Digital Islamic Studies and Humanities

www.disah.inoor.ir

Noor intelligent Arabic word stemmer engine

Seyyed Mohsen Hashemi [1]✉

1. Programmer engineer of intelligent processing group of computer research center of Islamic sciences: m-hashemi@noornet.net

Article Info

Article type:
Research Article

Article history:
Received: 12 January 2025
Received in revised form: 19 February 2025
Accepted: 9 April 2025
Available online: 23 August 2025

Keywords:
Arabic stemmer,
stemming engine,
stemmer engine,
Arabic stemming.

ABSTRACT

The term "stemmer" refers to an algorithm used in Natural Language Processing (NLP) for morphological analysis, aimed at extracting and representing the root or base form of words. In other words, stemming is achieved by removing prefixes, suffixes, and vowels from conjugated verbs, derived nouns, and other word forms. The purpose of an Arabic stemmer is to reduce each word to its base form while preserving its semantic identity and syntactic function, thus facilitating efficient indexing, searching, and categorization of large Arabic text corpora. This tool plays a vital role in information retrieval, document classification, machine translation, summarization, and question-answering systems .

A unique feature of the intelligent Arabic stemmer is its simultaneous use of rule-based, data-driven, and learning-based methods. As a result, the performance of this engine surpasses that of other stemming engines at a significant level. This engine utilizes Arabic text data from both classical and modern sources, combined with intelligent morphological analyses and rule-based refinement techniques grounded in the structured nature of the Arabic language. The integration of these three methods is highly effective in resolving ambiguities in words that do not conform neatly to Arabic language rules, contributing to its overall efficiency. This capability has allowed it to overcome some of the challenges faced by other stemming engines .

Furthermore, this engine has been compared and evaluated using human-generated stems provided by experts, and its accuracy, compared to other stemmers, is of a very high quality. Another valuable feature of this engine is its ability to offer stems at different levels, which is a new and highly beneficial capability that can address the needs of a wide range of users.

Cite this article: Hashemi, S.M. (2025). Noor intelligent Arabic word stemmer engine. *Digital Islamic Studies and Humanities*, 1 (1), 179-224.
https://doi.org/10.22034/disah.2024.716144

 Publisher: Research Center for Digital Islamic Studies and Humanities (RCDISAH).
DOI: https://doi.org/10.22034/disah.2024.716144

علوم اسلامی و انسانی دیجیتال

www.disah.inoor.ir

موتور پیراسته‌ساز هوشمند کلمات عربی نور

سید محسن هاشمی[۱] ✉

۱. مهندس برنامه‌نویس گروه پردازش هوشمند مرکز، پژوهشگر پژوهشکده علوم اسلامی و انسانی دیجیتال، رایانامه: m-hashemi@noornet.net

اطلاعات مقاله

نوع مقاله:
مقاله پژوهشی

تاریخ دریافت: ۱۴۰۳/۱۰/۲۳
تاریخ بازنگری: ۱۴۰۳/۱۲/۰۱
تاریخ پذیرش: ۱۴۰۴/۰۱/۲۰
تاریخ انتشار: ۱۴۰۴/۰۶/۰۱

کلیدواژه‌ها:
پیراسته‌ساز عربی،
موتور استمر،
موتور پیراسته‌ساز،
استمر عربی.

چکیده

استمر به الگوریتمی اطلاق می‌شود که در پردازش زبان طبیعی (NLP) و تحلیل ریخت‌شناسی برای استخراج و نمایش شکل اصلی یا ریشه کلمات به کار می‌رود. به‌عبارت دیگر، استمر با حذف پیشوندها، پسوندها و حروف میان‌وند، افعال صرف‌شده، اسم‌های مشتق و سایر اشکال کلمات را به حالت پایه‌شان باز می‌گرداند. هدف از استمر عربی این است که با حفظ هویت معنایی و نقشی کلمات، آن‌ها را به شکل پایه‌شان تبدیل کند و بدین‌ترتیب فهرست‌بندی، جستجو و دسته‌بندی مجموعه‌های بزرگ اسناد عربی را تسهیل کند. این ابزار نقش اساسی در بازیابی اطلاعات، طبقه‌بندی اسناد، ترجمه ماشینی، خلاصه‌سازی و سیستم‌های پاسخگویی به سوالات دارد. ویژگی متمایز استمر هوشمند عربی، بهره‌گیری هم‌زمان از سه روش قاعده‌محور (Rule-Based)، آماری (Data-driven) و یادگیری‌محور (Learning-Based) است. به همین دلیل، عملکرد این موتور به‌مراتب از سایر موتورهای استمر فراتر رفته است. این موتور از متون کهن و جدید عربی، همراه با تحلیل‌های صرفی هوشمند و تکنیک‌های پالایش بر اساس ساختار قواعد زبان عربی بهره می‌گیرد. تلفیق این سه روش در جهت رفع ابهام از کلماتی که با ساختار استاندارد زبان عربی تطابق ندارند، کارآمدی آن را به‌طور قابل توجهی افزایش می‌دهد و به همین دلیل، چالش‌هایی که دیگر موتورهای استمر با آن‌ها مواجه‌اند، در این موتور برطرف شده است. این موتور با استم‌های انسانی که توسط متخصصان ارائه شده، مقایسه و ارزیابی شده است و نتایج نشان می‌دهد که در مقایسه با سایر استمرها از سطح کیفی بالاتری برخوردار است. از دیگر ویژگی‌های باارزش این موتور، امکان ارائه استم‌ها در سطوح مختلف است که این قابلیت نوین، نیازهای مختلف کاربران را به‌خوبی پوشش می‌دهد.

استناد: هاشمی، سید محسن (۱۴۰۴). موتور پیراسته‌ساز هوشمند کلمات عربی نور. علوم انسانی و اسلامی دیجیتال، ۱ (۱)، ۱۷۹-۲۲۴. https://doi.org/10.22034/disah.2024.716144

ناشر: پژوهشکده علوم اسلامی و انسانی دیجیتال (مرکز تحقیقات کامپیوتری علوم اسلامی نور).

مقدمه

زبان عربی متعلق به خانواده زبان‌های سامی است و به‌طور گسترده در بسیاری از نقاط جهان، به‌ویژه در خاورمیانه و شمال آفریقا صحبت می‌شود. این زبان ساختار دستوری پیچیده‌ای دارد که آن را از سایر زبان‌ها متمایز می‌کند. واحد اصلی معنی در زبان عربی «ریشه» یا «ریشه سه‌حرفی» است که از سه حرف صامت تشکیل شده و به‌عنوان پایه‌ای برای تشکیل کلمات جدید عمل می‌کند. به‌عنوان مثال، ریشه K-T-B (کتب) به معنای «نوشتن» و ریشه S-L-M (سلم) به معنای «امن/آرامش» است. این ریشه‌ها را می‌توان با حروف صدادار و الگوهای مختلفی که به نام «صیغه‌ها» شناخته می‌شوند ترکیب کرد و کلمات جدید ایجاد نمود. این ساختار به تعداد زیادی مشتق از هر ریشه اجازه می‌دهد که زبان عربی را به یک زبان بسیار متمایز تبدیل کند.

با توجه به پیچیدگی این ساختار زبانی، پیراسته‌سازی برای کارهایی مانند خلاصه‌سازی متن، نمایه‌سازی موتور جستجو و بازیابی اطلاعات ضروری است، جایی که شناسایی معانی اصلی کلمات بسیار مهم است. در این زمینه، طراحی یک موتور پیراسته‌ساز قوی که بتواند روابط معنایی بین کلمات مرتبط را به‌طور مؤثر استخراج کند، یک حوزه تحقیقاتی مهم به شمار می‌رود.

هدف این مقاله ارائه چنین الگوریتمی است. در نهایت، هدف ما کمک به توسعه ابزارهای کارآمدتر و دقیق‌تر است تا راه را برای دسترسی بهتر به متون و منابع عربی در پلتفرم‌ها و برنامه‌های مختلف هموار کند.

الف. تاریخچه موتورهای پیراسته‌ساز عربی

مفهوم پیراسته‌ساز[1] یا تقلیل کلمات به اشکال پایه، در زمینه زبان‌شناسی محاسباتی در اواخر قرن بیستم به‌وجود آمد. هدف آن تسهیل جستجو، نمایه‌سازی و دسته‌بندی اسناد، از طریق شناسایی کلماتی بود که با وجود اشکال سطحی متفاوت، معانی مشابهی دارند. الگوریتم‌های پایه اولیه عمدتاً بر زبان انگلیسی متمرکز بودند، زیرا ساختار ریخت‌شناسی نسبتاً ساده‌تری در مقایسه با زبان‌های دیگر مانند عربی داشتند.

1. Stemming.

با این‌حال، با تکامل فناوری رایانه، تقاضا برای تجزیه و تحلیل خودکار زبان‌های غیراروپایی، از جمله عربی، افزایش یافت. محققان شروع به بررسی راه‌هایی برای انطباق روش‌های پیراسته‌سازی سنتی با چالش‌های منحصربه‌فرد ناشی از ریخت‌شناسی پیچیده زبان عربی کردند. برخی از تلاش‌های اولیه شامل قواعد دست‌ساز بود که به شدت بر دانش بشری دستور زبان عربی متکی بود، اما این قواعد از نظر دامنه محدود و مستعد خطا بودند.

با ظهور یادگیری ماشینی و تکنیک‌های آماری پردازش زبان طبیعی، پیشرفت‌های قابل‌توجهی حاصل شد. به‌جای تکیه صرف بر قوانین از پیش تعریف‌شده، محققان توانستند مدل‌هایی را با استفاده از مجموعه‌های بزرگی از داده‌ها و روابط بین کلمات و معانی زیربنایی آن‌ها ایجاد کنند. این رویکردها بسیار متنوع‌تر و انعطاف‌پذیرتر از سیستم‌های مبتنی بر قاعده بودند و به آن‌ها اجازه می‌دادند تا طیف وسیع‌تری از پدیده‌ها، مانند مشتقات، اعلام و کلمات مرکب را مدیریت کنند.

با گذشت زمان، چندین نسل از موتورهای پیراسته‌ساز عربی ظهور کردند که هرکدام بر اساس کارهای قبلی توسعه یافته و عملکرد بهتری داشتند. برخی از نمونه‌های محبوب شامل Porter stemmer برای انگلیسی، کتابخانه Snowball برای چندین زبان اروپایی، و Regexp stemmer برای چینی هستند.

در سال‌های اخیر، به دلیل نیاز روزافزون به ابزارهای کارآمد و دقیق برای پردازش حجم زیادی از متن عربی، مطالعه و توسعه پیراسته‌ساز عربی مورد توجه قرار گرفته است. پیراسته‌ساز جزء ضروری هر ابزار پردازش زبان طبیعی است، زیرا ما را قادر می‌سازد کلمات را به شکل پایه یا متعارف خود، که به‌عنوان ریشه یا لمَّا (lemmas) شناخته می‌شود، کاهش دهیم. این امر به ما کمک می‌کند معانی و روابط کلمات را در یک بافت زبانی معین بهتر درک کنیم.

در عصر دیجیتال امروز، که در آن روزانه مقادیر زیادی محتوای عربی در زمینه‌های مختلف تولید می‌شود، موتورهای پیراسته‌ساز نقش‌های حیاتی فزاینده‌ای در برنامه‌های کاربردی متعددی از جمله بهینه‌سازی موتور جستجو(SEO) ، بازیابی اطلاعات(IR) ، مدل‌سازی، طبقه‌بندی متن، تحلیل داده و ترجمه ماشینی ایفا می‌کنند.

در این مقاله به پنج مورد از موتورهای پیراسته‌ساز برجسته عربی که در حال حاضر در محیط‌های مختلف دانشگاهی و صنعتی به کار می‌روند، پرداخته شده و ویژگی‌های متمایز، مزایا، محدودیت‌ها و زمینه‌های کاربردی آن‌ها به‌طور مختصر ارائه شده است.

۱. The CLAWS (Concordance Lexical Analysis System) Lemmatizer: این ابزار توسط دانشگاه لنکستر ارائه شده است و با هدف پیراسته‌سازی (لِمَّا) کلمات عربی طراحی شده است. این ابزار کلمات را به ساختار فرهنگ لغتی کاهش می‌دهد و صرفاً ریشه کلمات را ارائه نمی‌دهد. این مدل از ترکیبی از روش‌های آماری و مبتنی بر قوانین (Rules) برای دستیابی به شکل پیراسته کلمات با دقت و انعطاف‌پذیری بالا استفاده می‌کند و برای انواع استاندارد زبان عربی مدرن و کلاسیک مناسب است.

۲. The Al-Manhal Stemmer: این موتور توسط دانشگاه نفت و مواد معدنی ملک فهد (KFUPM[1])ارائه شده است و به‌طور خاص بر عربی استاندارد مدرن تمرکز دارد. این مدل از یک رویکرد خودکار متناهی برای رسیدگی به ساختارهای صرفی رایج عربی مانند افعال، اسم‌ها، صفت‌ها و قیدها استفاده می‌کند و از نظر دقت، کارآیی و سرعت، عملکرد بالایی دارد.

۳. TALP Stemmer: این مدل در ابتدا توسط دانشگاه کارنگی ملون[2] برای زبان اسپانیایی و پرتغالی ساخته شد و سپس توسط محققان دانشگاه Autònoma de Barcelona برای زبان عربی توسعه یافت. این مدل با استفاده از یک معماری شبکه عصبی، می‌تواند الگوهای ظریف‌تر و وابستگی‌های زمینه‌ای را به‌خوبی بازنمایی کند و عملکردی برتر نسبت به سیستم‌های مبتنی بر قوانین سنتی دارد.

۴. Pharos Stemmer: طراحی شده توسط مرکز ملی پردازش زبان طبیعی در آکادمی تحقیقات علمی و فناوری مصر. این stemmer از انواع تکنیک‌ها از جمله برنامه‌نویسی پویا، درخت تصمیم‌گیری و مدل‌های بُرداری بهره می‌گیرد. این موتور گزینه‌هایی برای تنظیم پارامترها ارائه می‌دهد که تعادل بین سرعت و دقت را فراهم کرده و آن را برای برنامه‌های کاربردی دنیای واقعی مناسب می‌سازد.

۵. LTR Stemmer: این ابزار توسط محققان دانشگاه ژنو توسعه یافته و از روش جدید "طولانی‌ترین دنباله تکراری" برای استخراج تکواژهای معنی‌دار از کلمات عربی استفاده می‌کند. این مدل در تجزیه و تحلیل متون تاریخی نتایج بسیار خوبی داشته است.

1- King Fahd University of Petroleum and Minerals.

2- Carnegie Mellon University.

مدل‌های مذکور عمدتاً رویکرد آکادمیک داشته‌اند. با این حال، با ارتقا و گسترش سیستم‌های بازیابی اطلاعات از منابع متنی، پروژه‌های متعددی در ساخت و توسعه استمرهای عربی پیش‌قدم بوده‌اند و هر یک با انتخاب یکی از رویکردهای مطرح در این عرصه، سعی در ارائه نتایج مطلوب داشته‌اند. از جمله این رویکردها، رویکرد مبتنی بر ریشه است که تمرکز پیشگامان حوزه متن‌کاوی در زبان عربی نیز بر این رویکرد بوده است (Al-Fedaghi and Al-Anzi; 1989). رویکرد ریشه در واقع یک تحلیل صرفی از کلمات عربی جهت دستیابی به ریشه است.

