژک‌های قابل تفکیک از روی اینکه بوسیله کاراکتر فضای خالی کوچک (اتصال با طول صفر[1]) بصورت تشکیل یک واحد توکن مجزا با کلمه آنالیز خواهد شد. اگر رشته مبهم باشد توکنایزر چندین قطعه‌بندی فراهم می‌کند.

۱-۹-۱. نقطه‌گذاری

علائم نشانه‌گذار خاص حدود جمله را نشان می‌دهد. در فارسی، نقطه، علامت سئوال و تعجب مشخص کننده حدود غیر مبهم هستند. نقطه علاوه بر حدود جمله، در فرم اختصار یا سرنام نیز به کار می‌رود. صرفنظر از "/" که در اعداد بکار می‌رود و "-" که می‌تواند بکار رود برای جداکردن کلمات مرکب، علائم نشانه‌گذاری دیگر حدود کلمه را بطور واضح مشخص می‌کند، این شامل کوتیشن، کاما، براکت و کالون است. توکنایزر سطح پایین همه علامت‌های نشانه‌گذاری و خط تیره‌ها را بعنوان جداکننده توکن برچسب می‌زند. تجزیه کننده بکار رفته در این پروژه هیچ توکنی را که برچسب‌گذاری نشده بعنوان توکن‌های کلمه به حساب نمی‌آورد. همه توکن‌ها که توکن‌های کلمه

[1] zero-width-joiner

نیستند یک مرز (مثل جمله) سخت در نظر گرفته می‌شود. از آنجاییکه، برای تشخیص بین جداکننده‌هایی که نشانه‌های حدود جمله هستند و آنهایی که نشانه‌های حدود کلمه هستند ماژول دیگری نیاز است، وگرنه هیچ تجزیه نحوی به حرف اعمال نخواهد شد و آنالیز جمله‌ای بدست نخواهد آمد. در این سیستم جداکننده‌های در این سیستم جداکننده‌های نشانه‌گذاری حدود جمله، گرامر نحوی را شامل خواهند شد. برای نمونه کاما در قوانین نحوی برای تعیین یک حدود عبارت است.

۱-۹-۲. فاصله

در متن فارسی، محدوده‌ها برای کلمات مجزا معمولاً بوسیله فاصله‌شان نشان داده می‌شود. هیچ فاصله بین دو کلمه به‌شکل ترکیب شده یا ساختار فعل ساده ظاهر نمی‌شود. این الگوی فاصله گذاری به هر حال خیلی مستحکم نیست و برخی اوقات کلمات مختلف ممکن است بدون فاصله جداکننده آنها ظاهر شوند. این کلمات متصل در بخش بعدی مورد بحث قرار خواهند گرفت. در توکنایزر سطح پایین یک فاصله بعنوان یک توکن جداکننده برچسب‌گذاری می‌شود و می‌تواند کلمات توکن مجزا را بطور موفق جداکنند.

۱-۹-۳. حدود کلمه و واژک

۱-۹-۳-۱. اشکال کاراکتر

سیستم نوشتار فارسی بین شکل ابتدایی، میانی و انتهایی یک کاراکتر بسته به موقعیت آن در کلمه تفاوت قائل می‌شود. این درشکل ۱ نشان داده شده است. فرم ابتدایی به این معنی نیست که حرف در ابتدای کلمه است، این فقط مشخص می‌کند که کاراکتر در پایان کلمه نیست، کاراکترها به فرم میانی هستند اگر کاراکتر متصل قبل و بعد داشته باشد. یک کاراکتر به‌شکل پایانی پایان کلمه را

مشخص می‌کند و می‌تواند توسط توکنایزر برای تعیین حدود کلمه بکار رود. از آنجاییکه دو کلمه متصل می‌توانند در توکن‌های مجزا قرار گیرند اگر کلمه اول با یک کاراکتر به فرم انتهایی پایان یابد. این شکل پایانی بوسیله متصل‌کننده با طول صفر طبق کاراکتر در یونیکد مشخص می‌شود. برای اهداف نویسه‌گردانی " ~ " تیلد را برای نشان دادن این کاراکتر کنترلی استفاده می‌کنیم.

final	*medial*	*initial*	
ب	ـبـ	بـ	"b"
گ	ـگـ	گـ	"g"
ج	ـجـ	جـ	"J"

شکل۱-۲. نمونه‌ای از اشکال کاراکترهای فارسی

حروف خاص (الف، دال، ذال، ر، ز، ژ، و) که فقط یک فرم صرفنظر از موقعیتشان در کلمه دارند وجود دارد. اگر چنین کاراکتری پایان کلمه اول جفت متصل باشد، توکنایزر نمی‌تواند کاراکتر را برای تعیین حدود کلمه استفاده کند (قسمت ۳.۳.۳ - الگوریتم برای کلمات ناشناخته – برای الگوریتم بکار رفته در این موقعیت را ببینید).

ساختار فعل ساده و مرکب اغلب بدون یک فاصله جداکننده بین دو قسمت ظاهر می‌شود. اگر کلمه اول ترکیب با یک کاراکتر به فرم پایانی پایان پذیرد دو قسمت به توکن‌های مجزا توسط توکنایزر قسمت خواهند شد، در (۱) برای یک رشته ترکیبی و در (۲) برای یک رشته فعل ساده نشان داده شده است. هم فعل ساده و هم مرکب بعنوان یک واحد حرفی منفرد در یک مرحله بعد از پردازش تشخیص داده می‌شوند.

(1) “riys~Jmhvr”-->“riys”“Jmhvr”
Lit.: head republic
‘President‘

(2) “zng~zdnd”-->“zng”“zdnd”
Lit.: bell hit (past/3pl)
‘(they) phoned‘

واژک‌های خاص همیشه متصل به کلمه ظاهر می‌شوند در صورتیکه می‌توانند یا متصل یا منفصل با یک اتصال بطول صفر نوشته شوند. بندرت واژک‌های غیر متصل به فرم جدا از کلمه با یک فاصله در میانشان ظاهر می‌شود. واژک‌های متصل بعنوان یک توکن با کلمه‌ای که با آن ظاهر می‌شوند آنالیز می‌شوند، اما واژک‌های غیرمتصل بعنوان یک توکن مجزا بوسیله توکنایزر سطح پایین در نظر گرفته می‌شود. پس‌توکنایزر سپس برای اتصال واژک مجزا پشت کلمه، بمنظور شکل‌دهی یک توکن منفرد بعنوان ورودی به آنالیزگر ریخت‌شناسی، استفاده می‌شود. وقتی یک واژک مجدداً به قسمت اصلی کلمه متصل می‌شود یک کاراکتر فاصله کوتاه " ~ " یا اتصال با طول صفر باید واژک و کلمه را جدا کند. دلیل اضافه کردن این کاراکتر در مثالهای زیر نشان داده شده است. رشته (۳) را در نظر بگیرید که شامل یک اسم به معنی نامه و بدنبال آن یک علامت نامعین است. توکنایزر سطح پائین ریشه اسم و واژک تعریف نشده را به دو توکن مجزای نشان داده شده جدا خواهد کرد. وقتی پس‌توکنایزر واژک را به ریشه آن مجدداً متصل می‌کند، باید یک متصل‌کننده بطول صفر را بمنظور بدست آوردن رشته اصلی اضافه کند. اگر متصل کننده با طول صفر بین کلمه و واژک اضافه نشود رشته نتیجه بصورتی که در (۴) نشان داده شده است می‌آید، که ترکیب کلمه و واژک طوریکه از ترجمه بدست آمده مشاهده می‌شود کاملاً متفاوت است. از آنجاییکه الگوریتم پس‌توکنایزر باید یک اتصال بطول صفر را قبل از اتصال مجدد واژک، بمنظور اینکه تمایز بین صرف فعل متصل و غیر متصل در فارسی حفظ شود، اضافه کند.

(3) ‘‘namh~ay’’-->‘‘namh’’‘‘ay’’
letter-Indef
‘a letter’

(4) ‘‘namhay’’
name-Plur-Ezafe
‘(the) names of’

۱-۹-۳-۲. الگوریتم برای تشخیص حدود کلمه و واژک

الگوریتم استفاده شده بوسیله پس‌توکنایزر نیاز به در نظرگرفتن تمام ترکیبات کلمه و واژک ممکن و فراهم کردن همه الگوهای قطعه‌بندی دارد. در نتایج نهایی کلمات به توکن‌های مجزا جدا می‌شوند. واژک‌هایی که به فرم متصل در متن ظاهر می‌شوند متصل باقی خواهند ماند و واژک‌هایی منفصل با یک فاصله کوتاه " ~ " یا متصل‌کننده با طول صفر جدا می‌شوند. بعنوان مثال رشته "shrab~xvb" (= شراب خوب) را که با یک فاصله کوتاه جدا شده‌اند در نظر بگیرید که نشان‌دهنده یک حرف به شکل پایانی است. برای همه رخدادهای حروف به شکل پایانی که بوسیله یک کاراکتر به فرم ابتدایی دنبال می‌شود، توکنایزر سطح پایین رشته را به دو توکن مجزا جدا می‌کند. رشته در این مثال به دو توکن مجزای "shrab" و "xvb" جدا خواهد شد. اگر این دو توکن‌ها دو کلمه مجزا را نشان دهند، قطعه‌بندی تمام شده است. اگر یکی از رشته‌ها واژک باشد نیاز دارد که مجدداً متصل شود قبل از اینکه رشته به آنالیزگر ریخت‌شناسی فرستاده شود، اینکار در پس‌توکنایزر انجام می‌شود. در زیر این الگوریتم ارائه شده است که به خروجی توکنایزر سطح پایین اعمال شده است.

الگوریتم پس‌توکنایزر:

ما به دو رشته متوالی توجه می‌کنیم، در هر مورد هر توکن نسب به لیست واژک‌ها بررسی می‌شود:

- اگر هر دو رشته واژک نیستند آنها جدا باقی می‌مانند، نیاز به انجام هیچ کاری نیست.

(5) "shrab" "xvb" --> "shrab""xvb"
wine good

- اگر یک از رشته‌ها واژکی است که با یک کلمه ابهام ندارد، مجدداً واژک با اضافه کردن یک متصل‌کننده بطول صفر بین واژک و کلمه، متصل می‌شود.

(6) "kvtah" "tryn"-->"kvtah~tryn"
short est

- اگر یکی از رشته‌ها با یک کلمه ابهام دارد، در قسمت‌بندی نتیجه در مورد اول رشته‌ها جدا باقی می‌مانند و در مورد دیگر مجدداً با حائل متصل‌کننده بطول صفر متصل می‌شوند.

(7) “my~” “rqSm~”-->1. “my” “rqSm”
IMPdancing(1sg)2.“my ~rqSm”
‘(I) am dancing.’
[where “my” can also mean “wine”]

۱-۹-۳-۳. الگوریتم برای کلمات ناشناخته

کلمات متصل یک جایگاه عمومی در متن فارسی دارد. مثال (۸) از مجموعه روی خط[1] استخراج شده است، آنها موارد مختلف کلمات متصل را نشان می‌دهد. در همه این موارد کلمه اول با یک کاراکتر به شکل پایانی پایان نمی‌یابد. از آنجاییکه توکنایزر سطح پایین نمی‌تواند کلمات را به توکن‌های مجزا جدا کند، طوریکه در این مثال نشان داده شده است، نزدیک ترکیبات واقعی و افعال ساده، حروف اضافه کوتاه، حرف ربط و لفظ الحاقی مورد نظری که تکیه نداشته باشد، قبل و بعد از کلمه بدون فاصله حائل ظاهر می‌شوند. مثال آخر یک مورد متصل که اجزاء فعل ساده جدا شده‌اند را نشان می‌دهد. در این مثال لفظ الحاقی مورد نظری که تکیه نداشته باشد ra کلمه قبلی به المان پیش‌کلامی rd فعل ساده rd kardand (رد کردند) کلمه بعدی متصل شده است، بنابراین دو قسمت فعل ساده جدا می‌شوند.

(8) a. ***compounds***amvrxarJh--> amvr xarJh
affairs foreign
‘foreign affairs’

b.***light verbs*** pyshnhadkrdnd--> pyshnhad krdnd
proposedi(3 pl)
‘(they) proposed.’

c. ***preposition***azShyvnystha--> az Shyvnystha
fromZionists
‘from the Zionists’

d.***conjunction*** tablv~~v~~ >tablvv
painting and

[1] On Line

‘painting and’

e. ***postposition***kshvrra-->kshvrra
countryObj
‘the country’ (object of sentence)

f. ***distinct words***frAyndbykarsazy-->frAyndbykarsazy
processunemployment
‘(the) process of unemployment’
nystndayran-->nystndayran
are notIran
‘..(they) are not Iran..’

g. ***separated words***rard krdnd-->rard krdnd
Obj refusal did(3pl)
‘.. (they) refused..’

این مثالها نشان می‌دهد که برای توکنایزر فارسی لازم است که شامل یک الگوریتم برای تشخیص و جداکردن کلمات متصل باشد. قسمت قبلی مواردی از کلمات متصل که پایان کلمه اول یک کاراکتر پایانی است که می‌تواند برای تشخیص حدود کلمه استفاده شود مورد بحث قرار داد. اگر کلمه متصل با یک کاراکتر به فرم پایانی نیست پایان یابد طوریکه در مثال (۸) است سپس توکنایزر سطح پایین نمی‌تواند دو کلمه را بعنوان دو توکن مجزا تشخیص دهد. پیشنهاد شده که یک الگوریتم، که یک فاصله بعد از کاراکترهای بدون فرم پایانی اضافه می‌کند، اعمال شود. این کاراکترها الف، دال، ذال، ر، ز، ژ، واو هستند. از آنجاییکه یکی یا دو کلمه ممکن است حذف شود، توکن‌های جدا شده همچنین نیاز به متحمل شدن آنالیز ریخت‌شناسی دارند. در سیستم فعلی شیراز الگوریتم پیاده‌سازی شده (با اختلاف کم) اجزاء پیش‌پردازنده سیستم مسئله اتصال را قبل از اینکه متن آنالیز شود حل کرده‌اند.

الگوریتم کلمه ناشناخته:

- برای همه رخدادهای کاراکترهایی که یک فرم پایانی ندارند (الف، دال، ذال، ر، ز، ژ، واو) توکنایزر دو قسمت تولید می‌کند: یکی که یک فضای خالی بعد از کاراکتر اضافه می‌شود و یکی بدون فضای خالی . یک فاصله نیاز نیست بعد از کاراکتر پایانی در رشته اضافه شود. می‌توان برخی موارد ناخواسته را با دور انداختن هر ترکیبی که شامل یک حرف منفرد است (به جز برای "و" که حرف ربط است) حذف کنیم. این همچنین برای حذف ترکیبات خاص، اگر رشته آخر یک واژک که با کلمه ابهام ندارد، امکان دارد. برای نمونه

اگر رشته "مسافرین" در قطعه‌بندی "مسافر" و "ین" شود، از آنجایی که "ین" یک واژک صیغه جمع است و با کلمه‌ای ابهام ندارد، این قطعه‌بندی خاص حذف خواهد شد. برای تشریح مثال (۹) را در نظر بگیرید که همه قطعات حروف واحد حذف شده‌اند در این موارد یک فضای خالی بدنبال کاراکترهای دال، واو، ر اضافه شده است.

(9) “dvrdnya”-->1."dv” “rd” “nya”around world
2."dvr” “dnya” [correct segmentation]
3."dvrd” “nya”
4."dv” “rdnya”

- از آنجاییکه کلمات متصل شاید صرف شده باشند، قسمتهای قطعه‌بندی شده لازم است آنالیز ریخت‌شناسی را متحمل شوند قبل از اینکه در فرهنگ لغت جستجو صورت گیرد. در مثال (۱۰) فعل ساده "چاقو زد" با اجزاءاش متصل است، جزء پیش‌فعلی "چاقو" با "و" پایان می‌پذیرد، یک کاراکتری که شکل پایانی ندارد، و با قسمت فعلی "زد" ترکیب شده است. بمنظور تشخیص فعل در فرهنگ لغت روی "زد" باید آنالیز ریخت شناسی بمنظور بدست آوردن فرم مصدری فعل انجام شود "زدن".

(10) “chaqvzd”- > 1.“cha” “qvzd”
2.“chaqv” “zd” [correct segmentation]
3.“cha” “qv” “zd”

۱-۱۰. رفع ابهام حدود جمله

نقطه یک علامت نشانه‌گذاری مبهم در متن فارسی است، چون می‌تواند علاوه بر پایان جمله، یک قسمت از اختصار یا سرنام نیز باشد. این گزارش است برای رفع ابهام حدود جمله بوسیله تعیین اینکه آیا نقطه‌ای که در متن با آن برخورد کردیم یک نشانه حدود جمله مطمئن است یا قسمتی از یک اختصار یا سرنام است. این توضیح در ورژن فعلی توکنایزر شیراز وجود ندارد، زیرا اختصارات و سرنامها به شدت در مجموعه کمیاب است (۳ سرنام و هیچ اختصاری در میان ۳۰۰۰ جمله مجموعه)، اما این باید در فرآیند توکنایزشن کاملتر در نظر گرفته شود. اگرچه یک نقطه معمولاً بعد از یک فاصله

خالی در متن فارسی می‌آید، این الگو که بلافاصله بعد از نقطه کاراکتر دیگری بیاید زیاد مستحکم و مکرر نیست. سرنامها و اختصارات به هر حال یک ساختار قابل تشخیص راحت دارند، از اینرو بمنظور مجزا کردن جملات در متن فارسی توکنایزر باید تعیین کند که آیا نقطه جزء سرنام یا اختصار است قبل از اینکه بعنوان یک حدود جمله مورد عمل قرار گیرد. توجه کنید که اگر اختصار یا سرنام در پایان جمله ظاهر شود نقطه می‌تواند حدود جمله باشد.

این گزارش ساختار خاصی که باید بصورت بالقوه سرنام یا اختصار تشخیص داده شود تعریف می‌کند. وقتی که با یک نقطه در رشته برخورد می‌کنیم، این یکی می‌تواند در برابر این ساختار تطبیق یابد. سیستم ممکن است همچنین شامل زیر فرهنگ لغاتی بمنظور تعیین اینکه آیای رشته خاص متعلق به کلاس توکن‌های سرنام یا اختصار است یا نه باشد، اگر این تطبیق شکست بخورد توکنایزر می‌تواند به قطعه‌بندی جملات بپردازد. در موارد خاص رشته شامل نقطه ممکن است بین دو ساختار مبهم باشد.

قسمت ۴-۴-۳ همه ترکیبات ممکن اختصار، سرنام و نقطه را باهم آورده که حدود جمله را نشان می‌دهد و یک نمای کلی از قوانین توکنایزشن مورد نیاز برای حل پیچیدگی‌های موجود را نشان می‌دهد.

۱-۱۰-۱. سرنام

۱-۱۰-۱-۱. آنالیز وصفی

سرنام می‌تواند بوسیله روساختش بر اساس ترکیبی از کاراکترها و نقطه‌گذاری شناخته شود. فرمت کلی برای ساخت یک سرنام شامل یک یا بیشتر کاراکتر (در یک فرم پایانی اگر وجود داشته باشد) که بدنبال آنها نقطه قرار می‌گیرد باشد. در این فرمت، هر کاراکتر رومن اختصار حرف به حرف فارسی مطابق با الگوهای جدول ۴-۳ نوشته می‌شود. این در مثال زیر نشان داده شده است. در مثال (۱۱) کاراکتر آخر قبل از نقطه در یک فرم پایانی طوریکه بوسیله کاراکتر فاصله کوتاه " ~ " نشان داده

شده است پایان می‌یابد. در سرنام (۱۲) از آنجاییکه کاراکترهای فارسی "ر" و "ا" فرم پایانی ندارند کاراکتر فضای خالی کوتاه در دسترس نیست.

(11) *af~*.by~.Ay~FBI
*by~.by~.sy~*BBC

(12) *ar.py~.Jy~*RPG
*ka.g~.b~*KGB

به هر حال انواعی از این فرمت وجود دارد. روزنامه و مجلات خاص از سرنام بدون نقطه طوریکه مثال (۱۳) نشان داده شده است استفاده می‌کنند.

(13) *by~by~sy~*BBC

سرنامهایی که می‌توانند در این فرمت نشان داده شوند معمولاً شامل حروف رومن است که حرف به حرف به فارسی با یک کاراکتر پایانی که به فرم پایانی نوشته شده است برگردانده شده‌اند. به این معنی که مثال (۱۱) می‌توانست بدون نقطه نوشته شود به فرم حرف به حرف از آنجاییکه آخرین کاراکتر قبل از هر نقطه به فرم پایانی هستند. مثال (۱۲) شامل کاراکترهایی به فرم غیرپایانی قبل از نقطه است (مثل "ر" و "ا") و بنابراین نمی‌تواند بدون آن نوشته شود.

کلمات خاص در انگلیسی، سرنام در نظر گرفته می‌شوند، در حالی که در فارسی بعنوان اسم خاص تلقی می‌شوند و حرف به حرف نمایش داده نمی‌شوند. آنها در عوض طوریکه تلفظ می‌شوند نوشته می‌شوند طوریکه در مثال زیر نشان داده شده است:

(14) *sya*CIA
*aydz*AIDS

توجه کنید که همه سرنام‌ها در زبان به فرم حرف به حرف نوشته شده سرنام خارجی هستند، هیچ سرنام فارسی پیدا نشد که به فرمت تعریف شده در مثال (۱۱) تا (۱۳) نوشته شده باشد. در عوض سرنام فارسی از الگوی تشریح شده در مثال (۱۴) پیروی می‌کند و باید بعنوان اسم خاص در نظر گرفته شود. موارد نشان داده شده در زیر نمونه‌هایی از این سرنامها هستند، "ساواک" برای "سازمان امنیت و اطلاعات کشور" و "نداجا" برای "نیروی دریایی ارتش جمهوری اسلامی ایران".

(15) *savak~* (Savak)
sazman amnyt v a'Tla'at kshvr
*ndaJa(*Nedaja)
nyrvy dryayy artsh Jmhvry aslamy yran

جدول۱-۶. کاراکترهای رومن و حرف به حرف نویسی فارسی

Roman character	Persian transliteration
A	*a* (or *A* word-initially)
B	*by~* or *b~*
C	*sy~*
D	*dy~*
E	*ay~*
F	*af~*
G	*g~*
H	*ach~*
I	*ay~*
J	*Jy~*
K	*ky~* or *ka*
L	*al~*
M	*am~*
N	*an~*
O	*a*
P	*py~*
Q	
R	*ar*
S	*as~*
T	*ty~*
U	*yv*
V	*vy~*
W	*dblyv*
X	*ayks~*
Y	*vay~*
Z	*z*

۱-۱۰-۱-۲. ساختارهای مبهم

از آنجاییکه سرنامها یک ساختار قابل تشخیص ساده دارد، آنها معمولاً با دیگر ساختارها ابهام ندارد، فقط ساختار سرنامی که ممکن است با کلمات ابهام داشته باشد که در (۱۳) توضیح داده شده است که بدون هیچ نقطه‌ای ظاهر می‌شوند. ساختار این مثال یک فرم ظاهری سه قسمتی دارد و براحتی در اجزای جستجوی ترکیب تشخیص داده می‌شود.

وقتی یک سرنام بالقوه تشخیص داده شد سپس می‌تواند در فرهنگ لغت چک شود، اگر توکن در

فرهنگ لغت وجود ندارد، می‌تواند بصورت الگوی حرف به حرف داده شده در جدول ۴-۳ ترجمه شود. نقطه پایانی در سرنام نیز با یک نقطه کامل، یک نشاندهنده حدود جمله، مبهم است.

۱-۱۰-۱-۳. خلاصه

فرمت:

فرمت سرنام می‌تواند بوسیله قوانین زیر نشان داده شود:

برای موارد سرنام، کاراکترهای [Aa-y]، یک رشته دلخواه از کاراکترها نشان داده نمی‌شود، بلکه حرف به حرف نوشتن حروف خارجی به فارسی به فرمت جدول ۴-۳ است.

