اولین همایش ملی جویشگر بومی
۲۷ و ۲۸ بهمن ماه سال ۱۳۹۴
پژوهشگاه ارتباطات و فناوری اطلاعات
(مرکز تحقیقات مخابرات ایران)
وزارت ارتباطات و فناوری اطلاعات
ITRC

استفاده از تجزیه وابستگی سریع برای استخراج آزاد اطلاعات

مجید عسگری بیدهندی[۱]، بهروز مینایی بیدگلی[۲]

[۱] دانشجوی دکتری، هوش مصنوعی و رباتیک، دانشگاه علم و صنعت، تهران، ایران
majid.asgari@gmail.com

[۲] دکترای علوم و مهندسی کامپیوتر، دانشکده کامپیوتر، دانشگاه علم و صنعت، تهران، ایران
b_minaei@iust.ac.ir

چکیده

اینترنت حجم عظیمی از اطلاعات با ساختارهای گوناگون را در خود جای‌داده است. مدت‌هاست که دانشمندان روی سامانه‌های استخراج اطلاعات از متون ساختارنیافته کار می‌کنند؛ سامانه‌هایی که به صورت انعطاف‌پذیر و قابل اطمینان، صفحات وب را به ساختارهای مناسبی برای سامانه‌های نرم‌افزاری، مانند پایگاه داده رابطه‌ای یا داده‌های اندیس‌گذاری‌شده، تبدیل کنند. از اواسط دهه گذشته میلادی تلاش‌های بسیاری صورت گرفت تا بتوان بدون دخالت انسانی و با همان ابزارهای در دسترسِ پردازش زبان طبیعی، مجموعه‌ی بزرگی از روابط معنایی را از حجم عظیم اطلاعات موجود در وب به دست آورد. یکی از روش‌های مهم برای دستیابی به این هدف استخراج آزاد اطلاعات است. در حقیقت استخراج آزاد اطلاعات توانایی استخراج دانش از حجم وسیعی از اطلاعات وب است؛ و یا از نگاهی دیگر، عملیات استخراج اعلان‌ها، یا رابطه‌ها، از پیکره‌های متنی انبوه، بدون نیاز به فرهنگ لغات از پیش تعیین‌شده؛ بنابراین مهم‌ترین ویژگی استخراج آزاد اطلاعات وابسته نبودن آن به دامنه‌ی خاصی از اطلاعات و استخراجِ اطلاعاتِ بامعنی است. استخراج آزاد اطلاعات سعی می‌کند پایگاه دانشی را باقابلیت پرس‌وجوی کارا از وب استخراج کند. ویژگی مهم دیگر این عملیات این است که سعی دارد بر گلوگاهِ مشکلِ اکتسابِ دانش، که اغلب هزینه‌بر است، با استخراج یکجای تعداد بزرگی از روابط فائق آید. در این مقاله روشی جدید برای سرعت بخشیدن به استخراج آزاد اطلاعات با نام «تجزیه وابستگی سریع» پیشنهاد شده است. استفاده از تجزیه وابستگی سریع، روی افزایش سرعت عملیات استخراج آزاد اطلاعات تأثیر گذار است و سعی دارد مسأله‌ی تجزیه وابستگی را، در ازای افتِ کیفیتِ تجزیه‌ی وابستگی، به یک مسأله‌ی برچسب‌گذاریِ ترتیبی تبدیل نماید. نتایج نشان داده است که استفاده از این ایده توانسته است در ازای افت کیفیت تقریبا ۱۰ درصدی، سرعت طولانی‌ترین بخش از استخراج روابط را افزایش دهد.

کلمات کلیدی

استخراج اطلاعات، استخراج آزاد اطلاعات، پردازش زبان طبیعی، تجزیه وابستگی، استخراج رابطه

۱- مقدمه

استخراج اطلاعات عملیات استخراج خودکار اطلاعت ساختاریافته از داده‌های موجود در اسناد غیرساخت‌یافته یا نیمه ساخت‌یافته‌ی قابل خواندن توسط ماشین است. در بیشتر موارد، این عملیات روی متون به زبان انسانی با کمک علم پردازش زبان طبیعی انجام می‌شود. استخراج آزاد اطلاعات که این مقاله بر پایه‌ی آن بنا شده است، نوعی از عملیات استخراج اطلاعات است. قبل از به وجود آمدن بحث استخراج آزاد اطلاعات، سامانه‌های استخراج اطلاعات زیادی مطرح‌شده بودند که عمده تمرکز آن‌ها روی استخراج با ناظر یا شبه نظارتی بود. نزدیک‌ترین سامانه‌های استخراج اطلاعات به سامانه‌هایی که بعداً سامانه‌های استخراج آزاد اطلاعات نامیده شدند، سامانه‌هایی بودند که سعی داشتند اطلاعات را به روش بدون ناظر از صفحات وب بدون ساختار خاص استخراج کنند. این سامانه‌ها سعی می‌کردند اطلاعات را بدون هیچ کمک انسانی استخراج نمایند.