استمر Khoja (الخوجة) که با تکیه بر یک داده لغتنامه‌ای طراحی شده، با ارتقای خود در سال‌های پس از آن، به یکی از شناخته‌شده‌ترین استمرهای این رویکرد تبدیل شد. پس از آن، استمرهای Taghva et al. (2005) و ISRI (2005) نیز بدون تکیه بر هیچ لغتنامه‌ای، نتایجی مشابه Khoja داشتند. در سال‌های اخیر نیز استمرهای جدیدی همچون Alkabi et al (2015) با رویکرد ریشه مورد آزمایش قرار گرفته‌اند و ادعای ۷۵٪ پاسخ صحیح را دارند. همچنین، Al-Shalabi et al (2007) استمری جهت دستیابی به ریشه بر پایه حذف «سألتمونیها» طراحی کرده‌اند که دارای عملکردی متفاوت نسبت به دیگر استمرها بوده و میزان اثربخشی آن را ۹۵ درصد اعلام کرده‌اند؛ اما نتایج پروژه آن‌ها جهت ارزیابی در معرض استفاده عمومی و برخط قرار نگرفته است.

۶. CAMeL Tools: The CAMeL Tools Stemmer مجموعه‌ای از ابزارهای متن‌باز برای پردازش زبان طبیعی عربی است که توسط آزمایشگاه CAMeL در دانشگاه نیویورک ابوظبی توسعه یافته است. استمر CAMeL یکی از اجزای این مجموعه است که برای ریشه‌یابی متن عربی استفاده می‌شود و شامل کاهش کلمات به شکل ریشه یا شکل پایه آن‌ها است. این استمر بر اساس دیتای طلایی استم شده، میانگین دقت ۳۷ درصد را ارائه کرده است.

استمر CAMeL از قوانین و الگوریتم‌های زبانی برای شناسایی و استخراج ریشه یا شکل پایه کلمات عربی استفاده می‌کند. این ابزار با تجزیه و تحلیل کلمات و حذف پیشوندها و پسوندهای احتمالی بر اساس ریخت‌شناسی زبان عربی، شکل پایه کلمات را شناسایی می‌کند.

مهم‌ترین مشکل استمرهایی با رویکرد فوق، ارتباط کلمات ذیل یک عنوان کلی «ریشه» است که گاه مشتقات آن ریشه از هزار کلمه غیرتکراری نیز تجاوز می‌کند. کثرت کلمات در یک رشته

مرتبط ذیل یک ریشه، موجب کاهش شدید عملکرد سیستم‌های بازیابی اطلاعات شده (Ababneh et al, 2012: 371) و محققین این عرصه را به سمت طراحی استمرهای سبک و تولید استم واقعی سوق داده است. از آنجا که رویکرد استمر پیشنهادی در این مقاله نیز مبتنی بر تولید استم صحیح و واقعی از کلمات عربی است (و نه صرفاً مبتنی بر ریشه)، تنها به معرفی موتورهای هوشمند بر اساس این رویکرد اکتفا می‌شود.

نکته حائز اهمیت در این بحث، نگرش عملی و آزمایشگاهی به عملکرد هر استمر، فارغ از نظریه‌پردازی‌های صرف است. تحقیقات حاکی از وجود نمونه‌های متعددی از الگوریتم‌های ساخت استم عربی و اثربخشی آن‌ها بوده و اکثر آن‌ها نیز با ادعای صحت بیش از ۸۵٪ سعی در معرفی خود به‌عنوان یک استمر با ویژگی‌های خاص داشته‌اند؛ ولی به دلیل عدم دسترسی به کدهای منبع، دادهٔ ارزیابی، الگوریتم معیار ارزیابی عملکرد و عدم عرضه برخط، بررسی صحت ادعای خود را ناتمام گذاشته‌اند. از این رو در ادامه بر تحقیقاتی تاکید شده است که دارای خروجی واقعی بوده و محققین نتایج آن‌ها را مورد مقایسه قرار داده‌اند. بر همین اساس، یکی از نکات قوت این نوشتار، ارائه خروجی کاربردی است که کاربر را در معرض آزمایش و بررسی نتایج قرار داده و امکان استفاده برخط را نیز برای او فراهم آورده است.

استمر لارکی (Larkey et al, 2002) از پیشگامان طراحی و تولید استمرهای سبک با رویکرد ساخت استم واقعی از کلمات عربی بوده و نتایج حاصل از این رویکرد را بسیار اثربخش‌تر از رویکرد ریشه بیان داشتند. آن‌ها با ارتقای استمر خود در سال ۲۰۰۷، مقاله‌ای مستقل در این زمینه منتشر کردند (Larkey et al, 2007) و با استفاده از داده‌های استاندارد TREC[1]، اثربخشی استمرهای مختلف را در بازیابی اطلاعات متنی مورد ارزیابی قرار دادند. از میان آن‌ها، استمر پیشنهادی Light10 از روش‌های دیگر بهتر بوده و در بازیابی اطلاعات متون عربی نیز بسیار مورد توجه محققین قرار گرفته است. Aljlayl and Frieder (2002) نیز در مقاله خود تأثیر استم را در بهبود سیستم‌های بازیابی اطلاعات عربی مورد مطالعه قرار دادند. آن‌ها با پیشنهاد دو مدل استمر: الگوریتم ریشه‌ای مبتنی بر کار Khoja و الگوریتم استمر سبک (Light Stemming (LS) Algorithm)، تأیید کردند که دقت الگوریتم LS به‌طور قابل توجهی از دقت الگوریتم ریشه در IR پیشی می‌گیرد.

1. Text REtrieval Conference.

پردازش استمرهای سبک بر پایه جداسازی تعداد محدودی از پیشوندها و پسوندها بدون استفاده از قواعد مربوط به ساختارهای ساخت کلمه است که گاه از آن به‌عنوان استم‌سازی «Elementary»یا «Shallow» نیز یاد می‌شود. به‌عنوان مثال، در استمر سبک Chen and Gey (2002)، علاوه بر تهیه لیستی از کلمات ثابت (Stop Words)، تنها به حذف پیشوند «ال» و حذف چهار پسوند «ات/ان/ون/ه» به‌عنوان وندهای پرکاربرد در زبان عربی اکتفا شد و نتایج مطلوبی را ارائه کرد. در استمر Rogati et al. (2003)، یک روش یادگیری بدون نظارت برای استمر سبک عربی ارائه شده است. نویسندگان نتایج حاصل از استمر خود را با دادهٔ طلایی خودشان مقایسه کرده و ادعا نمودند که خروجی آن‌ها در مقایسه با این داده ۵۰٪ تطابق دارد.

از دیگر استمرهای سبک می‌توان به الگوی مورد پیشنهادی Darwish (2002)و همچنین استمر پیشنهادی(2010) Saad and Ashour با نام استمر Motaz اشاره کرد که برای بررسی تأثیر پردازش متن بر طبقه‌بندی موضوعی[1] متن‌های عربی پیشنهاد شده است. این سیستم در ابزارهای WEKAو RapidMiner نیز ادغام شده است.

موتور تحلیل‌گر صرفی MADAMIRA که توسط محققین بخش NLP دانشگاه کلمبیا در سال‌های اخیر طراحی و معرفی گردید(Pasha, 2014) ، ترکیبی از دو موتور هوشمند قبلی MADAو Amira است که توانسته با تکیه بر متون عربی معاصر به نتایج قابل‌ملاحظه‌ای در تجزیه صرفی کلمات دست یابد. این موتور هوشمند برای بررسی نتایج به‌صورت برخط در اختیار محققین قرار گرفته است.

موتور هوشمند دیگری که در حال حاضر به‌صورت برخط و آفلاین در اختیار محققین قرار دارد، تحلیل‌گر هوشمند SAFARاست. در داخل این تحلیل‌گر، بخشی مستقل با عنوان SAFAR-Stemmerقرار گرفته و نتایج خود را در مقایسه با پنج استمر دیگر یعنی Khoja، ISRI، Motaz، Light10 و استمر برخط Tashaphyne، در معرض بررسی و تحقیق قرار داده است. مطابق با دادهٔ ارزیابی واحدی که مورد پردازش این ۶ استمر قرار گرفته است، درصد صحت نتایج استمر SAFAR بیش از بقیه بوده و در مورد کلمات قرآن نیز با ارزیابی نسبت به دادهٔ معیار قرارداده‌شده بر پایگاه corpus.quran.com، استمر SAFAR با ۳۳٫۷٪ بالاترین درصد پاسخ صحیح را به خود اختصاص داده است (Jaafar et al, 2017: 169).

1. Classification.

لازم به ذکر است که تلاش برخی محققان در راستای طراحی استمرهایی برای شناسایی گویش‌های مختلف عربی در زبان محاوره‌ای بوده است؛ مانند الگوی پیشنهادی AbuAta and Al-Omari (2014)که با تمرکز بر عربی محاوره‌ای کشورهای حوزه خلیج فارس (عربی خلیجی) طراحی شده است. از آنجا که استمر پیشنهادی این مقاله یعنی استمر «نور» متکی بر عربی رسمی است، تنها به معرفی استمرهای فعال در حوزه عربی رسمی بین‌المللی به‌ویژه عربی کلاسیک اکتفا شده است.

ب. چالش‌های موتورهای پیراسته‌ساز در زبان عربی

استمرها به دلیل توانایی آن‌ها در کاهش کلمات به شکل پایه یا متعارف خود، به نام پیراسته، ریشه یا لِمَّا، به ابزارهای ضروری در پردازش زبان طبیعی (NLP) تبدیل شده‌اند که درک و شناسایی رابطه را در یک بافت زبانی خاص تسهیل می‌کنند. با این حال، استفاده از استمرها در زبان عربی چالش‌های منحصر به فردی را در مقایسه با زبان‌های دیگر مانند انگلیسی ایجاد می‌کند. بخشی از این چالش‌ها عبارتند از:

۱. **ریخت‌شناسی پیچیده:** عربی، زبانی بسیار عطفی است که برای تغییر معنا به شدت به ضمائم و اصلاحات درونی متکی است. کلمات را می‌توان با توجه به گرامر، زمان، عدد، جنسیت و حروف به چندین شکل صرف کرد. در نتیجه، یافتن استم مناسب نیاز به دانش پیچیده‌ای از ساختار ریخت‌شناسی و معناشناسی دارد.

۲. **حذف نادرست:** کلماتی که حروف اصلی آغازین آن‌ها مشابه برخی پیشوندها بوده یا به حروفی مشابه با برخی پسوندها ختم می‌شوند، در معرض حذف نادرست حروف اصلی هستند. به‌عنوان مثال، تجزیه کلمه «والده» (پدر او) با موتور استمر Light10 (Larkey et al, 2007) و در نظر گرفتن «و + ال» به‌عنوان پیشوند و «ه» به‌عنوان پسوند، به استم نادرست «د» منجر خواهد شد؛ درحالی‌که «وال» جزئی از استم صحیح والد است. همچنین است در مورد کلمه «ألم»، «فتح» و دیگر موارد مشابه.

۳. **اشکال افعال بی‌قاعده:** افعال عربی بی‌نظمی قابل توجهی را نشان می‌دهند، به‌ویژه در زمان‌های گذشته و ناقص، که مشکلات قابل‌توجهی را برای استمرهایی که صرفاً بر قوانین ثابت تکیه می‌کنند، ایجاد می‌کند. این بی‌نظمی‌ها به الگوریتم‌های پیچیده‌ای نیاز دارند تا بتوانند استثناها و ابهاماتی را که در طول پیراسته‌سازی با آن‌ها مواجه می‌شوند، مدیریت کنند.

۴. ابهام ناشی از اعراب[1]: حذف حروف آغازین و پایانی بدون در نظر گرفتن اعراب آن‌ها منجر به ساخت استم نادرست خواهد شد. به‌عنوان مثال، تجزیه صرفی کلمه «بکر» بدون در نظر گرفتن اعراب، منجر به ساخت استم‌های «کر» و «ر» خواهد شد؛ درحالی‌که چه بسا اعراب کلمه به‌صورت «بَکر» بوده و حرف «ب» جزء اصلی استم باشد. همچنین است تجزیه کلمه «فحسن» که ممکن است با در نظر گرفتن اعراب، هر یک از استم‌های «حُسن» (مصدر)، «حَسَنَ» (فعل ماضی)، «حَسَنٌ» (اسم معرب) صحیح باشد.

۵. تک‌جوابی بودن: بسیاری از موتورهای استمر عربی موجود، تنها یک راه‌حل را در خروجی استم خود ارائه می‌دهند؛ درحالی‌که قواعد دستوری تجزیه کلمات عربی، وجود بیش از یک استم را تصدیق می‌کنند. به‌عنوان مثال، کلمه «لهم» می‌تواند نماینده چهار استم مختلف باشد: فعل «لَهَمَ»، اسم «لَهِمٌ» و فعل «هَمّ» (ل به‌عنوان پیشوند) و ضمیر «هُم». عدم توجه به استم صحیح می‌تواند این کلمه پر استفاده را به رشته کلمات نامرتبط متصل کند.(Jaafar et al, 2017: 164-165)

نکته‌ای که برخی نویسندگان بر آن دقت کافی نداشته‌اند، نقطه مقابل چالش فوق یعنی معایب چندجوابی بودن استم است. چه بسا اگر برای یک کلمهٔ عربی، تمامی استم‌ها بر اساس منطق صرفی استخراج شده و بدون پالایش و رتبه‌بندی، در فرآیندهای هوشمندسازی همچون لایه‌های نحوی و معنایی به کار رود، نتیجه‌ای جز آشفتگی دوچندان نسبت به حالت تک‌جوابی نخواهد داشت. به‌عنوان مثال، هر یک از کلمات «للبنات» و «لبناتک» بر اساس منطق صرفی می‌توانند دارای دو استم «لبن» و «بنت» باشند؛ ولی با بررسی استعمال «للبنات» و «لبناتک» متوجه می‌شویم که کلمات فوق بر اساس استم «لبن» یا اصلاً استعمال ندارند یا به قدری کم هستند که به نوعی نامتعارف محسوب می‌شوند.

مثال دیگر، کلمه قرآنی «لمسنا» (جن: ۸) است که در حالت بدون اعراب می‌توان سه استم منطقی «لمس»، «مس» (از ریشه مسس) و «مسن» (از ریشه سنن) را برای آن در نظر گرفت؛ درحالی‌که کلمه «لمسنا» بر پایه استم «مسن» به هیچ‌وجه استعمال ندارد. مثال دیگر نیز کلمه قرآنی «سندس» (کهف: ۳۱) و دو استم منطقی «سندس» و «ندس» برای آن است که استم دوم در متون منابع اسلامی استعمالی ندارد. با این اوصاف، چالش مطرح‌شده در تعدد استم، مسئله‌ای است که هیچ‌یک از طراحان موتور استمر به آن توجه نداشته‌اند.

1. Diacritic Marks.

۶. **تغییر کلمات:** علم صرف در زبان عربی دارای قواعد پیچیده‌ای با عنوان اعلال، ادغام و تخفیف است که گاه به هنگام تصریف کلمات عربی و ساخت قالب‌های جدید، موجب تغییر کامل کلمه و تبدیل آن به یک ساختار متفاوت از اصل خود می‌شود. به‌عنوان مثال، فعل ماضی «ضرب» با صرف صیغه‌های مخاطب تبدیل به «ضربت»، «ضربتما» و... می‌شود که موتورهای ساده استمر با حذف پسوند «ت» و «تما» آن را به اصل خود یعنی «ضرب» باز می‌گردانند. اما فعل ماضی «قال» از ریشه «قول» وقتی در صیغه‌های مخاطب صرف می‌شود، ساختارش کاملاً تغییر می‌کند و به واسطه قواعد اعلال، تبدیل به «قُلتَ»، «قُلتما» و... می‌شود. استمرهای سبک با حذف پسوند «ت» و «تما»، نهایت این کلمات را به استم «قل» (فعل امر از «قول» یا فعل ماضی از «قلل») بازمی‌گردانند که دارای ماهیتی متفاوت از «قال» فعل ماضی است.

