دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی ارشد هوش مصنوعی

تشخیص شباهت میان جملات و پاراگراف‌ها به کمک مدل جایگذاری کلمات

نگارنده: مرتضی اله‌پور فدافن

استاد راهنما

دکتر مرتضی زاهدی

استاد مشاور

دکتر هدی مشایخی

تیر ۱۳۹۸

تقدیم به پدر و مادرم

سپاس‌گزاری

از استاد بزرگوارم، آقای دکتر مرتضی زاهدی برای راهنمایی‌های بی‌دریغشان کمال تشکر و قدردانی را دارم.

مرتضی اله‌پور فدافن
تیر ۱۳۹۸

تعهد نامه

اینجانب **مرتضی اله‌پور فدافن** دانشجوی کارشناسی ارشد رشته **مهندسی کامپیوتر مهندسی کامپیوتر** دانشگاه شاهرود، نویسنده پایان‌نامه با عنوان **تشخیص شباهت میان جملات و پاراگراف‌ها به کمک مدل جایگذاری کلمات** ، تحت راهنمایی **مرتضی زاهدی** متعهد می‌شوم:

- تحقیقات در این پایان‌نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش‌های دیگر پژوهش‌گران، به مرجع مورد استفاده استناد شده است.
- مطالب این پایان‌نامه، تا کنون توسط خود، یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ‌جا ارایه نشده است.
- حقوق معنوی این اثر، به دانشگاه صنعتی شاهرود تعلق دارد، و مقالات مستخرج با نام ‘‘ دانشگاه صنعتی شاهرود’’ یا “ Shahrood University of Technology ” به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به‌دست آوردن نتایج اصلی پایان‌نامه تاثیرگذار بوده‌اند، در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در تمام مراحل انجام این پایان‌نامه، در مواردی که از موجود زنده (یا بافت‌های آنها) استفاده شده است، ضوابط و اصول اخلاقی رعایت شده است.
- در تمام مراحل انجام این پایان‌نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته (یا استفاده شده است)، اصل رازداری و اصول اخلاق انسانی رعایت شده است.

مرتضی اله‌پور فدافن
تیر ۱۳۹۸

مالکیت نتایج و حق نشر

- تمام حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده) متعلق به دانشگاه صنعتی شاهرود می‌باشد. این مطلب باید به نحو مقتضی، در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در این پایان‌نامه بدون ذکر منبع مجاز نمی‌باشد.

چکیده

امروزه با توجه به افزایش حجم اطلاعات و مستندات در زمینه های مختلف، دسترسی سریع به اطلاعات مورد نظر برای هر فرد از اهمیت ویژه ای برخوردار است. از این رو علاوه بر تکنیک‌های بازیابی اطلاعات، تکنیک‌های خلاصه سازی و دسته بندی نیز میتواند کمک موثری در افزایش سرعت دسترسی کاربران به اسناد مورد نظرشان باشد . ساخت سیستمی که بتواند به صورت موثری شباهت بین دو عبارت را تشخیص دهد موضوع بسیاری از پژوهش ها بوده است. تعیین شباهت بین دو عبارت می‌تواند از طریق محاسبه شباهت میان واژه‌ها و یا توسط روش‌های یادگیری ماشین انجام شود.

در این تحقیق روشی ارائه می‌شود که با توجه معنای کلمات در هر جمله، شباهت بین جمله‌ها را مشخص می‌کند. برای بدست آوردن معنای هر کلمه از مدل‌های جایگذاری کلمات استفاده می‌کنیم. یکی از ویژگی‌های این مدل‌ها این است که هر کلمه را در یک فضای چند بعدی نشان می‌دهند در نتیجه با عملیات مختلف بردارها مانند جمع دوبردار می‌توان معنای حاصل از هم‌جواری دو کلمه را بدست آورد. در ادامه به کمک چهار تابع استخراج ویژگی، ویژگی‌های مخلفی از جمله‌ها استخراج می‌شود سپس از این ویژگی‌ها در یک دسته‌بند استفاده می‌شود تا شباهت یا عدم‌شباهت دو جمله تشخیص داده شود.

در ساخت چنین سیستمی یکی از مولفه‌های مهم و اساسی توانایی تشخیص شباهت بین جملات و پارگراف‌های متون است که موضوع تحقیقات بسیار زیادی بوده است. این روش می‌تواند شباهت معنایی بین دو جمله را علی‌رغم عدم شباهت لغوی آن‌ها تشخیص دهد. این روش علاوه بر تشخیص شباهت، در تشخیص عدم شباهت دو عبارت نیز کارآمد است به طوری که پس انجام آزمایشات این روش با دقت ۸۳درصد داده‌های مورد آزمایش را دسته‌بندی کرد که در مقایسه با روش‌های معرفی شده، عملکرد بهتری دارد.

کلمات کلیدی: جایگذاری کلمات، شباهت متون، پردازش متن، متن‌کاوی

فهرست مطالب

فهرست تصاویر **س**

فهرست جداول **ف**

۱ مقدمه **۱**
۱.۱ انگیزش و هدف پژوهش ۱
۲.۱ تعریف مسئله ۲
۳.۱ کاربردهای محاسبه شباهت بین اسناد ۳
۴.۱ ساختار پایان‌نامه ۴

۲ معیارهای محاسبه شباهت متون **۵**
۱.۲ حوزه‌های بررسی شباهت ۵
۲.۲ معیارهای مبتنی بر رشته ۶
۳.۲ معیارهای مبتنی بر پیکره ۱۲
۱.۳.۲ روش‌های لغوی محاسبه شباهت بین دو رشته ۱۳
۴.۲ روش‌های مبتنی بر اثر انگشت ۱۶
۵.۲ کارهای پیشین ۱۷
۱.۵.۲ SVSM ۱۹
۲.۵.۲ شبکه‌های عصبی کانولوشنی ۲۱
۶.۲ نتیجه‌گیری ۲۸

۳ روش پیشنهادی **۳۱**
۱.۳ معرفی پایگاه داده ۳۱
۲.۳ بردارهای جایگذاری کلمات ۳۴
۱.۲.۳ Word۲Vec ۳۴
۲.۲.۳ GloVe ۳۵
۳.۳ پیش پردازش ۴۴

۴.۳ سیستم پیشنهادی ۴۴
۱.۴.۳ توابع استخراج ویژگی ۴۵
۲.۴.۳ موتور استنتاج ۵۰
۵.۳ نتیجه گیری ۵۲

۴ جزئیات پیکربندی و پیاده‌سازی **۵۳**
۱.۴ پیش پردازش ۵۳
۲.۴ پارامترها ۵۴
۳.۴ معیارها ۵۵
۱.۳.۴ معیار دقت ۵۶
۲.۳.۴ معیار precision ۵۶
۳.۳.۴ معیار Recall ۵۷
۴.۳.۴ معیار F_1 . . . . . ۵۷
۴.۴ نتایج آزمایش ۵۷
۱.۴.۴ اهمیت ویژگی‌ها ۵۸
۲.۴.۴ بررسی خطا ۵۹
۵.۴ نتیجه گیری ۶۲

۵ نتیجه گیری و پیشنهادات **۶۳**
۱.۵ یافته‌های تحقیق ۶۳
۲.۵ پیشنهادات و کارهای آینده ۶۴

مراجع **۶۵**

فهرست تصاویر

۱.۲ معماری کلی شبکه عصبی استفاده شده در مدل کانولوشنی ۲۲
۲.۲ ساختار مدل اول که برای تشخیص شباهت بین جملات استفاده شده‌است. ۲۵
۳.۲ معماری مدل ARC-II. در این تصویر کانولوشن‌های یک بعدی و دو بعدی و ساختار هر کدام مشخص است. ۲۶
۴.۲ ساختار pooling دو بعدی که ترتیب را نیز حفظ می‌کند ۲۷

۱.۳ شبکه عصبی مدل CBOW ۳۶
۲.۳ مدل شبکه عصبی استفاده در روش Skip-Gram ۳۷
۳.۳ نمونه‌ای از فضای برداری مدل GloVe ۴۱
۴.۳ نمونه‌ای از فضای برداری مدل GloVe ۴۲
۵.۳ نمونه‌ای از فضای برداری مدل GloVe ۴۳
۶.۳ نمونه ای از نمایش بردارهای جایگذاری کلمات در فضای دو بعدی . . . ۴۸
۷.۳ دوبردار مختلف با شباهت کسینوسی برابر ۰/۹ . . . . . ۵۱

۱.۴ نتایج آزمایشات مربوط به تعداد پارامترها ۵۴
۲.۴ نتایج اجرا که براساس طول جملات تفکیک شده‌است ۶۰
۳.۴ نتایج اجرا برای بررسی تاثیر هم‌پوشانی لغوی بر دقت روش. ۶۱

فهرست جداول

۱.۲ مثالی از شباهت جاکارد ۱۵
۲.۲ نتایج آزمایش‌ها بر روی پایگاده داده استاندارد MSRP ۲۱
۳.۲ نتایج آزمایش‌ها بر روی پایگاده داده استاندارد MSRP ۲۹

۱.۳ نمونه داده‌های موجود در پایگاه داده ۳۳
۲.۳ مثالی از ماتریس هم‌رخدادی در روش GloVe ۳۸
۳.۳ نحوه محاسبه نسبت هم رخدادی در روش GloVe ۳۹

۱.۴ مقایسه نتایج حاصل از پیاده سازی با روش‌های پایه ۵۸
۲.۴ بررسی تاثیر هر مجموعه از ویژگی‌ها ۵۹

فصل ۱

مقدمه

۱.۱ انگیزش و هدف پژوهش

امروزه با توجه به افزایش حجم اطلاعات و مستندات در زمینه‌های مختلف، دسترسی سریع به اطلاعات مورد نظر برای هر فرد از اهمیت ویژه‌ای برخوردار است. از این رو علاوه بر تکنیک‌های بازیابی اطلاعات، تکنیک‌های خلاصه‌سازی و دسته‌بندی نیز می‌تواند کمک موثری در افزایش سرعت دسترسی کاربران به اسناد مورد نظرشان داشته‌باشد [۱].

