مروری بر روش‌های شباهت‌سنجی متون کوتاه

احمد ربیعی‌زاده*[1]، حسین امیرخانی[2]

[1] آزمایشگاه هوش مصنوعی مرکز تحقیقات کامپیوتری علوم اسلامی نور، قم، ایران

[2] گروه مهندسی کامپیوتر و فناوری اطلاعات دانشکدهٔ فنی دانشگاه قم، قم، ایران

چکیده

مشابهت‌سنجی بین متون کوتاه یکی از نیازهای بنیادین در بسیاری از مسائل پردازش زبان طبیعی است؛ که باتوجه‌به اهمیت آن پژوهش‌گران کماکان به‌دنبال بهبود کیفیت الگوریتم‌های موجود هستند. در این مطالعه صدوپنجاه مقاله بررسی شدند و دسته‌بندی جامعی برای روش‌های موجود در زمینه مشابهت‌سنجی تک‌زبانهٔ جملات و متون کوتاه ارائه شد. به‌طورکلی روش‌های ارائه‌شده را می‌توان در سه گروه دسته‌بندی کرد. گروه نخست، روش‌هایی که بر مشابهت لفظی تمرکز می‌کنند. در این روش‌ها متن به‌عنوان رشته‌ای از نویسه‌ها یا مجموعه‌ای از کلمات یا ترکیبی از این دو درنظرگرفته‌می‌شود. گروه دوم، روش‌هایی هستند که به ارتباط معنایی کلمات نیز مبتنی‌بر پایگاه دانش یا تحلیل پیکره‌های متنی توجه دارند. در مطالعات اخیر از روش‌های یادگیری عمیق مبتنی‌بر ترنسفورمرها بهره‌برداری شده و نتایج حاکی‌از بهبود چشم‌گیر کیفیت این روش‌هاست. گروه سوم، به ترکیب روش‌های لفظی و معنایی و گاهی روش‌های تحلیل نحوی پرداخته‌اند. البته تحلیل‌گرهای نحوی باکیفیتی برای تمامی زبان‌ها نبوده و به‌کارگیری آن‌ها سرعت را نیز به‌مراتب کاهش می‌دهد. در این مطالعه، همچنین، به مهم‌ترین مجموعه دادگان موجود و روش‌های ارزیابی مشابهت‌سنجی بین متون کوتاه نیز اشاره شده‌است.

واژگان کلیدی : مشابهت معنایی متون، مشابهت لفظی، پردازش زبان طبیعی، مشابه‌یابی، بردار تعبیه جملات

A Survey on Short Text Similarity Measurement Methods

Ahmad Rabiei Zadeh*[1], Hossein Amirkhani[2]

[1] AI Laboratory of Computer Research Center of Islamic Science (Noor),

[2] Computer Engineering & Information Technology Faculty, University of Qom, Qom, Iran

Abstract

Measuring similarity between two text snippets is an essential yet challenging task in many Natural Language Processing problems. Various methods have been proposed over the past years to measure text similarity. This survey reviews more than 150 of these papers, classifies them into three categories, and discusses their pros and cons. The first category includes lexical methods that only focus on the surface similarity of the text pairs. In edit-based lexical methods, texts are considered as a sequence of characters; while in set-based and vector-based methods, they are treated as tokens. There are some methods that consider texts as a mixture of characters and tokens as well. Some recent studies use modern deep learning techniques for detecting lexical similarity in alias detection and near-duplicate detection tasks. The edit-based method is a good choice for similarity detection at the word and phrase-level, but it is not suitable for sentence similarity calculation. The hybrid method outperforms other methods in lexical similarity measurement because of taking care of similarity in both word and character levels. The second category includes semantic methods that take into consideration the meaning of the words using either a pre-prepared knowledge base like WordNet or corpus-based methods. These methods define sentence-level similarity as an aggregation of word-level similarities.

* نویسندهٔ عهده‌دار مکاتبات

* Corresponding author

• تاریخ ارسال مقاله: ۱۴۰۱/۱/۳۱ • تاریخ پذیرش: ۱۴۰۲/۹/۵ • تاریخ انتشار: ۱۴۰۲/۱۰/۲۴ • نوع مطالعه: پژوهشی

[DOI: 10.61186/jsdp.20.3.103]

Some knowledge-based methods calculate the word similarity based on the length of the path between the two words in the knowledge base, while some recent studies use novel WordNet embedding methods. The high cost of creating and maintaining the knowledge base is one of the main disadvantages of these methods. On the other hand, corpus-based methods calculate the words relatedness based on the distributional hypothesis. The count-based and predict-based approaches are some representative methods in this category. Recent studies show that using modern deep learning techniques like Transformers and Siamese networks to create a contextual word/document embedding outperform other methods in the semantic similarity measurement. The final category includes the hybrid methods that take advantage of different methods. Some methods in this category use syntactic parsing. However, high-quality syntactic parsers are not present for many languages and using them has some side effects on the overall speed of the system. Furthermore, this study also identifies the main datasets and evaluation methods that can be utilized for the short text and sentence similarity measurement.

Keywords: short text similarity, lexical similarity, semantic similarity, natural language processing, sentence embedding, transformer

۱- مقدمه

روش‌های مشابهت‌سنجی متون، میزان شباهت میان دو متن را به‌صورت یک کمیت عددی بیان کرده و بدین‌ترتیب امکان مقایسه شباهت بین متون را فراهم می‌کنند. روش‌های مشابهت‌سنجی باتوجه‌به ویژگی‌های مختلف متن ورودی و نوع نیاز، گونه‌های مختلفی دارند. برخی از این ویژگی‌ها عبارتند از واحد مشابهت (کلمه/ جمله و متن کوتاه در حد بند / متن بلند)، یکسان‌بودن زبان دو متن ورودی (زبان یکسان/ بین‌زبانی)، هم‌گن‌بودن دو متن (شباهت‌سنجی عبارت کوئری جستجو با اسناد متنی از نوع غیر هم‌گن است) و پشتیبانی از زبان‌های متعدد متون (تک‌زبانه/ چندزبانه). نوع مشابهت (لفظی/ معنایی/ ساختاری).

مشابهت لفظی عبارت است از شباهت ظاهری و سطحی دو متن باتوجه‌به میزان نویسه‌ها و کلمات مشترک آنها و در نقطهٔ مقابل، مشابهت معنایی عبارت است از نزدیکی مفهوم دو متن بر اساس میزان واژگان هم‌معنا یا مرتبط موجود در دو متن. برای نمونه، دو جمله زیر را در نظر بگیرید:

جملهٔ اول: «*هنگامی که آنان را موجی همچون سایبان فراگیرد، خداوند را مخلصانه بخوانند.*»[۱] جملهٔ دوم: «*چون به مردم گزندی برسد، به سوی پروردگارشان روی آورده و او را می‌خوانند.*» [۲] این جملات از نظر لفظی شباهت بسیار پایینی دارند، اما درعین‌حال از شباهت معنایی و مفهومی بالایی برخوردار هستند. در این مطالعه شباهت لفظی و معنایی بین جملات و متون کوتاه تک‌زبانه و هم‌گن بررسی خواهندشد. مشابهت بین دو جمله اغلب به‌طورکلی در نظر گرفته‌می‌شود؛ اما به‌تازگی، به این نکته نیز توجه شده که دو متن ممکن است از جنبه‌هایی مشابه باشند، ولی درعین‌حال از جنبه‌های دیگری نیز متفاوت به‌نظربرسند[۱].

مشابهت‌سنجی جملات و متون کوتاه یکی از نیازهای زیربنایی برای بسیاری از مسائل پردازش زبان‌های طبیعی به‌شمارمی‌آید. برخی کاربردهای اصلی مشابه‌یابی متون عبارتند از رده‌بندی و خوشه‌بندی متون[۳] [۳] ,[۲]، سامانه‌های پیشنهاددهندهٔ محتوا[۴] [۴]، خلاصه‌سازی متن [۵]، شناسایی سرقت علمی[۵] [۶]، سامانه‌های پرسش و پاسخ [۷]، بازیابی اطلاعات [۸]، ارزیابی خودکار پاسخ‌ها[۶] [۹]، اعتبارسنجی ادعا در شبکه‌های اجتماعی[۷] [۱۰] . مشابه‌یابی علاوه‌بر کاربرد در مسائل مختلف پردازش زبان طبیعی و دامنهٔ محتوای عمومی، در دامنهٔ محتوای تخصصی نیز کاربردهای ویژه‌ای دارد. برای نمونه در متون تخصصی علمی، محتوای پزشکی و حقوقی.

باتوجه‌به گستردگی کاربردهای مشابهت‌سنجی متون، روش‌های متعددی برای آن ارائه شده و به همین دلیل تقسیم‌بندی‌های مختلفی نیز برای انواع روش‌های مشابهت‌سنجی بین جملات و متون کوتاه نیز ارائه شده‌است [۱۷]–[۱۱]. طبق بررسی‌های انجام‌شده، جامع‌ترین شیوهٔ تقسیم‌بندی این روش‌ها در (شکل- ۱) ارائه شده‌است. در ادامه به توضیح انواع این روش‌ها می‌پردازیم.

۲- روش‌های مشابهت‌سنجی لفظی

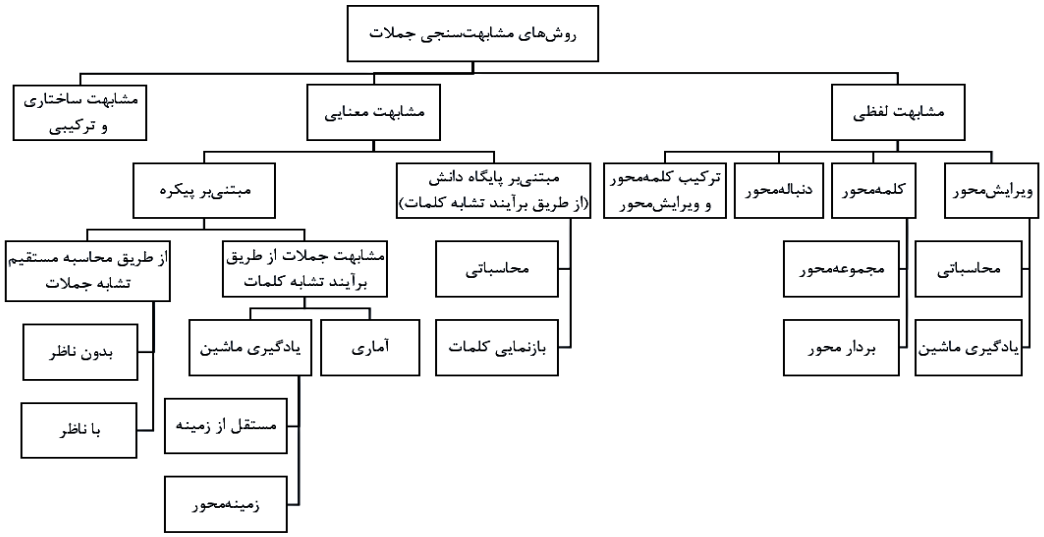
مشابهت‌سنجی لفظی از جمله ساده‌ترین روش‌های مشابهت‌سنجی بین دو متن است که با استفاده از آن میزان شباهت سطحی دو متن بر اساس شباهت مجموعه

[۱] ترجمهٔ آیه ۳۲ سورهٔ لقمان
[۲] ترجمهٔ آیه ۳۳ سورهٔ روم

[3] Text Classification and Clustering
[4] Recommender System
[5] Plagiarism Detection
[6] Automatic Short Answer Scoring
[7] Social Media Claims Verification

[DOI: 10.61186/jsdp.20.3.103]

نویسه‌ها، مجموعه کلمات سازنده یا دنبالـه نویسـه‌های آن دو جمله تعیین می‌شود.



(شکل- ۱)- دسته‌بندی جامع انواع روش‌های مشابهت‌سنجی متون کوتاه

(Figure- 1). Comprehensive taxonomy of short text similarity approaches

از مزایای روش‌های مشابهت‌سـنجی لفظـی، سـادگی پیاده‌سازی، عدم‌نیاز بـه داده‌هـای جـانبی از قبیـل پایگـاه دانش معنایی ارتباط واژگـان و پیکره‌هـای حجـیم متنـی، مستقل از زبـان‌بودن و قابلیـت اسـتفاده بـرای حوزه‌هـای کاربردی مختلف است. بااین‌حال ضعف اصلی این روش‌هـا عدم توانایی در شناسایی شباهت‌های معنایی بـین واژگـان دو جمله است. روش‌های مشابهت‌سنجی لفظی، گونـه‌های مختلفـی دارنـد؛ از قبیـل روش‌هـای ویرایش‌محـور[1]، کلمه‌محور، دنباله‌محور[2] و ترکیب دو روش ویرایش‌محور و کلمه‌محور. در (شکل- ۱(۲))، انواع ایـن روش‌هـا به‌صـورت دسته‌بندی‌شده، ارائه شده‌اند.

۱-۲- روش‌های ویرایش‌محور

روش‌های ویرایش‌محور به دو دسته تقسیم می‌شوند. دستهٔ اول روش‌هایی هستند که با استفاده از روابط سادهٔ ریاضی به محاسبهٔ شباهت دو جمله می‌پردازند، و دستهٔ دوم از روش‌های یادگیری ماشین استفاده می‌کنند. در مشابهت‌سنجی ویرایش‌محور، دو عبارت موردنظر به‌عنوان رشته‌ای از نویسه‌ها درنظرگرفته‌می‌شوند و تعداد عملیات لازم برای ویرایش یک رشته و تبدیل آن به رشتهٔ دیگر از طریق روش‌های مختلف شمارش می‌شود. به برخی از روش‌های متداول در ادامه اشاره می‌شود.

فاصله همینگ[3][۱۸]: بر اساس شمارش تعداد نویسه‌هایی در دو رشته با طول برابر که مقادیر آنها متفاوت است؛ به عبارت دیگر، تعداد عمل‌های جایگزینی موردنیاز برای تبدیل یک رشته به رشته دیگر شمارش می‌شود. فاصله دِیمرالونشتاین[4][۱۹]: بادرنظرگرفتن عملیات مختلفی از قبیل درج، حذف، جابه‌جایی یا جایگزینی، دست‌کم عملیات موردنیاز برای تبدیل یک رشته به رشتهٔ دیگر شمارش می‌شود که در نسخه‌های پیشرفته‌تر برای عملیات مختلف، وزن متفاوتی درنظرگرفته‌می‌شود. معیار شباهت جارُو[5][۲۰]: شباهت دو رشتهٔ متنی با درنظرگرفتن تعداد نویسه‌های مشترک و ترتیب آنها محاسبه می‌شود. در نسخهٔ توسعه‌یافته جارُو-وینکلر[6]، به نویسه‌های مشترک در آغاز رشته اهمیت بیشتری داده می‌شود.

معیار شباهت نیدِلمن-وانچ[7] [۲۱]: بر اساس یافتن بهترین حالت برای هم‌ترازی نویسه‌های یک رشته با رشتهٔ دیگر مبتنی‌بر روش‌های برنامه‌نویسی پویاست. این روش برای مقایسه رشته‌های DNA و دنباله‌های زیست‌شناختی با طول یکسان طراحی شده‌است.

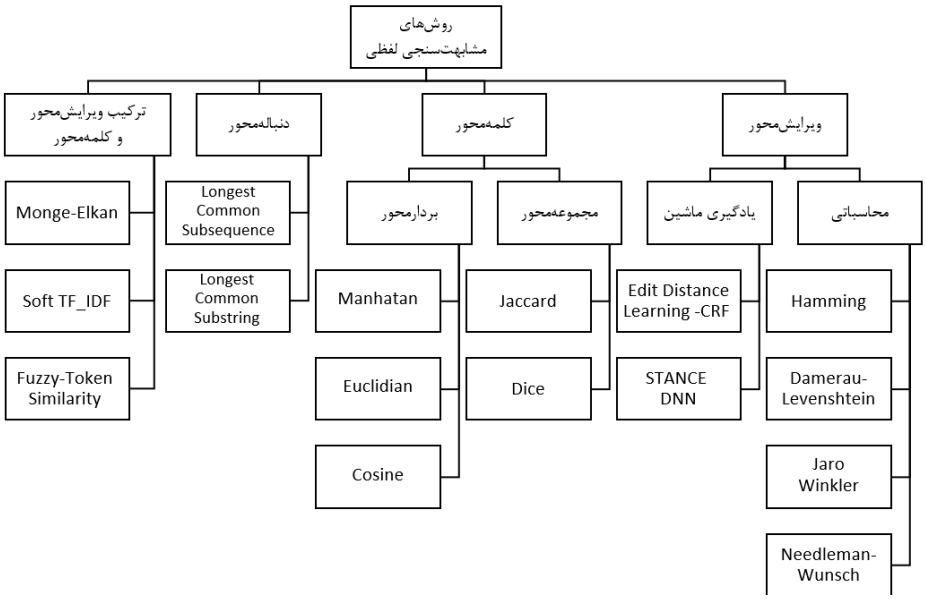
از سوی دیگر، روش‌های یادگیری ماشین نیز در محاسبه مشابهت‌سنجی لفظی به‌کارگرفته‌شده‌اند. در مقالهٔ [۲۲] از روش SVM برای وزن‌دهی بهینه عمل‌های ویرایش متفاوت استفاده شده‌است.

