بررسی روش‌های مؤثر بر عملکرد تجزیه‌گر دستور مستقل از متن آماری زبان فارسی

محمدباقر صادق‌زاده*[1]، محمدرضا رزازی[2] و مسعود قیومی[3]
[1و2]دانشگاه صنعتی امیرکبیر، تهران، ایران
[3] پژوهشگاه علوم انسانی و مطالعات فرهنگی، تهران، ایران

چکیده

عدم دقّت در طراحی دستورهای مستقل از متن و استفاده از ساختارهای نامناسب مانند فرم نرمال چامسکی به خودی خود می‌تواند عملکرد تجزیه‌گرهای آماری مستقل از متن را تضعیف کند. در این پژوهش ساختار ترکیبات عطفی درخت‌بانک فارسی را مورد بررسی قرار دادیم. نتایج حاصل از این پژوهش نشان می‌دهد که با اضافه‌کردن وابستگی‌های ساختاری به دستورهای مستقل از متن و اصلاح قواعد اولیه، می‌توان از ترکیبات عطفی رفع ابهام کرد و صحت عملکرد تجزیه‌گر دستور مستقل از متن آماری را افزایش داد. فرض استقلال ضعیف، یکی از مشکلات مربوط به دستورهای مستقل از متن است که سعی شده است تا با تزریق وابستگی‌های ساختاری از طریق نشانه‌گذاری گره‌های والد و فرزند مرتفع شود. تأثیر ریزدانگی و درشت‌دانگی برچسب‌های اجزای واژگانی کلام و همین‌طور ادغام ناپایانه‌ها بر تجزیه‌گر دستور مستقل از متن آماری فارسی از جمله مواردِ مورد بررسی قرار گرفته‌شده در این پژوهش است.

واژگان کلیدی: دستور مستقل از متن آماری، تجزیه‌گر، ترکیبات عطفی، نشانه‌گذاری قواعد، برچسب اجزای واژگانی کلام

Studying impressive parameters on the performance of Persian probabilistic context free grammar parser

Mohammad Bagher Sadeghzadeh*[1], Mohammad Reza Razzazi[2] & Masood Ghayoomi[3]
[1,2]Amirkabir University of Technology, Tehran, Iran
[3]Institute for Humanities and Cultural Studies, Tehran, Iran

Abstract

In linguistics, a tree bank is a parsed text corpus that annotates syntactic or semantic sentence structure. The exploitation of tree bank data has been important ever since the first large-scale tree bank, The Penn Treebank, was published. However, although originating in computational linguistics, the value of tree bank is becoming more widely appreciated in linguistics research as a whole. For example, annotated tree bank data has been crucial in syntactic research to test linguistic theories of sentence structure against large quantities of naturally occurring examples.

The natural language parser consists of two basic parts, POS tagger and the syntax parser. A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some languages and assigns parts of speech to each word (and other token), such as noun, verb, adjective, etc., although generally computational applications use more fine-grained POS tags like 'noun-plural'. A natural language parser is a program that works out the grammatical structure of sentences, for instance, which groups of words go together (as "phrases") and which words are the subject or object of a verb.

Probabilistic parsers use knowledge of language gained from hand-parsed sentences to try to produce the most likely analysis of new sentences. These statistical parsers still make some mistakes, but commonly

* Corresponding author
*نویسندهٔ عهده‌دار مکاتبات

● تاریخ ارسال مقاله: ۱۳۹۷/۲/۵ ● تاریخ پذیرش: ۱۳۹۸/۴/۱۹ ● تاریخ انتشار: ۱۳۹۸/۱۰/۰۷ ● نوع مطالعه: پژوهشی

[DOI: 10.29252/jsdp.16.3.36]

work rather well. Inaccurate design of context-free grammars and using bad structures such as Chomsky normal form can reduce accuracy of probabilistic context-free grammar parser .

Weak independence assumption is one of the problems related to CFG. We have tried to improve this problem with parent and child annotation, which copies the label of a parent node onto the labels of its children, and it can improve the performance of a PCFG.

In grammar, a conjunction (conj) is a part of speech that connects words, phrases, or clauses that are called the conjuncts of the conjunctions. In this study, we examined the conjunction phrases in the Persian tree bank. The results of this study show that adding structural dependencies to grammars and modifying the basic rules can remove conjunction ambiguity and increase accuracy of probabilistic context-free grammar parser.

When a part-of-speech (PoS) tagger assigns word class labels to tokens, it has to select from a set of possible labels whose size usually ranges from fifty to several hundred labels depending on the language. In this study, we have investigated the effect of fine and coarse grain POS tags and merging non-terminals on Persian PCFG parser.

Keywords: Probabilistic context free grammar, parser, tree bank, conjunction phrases, parent annotation, child annotation, part of speech tags

۱- مقدمه

دستور[1]، جزئی از ساختار زبان طبیعی است که تشکیل گروهی از واژگان را برای ساخت سازه‌ها[2] یعنی گروهی از واژگان و یا عبارات واحد، بررسی می‌کند. این سازه‌ها می‌توانند با یکدیگر ترکیب شوند تا سازه‌های بزرگ‌تر و درنهایت جملات را تشکیل دهند. دستور مستقل از متن، رایج‌ترین مدل ریاضی برای مدل‌سازی ساختار سازه‌ای در زبان طبیعی است [1]. این دستور متعلق به نظریهٔ زبان‌های صوری است؛ به‌طوری‌که، زبان را به‌صورت مجموعه‌ای از جملات، جمله را به‌صورت دنباله‌ای از واژگان موجود در دایره لغات آن زبان و دستور را به‌صورت توصیفات رسمی از مجموعه جملات تشکیل‌دهندهٔ زبان در نظر می‌گیرد. علاوه‌بر دستورهای مستقل از متن، دستورهای صوری دیگری به نام دستورهای وابستگی وجود دارند که در آن سازه‌ها یا ساخت‌های عبارتی هیچ نقشی بازی نمی‌کنند؛ اما درعوض، ساختارهای نحوی یک جمله به‌کمک واژگان و روابط نحوی و معنایی بین آنها تعریف می‌شود. تجزیه‌گر[3] زبان طبیعی، برنامه‌ای است که با پردازش جملات و تشخیص نقش دستوری هر کلمه، ساختار دستوری کل جمله را مشخص می‌کند؛ به‌عنوان مثال کدام واژگان در یک گروه واقع می‌شوند و فاعل و مفعول یک فعل کدام کلمه است.

دستورهای ساخت‌عبارتی[4] در ابتدا توسط نوام چامسکی معرفی شد [2]. به دستورهای ساخت عبارتی، دستورهای سازه‌ای نیز اطلاق می‌شود و به دو نوع محدود آن می‌توان به دستورهای وابسته به متن و مستقل از متن اشاره کرد. وجود رابطهٔ سازه‌ای، مشخصهٔ دستورهای ساخت عبارتی است. بسیاری از تجزیه‌گرهای زبان طبیعی با توجه به دستورهای سازه‌ای شکل گرفته‌اند [3,4].

تجزیه‌گر ساخت‌سازه‌ای را به دو دسته اصلی قاعده‌بنیان و آماری تقسیم می‌کنند. در تجزیه‌گرهای قاعده‌بنیان مجموعه‌ای از قواعد زبانی تحت یک نظریه زبانی گردآوری و جملات با توجه به آن قواعد تجزیه می‌شوند. در تجزیه‌گرهای نوع دوم با استفاده از مدل‌های آماری جملات را تجزیه می‌کنند [5].

در [6] یک تجزیه‌گر قاعده‌بنیان به‌کمک زبان برنامه‌نویسی C و نرم‌افزارهای Lex و Yacc طراحی شده است. این تجزیه‌گر از سه بخش، فرهنگ لغت، تحلیل‌گر لغوی و تحلیل‌گر نحوی تشکیل شده است. در [7] نیز یک تجزیه‌گر قاعده‌بنیان معرفی شده و این تجزیه‌گر، متن ورودی را با استفاده از پیش‌پردازش‌هایی مانند اصلاح نویسهٔ فاصله و نیم‌فاصله، حذف نویسه‌های مربوط به رسم‌الخط عربی و جایگزینی آن با رسم‌الخط فارسی و غیره، استاندارد می‌سازد و درخت تجزیه متن ورودی را رسم می‌کند.

ایجاد دستی و اصلاح دستورهای زبان طبیعی، امری دشوار و زمان‌بر است و به تلاش زیادی نیاز دارد. از طرف دیگر، همواره دستور انسان‌نویسی‌شده به‌طور کامل رضایت‌بخش نیست و به کررات در پوشش جملات دیده‌نشده با شکست مواجه می‌شود [8]. استخراج خودکار دستورهای زبان، راه‌حلی برای این مسأله است. با توجه به سطح نظارت، روش‌های استخراج دستور به سه دستهٔ عمدهٔ با نظارت، نیمه‌نظارتی و بدون نظارت، تقسیم می‌شوند.