به طورکلی در فرآیند استخراج آزاد اطلاعات می‌خواهیم حجم وسیعی از رابطه‌ها را با سرعت معقول و منطقی از متونی با دامنه باز به‌صورت قابل‌اعتماد و بدون نظارت انسان استخراج‌کنیم. این تعریف روی چند عبارت تأکید می‌کند: حجم وسیع، سرعت بالا، دامنه‌ی نامشخص و عدم نظارت انسانی. نبود هرکدام از این ویژگی‌ها تعریف مسئله را ساقط می‌کند.

- حجم وسیع. در اینجا می‌خواهیم به‌نوعی پایین بودن دقت را با افزایش حجم استخراج‌ها جبران کنیم. داده‌های زیادی استخراج می‌شوند با این فرض که آن‌هایی که بیشتر تکرار شده‌اند احتمالاً درست‌ترند. جمله بالا، یک جمله احتمالاتی است و ممکن است یک اعلان تکرار زیادی داشته باشد اما به‌طورکلی غلط باشد.
- سرعت بالا. در استخراج آزاد اطلاعات به دنبال این هستیم که با بیشترین سرعت متون انبوه را پردازش کنیم. چون سروکار ما با حجم زیادی از متون است، واضح است که باید میزان پردازش را تا حد ممکن کاهش دهیم. البته در این حوزه روش‌هایی ارائه‌شده‌اند که با سرعت پایین و دقت بالاتر روی متون کار می‌کنند اما مقبولیت نیافته‌اند.

- دامنه نامشخص. یکی دیگر از ویژگی‌های استخراج آزاد اطلاعات، باز و بدون دامنه بودن آن است؛ یعنی روش‌های پیشنهادی باید بتوانند روی متونی با دامنه نامشخص خوب عمل کنند تا حجم وسیعی از متون را پوشش دهند.
- عدم نظارت انسانی. ویژگی دیگری از استخراج آزاد اطلاعات که در حقیقت زاییده‌ی ویژگی اول است، عدم نظارت انسانی است. متون بسیار زیادی قرار است در دامنه‌های مختلف و نامعین پردازش شود. هزینه نظارت نیروی انسانی روی این داده‌ها هزینه‌ی بسیار بالایی است. همچنین درروش‌های استخراج نیز اصولاً از روش‌های نیمه نظارتی یا بی ناظر یا اکتشافی بهره می‌بریم؛ زیرا هزینه ساخت سامانه‌های باناظر روی دامنه‌ی وسیعی از متون بسیار سخت و هزینه‌بر است.

همانطور که اشاره شد در بیشتر روش‌های استخراج آزاد اطلاعات از روش‌های بدون ناظر استفاده می‌شود. در بخش بعدی با مرور کارهای دیگران بر روی استخراج آزاد اطلاعات، خواهیم دید که یکی از مهمترین ابزارهایی که برای این کار به کمک محققان می‌آید استفاده از تجزیه‌ی وابستگی است. زیرا با استفاده از ساختار تجزیه‌ی وابستگی یک جمله می‌توان روابط موجود در آن را بهتر شناسایی کرد. در بخش سوم روش کار تجزیه‌ی وابستگی سریع به عنوان روشی نوآورانه معادل تجزیه‌ی وابستگی مخصوص کار در استخراج آزاد اطلاعات را شرح می‌دهیم. در بخش چهارم روش مزبور را با آزمایشاتی ارزیابی کرده‌ایم و در بخش پنجم نیز نتیجه‌گیری و کارهای آینده ارائه خواهد شد.

۲- کارهای مرتبط

در سال ۲۰۰۷ برای اولین بار مقاله [1] تصویری از استخراج آزاد اطلاعات را ارائه داد: پارادایم استخراج جدیدی که در آن «سامانه‌ای مبتنی بر یک گذار روی داده» معرفی شد که مجموعه‌ی بزرگی از تاپل‌ها (سه‌تایی‌ها) را بدون نیاز به هیچ ورودی انسانی استخراج می‌کند. مقاله، نام TextRunner را روی سامانه مزبور گذاشت و ادعا کرد که یک پیاده‌سازی کامل و بسیار مقیاس‌پذیر از استخراج آزاد اطلاعات را ارائه داده است که در آن یک احتمال به تاپل‌ها تخصیص می‌یابد و برای پشتیبانی از یک استخراج کارا اندیس گذاری می‌شوند و توسط پرس‌وجوهای کاربر اکتشاف می‌گردند.