مثال دیگر کلمه «ذَرَرتُ» است که با حذف پسوند توسط موتورهای استمر عادی به کلمه «ذرر» تبدیل می‌شود که فاقد معنا در زبان عربی است و فقط حاکی از حروف ریشه است؛ درحالی‌که استم صحیح آن «ذرَّ» مشدد است.

۷. **ساختارهای بی‌قاعده:** برخی از تغییرات ساختاری کلمه در زبان عربی فاقد یک قاعده مشخص بوده و اصطلاحاً به آن‌ها ساختار سماعی گفته می‌شود. نمونه بارز این تغییرات را می‌توان در جمع بستن غالب کلمات عربی مشاهده نمود. جمع در زبان عربی به دو دسته کلی جمع سالم و مکسر تقسیم می‌شود. در جمع سالم، حروف مشخص «ین، ون، ات» به کلمات اضافه شده و با اندک تغییراتی جمع این کلمات ساخته می‌شود. به‌عنوان مثال، «عالمون» و «عالمات» که جمع «عالم» و «عالمة» هستند. موتورهای سبک استمر با حذف «ون» و «ات» می‌توانند به استم این کلمات دست یابند؛ اما در جمع مکسر، ساختار کلمه کاملاً دگرگون شده و موتورهای استمر غالباً عاجز از تشخیص استم صحیح هستند (Y. Dahab, 2015: 39). به‌عنوان مثال، کلمه «رِجال» و «مفاتیح» که جمع مکسر «رَجُل» و «مِفتاح» هستند، هیچ حرفی از آن‌ها قابل حذف نیست تا بتوان با حذف آن‌ها به مفردشان دست یافت. از این رو، نیازمند طراحی یک استمر با قابلیت هوشمندسازی ویژه هستند.

برخی نویسندگان عدم بازگشت مضارع و امر به ماضی در موتورهای استم عربی را نیز به‌عنوان یک اشکال مطرح کرده‌اند (A. Otair, 2013: 9)؛ درحالی‌که به نظر می‌رسد عدم تفکیک صحیح بین استم و لِمَّا در زبان عربی موجب چنین خلطی شده است. لازم به ذکر است که استم و لِمَّا در هر زبانی باید با توجه به ویژگی‌های دستوری و استعمالی آن زبان تعریف شود. به‌عنوان مثال، در زبان انگلیسی وصف ساده، قید و صفت تفضیلی غالباً با اضافه کردن برخی پسوندها قابل ساخت است؛ مانند («Tired» به معنای (فرد خسته از فعل «tire»)، «Badly»(به شکلی بد) و «Taller» (بلندتر). اما در زبان عربی، ساخت صفت عموماً قاعده‌مند نیست؛ مانند «شهید» از فعل «یشهد» و «شجاع» از فعل «یشجع». ساخت صفت‌های تفضیلی نیز با یک قالب جدید در وزن «أفعل/فُعلی» صورت می‌پذیرد. ازاین‌رو، بیشتر متخصصین فعال در حوزه تولید «لِمَّا» از کلمات عربی همچون Boudchiche and Mazroui (2018) و Mubarak (2017)، صیغه یک ماضی را (برخلاف زبان انگلیسی که استم می‌دانند) به‌عنوان «لِمَّا» برای افعال مضارع و امر معرفی کرده‌اند.

۸. **ساخت واژگان مرکب:** عربی امکان تشکیل کلمات جدید را از طریق پیشوند و پسوند فراهم می‌کند که در نتیجه، بسیاری از اشکال بدیع توسط این افزونه‌ها ایجاد می‌شود. اما فرآیند حذف افزونه‌ها از کلمات مرکب، قدری پیچیده و همراه با چالش‌های فراوانی است؛ چرا که گاهی پسوند یا پیشوند جزئی از کلمه می‌باشد و به اشتباه به‌عنوان پیشوند یا پسوند شناسایی می‌شود.

۹. ابهام معنایی: بسیاری از کلمات عربی دارای تعابیر یا معانی متعددی هستند که بسته به زمینه، مستلزم بررسی دقیق بافت معنای مورد نظر است. از این رو ، در طراحی موتورهای پیراسته‌ساز، باید تکنیک‌های ابهام‌زدایی معنایی نیز مورد توجه قرار گیرد. به‌عنوان مثال، کلمه «بطریق» به دو صورت قابل تصور است: یکی به‌صورت «بِطْرِیْق» که به معنای «پنگوئن» است و دیگری به‌صورت «بِطَرِیْقٍ» که حرف باء پیشوند و «طریق» هسته اصلی کلمه است. اگر صورت اول لحاظ شود، نباید حرف باء حذف شود چون جزء اصل کلمه است و اگر صورت دوم لحاظ شود، باید حرف باء حذف شود چون پیشوند است و زائده محسوب می‌شود.

به‌طور خلاصه، توسعه استمر عربی دقیق و همه‌کاره مستلزم پرداختن به این مسائل و سایر مسائل مرتبط با ویژگی‌های زبان عربی است. این مسائل نشان‌دهنده یک مسیر تحقیقاتی چالش‌برانگیز و در عین حال هیجان‌انگیز برای متخصصان حوزه پردازش زبان طبیعی (NLP) می‌باشد.

بدین ترتیب، هدف نگارندگان در این مقاله مشتمل بر دو بخش است:

۱. طراحی موتور پیراسته‌ساز «نور» در مواجهه با چالش‌های مطرح شده: استمر جدید در مرحله حذف وندها از قواعد دقیق و پیشرفته‌ای بر حسب استعمال کلمات عربی پیروی نموده و تمامی حالات ممکن صحیح برای ساخت استم را پیشنهاد می‌دهد. در این استمر با لحاظ اعراب کلمات و مطابق با خوانش‌های مختلف یک کلمه با ریشه‌های متفاوت، پاسخ‌های متعددی بر پایه استعمال و بر اساس یک رده‌بندی (Rank) هوشمند در اختیار محققین قرار می‌گیرد. در استمر نور علاوه بر تشخیص تغییرات پیچیده و قاعده‌مند کلمات، از شبکه واژگانی عربی نیز جهت تجزیه صرفی ساختارهای سماعی نهایت استفاده صورت گرفته است.

لازم به ذکر است که در متن‌های منسجمی که دارای یک ارتباط منطقی در کلام هستند، می‌توان با بهره‌گیری از سیستم‌های هوشمند در لایه‌های نحوی و معنایی، استم‌های متعدد را پالایش نموده و استم صحیح را در اختیار محققین قرار داد. به‌عنوان مثال، استم کلمه «طهران» در یک متن طبقه‌بندی شدهٔ فقهی «طُهر» (پاکی) بوده ولی در متنی با طبقه‌بندی موضوعی تاریخ یا جغرافیای ایران «طهران» (شهر تهران) خواهد بود. مثال دیگر کلمه «جاز» است که مطابق با آیه ۳۳ لقمان: (لا مَوْلُودٌ هُوَ جازٍ عَنْ والِدِهِ شَيْئاً) در متون عربی کلاسیک دارای استم «جازی» (ضامن و کفیل) است؛ ولی همین کلمه در عربی معاصر دارای استم «جاز» (یک سبک موسیقی) خواهد بود. از این روست که ما معتقدیم یک استمر عربی مطلوب باید تمامی استم‌های ممکن صحیح در مورد یک کلمه عربی را محاسبه و رتبه‌بندی نماید.

۲. ارائه یک الگوی منحصربه‌فرد در ارزیابی و مقایسه استمرهای عربی: اگر یک استمر مدعی ارائه راه‌حل برای تمامی چالش‌های فوق باشد، جهت سنجش اعتبار آن و مقایسه با دیگر موتورهای

استمر، باید مطابق با یک داده طلایی[۱] جدید و بر اساس الگوریتم ارزیابی مطابق با آن، مورد سنجش قرار گیرد. در الگوریتم ارزیابی پیشنهادشده، استم‌های صحیح در حالت تک‌جوابی و چندجوابی به ازای هر کلمه عربی (دارای اعراب یا فاقد آن) لحاظ خواهد شد.

ج. موتور پیراسته‌ساز هوشمند نور

چالش‌های مطرح‌شده در پیشبرد طراحی یک موتور استمر عربی، در متون عربی کلاسیک نمود بیشتری دارند. عربی رسمی و ادبیات که نقطه مقابل عربی محاوره‌ای قرار دارد، به دو دسته عربی کلاسیک و عربی معاصر تقسیم می‌شود. عربی کلاسیک مربوط به متون قدیمی است و از آن به‌عنوان عربی نوشتار قرآنی نیز یاد می‌شود. اما امروزه نوشتار رسمی و زبان گفتار غالب کشورهای عربی بر اساس عربی معیار معاصر یا عربی استاندارد فصیح است. تجزیه و تحلیل متن اخبار تلویزیونی، روزنامه‌ها و کتاب‌های معاصر و مقایسه آن با متون عربی کلاسیک به‌ویژه منابع متنی علوم اسلامی همچون علوم قرآنی، تاریخ، فقه و اصول، نشانگر تفاوت ساختاری این دو بوده و حکایت از پیچیدگی دوچندان قواعد صرفی و نحوی در عربی کلاسیک دارد.

موتور استمر «نور» که در مرکز تحقیقات کامپیوتری علوم اسلامی (CRCIS) طراحی و ساخته شده است، با تکیه بر داده‌ای از کلمات عربی بنا شده است که بیش از ۸۰ درصد آن از متون عربی کلاسیک است. بیش از ۹۹ درصد کلمات غیرتکراری و بدون اعراب این دادهٔ ارزشمند توسط محققین مرکز نور تعیین ریشه و برچسب‌گذاری شده است. تعداد تقریبی کل این دیتا یک میلیارد و پانصد میلیون کلمه است. تعداد این کلمات در حالت بدون اعراب و به‌صورت غیرتکراری نزدیک به ۲٫۶۰٫۰۰۰ کلمه بوده که بیش از دو میلیون و دویست هزار کلمه آن که ۹۹ درصد دادهٔ کل را تشکیل می‌دهند، تعیین وضعیت شده‌اند. اگر کلمه دارای ریشه‌ای معتبر در لغتنامه‌های متقدم و متأخر باشد، آن ریشه متناسب با استعمال آن لغت انتخاب شده است؛ مانند کلمه «فتاه» که این کلمه در دادهٔ مرکز نور با در نظر گرفتن همه حالات اعرابی و نوشتاری، ۱۲۸۵ بار تکرار شده است. این کلمه در برخی متن‌ها از ریشه «فتو» به معنای «جوان خدمتگزارش» و در برخی دیگر از موارد

۱. منظور از داده‌ی طلایی (GOLD) یک داده ارزشیابی استاندارد است که توسط محققین خبره انسانی ساخته شده است.

استعمال به معنای «تکبر کرد/گمراه شد» از ریشه «تیه» است. این دو ریشه توسط محققین برای کلمه «فتاه» در نظر گرفته شده است. در صورت نداشتن ریشه، برچسب‌های دیگری مانند علم، دخیل و معرب، غیر عربی، دخیل در فارسی و ... برای کلمات انتخاب شده است؛ به‌عنوان مثال، کلمه «النیسابوری» با ۳۴۸۱۹ بار تکرار دارای برچسب معرّب است که در زبان عربی ریشه‌ای ندارد.

نکته ضروری این است که انتخاب ریشه توسط محققین مرکز تحقیقات نور با نگرش استعمالی صورت گرفته است، نه نگرش منطقی. به‌عنوان مثال، در همان نمونه «فتاه»، می‌توان با منطق صرفی ریشه «فتت» را نیز در نظر گرفت؛ به معنای «آن دو نفر آن چیز را ریز ریز کردند». اما در میان ۱۲۸۵ مورد از تکرار این کلمه، مواردی که از ریشه «فتت» باشد یافت نمی‌شود.

مثال دیگر، کلمه «لبناتک» است که در تمام ۱۲۵ مورد تکرار آن، موردی که بر اساس ریشه «لبن» باشد نیز یافت نشده است.

تصویر (۱): نمونه‌ای از ابزار واژگان استفاده‌شده در مرکز تحقیقات کامپیوتری علوم اسلامی نور

بدیهی است که با داشتن این دادهٔ ارزشمند، طراحی یک موتور استمر تنها با رویکرد یافتن ریشه امری بیهوده خواهد بود. در واقع، به واسطه تلاش پژوهشگران مرکز تحقیقات نور و بهره‌گیری از فعالیت‌های ماشینی، برای غالب کلمات، ریشه صحیح بر اساس استعمال صحیح انتخاب شده است و به نوعی می‌تواند نقش یک داده ارزیابی برای سیستم‌های هوشمند مدعی پاسخ در ریشه،

همچون(Khoja (Khoja and Garside; 1999 را ایفا کند. ازاین‌رو، رویکرد موتور استمر نور کاملاً بر پایه ساخت استم صحیح قرار گرفته است و از پاسخ‌های ریشه نیز در پالایش نتایج خود نهایت بهره را می‌برد؛ با این حال، این موتور توانایی کاملی برای تعیین ریشه کلمات جدید، به‌ویژه کلمات متون عربی معاصر که در آن دادهٔ یک و نیم میلیاردی وجود ندارند، دارد.

نقطه قوت موتور استمر نور در راستای هوشمندسازی علوم اسلامی، تکیه بر همین داده پشتیبانی متون عربی کلاسیک و معاصر و فعالیت پژوهشگران در برچسب‌گذاری کلمات آن، علاوه بر پیاده‌سازی قواعد پیچیده زبان عربی است. اگرچه غالب کلمات این داده برگرفته از کتاب‌های عربی کلاسیک است، اما همان معدود مجلات و کتاب‌های عربی معاصر نیز تشخیص ریشه و برچسب‌گذاری شده‌اند. ازاین‌رو، موتور استمر هوشمند «نور» بر پایه الگوی منحصربه‌فردی طراحی شده است که تاکنون نمونه مشابهی برای آن وجود ندارد.

در ادامه، گام‌هایی از فرآیند هوشمندسازی در این موتور تبیین می‌گردد:

گام نخست: پالایش کلمات

اولین فرآیند در محاسبه استم یک کلمه عربی، حذف اعراب، جستجوی کلمه و بازخوانی ریشه و برچسب‌های آن از ابزار واژگان است. با استفاده از برچسب‌هایی که توسط محققین برای کلمات موجود در منابع متنی مرکز تحقیقات کامپیوتری علوم اسلامی اعمال شده است، پالایش‌هایی صورت گرفته و برخی لغات از چرخه پیراسته‌سازی خارج می‌شوند. به‌عنوان مثال، کلمات فارسی که بسیار شبیه الفاظ عربی هستند و ممکن است در برخی کتاب‌های دارای متن عربی و فارسی، همچون شرح‌های مزجی با کلمات عربی خلط شوند، از فرآیند استم‌سازی خارج می‌شوند. مانند کلمه «سبزوار» که ممکن است با حذف «س» و «ب» در موتورهای استمر عربی به استم نادرست «زوار» بازگشته و به رشته‌ای از کلمات نامرتبط متصل شود. کلمه «سبزوار» با داشتن برچسب «کلمات غیر عربی» وارد چرخه استم‌سازی عربی نخواهد شد.