ساخت سیستمی که بتواند به صورت موثری شباهت بین دو عبارت را تشخیص دهد موضوع بسیاری از پژوهش ها بوده‌است. تعیین فاصله بین دو عبارت می‌تواند از طریق محاسبه شباهت میان واژه‌ها و یا توسط روش‌های یادگیری ماشین انجام شود. معیارهای شباهت میان دو عبارت در زمینه‌های مختلفی مانند پردازش زبان‌های طبیعی [۲]، اصلاح پرس‌وجوهای جستجو [۳]، مقایسه اسناد [۴] و دیگر زمینه‌ها کاربرد دارد. تکنیک‌های بررسی شباهت میان دو عبارت با جایگزینی کلمات مترادف، عبارات مشابه و جملات هم‌معنی انجام می‌پذیرد. ورودی این سیستم‌های می‌تواند دو کلمه، جمله، پاراگراف و یا سند باشد و خروجی سیستم

نیز میزان شباهت محاسبه شده‌است. دقت اینگونه سیستم‌ها با توجه به نزدیکی به قضاوت انسان، مشخص می‌شود.

رویکردهای متفاوتی برای محاسبه شباهت بین دو عبارت استفاده می‌شود از جمله رویکردهای مبتنی بر روش‌های آماری [۵]، رویکردهای مبتنی بر یادگیری‌ماشین [۶]، رویکردهای مبتنی درخت تصمیم [۷]، رویکردهای مبتنی بر معیارهای مختلف شباهت و تعداد کلمات مشترک و شبکه‌های عصبی [۸]. هر رویکرد روش و راهکار متفاوتی برای بررسی شباهت میان دو عبارت ارائه می‌دهد.

۲.۱ تعریف مسئله

با توجه آنچه در بخش قبل گفته شد، محاسبه شباهت دو عبارت بخش بسیار مهمی در سیستم‌هایی است که از تکنیک‌های پردازش متن استفاده می‌کنند. هدف از این پژوهش ارائه روشی است که بتوان از آن در سیستم‌های مختلفی که از تشخیص شباهت متون استفاده می‌کنند به عنوان موتور استنتاج استفاده کرد.

معیارهای شباهت بسیاری برای یافتن شباهت بین عبارت تعریف شده از جمله معیارهای مبتنی بر رشته [۱] [۹]، معیارهای مبتنی بر معنا [۲][۱۰] و معیارهای مبتنی بر پیکره [۳][۱۱].

در این پژوهش ما از معیارهای مبتنی بر معنا استفاده می‌کنیم تا شباهت بین دو جمله یا پاراگراف را تشخیص دهیم. در این روش از مدل‌های جایگذاری کلمات [۴] استفاده می‌کنیم تا ویژگی‌های معنایی مختلف از دو عبارت استخراج کرده، سپس به کمک روش‌های یادگیری ماشین و ویژگی‌های استخراج شده، عبارات مشابه را تشخیص می‌دهیم.

در این روش ورودی‌ها شامل عبارات و مدل‌های جایگذاری کلمات می‌شود. عبارت به دو گروه تقسیم می‌شوند:

۱. عباراتی که معنای مشابه دارند

۲. عباراتی که معنای مشابه ندارند

[1] String based approaches
[2] Semantic based approaches
[3] Corpus based approaches
[4] Word embeding models

هر ورودی از دو عبارت تشکیل شده که هر کدام میتواند شامل یک جمله یا بیشتر باشد. ابتدا به کمک مدل‌های جایگذاری کلمات و توابع استخراج ویژگی، برای جملات ورودی ویژگی‌های معنایی استخراج و به صورت یک بردار نمایش داده می‌شود. سپس این بردار به کمک یک دسته‌بند بردار پشتیبان [۵] به دو دسته تقسیم می‌شود.

برای ارزیابی روش از پایگاه داده استاندارد MSRP[۶] استفاده شده‌است. این پایگاه داده در سال ۲۰۰۵ توسط توسط شرکت مایکروسافت [۷] توسعه داده شده‌است. این پایگاه داده شامل ۵۸۰۰ جفت جمله است که از منابع خبری آنلاین استخراج شده‌اند.

۳.۱ کاربردهای محاسبه شباهت بین اسناد

تشخیص شباهت بین دو عبارت در زمینه‌های گوناگونی کاربرد دارد. در زیر به چند مورد اشاره شده‌است:

۱. تشخیص شباهت دو فایل صوتی، تصویری و یا متنی [۱۲]

۲. اصلاح غلط املایی [۱۳]

۳. شناسایی سرقت علمی‌ـادبی

۴. تشخیص شباهت کدهای نرم‌افزار [۱۴]

۵. استخراج پاسخ در سیستم‌های پرسش و پاسخ [۱۵]

۶. دسته بندی اسناد

۷. یافتن عبارات در موتورهای جستجو

کاربردهای زیاد تشخیص شباهت اسناد، این روش را یکی مراحل مهم و پایه‌ای در بسیاری از کاربردهای پردازش زبان طبیعی و متن‌کاوی [۸] کرده‌است.

[5] Support vector classifier
[6] Microsoft research paraphrase
[7] Microsoft
[8] Text mining

۴.۱ ساختار پایان‌نامه

در فصل دوم ابتدا روش‌ها و الگوریتم‌های ارائه شده در زمینه شباهت معنایی عبارات در سه زمینه مبتنی بر رشته، مبتنی بر پیکره و مبتنی بر پایگاه‌دانش [۹] به صورت خلاصه بررسی می‌شوند. در ادامه برخی از پژوهش‌های انجام شده در این زمینه ارائه می‌شوند.

در فصل سوم ابتدا معماری کلی روش پیشنهادی و مزیت‌های این روش نسبت روش‌های بیان شده در فصل دوم تشریح می‌شود.

در این فصل دو مورد از روش‌های تولید بردار جایگذاری کلمات که در این پژوهش از آن‌ها استفاده شده‌است نیز به صورت خلاصه بیان می‌شود.

در فصل چهارم پایگاه داده MSRP معرفی می‌شود. هم‌چنین برخی تغییراتی که نیاز است در پایگاه داده اعمال شود بیان می‌شوند. در نهایت پارامترهای استفاده شده در آزمایش‌ها و نتایج پیاده سازی ارائه می‌شود.

در فصل پنجم خلاصه‌ای از روش پیشنهادی، نتایج و هم‌چنین پیشنهاداتی برای بهبود روش ارائه می‌شود که می‌توان زمینه ساز ادامه تحقیق در آینده باشد.

[9] Knowledg based approach

فصل ۲

معیارهای محاسبه شباهت متون

۱.۲ حوزه‌های بررسی شباهت

محاسبه شباهت میان متون، نقشی اساسی در تحقیقات در زمینه پردازش متن ایفا می‌کند. این زمینه کاربردهای فراوانی در بازیابی اطلاعات، دسته‌بندی اسناد، تولید خودکار سوال، سیستم‌های پرسش و پاسخ، درجه‌بندی مقالات، ترجمه ماشینی، خلاصه سازی متون و تشخیص سرقت علمی دارد.در زمینه محاسبه شباهت میان متون یک راه‌حل این است که ابتدا شباهت بین پاراگراف‌ها و جملات محاسبه و سپس شباهت بین متون بلندتر مطرح گردد.

مشابهت کلمات می‌تواند از دو دیدگاه بررسی شود. یک دیدگاه بررسی مشابهت فقط از طریق قواعد نحوی [1] است. در این روش ما تنها با توجه ظاهر کلمات شباهت کلمات را اندازه‌گیری می‌کنیم و در نتیجه گیری از معنای کلمات استفاده نمی‌کنیم. در این روش اگر کلمات دارای رشته حروف مشابه باشند، مشابه درنظر گرفته می‌شوند در غیر این‌صورت

[1] Lexical

شباهتی با یکدیگر ندارند.

دیدگاه دیگر حوزه معنایی [۲] است. در این حوزه علاوه بر شباهت ظاهری، به معنای دو کلمه هم توجه می‌شود. نمونه‌ای از این شباهت‌ها عبارتند از: مترادف بودن و داشتن طرح موضوع یکسان.

هر دو دیدگاه محاسبه شباهت در سه دسته الگوریتم گوناگون مورد بررسی و ارزیابی قرار می‌گیرند. این سه دسته الگوریتم شامل روش‌های مبتنی بر رشته، روش‌های مبتنی بر پیکره و روش‌های مبتنی بر دانش هستند. حوزه شباهت لغوی در دسته مبتنی بر رشته و حوزه معنایی در دو دسته مبتنی بر پیکره و مبتنی بر دانش مورد بررسی قرار می‌گیرند [۱۶].