نویسندگان مقاله [2] یک تحلیل معنایی صریح را پیشنهاد کرده‌اند. یک روش جدید که معنای متون را در یک فضای بعد-بالا از مفاهیم استخراج‌شده از ویکی‌پدیا نمایش می‌دهد. به طور کلی استفاده از ویکی‌پدیا و مفاهیم آن در استخراج آزاد اطلاعات مفید است. آن‌ها از یادگیری ماشین برای نمایش صریح معنایی هر متن به‌صورت یک بردار وزن دهی شده استفاده کرده‌اند.

مقاله [3] نیز در سال ۲۰۰۷ روی استنتاج مترادفی تمرکز کرده است. عملیات تشخیص روابط و اشیاء هم‌معنی یا استنتاج مترادفی عملیاتی حیاتی برای استخراج اطلاعات باکیفیت بالاست. حجم وسیعی از کارهای قبلی درزمینه‌ی استنتاج مترادفی فرض کرده بودند دانش‌های دامنه‌ای قدرتمند یا نمونه‌های آموزشی برچسب خورده توسط انسان در دسترس هستند. این مقاله استنتاج مترادفی را از دید استخراج اطلاعات بدون ناظر بررسی می‌کند.

در [4] یک روش بدون ناظر برای استخراج شبکه‌های معنایی از حجم‌های وسیعی از متن ارائه‌شده است. برای استخراج تاپل‌ها از متن از سامانه TextRunner استفاده‌شده است. سپس مفاهیم و رابطه‌های کلی از آن‌ها استخراج‌شده است. این کار به‌وسیله خوشه‌بندی اتصال رشته‌های اشیاء و روابط تاپل‌ها صورت گرفته است.

در [5] ادعاشده است که برخلاف روش‌های موجود تا سال ۲۰۱۰ که تنها از ویژگی‌های نحوی کم‌عمق استفاده‌شده است روشی ارائه کرده که از ویژگی‌های معنایی (نقش‌های معنایی) برای عملیات استخراج آزاد اطلاعات استفاده‌شده است.

[6] سامانه WOE را معرفی می‌کند که یک سامانه استخراج آزاد اطلاعات دیگر است و دقت و بازخوانی TextRunner را به طرز شگفت‌آوری افزایش می‌دهد. ویژگی‌های کلیدی WOE در کارایی از ایده تازه‌ی یادگیری خود-ناظر برای استخراج‌گرها بهره برده است. یادگیری خود-ناظر با استفاده از نوعی از تطابق‌های اکتشافی بین جعبه اطلاعات ویکی‌پدیا و جملات متناظر برای ساخت داده‌های آموزشی عمل می‌کند سامانه WOE با استفاده از ویکی‌پدیا به‌عنوان یک منبع از داده‌های آموزشی برای استخراج‌هایشان موفق به بهبودهای بیشتری روی TextRunner شدند. آن‌ها همچنین نشان دادند که ویژگی‌های تجزیه وابستگی منجر به افزایش چشمگیری در دقت و بازخوانی نسبت به استفاده از ویژگی‌های زبان‌شناسی کم‌عمق و سطحی می‌شوند، اما در ازای سرعت استخراج پایین‌تر.

در [7] سامانه ReVerb معرفی شد که سعی در تشخیص گروه عبارت ربطی بر اساس محدودیت‌های لغوی و نحوی داشت. یک تحلیل کامل زبان‌شناسی روی تعدادی از جملات نمونه‌گیری شده صورت گرفت. برای مثال مشخص شد آرگومان‌های اول و دوم یک رابطه دارای برچسب‌های ادات سخن محدودی هستند. در نتیجه این تحلیل سامانه‌ی R2A2 معرفی شد که یک سامانه‌ی تشخیص دهنده آرگومان است؛ و به طور کلی سعی شد با دو مشکل اساسی سامانه‌های قبلی مقابله شود: استخراج‌های متناقض و بی‌ربط و استخراج‌های فاقد اطلاعات .

[8] ادعا می‌کند که سامانه‌های استخراج آزاد اطلاعات قبل از او مانند ReVerb و WOE دو ضعف عمده داشته‌اند: اولاً آن‌ها فقط روابطی را استخراج می‌کنند که با میانجیگری فعل درست‌شده‌اند. ثانیاً آن‌ها زمینه متن را نادیده می‌گیرد؛ بنابراین تاپل‌های استخراج‌شده همیشه واقعی و درست نیستند.

این مقاله سامانه OLLIE را به‌عنوان سامانه بهبودیافته از استخراج آزاد اطلاعات که هردوی این مشکلات را حل کرده است پیشنهاد می‌کند. اولاً OLLIE نتیجه بالایی را با استخراج روابط میانجی‌گری شده توسط اسامی و قیدها و ... به دست می‌آورد. ثانیاً با یک گام تحلیل زمینه دقت را با شامل کردن اطلاعات زمینه‌ای ازجمله در استخراج‌ها افزایش می‌دهد. در مقاله مزبور این سامانه با سامانه WOE در حالت تجزیه وابستگی نیز مقایسه شده است.