یکی از ویژگی‌های منحصربه‌فرد استمر «نور»، پشتیبانی از کلمات خاص قرآنی است. قرآن کریم به‌دلیل پیچیدگی قواعد صرفی و وجود قرائت‌های چهارده‌گانه و چندین نسخه نوشتار مختلف در طول قرون متمادی، به‌عنوان متنی ویژه در فعالیت‌های پردازش هوشمند زبان عربی شناخته

می‌شود (Jaafar et al, 2017: 170). به‌عنوان مثال، نسخه عثمان طه در عصر حاضر بیش از دیگر نسخه‌ها مورد استقبال ناشران قرار گرفته است؛ اما در همین نسخه با کلماتی مواجه هستیم که کاملاً مخالف قواعد نگارش عربی نگاشته شده‌اند؛ مانند کلمات «یسمریّ»، «بقیت الله»، «امرأت نوح» که نوشتار صحیح آن‌ها «یا سامریّ»، «بقیة الله» و «امرأة نوح» است. بانکی از کلمات قرآن در نسخه‌های مختلف توسط محققین گردآوری شده است که موتور استمر را در تشخیص استم صحیح پشتیبانی می‌کند. موتور استمر با تکیه بر این بانک، کلمات قرآنی با نوشتار نادرست را پالایش نموده و پس از تبدیل به نوشتار صحیح، آن‌ها را در چرخه تولید استم قرار می‌دهد.

گام دوم: محاسبه وندها[1]

نتایج تحقیقات نشان می‌دهد که یکی از معیارهای سنجش پاسخ‌های یک استمر، ارزیابی توانایی آن در حذف وندها و قرار دادن تعداد بیشتری از کلمات مرتبط ذیل یک استم است. در واقع، هرچقدر یک استمر در ایجاد تغییرات در کلمه موفق‌تر باشد، قدرت استم‌سازی آن در درجه بالاتری قرار می‌گیرد(Frakes and Fox, 2003: 1) . این معیار در مورد کلمات انگلیسی که غالباً با حذف پیشوند و پسوندهای محدود، به دگرگونی خاصی در کلمه منجر نمی‌شود، قابل قبول است؛ مانند بازگشت کلمات «engineered» و «engineering» به استم «engineer» یا بازگشت کلمه «skiing»به استم «sky» که با حذف پسوند «ing» نهایتاً با یک تغییر در حروف اصلی همراه خواهد بود.

اما در مورد زبان عربی که گسترهٔ استعمالی پیشوندها و پسوندها دایره وسیع‌تری نسبت به دیگر زبان‌ها دارد، همواره چالش‌هایی مانند حذف نادرست وندها (همچون حذف «وال» از کلمه «والده») و تغییرات کلمه (مانند حذف «ت» از «قلت») مطرح بوده و لذا معیار «قدرت» معیار مناسبی جهت ارزیابی استمرهای عربی نیست؛ از این رو بسیاری از سازندگان استمر عربی (Jaafar et al, 2017)معیار «دقت» را در ارزیابی خود ترجیح داده‌اند.

1. Clitics.

در ساخت استم از کلمات عربی، جهت دستیابی به دقت حداکثری در حذف وندها، نیازمند الگویی قابل اعتماد بر اساس قواعدی متقن از علم صرف هستیم. این امر در استمر «نور» با دو راهکار همراه بوده است:

۱. پیاده‌سازی قواعد مربوط به وندیت: استمر «نور» با لحاظ اعراب وندها در کلمات عربی، استم‌های نادرست را بر اساس این کاراکترها حذف می‌نماید. در این موتور، تمام حالات اعرابِ صحیحِ هر یک از پیشوندها و پسوندهای موجود در کلمات عربی به‌طور کامل احصاء شده است. برای مثال، پیشوند «فـ»، چهار صورت اعراب برای آن قابل تصور است؛ ولی تنها علامت اعراب «فَـ» با فتحه، به‌عنوان اعراب صحیح قلمداد می‌شود؛ از این رو در محاسبه استم «فُسّاق» (انسان‌های فاسق)، حرف «فـ» حذف نمی‌شود؛ ولی در فرض بدون اعراب یعنی «فساق»، پیشوند «ف» امکان حذف دارد.

پیشوندهای زبان عربی شامل «ال، ل، ک، ب، ف، س، أ، و، ت» و پسوندها شامل «ا، و، ن، ت، تا، تما، تم، تن، نا (اعم از ضمایر مرفوعی)، ک، کما، کم، کن، ه، ها، هما، هم، هن، ی (اعم از ضمایر منصوبی مجروری)، (ان، ون، ین، ات)» است که اختصاص هر یک از وندها به انواع کلمه یعنی فعل، اسم و حرف نیز مشخص شده است. با توجه به دانسته‌های علم صرف می‌توان قواعد متقنی را برای هر یک از این وندها تأسیس نمود. برخی از این قواعد در کتاب‌های علم صرف تصریح شده و برخی دیگر تنها انتزاع ذهن ریاضی محققین از دانسته‌های تجزیه کلمات عربی و بررسی استعمال آن‌ها بوده و در کتابی به‌صورت قاعده‌مند اشاره نشده است. به‌عنوان مثال، برخی از این قواعد چنین است:

– هیچ‌یک از پیشوند و پسوندها دو بار تکرار نمی‌گردد؛ برای مثال، کلمه «کَکَلامٍ»، صرفاً دارای یک پیشوند «ک» است. از این قاعده حالت «لل» به جهت مثال‌هایی همچون «للرجل» مستثنی است.

– پیشوند «س» قبل از پیشوند دیگری قرار نمی‌گیرد (به‌عبارت‌دیگر بعد از آن پیشوندی وجود ندارد و آخرین پیشوند متصل به استم است). به‌عنوان مثال، در کلمات «سأعلم / ستعلم» دیگر «أ» و «ت» پیشوند نخواهند بود. پیشوند «س» با پیشوندهای «ب» و «ال»[1] نیز قابل جمع نیست.

– پیشوند «ف» پس از پیشوندهای «ب / ک / ل / ال» قرار نمی‌گیرد.

– پسوندهای «ان / ون / ین / ات» که برای ساخت جمع سالم استفاده می‌شوند، با یکدیگر جمع نمی‌شوند.

---

۱. پیشوند «س» با «ک» نیز جمع نمی‌شود؛ ولی به دلیل مثال‌هایی چون «کسیکفی» که در مقام مثال استعمال شده است، این مورد استثناء شده است.

تبصره ۱: قبل از این چهار پسوند، تنها «ی» می‌تواند پسوند باشد.

تبصره ۲: اگر پس از حذف پسوند «ات»، استم کمتر از چهار حرف باشد، باید به انتهای استم یک «ة» اضافه شود؛ مانند مثال‌های «ذرات، طلحات، حسنات» که استم آن‌ها به‌ترتیب «ذرة، طلحة، حسنة» است؛ مگر اینکه کلمه دارای برچسب علم باشد، مانند کلمه «هندات» که استم آن «هند» است و نه «هندة».

در تمامی استمرهای شناخته‌شده، حرف «ی» مطلقاً به‌عنوان یک پسوند لحاظ شده و غالباً در فرآیند ساخت میانوند حذف می‌گردد؛ درحالی‌که «ی» در کلمات منسوب کاملاً ماهیت کلمه را تغییر داده و جزئی از کلمه است. به‌عنوان مثال، کلمه «السبحانی» یک اسم علم بوده ولی کلمه «سبحان» یک مصدر است؛ از این رو در استمر «نور»، یاء نسبت در کلمات منسوب حذف نمی‌گردد. ثمره تفکیک کلمات منسوب از اصل خود در لایه‌های نحو و معنا کاملاً نمود پیدا می‌کند؛ هرچند در طراحی استمر نور امکان حذف این یاء جهت گستره بیشتر یک استم نیز پیش‌بینی شده است.

این قواعد که تنها بخشی از قواعد به کار رفته در موتور استمر نور است، نشانگر پیچیدگی قواعد وندیت در زبان عربی و ضرورت کدنویسی دقیق در مرحله حذف وندهاست که می‌تواند وجه تمایز برخی موتورهای هوشمند از باقی باشد.

۲. حذف حالات ناممکن قطعی: اگرچه قواعد دقیق مربوط به وندیت در ساخت استم بسیار راهگشاست، ضبط تمام این قواعد بسیار مشکل بوده و از سوی دیگر، استثنائات موجود در کلمات عربی، پیاده‌سازی تمامی آن‌ها را ناممکن می‌سازد. ازاین‌رو، تمام حالات ممکن ترکیبی از پیشوندها و پسوندها توسط ماشین محاسبه شده و در اختیار محققین جهت بازبینی قرار گرفت. به‌عنوان مثال، اگر برای کلمات عربی حداکثر امکان ترکیب چهار پیشوند قابل تصور باشد؛ همچون کلمه «أفَبِالباطل» در آیه ۷۲ سوره نحل، بیش از ۳۰۰۰ حالت ترکیبی پدید می‌آید که به واسطه قواعد وندیت در بخش اول، می‌توان این تعداد را به تقریباً ۸۰۰ مورد کاهش داد. از این میان، نزدیک به ۱۲۰ مورد توسط محققین به‌طور قطعی یک ترکیب نادرست تشخیص داده شده و با تهیه لیستی از آن‌ها، برنامه‌نویسان موتور استمر «نور» از پیاده‌سازی قواعد پیچیده بی‌نیاز شده‌اند. به‌عنوان مثال، ترکیب‌های پیشوندی «أ-كَ-فَ»، «بِ-لَ-الْ» و «وَ-بِ-فَ» توسط محققین مرکز نور به‌عنوان حالات ترکیبی نادرست شناسایی شده‌اند.

گام سوم: تطبیق ریشه

دانستیم تک‌جوابی یا چندجوابی بودن نتایج یک استمر هر دو با چالش‌هایی روبروست. مشکل تک‌جوابی با در نظر گرفتن تمام استم‌های ممکن برای یک کلمه قابل رفع است؛ ولی مشکل چندجوابی و تولید استم غیر مستعمل برای یک کلمهٔ عربی، چالشی است که در هیچ‌یک از موتورهای استمر فعلی مطرح نبوده است. یکی از وجوه تمایز موتور استمر «نور» از دیگر استمرهای زبان عربی، رفع این چالش از دو روش منحصربه‌فرد است:

۱. روش تطبیق ریشه با ابزار واژگان

۲. استفاده از الگوی استعمال که در گام بعدی خواهد آمد.

گذشت که انتخاب ریشه توسط محققین برای کلمات غیرتکراری دادهٔ مرکز تحقیقات کامپیوتری علوم اسلامی «نور» با نگرش استعمالی بوده است. به‌عنوان مثال، برای کلمه «یبسها» سه ریشه «ب‌س‌س»، «ی‌ب‌س» و «ب‌ی‌س» (مضارع مجزوم) از لحاظ منطق صرفی قابل تصور است؛ حال آنکه با نظر به استعمال این کلمه در منابع متنی موجود، تنها دو ریشه «ب‌س‌س» و «ی‌ب‌س» مستعمل است. مثال دیگر کلمه «لبناتک» است که ریشه «لبن» در هیچ‌یک از موارد استعمال این کلمه مشاهده نشده است.

الگوریتم تطبیق ریشه بدین صورت است که پس از تحصیل تمامی استم‌های ممکن بر اساس منطق صرفی، فیلتری جهت پالایش پاسخ‌های غیر مستعمل اعمال می‌گردد. اگر کلمه ورودی دارای ریشه باشد، استم جاری نیز باید دقیقاً همان ریشه را داشته باشد. به‌عنوان مثال، اگر کلمه ورودی «فسق» باشد، روند استم‌سازی و حذف استم‌های نامعتبر به‌صورت زیر خواهد بود:

فَسَقَ				
پیشوند	استم	پسوند	ریشه استم جاری	وضعیت
-	فسق	-	فسق	استم صحیح
فَ	سق	-	سوق/سقی/سقق	عدم تطابق ریشه (غلط)
فَ + سَ	ق	-	وقي	عدم تطابق ریشه (غلط)

جدول (۱): نمونه‌ای از تطبیق ریشه‌ی استم با ریشه‌ی کلمه ورودی

گام چهارم: الگوی بررسی استعمال

با توجه به ظرفیت دادهٔ موجود در مرکز تحقیقات کامپیوتری علوم اسلامی، می‌توان از میزان تکرار هر یک از کلمات در متون عربی کلاسیک به‌عنوان عاملی مؤثر در فرآیندهای هوشمندسازی استفاده نمود. اساس و مبنای کارکرد الگوی بررسی استعمال در فرآیند پیراسته‌سازی، بر مبنای عملکرد ذهن یک انسان آشنا به زبان عربی است. به‌عنوان مثال، اگر به شخصی که آشنایی اجمالی با زبان عربی دارد، گفته شود که ساخت استم با حذف پیشوند و پسوند صورت گرفته و استم کلمات «المعلمان» و «بکتاب»، به‌ترتیب «معلم» و «کتاب» خواهد بود، بی هیچ تردیدی استم کلمات «فإبراهیم» و «لخراسان» را به‌ترتیب «إبراهیم» و «خراسان» اعلام می‌دارد و نه «راهیم» (حذف «ا / بـ» به‌عنوان پیشوند) و نه «خراس» (حذف «ان» به‌عنوان پسوند)؛ زیرا استعمال چنین کلماتی برای او آشنا نیست.

الگوی فوق در موتورهای هوشمند ساخت استم کلمات عربی، به‌ویژه کلمات فاقد ریشه همچون غالب کلمات دخیل، معرب و اعلام، بی آنکه نیاز به بانکی از کلمات باشد، کاملاً راهگشاست. بدین ترتیب، اگر میزان استعمال کلمات بر مبنای داده‌ای قابل توجه (اعم از عربی کلاسیک و عربی معاصر) محاسبه شود، اختلاف فاحش میان میزان تکرار کلمه ورودی و استم آن می‌تواند به‌عنوان یک راه‌حل در حذف استم‌های نادرست باشد. به‌عنوان مثال، در کلمه معرب «تاشفین» (یکی از سلاطین سرزمین اندلس قدیم) با ۴۰۷۸ بار تکرار در منابع متنی مرکز نور، میزان استعمال استم‌های ممکن آن در حالت پیراسته مطابق جدول شماره ۲ چنین خواهد بود:

تاشفین						
پیشوند	استم	پسوند	ریشه	استعمال	وضعیت	علت
-	تاشفین	-	-	٤٠٧٨	صحیح	
ت	اشفین	-	-	٦	غلط	پیشوند «ت» فقط در کلمات خاص حذف می‌شود
-	تاشفی	ن	-	٠	غلط	عدم استعمال کافی
-	تاشف	ین	-	٠	غلط	عدم استعمال کافی

جدول (۲): نمونه‌ای از بررسی الگوی استعمال در پالایش استم‌ها

با توجه به استعمال کم یا عدم استعمال هر یک از استم‌ها، خود کلمه «تاشفین» به‌عنوان استم صحیح اعلام خواهد شد. ذکر این نکته ضروری است که به‌طور معمول، میزان تکرار یک کلمه در حالت پیراسته، همچون کلمه «کتاب»، بیش از میزان تکرار آن در حالت متصل به یک پیشوند یا پسوند است. بدیهی است ریشه و برچسبی که برای یک کلمه پیراسته اعمال می‌شود، در فرض اتصال آن کلمه به وندها نیز جاری است.

به‌عنوان مثال، کلمه «إسماعیل» دارای ۳۰۷۰۲۱ بار تکرار، یک کلمه معرب از ریشه «سمعل» است و کلمه «بإسماعیل» با ۹۴۸ بار تکرار نیز یک کلمه معرب و از همان ریشه فوق است.