معیارهای اندازه‌گیری مبتنی بر رشته براساس ترکیب و توالی حروف در یک کلمه طراحی و پیاده‌سازی می‌شوند. معیارهای مبتنی بر پیکره، معیارهای مبتنی بر شباهت معنایی می‌باشند که شباهت بین کلمات و عبارت‌ها را براساس اطلاعات استخراج شده از یک سند بزرگ محسابه می‌کنند. معیارهای مبتنی بر دانش نیز معیارهای شباهت معنایی هستند که درجه شباهت بین کلمات و عبارت‌ها را با استفاده از دانش کسب شده از شبکه‌های معنایی و یا منابع خارجی دانش مانند فرهنگ لغت‌ها، اندازه‌گیری می‌کنند.

در ادامه بعضی از الگوریتم‌هایی که در این سه حوزه معرفی شده‌اند بررسی می‌شوند.

۲.۲ معیارهای مبتنی بر رشته

این دسته از معیارها به طور کلی براساس ترکیب و ترتیب حروف، شباهت میان کلمه‌ها را بیان می‌کنند. این معیارها شباهت را با عددی که بین صفر و یک است نمایش می‌دهند. این الگوریتم‌ها به دو دسته مبتنی بر نوالی حروف و مبتنی بر توالی عبارات تقسیم می‌شوند که در ادامه به آن‌ها می‌پردازیم.

۱.۰.۲.۲ معیار شباهت مبتنی بر حرف

در این بخش برخی از الگوریتم‌ها که وظیفه محاسبه شباهت دو عبارت یا کلمه را دارند بررسی می‌شود. این روش‌ها معمولا براساس افزودن، حذف و یا تغییر مکان یک حرف در کلمه شباهت

[2] Semantic

را تشخیص می‌دهند. از این روش‌ها معمولا در برنامه‌های اصلاح نگارش و تشخیص خطاهای املایی استفاده می‌شود. یک معیار پایه برای بررسی شباهت لغوی تعداد عملیات‌های حذف، اضافه و یا تغییر مکان برای تبدیل یک کلمه به کلمه دیگر است که به فاصله ویرایش [۳][۱۷] معروف است.

به عنوان مثال دو کلمه زیر را در نظر بگیرید:

$$T_1 = "asdf"$$

$$T_2 = "zasdf"$$

$$ED(T_1, T_2) = 1$$

فاصله ویرایش در مثال بالا برابر ۱ است، زیرا با حذف حرف z از ابتدای T_2 می‌توان رشته T_1 را تولید کرد. در این روش می‌توان یک حد آستانه t تعریف کرد، در صورتی فاصله ویرایش از t کمتر بود آن‌گاه دو کلمه مشابه در نظر گرفته می‌شود. در ادامه چند مورد از الگوریتم‌های ارائه شده براساس فاصله ویرایش بررسی می‌شوند.

۱. الگوریتم بزرگترین زیر رشته مشترک [۴]

این الگوریتم که جز قدیمی‌ترین مسئله‌های علمم کامپیوتر است، روشی برای پیدا کردن بزرگترین زیر رشته مشترک دو رشته است.

این الگوریم پایه‌ای برای برنامه‌های مقایسه فایل است که در نهایت فایل‌های مشابه را پیدا می‌کنند.

بزرگترین زیر رشته مشترک برای دو رشته T_1 و T_2 به رشته ای مانند T_3 است به طوریکه حرف‌های موجود در رشته T_3 با حفظ ترتیب و نه الزاما توالی آن‌ها، در هر دو رشته T_1 و T_2 موجود باشند و T_3 بزرگ‌ترین رشته‌ای باشد که این خاصیت را دارد.

برای حل این مسئله می‌توان از برنامه‌نویسی پویا [۵] استفاده کرد.

۲. الگوریتم Levenshtein[۱۸]

[3] Edit distance
[4] Longest Common Substring
[5] Dynamic Programming

این الگوریتم تابع فاصله بین دو رشته را به صورت کمترین تعداد عملیاتی که نیاز است تا یک رشته به دیگری تبدیل شود، تعریف می‌کند. علاوه بر سه عملیات مورد استفاده در فاصله ویرایش، گاهی از ترانهاده کردن دو حرف مجاور نیز استفاده می‌شود.

۳. الگوریتم جارو [۶][۲]

این الگوریتم براساس محل قرارگیری و تعداد کاراکترهای مشترک یک رشته معیار فاصله را تعریف می‌کند. این الگوریتم یک درجه پیشوند دارد که باعث می‌شود رشته‌هایی که از ابتدا با یکدیگر مشابه‌اند رتبه بهتری داشته باشند.

این الگوریتم در حوزه ارتباط رکوردها و تشخیص اسناد و موجودیت‌های تکراری استفاده می‌شود. این معیار برای رشته‌هایی که طول کمتری دارند بهتر عمل می‌کند. بعد از انجام عملیات، نتیجه به کمک یک تابع نرمال‌سازی تبدیل به صفر یا یک می‌شود. صفر بیانگر عدم شباهت و یک نشان شباهت است.

معیار فاصله جارو که با d_j نشان داده می‌شود برای دو رشته ورودی S_1 و S_2 به صورت زیر تعریف می‌شود:

$$d_j = 1/3(\frac{m}{|S_1|} + \frac{m}{|S_2|} + \frac{m-t}{t}) \qquad (۱.۲)$$

در معادله ۱.۲ m بیانگر تعداد کاراکترهایی است که بین دو رشته مشترک [۷] است و t نصف تعداد جابجایی‌ها است. دو حرف s_1 و s_2 از هر رشته با هم مشترک در نظر گرفته می‌شوند در صورتی که با هم برابر باشند و فاصله آن‌ها در دو رشته از مقدار d در معادله زیر بیشتر نباشد:

$$d = |\frac{max(|S_1|, |S_2|)}{2}| - 1 \qquad (۲.۲)$$

برای بدست آوردن تعداد حرف‌های مشترک، تمام حروف یکسان طبق رابطه بالا مقایسه می‌شوند.

[6] Jaro
[7] Matching character

یک نسخه کامل شده از این الگوریتم جارو_وینکلر [۸][۱۹] نام دارد که همانطور که گفته شد از یک درجه پیشوند(P) استفاده کرده. این روش برای رشته‌هایی که از ابتدا تا L کاراکتر با هم برابرند ارزش بالاتری قائل می‌شود.

فاصله جارو_وینکلر برای دو رشته S_1 و S_2 به صورت زیر محاسبه می‌شود:

$$d_w = d_j + (L_p(1 - d_j)) \tag{۳.۲}$$

در برخی از نسخه‌های الگوریتم جارو_وینکلر، مقدار جایزه $L_p(1 - d_j)$ تنها در صورتی اعمال می‌شود که دو رشته فاصله جارو بیشتر از حد تعیید شده‌ای داشته‌باشند، در نتیجه رابطه ۳.۲ به صورت زیر بازنویسی می‌شود:

$$d_w = \begin{cases} d_j & d_j < d_t \\ d_j + (L_p(1 - d_j)) & otherwise \end{cases} \tag{۴.۲}$$

۴. الگوریتم n ـ گرام [۲۰] این الگوریتم یک زیر سری از n آیتم از رشته ورودی تشکیل می‌دهد. الگوریتم‌های مشابهت مبتنی بر n ـ گرام با مقایسه گرام‌های هر کاراکتر یا متن در دو رشته، فاصله را محاسبه می‌کند. در نهایت فاصله با تقسیم ساده n ـ گرام‌های مشابه بر تمامی n ـ گرام‌ها بدست می‌آید.

۲.۰.۲.۲ معیارهای مبتنی بر عبارت

در این بخش الگوریتم‌هایی را مورد بررسی قرار می‌دهیم که عملیات مختلف را برخلاف روش‌های قبلی که روی کاراکترها انجام می‌شد، بر روی عبارات و کلمات انجام می‌دهند. این روش‌های ابتدا ورودی‌ها را به مجموعه‌ای از توکن‌ها [۹] تبدیل می‌کنندو سپس از معیارهای شباهت مبتنی بر عبارت برای تشخیص و محاسبه میزان شباهت استفاده می‌کنند. معیارهای مبتنی بر توکن برای عبارت‌های طولانی مانند سندها مناسب هستند. در این روش‌ها برای تبدیل ورودی به دنباله‌ای از توکن‌ها به‌طور کلی دو مرحله استفاده می‌شود [۲۱]:

[8] Jaro-Winkler
[9] Token

۱. پیش پردازش و تبیدل ورودی به توکن‌ها

۲. تبدیل توکن‌ها به n ـ گرام‌ها

در بخش اول، رشته‌ها براساس حروف مشخصی توکن‌بندی می‌شوند. به عنوان مثال کاراکتر فضای خالی و یا Tab نمونه‌هایی از حروفی هستند که برای توکن‌بندی استفاده می‌شوند. قسمت دوم از زیر رشته‌هایی با طول n برای ایجاد سری استفاده می‌کند که به آنها n ـ گرام می‌گویند.