در [9] مسئله‌ی پاسخ‌گویی به سؤالات با دامنه باز روی ترکیبی از پایگاه دانش‌های دقیق (فراهم‌شده به روش‌های نظارتی) و استخراج‌شده را موردتوجه قرار گرفته است. اینجا اولین جایی است که یک سامانه پاسخ‌گویی به سؤالات ارائه می‌شود که هم بر پایه‌ی پایگاه دانش‌های دقیق (مثل Freebase) و هم پایگاه دانش استخراج‌شده توسط استخراج آزاد اطلاعات بناشده است. مهم‌ترین چالش فنی در این مقاله این است که طراحی سامانه معرفی‌شده باید نسبت به تنوع اطلاعات موجود هم در سامانه‌های پاسخگویی به پرسش و هم پایگاه دانش‌های بزرگ استخراج‌شده قابل‌اطمینان باشد.

در [10] راه حلی برای مسأله‌ی ساخت یک یادگیرنده بی‌پایان زبان ارائه شده است که یک عامل رایانه‌ای هوشمند است که برای همیشه اجرا می‌شود و هر روز باید (۱) برای ساخت پایگاه دانشی در حال رشد، اطلاعات را از وب استخراج کند، یا بخواند، و (۲) این یادگیری را بهتر از روز قبل انجام دهد. به طور خاص، در NELL یک رویکرد و مجموعه ای از اصول طراحی برای چنین عاملی پیشنهاد شده است، و بخشی از پیاده‌سازی چنین سیستمی توصیف شده است که سیستم مزبور در ابتدای معرفی‌اش برای استخراج یک پایگاه دانش حاوی بیش از ۲۴۲۰۰۰ اعلان‌ها با دقت ۷۴٪ پس از ۶۷ روز اجرا آموزش داده شده، و در نهایت در مورد درسهایی از این تلاش اولیه برای ساخت یک عامل یادگیری بی پایان بحث‌هایی ارائه شده است.

همان‌طور که اشاره شد، قدرت سامانه‌های استخراج آزاد اطلاعات، پردازش کارایشان به همراه توانایی آن‌ها در استخراج تعداد بی‌شماری از روابط است. سامانه‌های استخراج اطلاعات زیادی قبلا در همین سند معرفی‌شده‌اند ازجمله TextRunner و WOE که تمام آن‌ها از روشی شامل سه مرحله استفاده می‌کنند [7]:

- برچسب زدن: جملات به‌صورت خودکار برچسب می‌خورند. چه به‌صورت روش‌های اکتشافی یا نظارت راه دور.
- یادگیری: یک استخراج‌گرِ عباراتِ ربطی آموزش داده می‌شود. اغلب با استفاده از یک مدل گرافیکی برچسب‌گذاری توالی مانند میدان‌های تصادفی شرطی.
- استخراج: سامانه یک جمله را به‌عنوان ورودی می‌گیرد، یک جفت کاندید از آرگومان‌های NP (Arg1, Arg2) ازجمله تشخیص می‌دهد و سپس از استخراج‌گر آموزش داده‌شده استفاده می‌کند تا بتواند هر کلمه بین دو آرگومان را به‌عنوان بخشی از رابطه یا خارج از رابطه برچسب بزند. استخراج‌گر به جملات پی‌درپی در پیکره متنی اعمال می‌شود و استخراج‌های نتیجه جمع‌آوری می‌شوند.

اولین مرحله، مرحله برچسب‌زنی است که با توجه به ماهیت بدون ناظر بودن عملیات استخراج آزاد اطلاعات باید به صورت خودکار صورت بگیرد.

برای مثال در [۵] این گام شامل دو مرحله است. مرحله اول: به‌صورت خودکار داده آموزشی خودش را به‌صورت مثبت و منفی برچسب‌گذاری می‌کند. و مرحله دوم: از این داده‌های برچسب‌گذاری شده برای یادگیری یک رده‌بند Naïve Bayes استفاده می‌کند که بعدها توسط ماژول استخراج‌گر مورداستفاده قرار می‌گیرد.

به دلیل اینکه استفاده از یک تجزیه‌گر وابستگی به دلیل سرعت پایین برای استخراج رابطه‌ها در مقیاس وب عملیاتی نیست[۵]، فرض شده که یک تجزیه‌گر می‌تواند در یادگرفتن استخراج‌گر کمک کند. از تجزیه‌گر [۷] برای شناسایی و برچسب‌گذاری یک مجموعه از استخراج‌های قابل‌اعتماد و غیرقابل‌اعتماد استفاده‌شده.