۱- ([Aa-y]+\~?\.)+[Aa-y]+\~?\.?

یک یا چند ترکیب از یک یا چند حرف که بعد از آن نقطه قرار می‌گیرد. نقطه پایانی اختیاری است. این قانون در مثالهای (۱۱) و (۱۲) نشان داده شده است.

۲- ([Aa-y]+\~)+[Aa-y]+\~?

یک یا چند ترکیب از یک یا چند حرف که با یک کاراکتر به فرم پایانی پایان می‌یابد و با نقطه دنبال نمی‌شود. این فرمت نشان داده شده در مثال (۱۳) است.

ابهامات:

فرمت ۱ بالا با کلمات ابهامی ندارد ، اما ممکن است با نقطه تمام شود که با نقطه پایان جمله دچار ابهام می‌شود.

فرمت ۲ سرنامهایی را نشان می‌دهد که که با کلمات و واژک‌ها ابهام دارد، اما نشاندهندهای بالقوه‌ای برای یک پایان جمله نیستند، زیرا به نقطه ختم نمی‌شوند. این توکن‌ها بعنوان مرکب تلقی می‌شوند زیرا آنها به توکنایزر سطح پایین فرستاده می‌شوند.

۱-۱۰-۲. اختصار

۱-۱۰-۲-۱. آنالیز وصفی

اختصارات می‌توانند بعنوان یک کاراکتر واحد با یا بدون نقطه ظاهر شود، طوریکه در (۱۶) نشان داده شده است. حرف "و" معمولاً نوشته می‌شود بدون یک نقطه از آنجایی که "و" یک کلمه در فارسی است. یک کلمه اگر شامل یک یا چند کاراکتر باشد و کاراکتر آخر قبل از نقطه غیر پایانی باشد مختصر شده است طوریکه در مثال (۱۷) نشان داده شده است، کاراکترهایی مثل "م" و "گ" که فرم پایانی دارند بدون یک کاراکتر فضای خالی " ~ " ظاهر می‌شوند. توجه کنید که اگر کاراکتر آخر از نوعی است که فرم پایانی ندارد این تشخیص دردسترس نخواهد بود، بعنوان مثال (۱۸) را ببینید.

(16) *S*~for *SfHh*~ (=page)
m~for *mylady*~ (=A.D.)
m~.for*mylady*~ (=A.D.)

(17) *Alm.*for*Almany*~ (=German)
*ang*for *anglysy*~ (=English)

(18) *fr.*for*fransh*~ (=French)
*ar.*for *armny*~ (=Armenian)

مثال زیر فرمت عادی برا مختصر کردن نامهای نویسنده را نشان می‌دهد:

(19) *J*~. *m*~.for*Jlal*~ *mtyny*~ (Jalal Matini)

فرمت‌های اختصار به هر حال خیلی مستحکم نیستند. مثال (۲۰) نشان می‌دهد که سه فرم اختصار بکار رفته ممکن در مقالات مختلف مجله ایرانشناسی را برای هجری قمری نشان می‌دهد. طوریکه از این موارد دیده می‌شود، دو کاراکتر که مختصر شده المان لغوی است می‌تواند با نقاطی بدنبال‌شان یا یک نقطه در پایان کاراکتر آخر یا فقط بوسیله فاصله به سادگی بدون هیچ نقطه‌ای جدا شوند. توجه کنید که اگر در همه این موارد کاراکتر اول در فرم غیرپایانی است زیرا کاراکتر آخر به فرم پایانی است.

(20) *h*.q~.
h q~.

h q~

۱-۱۰-۲-۲. ساختارهای مبهم

بار دیگر، هر نقطه ظاهر شده در هر اختصار با نقطه پایان جمله ابهام دارد. هر الگوی اختصار کاراکتر مجزا (مثل مثالهای ۱۶، ۱۹ و ۲۰) با کلمات ابهامی ندارند. اما اگر بوسیله نقطه دنبال شوند، این توکن‌ها ممکن است پایان جمله را نشان دهند. در این موارد توکنایزر باید هر دو احتمال را فراهم کند: توکن فقط بعنوان یک اختصار یا توکن بعنوان یک اختصار نشان‌دهنده پایان جمله. خروجی‌های مبهم مشابه باید وقتی که توکنایزر با یک توکنی به فرمت (۱۷) مواجه می‌شود، در دسترس باشد.

اگر توکنایزر ترکیبی از دو یا چند حرف پایانی را در میان کاراکترهای که فرم پایانی ندارد پیدا کند طوریکه در (۱۸) نشان داده شده است، رشته می‌تواند یک اختصار یا یک کلمه باشد. بعلاوه نقطه با نقطه پایان جمله ابهام دارد، در این موارد توکنایزر سه خروجی تولید خواهد کرد: توکن می‌تواند یک اختصار، یک اختصار نشاندهنده پایان جمله یا یک کلمه نشان‌دهنده پایان جمله باشد.

۱-۱۰-۲-۳. خلاصه

فرمت:

فرمت‌های زیر برای اختصار در متن فارسی در دسترس هستند.

۱- حرف واحد: یک حرف واحد در فرم پایانی ، نقطه اختیاری است.

[Aa-y]\~\.?

این فرمت در مثال (۱۶) نشان داده شده است.

۲- فرم غیرپایانی: یک یا بیشتر حرف که در یک فرم غیرپایانی (تحمیلی) پایان می‌یابد و بعد از آن نقطه قرار می‌گیرد.

[Aa-y]+[FF]\.

که [FF] یک مجموعه از کاراکترهای دارای فرم پایانی است. این قانون نمایش در مثال (۱۷) نشان داده شده است.

۳- کاراکترهای غیرپایانی: یک یا بیشتر حرف که با یک کاراکتر بدون فرم پایانی و یک نقطه پایان می‌پذیرد.

[Aa-y]+[NFF]\.

که [NFF] یک مجموعه از کاراکترهایی است که فرم پایانی ندارند. این قانون در مثال (۱۸) نشان داده شده است.

۴-مقادیر اولیه: حروف مجزای با نقطه و / یا فضای خالی دنبال می‌شود با یک کاراکتر به فرم پایانی. نقطه پایانی اختیاری است.

[Aa-y]\~?[\.] ?[Aa-y]\~?\.?

مثالهای (۱۹) و (۲۰) این قانون را نشان می‌دهند.

ابهامات:

فرمت قوانین ۱، ۲ و ۴ همه توکن‌هایی که با کلمه ابهامی ندارند، اما در صورتی که با نقطه دنبال شوند با پایان جمله ابهام دارند، را نشان می‌دهند. توجه کنید که در قوانین ۱ و ۴ نقطه اختیاری و در قانون اجباری است. توکنایزر دو خروجی را ممکن است تولید کند: در مورد اول توکن فقط بعنوان اختصار تلقی خواهد شد و در مورد دوم توکن یک اختصار نشاندهنده پایان جمله است.

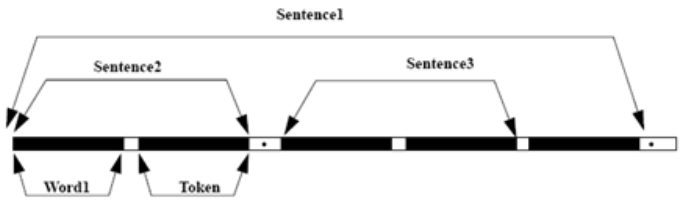
قانون ۳ یک توکن که بالقوه بین اختصار و کلمه ابهام دارد را نشان می‌دهد. بعلاوه، از آنجاییکه با یک نقطه دنبال می‌شود، یک نشانه بالقوه برای پایان جمله نیز است. خروجی پایانی شامل سه مورد ممکن زیر است: توکن یک اختصار، یک اختصار نشاندهنده پایان جمله یا یک کلمه نشاندهنده پایان جمله است.

۱-۱۰-۳. اختصارات، سرنامها و برهمکنش جمله

در این بخش ما بحث‌های قبلی را به عنوان یک نمای کلی برای تمام موارد اختصارات و سرنامها با هم آورده‌ایم. این بخش همچنین قوانین توکنایزشن مورد نیاز برای رفع ابهامات نتیجه شده از برهمکنش این توکن‌ها با حدود جمله را ارائه می‌کند. در زیر کاربردی از قوانین فرمت قسمت‌های قبلی توصیف شده‌اند، توکنایزر سرنامها و اختصارات بالقوه در متن را تشخیص می‌دهد. هر توکن باید نشان داده شود بصورت:

- با ابهام یا بدون ابهام با کلمه.
- با ابهام یا بدون ابهام با یک پایان جمله (مثلاً تعیین اینکه آیا توکن بوسیله نقطه دنبال می‌شود یا نه)

اگر یک توکن با کلمه ابهامی ندارد، سپس توکنایزر توکن را می‌سازد. اگر توکن با یک کلمه ابهام دارد سپس توکنایزر باید همچنین یک توکن کلمه بسازد. بعلاوه اگر توکن با پایان جمله ابهامی ندارد هیچ عملی توسط توکنایزر نباید انجام شود. به هر حال اگر توکن با نقطه دنبال می‌شود سپس توکنایزر باید یک محدوده جمله جدید بسازد طوریکه در شکل زیر نشان داده شده است، بمنظور انجام این توکنایزر نیاز دارد مجموعه جمله باز را کپی کند و مجموعه جدیدی که بخاطر آن ساخته شده است را ببندد (مثل کپی کردن جمله ۱ داخل جمله ۲ در شکل). یک مجموعه جمله جدید سپس باز می‌شود (جمله ۳ در شکل).



شکل۱-۳. ساخت حدود جمله

جدول زیر ترکیبات مبهم ممکن عملی که توکنایزر باید در هر مورد قبول کند را نشان می‌دهد. قوانین فرمت که برای تشخیص توکن‌هایی که در هر مورد شامل می‌شوند بکار رفته است در ستون توکن ورودی مشخص شده‌اند (قانون سرنام ۱ و قانون اختصار ۱ و ۴ که در هر دو حالت اول نشان

داده شده‌اند زیرا نقطه در این موارد اختیاری است).

جدول۱-۷. قوانین توکنایزشن برای قطعه‌بندی سرنامها، اختصارات و حدود جملات

Input token	Tokenizer output
1. Token is not ambiguous with a word/ morpheme Token is not ambiguous with EOS ***Format Rules:*** *Acronym rule 1* *Abbreviation rules 1, 4*	• Create the token as an acronym or an abbreviation • Proceed to the next token
2. Token is not ambiguous with a word/ morpheme Token is ambiguous with EOS ***Format Rules:*** *Acronym rule 1* *Abbreviation rules 1, 2, 4*	• Create the token as an acronym or as an abbreviation • Create sentence boundary • Proceed to the next token. If there is a stop, proceed to the next token following the stop.
3. Token is ambiguous with a word/morpheme Token is not ambiguous with EOS ***Format Rules:*** *Acronym rule 2*	• Create the token as word • Proceed to the next token
4. Token is ambiguous with a word/morpheme Token is ambiguous with EOS ***Format Rules:*** *Abbreviation rule 3*	• Create the token as an abbreviation • Create word token • Create sentence boundary • Proceed to the next token following the stop

۱-۱۱. نتیجه

توکنایزر فارسی بکار رفته در پروژه شیراز از یک توکنایزر سطح پایین مستقل از زبان، برای جدا کردن المانهای متنی به توکن پایه، و یک پس‌توکنایزر شامل اطلاعات زبان خاص که بمنظور اتصال مجدد هر واژک قابل تفکیک به خروجی توکنایزر سطح پایین اعمال می‌شود، استفاده کرده است. از اینرو توکنایزر فارسی می‌تواند بطور موفقیت‌آمیزی بیشتر کلمات متصل را به توکن‌های مجزا، بدون از

دست دادن خمش که در کلمات نشان داده می‌شود، تقسیم کند. باقیمانده کلمات متصلی که در جزء پیش‌پردازش به حساب می‌آیند که به درستی یک فاصله در بین کلمات اضافه می‌کند.

برای یک توکنایزر کاملتر، گرامرهایی برای تشخیص عبارات عددی، تاریخ و زمان را نیز باید شامل شود. بعلاوه، برای رفع ابهام حدود جمله و تشخیص سرنامها و اختصارات، باید خصوصیاتی با توکنایزر ترکیب شود.

۱-۱۲. پیوست

این پیوست شامل یک لیست از پیشوندها و پسوندهای فارسی قابل تفکیک که در اتصال مجدد واژک‌های جدا شده در پس‌توکنایزر بکار می‌رود است. این پیوست ممکن است با کلمات ابهام داشته باشد. توجه کنید که برخی اعضاء این لیست شامل ترکیبی از چند پیوست است.

جدول۱-۸. پیشوند و پسوندهای قابل تفکیک فارسی

Persian Affix	Inflection Information	Affix Type	Ambiguous with word
by	derivational *or* subjunctive particle	prefix	no
my	imperfective verbal particle	prefix	no
nmy	negation + imperfective particle	prefix	no
ha	plural	suffix	yes (interjection 'hey')
hay	plural + ezafe	suffix	yes (interjection 'hey')
ay	indefinite marker	suffix	yes (interjection 'hey')
am	copula/auxiliary/pronoun clitic (1sg)	suffix	yes (noun 'mother' -arabic)
shan	pronoun clitic (3pl)	suffix	yes (noun 'dignity')
tr	comparative	suffix	yes (adjective 'wet')
try	comparative + copula (2sg)	suffix	yes (adj. 'wet' + copula)
ayst	indefinite + copula (3sg)	suffix	yes (noun/interj. 'stop')
ast	auxiliary (3sg)	suffix	yes (copula/3sg 'is')
hayy / haiy	plural + indefinite	suffix	no
haym	plural + pronoun clitic (1sg)	suffix	no
hayt	plural + pronoun clitic (2sg)	suffix	no
haysh	plural + pronoun clitic (3sg)	suffix	no
hayman	plural + pronoun clitic (1pl)	suffix	no
haytan	plural + pronoun clitic (2pl)	suffix	no
hayshan	plural + pronoun clitic (3pl)	suffix	no
hast	plural + copula (3sg)	suffix	no
hayym	plural + copula (1pl)	suffix	no
hayyd	plural + copula (2pl)	suffix	no
haynd	plural + copula (3pl)	suffix	no
at	pronoun clitic (2sg)	suffix	no
ash	pronoun clitic (3sg)	suffix	no
man	pronoun clitic (1pl)	suffix	no
tan	pronoun clitic (2pl)	suffix	no
aym	auxiliary (1pl)	suffix	no

Persian Affix	Inflection Information	Affix Type	Ambiguous with word
ayd	auxiliary (2pl)	suffix	no
and	auxiliary (3pl)	suffix	no
tryn	superlative	suffix	no
trynha	superlative + plural	suffix	no
trha	comparative + plural	suffix	no
trynm	superlative + pronoun clitic (1sg)	suffix	no
trynt	superlative + pronoun clitic (2sg)	suffix	no
trynsh	superlative + pronoun clitic (3sg)	suffix	no
trynman	superlative + pronoun clitic (1pl)	suffix	no
tryntan	superlative + pronoun clitic (2pl)	suffix	no
trynshan	superlative + pronoun clitic (3pl)	suffix	no
; (hamze)	ezafe	suffix	no

مدل‌سازی آماری زبان

۱-۱۳. مقدمه

مدل زبانی آماری برای پیدا کردن نظم زبان طبیعی به منظور بهبود کارآیی برنامه‌های کاربردی گوناگون زبان طبیعی تلاش می‌کند. رویهم‌رفته مدل زبان طبیعی توزیع احتمال واحدهای زبان طبیعی مختلف مثل کلمه، جمله و کل سند را بطور قابل فهمی تخمین می‌زند. مدل زبانی آماری برای برنامه‌های کاربردی تکنولوژی زبانی گوناگون بسیار مهم است. این شامل تشخیص گفتار (که مدل زبانی آماری از آن شروع شد)، ماشین ترجمه، دسته بندی اسناد و مسیریابی، تشخیص کاراکترهای نوری[1]، تشخیص دستخط[2]، بازیابی اطلاعات و تصحیح املا و بسیاری کاربردهای دیگر. در ماشین ترجمه، به عنوان مثال، روشهای آماری در [۲۸] معرفی شده است، اما حتی محققین از روش-های بر اساس قانون استفاده کرده‌اند و فهمیدند که برای معرفی کردن برخی المانهای مدل زبانی آماری و تخمین آماری [۲۹]مفید است. در بازیابی اطلاعات، یک متد مدل‌سازی زبانی اخیراً به‌وسیله [۳۰] پیشنهاد شده است، و یک روش تئوری اطلاعاتی - آماری بوسیله [۳۱]گسترش یافته است. مدل زبانی آماری تکنیک‌های تخمین آماری را روی داده‌های آموزش زبانی - که متن است – بکار می‌برد. بخاطر اینکه طبیعت صریح زبان و فرهنگ لغات وسیعی که مردم به طور طبیعی به کار می‌برند، تکنیک‌های آماری باید یک تعداد از پارامترها را تخمین بزند و بالتبع به میزان فراهم بودن داده‌های آموزشی زیاد وابسته است. در دو دهه اخیر میزان زیادی از متون گوناگون به‌طور روی‌خط در دسترس قرار گرفته‌اند، نتیجتاً در دامنه‌هایی که چنین داده‌هایی در دسترس باشند کیفیت مدلهای زبانی بطور چشمگیری افزایش یافته است. اگر جمع شدن متون روی‌خط با نرخ نمایی ادامه پیدا کند – که این با نرخ رشد وب امکانپذیر خواهد بود – کیفیت مدل زبان طبیعی بکار رفته با فاکتورهای مهمی بهبود پیدا کند. یک تخمین غیر رسمی از IBM نشان می‌دهد مدل ۲-تایی به‌طور موثر با چند صد میلیون کلمه اشباع می‌شود و مدلهای Trigram محتملاً با چند میلیون کلمه اشباع می‌شوند. بیشتر

[1] Optical Character Recognition (OCR)
[2] Handwriting Recognition

تکنیک‌های مدل آماری زبان موفق دانش خیلی کمی نسبت به آنچه زبان واقعاً هست استفاده می‌کنند. بیشتر مدلهای زبانی مشهور – مثل مدل چند-تایی – از این واقعیت که آنچه مدل شده زبان است هیچ استفاده‌ای نمی‌کنند و به‌همان اندازه یک سری سمبل‌های اختیاری بدون هیچ ساختار عمیق، مفهوم یا عقیده پشت آنها می‌تواند باشد. یک دلیل برای این وضعیت که دانش بی اثر شده است این است که تکنیک‌های بهینه‌سازی داده مدل چند-تایی‌ها خیلی موفق هستند و به این خاطر کار روی متدهای برپایه دانش کمتر انجام شده است.

کلمات مهمترین طرفداران متد آماری مدل‌سازی زبان هستند. باید تکیه‌گاه زبان را بر مدل‌سازی زبانی بگذاریم. متاسفانه تعداد کمی تا به امروز روی ترکیب ساختار زبانی و تئوری‌های دانش را در مدل‌های زبانی آماری کار کرده‌اند و بیشتر این تلاشها موفقیت کمی داشته‌اند.

۱-۱۴. مدل‌سازی آماری زبان

۱-۱۴-۱. تعریف و کاربرد

یک مدل زبانی آماری یک توزیع احتمال P(s) روی تمام جملات ممکن s است. مقایسه مدل آماری زبان با زبان محاسباتی آموزنده است. مسلماً این دو فیلد محدوده‌های فازی و همپوشانی زیادی دارند. با این وجود یک روش مشخص کردن این تفاوت به صورت زیر است: فرض کنید S سری کلمات جمله داده شده یعنی فرم ظاهری آن باشد و H برخی ساختارهای مخفی مرتبط با آن – یعنی درخت تجزیه، مفاهیم کلمه و... – مدل آماری زبان بیشتر در مورد تخمین Pr(S) است، در حالیکه زبان محاسباتی بیشتر راجع به تخمین Pr(H|S) است. البته اگر تخمین Pr(S,H) به خوبی تخمین زده شود، هر دوی Pr(S) و Pr(H|S) از آن میتواند به‌دست آید. در عمل این معمولاً شدنی نیست. مدلهای آماری زبان معمولاً در مفهوم دسته‌بندی کننده‌های بیز بکاررفته‌اند در حالی که می‌توانند نقش احتمال اولیه یا تابع درست‌نمایی را بازی کند. برای مثال در تشخیص اتوماتیک گفتار یک سیگنال صوتی a را می‌دهند و هدف یافتن جمله S است که بیشترین احتمال گفته شدن را دارد. با استفاده

از چارچوب‌کاری بیز راه حل این است:

$$s^* = \arg_s \max P(s \mid a) = \arg_s P(a \mid s).P(s) \qquad (۱-۱)$$

که مدل زبانی P(s) نقش احتمال اولیه را بازی می‌کند. در مقابل در دسته‌بندی اسناد، یک سند d داده می‌شود و هدف یافتن کلاس C است که سند متعلق به آن است. نوعاً، نمونه‌های اسناد از هر k کلاس داده می‌شود و نتیجتاً k مدل زبانی مختلف $\{P_1(d), P_2(d), \ldots, P_k(d)\}$ ساخته می‌شوند. با استفاده از یک دسته بندی کننده بیز راه حل c^* است.

$$c^* = \arg_c \max P(c \mid d) = \arg_c P(d \mid c).P(c) \qquad (۲-۱)$$

که مدل نمایی $P_C(d)$ نقش درست نمایی را بازی می‌کند. در یک روش مشابه، یکی می‌تواند نقش مدل زبانی در دسته‌بندی کننده بیز برای دیگر تکنولوژی‌های زبانی گفته شده را به‌دست آورد.

۱-۱۴-۲. معیارهای پیشرفت

برای ارزیابی کیفیت مدل زبانی داده شده، درست نمایی داده‌های جدید به طور عادی بیشتر به‌کار می‌رود. متوسط لگاریتم درست نمایی یک نمونه تصادفی جدید گرفته‌می‌شود به‌وسیله:

$$average\_\log\_likelihood(D \mid M) = \frac{1}{n}\sum_i \log P_M(D_i) \qquad (۳-۱)$$

که $D = \{d1, d2, \ldots, dn\}$ نمونه‌های داده جدید و M مدل زبانی داده شده است. این کمیت می‌تواند به عنوان تخمین تجربی آنتروپی متقاطع توزیع داده درست (اما ناشناخته) نسبت به توزیع مدل P_M نشان داده شود:

$$cross-entropy(P;P_M) = -\sum_{D} P(D).\log P_M(D) \qquad (۴-۱)$$

کارآیی واقعی مدلهای زبانی اکثراً به شکل پیچیدگی بیان شده‌اند:

$$Perplexity(P;P_M) = 2^{cross-entropy(P;P_M)} \qquad (۵-۱)$$

پیچیدگی می‌تواند به عنوان متوسط فاکتور شاخه‌ای زبان بر طبق مدل بیان شود. این یک تابع از زبان و مدل است. وقتی به عنوان یک تابع از مدل مطرح می‌شود، میزان خوبی مدل -که مدل بهتر پیچیدگی کمتر است- را اندازه می‌گیرد، و وقتی به عنوان یک مدل از زبان مطرح می‌شود، آنتروپی یا پیچیدگی زبان را تخمین می‌زند. سرانجام کیفیت مدل زبانی باید با تاثیرش روی کاربرد خاصی که برای آن در نظر گرفته شده است یعنی به‌وسیله تاثیرش رو نرخ خطای آن برنامه کاربردی اندازه‌گیری شود. به هر حال، نرخ خطا معمولاً توابع غیر خطی و بسیار ناچیز بصورت توابع ضمنی از مدل زبانی هستند. کاهش پیچیدگی معمولاً باعث کاهش نرخ خطاست، اما میزان زیادی نمونه شمرده شده در نوشته‌ها وجود دارد. به عنوان یک قانون سرانگشتی کاهش ۵ درصدی پیچیدگی معمولاً ارزش عملی ندارد. یک کاهش ۲۰ -۱۰ درصدی قابل توجه است و معمولاً – نه برای همیشه – به عنوان بهبود در کارآیی مطرح می‌شود.، یک بهبود ۳۰ درصدی یا بیشتر کاملاً قابل توجه و نایاب است. برای تعبیه معیاری که با نرخ خطای برنامه کاربردی بهتر از پیچیدگی – که درعین حال برای بهینه‌سازی از خود نرخ خطا آسانتر است - همبسته باشد تلاشهایی صورت گرفته که موفقیتهای محدودی داشته‌اند. در حال حاضر پیچیدگی معیار مقدم برای ساختار مدل زبانی را دنبال خواهد کرد. برای جزئیات بیشتر [۳۴] را ببینید.