[1] grammar
[2] constituent
[3] parser
[4] Phrase Structure Grammar

[DOI: 10.29252/jsdp.16.3.36]

در [8] تلاشی صورت پذیرفته تا با استفاده از یک الگوریتم بدون نظارت به نام IIIO[1] که توسعه‌یافتهٔ الگوریتم IO[2] [9] است، به‌صورت خودکار دستورهای مستقل از متن فارسی را از پیکره‌های زبانی استخراج کند. فیلی در روش HIO با تغیر مدل آماری الگوریتم IO از $P(\alpha\rightarrow\beta|\alpha)$ به $P(\alpha\rightarrow\beta|\alpha,Parent(\alpha))$ سعی کرده است تا وابستگی به متن را مطابق با (Johnson, 1998) بهتر مدل کند. مدل HIO با دیگر روش‌های بدون نظارت استنتاج دستور از نگاه شکل دستور خروجی متفاوت است. دستور خروجی مدل‌های بدون نظارت، اغلب یک دستور مستقل از متن آماری به شکل عمومی یا شکل خاصی از فرم نرمال چامسکی هستند؛ اما در HIO یک قاعده به‌صورت دوتایی (R,C) تعریف می‌شود، به‌طوری‌که R یک قاعده در فرم نرمال چامسکی و C پدر غیرپایانهٔ سمت چپ قاعده R است [8].

در [10] به ارائهٔ روشی، جهت تشخیص کسرهٔ اضافه در متون فارسی، با استفاده از دستور مستقل از متن آماری همراه با تحلیل لغوی پرداخته شده است. عیسی‌پور با توجه به قواعد دستوری زبان فارسی و خصوصیات کسرهٔ اضافه، یک درخت‌بانک تنها متشکل از گروه‌های اسمی در زبان فارسی ایجاد کرده و یک تجزیه‌گر، دستورهای مستقل از متن آماری را بر روی آن آموزش داده است، بدیهی است که این تجزیه‌گر تنها قادر به شناسایی گروه اسمی در عبارات است.

قیومی در [11] تلاش کرده است تا با ایجاد یک درخت‌بانک فارسی[3] و آموزش تجزیه‌گر استنفورد[4]، نخستین تجزیه‌گر سازه‌ای آماری بانظارت را برای زبان فارسی ایجاد کند. نتایج حاصل از آزمایش این تجزیه‌گر نشان داده که ۵۳.۱۱ درصد تجزیهٔ ارائه‌شده توسط آن صحیح است. توسعهٔ یک تجزیه‌گر آماری، همواره تحت تأثیر داده‌های آموزش قرار می‌گیرد. پراکندگی داده، بزرگ‌ترین چالش در تحلیل‌های داده‌گراست و اگر تجزیه‌گرها با میزان کمی از داده، آموزش داده شوند و یا نوع داده آزمون و آموزش یکسان نباشد، آنها کارایی خوبی را از خود نشان نمی‌دهند [12]. قیومی با یک رویکرد خوشه‌بندی واژگان توسط الگوریتم برون[5] [13]، تلاش کرده است تا بر این مشکلات غلبه کند. نتایج حاصل از آزمایش تجزیه‌گر استنفورد با رویکرد خوشه‌بندی واژگان، دقت ۵۹/۳۲ درصد را گزارش کرده است [12].

[1] History based Inside Outside
[2] Inside Outside
[3] http://hpsg.fu-berlin.de/~ghayoomi/PTB/
[4] http://nlp.stanford.edu/software/lex-parser.shtml
[5] Brown algorithm

۲- دستور مستقل از متن آماری

شاید مهم‌ترین مسأله‌ای که تجزیه‌گرها با آن دست و پنجه، نرم می‌کنند، ابهام باشد. تجزیه‌گرها با ساختار نحوی در ارتباط هستند و ابهام در این حوزه به ابهام ساختاری[6] شناخته می‌شود [14]. ابهام ساختاری زمانی به وقوع می‌پیوندد که یک تجزیه‌گر بیش از یک درخت تجزیه به یک جمله اختصاص دهد.

ساده‌ترین راه تقویت دستورهای مستقل از متن برای رفع ابهام از ساختارهای زبان طبیعی، اضافه‌کردن احتمال به آن است که در سال ۱۹۶۹ توسط [15] معرفی شد و با نام دستورهای مستقل از متن آماری[7] (PCFG) شناخته می‌شود. هر دستور مستقل از متن آماری با چهار پارامتر $(N, \sum, R, S)$ تعریف می‌شود. این نوع از دستور، هر قانونی را با اضافه‌کردن یک احتمال تقویت می‌کند؛ به‌طوری‌که یک PCFG را می‌توان به شکل زیر تعریف کرد [16]:

N: یک مجموعه از نمادهای غیرپایانی است.

$\sum$: یک مجموعه از نمادهای پایانی است.

R: مجموعه‌ای از قواعد است که هر کدام به شکل $A\rightarrow B$ [p] تعریف می‌شوند. A یک ناپایانه است، B یک رشته متناهی از مجموعه پایانه‌ها و ناپایانه‌هاست و p احتمال $P(B|A)$ را بیان می‌کند.

S: نماد شروع دستور است.

تفاوت PCFG با یک CFG استاندارد، در اضافه‌کردن یک احتمال شرطی به هر قاعده مطابق $A\rightarrow B$ [p] است. در واقع در اینجا، p احتمال گسترش ناپایانهٔ A را به B بیان می‌کند. این احتمال را به‌طورمعمول به چند صورت $p(A\rightarrow B)$، $p(A\rightarrow B|A)$ و یا $p(RHS|LHS)$ نشان می‌دهند.

به‌دلیل این‌که PCFG خود توسعه‌یافتهٔ دستورهای مستقل از متن است، دو مشکل اصلی مربوط به این دستورها (فرض استقلال ضعیف و عدم حساسیت به وابستگی‌های لغوی) را به ارث می‌برد [14]. در دستورهای مستقل از متن، توسعهٔ یک ناپایانه، آزاد از متن است و به ناپایانه‌های مجاور آن وابسته نیست. در دستورهای مستقل از متن آماری نیز احتمال یک قاعده خاص مستقل از باقی درخت محاسبه می‌شود؛ درواقع، زبان طبیعی دارای مشخصاتی است که دستورهای مستقل از متن قادر به پوشش آنها نیست و باعث ایجاد ابهام در تجزیه جملات زبان می‌شود. عدم حساسیت به

[6] structural ambiguity
[7] Probabilistic Context Free Grammar

[DOI: 10.29252/jsdp.16.3.36]

وابستگی‌های لغوی باعث می‌شود تا تجزیه‌گر دستورهای مستقل از متن آماری به هر واژه، یک برچسب جزء واژگانی کلام اختصاص دهند و در ادامه عمل تجزیه مقدار لغوی آن واژه را در نظر نگیرد؛ به‌عنوان مثال، تجزیه‌گر به واژهٔ «رفت» برچسب Verb را اختصاص می‌دهد و در مراحل بعدی همین برچسب Verb را بدون در نظرگرفتن واژهٔ «رفت» استفاده می‌کند.

۳- درخت‌بانک فارسی

درخت‌بانک‌ها[1]، پیکره‌ای[2] هستند که به‌صورت دستی یا خودکار و به‌طور کامل ساختار نحوی را در سطح جمله و POS[3] یا اطلاعات مورفولوژیکی را در سطح کلمه نشانه‌گذاری کرده‌اند [17]. تجزیه‌گرهای آماری، دانش زبانی به‌دست‌آمده از درخت‌بانک را جهت یافتن محتمل‌ترین تحلیل از جملات جدید، استفاده می‌کنند.

درخت‌بانک فارسی[4] [5] یا PerTreeBank، نخستین درخت‌بانک برای زبان فارسی است که در قالب دستور ساخت‌سازه‌ای هسته‌بنیان [18] توسعه یافته و به‌صورت رایگان موجود است. یکی از ویژگی‌های دستور ساخت‌سازه‌ای هسته‌بنیان این است که علاوه‌بر بیان توصیف ساختاری سازه‌ها، دانش واژگانی واژه‌ها را توصیف می‌کند و روابط بین واژه‌های یک سازه در آن مشخص می‌شود [11]. این روابط می‌تواند به‌عنوان روابط معنایی نیز تلقی شوند. جهت توصیف ساختاریِ سازه‌های این درخت‌بانک از انشعاب دودویی[5] (قواعد به شکل نرمال چامسکی) استفاده شده، به‌طوری‌که یک عنصر سازه، هسته و دیگری وابسته است. روابط عناصر هسته با وابسته به سه شکل در این درخت بانک ظاهر می‌شود. الف) هسته–متمم[6] (وجود عنصر وابسته در ساخت عبارت الزامی است) و با برچسب‌هایی مانند: ADVPC، ADJPC، PPC، DPC، VPC،NPC مشخص شده است. ب) هسته–ادات[7] (وجود عنصر وابسته در ساخت عبارت الزامی نیست) و با برچسب‌هایی مانند: ADJPA، ADVPA، PPA، DPA، NPA، VPA مشخص شده است. ج) هسته–فاعل[8] (نقش فاعلی عنصر وابسته در جمله) و با برچسب VPS مشخص شده است. از برچسب VPF نیز برای دیگر روابط و برای ساخت‌هایی که در آن جابه‌جایی عناصر از جایگاه اصلی خود اتفاق افتاده است، استفاده می‌شود.