اعلان‌های استخراج شده، تاپل‌هایی به‌صورت زیر هستند:

$$t = (e_i, r_{i,j}, e_j)$$

که در آن eها رشته‌های موجودیت و r رشته‌ای نماینده رابطه بین دو موجودیت است.

تجزیه‌گر چند هزار جمله را پارس می‌کند. برای هر جمله پارس شده سامانه تمام e_iها که جز اصلی سازنده همه عبارات اسمی پایه هستند را پیدا می‌کند. برای هر جفت عبارات اسمی (e_i, e_j) که $i < j$ سامانه ساختار پارس را می پیماید تا یک توالی از کلمات را بیابد که می‌توانند یک رابطه بالقوه به‌صورت r_i در تاپل t باشند.

یادگیرنده تاپل t را مثبت ارزیابی می‌کند اگر محدودیت‌های خاصی را روی ساختار نحوی مشترک بین e_iو e_j ارضا کند. اگر هرکدام از محدودیت‌ها ارضا نشود t به نوان نمونه متنی تلقی می‌شود.

تعدادی از این محدودیتهای اکتشافی عبارت‌اند از:

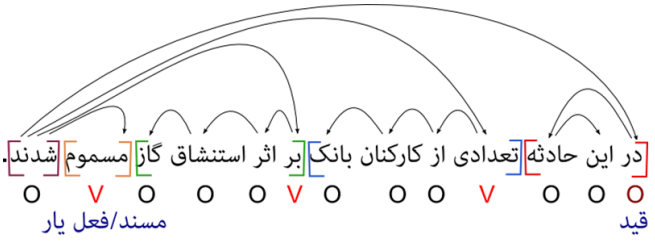
- یک حلقه وابستگی بین e_i و e_j مو د باشد که طولانی تر از یک اندازه خاص نباشد.
- مسیر بین e_i تا e_j در درخت نحوی از یک محدوده‌ی شبه جمله عبور نمی‌کند. مثل عبارات ربط دهنده (آنگاه، اگرچه و ...).
- نه e_i و نه e_j فقط شامل یک ضمیر نباشند.

وقتی یادگیرنده یک مجموعه از تاپل‌ها $t = (e_i, r_{i,j}, e_j)$ را پیداکرده و برچسب زده هر تاپل را به یک بردار ویژگی نگاشت می‌کند. همه ویژگی‌ها مستقل از دامنه و قابل‌سنجش درزمان استخراج بدون نیاز به تجزیه‌گر هستند. خروجی رده‌بند که توسط ماژول یادگیرنده تولید می‌شود منحصر به زبان است اما شامل ویژگی‌های منحصر به رابطه یا ویژگی‌های لغوی نمی‌شود. بنابراین می‌تواند به‌صورت مستقل از دامنه مورداستفاده قرار بگیرد. بر اساس تجربه نویسندگان مقاله روش‌های مختلفی برای انتخاب رابطه آزموده شده است. در یک رویکرد افراط‌گرایانه، تمام کلمات بین دو موجودیت آزموده شدند و نتیجه این بوده که پردازش‌ها طولانی شده و نتایج هم چندان جالب نبوده است. در یک رویکرد تفریط‌گرایانه فقط فعل‌ها در نظر گرفته شدند.

در این مقاله روشی را ارائه خواهیم داد که سعی دارد با کمک برچسب‌های تجزیه وابستگی سه عملیات بالا را با سرعت مناسب انجام دهد.

۳- روش کار

تجزیه وابستگی سریع را به این صورت تعریف می‌کنیم؛ روی تعدادی زیادی جمله، یک الگوریتم تجزیه وابستگی را اجرا می‌کنیم. البته برای دقیق‌تر شدن کار می‌توان از همان جملاتی که تجزیه وابستگی طلایی روی آن‌ها موجود است استفاده کنیم. حال به‌جای برچسب ادات سخن هر کلمه (وابسته)، برچسب ادات سخن کلمه‌ی هسته (سر یا Head) آن را در صورت وجود قرار می‌دهیم. این برچسب‌ها را برچسب‌های تجزیه وابستگی سریع می‌نامیم (به کلمه‌ی برچسب توجه کنید! مسأله تجزیه وابستگی به یک مسأله برچسب‌زنی تبدیل شده است). سپس از یک روش یادگیری توالی استفاده می‌کنیم تا مدلی را از روی این جملات برچسب خورده آموزش دهد. این مدل می‌تواند برچسب‌های هسته را روی جملات با دقت احتمالاً متوسط قرار دهد و از همین برچسب‌ها به‌عنوان یک ویژگی جدید استفاده کند. این ویژگی به همراه ویژگی‌های دیگر روی جمله‌های آموزشی تولیدشده در مرحله قبل اعمال‌شده و مدل دیگری را برای مشخص کردن برچسب‌های B-Arg1 و ... و I-Arg2 آموزش دهد.