۱-۱۴-۳. نقاط ضعف آشکار مدل‌های فعلی

حتی ساده‌ترین مدل زبانی یک تاثیر جدید روی برنامه کاربردی که در آن استفاده می‌شود دارد، اینرا می‌توان مثلاً با حذف مدل زبانی از سیستم تشخیص گفتار مشاهده کرد. به هر حال تکنیک‌های

مدل زبانی جاری دور از کمال مطلوب هستند. دلایل برای این از چندین منبع می‌آید:

شکنندگی سرتاسر دامنه‌ها: مدل زبانی فعلی به‌شدت نسبت به تغییرات در سبک، موضوع یا نوع متن که آموزش دیده حساس است. برای مثال، برای مدل‌سازی مکالمات تلفنی، ۲ میلیون کلمه رونوشت از چنین مکالماتی غنی‌تر از ۱۴۰ میلیون کلمه رونوشت از اخبار رادیو تلوزیون است. این تاثیر حتی برای تغییراتی که از نظر انسان جزئی می‌آید کاملاً قوی است: یک مدل زبانی آموزش دیده با متون مجله آنلاین دو-جونز وقتی برای متون مجله آن لاین اسیوشیتدپرس که بسیار به آن شبیه است در همان دوره زمانی به‌کار می‌رود پیچیدگی دو برابر مشاهده می‌شود.

فرض استقلال غلط: به منظور باقی ماندن نظم، همه تکنیکهای مدل‌سازی زبانی موجود برخی اقسام استقلال را در میان بخش‌های جزئی همان سند فرض می‌کنند. برای مثال رایج‌ترین مدل بکاررفته، مدل چند-تایی، فرض می‌کند که احتمال کلمه بعدی در یک جمله فقط به خصوصیات n-1 کلمه قبل بستگی دارد. با این حال، حتی یک نگاه سرسری به هر متن طبیعی غلط بودن فرض را به‌طور آشکار اثبات می‌کند. فرضیات استقلال غلط در مدلهای آماری معمولاً منجر به توزیع خیلی تیزی می‌شود، این صریحاً آنچه در مدل زبانی اتفاق می‌افتد است، بطوریکه برای مثال می‌تواند در دسته‌بندی اسناد دیده شود. احتمال محاسبه شده بوسیله معادله ۵-۲ به شدت تیز است، واقعاً رسیدن یکی از کلاسها به یک و صفر شدن برای بقیه. این قطعاً یک احتمال درست نمی‌تواند باشد از آنجایی که متوسط نرخ خطای دسته بندی بزرگتر از صفر است.

آزمایشات شیوه شانون: کلود شانون[1]، تکنیک استخراج دانش انسانی را از زبان بوسیله جویاشدن موضوعات انسانی برای پیش‌بینی المان بعدی متن را پایه‌گذاری کرد[۳۷،۳۶]. شانون این تکنیک را برای محدود کردن آنتروپی انگلیسی استفاده کرد. [۳۸] یک وضع تصادفی را فرموله کرد و آنرا برای بدست آوردن تخمین آنتروپی انگلیسی استفاده کرد. در ۱۹۸۰ گروه تحقیقات زبان و گفتار در IBM آزمایشات شیوه شانون را انجام دادند، که منابع بالقوه برای بهبود مدل‌سازی زبانی بوسیله مشاهده و آنالیز کارآیی موضوعات انسانی در پیش‌بینی یا تصحیح متن مشخص شدند. این آزمایشات توسط چندین محقق دیگر صورت پذیرفته است. برای مثال آزمایشاتی در جهت تصدیق توان بهبود

[1] Cloud Shannon

مدل‌سازی زبانی در محیط‌های زبانی خاص انجام شده است [۳۹]. یک مشاهده معمولی در طی تمام این آزمایشات این است که مردم به راحتی، به صورت عادی و به صورت اساسی کارآیی مدل زبـانی را بهبود می‌بخشند. اینها ظاهراً با استفاده از استدلال در زبان، حس عام و سطوح دامنه انجام می‌شود.

۱-۱۵. بررسی تکنیک‌های اصلی مدل آماری زبان

در این قسمت بطور خلاصه تکنیک‌های مهم ثابت شده مدل آماری زبان را بررسی می‌کنیم. بـرای یک شرح کار تکنیکی تشریح شده [۴۰] را ببینید. تقریباً همه مدلهای زبانی در حال حاضـر احتمـال یک جمله را به حاصل‌ضربی از احتمالات شرطی تجزیه می‌کنند.

$$\Pr(s) \stackrel{def}{=} \Pr(w_1,...,w_n) = \prod_{i=1}^{n} \Pr(w_i \mid h_i) \qquad (۶-۱)$$

که W_i، i امین کلمه جمله است و $hi \stackrel{\text{def}}{=} \{ W_1 , W_2 ,..., W_{i-1}\}$ سابقه نامیده می‌شود.

۱-۱۵-۲. مدل مدل چند-تایی

مدل چند-تایی‌ها قسمت اصلی تکنولوژی تشخیص گفتـار فعلـی هسـتند. واقعـاً همـه محصـولات تشخیص گفتار تجاری از برخی انواع مدل چند-تایی‌ها استفاده می‌کنند. یک مدل چند-تـایی مسـئله تخمین بوسیله مدل‌سازی زیان را از لحاظ ابعادی مانند یک منبع مارکوف مرتبه n-1 کاهش می‌دهد:

$$P(w_i \mid h_i) \approx P(w_i \mid w_{i-n+1},...,w_{i-1}) \qquad (۷-۱)$$

مقدار n پایداری تخمین – مثل واریانس - را جایگزین تناسب آن – مثل بایاس – می‌کنـد. یـک

trigram (n=3) یک انتخاب معمولی با مجموعه آموزشی بزرگ (میلیونها کلمه) است درحالیکه یک ۲-تایی (n=2) اغلب برای میزان کمتری از آن استفاده می‌شود. احتمالات trigram و حتی ۲-تایی بدست آمده حتی با مجموعه‌های خیلی بزرگ هنوز یک مسئله تخمین پراکنده است. برای مثال بعد از مشاهده همه trigramهای (مثل کلمه‌های سه‌قلو پشت سرهم) ۳۸ میلیون کلمات با ارزش مقالات روزنامه، یک ثلث کامل از trigram ها در مقاله‌های جدید از همان منبع، جدید هستند[۸]. بعلاوه حتی در میان trigramهای مشاهده شده تعداد زیادی یک بار اتفاق افتاده است و اکثریت باقیمانده مقادیر کم مشابهی دارند. بنابراین تخمین حداکثر درست‌نمایی درست احتمالات مدل چند-تایی با شمردن توصیه نمی‌شود. در عوض تکنیک‌های هموارسازی گوناگونی ایجاد شده‌اند. این شامل کاستن تخمین حداکثر درست نمایی [۴۲،۴۱]، پس‌پیچاندن[1] به طور بازگشتی برای کم کردن مرتبه مدل چند-تایی‌ها [۴۵-۴۳]و درونیابی خطی[2] مدل چند-تاییهای مراتب مختلف[۴۶]. روش‌های دیگر شامل مدل چند-تایی با طول متغییر [۵۱-۴۷] بعلاوه یک متد شبکه بندی[3] [۵۲] است. برای مقایسه و تکمیل تکنیک‌های هموارسازی تحت شرایط مختلف کارهای زیادی انجام شده است. یک آنالیز خوب ممکن است در [۵۳] پایه گذاری شده باشد. بعلاوه تولکیت‌ها[4] تکنیک‌های مختلفی را اجرا می-کنند که در[۵۷-۵۴] منتشر شده است. در عین حال روش دیگر برای مبارزه با تنکی از راه خوشه-بندی[5] کلمات است. فرض کنید C_i کلاس کلمه W_i باشد. سپس ساختار مدلی مختلف می‌تواند بکار رود. برای مثال برای یک trigram:

$$\Pr(w_3 | w_1, w_2) = \Pr(w_3 | C_3).\Pr(C_3 | w_1, w_2) \qquad (۸-۱)$$

$$\Pr(w_3 | w_1, w_2) = \Pr(w_3 | C_3).\Pr(C_3 | w_1, C_2) \qquad (۹-۱)$$

[1] Backing off
[2] Linearly Interpolating
[3] Lattice
[4] Toolkit
[5] Clustering

$$\Pr(w_3 | w_1, w_2) = \Pr(w_3 | C_3).\Pr(C_3 | C_1, C_2) \qquad (۱۰-۱)$$

$$\Pr(w_3 | w_1, w_2) = \Pr(w_3 | C_1, C_2) \qquad (۱۱-۱)$$

کیفیت مدل بدست آمده قطعاً به خوشه‌بندی C() بستگی دارد. در دامنه‌های ادای کوتاه – مثل ATIS [۵۸]– اغلب بوسیله خوشه‌بندی دستی دسته‌های معنایی [۵۹] نتایج خوبی به‌دست می‌آید. اما در دامنه‌های که کم بودن به آنها تحمیل شده است خوشه‌بندی دستی با دسته‌های زبانی (مثل نحوی) معمولاً مدل بر اساس کلمه را بهبود نمی‌بخشد. خوشه‌بندی چرخشی با استفاده از ضوابط تئوری اطلاعات بکار رفته روی مجموعه‌های بزرگ می‌تواند بعضی اوقات پیچیدگی را در حدود ۱۰ درصد کاهش دهد اما فقط وقتی مدل بوسیله همتای بر اساس کلمه‌اش درونیابی شد.

۱-۱۵-۳. مدل درختهای تصمیم‌گیری

درخت تصمیم‌گیری و الگوریتم‌های شیوه کارت[1] [۶۲] اولین بار روی مدل زبانی بوسیله [۶۳] بکار رفته است. یک درخت تصمیم گیری می‌تواند به صورت دلخواهانه فضای سابقه را، بوسیله پرسیدن سئوالات دودوئی اختیاری در مورد سابقه h در هریک از نودهای داخلی، تقسیم کند. داده‌های آموزش در هر برگ برای ساخت توزیع احتمال Pr(w|h) روی کلمه بعدی به کار می‌روند. برای کاهش واریانس تخمین، این توزیع برگ با توزیعهای نود داخلی یافت شده در طول مسیر تا ریشه درونیابی می‌شود. مطابق معمول درختها با انتخاب حریصانه سئوال دارا اطلاعات مفیدتر، در هر نود، رشد پیدا می‌کنند (مانند تشخیص دادن بوسیله کاهش آنتروپی). هرس و وارسی اعتبار نیز به کار برده می‌شود.

انجام تکنولوژی کارت در مدلسازی زبان کاملاً یک چالش است: فضای سابقه‌ها خیلی بزرگ است ($۱۰^{۱۰۰}$ برای یک سری ۲۰ کلمه‌ای روی یک فرهنگ لغت ۱۰۰۰۰۰کلمه‌ای) و حتی فضای سئوالات

[1] Cart

ممکن خیلی بزرگتر است ($2^{10^{100}}$). حتی اگر سئوالات به کلمات مجزا در سابقه محدود شود هنوز 20.2^{10^5} چنین سئوالاتی وجود دارد. بایاس بسیار قوی باید با محدود کردن کلاس سئوالاتی که باید درنظر گرفته شود و بکاربردن الگوریتمهای جستجوی حریصانه مطرح شود. برای تایید سئوالات تک‌کلمه‌ای بهینه در یک نود داده شده، الگوریتم‌ها برای تقسیم بندی دودوئی بهینه سریع لغات بسط یافتند[۶۴]. تلاش‌های اولیه در مدل زبانی شیوه کارت [۶۵] از یک پنجره سابقه ۲۰ کلمه‌ای و سئوالات محدود به کلمات مجزا استفاده کرد، اگرچه سئوالات پیچیده را به صورت ترکیبی از سئوالات ساده می‌پذیرفت. یادگیری آن ماههای زیادی طول کشید و نتایج کمتر از حد انتظار بود: یک کاهش ۴ درصدی در پیچیدگی روی trigram پایه و کاهش ۹ درصدی وقتی درونیابی می‌شد. در تلاش دوم [۶۵] بایاس قوی‌تری مطرح شد: ابتدا فرهنگ لغات به یک سلسله مراتب دودوئی همانطور که در [۶۰] آمده است خوشه‌بندی می‌شود، و به هر کلمه یک نمایش رشته بیتی مسیری که از ریشه به آن می‌رسد اختصاص می‌یابد. سپس، سئوالات درخت به مهمترین بیتهای که در هر کلمه در سابقه هنوز ناشناخته‌اند محدود می‌شود. این مجموعه کاندید را به یک مشت سئوال در هر نود کاهش می‌دهد. متاسفانه نتایج در اینجا ناامید کننده و متد تا حد زیادی متروک است. از لحاظ تئوری، درخت‌های تصمیم‌گیری آخرین مدلهای مبتنی بر تقسیم‌بندی را نشان می‌دهد. این محتمل است که درخت‌هایی وجود داشته باشد که کارآیی آنها از مدل چند-تایی‌ها بیشتر باشد، اما یافتن آنها به علت محاسباتی و تنکی داده‌ها مشکل به نظر می‌رسد.

۱-۱۵-۴. مدل انگیزش زبانی[1]

تا موقعی که همه مدل‌های آماری زبان برخی الهامات را از یک مشاهده مستقیم از زبان به دست می‌آورند، در بیشتر مدلها محتوای زبان واقعی کاملاً قابل چشم‌پوشی است. تکنیک‌های مختلف مدلسازی آماری زبان، هنوز از گرامرهایی که بطور مستقیم بوسیله زبان‌شناسان استفاده می‌شود به

[1] Linguistically motivated models

دست می‌آید. گرامر مستقل از متن، هنوز یک مدل خوب فهم زبان طبیعی است. یک مدل مستقل از متن بوسیله کلمات، یک مجموعه از سمبل‌های غیرپایانی و یک مجموعه از قوانین تولید یا انتقال تعریف می‌شود. جملات ساخته شده، با یک غیر پایانی اولیه شروع می‌شوند، با به کار بردن مکرر قوانین انتقال ، هر تبدیل یک غیرپایانی را به یک سری پایانی – یعنی کلمات – و غیرپایانی‌ها تبدیل می‌کند تا زمانی که یک سری فقط پایانی به‌دست آید. گرامرهای مستقل از متن خاص بر اساس مجموعه‌های تفسیر شده و تجزیه شده مانند [۶۶] با پوشش خوب اما هنوز ناقص داده‌های جدید ساخته می‌شوند.

یک گرامر مستقل از متن[1] احتمالی یک توزیع احتمال روی انتقال ناشی از هر غیر پایانی قرار می‌دهد، در نتیجه شامل یک توزیع روی مجموعه همه جملات است. این احتمالات انتقال از روی مجموعه‌های تفسیر شده بوسیله الگوریتم داخل – خارج[2] [۶۷]و الگوریتم حداکثر تخمین[3] [۶۸] می‌تواند تخمین زده شود. به هر حال، سطح درست‌نمایی این مدلها منجر می‌شود ماکزیمم‌های محلی زیادی را دربر بگیرد و نقاط درست‌نمایی حداکثر محلی که بوسیله الگوریتم پیدا شده است معمولاً جزئی از ماکزیمم‌های مطلق قرار می‌گیرد. بعلاوه، حتی اگر تخمین حداکثر درست‌نمایی مطلق میسر باشد، آن معتقد است که احتمالات انتقال حساس به متن نیاز دارند برای رفتار واقعی زبان بقدر کافی مناسب باشد. متاسفانه هیچ الگوریتم آموزش موفقی برای این وضعیت شناخته نشده است.

با وجود این، [۶۹] با موفقیت منابع دانش گرامر مستقل از متن را با یک مدل آماری زبان ترکیب کرده و یک کاهش ۱۵ درصدی در نرخ خطای تشخیص گفتا ر روی دامنه ATIS به دست آورده است. آنها همچنین بوسیله تجزیه کردن گفتار با یک گرامر مستقل از متن برای فراهم کردن یک سری از انواع مختلف قطعات گرامری، و سپس ساخت یک trigram از انواع مختلف گرامری انجام می‌گیرد تا جای مدل چند-تایی استاندارد را بگیرد.

گرامر اتصال[4]، یک گرامر وابسته به فرهنگ لغات است که بوسیله [۷۰] پیشنهاد شده است، هر کلمه با یک یا بیشتر مجموعه مرتب شده از انواع اتصالات روابط مرتبط است. هر اتصال باید به اتصال

[1] Context free grammar (CFG)
[2] Inside-Outside algorithm
[3] Estimation-Maximization (EM) algorithm
[4] Link Grammer

نوع مشابه کلمه دیگر متصل شود. یک تجزیه قانونی شامل ارضاء همه اتصالات جمله از طریق یک گراف مسطح است. گرامر اتصالات همان توان قابل توجه مانند یک گرامر مستقل از متن را دارد. اما به طور تقریبی بهتر از درک مستقیم زبانی وفق پیدا می‌کند. یک گرامر اتصال برای انگلیسی بطور دستی با یک پوشش خوب ساخته می‌شود. فرم احتمالی گرامر اتصال نیز در [۷۱] کار شده است. گرامر اتصال به گرامر وابستگی که در قسمت ۵-۴ بحث می‌شود وابسته است.

۱-۱۵-۵. مدل نمایی

همه مدلهایی که تا اینجا بحث شدند به قطعه‌قطعه شدن داده تن در داده‌اند، چونکه مدل‌سازی با جزئیات بیشتر لزوماً باعث می‌شود هر پارامتر جدید با داده‌های کمتر و کمتری تخمین زده شود. این در درخت تصمیم‌گیری خیلی مشهود است، چونکه وقتی درخت رشد می‌کند برگ‌ها نقاط داده کمتر و کمتری را شامل می‌شود. قطعه‌قطعه شدن می‌تواند بوسیله استفاده از مدل نمایی به فرم زیر جلوگیری شود:

$$P(w \mid h) = \frac{1}{Z(h)} . \exp[\sum_i \lambda_i f_i(h,w)] \qquad (۱۲-۱)$$

که λ_i پارامترها هستند و Z(h) یک ترم نرمال‌سازی است و خصوصیات $f_i(h,w)$ توابع اختیاری سوابق جفت کلمه‌ها است. یک مجموعه آموزش داده شده است، حداکثر درست‌نمایی تخمین زده شده می‌تواند برای ارضاء محدودیت‌ها نشان داده شود:

$$\sum_h \tilde{P}(h) . \sum_w P(w \mid h) . f_i(h,w) = E_{\tilde{P}} f_i(h,w) \qquad (۱۳-۱)$$

که $\tilde{P}$ یک توزیع تجربی مجموعه آموزش است. می‌توان نشان داد که تخمین حداکثر درست‌نمایی می‌تواند با توزیع حداکثر آنتروپی همزمان اتفاق بیافتد[۷۲] ، یعنی یکی که بالاترین آنتروپی را در

میان توزیع‌ها دارد معادله ۵-۱۳ را ارضا می‌کند. این راه حل حداکثر درست‌نمایی / حداکثر آنتروپی بی نظیر می‌تواند با یک فرآیند چرخشی انجام گیرد[۷۴،۷۳].

الگوی حداکثر آنتروپی و چارچوب‌کاری عمومی‌تر MDI برای مدل زبانی بوسیله [۷۵] پیشنهاد شد و سپس موفقیت زیادی کسب کرد[۷۷،۷۶،۳۵]. استحکام آن در ترکیب کردن منابع دانش اختیاری است در حالیکه از قطعه‌قطعه کردن خودداری می‌شود. برای مثال در[۳۵] مدل چند-تاییهای متداول، مدل چند-تاییهای فاصله ۲ و جفت کلمات با فاصله زیاد بعنوان خصیصه کد شده‌اند و کاهش بیش از ۳۹ درصدی پیچیدگی و کاهش نرخ خطای بیش از ۱۴ درصد تشخیص گفتار روی پایه trigram نتیجه شد.

در حالیکه مدل حداکثر آنتروپی برازنده و کلی است، اما ضعف های خودش را دارد. آموزش یک مدل حداکثر آنتروپی از نظر محاسباتی چالش انگیز و بعضی اوقات کاملاً نشدنی است. استفاده از یک مدل حداکثر آنتروپی به قدرت پردازنده مرکزی وابسته است، زیرا نیاز به نرمال‌سازی آشکار دارد. مدل حداکثر آنتروپی نرمال نشده در [۷۸] کار شده است. حداکثر آنتروپی هموار شده در [۷۹] آنالیز شده است. موفقیت نسبی مدل حداکثر آنتروپی توجه‌اش را روی مسئله باقیمانده استنتاج خصیصه‌ها - یعنی انتخاب خصیصه‌های مفیدی که مدل شامل آنها است - خصیصه متمرکز می‌کند. یک فرآیند چرخشی اتوماتیک برای انتخاب خصیصه‌ها از یک مجموعه کاندید داده شده در [۷۴] توصیف شده است. باقیمانده مدل زبانی حداکثر آنتروپی موضوع تحقیقی قوی است. نمونه‌های [۸۵-۸۱] را ببینید.

۱-۱۵-۶. مدل وفقی

تا اینجا ما با زبان به عنوان یک منبع همگن برخورد کردیم، اما در واقع زبان طبیعی، با ضوابط، انواع و شیوه‌های گوناگون بسیار ناهمگن است.

در توافق دامنه متقاطع، داده‌های مورد استفاده در زمان آموزش مدل زبانی با داده‌های زمان تست متفاوت است. اطلاعات وفقی مفید فقط در سند جاری است. یک تکنیک کاملاً موثر و معمولی برای بهره‌گیری این اطلاعات مخزن است: سابقه – که بطور پیوسته رشد پیدا می‌کند – برای ساخت مدل چند-تایی دینامیک $P_{cache}(w \mid h)$ در زمان اجرا که با یک مدل ایستا درونیابی شده است

استفاده می‌شود:

$$P_{adaptive}(w \mid h) = \lambda P_{static}(w \mid h) + (1-\lambda) P_{cache}(w \mid h) \qquad (۱۴-۱)$$

با وزن λ که روی داده‌های خارجی بهینه‌سازی شده است. مخزن مدل‌های زبان اول توسط [۸۷،۸۶] معرفی شد. [۸۹،۸۸] کاهش در پیچیدگی و [۹۰] کاهش در نرخ خطای تشخیص را گزارش کردند و [۹۱] طرح وفقی دیگری را ارائه کرد.

در توافق دامنه درونی، داده تست از همان منبع آموزش می‌آید اما دومی ناهمگن است، شامل زیر مجموعه‌های زیادی با موضوعات و شیوه‌های مختلف است. انطباق از مراحل زیر به‌دست می‌آید:

۱. خوشه‌بندی کردن مجموعه آموزش همراه با ابعاد متغییر مثل موضوع [۹۲]
۲. مشخص کردن موضوع یا مجموعه موضوعات داده تست در زمان اجرا [۹۴،۹۳]
۳. تعیین زیرمجموعه‌های مناسب از مجموعه آموزش و استفاده از آنها برای ساخت یک مدل خاص
۴. ترکیب مدل خاص با مدلی که مجموعه وسیعی دارد (در اصطلاح علم آمار، کوچک شدن مدل خاص نسبت به مدلی که کلی‌تر است، برای تقابل واریانس اولی در برابر بایاس دومی). این معمولاً با درونیابی خطی در سطح احتمال کلمه یا سطح احتمال جمله انجام می‌شود [۹۲].