[1] Edit based
[2] Sequence based

[3] Hamming
[4] Damerau-Levenshtein
[5] Jaro
[6] Jaro–Winkler
[7] Needleman-Wunsch

[DOI: 10.61186/jsdp.20.3.103]

در مقالهٔ [۲۳]، با به‌کارگیری روش یادگیری ماشین کلاسیک CRF، مسألهٔ موردنظر در قالب ماشین حالات نامتناهی الگو شده‌است.



(شکل- ۲): روش‌های مختلف مشابهت‌سنجی لفظی
(Figure- 2): Short text lexical similarity approaches

مقدار تابع هزینه انتقال، مبتنی‌بر ویژگی‌هایی از قبیل امتیاز هریک از عملیات ویرایش، حالت قبلی و بعدی، موقعیت در رشته و... محاسبه می‌شود. خروجی نهایی بر روی پنج مجموعه‌دادهٔ واقعی و یک مجموعه‌دادهٔ مصنوعی با استفاده از معیار F ارزیابی شده و حاکی از ۰.۱۵ درصد بهبود دقت در مشابهت‌سنجی مجموعه داده نام رستوران‌هاست. پس‌از روش‌های یادگیری سنتی، روش‌های یادگیری عمیق نیز در مسأله مشابهت‌سنجی لفظی به‌کارگرفته‌شده‌اند. از جمله مقالهٔ [۲۴] که با ارائهٔ روشی تحت عنوان STANCE، با استفاده از شبکه عصبی BiLSTM، نویسه‌های یک رشته متنی در قالب جدیدی بازنمایی می‌شوند و پس‌از تشکیل ماتریس مشابهت بین نویسه‌های دو رشته، با ضرب داخلی بردارهای تعبیهٔ نویسه‌ی[1]، نرخ هم‌ترازی بین نویسه‌های دو رشته بر اساس مسألهٔ انتقال (با هدف تبدیل یک ورودی به دیگری با کمترین هزینه) محاسبه می‌شود. در پایان، امتیاز نهایی با استفاده از شبکهٔ CNN دوبعدی محاسبه می‌شود. این روش در مسألهٔ شناسایی عنوان‌های دیگرِ موجودیت‌های اسمی[2] ارزیابی شده و گزارش ارائه‌شده حاکی‌از بهبود کیفیت خروجی است.

۲-۲- روش‌های کلمه‌محور

روش‌های نویسه‌محور برای جملات و متون بلند، دقت و کارایی مناسبی ندارند. در مقابل، در روش‌های کلمه‌محور، دو متن به زیررشته‌ها یا توکن‌های کوتاه‌تری مثل کلمات یا دنباله‌های n- نویسه[3] شکسته‌می‌شوند و سپس مشابهت بین دو جمله از طریق یکی از روش‌های مشابهت‌سنجی بین مجموعه‌ها[4] یا بردارهای حاصل از کلمات محاسبه می‌شود. به برخی از این روش‌ها در ادامه اشاره شده‌است.

برای نمونه از روش‌های مجموعه‌محور در معیار جاکارد[5]، نسبت تعداد توکن‌های مشترک به تعداد غیرتکراری کل توکن‌ها و در معیار دایس[6]، دوبرابر تعداد توکن‌های مشترک به تعداد کل توکن‌ها. به‌عنوان معیار تشابه درنظرگرفته‌می‌شود. از جمله روش‌های بردار محور نیز در معیار مَنهتن[7]، فاصلهٔ بلوکی و در معیار اقلیدسی، فاصله بر مبنای فیثاغورس و در معیار فاصلهٔ کسینوسی نیز زاویهٔ بین دو بردار محاسبه می‌شود. همچنین، در الگوهای پیشرفته‌تر با تأثیر وزن متفاوت برای کلمات مختلف، از طریق روش‌هایی مثل TF-IDF می‌توان بازنمایی واقعی‌تری از متن در قالب بردار ارائه کرد. برای نمونه در یکی از مطالعات با استفاده از روش VSM و

[1] Character Embedding
[2] Alias Detection Task

[3] N-gram
[4] Set-based
[5] Jaccard
[6] Dice
[7] Manhattan

[DOI: 10.61186/jsdp.20.3.103]

وزن‌دهی TF-IDF و محاسبهٔ فاصلهٔ کسینوسی بر روی مجموعه داده MSRP به دقت ۷۰.۳٪ دست یافته‌اند [۲۵]. در مقالهٔ دیگری در زمینهٔ مشابهت‌سنجی دادهٔ توصیفی رشته‌های دانشگاهی به زبان عربی، استفاده از دنبالهٔ دوکلمه‌ای‌ها و فاصلهٔ کسینوسی، کارایی بهتری داشته و دقت ۸۷٪ گزارش شده‌است [۲۶].

به‌طورکلی در روش‌های مشابهت‌سنجی لفظی انجام پیش‌پردازش‌های اولیه روی متن تأثیر محسوسی در کیفیت نهایی مشابهت‌سنجی خواهدداشت. برای مثال، یکی از مطالعات به بررسی نقش فرایندهای پیش‌پردازش از قبیل ریشه‌یابی و بن‌یابی بر روی مشابهت‌یابی لفظی داده عربی پرداخته‌است و بررسی‌ها روی دادهٔ عربی SemEval2017 حاکی‌از بهبود بیش از هفت‌درصدی دقت نهایی بوده‌است [۲۷].

۳-۲- روش‌های دنباله‌محور

در این روش‌ها، همانند روش‌های ویرایش‌محور، متن به‌عنوان رشته‌ای از نویسه‌ها درنظرگرفته‌می‌شود؛ با این تفاوت که به‌جای محاسبهٔ اختلاف دو رشته بر اساس تعداد عملیات ویرایش، امتیاز مشابهت مبتنی‌بر طول زیردنبالهٔ مشترک بین دو رشته محاسبه می‌شود. معیار شباهت طولانی‌ترین زیررشتهٔ مشترک[1] [۲۸] و همچنین، معیار طولانی‌ترین زیردنبالهٔ مشترک[2] [۲۹]، دو نمونه از این روش‌ها هستند. در روش دوم، زیردنباله‌های غیرمتوالی نیز در امتیاز مشابهت دخالت داده می‌شوند.

۴-۲- ترکیب روش‌های ویرایش‌محور و کلمه‌محور

روش‌های کلمه‌محور به تغییر کلمات حساس هستند؛ به نحوی که اختلافات جزئی در کلمات نیز موجب عدم‌دخالت آنها در فرایند مشابهت‌سنجی خواهدشد. از همین روی، تلاش‌هایی انجام شده تا بتوان با ترکیب روش‌های ویرایش‌محور و کلمه‌محور از مزیت هر دو روش بهره برد. در این روش‌ها، نخست، جملات به‌عنوان مجموعه‌ای از کلمات درنظرگرفته‌می‌شوند. سپس، بین هریک از کلمات جملهٔ اول با کلمات جملهٔ دوم، مشابهت نویسه‌ی محاسبه می‌شود و در نهایت برآیند تشابه دو جمله به یکی از روش‌هایی که در ادامه آمده‌است، محاسبه می‌شود.

روش مونگه-اِلکان[3] [۳۰]: در این روش، میانگین مجموع بیشینهٔ مشابهت ویرایش‌محور هریک از کلمات جملهٔ اول با کلمات جملهٔ دوم بر اساس رابطهٔ (۱) تعیین می‌شود. A و B ناظر به دو متن ورودی و i و j نیز به‌ترتیب، مشخص‌کنندهٔ اندیس کلمات آنها هستند.

$$match(A,B) = \frac{1}{|A|}\sum_{i=1}^{|A|} \max_{j=1} match(A_i, B_j) \quad (۱)$$

روش Soft-tfIDF [۳۱]: در این روش مطابق با رابطهٔ (۲) به‌ازای همهٔ کلمات متن اول و دوم که شباهت آنها بیشتر از حد آستانهٔ θ باشد، مجموع حاصل‌ضرب وزن کلمه از متن اول در وزن کلمه از متن دوم، در بیشینهٔ امتیاز مشابهت کلمه موردنظر از جملهٔ اول با کلمات جملهٔ دوم به‌عنوان امتیاز نهایی مشابهت لفظی دو متن درنظرگرفته‌می‌شود.

$$SoftTFIDF(S,T) = \sum_{w \in CLOSE(\theta,S,T)} weight(w,s).weight(w,T).D(w,T) \quad (۲)$$

روش مشابهت Fuzzy-Token [۳۲]: در این روش، کلمات با شباهت بیشتر از حد آستانهٔ موردنظر، در رابطهٔ شباهت‌سنجیِ توکن‌محور (از نوع روابط مبتنی‌بر مجموعه‌ها) شرکت داده‌می‌شوند. برای نمونه، فرمول جاکارد فازی در رابطهٔ (۳) ارائه شده‌است. نماد δ نشان‌دهندهٔ حد آستانهٔ شباهت موردنظر و T و T` به‌ترتیب نشان‌دهندهٔ توکن‌های دو جملهٔ ورودی S و S` هستند.

$$FJACCARD_{\delta}(S,S') = \frac{|T \tilde{\cap} T'|}{|T| + |T'| - |T \tilde{\cap}_{\delta} T'|} \quad (۳)$$

۳- روش‌های مشابهت‌سنجی معنایی

در روش‌های مشابهت‌سنجی لفظی، به اشتراک معنایی و اشتراک لفظی بین کلمات توجه نمی‌شود. **اشتراک معنایی**، حالتی است که کلمات متفاوت به‌معنای واحدی اشاره کنند. مانند دو کلمه «انسان» و «بشر» که از نظر مفهومی یکسان هستند، ولی روش‌های مشابهت‌سنجی لفظی، این مشابهت را پوشش نمی‌دهند. **اشتراک لفظی**، نیز حالتی است که یک کلمهٔ واحد، معانی متعددی داشته‌باشد. مانند کلمه «شیر» که باتوجه‌به جملهٔ موردنظر به یکی از مفاهیم لبنیات|حیوان|وسیله دلالت می‌کند؛ که روش‌های مشابهت‌سنجی لفظی، این نکته را نیز پوشش نمی‌دهند و این قبیل کلمات را یکسان درنظرمی‌گیرند. برخلاف روش‌های مشابهت‌سنجی لفظی، اشتراک معنایی در تمامی روش‌های مشابهت‌سنجی معنایی، پوشش داده‌شده و اشتراک لفظی نیز در بسیاری

[1] Longest Common Substring
[2] Longest Common subsequence
[3] Monge-Elkan

[DOI: 10.61186/jsdp.20.3.103]

از این روش‌ها پشتیبانی می‌شود. انواع روش‌های مشابهت‌سنجی معنایی در تقسیم‌بندی جامع ارائه‌شده در (شکل- ۱) قابل‌مشاهده هستند. در ادامه، به توضیح هرکدام از این روش‌ها و نمونه‌های شاخص هرکدام پرداخته خواهدشد.

۱-۳- روش‌های مبتنی‌بر پایگاه دانش کلمات

در این گروه از روش‌ها، نخست، مشابهت کلمات بین دو جمله با یکدیگر بر پایهٔ پایگاه دانش معنایی کلمات، محاسبه شده و سپس برآیند میزان تشابه تمامی کلمات، به‌عنوان امتیاز مشابهت نهایی آن دو جمله درنظرگرفته‌خواهدشد.

پایگاه دانش معنایی کلمات، هستان‌شناسی واژگان یا آنتولوژی لغوی عبارت است از ساختاری سلسله‌مراتبی از واژگان یک زبان مشخص، که روابط معنایی بین کلمات از قبیل شمول، ترادف و تضاد به‌صورت صریح در آن مشخص شده‌اند. پایگاه دانش معنایی باتوجه‌به کاربرد، انواع مختلفی دارند. برخی از آنها حاوی واژگان رایج زبان هستند و کاربرد عمومی داشته و برخی دیگر به‌طور صرف، شامل دایرهٔ لغات تخصصی بوده و به‌صورت خاص‌منظوره طراحی می‌شوند. مثل شبکهٔ واژگان پزشکی و حقوق.

پایگاه دانش معنایی وُردنت[1] [۳۳] ویژهٔ واژگان زبان انگلیسی است. نمونه‌های دیگری نیز برای سایر زبان‌ها تهیه شده‌است؛ از جمله فارس‌نت [۳۴] برای زبان فارسی و AWN [۳۵] برای زبان عربی. در روش‌های مرسوم مشابهت‌سنجی معنایی دو متن با استفاده از پایگاه دانش، نخست، مشابهت معنایی هریک کلمات متن اول با کلمات متن دوم سنجیده‌می‌شود و پس‌از انتخاب بهترین کلمهٔ متناظر از جملهٔ دوم به‌ازای هر کلمه از جملهٔ اول، امتیازهای تشابه با استفاده از روش‌های مختلفی، تجمیع[2] شده و برآیند مشابهت معنایی در سطح متن محاسبه می‌شود.

برای تعیین میزان شباهت معنایی دو کلمه با استفاده از پایگاه دانش معنایی کلمات روش‌های مختلفی به‌کارگرفته‌شده‌است. برخی روش‌ها مبتنی‌بر شمارش یال‌ها هستند. در این روش‌ها نخست، دو کلمه در شبکهٔ کلمات، جانمایی می‌شوند و سپس، بر اساس فاصلهٔ یال‌های بین دو گره مرتبط در شبکه، میزان فاصلهٔ معنایی بین دو کلمه مشخص می‌شود [۳۶]. در برخی روش‌ها به ارتباط میزان مشابهت با عمق گره‌ها، توجه شده‌است [۳۷]،[۳۸]؛ طوری‌که فاصلهٔ یکسان در عمق بیشتر شبکه، دارای ارزش مشابهت بیشتری نسبت‌به همان فاصله در عمق کم‌تر خواهدبود؛ چرا که عمق بیشتر حاکی از اشتراک مفهومی بیشتری بین دو کلمه خواهدبود. برای نمونه دو کلمه «اسب» و «گوره‌خر» که در عمق پایین شبکهٔ کلمات هستند، مشابهت بسیار بیشتری نسبت به دو کلمهٔ «حیوان» و «گیاه» در عمق بالاتری دارند.

چالش مشابهت‌سنجی بر اساس این روش‌ها این است که یال‌ها به‌طور الزامی، دلالت بر فاصلهٔ معنایی یکسانی ندارند ازاین‌رو معیار دقیقی برای مشابهت‌سنجی نخواهدبود. برای حل این مشکل، برخی روش‌ها مبتنی‌بر مفهوم بار اطلاعاتی[3]، مشابهت معنایی را تعریف کرده‌اند. به‌عنوان نمونه، معیار رِزنیک[4]امتیاز بار اطلاعاتی را بر اساس شاخص IDF کلمه در وردنت یا یک پیکرهٔ خارجی درنظرگرفته و بار اطلاعاتی نزدیک‌ترین والد مشترک دو کلمه را به‌عنوان سنجه‌ای برای مشابهت معنایی دو کلمه تعریف می‌کند[۳۹]. در معیار لین[5]، علاوه‌بر بار اطلاعاتی نزدیک‌ترین والد مشترک، بار اطلاعاتی هر دو کلمه نیز در رابطه مشابهت دخالت داده‌شده‌اند[۴۰]. برخی دیگر از روش‌ها مثل لِسک[6] نیز از طریق شمارش کلمات مشترک در تعاریف دو کلمه و کلمات مرتبط با آنها در وردنت، میزان مشابهت را محاسبه می‌کنند[۴۱]. گروه دیگری نیز با ترکیب روش‌های قبلی، از امتیاز بار اطلاعاتی یا عمق والد مشترک برای وزن‌دهی یال‌ها در فرمول شمارش یال‌ها پرداخته‌اند [۴۳][۴۲].