(تعدادی از کارکنان بانک، مسموم شدند، بر اثر استنشاق گاز)

شکل ۱: مثالی از تجزیه وابستگی سریع

شکل ۱-۱ مثالی از تجزیه وابستگی سریع را عرضه می‌کند. درخت تجزیه‌ی وابستگی جمله «در این حادثه تعدادی از کارکنان بانک بر اثر استنشاق گاز مسموم شدند.» در این شکل نشان داده شده است. در صورتی که کلمه‌ی هسته‌ی یک هر کلمه‌ی وابسته از نوع فعل باشد، در زیر آن کلمه حرف V نوشته شده است و در غیر این صورت حرف O. تنها یک قانون ساده روی این برچسب‌ها تغییر ایجاد کرده‌اند:

در صورتی که کلمه‌ی اصلی قید بوده و کلمه‌ی هسته از نوع فعل، برچسب به O تغییر کرده است. این قانون موجب می‌شوند قیدها از رابطه در آینده حذف شوند. ممکن است در شرایطی در آینده بخواهیم قیدها را نیز به رابطه‌های فعلی اضافه کنیم که در این صورت این قانون هم کنار گذاشته می‌شود. مشاهده می‌شود که در اغلب موارد هم تنها برچسب V برای ما مهم هستند چون احتمالا دو سر یک رابطه را تشکیل می‌دهند. به طور قراردادی مسندها یا فعل‌یارها را نیز جزئی از رابطه فعلی تصور کرده و آن‌ها را به همراه فعل در هسته‌ی رابطه قرار می‌دهیم.

در نهایت برای تشکیل رابطه الگوریتم زیر را به کار می‌بریم:

- فعل را به همراه مسند یا فعل‌یار (در صورت وجود) به عنوان رابطه در نظر می‌گیریم.
- کلماتی را که دارای برچسب V هستند به عنوان هسته‌های آرگومان‌های رابطه در نظر می‌گیریم.
- با استفاده از یک تجزیه‌گر کم عمق تمام وابسته‌های آرگومان‌ها را به آرگومان‌ها وارد می‌کنیم.

طبق این الگوریتم رابطه (تعدادی از کارکنان، مسموم شدند، بر اثر استنشاق گاز) از جمله‌ی مزبور استخراج می‌شود.

البته این اولین بار نیست که محققی سعی تبدیل مسأله تجزیه وابستگی به مسأله برچسب‌زنی توالی تبدیل نماید. در [11] نویسندگان این کار را روی زبان چک به روش ساده‌ای انجام داده اند. آن‌ها برچسبی از روی داده‌ای اصلی ساخته‌اند که ترکیبی از برچسب ادات سخن هسته‌ی کلمه، جهت یال، فاصله‌ی کلمه‌ی هسته تا کلمه و برچسب روی یال است. آن‌ها در حقیقت یک برچسب با طول چهار برابر برچسب های معمولی ساخته‌اند و همین باعث شده که تعداد برچسب‌ها بسیار بالا رفته و دقت نتایج بسیار پایین (نزدیک ده درصد) باشد. تفاوت عمده‌ای این کار با کاری که ما صورت داده‌ایم این است که ما نمی‌خواهیم برچسب یال‌ها را پیش‌بینی کنیم زیرا اصلا نیازی به آن نداریم. همچنین نیاز نداریم تعداد زیادی برچسب را همزمان با هم پیش‌بینی کنیم و فقط کلمات متصل به فعل برای ما مهم هستند.

مسأله دیگر مقایسه کلی سرعت الگوریتم‌های برچسب‌گذاری توالی و الگوریتم‌های تجزیه وابستگی است. مخصوصا اینکه نیور در مقاله [12] اثبات کرد که عملکرد الگوریتمی که ارائه داده است روی تجزیه‌های غیر افکنشی از درجه خطی (بر اساس تعداد کلمه‌های جمله) است. او آزمایشات دیگری را هم در این زمینه اما برای درخت‌های افکنشی در [13] ارائه داد. مقاله‌ی [14] سعی کرده‌است برآوردی از سرعت انجام الگوریتم‌های مختلف تجزیه‌ی وابستگی بدست آورد. این مقاله نشان داده است که با روش کاوینگتون روی یک پردازنده تک هسته‌ای ۲,۴ گیگاهرتزی حدود چهل الی پنجاه هزار جمله قابل تجزیه هستند.