یک نمونه خاص و خیلی معمولی اینست که مقدار کمی داده در دامنه هدف و مقدار زیادی در دامنه دیگر وجود دارد، نتیجه اغلباً ناامید کننده است. بهرحال داده آموزش خارج از دامنه بطور شگفت‌انگیزی مزایای بسیار کمی دارد. در این حالت مرحله مناسب ترکیب مدل‌های دو دامنه است. برای مثال وقتی مدل‌سازی روی دامنه گفتار محاوره‌ای [۹۵]، ۴۰ میلیون کلمه از مجموعه مجله وال استریت (مقالات روزنامه [۹۶]) و ۱۴۰ میلیون کلمه از مجموعه BN (رونویسی اخبار پخش شده از رادیو و تلوزیون [۹۷])، فقط درصد کمی کارآیی برنامه کاربردی مدل آموزش دیده بر اساس دامنه را روی یک جزء ۲.۵ میلیون کلمه‌ای بهبود می‌بخشد. اگرچه این یک بهبود قابل توجه روی چنین مجموعه سختی است، با این وجود با توجه به میزان داده زیادی که شامل شده است ناامید کننده است. با توجه به برخی تخمین‌ها[۹۷] ۱ میلیون داده در دامنه بیش از ۳۰ میلیون کلمه خارج از

دامنه مدل را کمک می‌کند. این اشاره می‌کند که تکنیکهای وفقی ما خام هستند.

۱-۱۶. دستورالعمل‌های متداول امیدبخش

این بخش مسیر تحقیقات فعلی را که امید زیادی می‌دهد را بررسی می‌کند.

۱-۱۶-۱. مدل‌های وابستگی

گرامرهای وابسته جملات را برحسب روابط جفت کلمات نامتقارن توصیف می‌کند. هر کلمه در جمله به یک کلمه دیگر وابسته است، با یک استثناء - والد یا راس - که به عنوان سر جمله کامل عمل می‌کند. برای اطلاعات بیشتر راجع به گرامرهای وابستگی [۹۹] را ببینید. گرامرهای وابستگی احتمالی به همراه الگوریتم‌هایی برای آموزش آنها نیز ایجاد شده‌اند[۱۰۰]. گرامرهای وابستگی احتمالی مخصوصاً مناسب مدلسازی حالت مدل چند-تایی هستند، که هر کلمه بر اساس تعداد کمی از کلمات دیگر پیش‌بینی می‌شود. تفاوت کلی این است که در مدل چند-تایی متداول ساختار مدل مشخص است: هر کلمه باتوجه به یک تعداد کلمه قبل از آن پیش‌بینی می‌شود. در گرامر وابستگی کلمات به عنوان پیش‌بینی‌کننده وابسته به گراف وابستگی – که یک متغییر مخفی است – عمل می-کنند. یک پیاده‌سازی نوعی یک جمله s را برای ساخت گراف وابستگی محتمل‌تر G_i (با احتمالات وابسته $P(G_i)$) تجزیه می‌کند، برای هر یک از آنها یک احتمال تولید $P(S|G_i)$ (حالت مدل چند-تایی یا شاید بعنوان مدل حداکثر آنتروپی) محاسبه می‌کند و سرانجام احتمال جمله کامل بعنوان $P(s)$ $P(s|G_i).P(G_i) \approx \sum_I$ تخمین زده می‌شود (این فقط یک تقریب است به خاطر اینکه $P(Gi)$ ها خودشان از جمله s بدست می‌آیند). بعضی اوقات $P(s)$ زودتر تقریب زده می‌شود بعنوان $P(s|G^*)$ که G^* بهترین امتیازبندی تجزیه واحد است. یک مثال از این [۱۰۱] است، که تجزیه کننده [۱۰۲] را برای ساخت اجزاء کاندید استفاده می‌کند و پارامترها را با حداکثر آنتروپی آموزش می‌دهد. گرامر روابط احتمالی [۷۱] نامبرده در بخش ۵-۳-۴ تقریباً در این دسته قرار می‌گیرند. اخیراً [۱۰۳] یک

تجزیه کننده که از پارامترگذار احتمالی یک اتوماتای پائین به بالا استفاده می‌کند را بکار برده است، الگوریتم حداکثر آنتروپی را برای آموزش استفاده کرد و به نتایج امیدوار کننده‌ای – ۱ درصد کاهش نرخ خطای تشخیص روی مجموعه انتخابی مشکل – رسید. جمعاً، این متد ساختار زبانی مخفی با قوانین زنجیری پارامتری کردن می‌تواند به یک زمینه زبانی و درعین‌حال مدل محاسباتی نرم منجر شود.

۱-۱۶-۲. کاهش ابعادی

یکی از دلایلی که مدل‌سازی آماری زبان مشکل است، این است که با تعداد زیادی دسته و دامنه به ظاهر قطعی است. یک مثال اولیه فرهنگ لغات هستند. در بیشتر مدلهای زبانی فرهنگ لغات یک مجموعه خیلی بزرگ از ورودیهای نامرتبط هستند BANK اونقدری که نزدیک به BRAZIL است نزدیک BANKS یا LOAN نیست. این در یک تعداد زیادی پارامتر نتیجه می‌شود. هنوز درک مستقیم زبانی ما این است که، یک ساختار خیلی زیادی از روابط بین کلمات وجود دارد. احساس می‌شود که ابعاد درست فرهنگ لغات باید کمتر باشد. بطور مشابه، برای دیگر پدیده‌ها در زبان فضای اصولی می‌تواند از نظر ابعادی کم یا متوسط باشد. انطباق موضوع را در نظر بگیرید. همانطوریکه موضوع تغییر می‌کند احتمالات فرهنگ لغات تغییر می‌کند.از آنجائیکه هیچ دو سندی واقعاً در یک مورد نیستند، یک متد درست پارامترهای زیادی را نیاز دارد. فضای موضوع اصلی می‌تواند بطور معقولانه در ابعاد کمتری مدل شود. این انگیزه پشت [۱۰۴] است که تکنیک‌های آنالیز معنایی پنهان را [۱۰۵] برای کاهش ابعاد فرهنگ لغت در فضای موضوع در یک زمان استفاده می‌کند. ابتدا رخداد هر کلمه فرهنگ لغت در هر سند فهرست می‌شود. این ماتریس خیلی بزرگ از طریق تجزیه مقدار یکتایی در یک ابعاد کمتر - معمولاً ۱۵۰-۱۰۰ - کاهش می‌یابد. ماتریس کوچک جدید بیشتر همبستگی‌های برجسته را بین ترکیبات خاص کلمات از یک سو و خوشه‌های اسناد از سوی دیگر ثبت می‌کند. تجزیه نیز به ماتریسی که از فضای سند و فضای کلمه در فضای ترکیبی جدید تصویر می‌شود منجر می‌شود. بالنتیجه، هر سند جدید می‌تواند به فضای ترکیبی تصویر شود، از اینرو بعنوان یک ترکیبی از موضوعات ضمنی اساسی و وفق‌پذیر بطور موثر خوشه‌بندی شود. در [۱۰۴] این نوع

وفق‌پذیری با مدل چند-تایی ترکیب شده است و یک کاهش ۳۰ درصدی بر پایه trigram را گـزارش کرده است. در [۱۰۶] تکنیک بیشتر رشـد یافتـه و کـاهش خطـای ۱۶ درصـدی روی پایـه trigram گزارش شده است.

۱-۱۶-۳. مدل‌های جمله کامل[1]

همه مدلهای زبانی که تا اینجا توصیف شده‌اند از قوانین زنجیری برای تجزیه احتمال جمله به یک حاصلضرب احتمال شرطی P(w|h) استفاده کرده‌اند. این برای آسان کردن تخمین با شـمارش نسـبی انجام شده است. این تجزیه به ظاهر بی‌ضرر است. سرانجام این یک تخمین نیست، یک شباهت واقعی است. بعنوان نتیجه مدل‌سازی زبانی کلاً مدل‌سازی توزیع یک کلمه مجزا را کاهش می‌دهـد. ایـن در عوض ممکن است منع قابل توجهی در سـاختار مـدل‌سـازی زبـانی باشـد: برخـی پدیـده‌هـای زبـانی غیرممکن یا منتهای مراتب اندیشیدن در مورد آنها در یک چهارجوب کاری شرطی نادرست باشد. این شامل خصیصه‌های سطح جمله همچون مطابقهٔ نحوی عدد و شخص، ربـط معنـایی، قابلیـت تجزیـه کردن و حتی طول است. بعلاوه تاثیر خارجی روی جمله – مثل جمله گذشته یـا موضـوع – بایـد در پیش‌بینی هر کلمه بعنوان فاکتور در نظر گرفته شود که می‌تواند باعث بایاسهای کوچک روی ترکیـب شود. برای نشان دادن این موضوع [۸۰] یک مدل نمایی کل جمله را پیشنهاد کرد:

$$P(s) = \frac{1}{Z}.P_0(s).\exp[\sum_i \lambda_i f_i(s)] \qquad (۱۵-۱)$$

در مقایسه با مدل نمایی شرطی معادله ۵-۱۲، Z یک ثابت صحیح است که وزن نرمال‌سازی را کم می‌کند. بطور مهم، خصیصه‌های $f_i(s)$ می‌تواند خواص دلخواه جمله کامل را بگیرد.آمـوزش ایـن مـدل نمونه‌برداری از یک توزیع نمایی را نیاز دارد.

[1] Whole Sentence Model

استفاده از زنجیره مارکوف مونت کارلو و دیگر متدهای نمونه‌برداری برای زبان در [۱۰۷] مورد مطالعه قرار گرفته است. نمونه‌برداری موثر بسیار سخت است، در نتیجه، تنگنا در این مدل تعداد خصیصه‌ها یا میزان داده نیست اما ترجیحاً میزان کم بودن و دقت آنها برای مدل‌سازی نیاز است. بطور قابل توجهی این در [۱۰۸] نشان داده شده است که بیشتر مزایا از خصیصه‌ای عمومی بدست می‌آید. خصیصه‌های بر پایه تجزیه در [۱۰۹] بررسی شده است و خصیصه‌های معنایی در [۸۰] ثبت شده است. یک متدولوژی محاوره‌ای استنتاج خصیصه در [۸۰] پیشنهاد شده است. این متدولوژی منجر به یک فرمولاسیون مسئله آموزش بعنوان رگرسیون منطقی با مزایای کمکی قابل توجه روی آموزش حداکثر درست نمایی شده است.

۱-۱۷. چالش‌ها

شاید بیشترین جنبه ناامید کننده مدل‌سازی زبانی آماری درک مستقیم ما بعنوان گوینده زبان طبیعی و طبیعت ساده بیشتر مدلهای موفق ما است. بعنوان سخنگوی محلی، ما احساس می‌کنیم که زبان ساختار عمیقی دارد. بااین‌حال ما هنوز مطمئن نیستیم که چگونه شمرده سخن بگوییم که ساختار در یک چهارچوب احتمالی قرار بگیرد. تئوری‌های زبانی ثابت شده در اینجا کمک کمی به ما می‌کنند ، احتمالاً بخاطر اینکه هدفشان معین کردن حد بین آنچه در زبان درست است و آنچه درست نیست می‌باشد در حالیکه اهداف مدل‌سازی زبانی آماری کاملاً متفاوت است.

بعنوان یک مثال مسئله خوشه‌بندی کلمات فرهنگ لغت را که در بخش ۵-۲-۳ بحث شده، چون چندین متد چرخشی اتوماتیک پیشنهاد شده است [۸۰]، در نظربگیرید. جدول ۵-۱ لیست مثال کلاس کلمات بدست آمده بوسیله چنین متدی را نشان می‌دهد. در حالیکه جای بیشتر کلمات بظاهر درست به نظر می‌رسد یک تعدادی از این کلمات خارج از محل به نظر می‌رسند.

جدول۱-۹. لیست کلاس کلمات بدست‌آمده

COMMITTEE COMMISSION PANEL SUBCOMMITTEE WONK
THEMSELVES MYSELF YOURSELF UNBECOMING ...
ATTORNEY SURGEON RUKEYSER CONSUL RICKEY ...
ACTION ACTIVITY INTERVENTION ATTACHE WARFARE ...
CENTER ASSOCIATION FACETED INSTITUTE GUILD ...
PARTICULAR YEAR'S NIGHT'S MORNING'S FATEFUL ...

تکرار کلمات شمرده شده در مجموعه برای گرفتن اعتبار ناکافی هستند. بطور کلی قابلیت اعتبار بیشتر یک کلمه می‌تواند به یک کلاس اختصاص دهد کمتر از آنکه برای اعتبار مفید باشد. پس چگونه خوشه‌بندی فرهنگ لغات مفید است ؟

من معتقدم که راه حل این مسئله و مشابه‌های آن تزریق دانش انسانی زبان به فرآیند است که می‌تواند به فرمهای زیر باشد:

- **مدلسازی محاوره‌ای:** بهینه‌سازی داده‌گرا، دانش‌انسانی و تصمیم گیری می‌تواند نقش تکمیلی را در یک فرآیند چرخشی بهم‌پیچیده بازی کند. برای مسئله خوشه‌بندی فرهنگ لغات به این معنی است که انسان برای داوری کردن برخی تصمیمات مرزی و لغو کردن بقیه در چرخه مداخله کند. برای مثال یک انسان می‌تواند تصمیم بگیرد که TUESDAY متعلق به همان کلاس MONDAY , TURSDAY , FRIDAY است حتی اگر به تعداد کافی تکرار نشده باشد یا حتی اصلاً اتفاق نیفتاده باشد.مثال دیگر این متد متدلوژی استنتاج خصیصه محاوره‌ای توصیف شده در [۸۰] است.
- **کد کردن دانش بعنوان احتمال اولیه:** یکی از خطرات استفاده از دانش‌انسانی است که اغلب اغراق و برخی اوقات اشتباه می‌کنند. بنابراین یک راه حل بهتر می‌تواند کد کردن دانش بعنوان احتمال اولیه در یک رویه بروزرسانی بیز باشد. بعد از آموزش، هرچه بقدر کافی در مجموعه آموزش نمایش داده نشده باشد تا بواسطه احتمال اولیه گرفته شود ادامه پیدا می‌کند. هر وقت داده کافی وجود داشته باشد، آنها بر احتمال اولیه برتری پیدا می‌کنند. برای مسئله دسته‌بندی فرهنگ لغات عقاید خبره‌ها در حورد روابط بین ورودی‌های فرهنگ لغات باید بطور مناسب کد شود، و الگوی خوشه‌بندی باید برای بهینه‌سازی معیار احتمال بدست آمده مناسب تغییر پیدا کند. بنابراین در مثال بالا وجود داده کافی ممکن است FRIDAY را جدا کند بخاطر اینکه در عباراتی مثل Thank God It's Friday بکار می‌رود.

دانش‌زبانی کد شده بعنوان احتمال اولیه یک چالش تحریک‌کننده است که هنوز روی آن کارهای جدی انجام می‌شود. این احتمالاً می‌تواند شامل یک معیار فاصله روی کلمات و عبارات، و یک روش تصادفی آنتولوژهای کلمه ساخت‌یافته مثل وردنت[۱۱۰] باشد. همین‌طور در سطح نحوی این می‌تواند شامل روش بیز گرامر لغوی ساخته شده بصورت دستی باشد. در عمل چارچوب‌کاری بیز و روش محاوره‌ای ممکن است ترکیب شود.

۱-۱۸. مدلسازی آماری زبان فارسی

این بخش قسمتی از پژوهش‌های انجام شده در زمینه طرح تحقیقاتی" مدلسازی آماری زبان فارسی "را ارایه می‌دهد[۱۱۱]. در این گزارش پس از ارایه مقدمه و تاریخچه به توصیف چگونگی طراحی و تنظیم مدل‌های آماری برای کلمات و جملات فارسی پرداخته شده است .در هر مسئله مدل-سازی در ابتدا دو عامل اصلی می‌باید ابداع و تدوین شوند - ساختار مناسب و جامع برای مدل و سپس چگونگی تنظیم و محاسبه پارامترهای آن. در این تحقیق بر اساس مطالعات و بررسی‌های همه جانبه و با توجه به سوابق و کاربردهای متعدد، مدل پنهان مارکوف برای برای جملات در نظر گرفته شده‌ا ند. برای تنظیم و محاسبه مدل چند-تایی کلمات و مدل‌های پارامترهای مدل روشهای یادگیری هوشمند ارائه خواهند شد. از آنجا که هر روش یادگیری نیاز به مجموعه‌ای از داده‌های آموزشی دارد لذا روشی کارآمد و جامع برای نمایش متن باید در نظر گرفته شود. در این تحقیق از استاندارد های بین المللی جهت ذخیره و پردازش فایلهای متن که شرح آن در این گزارش آمده است استفاده شده است. سپس به چگونگی بهنجارسازی و ایجاد یک لغت‌نامه جامع پرداخته شده است. برای داده‌های آموزشی طیف وسیعی از متون زبان فارسی در زمینه‌های مختلف ادبیات فارسی در نظر گرفته شده است طوری که مدل یاد گرفته شده مبتنی بر توزیع‌های مختلف و وسیعی از داده¬های آموزشی باشد.

۱-۱۸-۱. مقدمه

یک مدل آماری زبان، احتمال جملات را توصیف می‌کند. یعنی با استفاده از آن می‌توان با توجه به اطلاعات جاری، احتمال کلمات بعد را پیش‌بینی کرد. هرچند که مدل‌های دیگری از جمله مدل‌های توانی یا گرامرهای مستقل از متن برای مدل‌سازی زبان‌های طبیعی پیشنهاد شده‌اند، اما در عمل مدل‌های آماری مدل چند-تایی از سایر روش‌ها عملی‌تر و موثرتر بوده‌اند .زیرا علاوه بر سادگی پیاده-سازی، نسبت به نویز هم مقاوم هستند.

کاربردهای عمده مدل آماری زبان در سیستم‌های اتوماتیک تشخیص گفتار و تشخیص متون ماشینی یا دستنویس است .با توجه به حجم روزافزون اطلاعات و اهمیت پردازش خودکار گفتار و اسناد توسط کامپیوتر، می‌توان به اهمیت مدل‌سازی زبان پی برد. از کاربردهای دیگر مدل زبان می‌توان به این موارد اشاره کرد: استفاده در سیستم های ترجمه متون، پردازش زبانهای طبیعی، پردازش و تصحیح خودکار لغات در پردازنده‌های متن، تبدیل متن به صدا و بهبود نتایج بازیابی اطلاعات متنی در موتورهای جستجو.

در همه این کاربردها، عملکرد سیستم با استفاده از مدل آماری زبان بهتر می‌شود اما در سیستم‌های بازشناسی گفتار وجود مدل زبان حیاتی‌تر است؛ در واقع یک سیستم تشخیص گفتار بدون استفاده از مدل‌های زبان به‌هیچ‌وجه نمی تواند دقت قابل قبولی داشته باشد. با توجه به کاربردهای گسترده مدل-های زبان، در سالهای گذشته تحقیقات بسیاری برای مدل‌سازی زبان‌های پراستفاده جهان و بیش از همه زبان انگلیسی انجام شده است و این تحقیقات همچنان ادامه دارند. جالب توجه است که مدل‌های آماری زبان انگلیسی تقریباً از سال ۱۹۸۰ در سیستم‌های واقعی به کار رفته‌اند اما برای مدل‌سازی زبان فارسی متاسفانه هنوز هیچ تحقیق گسترده و قابل‌اعتنایی صورت نگرفته است. البته با توجه به اینکه در کشور ما روند اتوماسیون عمر زیادی ندارد و عمده کاربردهای مدل آماری زبان درسیستم‌های اتوماتیک پردازش و تشخیص گفتار و متن است، کمبود تحقیق در این زمینه کاملاً قابل انتظار است. هدف اصلی این تحقیق و اولین گزارش آن که در پی می آید، جبران این کاستی و پایه‌ریزی اصول اولیه است. امید است که این پژوهش انگیزه و راهگشای کارهای بزرگتر بعدی باشد.

۱-۱۸-۲. شرح کارهای انجام شده

همانگونه که در شرح آمده است داده‌های آموزشی از طیف وسیعی از متون فارسی در زمینه‌ها و موضوعات مختلف جمع‌آوری شده‌ا ند .سپس روش بهنجارسازی این انبوه داده‌ها و روش استاندارد بین المللی یونیکد برای نمایش متون مورد بحث قرار گرفته است. در قسمت بعدی چگونگی ایجاد یک لغت‌نامه جامع و کامل از دیدگاه آماری پرداخته شده است. مدل‌های آماری مورد نظر در این تحقیق مبتنی بر مدل‌های پنهان مارکوف برای کلمات و مدل‌های مدل چند-تایی برای جملات می‌باشند که پارامترهای آن‌ها بر اساس الگوریتم‌های یادگیری که شرح داده خواهد شد محاسبه و تنظیم خواهند شد. در این گزارش مدل آماری مارکوف به طور کلی و مدل پنهان مارکوف به‌خصوص معرفی و شرح داده شده است.

۱-۱۸-۲-۱. داده‌های آموزشی

همانند هر مدل آماری دیگر در این سیستم نیز به داده‌هایی برای آموزش یا در واقع تنظیم پارامترهای آزاد مدل احتیاج داریم. برای تهیه یک مدل آماری، داده¬های آموزشی باید دارای توزیع احتمالی باشند که بعداً مدل با آنها سروکار دارد ؛ یعنی به طور کلی داده‌های آموزشی باید فضای احتمال را پوشش دهند. یعنی اگر بخواهیم مدل آماری به دست آمده در همه جا قابل استفاده باشد، واضح است که باید انواع مختلف متون - از جمله ادبی، سیاسی، اقتصاد ی،طنز، عقیدتی و. ..را برای آموزش سیستم به کار گیریم.

۱-۱۸-۲-۲. تهیه داده‌های آموزشی

برای تهیه داده‌های آموزشی با چنین حجم زیادی تنها راه عملی استفاده از متون موجود در شبکه اینترنت است. داده‌های آموزشی که در ا ین تحقیق به کار گرفته شده اند، شامل ۱) متون عمومی :

مجلات سروش(که ۹۵ % حجم کل داده‌های آموزشی را به خود احتصاص داده اند) و بعضی از مقالات موجود در سا یت الشیعه. ۲) متون تاریخی :از جمله کلیله و دمنه و ۳) اشعار :از جمله د یوان حافظ و برخی از دفاتر شعر نیما یوشیج و فروغ فرخزاد. حجم کل نوشته‌های آموزشی کـه در نهایـت بـه‌دسـت آوردیم، در حدود ۱۰۰ مگابایت (با فرمت یونیکد) است. شامل حدود ۷۳۰۰۰۰۰ کلمه می باشـد. لازم به توضیح است که مجلات سروش شامل نشریات سروش کودکـان، سـروش نوجـوان، سـروش جـوان، سروش بانوان، سروش اندیشه و هفته‌نامه سروش می‌باشد که این نشریات شامل انـواع مختلـف متـون فارسی هستند. در زیر از هر یک ازاین نشریات بخش‌هایی آورده شده است:

سروش کودکان:
خروس به روی بام ویرانه‌ای پرید و قوقولی قو سر داد. روباه از فرصت استفاده کرد. و به سر خروس پرید و خروس را به دندان گرفت و فرار کرد...
این طوری بود که کورینلیوس تصمیم گرفت کروکودیل‌ها را به حال خودشان بگذارد و از کنار رودخانه دور شود اما هنوز راه زیادی نرفته بود که به یک میمون برخورد کرد...

سروش نوجوان:
اتفاق‌هایی هم که در کارتون‌ها می‌افتد، خیلی وقت‌ها اتفاق‌های واقعی نیستند یا اگر اتفاق‌های واقعی‌اند، نتایجی غیرمعمول دارند؛ مثلاً گربه پایش به چیزی گیر می‌کند و به جای این که زمین بخورد، بدنش کش می‌آید...

سروش جوان:
زیبایی آمریکایی بیشترین جوایز اسکار امسال را برد. چیزی که از قبل می‌شد حدس زد در میا ن برندگان جوایز، نام *کوین اسپیسی* دیده می‌شود...
گزارشی که باعث می شود همان روز عصر تمام آن مزرعه و کشتارگاه قرنطینه و تمام سیصد *و چهل* خوک و سی ودو گاو موجود در آن ها معدوم شوند...