از سوی دیگر به‌تازگی نیز روش‌هایی مبتنی‌بر بازنمایی کلمات پایگاه دانش شکل گرفته‌اند. برای نمونه روش Wnet2Vec شبکه وردنت را در قالب ماتریس مجاورت بازنمایی کرده و سپس با محاسبهٔ شباهت برای گره‌های غیر مستقیم، ماتریس را غنی‌تر کرده، و پس‌از کاهش ابعاد با استفاده از روش PCA از طریق محاسبهٔ فاصلهٔ کسینوسی بین بردار مربوط به دو کلمه، شباهت معنایی بین آن دو را محاسبه می‌کند[۴۴]. روش Word2Set نیز به‌ازای هر کلمه، تمامی مجموعه کلمات همسایه در WordNet را درنظرگرفته و با محاسبهٔ امتیاز اشتراک آنها و ترکیب آن با امتیاز حاصل از یادگیری مبتنی‌بر SVR به تعیین امتیاز نهایی شباهت معنایی می‌پردازد[۴۵].

[1] WordNet
[2] Scale-up
[3] Information Content
[4] Resnik
[5] Lin
[6] Lesk

[DOI: 10.61186/jsdp.20.3.103]

در ادامه، به نمونه مقالاتی که برای مشابهت‌سنجی معنایی جملات از پایگاه‌های دانش استفاده کرده‌اند، اشاره خواهدشد. در بسیاری از مقالات به ترکیب معیارهای مشابهت مبتنی‌بر پایگاه دانش پرداخته شده‌است؛ از جمله مقاله‌ای که در آن شش معیار مشابهت از معیارهای فاصلهٔ شبکه واژگان را بر روی پایگاه دانش وردنت، ترکیب کرده و سپس، بر اساس امتیاز IDF به وزن‌دهی امتیازهای مشابهت کلمات پرداخته و در پایان، بر اساس مجموع بیشینهٔ امتیاز مشابهت به‌ازای تمامی کلمات جملهٔ اول با کلمات جملهٔ دوم، امتیاز مشابهت دو جمله را محاسبه می‌کند[۴۶]. این روش روی داده MSRP ارزیابی شده و دقت ۷۰.۳ درصد را کسب کرده‌است. در مقالهٔ دیگری از معیار لین در شبکهٔ وردنت استفاده شده و پس‌از تشکیل بردار خاص اسم و فعل در دو جمله و محاسبه فاصلهٔ کسینوسی جداگانه برای بردارهای هم‌نوع، ضمن درنظرگرفتن ترتیب کلمات به محاسبهٔ امتیاز نهایی می‌پردازد. این روش روی دادهٔ خبری CMU ارزیابی شده و دقت ۸۶.۸٪ را کسب کرده‌است[۴۷]. در مطالعه‌ای دیگر از معیار طول مسیر در وردنت استفاده شده و سپس مشابهت‌های هر کلمه از جملهٔ اول با تمامی کلمات جملهٔ دوم تجمیع شده و در نهایت، از فاصلهٔ کسینوسی برای محاسبه میزان مشابهت استفاده شده‌است. ضریب همبستگی پیرسون[1] این روش روی تعاریف داده RG65 به میزان ۸۰.۷٪ گزارش شده‌است [۴۸]. همچنین، در یکی از مطالعات علاوه‌بر استفاده از وردنت، راهکاری با عنوانFUZE برای مشابهت‌سنجی دقیق‌تر کلمات فازی مبتنی‌بر هستی‌شناسی (آنتولوژی) اصطلاحات فازی ارائه شده‌است [۴۹]. در این روش، ۳۰۹ کلمهٔ فازی در شش گروه مختلف برای مفاهیمی از قبیل اندازه، فاصله، سن و تعداد درنظرگرفته‌شده‌است. این روش توانسته نرخ همبستگی پیرسون روی مجموعه داده MWFD را تا ۳٪ بهبود دهد.

۲-۳- روش‌های مبتنی‌بر تحلیل پیکره متنی

استفاده از پایگاه دانش برای کشف میزان ارتباط معنایی کلمات، در کنار برتری خود، دو اشکال عمده دارد؛ چراکه برای بسیاری از زبان‌ها و حوزه‌های تخصصی هنوز پایگاه دانش معنایی جامع و دقیقی در دسترس نیست و در صورت وجود نیز برای پشتیبانی از لغات و اصطلاحات جدید، باید این پایگاه‌های دانش را به‌صورت مستمر به‌روز نگه داشت؛ که این مسأله فرایند پرچالش و پرهزینه‌ای است. در مقابل، روش‌هایی وجود دارند که به‌جای استفاده از پایگاه دانش، ارتباط معنایی بین کلمات را مبتنی‌بر نظریه توزیع، با تحلیل پیکره‌های متنی انبوه شناسایی می‌کنند. برای اولین بار پروفسور فیرث[2]، این نظریه را در سال ۱۹۵۰ بدین شکل بیان کرد: *هر کلمه را می‌توان با مجموعه کلمات پیرامون آن توصیف کرد*[۵۰] . در این روش، کلمات مرتبط بر اساس تحلیل پیکرهٔ متنی حجیم شناسایی می‌شوند. کلمات مرتبط کلماتی هستند که به‌طور صرف، در حوزهٔ کاربردی مشترکی به‌کاربرده‌می‌شوند و نرخ باهم‌آیی بالایی دارند ولی به‌طور لزوم، مفهوم یکسانی ندارند؛ مانند چای و قند. این روش‌ها برخلاف شبکهٔ معنایی، مستقل از زبان و مستقل از حوزهٔ کاربردی خاص هستند و با تغییر زبان پیکرهٔ ورودی می‌توان از این روش برای زبان‌های مختلف استفاده کرد.

بااین‌حال، این روش نقاط ضعفی نیز دارد. تحلیل دقیق پیکرهٔ متنی جامع با حجم انبوه نیازمند توان پردازشی بالایی است. از سوی دیگر، کیفیت و تنوع محتوای پیکره کماکان مسأله است؛ چراکه هنوز تعریف مشخصی از ویژگی‌های یک پیکره با کیفیت ارائه نشده‌است. مشکل تداخل معانی[3] برای کلمات چندوجهی چالش دیگر این روش است. برای نمونه کلمه چندوجهی «شیر» را در نظر بگیرید. این کلمه با کلمهٔ «پلنگ» بسامد هم‌رخدادی بالایی دارد. از طرف دیگر، با کلمهٔ «قهوه» نیز ممکن است به تعداد بالا دیده شده‌باشد. بنابراین، دو کلمه «پلنگ» و «قهوه» نیز به‌عنوان کلمات مرتبط تلقی خواهندشد؛ درحالی‌که این دو کلمه هیچ ارتباطی از نظر مفهومی با یکدیگر ندارند.

مشابهت‌سنجی جملات بر پایهٔ این روش‌ها به دو دسته تقسیم می‌شود. از طریق *برآیند تشابه معنایی کلمات* و از طریق *روش‌های تولید مستقیم بردار جملات*. در دستهٔ اول، نخست، مشابهت کلمات دو جمله با یکدیگر محاسبه شده و سپس برآیند میزان تشابه تمامی کلمات آن‌ها، به‌عنوان امتیاز مشابهت آن دو درنظرگرفته خواهدشد. در دستهٔ دوم به‌صورت مستقیم با بردار کلمات سروکار نداریم؛ بلکه نخست، هریک از جملات در قالب یک بردار، بازنمایی شده و در نهایت، با مقایسهٔ فاصله بردارهای مربوط به دو جمله، میزان مشابهت آن دو به یکدیگر محاسبه خواهدشد.

[1] Pearson

[2] Firth

[3] Meaning Conflation Deficiency

[DOI: 10.61186/jsdp.20.3.103]

۱-۲-۳- مشابهت‌سنجی در سطح جملات از طریق برآیند تشابه کلمات

روش‌های مشابهت‌سنجی معنایی کلمات بر مبنای تحلیل پیکره‌ها را می‌توان به دو دسته تقسیم کرد؛ *روش‌های مبتنی‌بر محاسبات آماری* که با تحلیل باهم‌آیی کلمات بر اساس روش‌های آماری، کلمات مرتبط را شناسایی می‌کنند و *روش‌های مبتنی‌بر روش‌های یادگیری ماشین* که اغلب به تولید بردار تعبیه برای هریک از کلمات می‌پردازند. در ادامه به توضیح هرکدام از این روش‌ها و معرفی نمونه‌های شاخص هرکدام پرداخته خواهدشد.

۱-۱-۲-۳- ایجاد بردار تعبیه مبتنی‌بر محاسبات آماری

روش HAL در سال ۱۹۹۶ با مطرح‌کردن *آبرفضای متناظر با زبان*[1]، از جمله نخستین روش‌ها برای بازنمایی کلمات پیکره‌های متنی در قالب بردارهای عددی است که بر اساس نظریهٔ توزیع به شناسایی ارتباطات معنایی بین کلمات می‌انجامد[۵۱]. در این روش، نخست، *ماتریس باهم‌آیی کلمات* از محتوای صدوشصت‌میلیون‌کلمه‌ای متون خبری تشکیل می‌شود. سلول‌های این ماتریس، بیانگر شدت باهم‌آیی دو کلمه است؛ طوری‌که کلمات نزدیک‌تر از شدت باهم‌آیی بالاتری برخوردار خواهندبود. پس‌از آن از طریق حذف ستون‌های با آشفتگی (آنتروپی) کم، ابعاد این ماتریس کاهش داده‌می‌شوند. در نهایت نیز امتیاز مشابهت دو کلمه، بر اساس محاسبهٔ فاصلهٔ اقلیدوسی یا بلوکی بین دو بردار مربوط محاسبه خواهدشد. ارزیابی این سامانه از طریق بررسی کیفی کلمات همسایه و کلمات هم‌نوع (حیوانات، جای‌ها و اعضای بدن) انجام گرفت.

در روش دیگر با نام LSA [2]به تحلیل ارتباط معانی نهفته بین کلمات پرداخته می‌شود [۵۲]. در این روش نخست، ماتریس کلمه-بند تشکیل شده و سپس با استفاده از روش فاکتورگیری SVD[3] ابعاد آن کاهش داده‌می‌شود. در نهایت، با استفاده از فاصلهٔ کسینوسی می‌توان به مقایسهٔ بردارهای مربوط به کلمات پرداخت. کیفیت خروجی این روش در تست‌های انتخاب کلمات مترادف آزمون TOEFL ارزیابی شده و دقت ۶۴.۴٪ گزارش شده‌است. در یکی از مطالعات، از روش LSA برای مشابهت‌سنجی متون کوتاه استفاده شده و در دادهٔ RG65 به ضریب همبستگی پیرسون به‌میزان ۸۳.۸٪ دست یافته‌اند [۵۳]. همچنین، در مطالعه‌ای دیگر در مورد به‌کارگیری LSA روی دادهٔ MSRP معیار F به‌میزان ۸۱.۸٪ گزارش شده‌است[۵۴]. برخی دیگر از روش‌ها از جمله PMI-IR [4] [۵۵] با رویکردی متفاوت، ارتباط معنایی دو کلمه را بر اساس تعداد کلمات مشترک در نتایج جستجوی پیشرفته موتور آلتاویستا[5] تعریف کرد. و در دادهٔ کلمات مترادف آزمون TOEFL به دقت ۷۴٪ رسید. روش NGD [6] [۵۶] نیز مشابه با همین رویکرد را بر اساس نتایج مشترک در موتور جستجوی گوگل پیاده کرده‌است.

روش‌های آماری که تاکنون گفته‌شدند، مبتنی‌بر تولید بردار کلمه بودند. علاوه‌بر این، روش‌هایی نیز هستند که بدون نیاز به تولید بردار کلمات، به‌طور مستقیم، به تولید بردار جملات می‌پردازند؛ از جمله روش ESA [7] [۵۷] که با استفاده از مفاهیم استخراج‌شده از ویکی‌پدیا و وزن‌دهی IDF به بازنمایی متن در قالب بردارهایی با ابعاد بالا می‌پردازد و در نهایت، از طریق فاصلهٔ کسینوسی بردارهای دو متن را مقایسه می‌کند. در امتیاز پیرسون این روش روی دادهٔ خبری ABC به‌میزان ۷۲٪ گزارش شده‌است. در مقالهٔ [۵۴]، روش LDA [8] که برای الگوسازی توزیع موضوعی[9] اسناد متنی ارائه شده‌بود[۵۸]، در مسألهٔ مشابهت‌سنجی کلمات و متون به‌کارگرفته‌شده و روی داده MSRP کیفیتی نزدیک به روش LSA گزارش شده‌است.

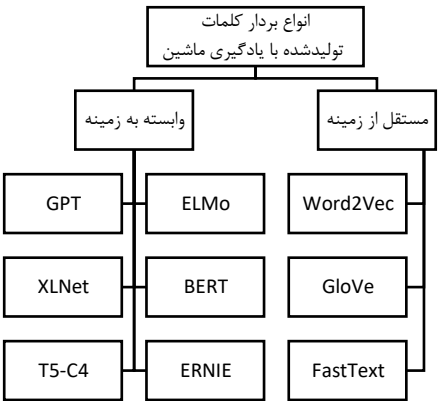
۲-۱-۲-۳- ایجاد بردار تعبیه مبتنی‌بر یادگیری ماشین

برای استخراج روابط معنایی از پیکره‌های متنی، روش‌های مبتنی‌بر یادگیری ماشین نیز به‌کارگرفته‌شدند و بدین ترتیب امکان تحلیل دقیق‌تر پیکره‌های با حجم بالاتر بیش‌ازپیش فراهم شد. از این روش‌ها برای ساخت ***بردار تعبیه در سطح کلمات***[10] و همچنین، ***بردار تعبیه برای متون***[11] استفاده شده‌است. روش‌های تولید بردار تعبیه در سطح کلمات را می‌توان به دو دسته تقسیم کرد. روش‌های مستقل از زمینه[12] و روش‌های وابسته به زمینه[13]. در (۳) انواع روش‌های تولید بردار کلمات مبتنی‌بر یادگیری

[1] Hyperspace Analogue to Language
[2] Latent Semantic Analysis
[3] Singular Values Decomposition
[4] Pointwise Mutual Information
[5] AltaVista
[6] Normalized Google Distance
[7] Explicit Semantic Analysis
[8] Latent Dirichlet Allocation
[9] Topic Modeling
[10] Word embedding
[11] Document embedding
[12] Context-free methods
[13] Contextual methods

[DOI: 10.61186/jsdp.20.3.103]

ماشین ارائه شده‌است. در ادامه به مرور روش‌های مربوط به هریک از این انواع می‌پردازیم.



(شکل- ۳): انواع روش‌های تولید بردار کلمات با یادگیری ماشین

(Figure- 3): Word embedding approaches based on machine learning techniques

۱-۲-۱-۲-۳- تولید بردار کلمه به‌صورت مستقل از زمینه

در این‌گونه روش‌ها با استفاده از روش‌های یادگیری ماشین پیش‌بینی‌کننده[1] از طریق پیاده‌سازی الگوهای زبانی بر پایهٔ شبکه‌های عصبی[2] و یا روش‌های یادگیری مبتنی‌بر شمارش[3]، در نهایت، به‌ازای هر کلمه یک بردار تعبیه ارائه می‌شود.

الگوی Word2Vec در سال ۲۰۱۳ توسط پژوهشگران شرکت گوگل ارائه شد [۵۹]. این روش مبتنی‌بر الگوهای‌های زبانی بر پایهٔ شبکه‌های عصبی است که برای اولین بار در سال ۲۰۰۳ مطرح شده‌بود [۶۰]. دو نوع الگو در این روش، پیشنهاد شده‌است؛ الگوی CBOW[4] که با داشتن کلمات پیرامون، کلمهٔ هدف را پیش‌بینی می‌کند و الگوب اسکیپ-گرام[5] که با داشتن کلمهٔ موردنظر به پیش‌بینی کلمات پیرامون می‌پردازد. برای فرایند آموزش از داده شش میلیاردکلمه‌ای اخبار گوگل استفاده شد. البته بانک واژگان تنها یک‌میلیون کلمهٔ پربسامد را مبنای کار قرار داده و بهبودهایی نیز برای موازی‌سازی فرایند یادگیری پیاده شده‌است. برای ارزیابی در مسألهٔ شناسایی کلمات متناسب[6] از دادهٔ هجده‌هزارتایی کلمات مرتبط معنایی و نحوی که در قالب ۱۴ دسته از جمله پایتخت کشورها، کلمات مذکر و مؤنث و کلمات مخالف استفاده شده‌است. بر اساس روش Word2Vec الگوهایی نیز برای سایر زبانها تولید شده‌است؛ از جمله Persian-Word2Vec برای زبان فارسی و AraVec [61] و ArWordVec [۶۲] برای زبان عربی. همچنین، از این روش برای ساخت بردار تعبیه در کاربردهای خاص نیز بهره‌برداری شده‌است. مانند الگوی Dict2Vec [۶۳] برای بازنمایی مدخل‌های لغت‌نامه و الگوی Imam [64] برای بازنمایی کلمات دادگان حدیثی.