درخت‌بانک فارسی، شامل ۱۰۲۸ جمله از پیکره بی‌جن‌خان[9] است که از طریق روش قائم بر ذات[10] و به‌صورت نیمه‌خودکار توسعه یافته است. این درخت‌بانک ساختار داده XML دارد که ساختار عبارت مربوط به جملات را به‌صورت دستور مستقل از متن فراهم کرده است. در [19] تبدیل ساختار این درخت‌بانک جهت استفاده در تجزیه‌گرهای زبان طبیعی تشریح شده است. این ساختار نیز برای نخستین‌بار به‌صورت رایگان منتشر شده است[11].

۴- راه‌اندازی یک تجزیه‌گر آماری برای زبان فارسی

جهت بررسی عوامل مؤثر بر عملکرد تجزیه‌گر دستورهای مستقل از متن آماری زبان فارسی، درخت‌بانک فارسی [5] را به‌عنوان پیکره آموزشی در تجزیه‌گر آماری زبان فارسی به‌کار می‌گیریم. پس از اعمال تغییرات مطلوب بر درخت‌بانک فارسی و استخراج قواعد، با استفاده از الگوریتم تجزیه ویتربی [20] که توسط [21] به زبان برنامه‌نویسی پایتون پیاده‌سازی شده است، به تجزیه جملات زبان فارسی می‌پردازیم. الگوریتم تجزیه ویتربی همان نسخهٔ آماری الگوریتم معروف تجزیه CKY است که مجموعه‌ای از قواعد آماری را دریافت و جملات ورودی را مطابق با این قواعد تجزیه و درخت نحو را تولید می‌کند.

روش استاندارد برای ارزیابی تجزیه‌گرها و دستورها، معیارهای «ارزیابی تجزیه‌گر[12]» نامیده می‌شوند که نخستین‌بار در [22] ارائه شده است. این معیارها عبارتند از: دقت[13]، فراخوانی[14] و معیار F[15] که برای تجزیه‌گر زبان فارسی محاسبه شده‌اند. جهت محاسبه این معیارها از ابزار evalb[16] بهره گرفته شده است. از آنجا که نتایج این معیارها بسیار نزدیک به هم بوده‌اند، در ارزیابی‌ها تنها نتایج معیار F به‌عنوان میانگین دو معیار دیگر ارائه شده است.

[1] Tree Bank
[2] Corpus
[3] Part of Speech
[4] http://hpsg.fu-berlin.de/~ghayoomi/PTB.html
[5] Binary Branching
[6] Complement
[7] Adjunct
[8] Subject

[9] http://ece.ut.ac.ir/dbrg/bijankhan
[10] Bootstrapping
[11] https://github.com/mohammadsadeghzadeh/PennPerTree Bank
[12] PARSEVAL
[13] Precision
[14] Recall
[15] F-Measure
[16] Nlp.cs.nyu.edu/evalb

[DOI: 10.29252/jsdp.16.3.36]

هدف از معیارهای «ارزیابی تجزیه‌گر» این است که بررسی شود چه تعداد از سازه‌ها در درخت تجزیه‌شده توسط تجزیه‌گر، شبیه به سازه‌های موجود در درخت تجزیه‌شده مرجع است. در این معیارها همواره برای هر جملهٔ موجود در مجموعهٔ آزمون یک درخت تجزیه مرجع وجود دارد.

سازهٔ C_h از درخت تجزیه تست‌شده[1] را صحیح می‌گویند، اگر سازهٔ C_r از درخت تجزیه مرجع با همان نقطه شروع، نقطه پایان و با همان ناپایانه‌ها وجود داشته باشد. با توجه به این تعریف، هر کدام از معیارهای دقت، فراخوانی و F-Measure به‌ترتیب محاسبه می‌شوند.

$$\text{دقت} = \frac{\text{تعداد اجزای صحیح در درخت نحو}}{\text{تعداد کل اجزای در درخت نحو}} \quad (۱)$$

$$\text{فراخوانی} = \frac{\text{تعداد اجزای صحیح در درخت نحو}}{\text{تعداد کل اجزای در درخت نحو مرجع}} \quad (۲)$$

$$F-Measure = \frac{2(\text{فراخوانی} * \text{دقت})}{\text{فراخوانی} + \text{دقت}} \quad (۳)$$

به‌دلیل حجم کم درخت بانک و نبود دادهٔ آزمون معیار استاندارد بر اساس ساخت سلسله‌مراتبی، از روش اعتبارسنجی متقابل ده‌قسمتی[2] جهت ارزیابی تجزیه‌گر فارسی استفاده کردیم. در این روش، کل داده موجود به ده قسمت مختلف تقسیم شده و در هر مرحله ۹ قسمت به‌عنوان داده آموزش تجزیه‌گر و یک قسمت برای ارزیابی تجزیه‌گر استفاده می‌شود. میانگین نتایج حاصل از ده آزمایش به‌عنوان کارایی تجزیه‌گر در نظر گرفته می‌شود.

۵- عوامل مؤثر بر عملکرد تجزیه‌گر آماری

قبل از بررسی عوامل مؤثر بر عملکرد یک تجزیه‌گر زبان طبیعی، می‌بایست اجزای تشکیل‌دهندهٔ آن را شناخت. تجزیه‌گر زبان طبیعی از دو بخش اساسی، تحلیل‌گر لغوی یا همان برچسب‌گذار اجزای واژگانی کلام (POS tagger) –که در اینجا به‌اختصار به آن برچسب‌گذار می‌گوییم– و تحلیل‌گر نحوی تشکیل می‌شود. جهت آموزش POS tagger و تحلیل‌گر نحوی می‌توان به‌ترتیب از پیکرهٔ برچسب‌گذاری‌شده و درخت‌بانک استفاده کرد.

برچسب‌گذار اجزای واژگانی، متن ورودی را دریافت می‌کند و بعد از جداسازی واژگان، متناسب با دادهٔ آموزشی و مدل آماری خاصی، برچسب‌های اجزای واژگانی کلام را به هر واژه اختصاص می‌دهد. بدیهی است که میزان دقت این بخش می‌تواند بر عملکرد تحلیل‌گر نحوی و درنتیجه بر میزان دقت تجزیه‌گر تأثیرگذار باشد. اگر برچسب‌گذار به واژه‌ای برچسب اشتباه اختصاص دهد، این اشتباه می‌تواند در سرتاسر درخت تجزیه منتشر شود و از دقت تجزیه‌گر بکاهد. پژوهش‌های زیادی در زمینهٔ برچسب‌گذار اجزای واژگانی کلام صورت پذیرفته است [23]، [24]، [25]. برچسب‌های اجزای واژگانی کلام نقش دستوری هر واژه را در جمله مشخص می‌کنند. بعضی از مجموعه‌برچسب‌ها مشخصه‌هایی دارند که برای مراحل بعدی پردازش یا برای پیش‌بینی رفتار واژگان مجاور مناسب خواهند بود؛ استفاده از برچسب‌های درشت‌دانه یا ریزدانه از جمله مواردی است که در ایجاد تجزیه‌گر مورد توجه قرار می‌گیرد.

تحلیل‌گر نحوی، برچسب‌های اجزای واژگانی کلام را از برچسب‌گذار دریافت می‌کند و بر اساس یک الگوریتم تجزیه و داده آموزشی مناسب مانند درخت‌بانک فارسی، شروع به ساخت درخت تجزیه می‌کند. کمیت و کیفیت درخت‌بانک استفاده‌شده جهت آموزش، مهم‌ترین عامل در میزان دقت این بخش است. همان‌طور که گفته شد، این بخش می‌تواند تحت تاثیر برچسب‌گذار قرار گیرد. جهت بررسی میزان اثربخشی یک درخت‌بانک بر عملکرد این بخش، برچسب‌های اجزای واژگانی کلام را می‌توان به‌طور مستقیم از ورودی به تحلیل‌گر نحوی ارسال کرد. تحلیل‌گر نحوی علاوه‌بر یک درخت‌بانک جهت استخراج قواعد نحوی زبان به یک الگوریتم تجزیه نیز نیاز دارد. با تغییر الگوریتم تجزیه، مدل آماری و قواعد استخراجی از درخت‌بانک می‌توان عملکرد تحلیل‌گر نحوی را بهبود بخشید. در این پژوهش ما از طریق نشانه‌گذاری قواعد، مدل آماری را اندکی تغییر داده و با تغییر قواعد مربوط به ترکیبات عطفی سعی خواهیم کرد تا از این ترکیبات رفع ابهام کنیم.