سروش بانوان:
بعد مردم می‌گویند چرا زندگی‌های الان این قدر بی‌دوام است؟ چرا بین زن و شوهرها جنگ و دعواست؟ چرا فلان است، چرا بهمان است؟...

سروش اندیشه:
مکتب فلسفهِ جاودانه (حکمت *خالده*) با دیدگاهی همبسته است که مطابق آن دانشی آغازین، یا دانشی قدسی وجود دارد که بر متافیزیک عام این حقیقت‌نمایی مبتنی است...

هفته نامه سروش:
سفر قندهار، فیلمی درباره مردم افغانستان است...
هرچه ز بیگانه وخیل تواند جمله در این خانه طفیل تواند...
رضا عطاران را حتما میشناسید، بازیگر مجموعه های طنز...

همانطور که در این نمونه‌ها مشاهده می‌شود، مجموعه متون آموزشی شامل اسم‌های خاص ایرانی (رضا عطاران) و غیرایرانی (کوین اسپیسی)، عدد(صدوچهل) و ترکیبات تخصصی (حکمت خالده) نیز هست. بنابراین کاملاً قابل انتظار است که لغت‌نامه فارسی که از این متون استخراج خواهد شد، لغات بسیار زیادی داشته باشد که بسیاری از آنها ممکن است در لغت‌نامه‌های عادی وجود نداشته باشند.

۱-۱۸-۲-۳. تهیه داده‌های آموزشی

فایلهایی متنی که از اینترنت جمع آوری شده، به فرمت HTML هستند. در این فایلها، کاراکترهای فارسی با سیستم‌های کدگذاری متفاوتی(UTF وUnicode 8 ،Cp1256) نشان داده می‌شوند. در اولین مرحله نرمال‌سازی، فایل ورودی را با هر سیستم کدگذاری به فایلی یونیکد تبدیل می‌کنیم. برای این کار یک برنامه جاوا نوشته شده است.

با توجه به مطالبی که درباره استاندارد یونیکد می‌دانیم، می‌توان دریافت که گاهی ممکن است کلمات یکسان با مجموعه ای از نویسه های غیر یکسان نمایش داده شوند. به عنوان مثال کلمه «یک » می‌تواند چهار شکل نمایش مختلف داشته باشد،که از ترکیب دو «ی » (فارسی و عربی) و دو «ک » (فارسی و عربی) به دست می‌آیند. بنابراین، در اولین مرحله نرمال‌سازی، هر حرفی را که در استاندارد یونیکد با دو یا چند نویسه مختلف نمایش داده میشود، فقط با یکی از این نویسه ها نشان می‌دهیم؛ به عنوان مثال هرجا« ک» عربی باشد، به جای آن « ک» فارسی می‌گذاریم. علاوه بر این، حرف« - »(تطویل) هم که برای کنترل طول کلمات استفاده می‌شود و هیچ اطلاعات مفیدی را دربر ندارد برای پردازش‌های بعدی حذف می‌کنیم.

مرحله بعدی، حذف ا عراب کلمات است، با وجود اینکه گاهی تفاوت دو کلمه فقط با ا عراب مشخص می‌شود، اما چون در اکثریت نوشته‌های آموزشی که از اینترنت جمع‌آوری شده‌اند، کلمات اِ عراب‌گذاری نشده‌اند، مجبور هستیم اِعراب کلمات را بطور کلی حذف کنیم . به بیان دیگر، کلمات یا باید همواره دارای اعراب باشند، یا همواره دارای اِعراب نباشند و چون کلمات موجود در نوشته‌های آموزشی همواره دارای اِ عراب نیستند، پس تنها راه برای نرمال‌سازی این است که کلمات هیچ اِعرابی نداشته باشند.

باید توجه داشت که قواعد نرمال‌سازی که در بالا ذکر کردیم، اگرچه بسیاری از موارد را پوشش می‌دهند اما هنوز حالت‌هایی وجود دارند که کلمات یکسان با مجموعه ای از نویسه‌های غیریکسان نمایش داده می‌شوند که عمدتاً به علت غیراستاندارد بودن شیوه نگارش است. به عنوان مثال، بعضی از نویسنده ها ترجیح می‌دهند « ء» آخر کلمات را ننویسند و به این ترتیب مثلا« اشیا » و «اشیاء »از نظر سیستم ما، که یک مجموعه از حروف فارسی پشت سر هم را به عنوان یک کلمه در نظر می‌گیرد، دو کلمه متفاوتند. اما مشکل بزرگتر، سرهم یا جدا نوشتن کلماتی است که از بخشهای مختلفی درست شده اند که این مشکل عمدتا در نگارش افعال استمراری فارسی و جمع بستن کلمات با « ها »پیش می‌آید. به عنوان مثال، با توجه به تعریف کلمه از نظر سیستم ما « می‌روم» شامل دو کلمه « می» و « روم» ، اما «میروم» یک کلمه متفاوت است. و به همین ترتیب« درخت ها» شامل دو کلمه « درخت » و « ها» اما «درختها » یک کلمه متفاوت است. نکته قابل ذکر دیگر این است که سیستم ارائه شده چون رویکردی کاملاً آماری دارد، تلاشی برای ریشه‌یابی لغات انجام نمی‌دهد و بنابراین به عنوان مثال« کتابش» ، « کتابت » و« کتابم» از نظر این سیستم، کلمات متفاوتی هستند. این مساله ناشی از اتصال ضمایر ملکی به انتهای کلمات است .

۱-۱۸-۲-۴. استخراج لغت‌نامه فارسی

پس از نرمال‌سازی مجموعه نوشته‌های (داده‌های) آموزشی، می‌توان لغت‌نامه فارسی را استخراج کرد. بار دیگر یادآوری می‌کنیم که در این مدل، یک دنباله پشت سر هم از نویسه‌ها را به عنوان یک کلمه در نظر می‌گیریم. همانطور که قابل انتظار است، مجموعه نوشته‌های آموزشی شامل نویز هم می‌باشد، مثلا وقتی که یک کلمه اشتباه تایپ شده باشد .گاهی اوقات هم بسته به نوع متن، کلمات

محاوره‌ای در متن وجود دارند؛ مثلا « بتونیم» به جای« بتوانیم».

و همانطور که قبلا ذکر شد، کلمات خاص زبانهای خارجی و اعداد هم کلماتی از مجموعه نوشته‌ها هستند. برای کاهش دادن این کلمات ناخواسته، یک روال ساده را در پیش می‌گیریم: فرکانس تکرار تمامی کلمات موجود در مجموعه نوشته‌ها را محاسبه می‌کنیم و سپس لغاتی را که فرکانس تکراری کمتر از یک حد آستانه از پیش تعیین شده دارند را حذف می‌کنیم. در این تحقیق مقدار آستانه را ۳ قرار دادیم، یعنی در واقع کلماتی را که کمتر از ۳ بارپیش آمده بودند، به عنوان نویز در نظر گرفته و حدف کردیم. با این رویه، یک لغت‌نامه شامل حدود ۴۱۰۰۰ کلمه به دست آمد. این مجموعه بر حسب فرکانس تکرار مرتب شده است، تا اگر برای کاربردی خاص مثلا به هزار لغت پر فرکانس‌تر فارسی احتیاج است بتوان به راحتی از هزار لغت ابتدای این مجموعه استفاده کرد. ده عضو ابتدای این لغتنامه به ترتیب اینها هستند :و، می، به، در، که، از، را، این، است، با .همانطور که قابل انتظار است پر فرکانس ترین لغت فارسی حرف ربط « و » است. بدیهی است که قرار گرفتن کلمه« می »به عنوان دومین کلمه لغت‌نامه به دلیل وجود تعداد تکرارهای کلمه نیست بلکه به دلیل تعداد تکرار زیاد در بخش ابتدایی افعال استمراری مانند می‌روم است که به شکل غیرچسبان نوشته شده است. نکته جالب توجه این است که پرفرکانس‌ترین لغات فارسی (یعنی اعضای نخست لغت‌نامه) حروف ربط هستند و به طور مشابه در زبان انگلیسی پرفرکانس ترین کلمه The است و کلمات پرفرکانس بعدی حروف ربط to ، and، of هستند.

در زیر بعضی از کلمات لغتنامه ۴۱۰۰۰ کلمه ای را مشاهده می کنید

جدول۱-۱۰. نمونه‌ای از کلمات لغت‌نامه فارسی استخراج شده

شانگهای	کورتکس	نوزدهم	اینتراکتیو
صدوچهل	بودیسم	جدیدترین	آفساید
سیدمحمدرضا	انتگرال	یورونیوز	نوددرصد
پدرومادرها	ارتودنسی	پرتابل	بلندبلند
ازتهران	توتیا	پراهمیت	دوقسمتی
گیگز	مغیلان	اسکیزوفرنی	اوپک
کامیونیکیشن	تغار	شیشلول	مستثناست
بتونیم	هزارتو	امپایر	مستظرفه

کاملاً واضح است که علیرغم تلاشی که برای حذف لغات نویز کردیم، هنوز لغات ناخواسته‌ای در لغت‌نامه وجود دارند. البته قابل انتظار است که با افزایش حجم نوشته‌های آموزشی و بالابردن حد آستانه حداقلِ فرکانسِ تکرار، کلمات ناخواسته کاهش یابند .اما نکته مثبت این لغت‌نامه، داشتن لغاتی است که در لغت‌نامه های عادی وجود ندار ند و حتی ممکن است لغات فارسی یا عربی هم نباشند , مثلاً انتگرال و اینترنت , اما به دلیل استفاده زیاد واقعاً بخشی از زبان فارسی جدید شده باشند. اشکالات تایپی رایج و حتی بی‌معنی بودن برخی اعضای لغت‌نامه استخراج شده اهمیت چندانی ندارد؛ زیرا مدل آماری بعداً با متونی شامل همین لغات سروکار خواهد داشت.

۱-۱۸-۳. مدلهای مارکف

چون مدلی که برای کلمات ارائه می دهیم، بر مبنای مدل های مارکوف است، در این بخش مقدمه‌ای بر مدل‌های مارکوف و سپس مدل‌های پنهان مارکوف ارائه می‌دهیم.

خروجی یک فرآیند در جهان واقعی به شکل یک سیگنال پیوسته یا گسسته مشاهده می‌شود. یک مسئله حیاتی در علوم ساختن مدل‌هایی برای این سیگنال‌های واقعی است. مدل‌سازی یک سیگنال مزایای فراوانی به همراه دارد. اولاً، مدل، پایه‌ای برای توصیف تئوری سیگنال فراهم می‌کند که می‌تواند برای پردازش سیگنال استفاده شود تا خروجی خواص مطلوبی داشته باشد. ثانیاً، مدل می‌تواند اطلاعات بسیار مفیدی درباره منبع سیگنال بدهد، بدون اینکه احتیاجی به خود منبع باشد. نهایتاً و از همه مهمتر، مدل‌ها می‌توانند در عمل به خوبی کار کنند و امکان تحقق سیستم‌های عملی مهمی را فراهم می‌آورند.

بسته به نوع و خواص سیگنال، راه‌های مختلفی برای مدل کردن یک سیگنال وجود دارد. به طور کلی، یک سیگنال می‌تواند معین یا نامعین (تصادفی یا آماری) باشد. مدل‌های معین از بعضی خواص شناخته شده سیگنال استفاده می‌کنند و مقادیر پارامترهای مدل را تخمین می‌زنند .در طرف دیگر، در مدل‌های آماری، یک فرآیند تصادفی، سیگنال را توصیف می‌کند. برای کاربردهایی نظیر تشخیص گفتار یا دستخط که با نویز و عدم قطعیت همراه هستند، مدل‌های آماری از کارایی بهتری برخوردارند مدل-های پنهان مارکوف،که همچنین منابع مارکوف یا توابع آماری زنجیره‌های مارکوف نامیده می‌شوند، در

تئوری مخابرات یکی از پرکاربردترین مدل‌های آماری هستند.

دسته مهمی از فرآیندهای تصادفی، فرآیندهای مارکوف است که دارای خواصی است که مطالعه ریاضی آنها را امکان‌پذیر می‌کند. در جهان واقعی، معمولاً مطلوب است که یک دنباله متغیرهای تصادفی وابسته را - که مقدار هر متغیر به عنصر (یا عناصر) قبلی در دنباله بستگی دارد - بررسی کنیم. در یک فرآیند مارکوف مقدار متغیر تصادفی جاری برای پیش بینی مقدار متغیرهای تصادفی آینده کافی است. به بیان دیگر، وقتی عنصر جاری را داشته باشیم، عناصر آینده از عناصر گذشته مستقل شرطی‌اند. فرض کنید$X = (X_1, \ldots, X_T)$ دنباله متغیرهای تصادفی باشد که مقادیری از فضای محدود$S = \{s1,\ldots,sN\}$ میگیرند .خواص مارکوف با دو رابطه زیر بیان می‌شوند:

$$P(X_{t+1} = s_k \mid X_1, X_2, \ldots, X_t) = P(X_{t+1} = s_k \mid X_t) \qquad (۱۶\text{-}۱)$$

$$P(X_{t+1} = s_k \mid X_t) = P(X_2 = s_k \mid X_1) \qquad (۱۷\text{-}۱)$$

خاصیت دوم را تغییر ناپذیری با زمان می‌نامیم .اگر دنباله X هر دو خاصیت مارکوف را داشته باشد، آن را یک زنجیره مارکوف می نامیم.

یک زنجیره مارکوف را می‌توان با بردار تصادفی وضعیت اولیه Π و ماتریس تصادفی انتقالA به طور کامل توصیف کرد:

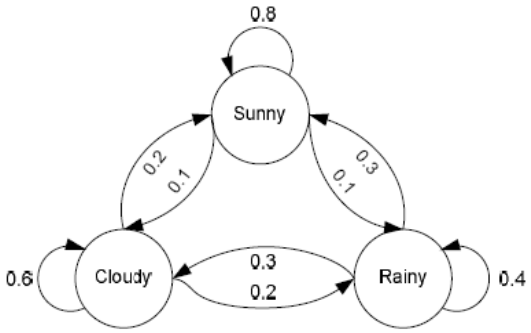
$$\pi_i = P(X_1 = s_i) \qquad (۱۸\text{-}۱)$$

$$a_{ij} = P(X_{t+1} = s_j \mid X_t = s_i) \qquad (۱۹\text{-}۱)$$

که $\pi_i \geq 0, \forall i$ و $\sum_{i=1}^{N} \pi_i = 1$ و $a_{ij} \geq 0, \forall i, j$ و $\sum_{j=1}^{N} a_{ij} = 1, \forall i$.

برای روشن کردن این مطالب، مثالی درباره پیش‌بینی وضع هوا را در نظر بگیرید که در آن با استفاده از تاریخچه مشاهدات وضع هوا در گذشته می‌خواهیم هوای فردا را حدس بزنیم. برای سادگی

فرض می‌کنیم هوا سه حالت بیشتر نداشته باشد: خورشیدی، ابری و بارانی و وضعیت هوا در طول یک روز یکسان است؛ یعنی تغییر حالتی در وسط روز اتفاق نمی‌افتد. اگر فرض کنیم زنجیره وضعیت هوا در روزهای متوالی مارکوف است (که البته در جهان واقعی فرض درستی نیست)، آنگاه ماشین حالت محدود شکل ۵-۱ با احتمالات انتقال (تغییر وضعیت) دلخواهی که روی پیکان ها داده شده است این زنجیره مارکوف را نشان می دهد. توجه کنیدکه مجموع احتمالات پیکانها ی خروجی در هر یک از سه حالت ۱ است .از شکل ۵-۱ واضح است که مدل مارکوف را می توان بعنوان یک ماشین حالت محدود نامعین(تصادفی) به شمار آورد که در آن احتمالات روی پیکانهای تغییر وضعیت قرار گرفته اند.



شکل۱-۴. مدل مارکوف پیش‌بینی هوا

فرض کنید s1 = Sunny، s2 = Cloudyوs3 = Rainy باشد و در اولین روز هوا خورشیدی باشد آنگاه:

$\Pi = (1.0, 0.0, 0.0)$

$$A = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.2 & 0.6 & 0.2 \\ 0.3 & 0.3 & 0.4 \end{bmatrix}$$

محاسبه احتمال یک دنباله از وضعیتها$X_1, \ldots, X_K$ برای یک زنجیره مارکوف به سادگی با رابطه زیر انجام می‌شود:

$$\begin{aligned} P(X_1, \ldots, X_K) &= P(X_1)\, P(X_2 \mid X_1)\, P(X_3 \mid X_1, X_2) \ldots P(X_K \mid X_1, \ldots, X_{K-1}) \\ &= P(X_1)\, P(X_2 \mid X_1)\, P(X_3 \mid X_2) \ldots P(X_K \mid X_{K-1}) \\ &= \pi_{X_1} \prod_{t=1}^{K-1} a_{X_t X_{t+1}} \end{aligned} \qquad (۲۰-۱)$$

بنابراین در مثال بالا احتمال اینکه هوا در هفت روز آینده به ترتیب خورشیدی، خورشیدی، بارانی، بارانی، خورشیدی، ابری و خورشیدی باشد یا در واقع احتمال O = s1, s1, s3, s3, s1, s2, s1 می‌تواند به شکل زیر محاسبه شود:

$$\begin{aligned} P(O \mid \text{Model}) &= \pi_{s1} P(s_1 \mid s_1)\, P(s_1 \mid s_1)\, P(s_3 \mid s_1)\, P(s_3 \mid s_3)\, P(s_1 \mid s_3)\, P(s_2 \mid s_1)\, P(s_1 \mid s_2) \\ &= \pi_1 a_{11}\, a_{11}\, a_{13}\, a_{33}\, a_{31}\, a_{12}\, a_{21} \\ &= 1.0\ (0.8)\ (0.8)\ (0.1)\ (0.4)\ (0.3)\ (0.1)\ (0.2) \\ &= 1.536 \times 10^{-4} \end{aligned} \qquad (۲۱-۱)$$

به طور کلی وقتی از مدلهای مارکوف صحبت می‌کنیم، منظور ما مدلهای مارکوف مرتبه اول است که در آن تاریخچه ای به طول ۱ (یعنی یک عنصر قبلی) برای پیش بینی رفتار آینده استفاده می‌شود. اما گاهی اوقات برای پیش بینی حالتهای آینده تاریخچه بزرگتری لازم است. در یک مدل مارکوف مرتبهn ، برای پیش بینی حالت بعدی، از n حالت قبلی استفاده می شود اما همواره می‌توان با تغییر شکل نمایش فضای حالت، هر مدل مارکوف مرتبه n را به یک مدل مارکوف مرتبه یک تبدیل کرد. بنابراین از نظر تئوری، فرض مارکوف مرتبه اول، محدود کننده نیست.

۱-۱۸-۴. مدلهای مخفی مارکف

مدلهای پنهان مارکوف، یکی از قوی‌ترین ابزارها برای پردازش سیگنال‌ها می‌باشند، انواع مختلف مدل مخفی مارکف علیرغم محدودیت‌هایی که دارند، هنوز پراستفاده‌ترین تکنیک در سیستم‌های مدرن بازشناسی گفتار و تشخیص متون هستند. مدل پنهان مارکوف، کل الگوی ورودی را به عنوان یک بردار ویژگی تکی مدل نمی‌کند، بلکه رابطه بین بخشهای متوالی یک الگو را استخراج میکند، زیرا هر بخش نسبت به کل ورودی کوچکتر و بنابراین مدل‌سازی آن ساده‌تر است.

یک مدل مخفی مارکف را در واقع می‌توان یک ماشین حالت محدود[1] احتمالاتی به حساب آورد که هر حالت با یک تابع تصادفی مرتبط است. فرض می‌شود که در یک دوره گسسته از زمان t ، مدل در یک حالت است و با یک تابع تصادفی از آن حالت خروجی (مشاهده ای) تولید می‌کند. بر مبنای تابع احتمال انتقال حالت جاری، مدل مارکوف در زمان t +1 تغییر حالت می‌دهد. دنباله حالت‌هایی که مدل از آن می‌گذرد معمولا پنهان است، و تنها یک تابع احتمالاتی از آن آشکار است، که مشاهدات تولید شده به وسیله تابع تصادفی مربوط به حالت‌ها است، به همین دلیل در نام‌گذاری این مدلها از صفت "پنهان " استفاده شده است . یک مدل مخفی مارکف را در واقع می‌توان یک فرآیند تصادفی به طور ناتمام(جزئی) مشاهده شده در نظر آورد .یک مدل مخفی مارکف با عناصر زیر توصیف می‌شود :.

تعداد حالت‌های مدل: N

مجموعه حالت‌ها: S = {s1,s2,..., sN}

احتمالات حالت اولیه : ∏={ p i = P(si at t=1)}

احتمالات تغییر حالت: A = {aij = P(sj at t+1 | si at t)}

تعداد علائم قابل مشاهده (تولید شده) : M

مجموعه علائم قابل مشاهده : V = {v1,v2,..., vM}

احتمالات تولید (انتشار)علائم قابل مشاهده : B = {bi(vk) = P(vk at t | si at t}

علامت مشاهده شده در زمان t: Ot

طول دنباله مشاهدات: T

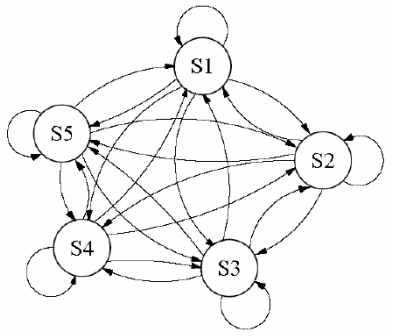
نماد خلاصه برای مدل پنهان مارکوف: λ = (A, B, Π)

واضح است که بر احتمالات Π ، A و B سه قید وجود دارد: $\sum_{i=1}^{n} \pi_i = 1$ ، $\forall i$ ، $\sum_{j=1}^{n} a_{ij} = 1$ و $\sum_{k=1}^{m} b_i(v_k) = 1, \forall i$.

ساختار ماتریس احتمالات تغییر حالت A توپولوژی مدل مخفی مارکف را تعیین می‌کند. اگر $a_{ij} \neq 0 \forall i, j$ یعنی هر حالت مدل از هر حالت دیگر با یک گذر قابل رسیدن باشد، مدل "کاملا متصل" یا

[1] Finite State Machine

"ارگودیک[1]" نامیده می‌شود(شکل ۵-۲). برای مدل‌سازی کلمات زبان که بعداً شرح داده خواهد شد، از مدل ارگودیک استفاده خواهد شد.



شکل۱-۵. یک مدل مارکوف ارگودیک پنج حالته

یک توپولوژی دیگر که در کاربردهای بازشناسی گفتار و متن بسیار پراستفاده است، توپولوژی "چپ به راست" یا "بکیس[2]" می‌باشد که در آن حالتهای با شماره پایین‌تر، مشاهدات نخستین را تولید می‌کنند. ترتیب زمانی در مدل‌های مخفی مارکف چپ به راست با قرار دادن صفرهای ساختاری در مدل به شکل قیدهای $\Pi = \{1, 0,.., 0\}$ $aij = 0, i > j$ اعمال می‌شود، که یعنی مدل از اولین حالت (حالت با کمترین شماره و در واقع سمت چپ ترین حالت) شروع می‌کند و در هر حالت تغییر فقط می‌تواند به حالتهای با شماره بالاتر صورت گیرد. به عنوان یک محدودیت دیگر، بیشتر اوقات در مدل-های مخفی مارکف چپ به راست، اندازه پرش‌های رو به جلو در هر حالت محدود می‌شود و به این ترتیب از تغییر حالت زیاد جلوگیری می‌شود، یعنی برای یک Δ ثابت $aij = 0, j > i + \Delta$ در نظر گرفته می‌شود .