رتبه‌بندی پاسخ‌ها

اگر چندجوابی در استمرهای زبان عربی به‌عنوان یک امتیاز در فرآیندهای هوشمندسازی شمرده شود، حذف استم‌های نادرست و غیر مستعمل از میان پاسخ‌های متعدد، در کنار رتبه‌بندی استم‌های صحیح به‌عنوان یک ویژگی منحصربه‌فرد استمر «نور» خواهد بود. با استفاده از الگوی بررسی استعمال می‌توان استم‌های صحیحی را که با استفاده از ریشه‌های مستعمل فیلتر شده‌اند، رتبه‌بندی نمود.

به‌عنوان مثال، کلمه «خطت» با ۱۵۰۰ بار تکرار دارای سه ریشه «خطط»، «خطو» و «خیط» بوده و بر مبنای آن نیز دارای سه استم «خطّ»، «خطا» و «خاط» است. با استفاده از الگوی بررسی استعمال، اولویت و رتبه‌بندی استم‌های خروجی مطابق جدول شماره (۳) چنین خواهد بود:

خطت (خطط/خطو/خیط)

استم	پسوند	ریشه	استعمال	وضعیت	رتبه
خط	ت	خطط	٥٥٦٣٩	صحیح	۱
خطا	ت	خطو	١١٥٦٧	صحیح	۲
خاط	ت	خیط	١٣٥٠	صحیح	۳

جدول (۳): نمونه‌ای از رتبه‌بندی پاسخ‌ها در استمر نور

تعداد مشتقات متصل به یک ریشه و تعداد مشتقات مرتبط با یک استم مشترک از دیگر الگوهای قابل استفاده در رتبه‌بندی پاسخ‌ها هستند. یکی از برنامه‌های پیش‌رو در بخش متن‌کاوی مرکز نور، تعیین وضعیت ریشه بر اساس اعراب در کلمات دارای بیش از یک ریشه تایید شده است. به‌عنوان مثال، در کلمه «خطت» اگر اعراب لحاظ شده و به‌صورت «خِطت» باشد، تنها ریشه «خیط» صحیح خواهد بود. پالایش ریشه بر اساس اعراب هر کلمه می‌تواند به‌عنوان الگویی جهت رتبه‌بندی ریشه‌ها و استم‌های آن کلمه در فرض بدون اعراب نیز قرار گیرد.

گام پنجم: تطبیق بر مداخل لغوی فرهنگ‌نامه‌ها

بسیاری از موتورهای استمر عربی با رویکرد ریشه‌محور، استم حاصل از کلمه را پس از حذف وندها در مداخل لغوی فرهنگ‌نامه‌ها جستجو کرده و بدین ترتیب ریشه به‌دست آمده از استم را برای کلمه ورودی نیز اعلام می‌دارند(alkabi et al, 2015: 96) . از این ترفند می‌توان برای دستیابی به استم واقعی کلمات، به‌خصوص کلماتی که در داخل فرهنگ‌نامه‌ها فاقد ریشه هستند، مانند کلمات دخیل و معرب (همچون «تلفاز» و «فیزیاء») استفاده کرد.

مداخل لغوی در معاجم متأخر عموماً به‌صورت پیراسته‌شده مرتب شده‌اند و تنها در مورد برخی اسم‌ها با پیشوند «ال» ضبط شده‌اند. بیش از صد هزار مدخل لغوی پیراسته‌شده از لغتنامه‌های قدیمی و معاصر همچون معجم «المعجم الوسیط» (مصطفی و همکاران، ۱۴۲۹) که توسط محققین مرکز نور نیز برچسب‌گذاری صرفی شده است، به‌عنوان یک دادهٔ پشتیبانی در موتور استمر «نور» به کار رفته است.

اگر کلمه‌ای پس از فرآیندهای پیشین به یکی از کلمات فوق برسد، روند استم‌سازی متوقف شده و همان کلمه به‌عنوان استم صحیح در خروجی درج می‌شود. به‌عنوان مثال، مطابق جدول شماره (۴) فرایند ساخت استم در کلمه «بالرمضان» با شناسایی «رمضان» به‌عنوان مدخل لغوی متوقف شده و دیگر پسوند «ان» از آن حذف نمی‌گردد.

بالرمضان				
پیشوند	استم	پسوند	ریشه	وضعیت
ب	الرمضان	-	رمض	-
ال	رمضان	-	رمض	استم صحیح (بانک مداخل لغوی)

جدول (٤): نمونه‌ای از تطبیق استم بر مداخل فرهنگ‌نامه‌ها

گام ششم: پیاده‌سازی قواعد اعلال و ادغام

پیاده‌سازی قواعد پیچیدهٔ اعلال، ادغام و تخفیف که به‌عنوان یکی از چالش‌های جدی در طراحی موتورهای استمر عربی مطرح است، کمتر در فعالیت‌های هوشمند صرفی مورد تحلیل قرار گرفته است. اگرچه طراحی و ساخت یک موتور هوشمند برچسب‌گذار کلمات عربی در لایه صرف در

سال‌های پیشین (سریانی، مینایی؛ ۱۳۹۰) و (دانش؛ ۱۳۹۳) قدم‌های مهمی را در این مسیر طی نمود، بررسی نتایج تحلیل‌گر صرفی «نور» و مقایسه آن با برخی دیگر از تحلیل‌گرهای صرفی حاکی از روند رو به رشد پاسخ‌های مطلوب در این موتور هوشمند داشت (الهی‌منش؛ ۱۳۹۴: ۱۷).

با این‌حال، توقف فعالیت آن موتور به لحاظ برخی ملاحظات برنامه‌نویسی و همچنین ضرورت استفاده از یک استمر سبک به‌ویژه در کلمات فاقد ریشه و یا دارای ساختارهای صرفی پیچیده، موجب شد قواعد اعلال و ادغام این بار به‌صورت یک سری قواعد ساده شده در دستیابی به استم کلمات معتل و مضاعف به کار گرفته شوند. نگارندگان با نمونه‌ای که به پیاده‌سازی این قواعد در موتورهای استمر تصریح داشته باشد یا در نتایج خروجی آن نسبت به ساخت استم از افعال معتل و مضاعف به‌درستی عمل کرده باشد، مواجه نشدند و تنها برخی استمرهای عربی (Ababneh et al, 2012) نسبت به ساخت استم از برخی کلمات جمع، به بیان مثال‌هایی همچون ساخت استم «محامی» از کلمه «محامون» اکتفا نموده‌اند.

یک نمونه از قواعد بسیار پرکاربرد که با بیانی ساده در فرایند هوشمندسازی موتور استمر «نور» به کار رفته است، قاعده «ادغام» است. بر اساس این قاعده، اگر در یک کلمه عربی، حروف دوم و سوم ریشه (مانند ق ر ر) یکسان باشند و در میان پاسخ‌های نهایی استمر نیز، یک استم سه حرفی تولید شود که حروف دوم و سوم آن همانند ریشه باشد (مانند قرر)، در اینصورت با حذف حرف سوم از استم، یک استم جدید به پاسخ‌ها اضافه خواهد شد.

به‌عنوان مثال، فعل مضاعف «قررتک» که بر اساس منطق صرفی می‌تواند به دو شکل ثلاثی مجرد یا مزید باشد، مطابق تصویر شماره (۲) دارای دو استم نهایی مبتنی بر قاعده فوق است.

ردیف	کلمه	پیشوند	استم	پسوند	نوع	ریشه	استعمال	وضعیت	علت
۰	قررتک	–	قررتك	–	–	قرر	۱۷	–	
۱	قررتک	–	قررت	ك	–	قرر	۳۰۳۱	–	–
۲	قررتک	–	قرر	ت	–	قرر	۲۴۷۵۸	مطلوب	–
۳	قررتک	–	قر	ت	–	قرر/قرر	۸۸۳۳	۰	مطلوب
۴	قررتک	–	قرة	–	–	–	۰	مطلوب	قواعد جمع
۵	قررتک	–	قرر	–	–	–	۰	مطلوب	قواعد جمع

تصویر (۲): نمونه‌ای از پیاده‌سازی قواعد ادغام جهت ساخت استم صحیح

قواعد مربوط به «اعلال» نیز با بیانی ریاضی در مورد کلمات مثال، اجوف، ناقص و لفیف در کدهایی ساده به موتور استمر «نور» اضافه شده است. بر اساس این قواعد، به‌عنوان مثال، استم کلمه معتل اجوفِ «یبسن» از ریشه «بوس»، صیغه یک مضارع یعنی «یبوس» بوده و استم کلمه معتل ناقصِ «ترمون» از ریشه «رمی»، کلمه «ترمی» خواهد بود.

گام هفتم: استفاده از شبکه واژگانی[1] در ساخت استم از جمع‌های مکسر

برخی از طراحان پیشرو در ساخت استمر عربی(Chen and Gey; 2002) ، در نخستین فعالیت‌های خود در این عرصه که با ارسال مقاله‌ای به یازدهمین اجلاس بازیابی اطلاعات متنی[2] در سال ۲۰۰۲ همراه شد، از جمع‌های مکسر به‌عنوان یک چالش جدی در پردازش هوشمند متون عربی یاد نموده و ویژگی استمر خود را حل این چالش و ساخت استم مفرد از جمع‌های مکسر عنوان داشته‌اند. طراحان این استمر با استفاده از دادهٔ موازی انگلیسی عربی که به‌عنوان مثال برای واژه عربی «طفل» واژه انگلیسی«Child» آمده و برای جمع مکسر «اطفال» معادل انگلیسی «Children» پیشنهاد شده است، توانسته‌اند برای تعدادی از جمع‌های مکسر، استم مفرد آن را نیز پیشنهاد دهند. با این حال، با توجه به گستره کلمات عربی و جمع‌های مکسری که نزدیک به ۲۰ هزار مورد از آن‌ها در کتاب‌های عربی شناسایی شده است، استفاده از ترجمه موازی راهکار جامع و متقنی نخواهد بود.

جمع‌های مکسر غالباً پیچیده و فاقد یک ساختار قاعده‌مند بوده و در متون عربی نیز بسیار استعمال می‌گردند. طبق ادعای برخی تحلیل‌گران متن‌کاوی، این جمع‌ها تقریباً ۱۰ درصد متون عربی و تقریباً ۴۱ درصد از کل جمع‌های عربی را تشکیل می‌دهند (Goweder et al, 2005: 246). اگرچه برخی استمرهای عربی با رویکرد N-Gram در تشخیص جمع‌های مکسر موفق عمل نموده‌اند، همچنان در ساخت استم مفرد از این جمع‌ها ناتوان‌اند (Mustafa et al, 2017: 58). از این رو برخی سازندگان استمرهای عربی در رویکردهای نوین خود به سمت استفاده از شبکه واژگانی کلمات عربی و به‌کارگیری بانک‌های اطلاعاتی در الگوریتم استم‌سازی خود رفته‌اند (Kreaa et al, 2014: 8).

1. Arabic Wordnet.
2. Text Retrieval Conference (TREC).

با توجه به کثرت جمع‌های مکسر، مؤثرترین راه‌حل ممکن در ساخت استم مفرد، استفاده از بانک‌های اطلاعاتی موجود در شبکه واژگانی زبان عربی است. راهکاری که در استمر «نور» نیز با بهره از یک شبکه واژگانی غنی مورد استفاده قرار گرفته است (سریانی، ۱۳۹۵). استخراج جمع‌های مکسر (همچنین جمع الجمع) از کتاب «المعجم المفصل فی الجموع» نوشته امیل بدیع یعقوب به همراه استخراج کلمات دارای برچسب جمع از میان مداخل لغتنامه‌های فرمت‌گذاری شده توسط محققین مرکز نور، بانکی غنی و جامع را جهت شناسایی این نوع جمع‌ها در استمر نور مهیا ساخته است.

از آنجا که جمع‌های مکسر مختص به «اسم» هستند، یکی از نکات قابل ملاحظه در این گام، با توجه به تفکیک پیشوند و پسوندهای فعلی و اسمی از یکدیگر، این است که کلمات دارای وندهای مختص به فعل در چرخه محاسبه گام هفتم وارد نخواهند شد. به‌عنوان مثال، اگر کلمه دارای یکی از پسوندهای «تن / تما / ن / تم / تا» یا پیشوند «س» باشد، آنگاه فعل بوده و در بانک جمع مکسر جستجو نمی‌شود.

بالکتاب						
پیشوند	استم	پسوند	ریشه	استعمال	وضعیت	علت
-	بالکتاب	-	کتب	۲۲۱۱۱	-	
ب	الکتاب	-	کتب	٦٧٤٧١٥	-	-
أل	کتاب	-	کتب	١٣١١١٣٦	مطلوب	-
ك	تاب	-	توب	٣٤٨٢٢	غلط	عدم تطابق ریشه
-	کاتب	-	-	٤٢٥٤٩	مطلوب	قواعد جمع

جدول (۵): نمونه‌ای از تشخیص استم مفرد در جمع‌های مکسر عربی

کلمه «بالکتاب» با توجه به اعراب آن می‌تواند به دو صورت جامد مفرد «کِتاب» و جمع مکسر «کُتّاب» (جمع کاتب؛ یعنی نویسندگان) خوانده شود. از این رو، در استمر نور نیز مطابق با جدول شماره ۵ دو پاسخ مستقل برای آن اعلام می‌گردد.

گام هشتم: ویرایش استم

یکی از اشکال‌های رایج در متون عربی کتاب‌های قدیمی، مشکلات فونتی و تایپی این متون و عدم تصحیح آن‌ها توسط ناشران است. در دادهٔ مرکز نور که دارای بیش از ۲٫۶ میلیون کلمه غیرتکراری است، بیش از ۳۵۰ هزار اشکال فونتی و غلط‌های نوشتاری شناسایی شده است که بی‌تردید تأثیر خود را بر فرآیندهای هوشمندسازی متون اسلامی خواهد گذاشت.

یکی از ویژگی‌های استمر «نور»، پیاده‌سازی برخی قواعد نوشتاری و استفاده از الگوریتم بررسی استعمال در تصحیح این کلمات است. در این فرآیند، کلماتی همچون «المومنون»، «بالاکرام»، «المسایل» و «المدینه» که دارای نوشتار نادرست عربی هستند، به‌صورت استم صحیح خود یعنی «مؤمن»، «إکرام»، «مسائل» و «مدینة» به‌عنوان پاسخ نهایی اعلام خواهند شد.

دادهٔ ارزیابی

بررسی عملکرد هر ماشین تحلیل‌گر صرفی در گرو سنجش آن با یک دادهٔ ارزیابی‌شدهٔ دقیق توسط زبان‌شناسان است؛ داده‌ای که متشکل از کلماتی متنوع و مستعمل از حوزه‌های مختلف علوم باشد. جامعه پردازش زبان عربی (ANLP) هنوز داده‌ای یکسان جهت ارزیابی استمرها ارائه نکرده است و بیشتر استمرها بدون ارائه منابع کد یا دادهٔ ارزیابی خود، صحت عملکرد خود را بر مبنای دادهٔ ارزیابی مؤسسات داخلی خود اعلام داشته‌اند. از جمله داده‌های ارزیابی عمومی که معدود استمرهای شناخته‌شدهٔ عربی نیز از آن‌ها استفاده کرده‌اند، می‌توان به قرآن کریم، کتاب زاد المعاد، مجموعه TREC و داده‌های پیشنهادی پایگاه(LDC (Linguistic Data Consortium اشاره نمود (Yusof et al, 2010: 39).