(جدول-۱): توزیع طول جملات در درخت‌بانک فارسی
(Table-1): Sentence length distribution in PerTreeBank

طول جملات کمتر از	تعداد جملات
۱۰	۱۰۷
۲۰	۴۱۰
۳۰	۶۸۵
۴۰	۸۴۶
تمام طول‌ها	۱۰۰۰

[1] Hypothesis
[2] 10-fold cross validation

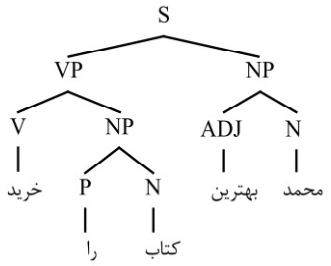
[DOI: 10.29252/jsdp.16.3.36]

از آنجا که طول جملات نیز عاملی جهت تأثیر بر عملکرد تجزیه‌گر هستند، در این آزمایش‌ها سعی شده است تا نتایج مربوط به طول جملات مختلف گزارش و بررسی شوند. جدول (۱) توزیع طول جملات مختلف را در درخت‌بانک فارسی نشان می‌دهد.

۱-۵- تأثیر دانگی برچسب‌های اجزای واژگانی کلام بر عملکرد تجزیه‌گر

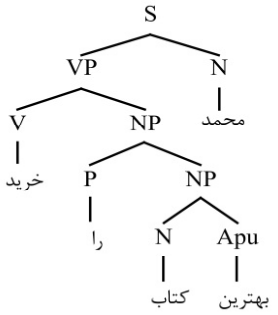
بعضی از برچسب‌ها مشخصه‌هایی دارند که در مراحل بعدی پردازش یا برای پیش‌بینی رفتار واژگان مجاور مناسب خواهند بود. برای مثال، در زبان فارسی صفت عالی قبل از اسم قرار می‌گیرد، درحالی‌که صفت ساده یا نسبی بعد از هستهٔ یک عبارت اسمی ظاهر می‌شود. تمایز قائل‌شدن بین این نوع از صفات در سطح برچسب‌گذار می‌تواند تحلیل را در سطح تجزیه عبارات آسان کند.

برچسب‌گذاری اجزای واژگانی کلام را می‌توان به دو صورتِ ریزدانه و درشت‌دانه انجام دارد. به‌عنوان مثال، برای واژهٔ «بهترین» می‌توان برچسب «صفت» با علامت ADJ یا برچسب «صفت، مثبت، عالی» با علامت Apu را در نظر گرفت [25]. با توجه به این‌که صفت در زبان فارسی هم می‌تواند قبل از اسم بیاید (صفت عالی) و هم بعد از اسم (صفت ساده)، درصورتی‌که برچسب درشت‌دانه اجزای واژگانی کلام استفاده شود، دو قاعده برای زبان فارسی می‌توان در نظر گرفت: ۱) NP → ADJ N و ۲) NP → N ADJ. درصورتی‌که یک تجزیه‌گر از این دو قاعده جهت تجزیه جملاتی (که در آن یک صفت عالی بین دو اسم قرار گرفته است) استفاده کند، مطابق شکل (۱) دچار اشتباه می‌شود.



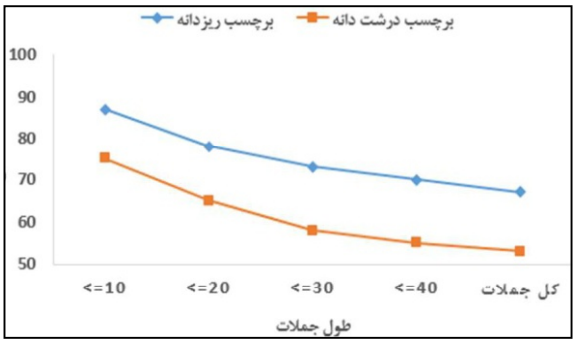
(شکل-۱): برچسب درشت‌دانه و تجزیه اشتباه
(Figure-1): Coarse POS tags and incorrect parsing

درصورتی‌که از برچسب‌های ریز اجزای واژگانی کلام جهت برچسب‌گذاری استفاده شود، قواعد قبلی را می‌توان به‌صورت: ۱) NP →Adj N و ۲) NP →N Apu بازنویسی کرد. همان‌طور که در شکل (۲) آمده است، به‌دلیل رفع ابهامی که در سطح برچسب‌ها، انجام شده، تجزیه‌گر توانسته است، تجزیهٔ درستی را ارائه دهد.



(شکل-۲): برچسب ریزدانه و تجزیه صحیح
(Figure-2): Fine POS tags and correct parsing

نتایج ارزیابی تجزیه‌گر با برچسب درشت‌دانه و ریزدانه اجزای واژگانی کلام در شکل (۳) ارائه شده است. در نخستین مدل از تجزیه‌گر زبان فارسی، از برچسب‌های درشت‌دانه اجزای واژگانی کلام درخت‌بانک فارسی که شمار آنها به پانزده عدد می‌رسد، استفاده شده است. در این مدل از هیچ نوع واحد برچسب‌گذاری استفاده نمی‌شود و تجزیه‌گر، تنها دنباله‌ای از برچسب‌ها را به‌عنوان ورودی دریافت می‌کند. امتیاز ۵۳.۶۷ درصد به‌عنوان کارایی این تجزیه‌گر در نظر گرفته شده است. در دومین مدل از تجزیه‌گر زبان فارسی، از برچسب‌های ریزدانه اجزای واژگانی کلام (تنوع این برچسب‌ها در درخت‌بانک فارسی به ۲۴۷ عدد می‌رسد) به جای برچسب‌های درشت‌دانه استفاده شده است. استفاده از این مدل می‌تواند، تعریف دستورهای زبان را دقیق‌تر و به بهبود عملکرد تجزیه‌گر حاصل از آن تا ۶۵/۵۵ درصد کمک کند. شکل (۳) نشان می‌دهد که با افزایش طول جملات، عملکرد تجزیه‌گر دستورهای مستقل از متن آماری زبان فارسی همواره کاهش پیدا می‌کند.



(شکل-۳): نتایج ارزیابی تجزیه‌گر حاصل از برچسب‌های ریزدانه و درشت‌دانه اجزای واژگانی کلام(معیار F)
(Figure-3): Test results of fine and coarse POS tags in Persian parser

[DOI: 10.29252/jsdp.16.3.36]

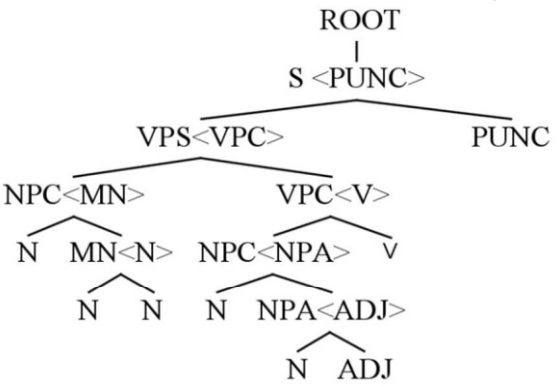
۲-۵- نشانه‌گذاری قواعد

زبان طبیعی، وابستگی‌هایی را بین ساختار جمله بیان می‌کند که دستورهای مستقل از متن آماری ساده قادر به نمایش آنها نیستند. انجام نشانه‌گذاری گره‌های درخت‌بانک، راهی جهت تزریق وابستگی ساختاری به دستورهای مستقل از متن است [26].

جانسون در [26] سعی کرده است تا با استفاده از نشانه‌گذاری پدر، میزانی از نمایش این وابستگی‌ها را توسط دستورهای مستقل از متن آماری محقق سازد و نشان داده است که کارایی یک تجزیه‌گر دستورهای مستقل از متن آماری، تنها با نشانه‌گذاری هر گره فرزند با برچسب گره پدرش می‌تواند به‌طور قابل ملاحظه‌ای بهبود یابد. این نوع نشانه‌گذاری بسته به قواعد مربوط به یک زبان خاص و حجم داده آموزشی، تا سطوح مختلفی قابل اعمال است و نتایج متفاوتی دارد. نویسندگان این مقاله در [27] با به‌کارگیری این نشانه‌گذاری بر روی درخت‌بانک فارسی نشان داده‌اند که کارایی تجزیه‌گر زبان فارسی با اعمال این تغییر، می‌تواند چهار درصد افزایش یابد.

نشانه‌گذاری به طرق مختلف می‌تواند انجام پذیرد: نشانه‌گذاری با پدر و یا فرزند. سؤالی که اینجا مطرح می‌شود، این است که آیا هر نوع نشانه‌گذاری برای هر نوع درخت‌بانک و با هر حجمی از داده‌ها مناسب است و این‌که کدام نشانه‌گذاری اثر بیشتری بر عملکرد تجزیه‌گر حاصل از درخت‌بانک فارسی می‌گذارد.