مثال زیر به درک کاربرد مدل‌های پنهان مارکوف کمک می‌کند. فرض کنید یک نفر برای مدتی در اتاقی زندانی است و می‌خواهد از وضعیت هوای بیرون اطلاع داشته باشد. تنها اطلاعاتی که او از دنیای بیرون دارد این است که آیا کسی که هر روز برای او وعده غذایش را می‌آورد با چتر وارد میشود یا نه؛ بنابراین برای مشاهده حمل چتر $V = \{True, False\}$. برای سادگی فرض می‌شود که هوا تنها سه وضعیت دارد :آفتابی، ابری و بارانی، و یک روز معادل یک بازه زمانی است، یعنی وضعیت هوا در طول

[1] Ergodic
[2] Bakis

روز تغییر نمی‌کند. فرض کنید احتمال حمل چتر در یک روز به شرط اینکه هوا آفتابی باشد ۰.۱ باشد، همچنین احتمال حمل چتر به شرط ابری بودن ۰.۳ و احتمال حمل چتر بـه شـرط بـارانی بـودن 0.7 باشد. هدف این است که فرد از مشاهدات(حمل کردن یا نکردن چتر) درباره وضعیت هوای بیرون(که از او پنهان است) نتیجه‌ای بگیرد .فرض کنید w_i وضعیت هوا در روز i باشد، و متغیر بولی u_i به ایـن معنی باشد که در آن روز چتر مشاهده می‌شود یا خیر. با استفاده از قانون بیز :

$$P(w_1, \dots, w_n | u_1, \dots, u_n) = \frac{P(u_1,\dots,u_n | w_1,\dots,w_n) P(w_1,\dots,w_n)}{P(u_1,\dots,u_n)} \quad (۲۲-۱)$$

احتمال$P(w_1,...,w_n)$ معادل مدل مارکوف مثال قبل است، و$P(u_1,...,u_n)$ احتمال از پـیش معلـوم دیدن دنباله‌ای خاص از رخدادهای حمل کردن یا نکردن چتر است. اگر فرض شود که به ازای همه iها، به شرط w_i، u_i از هر w_j و u_j به ازای هر $j \neq i$ مستقل باشد، آنگاه احتمال $P(u_1,...,u_n \mid w_1,...,w_n)$ این گونه محاسبه می‌شود: $\prod_{i=1}^{n} P(u_i|w_i)$.

در مثال پیش بینی وضع هوا (و خیلی از مسـائل دیگـر)مـی‌تـوان از احتمـال پیشـین$P(u_1,...u_n)$ صرفنظر کرد زیرا از وضع هوا مستقل است. بر مبنای فرض مارکوف مرتبه اول، معیار محتمل بودن که با احتمال متناسب است، اینگونه محاسبه میشود :

(۲۳-۱)

$$P(w_1,..., w_n \mid u_1, ..., u_n) \propto$$
$$L(w_1,..., w_n \mid u_1, ..., u_n) = P(u_1,...,u_n \mid w_1,..., w_n)\, P(w_1, ..., w_n)$$
$$= \prod_{i=1}^{n} P(u_i | w_i) \prod_{i=1}^{n} P(w_i | w_{i-1})$$

فرض کنید روز حبس شدن فرد آفتابی بوده باشد، و روز بعد مسئول غـذا بـا چتـر وارد شـده باشـد. مطلوب است پیش بینی وضعیت هوای روز بعد:

در ابتدا با این فرض که روز بعد آفتابی بوده باشد معیار محتمل بودن را حساب می‌کنیم:

$$L(w_2 = Sunny \mid w_1 = Sunny, u_2 = True) = P(u_2 = True \mid w_2 = Sunny) . \quad (۲٤-۱)$$
$$P(w_2 = Sunny \mid w_1 = Sunny) = 0.1\ (0.8) = 0.08$$

سپس با این فرض که روز بعد ابری بوده باشد معیار محتمل بودن را حساب می‌کنیم:

$$L(w_2 = Cloudy \mid w_1 = Sunny, u_2 = True) = P(u_2 = True \mid w_2 = Cloudy) . \quad (۲٥-۱)$$
$$P(w_2 = Cloudy \mid w_1 = Sunny) = 0.3\ (0.1) = 0.03$$

و سرانجام با این فرض که روز بعد بارانی بوده باشد معیار محتمل بودن را حساب می‌کنیم:

$$L(w_2 = Rainy \mid w_1 = Sunny, u_2 = True) = P(u_2 = True \mid w_2 = Rainy) . \quad (۲٦-۱)$$
$$P(w_2 = Rainy \mid w_1 = Sunny) = 0.7\ (0.1) = 0.07$$

بنابراین پر احتمال‌ترین حالت این است که روز بعد آفتابی بوده باشد.

در کاربردهای مدل مخفی مارکف نیازمند حل حداقل یکی از سه مساله زیر هستیم:

مساله ۱. اگر مدل $\lambda = (A, B, \Pi)$ را داشته باشیم، چطور می‌توانیم احتمال $P(O \mid \lambda)$ یعنی احتمال وقوع دنباله مشاهده $O = O1,O2,..., OT$ به شرط مدل را به شکل موثری حساب کنیم؟

مساله ۲. اگر مدل λ و بردار مشاهده O را داشته باشیم، چطور دنباله حالت $S = s1,s2,..., sT$ را انتخاب کنیم که $P(O, S \mid \lambda)$ یعنی احتمال مشترک دنباله مشاهده $O = O1, O2,..., OT$ و دنباله حالت S به شرط مدل، ماکزیمم شود؟ به بیان دیگر، هدف این است که دنباله حالتی پیدا کنیم که مشاهدات را به بهترین نحو توصیف کند.

مساله ۳ .اگر بردار مشاهده O را داشته باشیم، چطور پارامترهای مدل $\lambda = (A, B, \Pi)$ را تنظیم

کنیم که احتمال$P(O \mid \lambda)$ یا$P(O, S \mid \lambda)$ ماکزیمم شود. به بیان دیگر، هدف پیدا کردن (یا در واقع آموزش) مدلی است که داده‌های مشاهده شده را به بهترین نحو توصیف کند.

این سه مسئله با الگوریتم‌های مختلفی - چون روبه عقب و رو به جلو برای مسئله اول الگوریتم برنامه‌سازی پویای ویتربی[1] برای مسئله دوم و الگوریتم باملوچ[2] و الگوریتم کا مینز قطعه‌بندی شده[3] برای مسئله سوم – قابل حل است که توضیح روش‌های آن به علت خارج از حوصله بودن آورده نشده است.

۱-۱۹. مدل پنهان مارکوف برای کلمات فارسی

در استفاده از مدل مخفی مارکف برای مدل‌سازی، یک مساله اساسی تعیین تعداد حالت‌های مدل است؛ دربیشتر اوقات، ارتباط از پیش معلومی بین تعداد حالت‌ها و تعداد علائم قابل مشاهده وجود ندارد. مساله تعیین تعداد حالت‌ها در مدل مخفی مارکف مشابه مسئله تعیین تعداد لایه‌های مخفی و تعداد نرونهای آنها در شبکه‌های عصبی است. چون هنوز تئوری یا روشی اصولی برای انتخاب تعداد حالت‌های مدل مخفی مارکف وجود ندارد، به عنوان یک راه حل ساده، معمولا تعداد حالته‌های مدل را متناسب با تعداد علائم قابل مشاهده، که کاملاً مشخص است، در نظر می‌گیرند. مثلاً در کاربردهای تشخیص متون ماشینی، معمولا پنج (تا ده) حالت به هر حرف (کاراکتر) تخصیص داده می‌شود و بنابراین تعداد حالتهای مدل برابر می‌شود با پنج (تا ده) برابر اندازه مجموعه کاراکترها(علائم قابل مشاهده). اما در مدلی که برای کلمات فار سی پیشنهاد می‌دهیم، هرحالت مدل را با یک حرف الفبای فارسی متناظر می‌کنیم، بنابراین مدل قسمت پنهانی ندارد؛ اینگونه مدلها را" مدلهای آشکار مارکوف" می‌نامند. چون ترتیب زمانی در مدل کلمات اهمیتی ندارد و به طور طبیعی هر حالت مدل (یعنی هر

[1] Viterbi (Daynamic Programming) Algorithm
[2] Boum Welch Algorithm
[3] Segmental K Means Algorithm

حرف) باید با یک انتقال از هر حالت دیگر قابل رسیدن باشد، معقول است که برای این مدل توپولـوژی کاملا متصل (ارگودیک) را برگزینیم.

در مدل‌های آشکار مارکف چون دنباله حالت بهینه از پیش مشخص است، سه مسـئله مـدل‌هـای، مدل مخفی مارکف به شکل خیلی ساده‌تری حل می‌شود. در مدل پیشنهادی برای کلمات، دنباله‌هـای مشاهده (آموزشی) همان کلمات لغت‌نامه استخراج شده هستند، هرچنـد کـه مـی‌تـوان بـرای سـاخت (آموزش) مدل، یا در واقع محاسـبه احتمـالات اولیـه، انتقـال و انتشـار (مشـاهده)، از الگـوریتم‌هـای استاندارد بامولچ یا کا مینز قطعه‌بندی‌کننده استفاده کرد، اما این احتمالات را به سادگی مـی‌تـوان از روابط زیر به دست آورد:

$$\pi_i = \frac{\text{No. Of Words Starting With Letter } s_i}{\text{Lexicon Size}} \qquad (۲۷-۱)$$

$$a_{ij} = \frac{\text{No. Of Transitions From Letter } s_i \text{ to } s_j}{\text{Total Number Of Transitions From Letters } s_i} \qquad (۲۸-۱)$$

$$b_i(v_k) = \begin{cases} 1 & if\ s_i = v_k \\ 0 & otherwise \end{cases} \qquad (۲۹-۱)$$

حالا با داشتن این احتمالات، برای محاسبه $P(O \mid \lambda)$،یعنی حل مسئله ۱، بجای استفاده از الگوریتم رو به جلو یا رو به عقب می‌توان به سادگی از رابطه زیر استفاده کرد:

$$P(O|\lambda) = \pi_{o_1} a_{o_1 o_2} \dots a_{o_{T-1} o_T} \qquad (۳۰-۱)$$

۱-۲۰. نتایج آزمایشی مدل ارائه شده کلمات فارسی

در زیر چند نمونه از احتمالات اولیه و انتقال که با استفاده از فرمولهای (۵-۲۷) و (۵-۲۸) محاسبه شده اند آورده شده است:

$p_{ب} = 3946 \ 41000 \approx 0.096$
$p_{ژ} = 81 / 41000 \approx 0.002$
$p_{م} = 4590 \ 41000 \approx 0.112$
$a_{ب \to ا} = 1795 \ 8029 = 0.224$
$a_{ب \to ب} = 86 \ 8029 = 0.011$
$a_{ب \to ی} = 1035 \ 8029 = 0.129$
$a_{ژ \to خ} = 0.003$
$a_{ف \to ف} = 0.0005$
$a_{ذ \to ظ} = 0.0$

چون مدل پنهان (یا آشکار) مارکوف یک مدل مولد است، با استفاده از مدل ارائه شده برای کلمات می‌توان کلماتی را نیز تولید کرد، که در واقع محتمل‌ترین دنباله‌های مشاهده از دید مدل هستند .در جدول ۵-۳ نمونه هایی از کلمات تولید شده توسط مدل را مشاهده می‌کنید:

جدول۱-۱۱. نمونه‌هایی از کلمات تولید شده بوسیله مدل پیشنهاد شده کلمات فارسی

سه حرفی	چهار حرفی	پنج حرفی	شش حرفی
بکس	**جاشو**	**حدیده**	**رزماند**
رزک	فسور	میرپی	زراندر
حیر	ادهر	سنارا	پرتزاد
ساج	جمین	ستمرد	شیرداد
فرا	کافک	نانده	آیکابر
صیت	صصلو	جهاعا	سترُوجع

به منظور بررسی کیفی کلمات تولید شده توسط مدل، تعدادی از کلماتی را که بدون استفاده ازمدل و به شکل کاملا تصادفی تولید شده اند می آوریم :ضمچ، چضص، بططپ، ذضجژ، دلزنذ، اءژخذ، رصۀمژن و ظمشخجض ملاحظه می‌شود که این کلمات هیچ شباهتی به کلمات زبان فارسی ندارند و هیچ یک به سادگی قابل تلفظ نیستند. اما در مقابل، مدل پیشنهاد شده توانسته است کلمات خوش‌آهنگی را تولید کند و حتی در برخی موارد (سطر اول جدول) موفق به تولید کلماتی شده است که واقعا عضوی از مجموعه کلمات زبان فارسی می‌باشند اما در لغتنامه ۴۱۰۰۰ کلمه‌ای، که برای آموزش مدل نیز به کار رفت، موجود نمی‌باشند.

به عنوان کا ربردی دیگر، با استفاده از مـدل ارائـه شـده لگـاریتم احتمـال تعـدادی کلمـه (دنبالـه مشاهده) درست و نادرست محاسبه می‌شود:

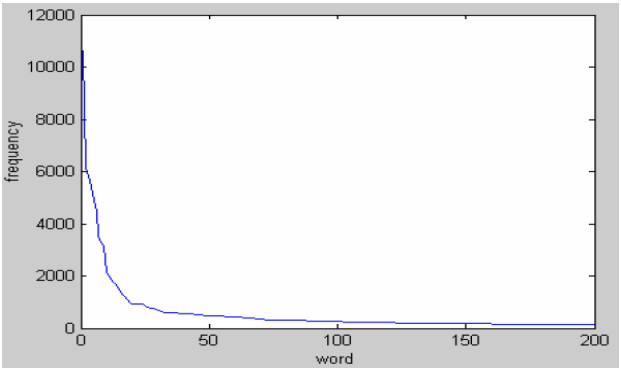
Logprob(تعالی) = -12.9
Logprob(نغالی) = -16.9
Logprob(مهندسی) = -15.4
Logprob(موندشی) = -14.1
Logprob(مهنذسی) = -36.7
Logprob(ستژوجع) = -22.95
Logprob(ظمشخجض) = -44.7

واضح است که معیار لگاریتم احتمال برای یک کلمه هرچه نزدیک‌تر به صفر باشد، یعنی آن کلمه به مدل نزدیکتر است. همانطور که قبلا گفته شد، به دلیل ماهیت تغییرپذیر کلمات زبان، این امکان وجود ندارد که برای تشخیص کلمات نادرست (غیر متعلق به زبان) از یک لغت‌نامه ثابت استفاده کرد ؛ یعنی تشخیص کلمات نادرست برخلاف آنچه که ممکن است در ابتدا به نظر برسد مساله ساده‌ای نیست. اما با توجه به مثالهای بالا به نظر می‌رسد که با استفاده از مدل ارائه شده می‌تـوان حتـی بـدون اسـتفاده از لغت‌نامه، کلما ت بسیار نادرست (بسیار نامحتمل) را تشخیص داد.

۱-۲۱. مدل آماری زبان

همانطور که بخشهای قبل ذکر شد، فرکانس یا احتمال وقوع کلمات یک زبان بسیار متفاوت اسـت . این احتمالات برای زبان های طبیعی از توزیعی به نام زیپف[1] تبعیت می‌کنـد .ایـن توزیـع بـرای ۲۰۰ کلمه اول (پر استفاده‌تر) زبان فارسی با استفاده از نوشته‌های آموزشی ما در نمودار شـکل ۵-۳ نشـان داده شده است.

[1] zipf



شکل۱-۶. نمودار فراوانی ۲۰۰ کلمه اول زبان فارسی

با شمارش دفعات وقوع یک کلمه در یک مجموعه نوشته جات و تقسیم آن بر کل تعداد کلمات می‌توان یک تخمین حداکثر درست‌نمایی برای احتمال وقوع هر کلمه به دست آورد. این ایده را می-توان به زوج کلمات (و ترکیب های طولانی تر) تعمیم داد با این هدف که بعد از دیدن یک یا چند کلمه بتوان کلمه بعدی را حدس زد. پیش بینی یا حدس زدن کلمه بعدی یکی از اعمال ضروری در کاربردهای بازشناسی گفتار، بازشناسی متون(دستنویس یا تایپی) و تشخیص و تصحیح غلط‌های املایی است که در آنها تشخیص کلمه کاری دشوار است زیرا ورودی مبهم و همراه با نویز زیاد است. و بنابراین در نظر گرفتن ورودی های قبلی می‌تواند اطلاعات مهمی درباره گزینه‌های ممکن برای ورودی جاری در اختیار بگذارد. در مدل‌های آماری زبان هدف پیش‌بینی کلمه بعدی یا به بیان دیگر محاسبه احتمال دنباله کلمات (و جملات) است.

روش‌های محاسبه (انتساب) احتمال به یک جمله می‌توانند برای محاسبه احتمال کلمه بعدی در یک جمله ناقص (و بر عکس) به کار روند. دانستن احتمال یک جمله همچنین در کاربردهایی از جمله مشخص کردن نقش کلمات، رفع ابهام معنای کلمات(در ترجمه) و تجزیه احتمالی بسیار موثر است. در مساله تصحیح غلطهای املایی ممکن است که یک یا چند اشتباه تایپی، کلمه مورد نظر را تبدیل به یک کلمه دیگر از زبان (یعنی یک کلمه درست و مجاز) کنند؛ واضح است که در این حالت استفاده از یک لغت‌نامه به تنهایی نمی‌تواند کمکی کند. اما با استفاده از اطلاعات کلمات مجاور و گرامرهای زبان در بیشتر موارد می‌توان غلط را تشخیص داد و حتی تصحیح کرد. به عنوان مثال کلمه« دوستش » که در جمله« کتاب را به درستش داد » به اشتباه «درستش » تایپ شده است، ترکیب بسیار غیرمحتمل «به درستش» را نتیجه داده است که با استفاده از یک گرامر آماری ساده نیز قابل تشخیص است .البته باید توجه کرد که بسیاری از کلمات و جملات کاملاً درست و بامعنی زبان ممکن است

احتمال وقوع پایینی داشته باشند، که این مساله تشخیص خطا را دشوار میکند.

مدل زبان یا به بیان دیگر مدل پیش‌بینی کلمه‌ای که برای فارسی استفاده کرده‌ایم و در اینجا شرح می‌دهیم یک مدل آماری بسیار پر استفاده به نام مدل چند-تایی است که در آن احتمال کلمهN ام با استفاده از N-1 کلمه قبلی تخمین زده می‌شود. موفقیت این مدل‌ها اولین بار در آزمایشگاه های بازشناسی گفتارIBM به اثبات رسید و پس از آن در بسیار از زمینه‌ها مورد توجه و استفاده محققین قرار گرفتند.

۱-۲۱-۲. شمارش کلمات

برای تخمین احتمال نیاز به شمارش ترکیب‌ها داریم. همانگونه که قبلا ذکر شد، پردازش آماری زبان بر مبنای یک مجموعه بزرگ نوشته‌ها است. البته باید توجه کرد که در بعضی کاربردها - از جمله بررسی گرامر، تولید صدا یا شناسایی نویسنده - لازم است تا علائمی همچون'، 'و '.' را نیز به عنوان کلمه در نظر گرفت و در شمارش کلمات و ترکیبات متوالی، آنها را نیز به حساب آورد. مسئله دیگری که باید به آن توجه کرد اشکال مختلف کلمات است. در بیشتر سیستم های مبتنی بر مدل چند-تاییاشکال مختلف یک کلمه به عنوان کلمات مجزا در نظرگرفته می‌شوند؛ یعنی هیچ تلاشی برای ریشه‌یابی انجام نمی‌شود که البته این ساده‌سازی در بعضی کاربردها مناسب نیست.

۱-۲۱-۳. مدل چند-تایی ساده

در این بخش مدل مدل چند-تایی ساده یا هموار نشده (unsmoothed) را شرح می‌دهیم. یادآوری می‌کنیم که مدلهایی که در اینجا برای دنباله کلمات در نظر میگیریم مدلهای احتمالاتی هستند یعنی روش‌هایی برای انتساب احتمال به رشته های کلمات که هم می‌توانند برای محاسبه احتمال یک دنباله کلی استفاده شوند و هم برای پیش‌بینی احتمالاتی کلمه بعدی دریک دنباله.

در ساده‌ترین مدلِ ممکن برای دنباله کلمات، وقوع هر کلمه پس از هر کلمه دیگر مجاز است. به

بیان احتمالاتی، هر کلمه احتمال وقوع یکسانی پس از هر کلمه دلخواه دارد. به عنوان مثال با داشتن لغتنامه ۴۱۰۰۰ کلمه ای، این احتمال برابر است با ۱/۴۱۰۰۰یا تقریباً ۰.۰۰۰۰۲۵ در یک مدل نسبتاً پیچیده‌تر فرض می‌کنیم باز هم هر کلمه می‌تواند پس از هرکلمه دلخواه قرار گیرد اما احتمال وقوع کلمه بعدی(کلمه دوم) برابر است با فراوانی تکرار نرمال (احتمال وقوع آن) . به عنو ان مثال کلمه « این » در نوشته‌های آموزشی ما فراوانی تکرار نرمال ۰.۰۱۸۲ دارد، پس احتمال اینکه« این» بعد از هر کلمه بیاید ۰.۰۱۸۲ است. یعنی در این مدل ساده که unigram نام دارد، کلمات گذشته در پیش‌بینی کلمه بعدی تاثیری ندارند .در واقع با استفاده از فراوانی‌های نسبی می‌توان یک توزیع احتمال روی کلمات بعدی به دست آورد. اما درنظر نگرفتن تاریخچه گذشته در بیشتر موارد نمی‌تواند پیش-بینی های خوبی را نتیجه دهد. به عنوان مثال می‌دانیم که پس از عبارت« خیلی دیر» قرار گرفتن « شده» بسیار معقول تر از« این» است، اما مدل ساده unigram ،« این» را محتمل‌تر از« شده» می‌داند، زیرا احتمال وقوعش به تنهایی بیشتر است. این مثال نشان می‌دهد که احتمال وقوع یک کلمه را باید به شرط کلمات قبل از آن تخمین زد نه به طور مجزا. اگر در مثال ذکر شده حتی تاریخچه‌ای به اندازه یک - یعنی یک کلمه گذشته - را در نظر بگیریم، مدلی به دست می آید که ترکیب« خیلی دیر شده» را به« خیلی دیر این» ترجیح می‌دهد زیرا (دیر |شده)P بیشتر از (دیر |این)P است .

با توجه به مطالب گفته شده، حالا روش محاسبه احتمال یک رشته کلمات (که به شکل $w_1 \ldots w_n$ یا w_1^n نمایش داده می‌شوند) را بررسی می‌کنیم. اگر فرض کنیم هر کلمه در موقعیت درستش یک رویداد مستقل است، می‌توانیم این احتمال را به شکل زیر نشان دهیم:

$$P(w^1, w^2, \ldots, w^{n-1}, w^n) \qquad (۳۱-۱)$$

با استفاده از رابطه زنجیری احتمال این رابطه را میتوان به شکل زیر تجزیه کرد:

$$P(w_1^n) = P(w_1)P(w_2|w_1)P(w_3|w_1^2)\ldots P(w_n|w_1^{n-1}) = \prod_{k=1}^{n} P(w_k|w_1^{k-1}) \qquad (۳۲-۱)$$

حالا سوال این است که چطور احتمالاتی همچون $P(w_n|w_1^{n-1})$ را محاسبه کنیم. به‌نظر می‌رسد

هیچ راه سرراستی برای محاسبه احتمال یک کلمه به شرط دنباله‌ای طولانی از کلمات قبلی وجود نداشته باشد، مگر اینکه یک مجموعه نوشته‌های بسیار بسیار بزرگ در دسترس باشد که عملاً این‌طور نیست .این مسئله را با یک ساده‌سازی مفید حل می‌کنیم. یعنی احتمال یک کلمه را به شرط کلمات قبلی را تقریب می‌زنیم .در ساده ترین تقریب احتمال یک کلمه را به شرط تنها یک کلمه قبلی به دست می‌آوریم. این مدل ساده2 -تایی نام دارد که در آن $P(w_n|w_1^{n-1})$ با $P(w_n|w_{n-1})$ تقریب زده می‌شود.