الگوی GloVe [7] در سال ۲۰۱۴ توسط پژوهشگران دانشگاه استنفورد، ارائه شد [۶۵]. این روش از یادگیری بدون ناظر و مبتنی‌بر شمارش، برای تولید بردار تعبیه استفاده کرده‌است. به‌طوری‌که علاوه‌بر استفاده از پنجرهٔ محلی کلمات پیرامون از ماتریس هم‌رخدادی سراسری کلمات نیز بهره برده‌است. دادهٔ آموزشی مورداستفاده، محتوای چهل‌وهشت‌میلیاردکلمه‌ای حاصل از Wikipedia2014، Gigaword5 و محتوای کراول‌شدهٔ وب بوده‌است. این روش در مسألهٔ کلمات متناسب، مشابهت کلمات و شناسایی موجودیت‌های اسمی ارزیابی شده‌است.

الگوی FastText در سال ۲۰۱۶ توسط پژوهشگران فیس‌بوک ارائه شد[۶۶]. این روش بر اساس الگوی اسکیپ‌گرام و بازنمایی هر کلمه به‌عنوان مجموعه‌ای از زیردنباله‌های نویسه‌ی آن[8] شکل گرفته‌است. به‌طوری‌که بردار هر کلمه از میانگین بردارهای مربوط به نویسه‌های آن ساخته‌می‌شود. به همین دلیل، گونه‌های مختلف صرفی کلمات، بردارهای نزدیک به هم خواهندداشت و کلمات دیده‌نشده یا OOV [9] نیز بر اساس اجزای داخلی خود، واجد بردار خواهندشد. نتایج ارزیابی ارائه‌شده روی دادهٔ انسانی کلمات مرتبط و داده کلمات متناسب، نشان‌دهندهٔ بهبود مناسب دقت به‌وسیلهٔ این روش هستند.

مقالاتی که با استفاده از برآیند بردار کلمات به مشابهت‌سنجی جملات و متون کوتاه پرداخته‌اند، در جدول (۱) معرفی می‌شود.

۲-۲-۱-۲-۳- تولید بردار کلمه به‌صورت وابسته به زمینه

یکی از مشکلات استخراج روابط معنایی مبتنی‌بر نظریه توزیع از پیکره‌ها، پدیدهٔ تداخل معانی برای کلمات چندوجهی است. برای حل این مشکل، روش‌های وابسته

[1] Predictive Methods
[2] Neural Language Models
[3] Count-Based Methods
[4] Continuous Bag of Words
[5] Skip-gram
[6] Word Analogy

[7] Global Vectors for Word Representation
[8] N-gram
[9] Out Of Vocabulary

[DOI: 10.61186/jsdp.20.3.103]

به زمینه باتوجه‌به معانی مختلف هر کلمه یا به‌عبارت دقیق‌تر، باتوجه‌به مجموعه کلمات پیرامون هر کلمه، می‌توانند بردارهای تعبیهٔ متعددی تولید کنند. در ادامه به بررسی هریک از این روش‌ها می‌پردازیم.

(جدول-۱) - مقالات مشابهت‌سنجی در سطح جملات مبتنی‌بر استفاده از بردار کلمات

(Table-1). Sentence similarity measurement methods based on word embeddings

نویسنده	سال	بردار کلمات	روش برآیندگیری از بردارهای کلمات	توضیحات	دادهٔ ارزیابی
Kusner [۶۷]	۲۰۱۵	Word2Vec	برآیند فاصله کمینه Word Mover Distance		مسألهٔ دسته‌بندی اسناد
Kenter [۶۸]	۲۰۱۵	Word2vec و Glove	مجموع بیشینهٔ مشابهت کلمات	وزن‌دهی به روش BM25 و به‌کارگیری SVC	MSRP
He [۶۹]	۲۰۱۶	GloVe و PARAGRAM-SL999	ترکیب فاصلهٔ کسینوسی- اقلیدسی	به‌کارگیری یک لایهٔ میانی برای تمرکز روی زوج کلمات مشابه و ۱۹ لایه CNN	STS2014، MSRVID و WikiQA
Wang [۷۰]	۲۰۱۶	Word2Vec	فاصلهٔ کسینوسی با سه روش مجموع، بیشینه و امتیاز محلی	تشکیل ماتریس کلمات مشابه و غیرمشابه به‌ازای هر کلمه و ترکیب آنها با استفاده از CNN	QASent و WikiQA
Moatez [۷۱]	۲۰۱۷	Word2Vec	مجموع بردارهای کلمات	وزن‌دهی بر اساس TF-IDF و استفاده از برچسب قِسم کلمه	۷۵۰ مورد از ترجمه عربی MSR-Vid
M-MaxLSTM-CNN [۷۲]	۲۰۱۸	word2vec، fastText، GloVe، Baroni و SL999	ترکیب پنج بردار برای هر کلمه و تجمیع با CNN و LSTM	مقایسه در سطوح مختلف کلمه و جمله	STSB و SICK-R
Mahmoud[۷۳]	۲۰۱۹	Word2Vec	میانگین بردار کلمات	استفاده از شبکه CNN	OSAC به زبان عربی
DCRN [۷۴]	۲۰۱۹	GloVe یا Word2vec	هم‌ترازی عناصر دو جمله با Attentioned RNN	ترکیب هشت شبکه با تنظیمات اولیه تصادفی	مسألهٔ تشخیص متون معادل در دادهٔ سوالات Quora
DRSASM [۷۵]	۲۰۲۰	Word2Vec	به‌کارگیری LSTM برای استخراج ویژگی‌ها مبتنی‌بر شبکه Siamese	یادگیری تقویتی و الگوی Attention شبکهٔ Policy برای شکستن جمله به واحدهای کوچک‌تر	SNLI و Chinese Car Description
Moghadam[۷۶]	۲۰۲۱	Word2Vec	میانگین بردار کلمات، LSTM و CNN	بررسی جداگانهٔ هر سه روش	Persian movie subtitles
Moravvej [۷۷]	۲۰۲۱	Word2Vec و Glove	BLSTM with Attention Mechanism	-	STS-B
Mahmoud [۷۸]	۲۰۲۱	Glove	RCNN	-	دادهٔ ساختگی مقالات بازنویسی‌شده OSAC به عربی

الگوی GPT [1] در سال ۲۰۱۸ توسط پژوهشگران شرکت OpenAI ارائه شد [۸۰]. در این روش برای نخستین بار از معماری ترنسفورمر[2] که در سال ۲۰۱۷ توسط پژوهشگران گوگل ارائه شده‌بود، استفاده شد [۸۱]. معماری ترنسفورمر شامل یک بخش رمزگذار و یک بخش رمزگشاست که توسط زیرساخت توجه[3]، به یکدیگر متصل شده‌اند و بدین ترتیب، علاوه‌بر هدفمندترکردن فرایند یادگیری، با فراهم‌ساختن امکان موازی‌سازی، سرعت یادگیری را به‌مراتب افزایش دادند.

الگوی GPT-1 روی داده ELMo و جملات پیکرهٔ هفت‌هزار کتابی آموزش و کیفیت آن در مسائل مختلفی از جمله مشابهت‌سنجی و تحلیل احساسات ارزیابی شد، که نشان‌دهندهٔ بهبود مناسبی در بیشتر مجموعهٔ دادگان مختلف این مسائل است. پس‌از GPT-1، GPT-2 در سال ۲۰۱۹ [۸۲] و GPT-3 در سال ۲۰۲۰ [۸۳] ارائه شدند.

[1] Generative Pretraining Transformer
[2] Transformer
[3] Attention

[DOI: 10.61186/jsdp.20.3.103]

این سه الگو به‌ترتیب حاوی ۱۱۷ میلیون، ۱.۵ میلیارد و ۱۷۵ میلیارد شاخص محاسباتی هستند. برای استفاده از این الگو در سطح جمله نیز، الگوی SGPT [۸۴] ارائه شده که به‌صورت‌خاص در مسألهٔ جستجوی معنایی[1] متون به‌کارگرفته شده‌است.

الگوی BERT [2] در سال ۲۰۱۹ توسط پژوهشگران گوگل ارائه شد [۸۵]. در این الگو از یک ترنسفورمر با رمزگذار دوسویه استفاده شده‌است. در مرحلهٔ پیش‌آموزش، کلمات به‌صورت تصادفی ماسک شده و الگو، پیش‌بینی آنها را آموزش می‌بینند. این الگو با استفاده از دادهٔ ۳.۳ میلیون کلمه‌ای پیکره کتب و ویکی‌پدیا انگلیسی آموزش داده شده‌است. در مرحلهٔ تنظیم دقیق، الگو برای یک مسألهٔ خاص NLP آموزش می‌بینند. پس‌از ارائهٔ موفق الگو BERT، تلاش‌های متعددی برای بهبود سرعت و کیفیت آن انجام شد که منتهی به ارائهٔ گونه‌های جدیدی از آن شد. از جمله RoBERTa [3] توسط فیس‌بوک [۸۶]، DistilBERT توسط هاگینگفیس[4][۸۷]، ALBERT [5] توسط پژوهشگران گوگل [۸۸] و الگوی tBERT [۸۹] که به ترکیب الگوسازی موضوعی با BERT پرداخت.

علاوه‌بر این با تغییر محتوای آموزشی، الگوهای مختلفی برای پشتیبانی از یک زبان خاص یا حالت چندزبانه نیز ارائه شد. برای نمونه گروه اینسپشن[6] در وظیفهٔ شمارهٔ دو از مجموعه چالش‌های مسابقات NSURL-2019 برای مشابهت‌سنجی سوالات عربی توانست با استفاده از الگوی MBERT [7] به مقام دوم دست پیدا کند [۹۰]. همچنین، الگوهای ParsBERT [۹۱] برای زبان فارسی و AraBERT [۹۲]، CAMeLBert [۹۳] و ARBERT [94] نیز برای زبان عربی ارائه شده‌اند. برخی مطالعات به بررسی کارایی الگوهای موجود برای زبانی عربی پرداخته‌اند. از جمله جدیدترین این مطالعات می‌توان به مقالهٔ [۹۵] اشاره کرد که به معرفی نه الگوی مبتنی‌بر ترنسفورمر برای زبان عربی و مقایسه آنها در سه مسأله برچسب‌گذاری ادات سخن در داده ArabicTreebank، برچسب‌گذاری ادات سخن در متون محاوره‌ای و شناسایی گونه‌های مختلف زبان محاوره‌ای پرداخته‌است. از جمله کاربردهای خاص الگوی ARBERT می‌توان به شناسایی آیات مرتبط قرآن بر اساس دادهٔ QurSim اشاره کرد [۹۶]. مبتنی‌بر الگوی BERT، الگوهایی نیز برای کاربردهای خاص ارائه شدند؛ از قبیل SciBERT برای محتوای علمی [۹۷]، BioBERT برای متون زیست‌پزشکی [۹۸] و LEGAL-BERT برای متون حقوقی [۹۹].

از جمله جدیدترین الگوهای بهبودیافته نیز می‌توان به الگوی PromptBert [۱۰۰] از گروه پژوهشی مایکروسافت اشاره کرد که از طریق پیاده‌سازی زیرساخت پرامپت[8] در معماری شبکه آموزشی [۱۰۱]، جملهٔ ورودی را به‌عنوان یک زیرجمله از جمله‌ای مرکب، در قالب‌های از پیش تعریف‌شده[9] درنظرمی‌گیرد، سپس تلاش می‌کند به تولید بردار تعبیه از طریق پیش‌بینی کلمه ماسک‌شده بپردازد و بدین‌ترتیب کارایی بردار تعبیه جملات را بهبود دهد. همچنین، در برخی مطالعات اخیر، ثابت شده که روش‌های یادگیری متضاد[10] و توجه به نمونه‌های منفی[11]، تأثیر به‌سزایی در کیفیت بردارهای تعبیه نهایی خواهندداشت [۱۰۳] ,[۱۰۲].

الگوی ERNIE [12] در سال ۲۰۱۹ توسط پژوهش‌گران بایدو ارائه شد [104]. در این الگو علاوه‌بر تک‌کلمه‌ای‌ها، چندکلمه‌ای‌هایی که اشاره به عناوین موجودیت‌های اسمی و یا عبارات خاص می‌کنند نیز، ماسک می‌شوند. در نسخهٔ تکامل‌یافته با نام ERNIE2 که در سال ۲۰۲۰ منتشر شد، ویژگی‌های ساختار متن مثل ترتیب و فاصله جملات نیز در بازنمایی دخالت داده می‌شوند و فرایند پیش‌آموزش به‌صورت مستمر با افزایش تدریجی مسأله‌ها انجام می‌شود [۱۰۵].

الگوی XLNet در سال ۲۰۱۹ توسط پژوهشگران گوگل ارائه شد [۱۰۶]. این الگو مبتنی‌بر معماری Transformer-XL است که بر اساس جای‌گشت مختلف و نامرتب کلمات به تولید الگوی زبانی نهایی می‌پردازد.

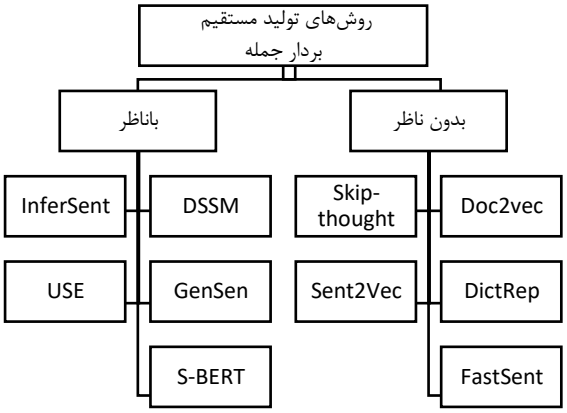
الگوی T5 [13] نیز در سال ۲۰۱۹ توسط پژوهشگران گوگل ارائه شد [۱۰۷]. این الگوی جامع، قادر است تمام مسائل متنی موجود را به قالب متن-به-متن تبدیل کرده و از آنها پشتیبانی کند. محتوای ورودی این الگو، پیکرهٔ عظیم پالایش‌شدهٔ شبکه C4[14] بوده‌است که بالغبر ۷۵۰ گیگابایت حجم دارد.

[1] Semantic Search
[2] Bidirectional Encoder Representations from Transformers
[3] Robustly Optimized BERT Approach
[4] HuggingFace
[5] A Lite BERT
[6] Inception
[7] multilingual BERT
[8] Prompt
[9] Template
[10] Contrastive Learning
[11] Negative Sample
[12] Enhanced Representation through Knowledge Integration
[13] Text-to-Text Transfer Transformer
[14] Colossal Clean Crawled Corpus

[DOI: 10.61186/jsdp.20.3.103]

۲-۲-۳- روش‌های تولید مستقیم بردار جملات

همان‌طورکه در آغاز بخش قبل گفته‌شد، برخی روش‌های مشابهت‌سنجی به‌جای تجمیع بردار کلمات و استفاده از آنها به‌عنوان بازنمایی نهایی جملات، به‌صورت مستقیم، بردار جملات را تولید کرده و به مقایسهٔ آنها می‌پردازند. این روش‌ها به دو دسته روش‌های تولید بردار جمله مبتنی‌بر یادگیری بدون ناظر و یادگیری باناظر تقسیم می‌شوند. در (شکل-)، انواع روش‌های بررسی‌شده برای این دو دسته ارائه شده‌اند.



(شکل- ۴): انواع روش‌های تولید مستقیم بردار جمله
(Figure- 4): Direct sentence embedding approaches

۱-۲-۲-۳- تولید بردار جملات به‌صورت بدون ناظر

الگوی Doc2Vec یا ParagraphVector در سال ۲۰۱۴ توسط پژوهشگران گوگل ارائه شد [۱۰۸]. در این روش، بازنمایی متن با ترکیب دو نوع بردار فشرده انجام می‌شود. بردار PV-DM که حاصل از پیش‌بینی کلمات بعدی متن است؛ و بردار PV-DBOW که حاصل از پیش‌بینی تمام کلمات یک بند متنی است. نویسنده، با ارزیابی این الگو روی دادهٔ اسناد مرتبط با یک‌میلیون جستجو (کوئری) کاهش ۳۲٪ نرخ خطا را گزارش کرده‌است. از الگوی Doc2Vec در کاربردهای متنوعی استفاده شده‌است؛ از قبیل مشابه‌یابی مفهومی اشعار فارسی [۱۰۹]، مشابه‌یابی بین آیات قرآن[۱۱۰] و شناسایی احادیث مشابه [۱۱۱] که البته ارزیابی استانداردی برای آن ارائه نشده‌است.