برای سادگی، هر عنصر در یک سری ـ کلمه یا گرام ـ توکن می‌نامند. از معیارهای موجود برای توکن‌ها شامل معیار همپوشانی [۱۰] ، معیار جاکارد [۱۱] ، فاصله کوسینوسی [۱۲] و دایس [۱۳] [۹] هستند. در ادامه هر یک را تشریح می‌کنیم.

۱. فاصله بلوک

این الگوریتم با نام فاصله منهتن نیز شناخته می‌شود. در این الگوریتم فاصله بین المان‌ها که به صورت خانه‌های شطرنج تعریف می‌شود بر اساس رابطه ۵.۲ محاسبه می‌شود:

$$D_m(V,W) = \sum_{i=1}^{n} |v_i - w_i| \qquad (۵.۲)$$

در رابطه بالا $\vec{V}$ و $\vec{W}$ بردارهایی به طول n هستند.

۲. فاصله کسینوسی

این معیار یک روش ارزیابی برای بردارها ارائه می‌کند. در این الگوریتم دو بردار ورودی در فضای برداری نمایش داده می‌شوند و سپس با اندازه‌گیری کسینوس زاویه بین دو بردار فاصله این دو ورودی محاسبه می‌شود.

در زیر روش محاسبه فاصله کسینوسی برای دو بردار به طول n ارائه شده است:

[10] Overlap
[11] Jacard
[12] Cosine distance
[13] Dice

$$D_c(\vec{V},\vec{W}) = \cos\theta = \frac{A.B}{|A||B|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \qquad (۶.۲)$$

هنگام اعمال رابطه فوق برای کاربردهای پردازش متن، درایه‌های هر آرایه همان توکن‌ها هستند. هر چه زاویه بین دو بردار کمتر باشد، شباعت دو بردار بیشتر است.

۳. روش دایس

این معیار نیز مانند سایر معیارها، برای عبارات مشابه مقدار بیشتری را ایجاد می‌کند. این معیار به صورت زیر تعریف می‌شود:

$$Dice(P_1,P_2) = \frac{2 * |P_1 \cap P_2|}{|P_1| + |P_2|} \qquad (۷.۲)$$

P_1 و P_2 هر کدام مجموعه‌ای از بردارها و یا n ـ گرام‌ها هستند.

۴. فاصله اقلیدسی [۱۴]

فاصله اقلیدسی معیاری است براساس اختلاف عناصر دوبردار ورودی و مانند فاصله بلوکی برای دو بردار کاربرد دارد. رابطه این فاصله به صورت زیر است:

$$E_d(\vec{V},\vec{W}) = \sqrt{\sum_{i=1}^{n}(v_i - w_i)^2} \qquad (۸.۲)$$

فاصله اقلیدسی حالت خاصی از فاصله مینکوفسکی [۱۵] است. این فاصله برای هر q دلخواه به صورت زیر تعریف می‌شود:

$$M(\vec{A},\vec{B}) = \sqrt[q]{\sum_{i=1}^{n}(|A_i - B_i|)^q} \qquad (۹.۲)$$

[14] Euclidean distance
[15] Minkowski distance

۵. معیار جاکارد

این معیار نیز برای یافتن شباهت مجموعه‌ها مناسب است و به صورت زیر تعریف می‌شود:

$$Jacard(P_1, P_2) = \frac{|P_1 \cap P_2|}{|P_1| + |P_2| - |P_1 \cap P_2|} \qquad (۱۰.۲)$$

حالت دیگری از این رابطه به نام ضریب همپوشانی وجود دارد که در یک حالت خاص که یک رشته زیر رشته‌ای از دیگری است آنها را کاملا یکسان در نظر می‌گیرد:

$$Overlap(P_1, P_2) = \frac{|P_1 \cap P_2|}{min(|P_1|, |P_2|)} \qquad (۱۱.۲)$$

۳.۲ معیارهای مبتنی بر پیکره

این معیارها شباهت میان دو عبارت را براساس معنا و محتوای بدست آمده از یک پیکره بزرگ اندازه‌گیری می‌کنند. پیکره مجموعه‌ای از متون نوشته شده و یا مجموعه‌ای گفتاری است که در تحقیقات مرتبط با پردازش زبان‌های طبیعی مورد استفاده قرار می‌گیرد.

۱.۰.۳.۲ معیارهای شباهت مبتنی بر دانش

این معیارها شامل روش‌هایی است که برای تعیین درجه شباهت به کمک یک منبع خارجی دانش مانند یک شبکه معنایی استفاده می‌شوند. یکی از معروف‌ترین شبکه‌های معنایی WordNet است که برای تعیین شباهت معنایی طراحی شده و استفاده می‌شود. WordNet یگ پایگاه بزرگ لغوی زبان انگلیسی است که در آن اسم‌ها، فعل‌ها، صفات و قیدها در یک مجموعه مترادف شناختی [۱۶] گروه‌بندی می‌شوند. به هر یک از این گروه‌ها یک Synset گفته می‌شود. هر کدام از Synset ها یک مفهوم مشخص می‌کنند. این مجموعه ها از طریق ارتباطات نحوی و معنا شناختی آنها به هم متصل می شوند.

[16] Cognitive Synonym

تعیین شباهت مبتنی بر دانش به دو گروه مجزای معیار شباهت معنایی و ارتباط معنایی تقسیم می شود. معیار شباهت معنایی بر اساس قرابت معنایی دو ورودی در شبکه واژگان یا واژه نامه ها شباهت را محاسبه میکند، اما ارتباط معنایی الزاما به شکل و فرم دو عبارت وابسته نیست. به عبارت دیگر شباهت معنایی نوعی از ارتباط بین دو کلمه است که یک محدوده از ارتباطات بین مفاهیم مختلف شامل ارتباطات معنایی دیگری از قبیل ''نوعی از'' [۱۷]، ''نمونه ای از'' [۱۸]، ''بخشی از'' [۱۹] و ''متضاد با'' [۲۰] را شامل می شود.

۱.۳.۲ روش‌های لغوی محاسبه شباهت بین دو رشته

در قسمت پیشین بر روش‌های مختلف محاسبه درجه شباهت اسناد مرور کلی داشتیم. در این قسمت ما روی شباهت میان رشته ای از کلمات (به اختصار رشته) در یک فضاب تک زبانه و روش های مستقل از زبان تمرکز خواهیم داشت. برخی از این روش‌ها در تشخیص سرقت علمی کاربرد دارد.

۱.۱.۳.۲ روش فشرده سازی دو رشته

روش های مبتنی بر تکنیک های فشرده سازی فایل ها در واقع به صورت مستقیم از الگوریتم های فشرده سازی استفاده میکنند. در این روش دو شند با یکدیگر ادغام شده و سپس فشرده سازی می شود همچنین هر دو سند به صورت جداگانه فشرده می شوند و در نهایت حجم فایل های فشرده سازی شده مجموع حجم فایل هایی که جداگانه فشرده شده اند، مقایسه می‌شود. اگر دو سند متفاوت باشند اندازه فایل حاصل شده از دو حالت فشرده سازی، برابر خواهد شد، اما اگر این هر کدام از سندها دارای قسمت های تکراری از سند دیگر باشد، اندازه حالت اول (فشرده سازی یک فایل حاوی دو سند) کمتر می شود. از مزیت های این روش سادگی و سهولت در پیاده‌سازی است اما این روش نمی تواند حالت پیچیده و جابجایی های کلمات و جملات را تشخیص دهد.

[17] Is A Kind Of
[18] Is An Example Of
[19] Is A Part Of
[20] Is the Opposite Of

۲.۱.۳.۲ مدل فضای برداری

مدل فضای برداری روش بر اساس محاسبه فرکانس هر کلمه در پیکره یا سند و تشکیل بردار وزنی TF-IDF[۲۱] بر اساس آن می‌باشد. شباهت بین بردارها در این روش با کمک روش های اندازه گیری شباهت برداری که بعضی از آن ها را مرور کردیم، انجام می‌گیرد. اطلاعات به شکل بردار در یک فضای چند بعدی نشان داده می شود. که هر بعد در این فضا، مربوط به یکی از ویژگی های اطلاعاتی مانند کلمه در شند می باشد. یک تابع فاصله روی بردار ها برای محاسبه انطباق و رتبه بندی اطلاعات اعمال می شود. هر کدام از روابط گفته شده در بخش شباهت مبتنی بر عبارت را می توان برای این منظور استفاده کرد. معمولا از معیار شباهت کسینوسی برای این مدل استفاده می‌شود.

بازیابی اطلاعات مبتنی بر مدل فضای برداری سازی ریاضی مناسب برای پردازش منابع اطلاعاتی بزرگ است و امکان تطبیق جزئی و رتبه‌بندی خروجی را فراهم می‌کند. با این حال این روش نمی تواند روابط معنایی بین کلمات را نمایش دهد. امروزه بسیاری از روش‌هایی که برای نمایش اسناد استفاده می شود بر مدل کیسه کلمات [۲۲] تکیه می‌کنند از این رو این روش به طور معمول به عنوان مدل فضای برداری شناخته می‌شود.