این فرض که احتمال یک کلمه تنها به کلمه قبلی وابسته است، فرض مارکوف(مرتبه اول) است . همانطور که در مدل کلمات دیدیم، مدل‌های مارکوف دسته‌ای از مدل‌های احتمالاتی هستند که در آنها فرض می‌کنیم که می‌توانیم احتمال رویدادی در آینده را بدون در نظر گرفتن تاریخچه‌ای طولانی از رویدادهای گذشته پیش بینی کنیم. بنابراین مدل2 -تاییپایه را می‌توان به عنوان نوعی زنجیره مارکوف که یک حالت به ازای هر کلمه دارد در نظر گرفت. بدیهی است که می‌توانیم مدل2 -تایی را , که تنها یک کلمه گذشته را در نظر می‌گیرد , به مدلtrigram که دو کلمه گذشته را در نظر می‌گیرد و در حالت کلی به مدل مدل چند-تاییکهN-1 کلمه گذشته را در نظر می‌گیرد تعمیم دهیم. پس مدل 2-تاییک مدل مارکوف مرتبه اول، trigram مدل مارکوف مرتبه دوم و مدل چند-تایی مدل مارکوف مرتبهN- 1 اُم است .در عمل مدلهای2 -تایی (یا حداکثر) trigram استفاده می‌شوند .

معادله کلی برای تقریب مدل چند-تایی - احتمال شرطی کلمه بعدی در یک دنباله - برابر است با :

$$P(w_n|w_1^{n-1}) \approx P(w_n|w_{n-N+1}^{n-1}) \qquad (۳۳-۱)$$

این رابطه نشان میدهد که احتمال کلمه w_n به شرط همه کلمات قبلی می‌تواند با احتمال این کلمه به شرط N کلمه قبلی تقریب زده شود . بنابراین برای یک گرامر۲-تایی می‌توانیم احتمال یک رشته کامل را با جایگزین کردن رابطه ۵-۳۳ در رابطه ۵-۳۲ محاسبه کنیم:

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1}) \qquad (۳۴-۱)$$

البته باید توجه کرد که پیاده‌سازی مستقیم این روابط ممکن است منجر به خطای پاریز[1] شود زیرا هرکدام از احتمال‌ها معمولا عددی بسیار کوچکتر از ۱ است و حاصل‌ضرب تعداد کمی از این اعداد هم از محدوده قابل نمایش در کامپیوتر تجاوز می‌کند. برای حل این مشکل , همانطور که قبلا نیز دیده اید، بجای ضرب احتمالات، لگاریتم آنها را با هم جمع می‌کنیم و در انتها معکوس لگاریتم نتیجه را حساب می‌کنیم. یعنی برای جلوگیری از خطای پاریز، بجای ذخیره و کار با احتمال، با لگاریتم آن کار می‌کنیم.
نکته دیگری که در اینجا لازم به توضیح است محاسبه احتمال یک کلمه در ابتدای جمله است. یعنی وقتی که کلمات گذشته‌ای وجود نداشته باشند. در این حالت می‌توانیم به سادگی از مدلunigram استفاده کنیم اما برای به دست آوردن کارآیی بهتر، فرض می‌کنیم شبه کلمه <s> در ابتدای هر جمله (آموزشی / آزمایشی) وجود دارد که برای محاسبه احتمال ها مانند کلمات عادی شمرده می‌شود. بنابراین احتمال کلمه w در ابتدای یک جمله با مدل2 -تایی برابر است با $P(w \mid <s>)$ و اگر از مدل trigram استفاده کنیم، احتمال کلمه اول جمله برابر است با $P(w \mid <s>, <s>)$ و به همین ترتیب .

آموزش مدلهای مدل چند-تایی از طریق شمارش و نرمال کردن انجام می‌شود. در مدل‌های احتمالاتی نرمال‌کردن به معنای تقسیم به یک مجموع است به طوریکه احتمالات نتیجه در محدوده ۰ و۱ قرار گیرند. به عنوان مثال برای محاسبه (آموزش) یک مدل۲-تایی ساده، با در دست داشتن یک مجموعه نوشته‌ها، تعداد بارهای تکرار هر2 -تایی (زوج کلمه) را می‌شماریم و سپس آن را بر مجموع دفعات تکرار تمام۲-تاییهایی که با کلمه اول شروع می‌شوند تقسیم می‌کنیم .یعنی:

$$P(w_n \mid w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)} \qquad (۳۵-۱)$$

این رابطه را می‌توان ساده کرد، زیرا مجموع همه۲-تایی‌هایی که با کلمه w_{n-1} شروع می‌شوند برابر است با دفعات تکرار این کلمه (یعنی تعدادunigram هایw_{n-1})بنابراین :

[1] underflow

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1}) \quad (۳۶-۱)$$

و برای حالت کلی تخمین مدل چند-تایی از رابطه زیر استفاده می‌شود :

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1}) \quad (۳۷-۱)$$

این رابطه احتمال مدل چند-تایی را با تقسیم کردن فراوانی مشاهده شده یک دنباله خاص برفراوانی مشاهده شده پیش دنباله تخمین میزند. این نسبت فراوانی نسبی نامیده میشود. استفاده از فراوانی نسبی به عنوان روشی برای تخمین احتمال مثالی از کاربرد تکنیک تخمین حداکثر درست-نمایی[1] است؛ زیرا مجموعه پارامترهای نتیجه شده، احتمال مجموعه آموزشی T به شرط مدل M (یعنی P(T | M)) را ماکزیمم می‌کند. البته به جای استفاده از فراوانی‌های نسبی روش‌های بهتری برای تخمین احتمالات مدل چند-تایی وجود دارد، اما آن روشهای پیشرفته نیز به نوعی از ایده فراوانی نسبی استفاده می‌کنند.

مثال:

جدول ۲ فراوانی تکرار ۲-تایی‌های زیر مجموعه‌ای از نوشته‌های آموزشی را نشان می‌دهد که شامل ۱۶۱۶ نوع کلمه و تقریبا ۱۰۰۰۰ جمله است. همانطور که ملاحظه می‌کنید بیشتر مقادیر صفر هستند. در واقع چون تنها هفت کلمه را انتخاب کرده‌ایم، این ماتریس خلوت تر نیز شده است. فراوانی تکرار این هفت کلمه به این شرح است: او :۳۴۳۷، میخواهد ۱۲۱۵، در :۳۲۵۶، مسابقات :۹۳۸، کشتی :۲۱۳، شرکت :۱۵۰۶، کند :۴۵۹.

[1] Maximum Likelihood Estimation

جدول۱-۱۲. فراوانی تکرار ۲-تایی‌های زیر مجموعه‌ای از نوشته‌های آموزشی

	او	میخواهد	در	مسابقات	کشتی	شرکت	کند
او	8	1087	0	13	0	0	0
میخواهد	3	0	786	0	6	8	6
در	3	0	10	860	3	0	12
مسابقات	0	0	2	0	19	2	52
کشتی	2	0	0	0	0	120	1
شرکت	19	0	17	0	0	0	0
کند	4	0	0	0	0	1	0

جدول ۵-۵ احتمالات ۲-تایی‌ها را پس از نرمال‌سازی - یعنی تقسیم هر سطر بـر فراوانـی تکـرار۱-تایی مناسب - نشان می‌دهد .

جدول۱-۱۳. احتمالات ۲-تاییها پس از نرمال‌سازی

	او	میخواهد	در	مسابقات	کشتی	شرکت	کند
او	.0023	.32	0	.0038	0	0	0
میخواهد	.0025	0	.65	0	.0049	.0066	.0049
در	.00092	0	.0031	.26	.00092	0	.0037
مسابقات	0	0	.0021	0	.020	.0021	.055
کشتی	.0094	0	0	0	0	.56	.0047
شرکت	.013	0	.011	0	0	0	0
کند	.0087	0	0	0	0	.0022	0

چند ویژگی مدلهای مدل چند-تایی

از ویژگی های قابل انتظار مدلهای مدل چند-تایی این است که دقت (کارایی) مدل با افزایش مقدار N افزایش می یابد. علیرغم این واقعیت، عملاً در بیشتر کاربردها از مدل‌های 2 -تاییا حداکثر3 -تـایی استفاده می‌شود؛ زیرا مدل‌های مرتبه بالاتر از ۳ برای آموزشِ مناسب، احتیاج به مجموعه نوشـته‌هـای بسیار بزرگی دارند و در غیر این صورت نمی‌توان تخمین‌های مناسبی برای احتمـالات بـه دسـت آورد. علاوه بر این، مدل‌های مرتبه بالاتر از ۳ بسیار بزرگ هستند و ذخیره و استفاده از آنها احتیاج به حافظه زیادی دارد. مرتبه حافظه مورد نیاز یک مدل مدل چند-تایی برابر است با تعداد ترکیبهای مختلف N کلمه ای که دارای حد بالای VN است (V اندازه لغت‌نامه مورد استفاده است).

ویژگی دیگر مدلهای مدل چند-تایی وابستگی زیاد آنها به نوشته جات(به خصـوص نـوع و انـدازه) آموزشی است. یکی از روش‌های معمول برای مشاهده عملکرد کیفی یک مدل مدل چند-تایی تولیـد، رشته‌های تصادفی کلمات با استفاده از مدل است. به این ترتیب که اولین کلمه بـر اسـاس احتمـال1 -

تایی انتخاب می‌شود و سپس دومین کلمه با مدل2 -تایی و پس از آن کلمات بعدی می‌توانند با مدل ۲-تایی یا ۳-تایی تولید شوند .

نظر به اینکه احتمالات در یک مدل آماری همچون مدل چند-تایی از یک مجموعه آموزشی استخراج می‌شوند، این مجموعه را باید با دقت انتخاب کرد .اگر مجموعه آموزشی تنها مربوط به یک زمینه(موضوع) خاص باشد، احتمالات نتیجه شده نمی‌توانند برای جملات جدید به شکلی مناسب تعمیم یابند. از طرفی اگر مجموعه نوشته‌های آموزشی بسیار عمومی و کلی باشد، ممکن است احتمالات نتیجه شده برای آن کاربرد خاص مناسب نباشند. برای آموزش و سپس محاسبه کارآیی یک مدل، همانند سایر مسائل یادگیری محاسباتی، مجموعه نوشته‌های موجود را به دو مجموعه مجزای آموزشی و آزمایشی تقسیم می‌کنیم؛ مدل را بر اساس مجموعه آموزشی می‌سازیم و سپس کارآیی آن را با استفاده از معیاری به نام پیچیدگی[1] - که معیاری مشابه با آنتروپی است - روی مجموعه آزمایشی محاسبه می‌کنیم. البته در برخی موارد به بیش از یک مجموعه آزمایشی احتیاج داریم. مثلا فرض کنید که چند مدل مختلف را در اختیار داریم و هدف این است که بهترین را انتخاب و سپس کارآیی آن را محاسبه کنیم .البته باید توجه کرد که برای مقایسه کارآیی این مدل‌ها لازم است از تست‌های آماری استفاده شود تا معلوم شود تفاوت بین دو مدل تا چه حد قابل توجه است.

هموارسازی

مساله اصلی مدل‌های استاندارد مدل چند-تایی که تا حالا بررسی کردیم، این است که تخمین‌های احتمال آن‌ها با استفاده از یک مجموعه نوشته‌های آموزشی" محدود "به دست می‌آیند که به هر حال نمی‌تواند شامل تمام ترکیبهایمدل چند-تایی(های) مجاز و بامعنی زبان باشد. یعنی مثلا ماتریس 2 - تاییبا هر مجموعه آموزشی که محاسبه شود، ماتریسی خلوت است. یعنی شامل تعداد بسیار زیادی2 - تایی با احتمال صفر است که بخشی از این ۲-تایی واقعا مجاز و بامعنی هستند و لذا باید احتمال غیرصفری داشته باشند. این مشکل را در مثال جدول ۵-۵ ملاحظه می‌کنید، به عنوان نمونه در سطر اول، ترکیب« او شرکت » علیرغم معقول بودن، احتمالی صفر دارد (یعنی از دید مدل غیرممکن تلقی می‌شود).

[1] Perplexity

حتی اگر مشکل فراوانی‌های صفر وجود نداشته باشد، روش تخمین حداکثر درست‌نمایی وقتی مقدار فراوانی‌ها کوچک باشد، باز هم تخمین‌های ضعیفی را نتیجه می‌دهد. برای حل این مشکلات از تکنیک-های هموارسازی استفاده می‌شود که در آنها به احتمالات صفر (و پایین) مقادیر غیرصفری (و بیشتری) منسوب می‌شود و به همین نسبت از احتمالات زیاد کاسته می‌شود (تا جمع احتمالات ۱ باقی بماند) در زیر، دو الگوریتم (ساده و پیشرفته) هموارسازی را به همراه مثال شرح می‌دهیم.

هموارسازیِ جمع با یک

اولین و ساده‌ترین روشی که برای هموارسازی - یا اجتناب از فراوانی‌های صفر - به ذهن می‌رسد، روش« جمع با یک» است و اگرچه که کارآیی خوبی ندارد و در عمل چندان مورد استفاده قرار نمی-گیرد اما مفاهیم کلی و اساسی هموارسازی را در بردارد و بررسی آن به درک و توصیف الگوریتم‌های پیچیده‌تر کمک می‌کند. در ابتدا به منظور سادگی، کاربرد روش جمع با یک را برای هموارسازی احتمالات ۱-تایی شرح می‌دهیم. تخمین حداکثر آنتروپی معمولی ناهموار، با تقسیم کردن دفعات تکرار یک کلمه بر تعداد کل کلمات موجود در مجموعه نوشته‌های آموزشی(که با N نمایش داده می-شود) به دست می‌آید:

$$P(w_x) = \frac{c(w_x)}{\sum_i c(w_i)} = \frac{c(w_x)}{N} \qquad (۳۸-۱)$$

روش‌های مختلف هموارسازی بر مبنای یک فراوانی تصحیح شده *c هستند. تصحیح فراوانی برای روش جمع با یک این‌طور تعریف می‌شود: مقدار فراوانی پس از جمع با یک، در ضریب نرمال‌سازی $\frac{N}{N+V}$ ضرب می‌شود که V اندازه لغت‌نامه (یا تعداد نوع کلمات، یا به عبارتی دیگر تعداد ۱-تایی های منحصر به فرد) است .چون به هر کلمه (نوع کلمه) یکی اضافه می‌کنیم، به اندازه کل کلمات موجود به اندازه V اضافه می‌شود و بنابراین فراوانی تصحیح شده برای روش هموارسازی جمع با یک این‌طور تعریف می‌شود:

$$c_i^* = (c_i + 1)\frac{N}{N+V} \qquad (۳۹-۱)$$

و این فراوانی‌ها با تقسیم (نرمال)کردن بر N تبدیل به احتمال می‌شوند .

در واقع ایده اصلی همه الگوریتم‌های هموارسازی کاستن از فراوانی‌های غیرصفر (و دارای مقدار زیاد) با هدف انتساب فراوانی (و در نتیجه احتمال غیر صفر) به فراوانی‌های صفر است بنابراین، بجای استفاده از فراوانی‌های کاهش یافته*c برخی محققین ترجیح می‌دهند که، الگوریتم‌های هموارسازی را بر حسب یک ضریب کاهش dc تعریف کنند که برابر است با نسبت فراوانی کاهش‌یافته به فراوانی اصلی است: $d_c = \frac{c^*}{c}$

و به عنوان روشی دیگر، می‌توانیم احتمالات *p را مستقیما از فراوانی‌های اصلی حساب کنیم :

$$p_i^* = \frac{C_i+1}{N+V}$$

حال که روش جمع با یک را برای ۱-تایی توضیح دادیم، آنرا به ۲-تایی مثال قبل اعمال می‌کنیم. جدول ۵-۶ فراوانی‌های با یک جمع شده (هموار شده) و جدول ۵-۷ احتمالات مربوطه را نشان می‌دهد.

جدول۱-۱۴. فراوانی های تکرار همواره شده با روش جمع با یک

	او	میخواهد	در	مسابقات	کشتی	شرکت	کند
او	9	1088	1	14	1	1	1
میخواهد	4	1	787	1	7	9	7
در	4	1	11	861	4	1	13
مسابقات	1	1	3	1	20	3	53
کشتی	3	1	1	1	1	121	2
شرکت	20	1	18	1	1	1	1
کند	5	1	1	1	1	2	1

جدول۱-۱۵. احتمالات همواره شده با روش جمع با یک

	او	میخواهد	در	مسابقات	کشتی	شرکت	کند
او	.0018	.22	.00020	.0028	.00020	.00020	.00020
میخواهد	.0014	.00035	.28	.00035	.0025	.0032	.0025
در	.00082	.00021	.0023	.18	.00082	.00021	.0027
مسابقات	.00039	.00039	.0012	.00039	.0078	.0012	.021
کشتی	.0016	.00055	.00055	.00055	.00055	.066	.0011
شرکت	.0064	.00032	.0058	.00032	.00032	.00032	.00032
کند	.0024	.00048	.00048	.00048	.00048	.00096	.00048

یادآوری می‌کنیم که احتمالات igram با نرمال‌سازی هر سطر با فراوانی ۱-تایی محاسبه می-

شود(رابطه ۵-۳۶).برای فراوانی های هموار شده (با روش جمع با یک)لازم است فراوانی هر۱-تایی را به اندازه تعداد کلمات منحصر به فرد (اندازه لغت‌نامه که با V نشان داده می‌شود) افزایش دهیم:

$$P(w_n \mid w_{n-1}) = \frac{C(w_{n-1}w_n)+1}{C(w_{n-1})+V} \qquad (۴۰-۱)$$

چون در این مثال V = 1616 است، فراوانی‌های ۱-تایی (تک کلمه ها) به این شکل افزایش می‌یابند:

او: ۲۸۳۱ = ۱۶۱۶ + ۵۰۵۳ میخواهد :۱۲۱۵ = ۱۶۱۶ + ۳۴۳۷

در :۲۵۵۴ = ۱۶۱۶ + ۴۸۷۲ مسابقات :۹۳۸ = ۱۶۱۶ + ۳۲۵۶

کشتی :۳۱۲۲ = ۱۶۱۶ + ۱۸۲۹ شرکت :۱۵۰۶ = ۱۶۱۶ + ۲۱۳

کند :۲۰۷۵ = ۱۶۱۶ + ۴۵۹

و نتیجه که احتمالات هموار شده است در جدول ۵-۷ نشان داده شد.

برای اینکه ببینیم یک الگوریتم هموارسازی چگونه فراوانی‌های اصلی را تغییر داده است، معمولاً ماتریس فراوانی را بازسازی می‌کنیم. فراوانی‌های تصحیح شده که با رابطه ۵-۳۹ محاسبه می‌شوند برای این مثال در جدول ۵-۸ نشان داده شده اند. توجه کنید که این روش (جمع با یک) فراوانی‌ها را به شدت تغییر داده است. مثلا (میخواهد در)C از ۷۸۶ به ۳۳۳ کاهش یافته است و در فضای احتمال نیز (میخواهد |در)P از ۰.۶۵ (در حالت ناهموار)به ۰.۲۸(در حالت هموار) کم شده است. ضریب کاهش dc نیز نشان می‌دهد که فراوانی برای هر کلمه اول چند برابر شده است .مثلا در اینجا۲-تایی‌هایی که با « کشتی» شروع شده‌اند، تقریبا ۸ برابر کم شده‌اند .

جدول۱-۱۶. فراوانی‌های بازسازی شده با روش جمع با یک

	او	میخواهد	در	مسابقات	کشتی	شرکت	کند
او	6	740	.68	10	.68	.68	.68
میخواهد	2	.42	331	.42	3	4	3
در	3	.69	8	594	3	.69	9
مسابقات	.37	.37	1	.37	7.4	1	20
کشتی	.36	.12	.12	.12	.12	15	.24
شرکت	10	.48	9	.48	.48	.48	.48
کند	1.1	.22	.22	.22	.22	.44	.22

تغییرات شدید مقادیر فراوانی و احتمال در اثر جابجایی زیاد توزیع احتمال به طرف احتمالات صفر (و نزدیک به صفر) می‌باشد .مشکل از اینجا ناشی می‌شود که مقدار ۱ برای جمع با هر فراوانی به طور اختیاری انتخاب شده است. البته میتوان با جمع کردن مقادیر کوچکتر (مثلا نیم یا یک هزارم) تا حدودی از این مشکل اجتناب کرد؛ اما برای انتخاب این عدد هم باید معیاری معقول وجود داشته باشد. به طور کلی هموارسازی جمع با یک روشی ضعیف است و در مقایسه با روشهای پیشرفته‌تر عملکرد بسیار بدتری دارد. همچنین تحقیقات نشان داده‌اند که واریانس‌های فراوانی‌های تولید شده با روشِ جمع با یک واقعا بدتر از چیزی است که با روش تخمین حداکثر درست‌نمایی معمولی (ناهموار) به دست آید .

هموارسازیWitten-Bell

روش هموارسازی یا کاهش Witten-Bell تنها کمی پیچیده‌تر از روش جمع با یک است، اما کارآیی بسیار بهتری دارد و یکی از روش‌های پراستفاده در سیستم‌های بازشناسی گفتار است .از روش‌های پراستفاده دیگر می‌توان از Good-Turing وKatz نام برد. روش Witten-Bell بر پایه یک ایده ساده اما زیرکانه درباره رویدادهای با فراوانی صفر (مشاهده نشده) است :یک رویداد یا مدل چند-تاییبا فراوانی صفر وقتی اتفاق می‌افتد که اولین باری است که آن را مشاهده می‌کنیم؛ پس احتمال دیدن یک مدل چند-تایی با فراوانی صفر می‌تواند با احتمال دیدن آن برای اولین بار مدل شود؛ که این یک مفهوم تکرارشونده در پردازش آماری زبان است. ایده اصلی این است که از فراوانی چیزهایی که تنها یک بار دیده شده اند (رخ داده اند) برای تخمین فراوانی چیزهایی که تا حالا دیده نشده اند (رخ نداده‌اند) استفاده کنیم .این ایده در برخی الگوریتم‌های دیگر هموارسازی و روشهای تشخیص نقش کلمه نیز استفاده شده است. برای محاسبه احتمال دیدن یک مدل چند-تایی برای اولین بار باید دفعاتی را که برای اولین بار در مجموعه نوشته‌ها مشاهده می‌شود حساب کرد؛ و این به سادگی قابل محاسبه است زیرا تعدادمدل چند-تایی‌های اولین بار (جدید) یعنی تعداد مدل چند-تاییهای منحصر به فرد در مجموعه داده‌ها. بنابراین مجموع احتمال همه مدل چند-تاییهای دارای فراوانی صفر را با تعداد انواع مدل چند-تاییها تقسیم بر مجموع تعداد کل و تعداد انواع مدل چند-تاییها تخمین می‌زنیم :

$$\sum_{i:c_i=0} p_i^* = \frac{T}{N+T} \qquad (۴۱-۱)$$

دلیل اینکه با مجموع تعداد کل و تعداد انواع مدل چند-تایی‌ها نرمال می‌کنیم این است که مجموعه نوشته‌ها را می‌توانیم به عنوان یک سری از رویدادها در نظر بگیریم: یک رویداد برای هر کلمه و یک رویداد برای هر نوع جدید. بنابراین رابطه ۵-۴۱ در واقع یک تخمینML برا ی رخدادن یک رویداد جدید است. باید توجه کرد که تعداد انواع مشاهده شدهTبا تعداد کل (یا اندازه لغت‌نامه) V که قبلاً در روش هموارسازی جمع با یک استفاده شد تفاوت دارد .T تعداد انواعی است که تا کنون دیده‌ایم، در حالیکهV تعداد" همه انواع ممکن "است .