در **الگوی Skip-Thought** در سال ۲۰۱۵ تلاش شده تا نظرِ SkipGram الگوریتم Word2Vec برای جملات شبیه‌سازی شود [۱۱۲]؛ طوری‌که الگو با داشتن جملهٔ موردنظر، جملات پیرامون آن را پیش‌بینی کند. ارزیابی انجام‌شده روی داده SICK نشان‌دهندهٔ دقتی در حد الگوهای پیچیده مبتنی‌بر تحلیل نحوی و گرامر وابستگی جملات است. ضعف این الگو، سرعت آموزش پایین آن است که **الگوی FastSent** با انجام بهبودهایی سرعت آن را تا صدوپنجاه‌برابر بهبود داد [۱۱۳].

در **الگوی DictRep** که در سال ۲۰۱۶ ارائه شد، پس‌از آموزش شبکهٔ RNN جهت انطباق متن مربوط به معنای یک عبارت از لغت‌نامه با بازنمایی عبارت موردنظر بردار مربوط به عبارات مداخل لغت‌نامه تولید می‌شود [۱۱۴]. این الگو در مسألهٔ لغت‌نامه معکوس و حل جدول متقاطع کلمات ارزیابی شده‌است.

در **الگوی Sent2Vec** در سال ۲۰۱۸، بردارهای جملات بر اساس میانگین بردارهای کلمات، تک‌گرام[1] و دوگرام[2] با استفاده از نسخهٔ تغییریافته C-BOW تولید شدند و پس‌از آموزش بر اساس دادهٔ متن کتب، محتوای ویکی‌پدیا و توییتر (X) روی داده‌های مختلفی از قبیل SICK، STS2014 و MSRP فرایند ارزیابی انجام شد [۱۱۵].

۲-۲-۲-۳- تولید بردار جملات به‌صورت باناظر

در **الگوی DSSM** [3] که در سال ۲۰۱۶ توسط پژوهشگران مایکروسافت ارائه شد، اسناد متنی و کوئری‌های مرتبط با آنها در فضای برداری بازنمایی می‌شوند و در نهایت، ارتباط اسناد به کوئری‌ها با فاصله‌سنجی بین بردارهای مربوط اندازه‌گیری می‌شود [۸]. آموزش این شبکه به‌صورت باناظر و بر اساس بیشینه‌سازی احتمال کلیک‌شدن نتیجه به‌ازای کوئری بر مبنای دادهٔ کلیک کاربران انجام می‌شود. کیفیت این الگو در مسأله رتبه‌بندی اسناد شبکه در مجموعهٔ شانزده‌هزارتایی از کوئری‌ها و ۱۵ نتیجهٔ بازیابی‌شده به‌ازای هر کوئری، براساس معیار NDCG [4] ارزیابی شده‌است.

الگوی InferSent در سال ۲۰۱۷ توسط پژوهشگران شرکت فیس‌بوک ارائه شد [۱۱۶]. ایدهٔ اصلی این الگو، مبتنی‌بر این است که کیفیت استنتاج زبان طبیعی[5]، مبتنی‌بر شناسایی مطلوب روابط معنایی بین جملات است؛ پس می‌تواند گزینه مناسبی برای یادگیری با نظارت بردار تعبیه جملات باشد. یادگیری اولیه با استفاده از شبکهٔ BiLSTM و مبتنی‌بر مجموعه داده SNLI انجام شده و سپس فرایند انتقال یادگیری به ۱۲ مسألهٔ کاربردی از قبیل مشابهت معنایی و تشخیص متون معادل انجام شده و دقت خروجی روی داده SICK و STS ارزیابی

[1] Uni-gram
[2] Bi-gram
[3] Deep Structured Semantic Models
[4] Normalized Discounted Cumulative Gain
[5] Natural Language Inference

[DOI: 10.61186/jsdp.20.3.103]

شده‌است. گونهٔ جدیدتر این الگو نیز با نام EFL به‌تازگی منتشر شده‌است که کارایی بسیار مناسبی در مسألهٔ مشابهت‌سنجی از خود نشان داده‌است.

الگوی GenSen [1] در سال ۲۰۱۸، روشی برای تولید بازنمایی چندمنظوره از متن بر اساس یادگیری باناظر در مسائل مختلف ارائه کرده‌است [۱۱۷]. این الگو، مبتنی‌بر ترکیب یادگیری در چهار مسألهٔ اصلی پیش‌بینی جملات پیرامون، ترجمهٔ ماشینی، تجزیه‌گر نحوی و استنتاج زبان طبیعی با استفاده از معماری GRU بر پایهٔ الگوی رمزگذار و رمزگشاست و کارایی آن در مجموعه داده‌های مختلف بررسی شده‌است.

الگوی USE [2] در سال ۲۰۱۸ توسط تیم پژوهشی گوگل ارائه شده‌است [۱۱۸]. هدف این الگو، تولید رمزگذاری استانداردی برای استفاده به‌صورت یادگیری انتقالی در سایر مسائل است. در این روش دو نوع الگوی مختلف پیشنهاد شده‌است. الگوی نخست، بر مبنای استفاده از معماری ترنسفورمر و برآیند کلمات بر افزایش دقت تمرکز داشته؛ در حالی‌که الگوی دوم با استفاده از شبکه DAN[3] بر افزایش بیشینگی سرعت تمرکز دارد. این روش از سه مسألهٔ اصلی زبان طبیعی یعنی پیش‌بینی جملات پیرامون، پیشنهاد پاسخ در مکالمات و رده‌بندی جملات پشتیبانی می‌کند. ارزیابی این الگو روی داده‌SemEval-2017 انجام شده و همچنین، شرکت گوگل از نسخه‌ای از این الگو در سرویس Talk-to-Book[4] به منظور جستجوی معنایی در متون کتب، استفاده کرده‌است.

الگوی Sentence-BERT در سال ۲۰۱۹ ارائه شده و به یادگیری تابع مشابهت با استفاده از شبکه سیامی[5] و ساختار سه‌قلو بر پایه BERT و در نهایت به تنظیم دقیق[6] الگو با استفاده از داده NLI می‌پردازد [۱۱۹]. هنگام به‌کارگیری این الگو، برخلاف حفظ دقت، سرعت مشابهت‌سنجی نیز در مقایسه با روش‌های دیگر به‌شدت افزایش پیدا کرده‌است؛ به‌طوری‌که طبق گزارش‌ها زمان این فرایند در میان مجموعه ده‌هزارجمله‌ای، از ۶۵ ساعت در الگوهای BERT و RoBERTa به پنج ثانیه با استفاده از این روش رسیده‌است. کارایی این الگو روی مجموعه دادهٔ STS بررسی شده و حاکی از بیست درصد بهبود نسبت به روش‌های مبتنی‌بر میانگین‌گیری از بردارهای BERT به‌ازای کلمات است. در برخی مطالعات به ترکیب این‌گونه الگوها پرداخته‌می‌شود؛ از جمله [۱۲۰] که برای بهبود کیفیت بردار تعبیه جملات به ترکیب دو الگوی باناظر SBERT (مبتنی‌بر برچسب‌های استلزام منطقی) با DefSent [۱۲۱] (مبتنی‌بر تعریف کلمات لغت‌نامه) می‌پردازد و در مسألهٔ مشابهت‌سنجی معنایی متون، کارایی مثبت آن را نشان می‌دهد.

۴- روش‌های مشابهت‌سنجی ترکیبی

در بخش‌های پیشین، انواع روش‌های موجود برای مشابهت‌سنجی لفظی و معنایی ارائه شده و به مزایا و معایب هرکدام اشاره شد. دستهٔ سوم از روش‌های مشابهت‌سنجی جملات، روش‌هایی هستند که برای بهره‌گیری از مزایای انواع روش‌های لفظی و معنایی به ترکیب آن‌ها با یکدیگر پرداخته‌اند. در برخی از مطالعات نیز علاوه‌بر روش‌های سابق از روش‌های مبتنی‌بر تحلیل ساختار نحوی جملات نیز استفاده شده‌است؛ البته تحلیل‌گرهای نحوی باکیفیتی برای تمامی زبان‌ها وجود نداشته و به‌کارگیری آن‌ها سرعت نهایی فرایند مشابهت‌سنجی را نیز به‌مراتب کاهش می‌دهد در جدول (۲) فهرستی از روش‌های ترکیبی مشابهت‌سنجی جملات به همراه جزئیات هرکدام ارائه شده‌است.

مجموعه دادگان مشابهت‌سنجی متون کوتاه

از جمله پیش‌نیازهای اصلی جهت بررسی کیفیت نهایی و ارزیابی کارایی الگوریتم‌های مشابهت‌سنجی، وجود یک مجموعه‌دادهٔ استاندارد برچسب‌گذاری‌شده توسط انسان است. از جمله ویژگی‌های یک پیکرهٔ استاندارد می‌توان به تنوع[7] و همگن‌بودن [8] متون موجود در آن اشاره کرد. [۱۴۱]–[۱۳۹] برای اطمینان از صحت برچسب‌های نهایی، مناسب است دادهٔ موردنظر توسط افراد متعدد، برچسب‌گذاری شده و میانگین امتیاز افراد، به‌عنوان امتیاز تهایی مشابهت در نظر گرفته‌شود. همچنین، جهت اطمینان از توافق بین افراد برچسب‌گذار[9] باید شیوه‌نامهٔ مشخصی تعریف و به شاخص‌هایی مثل شاخص کاپا[10] نیز توجه شود. [۱۴۴]–[۱۴۲] یکی از متداول‌ترین شیوه‌های برچسب‌گذاری پیکره‌های مشابهت‌سنجی، الگوی به‌کارگرفته‌شده در مجموعه‌داده STS-B است. بدین ترتیب که به‌ازای هر جفت متن، امتیازی از صفر تا چهار داده می‌شود. این امتیاز معرف میزان تشابه دو متن به‌یکدیگر خواهدبود؛ به‌نحوی‌که

[1] General Purpose Sentence Representation
[2] Universal Sentence Encoder
[3] Deep Averaging Network
[4] https://books.google.com/talktobooks/
[5] Siamese Network
[6] Fine-Tuning
[7] Saturation
[8] Homogeneity
[9] Inter-Coder Reliability & Agreement
[10] Kappa

[DOI: 10.61186/jsdp.20.3.103]

امتیاز صفر نشان‌دهندهٔ عدم مشابهت و امتیاز چهار، نشان‌دهندهٔ تشابه بیشینه دو متن به‌یکدیگر است.[۱۴۵]

(جدول- ۱): بررسی مقالات مشابهت‌سنجی ترکیبی متون کوتاه

(Table- 2): Hybrid short text similarity measurement methods

نام تیم/ نویسنده	سال	مشابهت لفظی	مشابهت دانش‌محور	مشابهت پیکره‌محور	تجزیهٔ نحوی	ابزارهای جانبی	نحوهٔ محاسبه امتیاز نهایی	دادهٔ ارزیابی	امتیاز Pearson
UKP [122]	2012	LCS, Ngram	Resnik	ESA	-	-	Regression	SemEval 2012-en	0.823
Takelab [123]	2012	Ngram	WordNet وزن‌دهی شده	LSA	-	NER , Number.	Support Vector Regression	SemEval 2012-en	0.813
Pilehvar [124]	2013	LCS در سطح کاراکتر و کلمه	WordNet با WSD و وزن‌دهی با Pagerank	-	-	-	Gaussian Processes Regression	SemEval 2012-en	0.866
Severyn [125]	2013	LCS, Ngram	Resnik	LSA-ESA	Syntactic Tree kernel	NER , number	Support Vector Regression	SemEval 2012-en	0.881
Tai [126]	2015	-	-	-	Tree LSTM Stanford PCFG	-	LSTM	SemEval 2014-SICK-en	0.86
He [127]	2015	-	-	GloVE	-	POS	CNN	SemEval2014-SICK-en	0.869
DLS-CU2 [128]	2015	کلمات مشترک	PPDB	Word2Vec	-	Lemma, NER	Regression	SemEval 2015-en	0.801
ECNU-1 [129]	2017	Ngram, BOW	-	Paragram Phrase	CoreNLP parser	Lemma, POS	traditional and deep ensemble	SemEval-2017-en	0.851
ECNU-2	2017	"	-	"	-	-	"	SemEval-2017-ar	0.74
Wali [130]	2017	Jaccard	لغت‌نامهٔ ساخت‌یافته المدار	-	Stanford, Madamira	-	SVM	۶۹۰ جمله لغت‌نامه-ar	0.92
UESTS [131]	2019	Ngram, Leven.	Babel-Synset.	Word2Vec	-	-	ensemble system	STS-B-Test-en	0.776
DAM [132]	2019	word ngram overlap	-	Glove	-	-	Decompos. Attention Model NN	STS-B-Test-en	0.773
SimiT [133]	2020	-	[134] ConceptNet برای مستحکم‌سازی[1] Word2Vec با WordNet		SpaCy	NLTK Lemma	weighted average	STS-B-Test-en	0.785
Luo et al [135]	2020	-	-	Word2Vec	گرامر وابستگی Stanford	-	attention with weight dependency	STS2015-en	0.855
Farouk [136]	2020	-	-	Word2Vec	C&C, word order	-	weighted average	RG65-en	0.8813
Almiman [7]	2020	Dirichl., TfIdf, Jaccard	-	Fasttext, WTMF	Madamira	-	Feature-Fusion DNN	SemEval2017 CQA-ar	0.681 (MRR)
Alam [137]	2020	Jaccard	BioPortal ontology API	-	-	-	Jaccard On Words Metaset	BIOSSES-en	0.61

[1] Retrofitting

[DOI: 10.61186/jsdp.20.3.103]

0.62	News, Image Caption, Student Answer-en	SVM, NB, KNN	Three level alignment Lancaster stemmer	-	Word2Vec, Glove	WordNet	Jaccard, Cosine, Common Word Count	2021	Majumder [138]
امتیاز Pearson	دادهٔ ارزیابی	نحوهٔ محاسبه امتیاز نهایی	ابزارهای جانبی	تجزیهٔ نحوی	مشابهت پیکره‌محور	مشابهت دانش‌محور	مشابهت لفظی	سال	نام تیم/ نویسنده
0.823	SemEval 2012-en	Regression	-	-	ESA	Resnik	LCS, Ngram	2012	UKP [122]
0.813	SemEval 2012-en	Support Vector Regression	NER , Number.	-	LSA	WordNet وزن‌دهی شده	Ngram	2012	Takelab [123]
0.866	SemEval 2012-en	Gaussian Processes Regression	-	-	-	WordNet با WSD و وزن‌دهی با Pagerank	LCS در سطح کاراکتر و کلمه	2013	Pilehvar [124]
0.881	SemEval 2012-en	Support Vector Regression	NER , number	Syntactic Tree kernel	LSA-ESA	Resnik	LCS, Ngram	2013	Severyn [125]
0.86	SemEval 2014-SICK-en	LSTM	-	Tree LSTM Stanford PCFG	-	-	-	2015	Tai [126]
0.869	SemEval2014-SICK-en	CNN	POS	-	GloVE	-	-	2015	He [127]
0.801	SemEval 2015-en	Regression	Lemma, NER	-	Word2Vec	PPDB	کلمات مشترک	2015	DLS-CU2 [128]
0.851	SemEval-2017-en	traditional and deep ensemble	Lemma, POS	CoreNLP parser	Paragram Phrase	-	Ngram, BOW	2017	ECNU-1 [129]
0.74	SemEval-2017-ar	"	-	-	"	-	"	2017	ECNU-2
0.92	۶۹۰ جمله لغت‌نامه-ar	SVM	-	Stanford, Madamira	-	لغت‌نامهٔ ساخت‌یافته المدار	Jaccard	2017	Wali [130]
0.776	STS-B-Test-en	ensemble system	-	-	Word2Vec	Babel-Synset.	Ngram, Leven.	2019	UESTS [131]
0.773	STS-B-Test-en	Decompos. Attention Model NN	-	-	Glove	-	word ngram overlap	2019	DAM [132]
0.785	STS-B-Test-en	weighted average	NLTK Lemma	SpaCy	[134] ConceptNet برای مستحکم‌سازی[1] Word2Vec با WordNet		-	2020	SimiT [133]
0.855	STS2015-en	attention with weight dependency	-	گرامر وابستگی Stanford	Word2Vec	-	-	2020	Luo et al [135]
0.8813	RG65-en	weighted average	-	C&C, word order	Word2Vec	-	-	2020	Farouk [136]
0.681 (MRR)	SemEval2017 CQA-ar	Feature-Fusion DNN	-	Madamira	Fasttext, WTMF	-	Dirichl., TfIdf, Jaccard	2020	Almiman [7]
0.61	BIOSSES-en	Jaccard On Words Metaset	-	-	-	BioPortal ontology API	Jaccard	2020	Alam [137]
0.62	News, Image Caption, Student Answer-en	SVM, NB, KNN	Three level alignment Lancaster stemmer	-	Word2Vec, Glove	WordNet	Jaccard, Cosine, Common Word Count	2021	Majumder [138]

[1] Retrofitting

[DOI: 10.61186/jsdp.20.3.103]

در مجموعه مقالات بررسی‌شده بیش از ۷۸ مجموعه‌داده با ویژگی‌هایی از جمله ابعاد، زبان و کاربردهای مختلف استفاده شده‌بودند که بسیاری از آنها به‌صورت آزاد منتشر نشده‌اند. در (جدول) به مهم‌ترین مجموعه دادگان موجود در زمینه مشابهت‌سنجی متون کوتاه اشاره شده‌است. موارد مشخص‌شده با نماد *، جزء مجموعه ابزار SentEval قرار داده‌شده‌اند؛ جملات این مجموعه‌داده به‌صورت گزینشی انتخاب شده و در قالب ابزاری استاندارد برای ارزیابی بردار تعبیه جملات در سال ۲۰۱۸ توسط شرکت فیس‌بوک ارائه شده‌است [۱۴۶].