ما مقاله‌ای را نیافته‌ایم که به طور مستقیم سرعت الگوریتم‌های برچسب‌گذاری توالی را با الگوریتم‌های تجزیه وابستگی بپردازد. زیرا اساسا کارکرد این روش‌ها با یکدیگر متفاوت است. البته به طور کلی ویژگی‌های تعریف شده بر روی تجزیه وابستگی بیشتر از وظایفی مانند تشخیص موجودیت‌های اسمی است. در روش‌ای حل مسأله‌ی توالی اصولا ویژگی‌های تعریف می‌شوند و الگوریتم یادگیری به هرکدام از ویژگی‌ها ضریبی را نسبت می‌دهد. به طور خاص وقتی مسأله مانند تجزیه وابستگی سریع در بالا دودویی باشد،کافی است یک جمع ضریب‌دار از روی ویژگی‌ها بدست آمده و در مورد برچسب تصمیم‌گیری شود که اساسا به تعداد کلمات جمله وابسته نیست.

مقایسه بهترین سرعت روی برچسب‌گذاری ادات سخن با تجزیه‌ی وابستگی این ایده را تایید می‌کند. هر چند مقایسه روی رایانه‌های متفاوت و پیکره‌ای مختلفی اجرا شده است اما یک دید کلی در مورد سرعت عملیاتی هر کدام از این موارد به خواننده می‌دهد. مقاله [15] نشان می‌دهد سامانه SpeedRead می‌تواند در هر ثانیه ۲۲۶۲۱۸۳ کلمه را توکن بندی کند، روی ۵۶۴۹۷۷ کلمه برچسب ادات سخن بزند و روی ۱۵۳۱۹۴ کلمه برچسب‌های موجودیت‌های اسمی را مشخص نماید. نتایج ارائه شده در این مقاله روی یک پردازنده با چهار هسته (۸ نخ) و حافظه نهان ۸ مگابایت صورت گرفته است که نشان می‌دهد سرعت الگوریتم برچسب‌گذاری ادات سخن عملا بسیار بیشتر از تجزیه‌ی وابستگی است. فراموش نکنید که برچسب‌زن تجزیه‌وابستگی سریع تنها روی دو برچسب عمل می‌کند.

بر طبق مقاله‌ی [16] درباره‌ی مقایسه ۱۹ الگوریتم روی ۱۱ زبان نشان می‌دهد که دقت الگوریتم‌های تجزیه‌ی وابستگی به طور متوسط در حدود ۸۰ درصد است.

آماده شدن بستر لازم برای تعامل با مراکز علمی به‌منظور ایندکس گذاری از اطلاعات متادیتای مستنداتی که اطلاعات آن‌ها در سطح وب وجود ندارد.

۴- ارزیابی

نگارنده با استفاده از الگوریتم CRF یا میدان‌های تصادفی شرطی مدلی را بر روی بانک درخت تجزیه‌ی وابستگی دادگان نسخه‌ی اول در بازنمایی شماره یک و دو اجرا کرده است که در آن داده‌ای آموزشی شامل ۲۲۵۲۷ جمله و داده‌های آزمون شامل ۲۸۲۳ جمله بوده است. نتایج حاصل شده در جدول ۱-۱ و جدول ۱-۲ آمده است و نشان می‌دهد الگوریتم مزبور برای

شناسایی برچسب‌های V دقت قابل قبولی دارد. تنها ویژگی‌های استفاده شده در این مسأله خود کلمه، دو کلمه قبلی و بعدی، برچسب ادات سخن آن و برچسب‌های ادات سخن دو کلمه‌ی قبل و بعد از آن است.

همانطور که مشاهده می‌شود معیار F بدست آمده در بازنمایی شماره دو بالاتر از بازنمایی شماره یک است. در بازنمایی شماره یک طبق قرارداد، همواره کلمه «را» به عنوان مفعول در نظر گرفته می‌شود و این کلمه است که با یک یال وابستگی به فعل متصل می‌شود. اما در بازنمایی شماره ۲ مفعول واقعی به فعل متصل شده است. به همین دلیل است که دقت بدست آمده در بازنمایی دوم بیشتر است. توضیح بیشتر آنکه اطلاعات حذف شده طی عملیات برچسب‌زنی وابستگی سریع (خود یال‌ها) در روش دوم کمتر است. در روش اول همواره کلمه «را» به فعل متصل شده است و با توجه به اینکه ما از یک روش یادگیری ماشین استفاده کرده‌ایم و چنین الگویی به دلیل تکرار بسیار زیادش بر الگوهای دیگر غالب می‌شود، مسأله به سمت تناسب بیش از حد میل خواهد کرد.