رابطه ۵-۴۱ مجموع احتمال مدل چند-تایی‌های دیده نشده است که باید بین هممدل چند-تاییهای دارای فراوانی صفر تقسیم شود. می‌توانیم این تقسیم را به طور مساوی انجام دهیم. فرض کنید Z تعداد مدل چند-تاییهای منحصر به فرد با فراوانی صفر باشد، حالا هر 1 -تاییصفر سهم مساوی از توزیع احتمال می‌گیرد:

$$Z = \sum_{i:c_i=0} 1 \qquad (۴۲-۱)$$

$$p_i^* = \frac{T}{Z(N+T)} \qquad (۴۳-۱)$$

احتمال مجموع مدل چند-تایی‌های صفر، که از رابطه ۵-۴۱ حساب میشود، با کاهش دادن احتمال مدل چند-تاییهای دیده شده تامین میشود :

$$p_i^* = \frac{c_i}{N+T} \quad \text{if } c_i > 0 \qquad (۴۴-۱)$$

به بیان دیگر، فراوانی‌های هموار شده را می‌توانیم مستقیما از رابطه زیر محاسبه کنیم:

$$c_i^* = \begin{cases} \frac{T}{Z}\frac{N}{N+T}, & \text{if } c_i = 0 \\ c_i \frac{N}{N+T}, & \text{if } c_i > 0 \end{cases} \qquad (۴۵-۱)$$

همانطور که ملاحظه می‌کنید، این روش هموارسازی برای ۱-تایی‌ها بسیار شبیه به روش جمع با یک است. اما وقتی معادلات به2 -تاییتعمیم داده می‌شوند، تفاوت بزرگی می‌بینیم که به این دلیل است که در اینجا فراوانی انواع به تاریخچه‌ای وابسته است. برای محاسبه احتمال یک 2 -تایی تا حالا مشاهده نشده$w_{n-1}w_{n-2}$ از احتمال مشاهده2 -تاییجدیدی که باw_{n-1} شروع می‌شود استفاده می‌کنیم. این یعنی تخمینی که برای2 -تاییهای جدید می‌زنیم به یک تاریخچه کلمه وابسته می‌شود. کلماتی که در تعداد کمتری از2 -تاییها اتفاق می‌افتند، تخمین2 -تاییمشاهده نشده کمتری نسبت بر کلمات فراوان‌تر ارائه می‌کنند. این واقعیت را با وابسته کردنT (تعداد انواع۲-تایی‌ها) و N (تعداد کل2 -تاییها) به (w_xکلمه قبلی) نشان می‌دهیم:

(۴۶)

$$\sum_{i:c(w_x w_i)=0} p^*(w_i \mid w_x) = \frac{T(w_x)}{N(w_x)+T(w_x)}$$

دوباره باید این جمع احتمال را بین همه2 -تایی‌های دیده نشده توزیع کنیم. فرض کنید Z تعداد 2-تایی‌های منحصر به فرد با فراوانی صفر باشد، حالا هر2 -تاییصفر سهم مساوی از توزیع احتمال می‌گیرد:

$$Z(w_x) = \sum_{i:c(w_x w_i)=0} 1 \qquad (۴۶-۱)$$

$$p^*(w_i \mid w_{i-1}) = \frac{T(w_{i-1})}{Z(w_{i-1})(N+T(w_{i-1}))} \quad \text{if } c_{w_{i-1}w_i} = 0 \qquad (۴۷-۱)$$

از روش کاهشWitten-Bell می‌توان به شکل دیگری نیز استفاده کرد .در رابطه ۵-۴۶ احتمالات 2-تایی هموار شده را به کلمه قبلی وابسته کردیم. یعنی$T(w_x)$ (تعداد انواع2 -تاییها) و$N(w_x)$ (تعداد

کل2 -تاییها) را بهw_x (کلمه قبلی) وابسته کردیم. بجای این کار، می‌توان هر2 -تاییرا بـه عنـوان یـک رویداد تکی در نظر گرفت بدون توجه به اینکه از دو کلمه تشکیل شده است. آنگاهT تعداد انواع همه۲-تایی‌ها وN تعداد کل2 -تاییهای اتفاق افتاده است. با این کار در واقع بجـای کـاهش دادن بـا احتمـال شرطی$P(wi|w_x)$ داریم از احتمال مشترک$P(w_iw_x)$ اسـتفاده میکنـیم. در اینجـا بـا$P(w_iw_x)$ دقیقـا همانند یک احتمال1 -تاییبرخورد می‌کنیم. البته این روش کاهش کمتر از روش اسـتاندارد (رابطـه ۵-۴۶) به کار رفته است.

مراجع و مآخذ

[1] ColeR.A., J.Mariani, H.Uszkoreit, G.Varile, A.Zaenen, V.Zue, A.Zampolli *Survey of the State of the Art in Human Language Technology* , Cambridge University Press and Giardini(Eds) (1997).

[2] Haoyi Wang and Yang Huang, *Bondec – A Sentence Boundary Detector*, phd thesis of Stanford Engineering Informatics 2001

[3] Aderdenn et al. *regular- expression rules in alembic System*,1995

[4] Stanford University 2001 David D.palmer and Marti A.hearst. *Adaptive sentence Boundary disambiguation*. In proceeding of the fourth ACL Conference on applied Natural Language Processing(13- 15October 1994Stuttgart)Pagr 78-83Morgan Kaufmann

[5] Mikheev,A .2000*Tagging Sentence Boun daries*.In NACL' 2000(Seattle) ACL,pp264 – 271

[6] Reynar,J.C .and A .Ratnaparkhi .1997*A Maximum Entropy Approach to Identifying Sentence Boundaries*. In Processing of the ANLP97,Washington , D.C.

[7] Manning,C.D .and H .Schütze .2002*Foundations of statist ical natural language processing*. The MIT Press, Cambridge/London.

[8] Shannon C.E.1948*A mathematical theory of communication* . Bell System Technical Journal 27:379 – 423,623 – 656

[9] Berger A.1996*A Brief Maxent Tutorial* .http://www - 2.cs.cmu.edu/~aberger/maxent.html

[10] Mikheev,A .1998*Feature Lattices and Maximum Entropy Models*

[11] Mikheev,A .2000*Document Centered Approach to Text Normalization* . In SIGIR'2000(Athens) ACM June 2000pp136 —143

[12] David D.Palmer ,*A Trainable Rule -based Algorithm for Word Segmentation*,The MITRE Corporation 202Burlington Rd .Bedford,MA 01730, USA

[13] Dekai Wu and Pascale Fung.1994*Improving chinese tokenization with linguistic filters on statistical lexical acquisition*. In Proceedings of the Fourth ACL Conference on Applied Natural Language Processing (ANLP94) ,Stuttgart , Germany.

[14] Eric Brill1993a .*A corpus -based approach to language learning*. Ph.D. Dissertation, University of Pennsylvania, Department of Computer and Information Science.

[15] J.R .Quinlan .1986*Induction of decision trees* . Machine Learning, 1(1):81- 106

[16] Eric Brill1994*Some advances in transformationbased part of speech tagging*. In Proceedings of the Twelfth National Conference on Artificial Intelligence,pages 722- 727.

[17] Eric Brill and Philip Resnik.1994*A rule -based approach to prepositional phrase attachment disambiguation*.In Proceedings of the Fifteenth International Conference on Computational Linguistics (COLING-1994).

[18] Eric Brill1993b.*Transformation -based errordriven parsing* . In Proceedings of the Third International Workshop on Parsing Technologies.

[19] Lance Ramshaw and Mitchell Marcus.1995.*Text chunking using transformation-based learning*. In Proceedings of the Third Workshop on Very Large Corpora (WVLC-3),pages 82-94.

[20] Kemal Oflazer and Gokhan Tur.1996.*Combining hand -crafted rules and unsupervised learning in constraint-based morphological disambiguation*. In Proceedings of the Conference on Empirical Methods in Language Processing (EMNLP).

[21] Marc Vilain and David Day.1996. *Finite-state phrase parsing by rule sequences*. In Proceedings of the Sixteenth International Conference on Computational Linguistics (COLING-96).

[22] C .J .Van Rijsbergen .1979 .*Information Retrieval* .Butterworths, London.

[23] Chris Buckley, Amit Singhal, and Mandar Mitra.1996.*Using query zoning and correlation within smart*: Trec 5. In Proceedings of the Fifth Text Retrieval Conference (TREC-5).

[24] John Broglio,Jamie Callan ,and W .Bruce Croft .1996.*Technical issues in building an information retrieval system for chinese*. CIIR Technical Report IR-86, University of Massachusetts,Amherst .

[25] Wanying Jin.1994.*Chinese segmentation disambiguation* .In Proceedings of the Fifteenth Interna. tional Conference on Computational Linguistics (COLING-94),Ja pan.

[26] Jay M.Ponte and W . Bruce Croft.1996.*Useg:A retargetable word segmentation procedure for information retrieval*.In Proceedings of SDAIR 96, Las Vegas, Nevada.

[27] Karine Megerdoomian and Rémi Zajac, *Processing Persian Text:Tokenization in the Shiraz Project*, Memoranda in Computer and Cognitive Science MCCS-00-322, Computing Research Laboratory New Mexico State University Las Cruces, New Mexico April 2000

[28] Peter F. Brown, John Cocke, Stephen A. Della Pietra,Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty,Robert L. Mercer, and Paul S. Roossin. A statisticalapproach to machine translation. *ComputationalLinguistics*, 16(2):79-85,June 1990

[29] Ralf brown and Robert & Frederking. Applying statistical English language modeling to symbolic machine
translation. In *Proceedings of the 6th International Conference on Theoretical and Methodological*
Issues in Machine Translation (TMI'95),pages 221- 239,July 1995

[30] J. Ponte and W. Bruce. Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st international conference on research and development in information retrieval (SIGIR'98)*,pages 275- 281, 1998

[31] Adam Berger and John Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd annual conference on r esearch and development in information retrieval (SIGIR'99)*, pages 222-229,1999

[32] Fred Jelinek.The 1995 language modeling summer workshop at Johns Hopkins University. Closing remarks.

[33] Lalit R. Bahl, Jim K. Baker, Frederick Jelinek, and Robert L. Mercer. Perplexity - a measure of the difficulty of speech recognition tasks. *Program of the 94th Meeting of the Acoustical Society of America J .Acoust .Soc .Am .*, 62S 63,1977Suppl .no 1.

[34] Stanley F. Chen, Douglas Beeferman, and Ronald Rosenfeld. Evaluation metrics for language models. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*,pages 275– 280,1998

[35] Ronald Rosenfeld. A maximum entropy approach to adaptive statistical language modeling. *Computer Speech and Language*,10 187–228,1996Longer version published as “Adaptive Statistical Language Modeling: A Maximum Entropy Approach,” Ph.D. thesis, Computer Science Department, Carnegie Mellon University, TR CMU-CS-94 138,April 1994.

[36] C.E. Shannon. A mathematical theory of communication.*Bell Systems Technical Journal*,27:379 – 423,623–656,1948

[37] C.E. Shannon. Prediction and entropy of printed English*Bell Systems Technical Journal*,30:50 –64,January 1951

[38] T.M. Cover and R.C. King. A convergent gambling estimate of the entropy of English. *IEEE Transactions on Information Theory*,24(4):413–421,1978

[39] E. Brill, R. Florian, C. Henderson, and L. Mangu. Beyond مدل چند-تایی‌s: Can linguistic sophistication improve language modeling? In *Proceedings of the 36th Annual Meeting of the ACL*,1998

[40] Frederick Jelinek. *Statistical Methods for Speech Recognition*. MIT Press,Cambridge ,Massachusetts ,1997

[41] I.J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*,40(3 and 4):237–264,1953

[42] [15] Ian H. Witten and Timothy C. Bell. The zerofrequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*,37(4):1085– 1094,July 1991

[43] Slava M. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*,35(3):400–401,March 1987

[44] Hermann Ney, Ute Essen, and Reinhard Kneser. On structuring probabilistic dependences in stochastic language modeling. *Computer Speech and Language*,8:1 –38,1994

[45] Reinhard Kneser and Hermann Ney. Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume I,pages 181–184,Detroit , Michigan,May 1995

[46] Frederick Jelinek and Robert L. Mercer. Interpolated estimation of Markov source parameters from sparse data. In *Proceedings of the Workshop on Pattern Recognition in Practice*,pages 381–397, Amsterdam,The N etherlands: North-Holland,May 1980.

[47] D. Ron, Y. Singer, and N. Tishby. The power of amnesia. In J. Cowan, G. Tesauro, and J. Alspector, editors,*Advances in Neural Information Processing Systems 6*,pages 176–183,Morgam Kaufmann ,San Mateo ,CA, 1994

[48] [2] I. Guyon and F. Pereira. Design of a linguistic postprocessor using variable memory length Markov models. In *Proceedings of the 3rd ICDAR*, pages 454457,1995

[49] Reinhard Kneser. Statistical language modeling using a variable context length. In *Proceedings of ICSLP*,volume1,pages 49449Philadelphia, October 1996

[50] Thomas Niesler and PhilipWoodland. Variable-length category -مدل چند تایی language models. *Computer Speech and Language*,21:1 –26,1999

[51] Man-Hung Siu and Mari Ostendorf. Variable مدل چند-تایی and extensions for conversational speech language modeling. *IEEE Transactions on Speech and Audio Processing*,8(1): 63–75,2000

[52] Pierre Dupont and Ronald Rosenfeld. Lattice based language models. Technical Report CMU-CS-97 173,Carnegie Mellon University ,Dep artment of ComputerScience,September 1997

[53] Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th Annual Meeting of the ACL*,pages 310318, Santa Cruz,California ,June 1996

[54] Ronald Rosenfeld. The CMU statistical language modeling toolkit and its use in the 1994ARPA CSR evaluation . In *Proceedings of the Spoken Language Systems Technology Workshop*,pages 47509Austin ,Texas, January 1995

[55] Philip Clarkson and Ronald Rosenfeld. Statistical languagemodeling using the CMU-Cambridge toolkit. In*Proceedings of the European Conference on SpeechCommunication and Technology (Eurospeech)*,1997

[56] Andreas Stolcke. SRILM—the SRI language modelingtoolkit. http://www.speech.sri.com/projects/srilm/,1999

[57] Stanley F.Chen . Language model tools (v0.) user'sguide http://www.cs.cmu.edu/ sfc/manuals/h015. ps,December 1998

[58] Patti J. Price. Evaluation of spoken language systems:the atis domain. In *Proceedings of the DARPA Speechand Natural Language Workshop*,June 1990

[59] Wayne H. Ward. The cmu air travel information service:understanding spontaneous speech. In *Proceedingsof the DARPA Speech and Natural Language Workshop*,pages 127129,June 1990

[60] Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza,Jennifer C. Lai, and Robert L. Mercer. Classbased مدل چند-تایی models of natural language. *Computational Linguistics*,18(4):467479,December 1992

[61] Reinhard Kneser and Hermann Ney. Improved clustering techniques for class-based statistical language modeling. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*,1993.

[62] Leo Breiman, Jerome H. Friedman, Richard A. Olshen,and Charles J. Stone. *Classification and RegressionTrees*. Wadsworth & Brooks/Cole Advanced Books & Software,Monterey ,California ,1984

[63] Lalit R. Bahl, Peter F. Brown, Peter V. de Souza, and Robert L. Mercer. A tree-based statistical language model for natural language speech recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*,37:1001–1008, July 1989

[64] Arthur N´adas, David Nahamoo, Michael A. Picheny, and Jeffrey Powell. An iterative "flip- flop" approximation of the most informative split in

the construction of decision trees. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Toronto, Canada, May 1991.

[65] Peter F. Brown, Steven A. Della Pietra, Vincent J.Della Pietra, Robert L. Mercer, and Philip S. Resnik.Language modeling using decision trees. research report,I.B.M. Research, Yorktown Heights,NY , 1991.

[66] M. Marcus, B. Santorini, and M. Marcinkiewicz. Building a large annotated corpus of English: the Penn Treeback. *Computational Linguistics*, 19(2),1993.

[67] James K. Baker. Trainable grammars for speech recognition. In *Proceedings of the Spring Conference of the Acoustical Society of America*, pages 547–550,Boston , MA,June 1979.

[68] Frederick Jelinek, John D. Lafferty, and Robert L. Mercer. Basic methods of probabilistic contextfree grammars. In P. Laface and R. De Mori, editors,*Speech Recognition and Understanding: Recent Advances, Trends, and Applications*,volume75of *F: Computer and Systems Sciences*,pages 345–360 Springer Verlag,1992.

[69] R. Moore, D. Appelt, J. Dowding, J. M. Gawron, and D. Moran. Combining linguistic and statistical knowledge sources in natural -language processing for atis. In *Spoken Language Systems Technology Workshop*, pages 261–264,Austin ,Texas ,February 1995.Morgan Kaufmann Publishers , Inc.

[70] Danny Sleator and Davy Temperley. Parsing English with a link grammar. Technical Report CMU-CS-91- 196,Computer Science Department , Carnegie Mellon University,Pittsburgh ,PA ,October 1991.

[71] John D. Lafferty, Danny Sleator, and Davy Temperley.Grammatical 3-تایی: a probabilistic model of link grammar. In *Proceedings of the AAAI Fall Symposium on Probabilistic Approaches to Natural language*, Cambridge, MA, October 1992.

[72] E.T. Jaynes. Information theory and statistical mechanics.*Physics Reviews*,106:620 –630,1957.

[73] J.N. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. *The Annals of Mathematical Statistics*,43:1470 –1480,1972.

[74] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*,19(4):380–393,April 1997.

[75] S. Della Pietra, V. Della Pietra, R.L. Mercer, and S. Roukos. Adaptive language modeling using minimum discriminant estimation. In *Proceedings of the Speech and Natural Language DARPA Workshop*,February 1992.

[76] Raymond Lau, Ronald Rosenfeld, and Salim Roukos. Trigger-based language models: A maximum entropy approach. In *Proceedings of ICASSP-93*, pages II–45
– II–48,April 1993.

[77] Adam Berger, Stephen Della Pietra, and Vincent Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*,22(1):39–71, 1996.

[78] Stanley F. Chen, Kristie Seymore, and Ronald Rosenfeld. Topic adaptation for language modeling using unnormalized exponential models. In *ICASSP-98*,Seattle ,Washington ,1998.

[79] Stan F. Chen and Ronald Rosenfeld. A survey of smoothing techniques for me models. *IEEE Transactions on Speech and Audio Processing*,8(1): 37–50,2000

[80] Ronald Rosenfeld, Larry Wasserman, Can Cai, and Xiaojin Zhu. Interactive feature induction and logistic regression for whole sentence exponential language models. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Keystone, CO, December 1999

[81] Doug Beeferman, Adam Berger, and John Lafferty. A model of lexical attraction and repulsion. In *Proceedings of the 35th Annual Me eting of the Association for Computational Linguistics*,pages 373–380,Madrid ,Spain , 1997

[82] John D. Lafferty and Bernard Suhm. Cluster expansions and iterative scaling for maximum entropy language models. In K. Hanson and R. Silver, editors, *Maximum Entropy and Bayesian Methods*,pages 195– 202,Kluwer Academic Publishers,1995 .

[83] Jochen Peters and Dietrich Klakow. Compact maximum entropy language models. In *Proceedings of the IEEEWorkshop on Automatic Speech Recognition and Understanding*, Keystone, CO, December 1999

[84] Sanjeev Khudanpur and Jun Wu. A maximum entropy language model integrating مدل چند-تایی s and topic dependencies for conversational speech recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*,Phoenix ,AZ ,1999 .

[85] JunWu and Sanjeev Khudanpur. Combining nonlocal,syntactic and مدل چند-تایی dependencies in language modeling.In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*,Budapest, Hungary,1999

[86] Roland Kuhn. Speech recognition and the frequency of recently used words: A modified markov model for natural language. In *12th International Conference on Computational Linguistics*,pages 348–350,Budapest ,August 1988

[87] Julian Kupiec. Probabilistic models of short and long distance word dependencies in running text. In *Proceedings of the DARPA Workshop on Speech and Natural Language*,pages 290–295, February 1989

[88] Roland Kuhn and Renato De Mori. A cache-based natural language model for speech reproduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-12(6):570–583,1990 .

[89] Roland Kuhn and Renato De Mori. Correction to:A cache-based natural language model for speech reproduction.*IEEE Transactions on Pattern Analysisand Machine Intelligence*, PAMI- 14(6):691–692,June 1992

[90] Fred Jelinek, Salim Roukos Bernard Merialdo, and M. Strauss. A dynamic language model for speech recognition. In *Proceedings of the DARPA Workshop on Speech and Natural Language*,pages 293–295,February 1991.

[91] Reinhard Kneser and Volker Steinbiss. On the dynamic adaptation of stochastic language models. In *Proceedings of the IEEE conference on acoustics, speech and signal processing*,pages 586–589,Minneapolis,MN , 1993,volume II .

[92] Rukmini Iyer and Mari Ostendorf. Modeling long distance dependence in language: Topic mixture vs. dynamic cache models. *IEEE Transactions on Speech and Audio Processing IEEE-SAP*,7:30 –39,1999

[93] Kristie Seymore and Ronald Rosenfeld. Using story topics for language model adaptation. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*,1997 .

[94] Kristie Seymore, Stanley Chen, and Ronald Rosenfeld. Nonlinear interpolation of topic models for language model adaptation. In *Proceedings of ICSLP-98*,1998

[95] J.J. Godfrey, E.C. Holliman, and J. McDaniel.SWITCHBOARD: Telephone speech corpus for research and development. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume I,pages 517-520 ,March 1992

[96] Douglas B. Paul and Janet M. Baker. The design for the Wall Street Journal-based CSR corpus. In *Proceedings of the DARPA Speech and Natural Language Workshop*,pages 357-362, February 1992.

[97] David Graff.The 1996broadcast news speech and language model corpus. In *Proceedings of the DARPA Workshop on Spoken Language technology*,pages 11- 14,1997

[98] Ronald Rosenfeld, Rajeev Agarwal, Bill Byrne, Rukmini Iyer, Mark Liberman, Elizabeth Shriberg, Jack Unverferth, Dimitra Vergyri, and Enrique Vidal. Error analysis and disfluency modeling in the switchboard domain. In *Proceedings of the International Conference on Speech and Language Processing*,1996

[99] http://ufal.mff.cuni.cz/dg-bib2.html.

[100] Glenn Carrol and Eugene Charniak. Two experiments on learning probabilistic dependency grammars from corpora. Technical Report TR 92- 16, Computer Science Department,Brown University , 1992

[101] Ciprian Chelba, David Engle, frederick Jelinek, Victor Jimenaz, Sanjeev Khudanpur, Lidia Mangu, Harry Printz, Eric Ristad, Ronald Rosenfeld, Andreas Stolcke, and Dekai Wu. Structure and performance of a dependency language model. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*,pages 2775-2778, 1997volume5.

[102] Michael Collins. A new statistical parser based on 2-تایی lexical dependencies. In *Proceedings of the 34th annual meeting of the association for Computational Linguistics*,pages 184-191,May 1996

[103] Ciprian Chelba and Fred Jelinek. Recognition performance of a structured language model. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*,pages 1567-1570, 1999 volume4.

[104] Jerome R. Bellegarda. A multi-span language modeling framework for large vocabulary speech Recognition.*IEEE Transactions on Speech and Audio Processing*,6:456 –467,1998

[105] S. Deerwester, S. T. Dumais, G.W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent Semantic analysis. *J. Am. Soc. Inform. Science*, 41:391–407,1990

[106] Jerome R. Bellegarda. Large vocabulary speech recognition with multi-span statistical language models.*IEEE Transactions on Speech and Audio Processing*,8(1): 76-84, 2000

[107] Stanley F. Chen and Ronald Rosenfeld. Efficient sampling and feature selection in whole sentence maximum entropy language models. In *ICASSP-99*, Phoenix,Arizona ,1999

[108] Xiaojin Zhu, Stanley F. Chen, and Ronald Rosenfeld. Linguistic features for whole sentence maximum entropy language models. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*, Budapest,Hungary ,1999 .

[109] Stanley F.Chen .Unpublished work .1998

[110] Christiane FellBaum, editor. *WordNet: An Electronic Lexical Database*. Language,Speech and Communication .MIT Press ,1998

[۱۱۱] "*مدل‌سازی آماری زبان فارسی*"، طرح تحقیقاتی، معاونت پژوهشی دانشگاه شیراز، شهریور ۱۳۸۵.