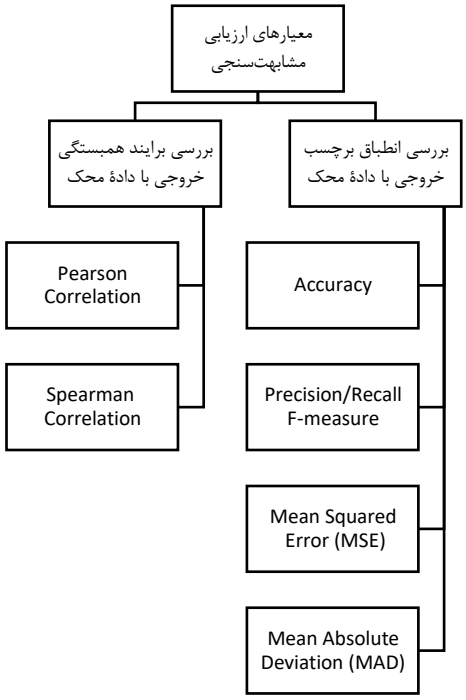
۵-روش‌های ارزیابی الگوریتم‌های مشابهت‌سنجی

روش‌های مختلفی برای ارزیابی کیفیت خروجی الگوریتم‌های مختلف در مسألهٔ مشابهت‌سنجی به‌کارگرفته‌شده‌اند که نمودار این روش‌ها در شکل (۵) ارائه شده‌است. بعضی از این روش‌ها به بررسی انطباق دقیق امتیاز پیش‌بینی‌شده با برچسب موجود در دادهٔ ارزیابی از طریق معیار دقت[1] یا محاسبهٔ نرخ خطای میانگین مربعات[2] می‌پردازند که این روش‌ها بیشتر برای مسائلی شبیه به مسألهٔ شناسایی متون بازنویسی‌شده[3] که تنها خروجی صفر یا یک موردنظر است، مناسب هستند.

از سوی دیگر، در برخی از روش‌های ارزیابی، به‌جای بررسی انطباق امتیاز هریک از مقادیر با خروجی، از طریق محاسبهٔ نرخ هم‌بستگی امتیازهای خروجی با دادهٔ شاهد به‌صورت کلی‌نگرانه به ارزیابی کارایی الگوریتم موردنظر پرداخته‌می‌شود. از جمله روش‌های متداول از این نوع، می‌توان به معیار نرخ همبستگی پیرسون یا اسپیرمن[4] اشاره کرد.

یکی از تفاوت‌های اصلی این دو معیار این است که معیار اسپیرمن نسبت به استثناءهای بسیار متفاوت[5]، انعطاف بیشتری نسبت به معیار پیرسون دارد؛ اما در مقابل نتایج نامطلوب، تأثیر منفی بیشتری در امتیاز نهایی در معیار پیرسون خواهنداشت. همچنین، معیار اسپیرمن قادر است موارد همبستگی غیرخطی را نیز شناسایی کند.[۱۴۷].

برای مسألهٔ مشابه‌یابی نیز که بر پایه مشابهت‌سنجی‌های متعدد بین متن ورودی و تمامی مجموعه متون است، روش‌های ارزیابی خاصی به‌کارگرفته‌می‌شود. در این روش‌ها کیفیت تعداد معینی از نتایج پیشنهادشده به‌ازای هر متن ورودی درنظرگرفته‌می‌شود. در برخی از معیارهای ارزیابی، ترتیب نتایج ارائه‌شده نیز از اهمیت برخوردار است؛ به‌طوری‌که پیشنهاد مقدم‌تر نتایج مطلوب، حاکی از کارایی مناسب‌تر الگوریتم موردنظر خواهدبود.



(شکل- ۵): انواع روش‌های ارزیابی مشابهت‌سنجی متون
(Figure- 5): Text similarity measurement evaluation methods

۶- تحلیل کارایی الگوها

مطابق با جدول (۳)، در حوزهٔ مشابهت‌سنجی متون فارسی، مجموعه دادهٔ آزاد و استانداردی با حجم و کیفیت مناسب یافت نشد. در یکی از مطالعات، با استفاده از متن زیرنویس فارسی فیلم‌های سینمایی، مجموعه‌داده‌ای حاوی سی‌وپنج‌هزار زوج جمله، تنها با برچسب‌های ۰ و ۱ به معنای متن معادل و غیرمعادل، به‌صورت خودکار ایجاد شده که به‌صورت عمومی در دسترس نیست.[۷۶]. در مطالعه‌ای دیگر برای شناسایی شباهت موضوعی اشعار با استفاده از الگوریتم Doc2Vec، مجموعه‌دادهٔ محدود و خاصی شامل صد مورد از اشعار هم‌موضوع فارسی ایجاد شده‌است [۱۰۹]. در صورت تولید مجموعه‌داده استاندارد فارسی در زمینهٔ مشابهت‌سنجی متون، با وجود پایگاه دانشی مانند شبکهٔ واژگان فارس‌نت [۳۴] و همچنین، الگوهای مبتنی‌بر ترنسفورمر از قبیل ParsBERT [۹۱]، زمینه برای بررسی کارایی روش‌های کلاسیک و مدرن در مسألهٔ مشابهت‌سنجی متون فارسی نیز فراهم خواهدشد.

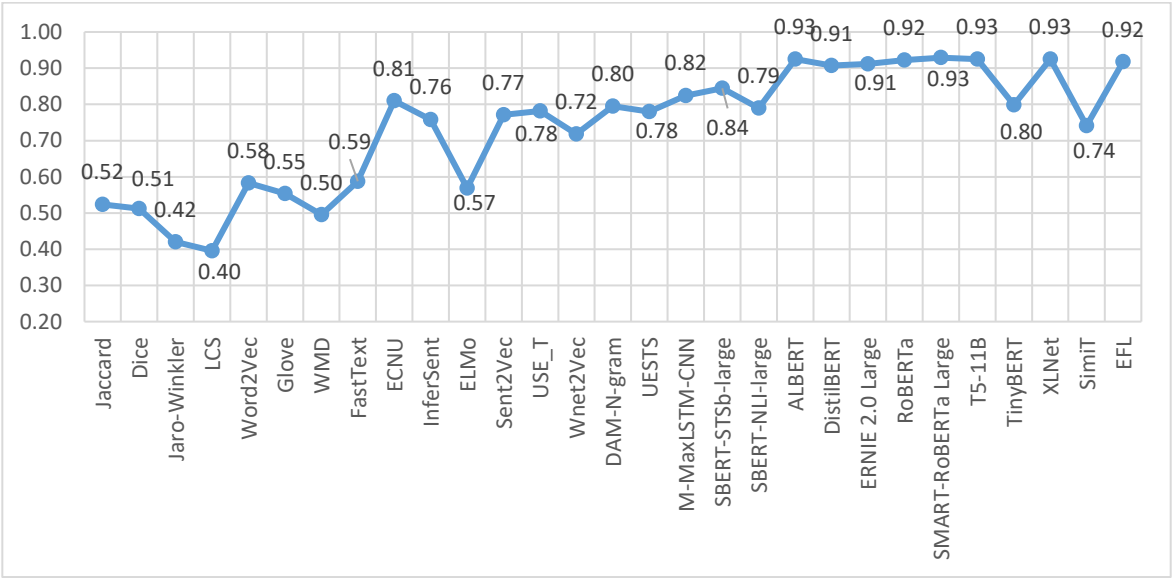
[1] Accuracy
[2] Mean Squared Error
[3] Paraphrase Identification
[4] Spearman
[5] Outlier case

[DOI: 10.61186/jsdp.20.3.103]

(جدول -۲) مجموعه دادگان اصلی مشابهت‌سنجی جملات

Table 3. Main sentence similarity datasets

نام مجموعه داده	نوع	زبان	تعداد	محدودهٔ امتیاز	تعداد برچسب‌گذار	سال انتشار
[۱۴۸] RG65 (Rubenstein and Goodenough)	کلمه-جمله	انگلیسی	۶۵	۰-۴	۵۱	۱۹۶۵
* [۱۴۹] MSRP (Microsoft Paraphrase Corpus)	جمله	انگلیسی	۵۸۰۰	۰-۱	۳	۲۰۰۵
* [۱۵۰] SICK (Sentences Involving Compositional Knowledge)	جمله	انگلیسی	۱۰۰۰۰	۱-۵	-	۲۰۱۴
[۱۳۰] Dictionary Definition Samples	جمله	عربی	۱۳۸۰	۰-۴	-	۲۰۱۶
* [۱۴۵] STS-B (SemEval 2012-2017)	جمله	انگلیسی	۸۶۲۸	۰-۴	-	۲۰۱۷
[۱۴۵] SemEval-2017- Arabic	جمله	عربی	۲۴۱۲	۰-۴	۵	۲۰۱۷



(شکل- ۶): مقایسهٔ کارایی الگوهای شاخص بر روی مجموعه داده استاندارد STS-B بر اساس معیار همبستگی Pearson

(Figure-6): Pearson correlation of the different models on STS-B dataset

به‌دلیل نبود مجموعه‌دادهٔ استاندارد فارسی، کارایی ۲۹ الگوی شاخص مشابهت‌سنجی متن بر روی مجموعه داده استاندارد انگلیسی STS-B، بررسی شد. میزان کارایی الگوهای مختلف، از مقالات موردمطالعه و همچنین، بخش مشابهت‌یابی متون از پایگاه اینترنتی PaperWithCode، استخراج شده و بر اساس شاخص ضریب همبستگی پیرسون به ترتیب زمانی در نمودار شکل(۶) ارائه شده‌است.[1]

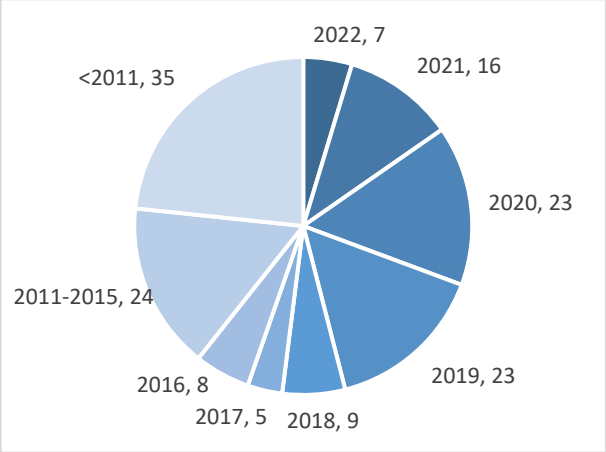
طبق بررسی‌های انجام‌شده، روش‌های مبتنی‌بر مشابهت لفظی، کمترین کارایی را در بین این الگوها داشته‌اند. پس‌از این روش‌ها، الگوریتم‌های شناسایی شباهت معنایی با استفاده از شبکه‌های معنایی جهت تولید بردار تعبیه براساس پیکره‌های متنی و همچنین، روش‌های ترکیبی، توانسته‌اند بهبود معناداری در کیفیت مشابهت‌سنجی خودکار، از خود نشان دهند. در نهایت، روش‌های شناسایی شباهت معنایی زمینه‌محور با استفاده از شبکه‌های عصبی عمیق با معماری ترنسفورمرها توانسته‌اند بهترین کارایی را در مسأله مشابهت‌سنجی متون کوتاه به زبان انگلیسی به ثبت برسانند.

[1] https://paperswithcode.com/sota/semantic-textual-similarity-on-sts-benchmark

[DOI: 10.61186/jsdp.20.3.103]

۷- جمع‌بندی

مشابهت‌سنجی بین متون کوتاه یکی از نیازهای بنیادین در بسیاری از مسائل پردازش زبان طبیعی است؛ که باتوجه‌به اهمیت آن، حجم مقالات جدید در این زمینه حاکی‌از این نکته است که پژوهشگران کماکان به‌دنبال بهبود کیفیت الگوریتم‌های موجود در این زمینه با استفاده از جدیدترین پیشرفت‌های فناوری‌های مرتبط با پردازش زبان طبیعی هستند. در این مطالعه صدوپنجاه مقالهٔ اصلی، شناسایی و بررسی شدند. نمودار توزیع مقالات بررسی‌شده براساس سال انتشار در (شکل-) ارائه شده‌است.



(شکل- ۷): نمودار توزیع مقالات بررسی‌شده بر اساس سال انتشار

(Figure- 7): Distribution of articles over years

روش‌های متعدد موجود در دسته‌بندی جامعی در (شکل- ۱) ارائه شده‌اند. به‌صورت کلی، این روش‌ها را می‌توان در سه گروه دسته‌بندی کرد؛ گروه اول روش‌هایی هستند که تنها بر روی مشابهت لفظی و تحلیل سطحی دو متن تمرکز می‌کنند. در این روش‌ها متن به‌عنوان رشته‌ای از نویسه‌ها یا مجموعه‌ای از کلمات و یا در حالت‌های پیشرفته‌تر به‌عنوان ترکیبی از این دو درنظرگرفته‌می‌شود. به‌تازگی، در برخی از گونه‌های جدید این روش‌ها از روش‌های نوین یادگیری ماشین نیز بهره‌برداری شده‌است. از این روش‌ها بیشتر برای یافتن عناوین هم‌سان موجودیت‌های اسمی، یا متون تقریباً تکراری[1] استفاده می‌شود.

گروه دوم روش‌هایی هستند که به ارتباط معنایی کلمات متن نیز توجه دارند. ارتباط معنایی بین کلمات به دو شیوه استفاده می‌شوند؛ با استفاده از پایگاه دانش ازپیش‌آماده‌شده در قالب شبکهٔ کلمات، یا مبتنی‌بر تحلیل پیکره‌های متنی و شناسایی کلمات مرتبط بر اساس نظریهٔ توزیع. باتوجه‌به هزینه‌بربودن تولید و به‌هنگام‌سازی شبکهٔ کلمات، به شیوهٔ دوم در عمل بیشتر اقبال شده‌است. در مطالعات اخیر از روش‌های یادگیری عمیق و همچنین، روش‌های مبتنی‌بر ترنسفورمرها و شبکه‌های سیامی نیز برای شناسایی کلمات مرتبط و در نهایت، تولید بردار تعبیه به‌ازای متون کوتاه بهره‌برداری شده‌است و نتایج حاکی‌از بهبود چشم‌گیر کیفیت مشابهت‌یابی بر پایهٔ این روش‌هاست.

گروه سوم، مطالعاتی هستند که برای دست‌یابی به بالاترین کارایی به ترکیب روش‌های لفظی و معنایی و حتی گاهی ترکیب آن‌ها با روش‌های تحلیل ساختاری و نحوی متن پرداخته‌اند. نیاز به یادآوری است که تحلیل‌گرهای نحوی باکیفیتی برای تمامی زبان‌ها وجود ندارد و از سوی دیگر به‌کارگیری تحلیل‌های نحوی سرعت مشابهت‌سنجی را نیز به‌مراتب کاهش می‌دهند. به‌طورکلی طبق بررسی‌های انجام‌شده، روش‌های نوین مشابهت‌سنجی معنایی بر پایهٔ ترنسفورمرها توانسته‌اند بهترین نتایج را در حوزهٔ مشابهت‌سنجی متون کوتاه کسب کنند.

[1] Near duplicate

۸- منابع 8-Refrence

[1] W. Liu et al., “Semantic Matching from Different Perspectives,” 2022, [Online]. Available: https://arxiv.org/abs/2202.06517.

[2] D. B. Bisandu, R. Prasad, and M. M. Liman, “Data clustering using efficient similarity measures,” J. Stat. Manag. Syst., vol. 22, no. 5, pp. 901–922, 2019, doi: 10.1080/09720510.2019.1565443.

[3] E. Zafarani-Moattar, M. R. Kangavari, and A. M. Rahmani, “A Comparative Study on Transfer Learning and Distance Metrics in Semantic Clustering over the COVID-19 Tweets,” pp. 1–22, 2021, [Online]. Available: http://arxiv.org/abs/2111.08658.