استفاده از نشانه‌گذاری فرزند نیز می‌تواند به‌عنوان روشی جهت تزریق وابستگی به دستورهای مستقل از متن آماری محسوب شود. در این نوع نشانه‌گذاری، برچسب هر گره را می‌توان با مقدار برچسب گره فرزندان بازنویسی کرد؛ اما از آنجا که استفاده از تمامی فرزندان در نشانه‌گذاری درخت‌بانک فارسی که حجم کمی از جملات را شامل می‌شود، ممکن است موجب بیش‌برازش[1] شود، در این راستا سعی می‌شود تا تنها یکی از فرزندان جهت نشانه‌گذاری انتخاب شود. بیش‌تر قواعد موجود در درخت‌بانک فارسی از انشعاب دودویی برخوردارند و بین فرزندان، رابطهٔ هسته-وابسته وجود دارد. بهره‌گیری از این رابطه پیش از این اثربخشی خود را در دستورهای واژگانی‌شده [28] نشان داده است. ما جهت انتخاب تنها یکی از فرزندان بر اساس معیارهایی چون میزان تکرار آن فرزند و یا موقعیت آن در قواعد تولید، چهار نوع سیاست: فرزند با بیشترین بسامد، فرزند با کمترین بسامد، فرزند چپ و فرزند راست را در انتخاب فرزندان معرفی و نتایج حاصل از آن را در کارایی تجزیه‌گر آماری گزارش می‌کنیم. شکل (۴) نشانه‌گذاری یک درخت نحو را با استفاده از فرزند راست سازه نشان می‌دهد.



(شکل-۴): نشانه‌گذاری یک درخت نحو با استفاده از فرزند راست سازه

(Figure-4): Right child annotation in a sample syntax tree

اگر C(lhs,ch) را تعداد قواعدی از درخت بانک در نظر بگیریم که LHS[2] آنها برابر با lhs و ch عضوی از RHS[3] آنها باشد، فرزند با بیشترین بسامد برای هر قاعده تولید را این‌گونه تعریف می‌کنیم: برای هر قاعده R، ch1 = MostFrequency (R)، به‌طوری‌که ch1 عضوی از RHS(R) باشد و به‌ازای هر ch2 عضو RHS(R)، رابطه C(LHS(R),ch1) > C(LHS(R),ch2) برقرار باشد. با توجه به اینکه درخت‌بانک فارسی حجم کمی از داده را شامل می‌شود، انتخاب فرزند با بیشترین بسامد می‌تواند علاوه بر اضافه‌کردن وابستگی ساختاری به دستور مستقل از متن زبان فارسی، از بیش‌برازش در داده آموزش جلوگیری کند.

فرزند با کمترین بسامد برای هر قاعده تولیدی نیز اینگونه تعریف می‌شود. برای هر قاعده R، ch1= LowestFrequency(R)، به‌طوری‌که ch1 عضوی از RHS(R) باشد و به‌ازای هر ch2 عضو RHS(R)، رابطه C(LHS(R),ch1) < C(LHS(R),ch2) برقرار باشد. انتخاب فرزند با کمترین تکرار می‌تواند ضمن تزریق وابستگی ساختاری به دستورهای مستقل از متن زبان، تنوع را در قواعد تولیدی افزایش دهد. افزایش قواعد هم می‌تواند موجب رفع ابهام از بعضی ساختارها شود و هم ممکن است، زمینهٔ بیش‌برازش را به وجود آورد.

در نشانه‌گذاری قواعد، مدل آماری جهت محاسبه برآورد بیشینه احتمال تغییر می‌کند. این تغییر موجب می‌شود تا احتمال دقیق‌تری از هر قاعده به‌دست آید. به‌عنوان مثال

[1] Overfitting

[2] Left Hand Side

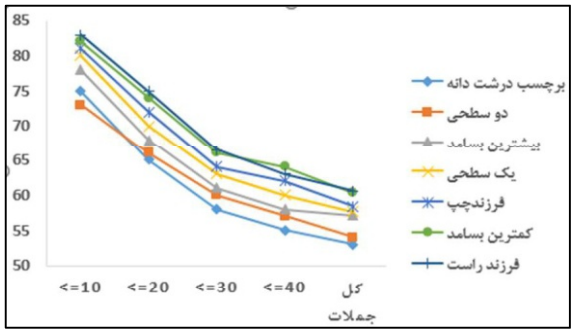
[3] Right Hand Side

[DOI: 10.29252/jsdp.16.3.36]

در انتخاب فرزند با بیشترین بسامد، مدل آماری از P(R) به P(R,MostFrequency(R)) تغییر می‌کند.

نتایج حاصل از شش نوع نشانه‌گذاری مختلف در شکل (۵) بیان شده است. کارایی تجزیه‌گر با هر کدام از نشانه‌گذاری‌های فرزند راست، فرزند با کمترین بسامد، فرزند چپ، پدر یک سطحی، فرزند با بیشترین بسامد و پدر دو سطحی به ترتیب ۶۰/۵۳، ۶۰/۲۹، ۵۸/۳۷، ۵۷/۶۱، ۵۷/۰۹ و ۵۴/۰۱ است. در بین این نشانه‌گذاری‌ها فرزند راست و فرزند با کمترین بسامد بهترین عملکرد را از خود نشان داده‌اند. درخت‌بانک فارسی و دستورهای زبان فارسی ویژگی‌هایی دارند که استفاده از این دو نوع نشانه‌گذاری، وابستگی ساختاری بهتری را به تجزیه‌گر اضافه می‌کنند. نمونه‌ای از این ویژگی‌ها را در (بخش ۵-۴) در رفع ابهام از ترکیبات عطفی بیان خواهیم کرد.

نشانه‌گذاری والد اگرچه ضعف دستورهای مستقل از متن را در پوشش وابستگی ساختاری زبان تا حدی از بین می‌برد، افزایش سطح آن موجب تنوع در دستورهای استخراجی از درخت بانک و بیش‌برازش در مجموعه آموزش می‌شود و از کارایی تجزیه‌گر می‌کاهد.



(شکل-۵): نتایج ارزیابی تجزیه‌گر حاصل از نشانه‌گذاری قواعد و تجزیه‌گر پایه (برچسب درشت دانه)

(Figure-5): evaluating grammar annotation effects on a base parser (Coarse POS tags)

۳-۵- ادغام ناپایانه‌ها

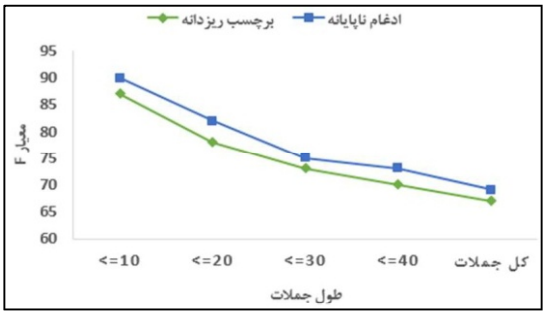
همان‌طورکه پیش از این گفته شد، درخت‌بانک فارسی بر اساس دستور ساخت‌سازه‌ای هسته‌بنیان ایجاد شده است و روابط موجود در این دستور (هسته-متمم، هسته-ادات، هسته-فاعل و هسته-پرکننده) به‌صورت انشعاب‌های دودویی به درخت‌بانک فارسی اضافه و باعث تنوع قواعد شده است؛ به‌عنوان مثال برای نمایش عبارات فعلی در درخت‌بانک فارسی از چهار ناپایانه VPC، VPA، VPS و VPF استفاده شده است که جهت ادغام، آنها را با ناپایانه VP جایگزین کردیم. دیگر عبارات اسمی، صفتی، قیدی و غیره هم مطابق با جدول (۲) ادغام می‌شوند.

(جدول-۲): تنوع ناپایانه‌ها و ادغام آنها در درخت‌بانک فارسی

(Table-2): merging non-terminals to reduce inappropriate variety in PerTreeBank

ادغام ناپایانه‌ها به	ناپایانه‌های نامزد جهت ادغام
ADVP	ADVPC, ADVPA
ADJP	ADJPC, ADJPA
PP	PPC, PPA
DP	DPC, DPA
NP	NPC, NPA
VP	VPC, VPA, VPS, VPF

ادغام ناپایانه‌ها، قواعد تولید حاصل از درخت‌بانک فارسی را به دستورهای مستقل از متن نزدیک‌تر می‌کند و علاوه‌بر کاهش ابهام، به‌طور طبیعی باعث افزایش دقت تجزیه‌گر می‌شود. شکل (۶) نشان می‌دهد که ادغام ناپایانه‌ها در یک تجزیه‌گر با برچسب‌های ریزدانه می‌تواند به میزان سه درصد عملکرد تجزیه‌گر پایه را بهبود بخشد. در بخش بعد، درخت‌بانک فارسی را که در آن ناپایانه‌ها ادغام شده‌اند، به‌کار می‌گیریم تا ضمن کاهش پیچیدگی قواعد، از ترکیبات عطفی موجود در آن رفع ابهام کنیم.