۵- نتیجه‌گیری و کارهای آینده

با توجه به آنچه در بخش‌های قبل اشاره شد، روش برچسب‌زنی وابستگی سریع، با افت دقت کمی (در حدود ۱۰ درصد) با سرعت بسیار بیشتر، به طوری که در مقیاس بزرگ قابل استفاده باشد، می‌تواند به تشخیص طرفین رابطه کمک کند. تشخیص طرفین رابطه در یک جمله، تنها شروعی برای تشخیص رابطه است. عمده‌ترین مشکل در تشخیص رابطه‌ها در بیشتر زبان‌ها وجود کلمات عطف در جملات است که ممکن است چند فاعل را به یک فعل متصل کند. یعنی یک جمله چند رابطه را تولید کند. این موضوع می‌تواند برای یک فاعل به چند فعل یا یک فاعل به چند مفعول و ... نیز صادق باشد. مشکل دیگر تشخص ضمایر و به طور کلی ارجاع‌ها در یک متن است که برای تشخیص رابطه واقعی در آن بایستی به تشخیص مرجع آن‌ها بپردازیم. در آینده می‌توان با توجه به طرفیت فعل و شناسایی الگوهای مبتنی بر آن و با استفاده از یک تجزیه‌گر نحوی کم عمق، دقت تشخیص طرفین رابطه را بیشتر کرد. و همچنین راه حل‌هایی برای فائق آمدن بر کلمات عطف پیش‌بینی کرد. همچنین تشخیص مرجع ضمیر در زبان فارسی می تواند هدف بعدی تحقیقات ما باشد.

جدول ۱- نتایج آموزش تجزیه‌ی وابستگی سریع روی بانک درخت تجزیه‌ی وابستگی دادگان نسخه ۱ بازنمایی شماره یک

برچسب	دقت	بازخوانی	معیار F
V	۷۸,۹۸	۷۱,۸۲	۷۵,۲۳
O	۸۷,۳۵	۹۱,۰۴	۸۹,۱۶

جدول ۲- نتایج آموزش تجزیه‌ی وابستگی سریع روی بانک درخت تجزیه‌ی وابستگی دادگان نسخه ۱ بازنمایی شماره دو

برچسب	دقت	بازخوانی	معیار F
V	۸۷,۶۳	۸۳,۵۹	۸۵,۵۶
O	۹۲,۴۹	۹۴,۴۷	۹۳,۴۷

مراجع

[1] M. Banko, M. Cafarella, and S. Soderland, “Open information extraction for the web,” in *IJCAI*, 2007, pp. 2670–2676.

[2] E. Gabrilovich and S. Markovitch, “Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis,” in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, 2007, vol. 6, pp. 1606–1611.

[3] A. Yates and O. Etzioni, “Unsupervised Resolution of Objects and Relations on the Web,” in *Proceedings of NAACL HLT*, 2007, pp. 121–130.

[4] S. Kok and P. Domingos, “Extracting Semantic Networks from Text Via Relational Clustering,” *ECML PKDD ’08 Proc. 2008 Eur. Conf. Mach. Learn. Knowl. Discov. Databases - Part I*, vol. 5211, pp. 624–639, 2008.

[5] J. Christensen, Mausam, S. Soderland, and O. Etzioni, “Semantic Role Labeling for Open Information Extraction,” in *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, 2010, pp. 52–60.

[6] F. Wu and D. S. Weld, “Open Information Extraction Using Wikipedia,” in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 2010, pp. 118–127.

[7] O. Etzioni, A. Fader, J. Christensen, S. Soderland, and M. Mausam, “Open Information Extraction : The Second Generation,” in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, 2011, pp. 3–10.

[8] Mausam, R. Bart, S. Soderland, O. Etzioni, and T. Center, “Open Language Learning for Information Extraction,” *Proc. 2012 Conf. Empir. Methods Nat. Lang. Process.*, 2012.

[9] A. Fader, L. Zettlemoyer, and O. Etzioni, “Open Question Answering Over Curated and Extracted Knowledge Bases,” in *KDD ’14 Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 1156–1165.

[10] T. M. M. Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, “Toward an Architecture for Never-Ending Language Learning,” in *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)*, 2010.

[11] and M. S. Spoustová, Drahomíra, “Dependency Parsing as a Sequence Labeling Task,” *Prague Bull. Math. Linguist. 94*, no. 94, pp. 7–14, 2010.

[12] J. Nivre, “An efficient algorithm for projective dependency parsing,” in *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT)*, 2003, pp. 149–160.

[13] J. Nivre, “Non-Projective Dependency Parsing in Expected Linear Time,” in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing (IJCNLP) of the AFNLP*, 2009, vol. 1, pp. 351–359.

[14] A. Volokh and G. Neumann, “Transition-based Dependency Parsing with Efficient Feature Extraction,” 2011.

[15] Rami Al-Rfou and S. Skiena, “SpeedRead : A Fast Named Entity Recognition Pipeline,” in *COLING 2012: Technical Papers*, 2012, no. December, pp. 51–66.

[16] S. Buchholz and E. Marsi, “CoNLL-X shared task on multilingual dependency parsing,” in *Proceedings of the Tenth Conference on Computational Natural Language Learning*, 2006, pp. 149–164.