[4] H. A. Mohamed Hassan, G. Sansonetti, F. Gasparetti, A. Micarelli, and J. Beel, “BERT, ELMo, use and infersent sentence encoders: The Panacea for research-paper recommendation?,” CEUR Workshop Proc., vol. 2431, no. September, pp. 6–10, 2019.

[5] S. Abujar, M. Hasan, and S. A. Hossain, “Sentence similarity estimation for text summarization using deep learning,” in Advances in Intelligent Systems and Computing, 2019, vol. 828, no. January, pp. 155–164, doi: 10.1007/978-981-13-1610-4_16.

[6] A. A. Aliane and H. Aliane, “Evaluating SIAMESE Architecture Neural Models for

[DOI: 10.61186/jsdp.20.3.103]

Arabic Textual Similarity and Plagiarism Detection," ISIA 2020 - Proceedings, 4th Int. Symp. Informatics its Appl., 2020, doi: 10.1109/ISIA51297.2020.9416550.

[7] A. Almiman, N. Osman, and M. Torki, "Deep neural network approach for arabic community question answering," Alexandria Eng. J., vol. 59, no. 6, pp. 4427–4434, 2020, doi: 10.1016/j.aej.2020.07.048.

[8] P. Huang et al., "Learning Deep Structured Semantic Models for Web Search using Clickthrough Data," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9626, no. 2012, pp. 115–128, 2016, [Online]. Available: http://dl.acm.org/citation.cfm?doid=2983323.2983818.

[9] C. Sung, T. I. Dhamecha, and N. Mukhi, Improving short answer grading using transformer-based pre-training, vol. 11625 LNAI. Springer International Publishing, 2019.

[10] M. Hasanain, F. Haouari, R. Suwaileh, and Z. S. Ali, "Overview of CheckThat ! 2020 Arabic : Automatic Identification and Verification of Claims in Social Media," pp. 22–25, 2020.

[11] W. H. Gomaa and A. A. Fahmy, "A Survey of Text Similarity Approaches," Int. J. Comput. Appl., vol. 68, no. 13, pp. 13–18, 2013.

[12] M. Farouk, "Measuring Sentences Similarity: A Survey," Indian J. Sci. Technol., vol. 12, no. 25, pp. 1–11, 2019, doi: 10.17485/ijst/2019/v12i25/143977.

[13] M. Alian and A. Awajan, "Semantic Similarity for English and Arabic Texts: A Review," J. Inf. Knowl. Manag., vol. 19, no. 4, 2020, doi: 10.1142/S0219649220500331.

[14] S. K. Gaddipati, "R & D Project Comparative Evaluation of Transfer Learning Models in Semantic Text Similarity Sasi Kiran Gaddipati," no. November, 2020, doi: 10.13140/RG.2.2.34085.12003.

[15] A. Abo-Elghit, A. Al-Zoghby, and T. Hamza, "Textual Similarity Measurement Approaches: A Survey (1)," Egypt. J. Lang. Eng., vol. 0, no. 0, pp. 0–0, 2020, doi: 10.21608/ejle.2020.42018.1012.

[16] D. W. Prakoso, A. Abdi, and C. Amrit, "Short text similarity measurement methods: a review," Soft Comput., vol. 25, no. 6, pp. 4699–4723, 2021, doi: 10.1007/s00500-020-05479-2.

[17] D. Chandrasekaran and V. Mago, "Evolution of Semantic Similarity-A Survey," ACM Comput. Surv., vol. 54, no. 2, pp. 1–35, 2021, doi: 10.1145/3440755.

[18] R. W. Hamming, "Error Detecting and Error Correcting Codes," J. Franklin Inst., vol. 196, no. 4, pp. 519–520, 1923.

[19] P. A. V. Hall and G. R. Dowling, "Approximate String Matching," ACM Comput. Surv., vol. 12, no. 4, pp. 381–402, 1980, doi: 10.1145/356827.356830.

[20] M. A. Jaro, "Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida," J. Am. Stat. Assoc., vol. 84, no. 406, pp. 414–420, 1989, doi: 10.1080/01621459.1989.10478785.

[21] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," J. Mol. Biol., vol. 48, no. 3, pp. 443–453, 1970, doi: https://doi.org/10.1016/0022-2836(70)90057-4.

[22] M. Bilenko and R. J. Mooney, "Adaptive duplicate detection using learnable string similarity measures," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., pp. 39–48, 2003, doi: 10.1145/956750.956759.

[23] A. McCallum, K. Bellare, and F. Pereira, "A conditional random field for discriminatively-trained finite-state string edit distance," Proc. 21st Conf. Uncertain. Artif. Intell. UAI 2005, pp. 388–395, 2005.

[24] D. Tam, N. Monath, A. Kobren, A. Traylor, R. Das, and A. McCallum, "Optimal transport-based alignment of learned character representations for string similarity," in ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 2020, pp. 5907–5917, doi: 10.18653/v1/p19-1592.

[25] P. Shrestha, "Corpus-Based methods for Short Text Similarity," Rencontre des Étudiants Cherch. en Inform. pour le Trait. Autom. des Langues, vol. 2, 2011, [Online]. Available: http://hal.archives-ouvertes.fr/hal-00609909.

[26] M. A. Al-Ramahi and S. H. Mustafa, "N-Gram-Based Techniques for Arabic Text Document Matching; Case Study: Courses Accreditation," Basic Sci. Eng., vol. 21, no. 1, pp. 85–105, 2012, [Online]. Available: http://journals.yu.edu.jo/aybse/Issues/Vol21No1_2013/07.pdf.

[27] M. O. Alhawarat, H. Abdeljaber, and A. Hilal, "Effect of Stemming on Text Similarity for Arabic Language at Sentence Level," PeerJ Comput. Sci., vol. 7, pp. 1–18, 2021, doi: 10.7717/PEERJ-CS.530.

[28] D. Gusfield, "Algorithms on stings, trees, and sequences: Computer science and computational biology," Acm Sigact News, vol. 28, no. 4, pp. 41–60, 1997.

[29] R. A. Wagner and M. J. Fischer, "The string-to-string correction problem," J. ACM, vol. 21, no. 1, pp. 168–173, 1974.

[30] A. E. Monge and C. P. Elkan, "The field matching problem: Algorithms and applications," Proc. Second Int. Conf. Knowl. Discov. Data Min., no. Slaven 1992, pp. 267–270, 1996.

[31] William W Cohen, Pradeep Ravikumar, and Stephen, "A Comparison of String Distance Metrics for Matching Names and Records," Proc. IJCAI-2003 Work., pp. 73--78, 2003.

[DOI: 10.61186/jsdp.20.3.103]

measures of text semantic similarity," in Aaai, 2006, vol. 6, no. 2006, pp. 775–780.
[47] Y. Li, H. Li, Q. Cai, and D. Han, "A novel semantic similarity measure within sentences," in Proceedings of 2012 2nd international conference on computer science and network technology, 2012, pp. 1176–1179.
[48] D. Croft, S. Coupland, J. Shell, and S. Brown, "A fast and efficient semantic short text similarity metric," in 2013 13th UK workshop on computational intelligence (UKCI), 2013, pp. 221–227.
[49] N. Adel, K. Crockett, A. Crispin, D. Chandran, and J. P. Carvalho, "FUSE (Fuzzy Similarity Measure) - A measure for determining fuzzy short text similarity using Interval Type-2 fuzzy sets," IEEE Int. Conf. Fuzzy Syst., vol. 2018–July, 2018, doi: 10.1109/FUZZ-IEEE.2018.8491641.
[50] J. R. Firth, "Personality and language in society," Sociol. Rev., vol. 42, no. 1, pp. 37–52, 1950.
[51] K. Lund and C. Burgess, "Producing high-dimensional semantic spaces from lexical co-occurrence," Behav. Res. Methods, Instruments, Comput., vol. 28, no. 2, pp. 203–208, 1996, doi: 10.3758/BF03204766.
[52] T. K. Landauer and S. T. Dumais, "A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge," Psychol. Rev., vol. 104, no. 2, pp. 211–240, 1997, [Online]. Available: http://www.indiana.edu/~pcl/rgoldsto/courses/concepts/landauer.pdf.
[53] J. O'Shea, Z. Bandar, K. Crockett, and D. McLean, "A comparative study of two short text semantic similarity measures," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 4953 LNAI, no. May 2014, pp. 172–181, 2008, doi: 10.1007/978-3-540-78582-8_18.
[54] V. Rus, N. Niraula, and R. Banjade, "Similarity measures based on Latent Dirichlet Allocation," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 7816 LNCS, no. PART 1, pp. 459–470, 2013, doi: 10.1007/978-3-642-37247-6_37.
[55] P. D. Turney, "Mining the web for synonyms: PMI-IR versus LSA on TOEFL," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 2167, pp. 491–502, 2001, doi: 10.1007/3-540-44795-4_42.
[56] R. L. Cilibrasi and P. M. B. Vitányi, "The Google similarity distance," IEEE Trans. Knowl. Data Eng., vol. 19, no. 3, pp. 370–383, 2007, doi: 10.1109/TKDE.2007.48.
[57] E. Gabrilovich, S. Markovitch, and others, "Computing semantic relatedness using
[32] J. Wang, G. Li, and J. Fe, "Fast-join: An efficient method for fuzzy token matching based string similarity join," Proc. - Int. Conf. Data Eng., pp. 458–469, 2011, doi: 10.1109/ICDE.2011.5767865.
[33] G. A. Miller, "WordNet: a lexical database for English," Commun. ACM, vol. 38, no. 11, pp. 39–41, 1995.
[34] M. Shamsfard, "Developing FarsNet: A lexical ontology for Persian," GWC 2008, p. 413, 2007.
[35] W. Black et al., "Introducing the Arabic wordnet project," in Proceedings of the third international WordNet conference, 2006, pp. 295–300.
[36] R. Rada, H. Mili, E. Bicknell, and M. Blettner, "Development and application of a metric on semantic nets," IEEE Trans. Syst. Man. Cybern., vol. 19, no. 1, pp. 17–30, 1989.
[37] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in Proceedings of the 32nd annual meeting on Association for Computational Linguistics -, 1994, pp. 133–138, doi: 10.3115/981732.981751.
[38] C. Leacock and M. Chodorow, "Combining local context and WordNet similarity for word sense identification," WordNet An Electron. Lex. database, vol. 49, no. 2, pp. 265–283, 1998.
[39] Y. Bin, L. Xiao-Ran, L. Ning, and Y. Yue-Song, "Using Information Content to Evaluate Semantic Similarity on HowNet," in 2012 Eighth International Conference on Computational Intelligence and Security, Nov. 2012, pp. 142–145, doi: 10.1109/CIS.2012.39.
[40] D. Lin and others, "An information-theoretic definition of similarity.," in Icml, 1998, vol. 98, no. 1998, pp. 296–304.
[41] S. Banerjee and T. Pedersen, "An adapted Lesk algorithm for word sense disambiguation using WordNet," in International conference on intelligent text processing and computational linguistics, 2002, pp. 136–145.
[42] J.-B. Gao, B.-W. Zhang, and X.-H. Chen, "A WordNet-based semantic similarity measurement combining edge-counting and information content theory," Eng. Appl. Artif. Intell., vol. 39, pp. 80–88, 2015.
[43] G. Zhu and C. A. Iglesias, "Computing semantic similarity of concepts in knowledge graphs," IEEE Trans. Knowl. Data Eng., vol. 29, no. 1, pp. 72–85, 2016.
[44] C. Saedi, A. Branco, J. António Rodrigues, and J. Silva, "WordNet Embeddings," pp. 122–131, 2019, doi: 10.18653/v1/w18-3016.
[45] S. Jimenez, F. A. Gonzalez, A. Gelbukh, and G. Duenas, "Word2set: WordNet-Based Word Representation Rivaling Neural Word Embedding for Lexical Similarity and Sentiment Analysis," IEEE Comput. Intell. Mag., vol. 14, no. 2, pp. 41–53, 2019, doi: 10.1109/MCI.2019.2901085.
[46] R. Mihalcea, C. Corley, C. Strapparava, and others, "Corpus-based and knowledge-based

[70] Z. Wang, H. Mi, and A. Ittycheriah, “Sentence similarity learning by lexical decomposition and composition,” COLING 2016 - 26th Int. Conf. Comput. Linguist. Proc. COLING 2016 Tech. Pap., no. challenge 2, pp. 1340–1349, 2016.

[71] E. Moatez, B. Nagoudi, D. Schwab, S. Similarity, E. Moatez, and B. Nagoudi, “Semantic Similarity of Arabic Sentences with Word Embeddings Embeddings,” 2018.

[72] N. H. Tien, N. M. Le, Y. Tomohiro, and I. Tatsuya, “Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity,” Inf. Process. Manag., vol. 56, no. 6, 2019, doi: 10.1016/j.ipm.2019.102090.

[73] A. Mahmoud and M. Zrigui, “Sentence Embedding and Convolutional Neural Network for Semantic Textual Similarity Detection in Arabic Language,” Arab. J. Sci. Eng., vol. 44, no. 11, pp. 9263–9274, 2019, doi: 10.1007/s13369-019-04039-7.

[74] S. Kim, I. Kang, and N. Kwak, “Semantic Sentence Matching with Densely-Connected Recurrent and Co-Attentive Information,” Proc. AAAI Conf. Artif. Intell., vol. 33, no. February, pp. 6586–6593, Jul. 2019, doi: 10.1609/aaai.v33i01.33016586.

[75] G. Chen, X. Shi, M. Chen, and L. Zhou, “Text similarity semantic calculation based on deep reinforcement learning,” Int. J. Secur. Networks, vol. 15, no. 1, pp. 59–66, 2020, doi: 10.1504/IJSN.2020.106526.

[76] Z. Sadat Hosseini Moghadam Emami, S. Tabatabayiseifi, M. Izadi, and M. Tavakoli, “Designing a Deep Neural Network Model for Finding Semantic Similarity between Short Persian Texts Using a Parallel Corpus,” in 2021 7th International Conference on Web Research, ICWR 2021, 2021, pp. 91–96, doi: 10.1109/ICWR51868.2021.9443108.

[77] S. V. Moravvej, M. Joodaki, M. J. Maleki Kahaki, and M. Salimi Sartakhti, “A method Based on an Attention Mechanism to Measure the Similarity of two Sentences,” in 2021 7th International Conference on Web Research, ICWR 2021, 2021, pp. 238–242, doi: 10.1109/ICWR51868.2021.9443135.

[78] A. Mahmoud and M. Zrigui, “BLSTM-API: Bi-LSTM Recurrent Neural Network-Based Approach for Arabic Paraphrase Identification,” Arab. J. Sci. Eng., vol. 46, no. 4, pp. 4163–4174, 2021, doi: 10.1007/s13369-020-05320-w.

[79] M. E. Peters et al., “Deep contextualized word representations,” NAACL HLT 2018 - 2018 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf., vol. 1, pp. 2227–2237, 2018, doi: 10.18653/v1/n18-1202.

[80] A. Radford, T. Narasimhan, T. Salimans, and I. Sutskever, “[GPT-1] Improving Language Understanding by Generative Pre-Training,” Preprint, pp. 1–12, 2018, [Online]. Available: Wikipedia-based explicit semantic analysis.,” in IJcAI, 2007, vol. 7, pp. 1606–1611.

[58] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” J. Mach. Learn. Res., vol. 3, pp. 993–1022, 2003.

[59] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” 1st Int. Conf. Learn. Represent. ICLR 2013 - Work. Track Proc., pp. 1–12, 2013.

[60] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, “A neural probabilistic language model,” J. Mach. Learn. Res., vol. 3, no. Feb, pp. 1137–1155, 2003.

[61] A. B. Soliman, K. Eissa, and S. R. El-Beltagy, “AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP,” Procedia Comput. Sci., vol. 117, pp. 256–265, 2017, doi: 10.1016/j.procs.2017.10.117.

[62] M. M. Fouad, A. Mahany, N. Aljohani, R. Ayaz, and A. S. Hassan, “ArWordVec : efficient word embedding models for Arabic tweets,” Soft Comput., 2019, doi: 10.1007/s00500-019-04153-6.

[63] J. Tissier et al., “Dict2vec : Learning Word Embeddings using Lexical Dictionaries To cite this version : HAL Id : ujm-01613953 Dict2vec : Learning Word Embeddings using Lexical Dictionaries,” 2017.

[64] A. M. Alargrami and M. M. Eljazzar, “Imam: Word Embedding Model for Islamic Arabic NLP,” 2nd Nov. Intell. Lead. Emerg. Sci. Conf. NILES 2020, pp. 520–524, 2020, doi: 10.1109/NILES50944.2020.9257931.