(شکل-۶): نتایج ارزیابی تجزیه‌گر حاصل از ادغام ناپایانه‌ها و تجزیه‌گر پایه (برچسب ریزدانه)

(Figure-6): merged non-terminals and its effect on the base parser (fine POS tags)

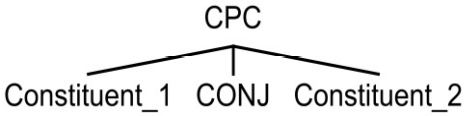
۴-۵- رفع ابهام از ترکیبات عطفی

عطف یا میانجی (با نشانهٔ: CONJ)، تکواژ دستوری است که برای به‌دست‌آوردن عبارتی تازه از دو عبارت کوچک‌تر به‌کار می‌رود؛ مانند «موهای [[زبر] و [خشن]]»، «تیم‌های [[پرسپولیس تهران] و [شموشک نوشهر]]» و «[[این مسابقه با نتیجه مساوی پایان یافت] و [به ضربات پنالتی کشیده شد]]».

این نوع عبارات را در زبان فارسی می‌توان همانند درخت شکل (۷) نشان داد. در قاعده تولید این نوع درخت تنها یک عبارت CPC موجود است و این عبارت به هر دو سازهٔ Constituent_1 و Constituent_2 اشراف دارد. این دو سازه در زبان فارسی در بیش‌تر مواقع همگن هستند (یا هر دو

[DOI: 10.29252/jsdp.16.3.36]

اسم‌اند، یا صفت‌اند، یا جمله‌اند و غیره). تشخیص همگن‌بودن این سازه‌ها با توجه به این قاعده تولید امکان‌پذیر است.



(شکل-۷): نمایش ترکیبات عطفی در زبان فارسی با استفاده از درخت نحو

(Figure-7): conjunction in Persian using the syntax tree

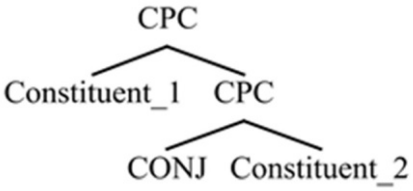
درصورتی‌که از نمایش درختی موجود در شکل (۷) برای نمایش درختی سازه‌های موجود در زبان فارسی استفاده شود، قواعدی مانند جدول (۳) تولید خواهد شد.

(جدول-۳): قواعد تولیدی مناسب برای نمایش ترکیبات عطفی در زبان فارسی

(Table-3): Appropriate CFG grammars for conjunction constituents

سازه نحوی	قاعده تولید
NP	CPC → NP CONJ NP
ADJP	CPC → ADJP CONJ ADJP
PP	CPC → PP CONJ PP
VP	CPC → VP CONJ VP

اما با توجه به این‌که در درخت‌بانک فارسی از انشعاب دودویی یا شکل نرمال چامسکی برای نمایش این ساختار استفاده شده، مطابق با شکل (۸)، قواعد متفاوت خواهند بود. وجود دو عبارت CPC، تشخیص شرط همگن‌بودن دو سازهٔ Constituent_1 وConstituent_2 را دچار مشکل می‌کند و زمینهٔ ابهام را در تجزیهٔ این ترکیبات فراهم می‌کند.



(شکل-۸): نمایش ترکیبات عطفی در درخت بانک فارسی با استفاده از انشعاب دودویی

(Figure-8): Binary Conjunction Constituents in PerTreeBank

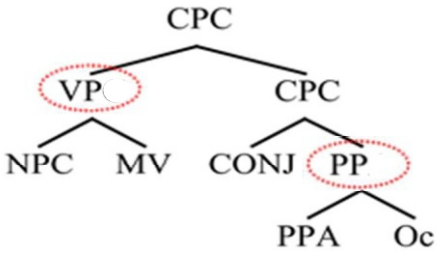
به‌دلیل این‌که از نمایش درختی شکل (۸) برای مدلسازی سازه‌های عطفی موجود در درخت‌بانک فارسی استفاده شده، قواعدی به شکل نرمال چامسکی مانند جدول (۴) از آن استخراج خواهد شد.

(جدول-۴): قواعد به شکل نرمال چامسکی برای ترکیبات عطفی در درخت‌بانک فارسی

(Table-4): Chomsky normal rules for conjunction constituents in PerTreeBank

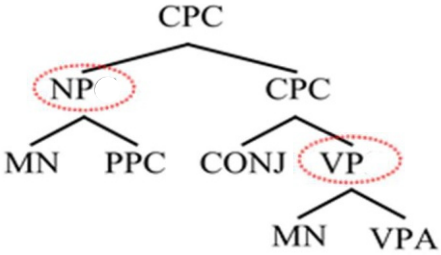
سازه نحوی	قاعده اول	قاعده دوم
NP	CPC → NP CPC	CPC → CONJ NP
ADJP	CPC → ADJP CPC	CPC → CONJ ADJP
PP	CPC → PP CPC	CPC → CONJ PP
VP	CPC → VP CPC	CPC → CONJ VP

درخت‌های شکل (۹) و شکل (۱۰)، همگی نمونه‌های اشتباهی از تجزیه‌های انجام‌شده توسط تجزیه‌گر حاصل از درخت‌بانک فارسی هستند؛ در همهٔ این نمونه‌ها، تجزیه‌گر در شناسایی همگن‌بودن سازه‌ها دچار اشتباه شده است. در درخت شکل (۹)، تجزیه‌گر برای قاعده نخست از CPC → VP CPC و برای قاعده دوم به اشتباه از CPC → CONJ PP به جای CPC → CONJ VP استفاده کرده است. در درخت شکل (۱۰) نیز تجزیه‌گر برای قاعده نخست از CPC → NP CPC و برای قاعده دوم به اشتباه از CPC → CONJ VP به جای CPC → CONJ NP استفاده کرده است.



(شکل-۹): درج سازه ناهمگن CPC→CONJ PPA در درخت نحو توسط تجزیه‌گر آماری زبان فارسی

(Figure-9): Incorrect syntax tree in Persian PCFG parser due to heterogeneous conjunction constituents



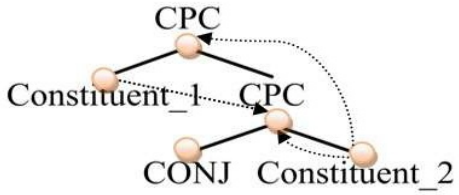
(شکل-۱۰): درج سازه ناهمگن CPC→CONJ VPS در درخت نحو توسط تجزیه‌گر آماری زبان فارسی

(Figure-10): Incorrect syntax tree in Persian PCFG parser due to heterogeneous conjunction constituents

مشکل در تحلیل نحوی این درخت‌ها از عدم تشخیص صحیح سازه‌ها توسط عبارت CPC و استفاده از شکل نرمال

[DOI: 10.29252/jsdp.16.3.36]

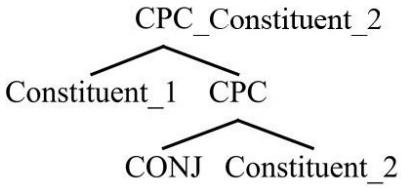
چامسکی در طراحی این قواعد ناشی می‌شود؛ بنابراین جهت رفع این مشکل در تجزیه‌گر باید سازه نحوی یک قاعده CPC را به دیگر قاعده CPC، مطابق با شکل (۱۱)، معرفی کرد تا تشخیص صحیح سازه‌ها امکان‌پذیر شود.



(شکل-۱۱): کمک به تشخیص صحیح سازه‌ها توسط عبارت CPC در درخت‌بانک فارسی

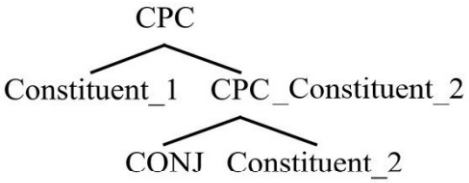
(Figure-11): Modifying CPC Constituents in PerTreeBank in different ways

درخت‌های شکل‌های (۱۲، ۱۳ و ۱۴) حاصل بازنویسی هستند و همهٔ آنها برای رفع ابهام مفید به نظر می‌رسند. درخت‌های شکل‌های (۱۳ و ۱۴) به طور عادی عملکرد بهتری را در رفع ابهام از ترکیبات عطفی نسبت به درخت شکل (۱۲) از خود نشان می‌دهند؛ این درحالی است که برای رفع ابهام از ترکیبات عطفی در درخت شکل (۱۲) هنوز به نشانه‌گذاری پدر یک سطحی نیاز است.