در این مدل اسناد به صورت یک بردار خطی از وقوع کلمات در پیکره نشان داده می‌شود به این شکل که در صورتی که سند شامل یه کلمه باشد جایگاه آن کلمه در بردار برابر ۱ و در غیر این صورت برابر ۰ است. یکی از مشکلات این روش این است که بسیاری از روابط معنایی ببین کلمات و سایر اطلاعات معنایی موجود در سند هنگام استفاده از مدل فضای برداری از دست می‌رود. همچنین این روش هنگامی که سند طولانی است، کارآمد نیست زیرا نمایش سند به صورت مدل برداری با توجه به ابعاد بسیار بالا بردار ایجاد شده، دشوار می باشد و انفجار فضای برداری رخ خواهد داد.

۳.۱.۳.۲ محاسبه شباهت بین پاسخ‌ها در سیستم‌های پرسش و پاسخ

این روش در پژوهش مربوط به محاسبه شباهت بین پاسخ ها در پرتال‌های پرسش و پاسخ آنلاین منتشر شده است .این پژوهش روشی مبتنی بر شکل کلمات و عبارات تشکیل دهنده یک

[21] Term frequency - Inverse document frequency
[22] Bag Of Words

جدول ۱.۲: مثالی از شباهت جاکارد

پارامتر	مقدار
تعداد کل کلمات در S_a	6
تعداد کل کلمات در S_b	5
تعداد کلمات مشترک در دو جمله	4
تعداد کلمات دو جمله	7
میزان شباهت جاکارد	0.57

جمله ارائه می دهد. در سیستم پرسش و پاسخ از الگوریتمهای محاسبه شباهت بین عبارات و شباهت بین دو سوال یا بین دو پاسخ استفاده می شود. با استفاده از معیارهای متفاوت شباهت به ویژه معیارهای شباهت معنایی و معیارهای تشابه آماری، می توان به این هدف دست یافت. بدلیل تعداد زیاد پرسش و پاسخ ها، پیاده سازی روش های شباهت معنایی بسیار پیچیده و زمانبر می باشد، بنابراین استفاده از الگوریتمهای شباهت آماری مقرون به صرفه است. تمرکز اصلی در اینجا این است که تشابه پاسخ کاربر را با پاسخ ذخیره شده در پایگاه داده پرسش و پاسخ بهبود بخشیده.

براساس آنچه در قبل گفته شد، شباهت جاکارد معیاری برای اندازه گیری شباهت مجموعه ها است. در اینجا نیز می توان با تبدیل هر جمله به مجموعه ای از توکن ها، از این معیار برا اندازه گیری شباهت بین دو جمله استفاده کرد. اگر S_a و S_b دو جمله باشند، برای مقایسه نیاز است که در ابتدا دو جمله به مجموعه ای از توکن ها تبدیل شوند. فرض کنید که دو جمله مانند زیر باشند:

S_a = من به ماشین‌های قرمز علاقه‌ای ندارم.

S_b = من به آنها علاقه‌ای ندارم.

پس از تبدیل هر یک به مجموعه ای از توکن‌ها، به شکل زیر در می‌آیند:

S_a = [''من'' ، ''به'' ، ''ماشین‌های'' ، ''قرمز'' ، ''علاقه‌ای'' ، ''ندارم'']

S_b = [''من'' ، ''به'' ، ''آن‌ها'' ، ''علاقه‌ای'' ، ''ندارم'']

شباهت جارکارد برا بدست آوردن شباهت مجموعه‌ها بسیار کارآمد است، اما به مسئله

دقت پاسخ ها در سیستم‌های پرسش و پاسخ نمی‌پردازد. برای بهبود این روش از معیار هم پوشانی می توان استفاده کرد. این معیار عبارات مشترک بین دو ورودی را مد نظر قرار می‌دهد در نهایت معیار شباهت به صورت زیر خواهد بود:

$$OverlapSimilaroty = \frac{W_{S_a} \cap W_{S_b}}{min(W_{S_a}, W_{S_b})} \quad (۱۲.۲)$$

در بعضی از حالت ها مانند سیستم پرسش و پاسخ چند سوال وجود دارد که پاسخ آن‌ها می‌تواند یک کلمه، یک خط یا در یک پاراگراف بیان شود. به عنوان مثال برای سوال ’’پایتخت هند کجاست؟’’ می توان سه پاسخ مختلف زیر را درنظر گرفت:

۱. پایتخت هند دهلی است

۲. دهلی پایتخت هند است

۳. دهلی

همه این پاسخ ها با توجه به سوال داده شده درست است. تنها تفاوت آن‌ها طول جملات است. دو مورد اول پاسخ‌های جامع‌تری به سوال مطرح شده بودند زیرا تنها برای این سوال کاربرد دارند در صورتی که پاسخ سوم برای سوال‌های دیگری نیز قابل استفاده است.

۴.۲ روش‌های مبتنی بر اثر انگشت

روش های مبتنی بر الگوریتم اثر انگشت [۲۳] و هش [۲۴] از روش‌های مبتنی بر محتوا به شمار می روند. روش های مبتنی بر محتوا به مقایسه صریح محتویات یک سند با شیوه نمایش مخصوص به خود می پردازند. در مدل اثر انگشت و مدل های بر پایه هش از یک مجموعه از اعداد صحیح برای کدگذاری کردن بخشی یا تمام سند، برای نمایش محتویات آن سند استفاده می شود.

فرآیند ایجاد اثر انگشت، انگشت نگاری نامیده می شود. به کمک اثر انگشت سند میتوان آن سند را شناسایی کرد. n ـ گرام به عنوان الگوریتم پایه برای بسیاری از روش های اثر انگشت استفاده می شود، زیرا فرایند انگشت‌نگاری سند را به گرام هایی به طول از پیش تعیین شده

[23] Fingerprint
[24] Hash

n تقسیم می کند. مدل های مبتنی بر هش نیز از یک تابع هش برای تبدیل اثر انگشت به مقدار هش که یک معمولا یک عدد است، استفاده می‌کنند. در نهایت با تبدیل تمامی توکن ها یا اثرانگشت های یک سند، می توان برداری از اثرانگشت های آن سند ساخت و شباهت بین دو سند را با روش های شباهت برداری محاسبه کرد. گرام ها جزء ویژگی های واژگانی محسوب می شوند که این ویژگی های واژگانی می‌تواند در سطح کاراکتر یا کلمه عمل کنند. این ویژگی ها را می توان به دو شکل مختلف گرام-n مبتنی بر کاراکتر [۲۵] (CNG) که دنباله ای از کاراکترها است و یا مبتنی بر کلمات یا [۲۶] (WNG) که دنباله ای از چند کلمات با نادیده گرفتن جملات و مرزهای ساختاری و طول متفاوت هستند، در نظر گرفت. WNG ساده ممکن است به صورت گرام ۲ تایی یا گرام سه تایی و بزرگتر ساخته شود. در زمینه بازیابی متن و تحقیقات محاسبه شباهت ای ن دو روش به نام اثر انگشت و یا شینگل شناخته می شوند. بعد از استخراج شینگل ها می توان مسئله شباهت متون را به صورت شباهت مجموعه ای در نظر گرفت و از روش ها شباهت مجموعه ها که بعضی از آن ها نیز شرح داده شد، استفاده کرد. مهم ترین ویژگی روش انگشت نگاری سریع بودن آن است که می‌تواند به طور موثر در مجموعه‌ای بزرگ استفاده شود و در عین حال نیز یکی از معایب این روش در نظر نگرفتن اطلاعات معنای جملات و عبارات می باشد.

۵.۲ کارهای پیشین

روش های بسیاری برای تشخیص شباهت بین عبارت ها و سندها در مورد بازیابی اطلاعات وجود دارد. یکی از شناخته شده ترین روشها در این زمینه، روش VSM است [۲۲]. که از تابع فاصله کسینوسی استفاده می‌کند. از این روش برای اندازه گیری شباهت بین جمله ها استفاده می شود. در [۲۳] به مسئله تشخیص شباهت بین جملات به صورت یک مسئله هم ترازی [۲۷] نگاه شده‌است. در این روش ابتدا دو پاراگراف با یکدیگر هم تراز [۲۸] شده و سپس جملات هر پاراگراف با جملات پاراگراف دیگر هم تراز می‌شود. در هر دو مرحله، هم ترازی پاراگراف ها و جمله ها به کمک شباهت کسینوسی انجام می‌شود. در [۲۴] نیز با رویکردی

[25] Character based n-grams
[26] Word based n-grams
[27] Alignment
[28] Align

مشابه برای بررسی شباهت بین جملات استفاده شده و از شباهت کسینوسی در یک مدل رگرسیون [۲۹] استفاده شده است، در نتیجه با امتیاز دهی به جملات، جمله ها با یکدیگر هم تراز شده و جمله های مشابه تشخیص داده می‌شوند. این دو روش از زمینه [۳۰] اطراف هر جمله برای تشخیص شباهت استفاده می‌کنند و هنگام هم‌ترازی، تمامی جمله‌ها به صورت اعضای یک رابطه Many-to-Many بررسی می‌شوند.