[65] J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global vectors for word representation,” EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf., pp. 1532–1543, 2014, doi: 10.3115/v1/d14-1162.

[66] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching Word Vectors with Subword Information,” Trans. Assoc. Comput. Linguist., vol. 5, pp. 135–146, 2017, doi: 10.1162/tacl_a_00051.

[67] M. J. Kusner, Y. Sun, N. I. Kolkin, and K. Q. Weinberger, “From word embeddings to document distances,” in 32nd International Conference on Machine Learning, ICML 2015, 2015, vol. 2, pp. 957–966.

[68] T. Kenter and M. De Rijke, “Short text similarity with word embeddings,” Int. Conf. Inf. Knowl. Manag. Proc., vol. 19-23-Oct-, pp. 1411–1420, 2015, doi: 10.1145/2806416.2806475.

[69] H. He and J. Lin, “Pairwise Word Interaction Modeling with Deep Neural Networks for Semantic Similarity Measurement,” in Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 937–948, doi: 10.18653/v1/N16-1108.

[DOI: 10.61186/jsdp.20.3.103]

ahttps://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf.
[81] A. Vaswani et al., "Attention is all you need," Adv. Neural Inf. Process. Syst., vol. 2017–Decem, no. Nips, pp. 5999–6009, 2017.
[82] I. Solaiman et al., "Release strategies and the social impacts of language models," arXiv Prepr. arXiv1908.09203, 2019.
[83] G. I. Winata, A. Madotto, Z. Lin, R. Liu, J. Yosinski, and P. Fung, "Language Models are Few-shot Multilingual Learners," in Proceedings of the 1st Workshop on Multilingual Representation Learning, 2021, pp. 1–15, doi: 10.18653/v1/2021.mrl-1.1.
[84] N. Muennighoff, "SGPT : GPT Sentence Embeddings for Semantic Search," pp. 1–17, 2022.
[85] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf., vol. 1, no. Mlm, pp. 4171–4186, 2019.
[86] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," no. 1, 2019, [Online]. Available: https://aclanthology.org/2021.ccl-1.108.
[87] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," pp. 2–6, 2019, [Online]. Available: http://arxiv.org/abs/1910.01108.
[88] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," pp. 1–17, 2019, [Online]. Available: http://arxiv.org/abs/1909.11942.
[89] N. Peinelt, D. Nguyen, and M. Liakata, "tBERT: Topic Models and BERT Joining Forces for Semantic Similarity Detection," no. section 5, pp. 7047–7055, 2020, doi: 10.18653/v1/2020.acl-main.630.
[90] H. Al-Theiabat and A. Al-Sadi, "The Inception Team at NSURL-2019 Task 8: Semantic Question Similarity in Arabic," 2020, [Online]. Available: http://arxiv.org/abs/2004.11964.
[91] M. Farahani, M. Gharachorloo, M. Farahani, and M. Manthouri, "ParsBERT: Transformer-based Model for Persian Language Understanding," Neural Process. Lett., vol. 53, no. 6, pp. 3831–3847, 2021, doi: 10.1007/s11063-021-10528-4.
[92] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," 2020, [Online]. Available: http://arxiv.org/abs/2003.00104.
[93] G. Inoue, B. Alhafni, N. Baimukan, H. Bouamor, and N. Habash, "The Interplay of Variant, Size, and Task Type in Arabic Pre-trained Language Models," 2021, [Online]. Available: http://arxiv.org/abs/2103.06678.
[94] M. Abdul-Mageed, A. R. Elmadany, and E. M. B. Nagoudi, "ARBERT & MARBERT: Deep bidirectional transformers for Arabic," ACL-IJCNLP 2021 - 59th Annu. Meet. Assoc. Comput. Linguist. 11th Int. Jt. Conf. Nat. Lang. Process. Proc. Conf., no. i, pp. 7088–7105, 2021,
[95] A. Abdelali, N. Durrani, F. Dalvi, and H. Sajjad, "Interpreting Arabic Transformer Models," 2022.
[96] A. Alsaleh, E. Atwell, and A. Altahhan, "Quranic Verses Semantic Relatedness Using AraBERT," Proc. Sixth Arab. Nat. Lang. Process. Work., vol. 3, pp. 185–190, 2021.
[97] K. Lo, "SciBERT: A Pretrained Language Model for Scientific Text," 2019.
[98] J. Lee et al., "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," Bioinformatics, vol. 36, no. 4, pp. 1234–1240, 2020.
[99] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, and I. Androutsopoulos, "LEGAL-BERT: The muppets straight out of law school," arXiv Prepr. arXiv2010.02559, 2020.
[100] F. Zhuang, F. Wei, H. Huang, L. Zhang, and Q. Zhang, "PromptBERT : Improving BERT Sentence Embeddings with Prompts," 2022.
[101] T. Gao, A. Fisch, and D. Chen, "Making Pre-trained Language Models Better Few-shot Learners," 2020.
[102] A. Neelakantan et al., "Text and Code Embeddings by Contrastive Pre-Training," 2022.
[103] H. Wang, Y. Li, Z. Huang, Y. Dou, L. Kong, and J. Shao, "SNCSE : Contrastive Learning for Unsupervised Sentence Embedding with Soft Negative Samples," 2022.
[104] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, and Q. Liu, "ERNIE: Enhanced Language Representation with Informative Entities," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 1441–1451, doi: 10.18653/v1/P19-1139.
[105] Y. Sun et al., "ERNIE 2.0: A continual pre-training framework for language understanding," AAAI 2020 - 34th AAAI Conf. Artif. Intell., pp. 8968–8975, 2020, doi: 10.1609/aaai.v34i05.6428.
[106] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized autoregressive pretraining for language understanding," Adv. Neural Inf. Process. Syst., vol. 32, no. NeurIPS, pp. 1–11, 2019.
[107] C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," J. Mach. Learn. Res., vol. 21, 2020.
[108] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," 31st Int. Conf. Mach. Learn. ICML 2014, vol. 4, pp. 2931–2939, 2014.

[DOI: 10.61186/jsdp.20.3.103]

[109] S. Akef, M. H. Bokaei, and H. Sameti, "Training Doc2Vec on a Corpus of Persian Poems to Answer Thematic Similarity Multiple-Choice Questions," in 2020 10th International Symposium on Telecommunications: Smart Communications for a Better Life, IST 2020, 2020, pp. 146–149, doi: 10.1109/IST50524.2020.9345918.

[110] M. Alshammeri, E. Atwell, and M. A. Alsalka, "Detecting Semantic-based Similarity between Verses of the Quran with Doc2vec," Procedia CIRP, vol. 189, pp. 351–358, 2021,

[111] A. M. Abdelghany, H. M. Abdelaal, A. M. Kamr, and P. M. Elkafrawy, "Doc2Vec: An approach to identify Hadith Similarities," Aust. J. Basic Appl. Sci., no. March, pp. 46–53, 2020, doi: 10.22587/ajbas.2020.14.12.5.

[112] R. Kiros et al., "Skip-thought vectors," Adv. Neural Inf. Process. Syst., vol. 2015–Janua, no. 786, pp. 3294–3302, 2015.

[113] F. Hill, K. Cho, and A. Korhonen, "Learning distributed representations of sentences from unlabelled data," 2016 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. NAACL HLT 2016 - Proc. Conf., pp. 1367–1377, 2016, doi: 10.18653/v1/n16-1162.

[114] F. Hill, K. Cho, A. Korhonen, and Y. Bengio, "Learning to Understand Phrases by Embedding the Dictionary," Trans. Assoc. Comput. Linguist., vol. 4, no. April, pp. 17–30, 2016, doi: 10.1162/tacl_a_00080.

[115] M. Pagliardini, P. Gupta, and M. Jaggi, "Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features," in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018,

[116] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised learning of universal sentence representations from natural language inference data," EMNLP 2017 - Conf. Empir. Methods Nat. Lang. Process. Proc., pp. 670–680, 2017, doi: 10.18653/v1/d17-1070.

[117] S. Subramanian, A. Trischler, Y. Bengio, and C. J. Pal, "Learning general purpose distributed sentence representations via large scale multitask learning," 6th Int. Conf. Learn. Represent. ICLR 2018 - Conf. Track Proc., no. 2016, pp. 1–16, 2018.

[118] D. Cer et al., "Universal sentence encoder for English," EMNLP 2018 - Conf. Empir. Methods Nat. Lang. Process. Syst. Demonstr. Proc., pp. 169–174, 2018, doi: 10.18653/v1/d18-2029.

[119] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," EMNLP-IJCNLP 2019 - 2019 Conf. Empir. Methods Nat. Lang. Process. 9th Int. Jt. Conf. Nat. Lang. Process. Proc. Conf.,

[120] H. Tsukagoshi, R. Sasano, and K. Takeda, "Comparison and Combination of Sentence Embeddings Derived from Different Supervision Signals," 2022.

[121] H. Tsukagoshi, "DefSent : Sentence Embeddings using Definition Sentences," pp. 411–418, 2021.

[122] D. Bär, C. Biemann, I. Gurevych, and T. Zesch, "Ukp: Computing semantic textual similarity by combining multiple content similarity measures," in * SEM 2012: The First Joint Conference on Lexical and Computational Semantics, 2012, pp. 435–440.

[123] F. Saríc, G. Glavaš, M. Karan, J. Šnajder, and B. D. Bašic, "TakeLab: Systems for measuring semantic text similarity," *SEM 2012 - 1st Jt. Conf. Lex. Comput. Semant., vol. 2, no. January, pp. 441–448, 2012.

[124] M. T. Pilehvar, D. Jurgens, and R. Navigli, "Align, disambiguate and walk: A unified approach for measuring semantic similarity," ACL 2013 - 51st Annu. Meet. Assoc. Comput. Linguist. Proc. Conf., vol. 1, pp. 1341–1351, 2013.

[125] A. Severyn, M. Nicosia, and A. Moschitti, "Learning semantic textual similarity with structural representations," ACL 2013 - 51st Annu. Meet. Assoc. Comput. Linguist. Proc. Conf., vol. 2, pp. 714–718, 2013.

[126] K. S. Tai, R. Socher, and C. D. Manning, "Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks," in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, vol. 1, pp. 1556–1566,

[127] H. He, K. Gimpel, and J. Lin, "Multi-perspective sentence similarity modeling with convolutional neural networks," Conf. Proc. - EMNLP 2015 Conf. Empir. Methods Nat. Lang. Process., no. September, pp. 1576–1586, 2015,

[128] M. A. Sultan, S. Bethard, and T. Sumner, "DLS@CU: Sentence Similarity from Word Alignment and Semantic Vector Composition," no. SemEval,

[129] J. Tian, Z. Zhou, M. Lan, and Y. Wu, "ECNU at SemEval-2017 Task 1: Leverage Kernel-based Traditional NLP features and Neural Networks to Build a Universal Model for Multilingual and Cross-lingual Semantic Textual Similarity," pp. 191–197, 2018,

[130] W. Wali, B. Gargouri, and A. Ben Hamadou, "Enhancing the sentence similarity measure by semantic and syntactico-semantic knowledge," Vietnam J. Comput. Sci., vol. 4, no. 1, pp. 51–60, 2016,

[131] B. Hassan, S. E. Abdelrahman, R. Bahgat, and I. Farag, "UESTS: An Unsupervised Ensemble Semantic Textual Similarity Method," IEEE Access, vol. 7, pp. 85462–85482, 2019,

[DOI: 10.61186/jsdp.20.3.103]

[132] I. Lopez-Gazpio, M. Maritxalar, M. Lapata, and E. Agirre, “Word n-gram attention models for sentence similarity and inference,” Expert Syst. Appl., vol. 132, pp. 1–11, 2019,
[133] E. Inan, “SimiT: A Text Similarity Method Using Lexicon and Dependency Representations,” New Gener. Comput., vol. 38, no. 3, pp. 509–530, 2020,
[134] R. Speer, J. Chin, and C. Havasi, “ConceptNet 5.5: An Open Multilingual Graph of General Knowledge,” no. Singh 2002, pp. 4444–4451, 2016, [Online]. Available: http://arxiv.org/abs/1612.03975.
[135] J. Luo et al., “Exploiting Syntactic and Semantic Information for Textual Similarity Estimation,” Math. Probl. Eng., vol. 2021, 2021, doi: 10.1155/2021/4186750.
[136] M. Farouk, “Measuring text similarity based on structure and word embedding,” Cogn. Syst. Res., vol. 63, pp. 1–10, 2020,
[137] F. Alam, M. Afzal, and K. M. Malik, “Comparative Analysis of Semantic Similarity Techniques for Medical Text,” in International Conference on Information Networking, 2020,
[138] G. Majumder, “Interpretable semantic textual similarity of sentences using alignment of chunks with classification and regression,” no. March, 2021,
[139] M. W. Bauer and B. Aarts, “Corpus construction: A principle for qualitative data collection,” Qual. Res. with text, image sound A Pract. Handb., pp. 19–37, 2000.
[140] A. O’Keeffe and M. McCarthy, The Routledge handbook of corpus linguistics, vol. 10. Routledge London, 2010.
[141] S. Atkins, J. Clear, and N. Ostler, “Corpus design criteria,” Lit. Linguist. Comput., vol. 7, no. 1, pp. 1–16, 1992.
[142] R. Artstein and M. Poesio, “Survey Article Inter-Coder Agreement for Computational Linguistics,” no. August 2005, 2008.
[143] J. Pustejovsky and A. Stubbs, Natural Language Annotation for Machine Learning: A guide to corpus-building for applications. “ O’Reilly Media, Inc.,” 2012.
[144] M. Lombard, J. Snyder-duch, and C. C. Bracken, “Practical Resources for Assessing and Reporting Intercoder Reliability in Content Analysis Research Projects,” no. January, 2005.
[145] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia, “SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation,” pp. 1–14, 2018, doi: 10.18653/v1/s17-2001.
[146] A. Conneau and D. Kiela, “SentEval: An evaluation toolkit for universal sentence representations,” Lr. 2018 - 11th Int. Conf. Lang. Resour. Eval., pp. 1699–1704, 2019.
[147] A. Bistacchi, S. Mittempergher, M. Martinelli, and F. Storti, “On a new robust workflow for the statistical and spatial analysis of fracture data collected with scanlines” Solid Earth, vol. 11, no. 6, pp. 2535–2547, 2020.
[148] H. Rubenstein and J. B. Goodenough, “Contextual correlates of synonymy,” Commun. ACM, vol. 8, no. 10, pp. 627–633, 1965.
[149] W. B. Dolan and C. Brockett, “Automatically constructing a corpus of sentential paraphrases,” 2005.
[150] M. Marelli et al., “A SICK cure for the evaluation of compositional distributional semantic models.,” in Lrec, 2014, pp. 216–223.
[151] F. Mashhadirajab, M. Shamsfard, R. Adelkhah, F. Shafiee, and C. Saedi, “A text alignment corpus for Persian plagiarism detection,” in CEUR Workshop Proceedings, 2016, vol. 1737, pp. 184–189.

احمد ربیعی زاده، در سال ۱۳۸۸ مدرک کارشناسی خود را در رشته مهندسی فناوری اطلاعات از دانشگاه آزاد اسلامی واحد تهران جنوب دریافت کرده و بیش از ده سال در آزمایشگاه هوش مصنوعی مرکز تحقیقات کامپیوتری علوم اسلامی نور فعالیت داشته‌است. وی تحصیلات خود را در مقطع کارشناسی ارشد رشته مهندسی فناوری اطلاعات در دانشگاه قم با موضوع پایان‌نامه مشابهت‌سنجی احادیث به پایان رسانده‌است. زمینه‌های پژوهشی موردعلاقهٔ ایشان عبارتند از پردازش زبان طبیعی، متن‌کاوی و علوم انسانی دیجیتال.

نشانی رایانامه ایشان عبارت است از:

rabieizadeh@noornet.net

حسین امیرخانی، مدرک کارشناسی خود را در رشته مهندسی کامپیوتر با گرایش نرم‌افزار در سال ۱۳۸۶ از دانشگاه اصفهان دریافت کرده‌است. ایشان در سال ۱۳۸۸ مدرک کارشناسی ارشد خود را در رشته مهندسی کامپیوتر با گرایش هوش مصنوعی از دانشگاه صنعتی امیرکبیر و در سال ۱۳۹۳ نیز مدرک دکترای خود را از همان دانشگاه و در همان رشته کسب کرده‌است. وی هم‌اکنون استادیار گروه مهندسی کامپیوتر و فناوری اطلاعات دانشگاه قم بوده و زمینه‌های پژوهشی موردعلاقهٔ ایشان یادگیری ماشین و پردازش زبان طبیعی است.

نشانی رایانامه ایشان عبارت است از:

amirkhani@qom.ac.ir

[DOI: 10.61186/jsdp.20.3.103]