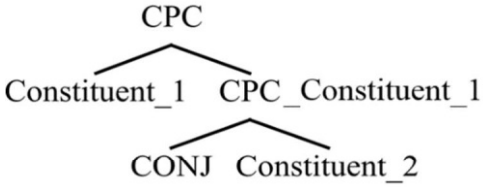
(شکل-۱۲): نشانه‌گذاری CPC نخست با سازهٔ دوم

(Figure-12): Annotating first CPC with second constituent



(شکل-۱۳): نشانه‌گذاری CPC دوم با سازهٔ دوم

(Figure-13): Annotating second CPC with second constituent



(شکل-۱۴): نشانه‌گذاری CPC دوم با سازهٔ نخست

(Figure-14): Annotating second CPC with first constituent

در این پژوهش برای رفع ابهام از ترکیبات عطفی از درخت شکل (۱۴) استفاده می‌کنیم که مطابق با آن قواعدی به‌صورت جدول (۵) تولید خواهد شد. در این قواعد شکل نرمال چامسکی حفظ شده اما قاعده دوم به نحوی تغییر کرده است تا با توجه به سازهٔ نحوی موجود در قاعده نخست به‌راحتی و بدون ابهام توسط تجزیه‌گر انتخاب شوند.

(جدول-۵): تصحیح قواعد مربوط به ترکیبات عطفی درخت‌بانک فارسی با استفاده از همزاد چپ

(Table-5): Modifying CPC Constituents in PerTreeBank with left sibling

سازه نحوی	قاعده اول	قاعده دوم
NP	CPC →NP CPC_NP	CPC_NP→ CONJ NP
ADJP	CPC→ADJP CPC_ADJP	CPC_ADJP → CONJ ADJP
PP	CPC → PP CPC_PP	CPC_PP → CONJ PP
VP	CPC → VP CPC_VP	CPC_VP → CONJ VP

شکل (۱۵) درخت تجزیه مرجع جمله «والد$^{Ncsp---}$ ش$^{Cez--nshc}$ کارگر$^{Ncsp-z}$ راه‌آهن$^{Ncsp--}$ بود$^{Vpyksht----}$ و$^{Jr-}$ تامین$^{Ncsp--z}$ معاش$^{Ncsp-z}$ زندگی$^{Ncsp---}$ برا$^{E-}$ یش$^{Cez--eshv}$ سخت$^{-Aps-}$.^{Oc}» را نشان می‌دهد. شکل (۱۶) درخت تجزیه همین جمله را با استفاده از یک تجزیه‌گر ساده (ناپایانه‌های ادغام شده) بیان می‌کند. اگرچه این تجزیه‌گر مرز بین سازه‌های عطفی را به‌درستی تشخیص داده اما در تعیین نوع عبارت «تأمین معاش زندگی برایش سخت.» (به‌دلیل حذف فعل به قرینه لفظی) دچار اشتباه شده و آن را به‌عنوان یک عبارت اسمی تجزیه کرده است. درخت شکل (۱۷)، تجزیه جمله یادشده را با استفاده از یک تجزیه‌گر مجهز به رفع ابهام عطفی نشان می‌دهد. از آنجا که این تجزیه‌گر ملزم به انتخاب سازه‌های همگن در ترکیبات عطفی است، نوع عبارت «تأمین معاش زندگی برایش سخت.» را به‌درستی همانند سازه نخست («والدش کارگر راه‌آهن بود»)، از نوع فعلی در نظر گرفته است. به هر حال این نوع تجزیه هم خالی از اشکال نیست و «سخت» را به‌عنوان فعل مرکب معرفی کرده است. اصلاح قواعد درخت‌بانک فارسی از طریق تزریق وابستگی ساختاری به ترکیبات عطفی و درنتیجه الزام تجزیه‌گر به انتخاب سازه‌های همگن در این ترکیبات موجب شده است تا این تجزیه‌گر عملکرد بهتری را نسبت به تجزیه‌گر پایه از خود نشان دهد.

[DOI: 10.29252/jsdp.16.3.36]

درخت‌بانک فارسی شامل ۲۸۳۴۱ قاعده داخلی است که ۳۲۹۲ قاعدهٔ آنها مربوط به ترکیبات عطفی است. بنابراین در تجزیه‌گر جاری با تغییر تنها دوازده درصد از قواعد درخت بانک فارسی به بهبود ۲/۷ درصدی مطابق با شکل (۱۸) دست یافته‌ایم.

۶- نتیجه‌گیری و کارهای آینده

در این پژوهش عوامل مؤثر بر عملکرد تجزیه‌گر دستورهای مستقل از متن آماری زبان فارسی بررسی شد و نشان داده شد که استفاده از برچسب‌های ریز اجزای واژگانی کلام و تزریق وابستگی به دستورهای مستقل از متن از طریق نشانه‌گذاری می‌تواند موجب بهبود عملکرد این نوع تجزیه‌گر شود.

با بررسی نتایج حاصل از تجزیه‌گرهای زبان فارسی متوجه شدیم که به‌دلیل استفاده از قواعد دودویی در توسعهٔ درخت‌بانک فارسی، این تجزیه‌گرها در هنگام مواجه با تکواژ دستوری میانجی، اجزای ناهمگنی را جهت ترکیب با هم‌دیگر انتخاب می‌کنند که درنتیجه در این پژوهش راه‌حلی جهت رفع این مشکل پیشنهاد شد.

از نتایج به‌دست‌آمده در این پژوهش می‌توان برای تولید هر چه بهتر درخت‌بانک فارسی بهره برد، درخت‌بانک که پاسخ‌گوی وسعت دستور زبان فارسی باشد و در توسعه آن به نکات طراحی تجزیه‌گرها برای رسیدن به کارایی مناسب در تجزیه جملات توجه شود. به بعضی از این نکات در این مقاله پرداخته شد، اما موارد دیگر از جمله بررسی تأثیر وابستگی‌های لغوی در کارایی تجزیه‌گرها و همچنین طراحی تجزیه‌گری که بتواند به‌درستی افتادگی فاعل را در جملات تشخیص دهد نیاز به پژوهش‌های بیشتری دارد.

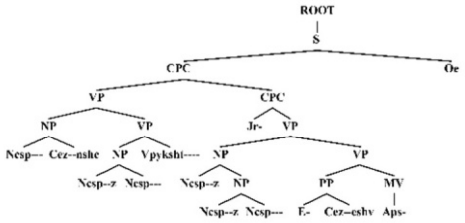
7- References ۷- مراجع

[1] J. E. Hopcroft, R. Motwani, and J. D. Ullman, "Automata theory, languages, and computation," *International Edition*, vol. 24, 2006.

[2] N. Chomsky, Syntactic structures. Walter de Gruyter, 2002.

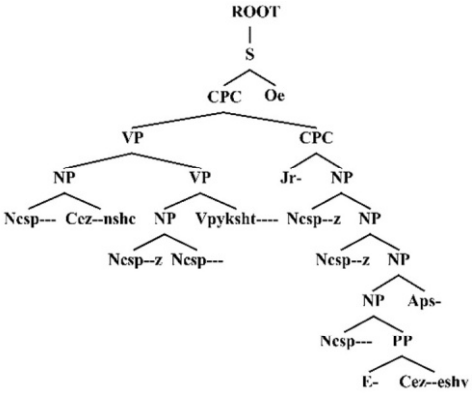
[3] E. Charniak and M. Johnson, "Coarse-to-fine n-best parsing and MaxEnt discriminative reranking," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*: *Association for Computational Linguistics*, pp. 173-180. 2005.

[4] S. Green and C. D. Manning, "Better Arabic parsing: Baselines, evaluations, and analysis," in



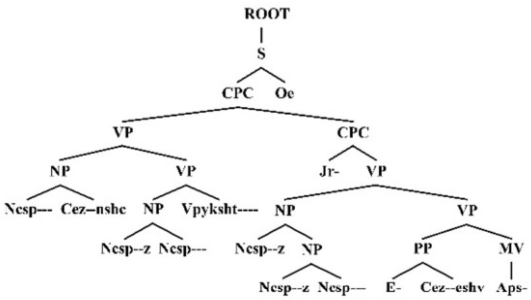
(شکل-۱۵): درخت تجزیه مرجع جمله «والدش کارگر راه‌آهن ... »

(Figure-15): A sample reference syntax tree



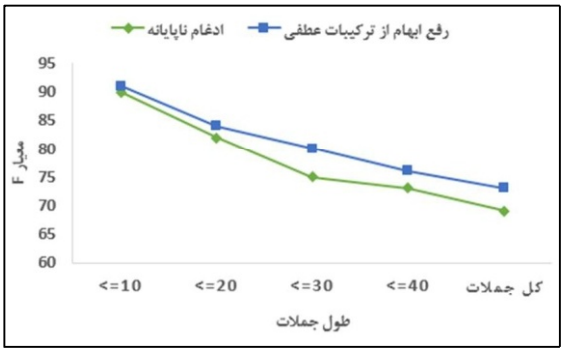
(شکل-۱۶): درخت تجزیه جمله «والدش کارگر راه‌آهن ... » توسط تجزیه‌گر پایه

(Figure-16): Parsed syntax tree of previous sample by the base parser



(شکل-۱۷): درخت تجزیه جمله «والدش کارگر راه‌آهن ... » توسط تجزیه‌گر مجهز به رفع ابهام عطفی

(Figure-17): Parsed syntax tree of previous sample by the refined conjunction constituent parser



(شکل-۱۸): نتایج ارزیابی تجزیه‌گر مجهز به رفع ابهام عطفی و تجزیه‌گر پایه

(Figure-18): Test results of the base parser and refined parser

[DOI: 10.29252/jsdp.16.3.36]

Proceedings of the 23rd International Conference on Computational Linguistics: Association for Computational Linguistics, pp. 394-402, 2010.