به‌عنوان مثال، Sawalha and Atwell (2008) در بیست و دومین اجلاس زبانشناسی محاسباتی[1] دو داده ارزیابی از کلمات قرآن (نمونه عربی کلاسیک) و کلمات مجلات و روزنامه‌های آنلاین را جهت مقایسه استمرهای عربی پیشنهاد نمودند. نمونه دیگر برای متون قرآنی، دادهٔ موجود در پایگاه «corpus.quran.com» است که توسط گروه زبان‌شناسی دانشگاه لیدز انگلستان تهیه شده و مورد استفاده برخی استمرها همچون SAFAR قرار گرفته است. دانشگاه لیدز از سال ۲۰۰۹

1. COLING 2008 22nd International Conference on Computational Linguistics

برچسب‌های صرفی و نحوی کلمات قرآن را به‌صورت رایگان در پایگاه خود قرار داده است که یکی از این برچسب‌ها، استم و لمّای کلمات قرآن است؛ ولی بر اساس نتایج پژوهش گروه متن‌کاوی مرکز تحقیقات کامپیوتری علوم اسلامی «نور»، این دیتا جهت سنجش عملکرد استمرها مورد اعتماد نبوده و دارای اشتباهات بسیاری است.

به‌عنوان مثال، برای برخی کلمات همچون «لئلا» که دارای یک صورت واحد است، دو استم متفاوت بیان شده است: «لِ + یَ + لَا» (بقره:۱۵۰) و «لِ + ئلا» (نساء: ۱۶۵). علائم جمع سالم اسمی «ون، ین، ات» از اسم‌ها جدا نشده و مفرد این کلمات و جمع‌های مکسر به‌صورت لمّا اعلام شده است؛ درحالی‌که در تشخیص بسیاری از جمع‌ها نیز دارای اشتباه است؛ همچون الشهوات (آل عمران: ۱۴) و مبینات (نور: ۳۴) که به همین شکل به‌عنوان استم نهایی اعلام شده است. در برخی کلمات معتل مانند یغنیهم (نور: ۳۳) نیز آن را به صیغهٔ نادرست (أَغْنَتْ) متصل نموده است.

درحالی‌که چالش تک‌جوابی در غالب استمرهای موجود باقی است، از توانایی استمر «نور» در تولید استم‌های چندجوابیِ مستعمل، به‌عنوان یک ویژگی منحصربه‌فرد نام برده شد. با این وجود، هنوز یک دادهٔ ارزیابی عمومی با لحاظ این معیار تولید نشده است. تنها در بیست و هشتمین نشست بین‌المللی مدیریت اطلاعات تجاری در سال ۲۰۱۶، طرحی با عنوان NAFIS از سوی پژوهشگران دانشگاه محمد مراکش جهت ارزیابی موتورهای صرف و استمر عربی چندجوابی ارائه گشت (Namly, 2016)و تا به امروز به‌عنوان یک پروژه عملیاتی در اختیار محققان قرار نگرفته است. این پژوهشگران، استمر SAFAR را بر اساس دادهٔ ارزیابی خود دارای بیش‌ترین درصد پاسخ صحیح معرفی نموده‌اند.

ضرورت یک دادهٔ استاندارد طلایی (Gold) جهت محک تمامی ابزارهای هوشمند متن‌کاوی، موجب گشته است که تلاش‌های لازم در ساخت داده‌های متقن و ارزیابی‌شده، بیش از تلاش‌های لازم در ساخت ابزارهای بهره‌برداری کننده از این داده‌ها باشد(Jaafar et al, 2017: 167) . ازاین‌رو، از نخستین روزهای طراحی استمر «نور»، ساخت یک دادهٔ ارزیابی متقن و دقیق بر اساس توانایی و ظرفیت استمر نور در دستور کار قرار گرفت. استمر «نور» با تکیه بر یک دادهٔ عربی کلاسیک از متون دیجیتالی موجود در مرکز تحقیقات کامپیوتری علوم اسلامی، فعالیت خود را در سال‌های اخیر جهت پیشبرد هوش مصنوعی در این علوم آغاز نموده است. نظر به اهداف و خاستگاه این استمر، دو الگوی ارزیابی طلایی نیز جهت سنجش عملکرد آن تولید شده است:

۱. کلمات غیرتکراری قرآن با لحاظ اعراب: این مجموعه با بیش از ۱۶ هزار کلمه[۱] کاملاً مطابق با استعمال قرآنی خود، توسط محققین بخش متن‌کاوی مرکز نور، برچسب‌گذاری صرفی از جمله تعیین ریشه و استم شده است. ضرورت تحلیل صرفی در متون قرآنی به‌عنوان نمونه شاخص متن‌های عربی کلاسیک، موجب پیدایش این داده ارزیابی بوده است.

۲. مجموعه ۱۰ هزار کلمه‌ای از کلمات غیرتکراری دادهٔ متنی مرکز نور بدون لحاظ اعراب: از میان بیش از ۲٫۶۰۰٫۰۰۰ هزار کلمه غیرتکراری و بدون اعراب کل داده متنی مرکز نور و با توجه به ظرفیت موجود در برچسب‌گذاری صرفی این کلمات توسط محققین، تعداد ۱۰ هزار کلمه با استفاده از الگوهایی جامع و منحصربه‌فرد مهیا شده است تا سنجشی کامل از استمر «نور» باشد. این الگوها سنجش مختلفی از توانمندی یک استمر در فرآیندهای مختلف هوشمندسازی نشان خواهند داد. به‌عنوان مثال، دقت یک استمر در مورد جمع‌های مکسر، حذف وندها، کلمات معتل و مضاعف، نسخه‌های نگارشی مختلف قرآن بایست به‌صورت مجزا مورد تحلیل و بررسی قرار گیرد. از این مجموعه تعدادی انتخاب شده و در بخش نتایج ارزیابی، استمر نور در مقایسه با دیگر استمرها مورد سنجش قرار گرفته است.

معیارهای متداول در ارزیابی عملکرد استمرها

محققان معیارهای سنجش استمر را به دو دسته کلی طبقه‌بندی می‌کنند:

دسته اول: معیارهای مربوط به «قدرت» یک استمر که مربوط به توانایی آن در حذف وندها و قرار دادن تعداد بیشتری از کلماتِ مرتبط ذیل یک استم است.

دسته دوم: معیار «دقت» که حاکی از دقت یک استمر در ساخت استم‌های صحیح از کلمات عربی است.

به برخی از مدل‌های بیان‌شده در پژوهش‌های محققین (Jaafar et al, 2017: 167) و (Namly, 2016: 5)اشاره می‌گردد:

۱. علت بیان تقریبی کلمات غیر تکراری قرآن به سبب اختلاف در نسخه‌های مختلف است.

۱. **درصد دقت:** این مدل توسط Flores and Moreira (2016) به شکل زیر بیان شده است. در این مدل، TPتعداد استم‌های صحیح، FP تعداد استم‌های نادرست و FN تعداد استم‌های صحیحی است که توسط موتور استمر اعلام نشده است.

$$\text{Accuracy} = \frac{TP}{TP + FP + FN}$$

۲. **میانگین کلماتی که با استم مطابقت دارند(WCC)** : این مدل که توسط Galvez et al. (2005)ارائه شده است، بر مبنای نسبت میان کلمات غیرتکراری به تعداد استم‌های غیرتکراری خروجی است. به‌عنوان مثال، فرض کنید کلمات «بالمعلم»، «والمعلم»، «المعلمات» و «المعلمون» دارای استم واحد «معلم» هستند. اگر این کلمات به استمری داده شوند و صرفاً یک استم واحد برگردانده شود، دقت آن حداکثری است ولی اگر تعداد غیرتکراری بیشتری بازگرداند، از دقت آن کاسته خواهد شد. لذا هر چه عدد بزرگ‌تر باشد دقت آن بیشتر است. مدل ارائه‌شده برای ارزیابی دقت استمر به شکل زیر است:

$$WCC = \frac{C}{S}$$

در این مدل، C تعداد کلماتی است که قرار است مورد تحلیل استمر قرار گیرد وS تعداد استم‌های غیرتکراری از آن کلمات است.

۳. **ضریب پیراسته‌سازی کلمات (ICF):** در مدل ارائه‌شده توسط(2003) Frakes and Fox ، نسبت میان تعداد کل کلمات و تعداد استم‌های واحد و غیرتکراری محاسبه می‌گردد. به‌عنوان مثال، اگر تعداد کل کلمات ۱۰۰٬۰۰۰ باشد و تعداد استم‌های غیرتکراری آن‌ها ۸۰٬۰۰۰ کلمه باشد، آنگاه بر این مبنا، استمر دارای ضریب ۲۰٪ است. لذا هر چه این شاخص بالاتر باشد، دقت استمر بالاتر خواهد بود. نحوه محاسبه این شاخص به این صورت است:

$$ICF = \frac{C-S}{C}$$

در این شاخص،C تعداد کلماتی که قرار است استم شود و SSتعداد کل استم‌های غیرتکراری است.

۴. **میانگین کلمات بدون تغییر(WCA)** : گروه Al-Kabi et al (2011) شاخص خود را بر اساس میانگین تغییر کلمه ارائه کرده‌اند. به این صورت که غالب موتورهای استمر، کلماتی را که

استم آن‌ها با خودشان یکی است تغییری نمی‌دهند؛ حال آنکه استمرهای قوی، برای به‌دست آوردن استم صحیح هر کلمه‌ای را تغییر می‌دهند. نحوه ارزیابی این مدل به این شکل است:

$$WCA = \frac{C - U}{C}$$

در این شاخص، Cتعداد کل کلمات و Uتعداد کلمه‌هایی است که بعد از استم شدن، تغییری در آن‌ها ایجاد نشده است.

۵. **میانگین حروف جداشده بعد از ساخت استم (ARC)**: از دیگر معیارهای پیشنهادشده در Al-Kabi et al. (2011) بر اساس شاخص قدرت موتور استمر در جداسازی تعداد وندهاست؛ بنابراین، استمر قوی‌تر استمری است که تعداد پیشوند و پسوند بیشتری را جداسازی نماید.

$$ARC = \frac{\text{مجموع کلمات جداسازی شده}}{\text{تعداد کلمات}}$$

۶. **درصد دقت و میانگین زمان صرف‌شده:** معیاری که سازندگان موتور استمر SAFAR (Jaafar et al, 2017: 168) بر آن تاکید خاص دارند، دخیل دانستن «زمان صرف‌شده» محاسباتی در الگوریتم معیار است. از نظر آنان با توجه به حجم بالای داده‌های عربی، زمان محاسبه استم توسط ابزارهای ارائه‌شده نقشی مهم در انتخاب استمر مطلوب خواهد داشت:

$$GS - Score = \frac{\alpha . \sum T_W}{\beta . \sum Accuracy_W}$$

مقصود از GS-Score در نظر پژوهشگران فوق، پیشنهاد یک معیار جهانی (Global Stemming Score)است. در این مدل، TW میزان زمان صرف‌شده به ازای تولید استم از هر کلمه و AccuracyW دقت نتایج استمر نسبت به آن کلمات است.

از نظر نگارندگان این مقاله، پارامتر زمان تأثیر قابل‌ملاحظه‌ای در ارزیابی یک موتور استمر ندارد؛ زیرا امروزه این موتورها غالباً محاسبات خود را به‌صورت آفلاین بر متن‌های حجیم انجام می‌دهند و پس از اتمام پردازش، نتایج جستجو شده را در اختیار سیستم‌های بازیابی اطلاعات قرار می‌دهند. سرعت ارائه پاسخ به این سیستم‌ها نیز منوط به بستر شبکه و ابزارهای ذخیره‌سازی دیتا همچون ابزار SQL است و ارتباطی با موتور استمر ندارد؛ بنابراین، عنصر زمان تأثیر به‌سزایی در سنجش عملکرد استمرها ندارد.

معیار پیشنهادی

اگر چندجوابی در موتورهای استمر به‌عنوان یک وجه امتیاز مطرح باشد، باید در معیارهای ارزیابی نیز بازتابی روشن داشته باشد. از نظر نگارندگان، اگر چندجوابی در موتورهای استمر با نگرش منطق صرفی صورت پذیرد، ثمری برای سیستم‌های بازیابی اطلاعات نخواهد داشت. از آنجا که این سیستم‌ها به تحلیل متن‌های واقعی از علوم مختلف می‌پردازند (و نه صرف کلمات)، باید تجزیه صرفی کلمات نیز با منطق استعمالی آن علوم صورت پذیرد و نه با نگرش منطق صرفی. از این رو ، مدل پیشنهادی سازندگان استمر «نور» با توجه به ظرفیت‌های نوین تحلیل هوشمند صرفی کلمات به شرح زیر است:

$$NScore = \frac{\alpha.\sum UF_W + \beta.\sum UL_W}{\sum UF_W + \sum UL_W + \sum F_W + NULL}$$

در این مدل، مقدار UF_W تعداد استم‌های صحیح اعلام‌شده ولی مستعمل (Useful) به ازای هر کلمه و UL_W تعداد استم‌های صحیح (بر اساس منطق صرفی ولی) غیر مستعمل (Useless) به ازای آن کلمه و F_W تعداد استم‌های غلط (False) اعلام‌شده توسط موتور است. در این مدل، α و β نیز ضریب‌هایی جهت میزان سنجش دقت استمر در مورد پاسخ‌های غلط یا غیر مستعمل هستند. با تغییر هر یک از این ضرایب می‌توان دقت استمر را نسبت به هر یک از دو نگرش استعمالی و منطقی سنجید.

در موتور استمر «نور» که تنها استم مستعمل به‌عنوان استم صحیح شناخته می‌شود، ضریب β برابر صفر خواهد بود؛ ولی در سنجش دیگر استمرها که با چنین رویکردی عمل نکرده‌اند، این ضریب می‌تواند غیر صفر باشد.

نتایج ارزیابی

با استفاده از دو دادهٔ ارزیابی طلایی که معرفی گردید، امکان مقایسه استمرهای عربی بر اساس یک دادهٔ واحد فراهم خواهد بود. از میان کلمات غیرتکراری قرآن که به حسب تعداد تکرار مرتب شده‌اند، تعداد ۷ هزار کلمه که بیش از باقی کلمات تکرار داشته‌اند مورد ارزیابی استمرهای مختلف قرار گرفت. مقایسه چند استمر سبک و چند تحلیل‌گر صرفی عربی بر اساس دادهٔ ارزیابی در این ۷ هزار کلمه صحیح و با اعراب قرآن مطابق با جدول (۶) حاکی از این نتایج است:

دقت در ۷ هزار کلمه پر تکرار قرآن						
Noor	MADAMIRA	Alkhalil	Motaz	CAMeL	Light ۱۰	
۸۸٫۹٪	۵۹٪	۳۸٪	٤۲٪	۳۷٪	۳۷٪	Score

جدول (٦): مقایسه عملکرد استمرها و تحلیل‌گرهای صرفی عربی با استمر نور در داده‌ی کلمات قرآنی

در کلمات قرآن که دارای اعراب و ریشه مشخص هستند، استم نیز غیر از موارد بسیار معدودی تک‌جوابی خواهد بود؛ ازاین‌رو، تنها معیار «دقت» جهت مقایسه موتورهای استمر و تحلیل‌گر هوشمند صرفی در کلمات قرآن لحاظ شده است.

با توجه به اینکه ابزار استمر SAFAR یک فایل خروجی را در اختیار کاربران قرار نمی‌دهد، امکان سنجش پاسخ‌های این موتور بر اساس دادهٔ کلمات قرآن در مرکز نور فراهم نگشت. بر اساس مقاله خود پژوهشگرانSAFAR ، درصد دقت برخی از استمرهای سبک در تمامی کلمات قرآن به‌ترتیب زیر است:

- استمر Tashaphyne با ۱۰٫۹۵%
- استمر Light10 با ۱۴٫۹۶%
- استمر Motaz با ۱۸٫۵۹%
- استمر SAFAR با ۳۳٫۷%

اگرچه برخی از این استمرها مطابق با جدول شماره (۷) نسبت به داده طلایی ۷۰۰۰ کلمه‌ای درصد بالاتری را نشان می‌دهند، تمامی آن‌ها در مقایسه با استمر «نور» دارای اختلاف فاحشی در میزان دقت هستند. استمر «نور» با ۸۸٫۹٪ دقت دارای بیشترین میزان دقت در بین استمرهای موجود بوده است. این داده طلایی به‌صورت آنلاین بر پایگاه لغتنامه هوشمند قرار خواهد گرفت تا صحت ادعای فوق به راحتی قابل سنجش برای کاربران باشد.