از ویژگی های زبان شناسی [۳۱] نیز می‌توان برای ایجاد بردارهای ویژگی و مدل فضای برداری استفاده کرد. در [۲۵] از ویژگی‌های زبان شناختی برای ایجاد مدل فضای برداری استفاده شده است تا به کمک این بردارها و یک دسته‌بند با ناظر [۳۲] پاراگراف‌های مشابه تشخیص داده شوند. ویژگی‌های زبان شناختی که در ساخت این مورد از آنها استفاده شده عبارت هستند از: همخوانی عبارت‌های اسمی [۳۳] کلمات هم‌معنی، شکل اسمی افعال متداول [۳۴] اسامی مشترک و ترکیب‌های آن‌ها. حتی با توجه به اینکه این روش به نسبت به شباهت کسینوسی بهتر عمل می‌کند این روش مبتنی بر منابع ساخت یافته دانش مانند دیکشنری WordNet است که برای بسیاری از زبانها موجود نیستند و یا به سختی قابل ایجاد هستند. در ادامه روش‌هایی که به عنوان روش‌های پایه برای مقایسه عملکرد روش پیشنهادی تحقیق استفاده شده‌اند تشریح می شود.

این دو روش به این منظور انتخاب شده‌اند با مقایسه با نتایج حاصل از این آزمایشات این تحقیق بتوانیم به این سوال پاسخ دهیم که آیا این روش با کمترین استفاده از منابع دانش‌خارجی و کمترین میزان محاسبات در مقتیسه با روش‌هایی که کاملا به منابع دانش‌خارجی و یا استخراج ویژگی‌ها با محاسبات پیچیده متکی هستند، نتایج قابل مقایسه‌ای ایجاد کند یا خیر.

[29] Regression
[30] Context
[31] Linguistics
[32] Supervised Classifier
[33] Noune phrase matching
[34] Common

SVSM ۱.۵.۲

این روش که در [۲۶] معرفی شده‌است مبتنی بر مدل فضای برداری است با این تفاوت که بر خلاف مدل BoW طول بردارهای ایجاد شده، با طول مجموعه کلمات سند برابر نیست و بر اساس بردارهای کلمه ساخته می‌شود. با داشتن یک پیکره C که از n جمله با m کلمه ی یکتا [۳۵] تشکیل شده است بردار کلمه T_j یک بردار n بعدی است که در آن هر مولفه بیانگر این است که کلمه در کدام جمله ها به کار رفته است و به صورت زیر نمایش داده می‌شود:

$$\vec{t_j} = [x_1, x_2, ..., x_n] \qquad x_i \in \{0, 1\}, i \in [1, n] \tag{۱۳.۲}$$

در این بردار اگر مولفه i برابر ۱ باشد به معنای این است که کلمه در جمله i استفاده شده و اگر ۰ باشد به معنای این است که کلمه در جمله i استفاده نشده است. این نمایش برداری با نمایش برداری که در [۲۷] معرفی شده شباهت دارد بااین تفاوت که در [۲۷] به جای صفر و یک، توزیع کلمات که عددی پیوسته است استفاده می‌شود.

این شیوه نمایش، یک ماتریس برای پیکره ایجاد میکند که ابعاد این ماتریس $m \times n$ است و با افزایش تعداد جملات اندازه آن افزایش می یابد. روش‌های مختلفی برای کاهش سایز این ماتریس پیشنهاد شده‌اند. روش‌هایی مانند حذف کلمات توقف [۳۶] و یا ریشه‌یابی [۳۷] اسم‌ها و فعل‌ها بعضی از روش های مبتنی بر ویژگی‌های زبان شناختی هستند. روش های ریاضیاتی و آماری نیز وجود دارند که می‌توان از آنالیز معنای پنهان [۲۸] [۳۸] و تحلیل مولفه های اساسی [۳۹] نام برد. در روش LSA از تجزیه مقادیر تکین [۴۰] ماتریس استفاده می شود تا ماتریسی کوچکتر حاصل شود و در عین حال ویژگی‌های آماری ماتریس اصلی تا حد امکان حفظ شوند. تحلیل PCA شامل تجزیه مقدارهای ویژه ماتریس کواریانس است که داده ها را به دستگاه مختصات جدید می‌برد به‌طوری‌که بزرگترین واریانس داده بر روی اولین محور مختصات و دومین بزرگترین واریانس بر روی دومین محور مختصات قرار می‌گیرد.

[35] Unique
[36] Stop word
[37] Stemming
[38] Latent semantic analysis
[39] Principal Component Analysis
[40] Singular

در این تحقیق از شیوه حذف حرف‌های توقف استفاده شده که روش ساده‌تری است. در این شیوه نمایش ماتریسی، حتی بعد از حذف کلمات توقف تعداد بسیار زیادی از مولفه ها صفر هستند که باعث می شود ماتریس حاصل یک ماتریس خلوت [۴۱] بشود. برای رفع این مشکل، شیوه نمایش بردارها براساس شیوه نمایش ماتریس های خلوت تغییر می کند:

$$\vec{t_j} = [(s_1, 1), (s_2, 1), ..., (s_n, 1)] \quad (۱۴.۲)$$

در رابطه ۱۴.۲ هر مولفه $(s_i, 1)$ به معنای این است که کلمه متناظر در جمله iام استفاده شده است.

این بردار کلمه احساسات مختلفی که کلمه در جمله های مختلف دارد را نمایش میدهد. منظور از احساسات مفهومی است که کلمه در آن جمله با آن در ارتباط است.

در این روش فرض شده است که جمله ها از نظر مفهومی نسبت به یکدیگر مستقل هستند و هر جمله یک مفهوم مجزا از سایر جملات را بیان می کند. در نتیجه اگر کلمه ای در هر یک از این جمله ها استفاده شود می توان نتیجه گرفت که آن کلمه با مفهوم آن جمله در ارتباط است. این فرضیه در عمل امکان پذیر نیست اما تاثیری که این فرضیه بر عملکرد مدل دارد را می توان با تکنیک خوشه بندی مانند خوشه بندی سلسله مراتبی کاهش داد، به طوریکه با خوشه بندی می توان گروه‌هایی از جمله‌ها داشت که یک مفهوم مشترک را بیان می‌کنند.

بعد از بدست آوردن بردار کلمات، می‌توان بردار هر جمله را ایجاد کرد. به این صورت که بردار تمام کلمات آن جمله را با یکدیگر جمع می‌کنیم و یک بردار n بعدی ایجاد می شود. عناصر هر بردار کلمه، از مقادیر منطقی (صفر و یک) تشکیل شده‌است، بنابراین اضافه کردن این بردارها به یکدیگر نمی تواند به خوبی بیانگر معنی باشند زیرا بعد از جمع کردن دو بردار، تمام بردارها تقریبا با هم برابر می شوند. برای رفع این مشکل در هنگام جمع کردن کلمات با یکدیگر، مقدار IDF هر کلمه نیز با آن اضافه می شود. مقادیر IDF براساس پیکره مورد استفاده استخراج شده و در هنگام محاسبه آن به جای در نظر گرفتن سند، از جمله استفاده شده است. در نتیجه شیوه به دست آوردن مولفه i ام بردار جمله را برای جمله ای که از کلمه های T_1، T_2، و T_n تشکیل شده است، به صورت زیر است:

[41] Sparse

$$d_i = \sum_{j=1;t_j \in S_i}^{n} \tag{۱۵.۲}$$

این روش مشابه روش شباهت مرتبه دوم در [۲۹] است با این تفاوت که اطلاعات بیشتری را در بردارها ذخیره می‌کند. سه عامل باعث می‌شود ذخیره اطلاعات به شکلی برداری در این روش بیشتر باشد:

- اهمیت هر کلمه به کمک IDF در بردار جایگذاری می شود
- هم رخدادی کلمات با توجه جمع شدن مقدار IDF آن‌ها
- در اختیار داشتن توزیع کلمات در سایر جمله‌ها

بعد از بدست آوردن این بردارها میتوان از توابع شباهت که برای بردارها معرفی شده اند استفاده کرد. در روش SVSM [۴۲] از شباهت کسینوسی استفاده شده است. با توجه به اینکه در روش بردارها براساس توزیع کلمات در جمله ها ساخته می شود، هرچه اندازه پیکره بزرگتر باشد، عملکرد بهتری خواهد داشت. در ادامه نتایج پیاده سازی این روش آورده شده است.

جدول ۲.۲: نتایج آزمایش‌ها بر روی پایگاده داده استاندارد MSRP

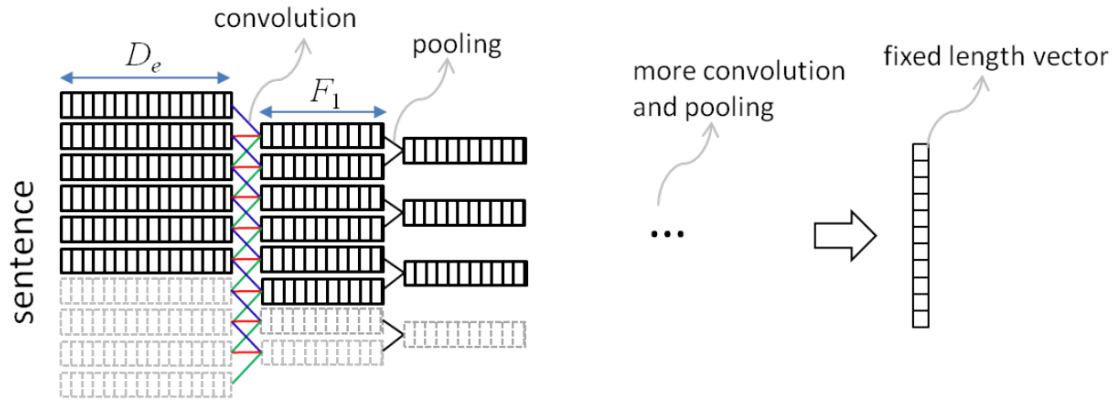
F1	Recall	Precicion	Accuracy	
79.0	87.9	71.7	68.9	SVSM

همانطور که در نتایج پیاده سازی و آزمایش بر روی پایگاه داده استاندارد MSRP می توان مشاهده کرد، این روش قابلیت بالایی در تشخیص جملات مشابه (مقدار زیاد Recall نسبت به سایر معیارها) دارد و جملاتی که مشابه نیستند (طبق تعریف Precision جمله هایی که در دسته TN قرار میگیرند) را با دقت کمتری تشخیص می دهد.