[5] M. Ghayoomi, "From Grammar Rule Extraction to Treebanking: A Bootstrapping Approach," in LREC, 2012, pp. 1912-1919.

[۶] م. رزازی، "پژوهش مستقل دانشگاه صنعتی امیرکبیر"، ۱۳۸۵.

[6] M. Razzazi, "Independent research at Amirkabir University Of Technology", 2006.

[۷] ا. استیری، م. کاهانی، ر. سعیدی و ا. عسگریان، "طراحی ابزار پارسر زبان فارسی"، کنفرانس بین المللی پردازش خط و زبان فارسی، ۱۳۹۱.

[7] A. Astiri, M. Kahani, R. Sacidi ,and A. Asgariyan, "Designing a parser for persian language",International Conference on persian language processing, 2012.

[8] H. Feili ,and G. Ghassem-Sani, "Unsupervised grammar induction using history based approach," *Computer Speech & Language*, vol. 20, no. 4, pp. 644-658, 2006.

[9] K. Lari and S. J. Young, "The estimation of stochastic context-free grammars using the inside-outside algorithm," *Computer Speech & Language*, vol. 4, no. 1, pp. 35-56, 1990.

[۱۰] ش. ع. پور، م. ه. پور و م. ب. ج. خان، "شناسایی محل کسره اضافه در زبان فارسی با استفاده از گرامر مستقل از متن احتمالاتی"، ارائه شده در سیزدهمین کنفرانس سالانه انجمن کامپیوتر ایران ۱۳۸۶.

[10] Sh.A. Poor, M.H. Poor ,and M. BijanKhan, "Identifying the location of the excess in Persian using PCFG", In *Procedings of 13th Conference of Computer Society of Iran*, 2008.

[۱۱] م. قیومی، "معرفی دادگان درختی و تجزیه‌گر خودکار فارسی "،ارائه شده در هشتمین همایش زبان‌شناسی ایران، تهران، دانشگاه علامه‌طباطبایی، ۱۳۹۲.

[11] M. Ghayoomi, "Persian Treebank and Autoamtion Parser", In *Procedings of Computational Lingustic of Iran*, 2013.

[12] M. Ghayoomi, "Word clustering for Persian statistical parsing," in Advances in Natural Language Processing: Springer, 2012, pp. 126-137.

[13] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, "Class-based n-gram models of natural language," *Computational linguistics*, vol. 18, no. 4, pp. 467-479, 1992.

[14] D. Jurafsky and J. H. Martin, Speech and Language Processing. Prentice Hall, Pearson Education International, 2014.

[15] T. L. Booth, "Probabilistic representation of formal languages," in *Switching and Automata Theory, 1969., IEEE Conference Record of 10th Annual Symposium on*, pp. 74-81, 1969.

[16] C. D. Manning and H. Schütze, Foundations of statistical natural language processing. MIT press, 1999.

[17] A. Bies et al., "Bracketing guidelines for treebank II style Penn Treebank project. Philadelphia: Linguistic Data Consortium," ed, 2013.

[18] C. Pollard, Head-driven phrase structure grammar, University of Chicago Press, 1994.

[۱۹] م. صادق‌زاده، م. رزازی و م. قیومی، "بررسی عوامل مؤثر بر عملکرد تجزیه گر آمارسی"، ارائه شده در سومین همایش زبانشناسی رایانشی، تهران، ۱۳۹۳.

[19] M. Sadeghzadeh, M.Razzazi and M. Ghayoomi, "Investigating effective factors on Persian Parser", In *Proceedings of the 3th Conference on Computatinal Lingustics*, Tehran, 2013.

[20] D. Klein and C. D. Manning, "A parsing: fast exact Viterbi parse selection," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*: *Association for Computational Linguistics*,vol.1, pp. 40-47, 2003.

[21] S. Bird, "NLTK: the natural language toolkit," in *Proceedings of the COLING/ACL on Interactive presentation sessions*: *Association for Computational Linguistics*, pp. 69-72, 2006.

[22] S. Abney and et al., "Procedure for quantitatively comparing the syntactic coverage of English grammars," in *Proceedings of the workshop on Speech and Natural Language: Association for Computational Linguistics*, pp. 306-311, 1991.

[23] K. Megerdoomian, "Developing a Persian part of speech tagger," in *Proceedings of the 1st Workshop on Persian Language and Computer*, , pp. 99-105, 2004.

[24] E. Rahimtoroghi, H. Faili, and A. Shakery, "A structural rule-based stemmer for Persian," in *Telecommunications (IST), 2010 5th International Symposium on, 2010: IEEE*, pp. 574-578, 2010.

[25] M. Mohseni and B. Minaei-Bidgoli, "A Persian Part-Of-Speech Tagger Based on Morphological Analysis," in *LREC*, 2010.

[26] M. Johnson, "The effect of alternative tree representations on tree bank grammars," in *Proceedings of the Joint Conferences on New Methods in Language Processing and Computa-*

[DOI: 10.29252/jsdp.16.3.36]

tional Natural Language Learning: *Association for Computational Linguis-tics*, pp. 39-48, 1998.

[۲۷] م. صادقزاده, م. رزازی و ح. محمودی, "تزریق وابستگی ساختاری به دستورهای مستقل از متن آماری زبان فارسی از طریق نشانه‌گذاری قواعد "ارائه شده در بیستمین کنفرانس ملی سالانه انجمن کامپیوتر ایران, مشهد, ۱۳۹۳.

[27] M. Sadeghzadeh, M. Razzazi ,and H. Mahmoodi, " Injecting Structural Dependency into Persian PCFG", In *Proceedings of 20th Conference of Computer Society of Iran*, 2013.

[28] M. Collins, "Head-driven statistical models for natural language parsing," *Computational linguistics*, vol. 29, no. 4, pp. 589-637, 2003.

محمدباقر صادق‌زاده دانشجوی دکترای مهندسی کامپیوتر با گرایش نرم‌افزار در دانشگاه صنعتی امیرکبیر است. زمینه‌های تخصصی مورد علاقه ایشان هندسه محاسباتی و هوش مصنوعی است. پردازش زبان طبیعی و تجزیه‌گرهای زبان فارسی از جمله زمینه‌های پژوهشی ایشان در دوره کارشناسی ارشد بوده است.

نشانی رایانامه ایشان عبارت است از:

mbs91@aut.ac.ir

محمدرضا رزازی عضو هیأت علمی دانشکده مهندسی کامپیوتر در دانشگاه صنعتی امیرکبیر است. ایشان مدرک دکترای خود را در رشته نرم‌افزار در سال ۱۳۶۸ از دانشگاه سانتاباربارا اخذ کرده‌اند. از جمله زمینه‌های پژوهشی ایشان می‌توان به هندسه محاسباتی، پردازش زبان طبیعی و پژوهش و توسعه نرم‌افزارهای پزشکی اشاره کرد.

نشانی رایانامه ایشان عبارت است از:

razzazi@aut.ac.ir

مسعود قیومی عضو هیأت علمی پژوهشگاه علوم انسانی و مطالعات فرهنگی است. وی فارغ‌التحصیل مقطع دکترای رایانه با گرایش زبان‌شناسی رایانشی در سال ۲۰۱۴ از دانشگاه آزاد برلین آلمان است. وی در سال ۲۰۰۹ از دانشگاه سارلند آلمان و در سال ۲۰۰۸ از دانشگاه نانسی۲ فرانسه موفق به اخذ مدرک کارشناسی ارشد در رشته زبان‌شناسی رایانشی شد. همچنین وی در سال ۱۳۸۳ دوره کارشناسی ارشد خود را در رشته زبان‌شناسی همگانی در دانشگاه آزاد واحد تهران مرکز به پایان رساند. وی در سال ۱۳۸۰ در رشته مترجمی زبان انگلیسی از دانشگاه آزاد اسلامی قم فارغ‌التحصیل شد. زمینه‌های تخصصی مورد علاقهٔ ایشان پردازش زبان طبیعی، مدل‌سازی زبانی، نحو و معناشناسی واژگانی است.

نشانی رایانامه ایشان عبارت است از:

m.ghayoomi@ihcs.ac.ir

[DOI: 10.29252/jsdp.16.3.36]