در مقابل داده قرآن که یک داده صحیح و با اعراب است، می‌توان جهت مقایسه استمرها و تحلیل‌گرهای صرفی عربی در داده‌های متنی عربی کلاسیک که عموماً بدون اعراب و دارای اشکالات فونتی بسیار است، از دادهٔ ارزیابی عربی کلاسیک که مشتمل بر ۱۰ هزار کلمه اتفاقی است، استفاده نمود.

با توجه به چالش تک‌جوابی در اکثر استمرهای سبک عربی و عدم پاسخ نسبت به تعدد پاسخ‌های صحیح، از معیار پیشنهادی NScore جهت مقایسه عملکرد این موتورهای هوشمند استفاده نشده و تنها به پاسخ تک‌جوابی اکتفا شده است. نتیجه مقایسهٔ تنها ۵۱۰۰ کلمه از این دادهٔ طلایی در ۲۴ الگو مطابق با جدول شماره (۷) حاکی از دقت بالای استمر «نور» است:

تعداد کلمه	الگوی ارزیابی	Motaz	Mada mira	Light 10	Alkhalil	Noor
۴۰۰	رندم از کلمات دارای حداقل برچسب دخیل	48%	52%	55%	56%	74%
۳۰۰	دارای برچسب ادات	21%	43%	24%	55%	73%
۵۰	دارای برچسب اشتقاق غیر قیاسی	16%	22%	22%	48%	72%
۵۰	دارای برچسب رسم الخط قرآنی	2%	8%	2%	4%	78%
۵۰	دارای برچسب اسم فعل	14%	42%	16%	56%	70%
۱۰۰	ریشه اجوف (ریشه سه حرفی؛ حرف وسط واو یا یاء)	38%	41%	43%	60%	89%
۱۰۰	ریشه ناقص (ریشه سه حرفی؛ حرف آخر واو یا یاء)	15%	33%	26%	37%	58%
۱۰۰	ریشه مثال (ریشه سه حرفی؛ حرف اول واو یا یاء)	29%	48%	33%	57%	79%
۱۰۰	ریشه لفیف (ریشه سه حرفی؛ حداقل دو حرف واو یا یاء)	13%	31%	23%	31%	60%
۳۰۰	ریشه مضاعف (ریشه سه حرفی؛ دو حرف متوالی ریشه مشابه هم)	40%	43%	43%	65%	82%
۳۰۰	ریشه مهموز (ریشه سه حرفی و یکی از حروف همزه)	30%	35%	33%	53%	77%
۳۰۰	ریشه بیش از سه حرف	50%	47%	53%	67%	83%
۲۰۰	کلمه دارای یکی از حروف (أ، إ) و ریشه بدون همزه	38%	49%	43%	50%	72%
۱۰۰	کلمه دارای یکی از حروف (ئ، ؤ، ء) و ریشه بدون همزه	49%	43%	53%	64%	89%
۲۰۰	کلمه دارای یکی از حروف (و، ی) و ریشه بدون حرف عله	33%	43%	33%	48%	88%
۱۰۰	کلمه دارای حداقل دوتا از حروف (ل، ک، ب، ف، س) و ریشه فاقد همان حروف	20%	9%	21%	70%	83%
۱۰۰	کلمه دارای حداقل دوتا از حروف (ت، ک، ه، ت، و، ن، ی، م) و ریشه فاقد همان حروف	24%	35%	24%	49%	83%
۱۰۰	آخر کلمه حرف ة	0%	17%	0%	84%	82%
۵۰	آخر کلمه حرف یة	0%	10%	0%	96%	68%
۱۰۰	آخر کلمه حرف ی و ریشه سالم	21%	28%	22%	30%	67%
۲۰۰	آخر کلمه یکی از (ان، ات، ون، ین)	16%	49%	18%	34%	74%
۱۰۰۰	کلمات دارای ریشه و کمتر از ۳۰ بار تکرار	29%	31%	34.5%	58%	77%
۳۰۰	دارای بیش از دو ریشه تایید شده	30%	39%	38%	57%	75%
۵۰۰	نمونه غیر ماشینی تهیه شده از کتاب‌های صرفی توسط محققین	21%	45%	24%	42%	79%
مجموع ۵۱۰۰	میانگین درصدها	24.87%	35.12%	28.47%	52.95%	76.33%

جدول شماره (۷): الگوی ساخت یک داده‌ی معیار در سنجش عملکرد استمرهای عربی

در این داده، متناسب با میزان تکرار کلماتِ دارای یک برچسب، تعداد آن کلمه در دادهٔ ارزیابی مورد ملاحظه قرار گرفته است. به‌عنوان مثال، به علت تعداد اندک کلماتی با برچسب «رسم‌الخط قرآنی» یا «اسم فعل» تنها ۵۰ کلمه اتفاقی از کل این کلمات در دادهٔ ارزیابی آمده است. درصد دقت موتور استمر «نور» در مورد هر یک از این نوع کلمات به‌صورت تفصیلی بیان شده است تا تحلیلی دقیق از توانایی این موتور هوشمند در هر دسته از کلمات باشد. میانگین دقت در استمر نور ٪۷۶٫۳۳ بوده و به نوعی نشان از درصد تقریبی صحت پاسخ‌های استمر «نور» در یک دادهٔ متنی پیچیده و نسبتاً پر اشتباه از متون عربی کلاسیک است. باید توجه داشت اگر معیار چندجوابی و نگرش به استعمال در سنجش با دادهٔ ارزیابی در محاسبه لحاظ می‌شد، درصد دقت موتورهای دیگر به‌مراتب کاهش می‌یافت.

همان‌گونه که مشخص است، بر اساس این الگو می‌توان محکی منصفانه از عملکرد تمامی استمرها در متون علمی عربی به‌ویژه عربی کلاسیک ارائه نمود. در مورد هر یک از این کلمات، ریشه و استم با نگرش استعمالی (و نه منطق صرفی) توسط محققین مشخص شده است. درصد دقت موتور استمر «نور» نیز بر اساس همین معیار محاسبه و اعلام شده است. تفاوت استمرها در پردازش کلمات پر استعمال و کم استعمال، کلمات فاقد ریشه به‌ویژه کلمات اعلام، دخیل و معرب، کلمات دارای نوشتار خاص قرآنی، ادات‌های اسمی و حرفی، انواع کلمات معتل، کلمات مهموز و مضاعف، جمع‌های مکسر، کلمات دارای چند ریشه مختلف و همچنین انواع حالات مختلف کلمات عربی با پیشوند و پسوندهای مختلف، می‌تواند میزان دقت و نقاط قوت و ضعف هر استمر را به خوبی مشخص کند.

این دو دادهٔ ارزیابی پس از رونمایی از پایگاه لغتنامه هوشمند مرکز نور، به‌صورت آنلاین در اختیار پژوهشگران قرار خواهد گرفت.

نتیجه‌گیری

با ارتقا و پیشرفت سیستم‌های بازیابی اطلاعات، موتورهای استمر نیز پیشرفت چشمگیری در تحلیل صرفی کلمات عربی داشته و بسیاری از پژوهشگران نیز بر کارکرد بهتر این سیستم‌ها به هنگام استفاده از استم کلمات تأکید داشته‌اند. با این وجود، موتورهای استمر در زبان عربی با چالش‌هایی مواجه هستند که همچنان برخی نویسندگان (Mustafa et al, 2017: 64) را به سبب عملکرد نامطلوب این استمرها به ترجیح رویکردهای ریشه‌محور واداشته است.

در این نوشتار، پیشینه موتورهای استمر و رویکرد آن‌ها بررسی شده و با تحلیل دقیق هر یک از چالش‌های مطرح در استمرهای عربی، الگویی نوین جهت طراحی یک استمر جدید به نام استمر «نور» معرفی شد. این استمر با پیاده‌سازی قواعد پیچیده اعلال و ادغام با یک زبان ساده محاسباتی، در کنار قواعد دقیق وندیت، توانسته است چالش مربوط به حذف نادرست وندها و تغییرات کلمات را تا حد بسیار مطلوبی رفع نماید. استفاده از الگوی بررسی استعمال و دادهٔ برچسب‌گذاری شده در مرکز تحقیقات کامپیوتری علوم اسلامی موجب شده، علاوه بر حل مشکل تک‌جوابی در استمرها، پاسخ‌های متعدد یک استمر نیز بر اساس استعمال و کاربرد یک کلمه در علوم مختلف پالایش و رتبه‌بندی شود. استفاده از مداخل لغتنامه‌ها و بهره‌گیری از شبکه واژگانی عربی در حل معضل جمع‌های مکسر، گسترهٔ پردازش استمر نور را به طرز قابل ملاحظه‌ای ارتقا داده است.

نبود یک داده ارزیابی استاندارد و جهانی موجب شده است بسیاری از محققان، عملکرد استمرهای خود را بر اساس داده‌های مؤسسات داخلی خود ارزیابی کرده و معیار دقیق و واقعی از عملکرد تولیدات خود ارائه ندهند. ضرورت یک دادهٔ استاندارد طلایی(Gold) جهت محک تمامی ابزارهای هوشمند متن‌کاوی، موجب شده است که تلاش‌های لازم در ساخت داده‌های ارزیابی‌شده بیش از تلاش‌های لازم در ساخت ابزارهای بهره‌برداری کننده از این داده‌ها باشد؛ ازاین‌رو، تلاش نگارندگان در این مقاله نیز بر روش تولید دو دادهٔ ارزیابی مختلف و عرضه عمومی آن‌ها جهت مقایسه تمامی استمرها متمرکز شده است. یک دادهٔ ارزیابی از تمامی کلمات قرآن و دیگری داده‌ای مشتمل بر ۱۰ هزار کلمه مبتنی بر کتاب‌های دارای متن عربی کلاسیک و معاصر.

با توجه به امکان تولید استم‌های مختلف از یک کلمه و پالایش آن بر اساس حالات پرکاربرد و مستعمل، معیاری جدید نیز جهت سنجش عملکرد استمرها معرفی شده است. دقت بالای استمر «نور» در تشخیص استم کلمات قرآنی و همچنین تمایز این موتور هوشمند نسبت به سایر ماشین‌های هوشمند در تشخیص استم‌های متعدد یک کلمه با نگرش کاربردی و استعمالی و رتبه‌بندی آن‌ها، موجب تأثیر بهینه فرآیندهای بازیابی اطلاعات در لایه‌های نحوی و معنایی بوده و گزینه مناسبی جهت به‌کارگیری در ابزارهای متن‌کاوی است.

استمر نور در راستای ارتقای خود می‌تواند با بهره‌گیری از تکنیک‌های مختلف هوشمندسازی در لایه نحو و معنا همچون یادگیری ماشینی، شبکه‌های عصبی، هم‌نشینی کلمات و شبکه واژگان، پاسخ‌های خود را در هر متنی از علوم مختلف، پالایش نماید. گذر از عربی کلاسیک و گسترش فعالیت استمر نور به دنیای عربی معاصر با استفاده از لغتنامه‌های جدید و شبیه‌سازی پاسخ‌های صحیح در کلمات عربی کلاسیک و پیشنهاد آن در متن‌های عربی معاصر، می‌تواند به‌عنوان گام‌های پیش‌روی این موتور هوشمند در آینده‌ای نه چندان دور باشد.

در حال حاضر، از استمر نور در پایگاه لغتنامه هوشمند به‌عنوان یک پروژه عملیاتی استفاده شده و بازخورد تجارب کاربران در دستیابی به مداخل صحیح لغتنامه‌ها می‌تواند کمکی شایان به ارتقای این ماشین هوشمند نماید.

فهرست مطالب

الهی‌منش، محمدحسین (۱۳۹۴)، ابهام‌زدایی هوشمند صرفی نور، ره‌آورد نور، شماره ۵۳، ص۱۳-۱۸.

بدیل یعقوب، إمیل (۱۴۲۵)، المعجم المفصل في الجموع، دارالکتب العلمیة، بیروت: لبنان.

دانش، سید محمد (۱۳۹۳)، تحلیل‌گر هوشمند صرفی نور، شماره ۴۹، ص۱۵- ۲۳.

سریانی حبیب (۱۳۹۵)، شبکه واژگانی زبان عربی با استفاده از فرآیند نیمه خودکار در داده‌های علوم اسلامی، ره‌آورد نور، شماره ۵۷، ص۴۷-۵۶.

سریانی حبیب، مینایی بهروز (۱۳۹۰). سیستم هوشمند برچسب‌گذار ادات سخن زبان عربی؛ لایه صرف، ره‌آورد نور، شماره ۳۴، ص۱۸ - ۲۸.

مصطفی إبراهیم، الزیات أحمد حسن، عبدالقادر حامد، النجار محمد علی (۱۴۲۹). المعجم الوسیط، تهران: موسسة الصادق للطباعة و النشر.

Ababneh Mohamad, Al-Shalabi Riyad, Kanaan Ghassan, and Al-Nobani Alaa (2012). Building an Effective Rule-Based Light Stemmer for Arabic Language to Improve Search Effectiveness. The International Arab Journal of Information Technology, Vol. 9, No. 4, July 2012: Pp 368-372.

Abu Ata, B, Al-Omari, A. (2014). A Rule-Based Stemmer for Arabic Gulf Dialect. Journal of King Saud University, Science 50(2), Computer and Information Sciences (JKSU). (Submitted). DOI:10.1016/j.jksuci.2014.04.003.

Al-Fedaghi S. and F. Al-Anzi. (1989). “A new algorithm to generate Arabic root-pattern forms”. In proceedings of the 11th national Computer Conference and Exhibition. PP 391-400 .

Aljlayl, M, Frieder, O. (2002). On Arabic search: improving the retrieval effectiveness via a light stemming approach. In: Proceedings of the Eleventh International Conference on Information and Knowledge Management, McLean, VA

Al-Kabi M.N, Al-Radaideh Q.A, Akkawi K.W. (2011). Benchmarking and assessing the performance of Arabic stemmers J. Inform. Sci, Vol 37 (2), pp. 111-119

Al-Kabi Mohammed N, Kazakzeh Saif A, Abu Ata Belal M, Al-Rababah Saif A, Alsmad Izzat M. i (2015). A novel root based Arabic stemmer. Journal of King Saud University – Computer and Information Sciences (2015) 27: Pp 94–103.

Al-Serhan, H, Ayesh, A. (2006). A triliteral word roots extraction using neural network for Arabic, In: The 2006 International Conference on Computer Engineering and Systems: Pp. 436–440.

Al-Shalabi, R, Kanaan, G, Ghwanmeh, S, Nour, F. M. (2007). Stemmer algorithm for Arabic words based on excessive letter locations. In: 4th International Conference on Innovations in Information Technology (IIT '07): Pp. 456–460.

Boubas, A, Lulu, L, Belkhouche, B, Harous, S. (2011). GENESTEM: A novel approach for an Arabic stemmer using genetic algorithms. In: International Conference on Innovations in Information Technology (IIT 2011): Pp. 77–82.

Boudchiche Mohamed, Mazroui Azzeddine (2018). A hybrid approach for Arabic lemmatization. International Journal of Speech Technology, https://doi.org/10.1007/s10772-018-9528-3 .