۲.۵.۲ شبکه‌های عصبی کانولوشنی

در این قسمت الگوریتم دیگری که به عنوان یکی از روش‌های پایه برای مقایسه عملکرد انتخاب شده‌اند شرح داده می‌شود. در این روش شباهت دو جمله به کمک مدل‌های مبتنی بر شبکه

[42] Short text based Vector Space Model



شکل ۱.۲: معماری کلی شبکه عصبی استفاده شده. مستطیل‌هایی که با خطچین در حاشیه‌ها مشخص شده‌اند، صفرهای اضافه شده به بردارها هستند که توسط توابع Gate حذف شده‌اند.

عصبی کانولوشنی [۴۳] بررسی می شود. یکی از مزایای این روش ها این است که نیازی به دانش پیشین در مورد زبان ندارند و می‌توانند برای هر زبانی استفاده بشوند. در ادامه ابتدا مدل پایه شبکه عصب کانولوشنی برای تشخیص شباهت بین جملات ارائه می‌شود و سپس مدل‌های این روش معرفی می‌شوند.

۱.۲.۵.۲ مدل جمله شبکه کانولوشنی

همانطور که در تصویر ۱.۲ پیداست ورودی این مدل بردار جایگذاری کلمات است که معمولا با روش های یادگیری بدون ناظر آموزش داده می شود. این بردارها براساس ترتیب جمله ها در پیکره و مکان هر کلمه در جمله وارد شبکه عصبی می شوند، در نتیجه هر لایه از شبکه عصبی معنای جمله را خلاصه‌تر می‌کند تا اینکه در لایه آخر یک نمایش برداری با طول ثابت از جمله ایجاد می‌شود.

مانند بسیاری از مدل‌های کانولوشنی دیگر [۳۰] و [۳۱] در این روش از واحدهای کانولوشنی با زمینه پذیرش [۴۴] محلی و وزن‌های اشتراکی استفاده شده‌است با این تفاوت که نگاشت ویژگی بزرگی ایجاد شده‌است تا بتواند به خوبی ترکیب‌های مختلف کلمات را نمایش بدهد. همانطور که در تصویر ۱.۲ قابل مشاهده است، لایه کانولوشنی در لایه اول شبکه عصبی بر

[43] Convolution
[44] Receptive field

روی پنجره‌های لغزنده بردار کلمات با عرض k_1 عمل می کند.

به صورت کلی برای جمله ورودی x ، در لایه کانولوشن تابع نگاشت ویژگی‌های نوع f (یکی از تابع‌های تعریف شده در F_l) در لایه l به صورت زیر است:

$$z_i^{(l,f)} \stackrel{def}{=} z_i^{(l,f)}(x) = \sigma(w^{l,f}\hat{z}_i^{(l-1)} + b^{(l,f)}) \qquad f = 1, 2, ..., F_l \qquad (۱۶.۲)$$

که می‌توان آن را به شکل ماتریسی زیر نوشت:

$$z_i^{(l)} \stackrel{def}{=} z_i^{(l)}(x) = \sigma(w^{l}\hat{z}_i^{(l-1)} + b^{(l)}) \qquad (۱۷.۲)$$

در دو رابطه ۱۶.۲ و ۱۷.۲ مقادیر مختلف در زیر بسط داده شده‌است:

- $z_i^{(l,f)}$ خروجی تابع نگاشت ویژگی نوع f را در مکان i از لایه l ایجاد می‌کند.
- $w^{(l,f)}$ پارامترهای تابع f در لایه l هستند که فرم ماتریسی آن عبارت است از: $w^{(l)} = [w^{(l,f)}, ..., w^{(l,F_l)}]$
- $\sigma(.)$ تابع فعال سازی استفاده شده در لایه l (توابع سیگموید و ReLu [۳۲])
- z_i^{l-1} بیانگر قسمتی از لایه $l-1$ که برای کانولوشن استفاده می‌شود. در حالت خاص $\hat{z_i^0}$ این تابع رشته‌های ورودی‌ها را به یکدیگر متصل می‌کند:

$$\hat{z}_i^{(0)} = x_{i:i+k_1-1} \stackrel{def}{=} [X_i^\top, X_{i+1}^\top, ..., X_{i+k_1-1}^\top]^\top \qquad (۱۸.۲)$$

همانطور که در تصویر ۱.۲ قابل مشاهده است به ازای هر دو واحد کانولوشن یک واحد Max pooling وجود دارد که به صورت زیر عمل می کند:

$$z_i^{(l,f)} = \max(z_{2i-1}^{(l-1,\,f)},\, z_{2i}^{(l-1,\,f)}) \qquad l = 2, 4, ... \qquad (۱۹.۲)$$

لایه‌های Max pooling دو اثر مهم در نتیجه ساختار و عملکرد نهایی شبکه عصبی دارند:

- در هر مرحله از pooling سایز بردار نصف می‌شود در نتیجه بعد از عبور از هر لایه، تاثیر تفاوت طول بردارها و درنتیجه تفاوت طول و تعداد کلمات به کار رفته در هر جمله کاهش می‌یابد.

- ترکیبات نامناسب کلمات را از جمله در هر مرحله حذف می‌کند.

در بیشتر کاربردها و پایگاه داده‌های مورد استفاده در روش‌های تشخیص شباهت جملات، طول جمله‌ها و تعداد کلمه‌های به کار رفته در هر جمله با دیگر جملات متفاوت است. به کمک استراتژی به کار رفته در شبکه‌های عصبی کانولوشنی تاثیر این تغییرات طول کاهش داده می‌شود. ورودی شبکه عصبی در هر مرحله یک بردار جمله است که از اتصال بردارهای کلمات تشکیل دهنده آن ایجاد شده‌است.

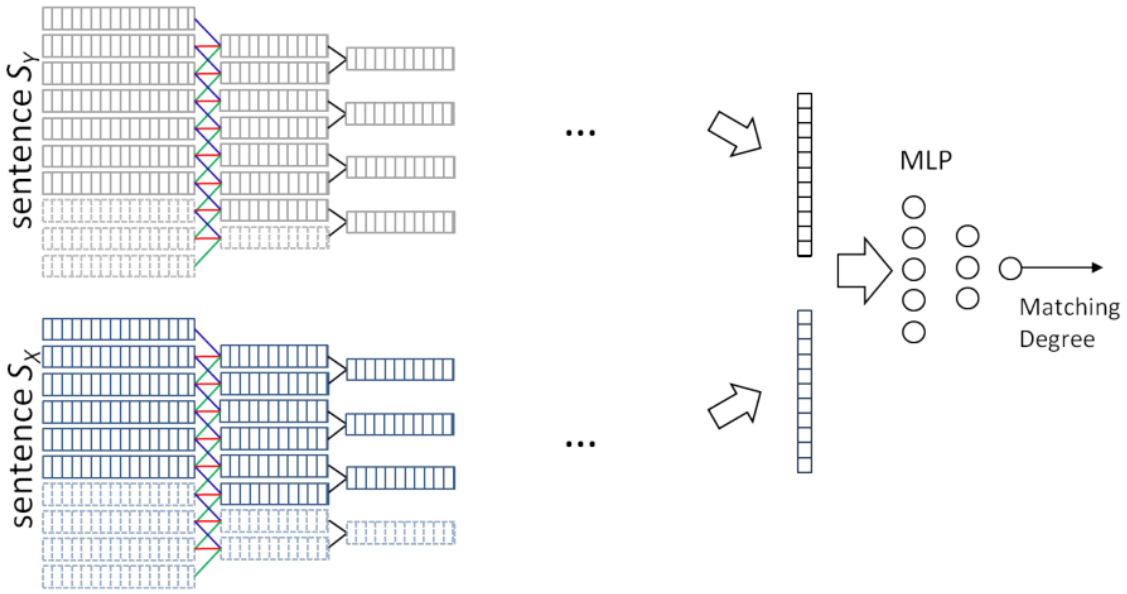
همانطور که گفته شد در بیشتر مواقع طول جمله‌ها با یکدیگر برابر نیست و از طرفی نیز ساختار شبکه عصبی به گونه‌ای است که نیاز دارد ورودی شبکه اندازه ثابت داشته باشد. برای رفع این مشکل ابتدا طول بزرگترین جمله در پایگاه داده را به عنوان اندازه ورودی شبکه در نظر گرفته و سپس برای سایر جمله‌ها که طول کمتری دارند از لایه‌گذاری [۴۵] با صفر استفاده می‌شود.