Boudlal, A, Lakhouaja, A, Mazroui, A, Meziane, A, Ould Abdallahi Ould Bebah, M,Shoul, M. (2010). Alkhalil Morpho SYS1: a morphosyntactic analysis system for arabic texts. In: International Arab Conference on Information Technology. Benghazi, Libya: Pp 1–6.

Buckwalter T. (2007). Issues in Morphological Analysis, in Arabic Computational Morphology. Eds. A. Soudi, A. van den Bosch, and G. Neumann. Springer, 2007: Pp. 23–41

Chen, A, Gey, F.C. (2002). Building an Arabic stemmer for information retrieval. In: Proceedings of the 11th Text Retrieval Conference (TREC).

Darwish, K. (2003). Probabilistic methods for searching OCR-degraded Arabic text. Unpublished Ph.D. Thesis. University of Maryland, USA.

El-Sadany, T.A, Hashish, M.A. (1989). An Arabic Morphological System, IBM System Journal. Vol.28 (4). Pp 600-612.

Flores F.N, Moreira V.P. (2016). Assessing the impact of stemming accuracy on information retrieval Inf. Process. Manage, Vol 52 (5): Pp 840-854.

Frakes William B, Fox Christopher J. (2003). Strength and Similarity of Affix Removal Stemming Algorithms. ACM SIGIR Forum, Volume 37, No. 1: Pp 26-30 .

Galvez C, F, Anegón de Moya, Solana V.H. (2005). Term conflation methods in information retrieval: non-linguistic and linguistic approaches Journal of Document, Vol 61 (4): Pp 520-547

Goweder, A, Poesio, M, De Roeck, A. and Reynolds, J. (2005) Identifying Broken Plurals in Unvowelised Arabic Text. Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Vancouver, 6-8 October 2005: Pp 246-253

Hadni Meryeme, El Ouatik Saîd Alaoui, Lachkar Abdelmonaime (2013). Effective Arabic Stemmer Based Hybrid Approach for Arabic Text Categorization. International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.3, No.4: Pp 1-14.

Hamdy Mubarak (2017). Build Fast and Accurate Lemmatization for Arabic, Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018), Miyazaki, Japan, LREC.

J. Yaghi and S. M. Yagi (2004). Systematic Verb Stem Generation for Arabic, in Proc. of the Workshop on Computational Approaches to Arabic Script-Based Languages (Geneva, Switzerland, August 28 - 28, 2004). ACL Workshops. Association for Computational Linguistics, Morristown, NJ: Pp. 23–30.

Kazem T, Rania E, and Je.rey C. (2005). Arabic Stemming Without A Root Dictionary, International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II, USA. DOI: 10.1109/ITCC.2005.90

Khoja, S. and Garside, R. (1999). Stemming Arabic Text. Technical report, Computing Department, Lancaster University,UK Lancaster.

Larkey L, Ballesteros L. Connell M.E. (2007). Light stemming for Arabic information retrieval. Arabic Computational Morphology: Knowledge-based and Empirical Methods, Springer, Netherlands (2007): Pp 221-243

Larkey, L, Ballesteros, L, Connell, M.E. (2002). Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis. In: SIGIR'02, Tampere, Finland: Pp. 275–282.

Lovins, J.B. (1968). Development of a Stemming Algorithm. Mechanical Translation and Computational Linguistics, vol.11, nos.1 and 2, March and June 1968: Pp 22-31.

Mustafa Mohammad, Salah Eldeen Afag, Bani-Ahmad Sulieman, Osman Elfaki Abdelrahman (2017). A Comparative Survey on Arabic Stemming: Approaches and Challenges. Intelligent Information Management, Vol 9: Pp 39-67.

Mustafa Suleiman H. (2012), Word Stemming for Arabic Information Retrieval: The Case for Simple Light Stemming. ABHATH AL-YARMOUK: "Basic Sci. & Eng.", by Yarmouk University, Irbid, Jordan, Vol. 21, No. 1, 2012: Pp. 123-144.

Namly Driss, Tajmout Rachida, Bouzoubaa Karim and Abouenour Lahsen (2016). NAFIS: A Gold Standard Corpus for Arabic Stemmers Evaluation, In Proc of the 28th International Business Information Management Association Conference IBIMA, Seville, Spain. ISBN: 978-0-9860419-8-3.

Pasha Arfath, Al-Badrashiny Mohamed, Diab Mona, El Kholy Ahmed, Eskander Ramy, Habash Nizar, Pooleery Manoj, Rambow Owen, and Roth Ryan. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In LREC, vol 14: Pp. 1094-1101.

Rogati, M, McCarley, S, Yang, Y. (2003). Unsupervised learning of Arabic stemming using a parallel corpus. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1, Stroudsburg, USA.

Saad, M.K, Ashour, W. (2010). Arabic morphological tools for text mining. In: 6th International Conference on Electrical and Computer Systems (EECS'10), Lefke, North Cyprus, 2010.

Sawalha Majdi and Eric Steven Atwell. (2008). Comparative evaluation of arabic language morphological analysers and stemmers. Proceedings of COLING 2008 22nd International Conference on Computational Linguistics. Companion volume Posters and Demonstrations, Manchester: Pp 107–110.

Sembok, T. M. T, & AbuAta, B. (2013). Arabic word stemming algorithms and retrieval effectiveness. In Lecture Notes in Engineering and Computer Science.Vol. 3 LNECS, London: Pp 1577-1582.

W.B. Frakes, C.J. Fox (2003). Strength and similarity of affix removal stemming algorithms, ACM SIGIR Forum, Vol 37 (1): Pp 26-30. DOI: 10.1145/945546.945548.

Younes Jaafar, Driss Namly, Karim Bouzoubaa, Abdellah Yousfi (2017). Enhancing Arabic stemming process using resources and benchmarking tools. Journal of King Saud University - Computer and Information Sciences. Volume 29, Issue 2, April 2017: Pp 164-170.

Yusof RJR, Zainuddin R, Mohd Sapiyan Baba, Zulkifli Mohd Yusoff (2010). QUR'ANIC WORDS STEMMING, the Arabian Journal for Science and Engineering, Volume 35, Number 2C, December 2010: Pp 37-49.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, Nizar Habash (2020), An Open Source Python Toolkit for Arabic Natural Language Processing, European Language Resources Association (ELRA), Pp 7022-7032.

References

Alahi-Manesh, M. H. (2015). Intelligent Morphological Disambiguation of Noor. *Rahavard Noor*, Winter 2015, No. 53, 13-18.

Badil Yaqoub, E. (2004). *Al-Mu'jam Al-Mufassal Fi Al-Jamoo*. Dar Al-Kutub Al-Ilmiyyah, Beirut, Lebanon.

Danesh, S. M. (2014). Intelligent Morphological Analyzer of Noor. *Rahavard Noor*, Winter 2014, No. 49, 15-23.

Soryani, H. (2016). The Lexical Network of the Arabic Language Using Semi-Automatic Processes in Islamic Sciences Data. *Rahavard Noor*, Winter 2016, No. 57, 47-56.

Soryani, H., & Minaei, B. (2011). Intelligent Tagging System for Arabic Speech Elements: Morphological Layer. *Rahavard Noor*, Spring 2011, No. 34, 18-28.

Mustafa Ibrahim, Al-Ziyat, A. H., Abd Al-Qader Hamid, Al-Najjar, M. A. (2008). *Al-Mu'jam Al-Wasat*. Al-Sadiq Printing and Publishing Foundation, Tehran, Iran.

Ababneh, M., Al-Shalabi, R., Kanaan, G., & Al-Nobani, A. (2012). Building an Effective Rule-Based Light Stemmer for Arabic Language to Improve Search Effectiveness. *The International Arab Journal of Information Technology*, 9(4), 368-372.

Abu Ata, B., & Al-Omari, A. (2014). A Rule-Based Stemmer for Arabic Gulf Dialect. *Journal of King Saud University, Science*, 50(2), Computer and Information Sciences. DOI: 10.1016/j.jksuci.2014.04.003.

Al-Fedaghi, S., & Al-Anzi, F. (1989). A new algorithm to generate Arabic root-pattern forms. In *Proceedings of the 11th National Computer Conference and Exhibition*, 391-400.

Aljlayl, M., & Frieder, O. (2002). On Arabic search: Improving the retrieval effectiveness via a light stemming approach. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, McLean, VA.

Al-Kabi, M. N., Al-Radaideh, Q. A., & Akkawi, K. W. (2011). Benchmarking and assessing the performance of Arabic stemmers. *Journal of Information Science*, 37(2), 111-119.

Al-Kabi, M. N., Kazakzeh, S. A., Abu Ata, B. M., Al-Rababah, S. A., & Alsmad, I. M. (2015). A novel root-based Arabic stemmer. *Journal of King Saud University – Computer and Information Sciences*, 27, 94-103.

Al-Serhan, H., & Ayesh, A. (2006). A triliteral word roots extraction using neural networks for Arabic. In *The 2006 International Conference on Computer Engineering and Systems*, 436–440.

Al-Shalabi, R., Kanaan, G., Ghwanmeh, S., & Nour, F. M. (2007). Stemmer algorithm for Arabic words based on excessive letter locations. In *4th International Conference on Innovations in Information Technology (IIT '07)*, 456–460.

Boubas, A., Lulu, L., Belkhouche, B., & Harous, S. (2011). GENESTEM: A novel approach for an Arabic stemmer using genetic algorithms. In *International Conference on Innovations in Information Technology (IIT 2011)*, 77–82.

Boudchiche, M., & Mazroui, A. (2018). A hybrid approach for Arabic lemmatization. *International Journal of Speech Technology*. https://doi.org/10.1007/s10772-018-9528-3

Boudlal, A., Lakhouaja, A., Mazroui, A., Meziane, A., Ould Abdallahi Ould Bebah, M., & Shoul, M. (2010). Alkhalil Morpho SYS1: A morphosyntactic analysis system for Arabic texts. In *International Arab Conference on Information Technology*, Benghazi, Libya, 1–6.

Buckwalter, T. (2007). Issues in Morphological Analysis. In A. Soudi, A. van den Bosch, & G. Neumann (Eds.), *Arabic Computational Morphology* (pp. 23–41). Springer.

Chen, A., & Gey, F. C. (2002). Building an Arabic stemmer for information retrieval. In *Proceedings of the 11th Text Retrieval Conference (TREC)*.

Darwish, K. (2003). Probabilistic methods for searching OCR-degraded Arabic text. Unpublished Ph.D. thesis, University of Maryland, USA.

El-Sadany, T. A., & Hashish, M. A. (1989). An Arabic Morphological System. *IBM System Journal*, 28(4), 600-612.

Flores, F. N., & Moreira, V. P. (2016). Assessing the impact of stemming accuracy on information retrieval. *Information Processing & Management*, 52(5), 840-854.

Frakes, W. B., & Fox, C. J. (2003). Strength and similarity of affix removal stemming algorithms. *ACM SIGIR Forum*, 37(1), 26-30.

Galvez, C. F., Anegón de Moya, & Solana, V. H. (2005). Term conflation methods in information retrieval: Non-linguistic and linguistic approaches. *Journal of Documentation*, 61(4), 520-547.

Goweder, A., Poesio, M., De Roeck, A., & Reynolds, J. (2005). Identifying broken plurals in unvowelised Arabic text. In *Proceedings of Human

Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Vancouver, 246-253.

Hadni, M., El Ouatik, S. A., & Lachkar, A. (2013). Effective Arabic Stemmer Based Hybrid Approach for Arabic Text Categorization. *International Journal of Data Mining & Knowledge Management Process*, 3(4), 1-14.

Hamdy, M. (2017). Build Fast and Accurate Lemmatization for Arabic. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan.

Yaghi, J., & Yagi, S. M. (2004). Systematic Verb Stem Generation for Arabic. In *Proc. of the Workshop on Computational Approaches to Arabic Script-Based Languages*, Geneva, Switzerland, 23–30.

Kazem, T., Rania, E., & Je.rey, C. (2005). Arabic Stemming Without A Root Dictionary. In *International Conference on Information Technology: Coding and Computing (ITCC'05)* - Volume II, USA. DOI: 10.1109/ITCC.2005.90

Khoja, S., & Garside, R. (1999). Stemming Arabic Text. Technical report, Computing Department, Lancaster University, UK.

Larkey, L., Ballesteros, L., & Connell, M. E. (2007). Light stemming for Arabic information retrieval. In *Arabic Computational Morphology: Knowledge-based and Empirical Methods* (pp. 221-243). Springer.

Larkey, L., Ballesteros, L., & Connell, M. E. (2002). Improving stemming for Arabic information retrieval: Light stemming and co-occurrence analysis. In *SIGIR'02*, Tampere, Finland, 275-282.

Lovins, J. B. (1968). Development of a Stemming Algorithm. *Mechanical Translation and Computational Linguistics*, 11(1-2), 22-31.

Mustafa, M., Afag, S. E., Bani-Ahmad, S., & Abdelrahman, O. E. (2017). A Comparative Survey on Arabic Stemming: Approaches and Challenges. *Intelligent Information Management*, 9, 39-67.

Mustafa, S. H. (2012). Word Stemming for Arabic Information Retrieval: The Case for Simple Light Stemming. *Abhath Al-Yarmouk: Basic Sci. & Eng.*, 21(1), 123-144.

Namly, D., Tajmout, R., Bouzoubaa, K., & Abouenour, L. (2016). NAFIS: A Gold Standard Corpus for Arabic Stemmers Evaluation. In *Proc. of the 28th International Business Information Management Association Conference IBIMA*, Seville, Spain. ISBN: 978-0-9860419-8-3.

Pasha, A., Al-Badrashiny, M., Diab,

M., El Kholy, A., Eskander, R., Habash, N., Pooleery, M., Rambow, O., & Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for

Morphological Analysis and Disambiguation of Arabic. In *LREC*, vol 14, 1094-1101.

Rogati, M., McCarley, S., & Yang, Y. (2003). Unsupervised learning of Arabic stemming using a parallel corpus. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, Stroudsburg, USA.

Saad, M. K., & Ashour, W. (2010). Arabic morphological tools for text mining. In *6th International Conference on Electrical and Computer Systems (EECS'10)*, Lefke, North Cyprus.

Sawalha, M., & Atwell, E. S. (2008). Comparative evaluation of Arabic language morphological analysers and stemmers. In *Proceedings of COLING 2008 22nd International Conference on Computational Linguistics. Companion Volume: Posters and Demonstrations*, Manchester, 107-110.

Sembok, T. M. T., & Abu Ata, B. (2013). Arabic word stemming algorithms and retrieval effectiveness. In *Lecture Notes in Engineering and Computer Science*. Vol. 3 LNECS, London, 1577-1582.

Frakes, W. B., & Fox, C. J. (2003). Strength and similarity of affix removal stemming algorithms. *ACM SIGIR Forum*, 37(1), 26-30. DOI: 10.1145/945546.945548.

Younes, J., Namly, D., Bouzoubaa, K., & Yousfi, A. (2017). Enhancing Arabic stemming process using resources and benchmarking tools. *Journal of King Saud University - Computer and Information Sciences*, 29(2), 164-170.

Yusof, R. J. R., Zainuddin, R., Mohd Sapiyan Baba, & Zulkifli, M. Y. (2010). QUR'ANIC WORDS STEMMING. *Arabian Journal for Science and Engineering*, 35(2C), 37-49.

Obeid, O., Zalmout, N., Khalifa, S., Taji, D., Oudah, M., Alhafni, B., Inoue, G., Eryani, F., & Habash, N. (2020). An Open Source Python Toolkit for Arabic Natural Language Processing. In *European Language Resources Association (ELRA)*, 7022-7032.