در ابتدا این‌گونه به نظر می‌رسد که این افزایش طول باعث از دست دادن اطلاعات و یا تغییر اطلاعات معنایی هر جمله می‌شود. برای رفع تاثیر صفرهای اضافه شده به هر جمله لایه‌های کانولوشنی به اینگونه تغییر می‌کنند که اگر ورودی لایه کاملا صفر بود آنگاه خروجی در آن واحد نیز باید کاملا صفر باشد. این تغییر در رابطه‌های لایه کانولوشنی به شکل زیر نشان داده می‌شود:

$$z_i^{(l,f)} \stackrel{def}{=} z_i^{(l,f)}(x) = g(\hat{z}_i^{(l-1)}).\sigma(w^{l,f}\hat{z}_i^{(l-1)} + b^{(l,f)}) \qquad f = 1, 2, ..., F_l \quad (۲۰.۲)$$

در این رابطه تابع $g(.)$ به این صورت تعریف می‌شود که اگر ورودی آن تمام صفر بود، خروجی نیز صفر باشد و در نتیجه خروجی لایه نیز صفر است، در غیر این صورت مقدار تابع $g(.)$ برابر یک است و مقدار متناسب با ورودی در لایه محاسبه می‌شود. این تغییرات به همراه

[45] Padding



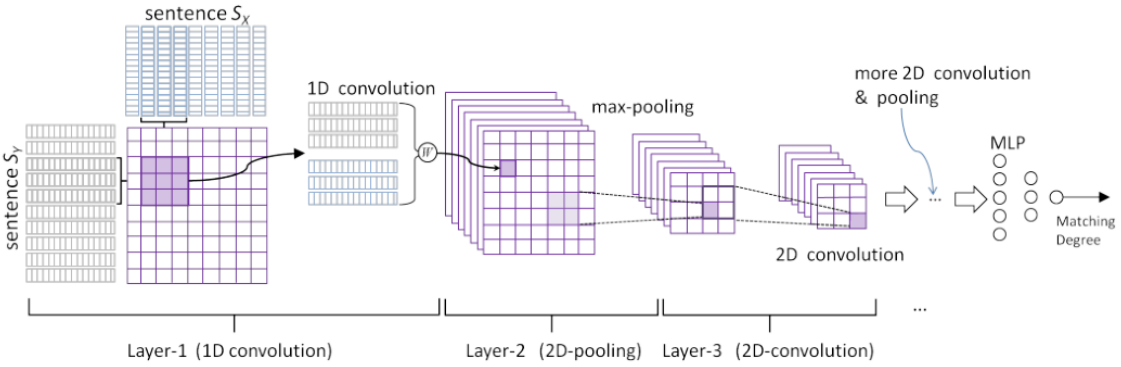
شکل ۲.۲: ساختار مدل اول که برای تشخیص شباهت بین جملات استفاده شده‌است.

لایه Max pooling و تابع مناسب فعال ساز در هر لایه، اثرات ناخواسته صفرهای حاشیه گذاری شده را حذف می‌کند.

در این قسمت ساختار کلی شبکه‌های عصبی کانولوشنی که برای تشخیص شباهت بین جملات استفاده می‌شود شرح داده شد. در ادامه دو مدل معرفی شده در این مقاله ارائه می‌شوند.

۲.۲.۵.۲ ARC-I

در این قسمت اولین مدل پیشنهادی را بررسی می کنیم. نام این مدل مخفف کلمه Architecture-I است. همانطور که در تصویر ۲.۲ قابل مشاهده است، این روش از دو شبکه عصبی تشکیل شده است. در ابتدا بوسیله یک شبکه عصبی کانولوشنی بردارهای ویژگی (بردار متناظر با هر جمله) استخراج شده و سپس از این بردارها برای آموزش شبکه عصبی چند لایه به منظور تشخیص شباهت بین جمله ها و یا پیش‌بینی یک جمله استفاده می‌شود. این روش برای اولین در [۳۳] و [۳۴] معرفی شده‌است و در زمینه متفاوتی استفاده می‌شود. این روش اگرچه از انعطاف پذیری شبکه عصبی کانولوشنی در استخراج برداهای جمله استفاده می‌کند اما مشکلی که در این شیوه وجود دارد این‌است در این روش اثرات متقابل دو جمله روی یکدیگر



شکل ۳.۲: معماری مدل ARC-II. در این تصویر کانولوشن‌های یک بعدی و دو بعدی و ساختار هر کدام مشخص است.

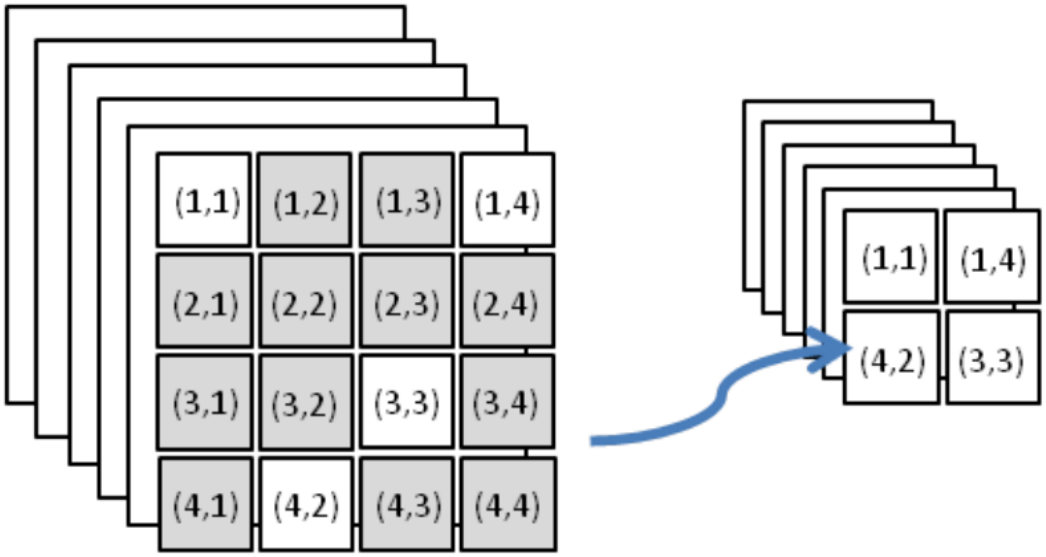
در شبکه عصبی چند لایه تا هنگامی که شیوه نمایش بردار آنها در شبکه کانولوشنی کامل بشود (نمایش برداری که اطلاعات متفاوت از هر جمله را نمایش می‌دهد) به تعویق می‌افتد درنتیجه این امر ممکن است در حین آموزش شبکه پیشرو (شبکه عصبی چند لایه) اطلاعات مهمی از هر جمله مانند نام شهرها یا افراد از بین برود.

Arch-II ۳.۲.۵.۲

با توجه به مشکلاتی که برای مدل قبلی وجود داشت، معماری Arch-II پیشنهاد شده است تا این مشکلات را برطرف کند. این مدل این امکان را به ساختار شبکه عصبی می‌دهد تا بردار دو جمله قبل از اینکه به شکل نهایی و کامل خود برسند در شبکه عصبی استفاده شده و با یکدیگر تعاول داشته باشند. در تصویر ۳.۲ ساختار کلی این مدل نشان داده شده‌است. برای این منظور در لایه اول شبکه عصبی کانولوشنی ابتدا بوسیله پنجره‌های لغزنده، قسمت هایی از هر جمله انتخاب شده و سپس تمام ترکیب‌های ممکن دو جمله به کمک یک لایه کانولوشون یک بعدی (1-D)ساخته می‌شود. در نتیجه برای قسمت i در جمله S_x و j در جمله S_y تابع نگاشت ویژگی به صورت زیر است:

$$z_{i,j}^{(l,f)} \stackrel{def}{=} z_{i,j}^{(l,f)}(x,y) = g(\hat{z}_{i,j}^{(0)})\sigma(w^{(l,f)}\hat{z}_{i,j}^{(0)} + b^{(l,f)}) \qquad (۲۱.۲)$$

در رابطه ۲۱.۲ $z_{i,j}^{0} \in \mathbf{R^{2k_1}}$ است. و تنها دوبردار ورودی که قسمتی از بردار دو جمله



شکل ۴.۲: ساختار pooling دو بعدی که ترتیب را نیز حفظ می‌کند

هستند را به یکدیگر متصل می کند:

$$\hat{z}_{i,j}^{(0)} = [x_{i:i+k_1-1}^{\top} + y_{j:j+k_1-1}^{\top}] \quad (۲۲.۲)$$

مشخص است که لایه یک بعدی استفاده شده، جایگاه و ترتیب اطلاعات هر قسمت و در نتیجه تمام جمله را حفظ می‌کند. بعد از لایه یک بعدی، یک لایه Max pooling ۲ بعدی روی داده ها اجرا می‌شود که مشابه ساختار کانولوشن برای داده‌های تصویری است [۳۰] . این لایه روی پنجره‌های 2×2 غیر هم‌پوشان عمل می‌کند و در تصویر ۴.۲ قابل مشاهده هستند. رابطه ۲۳.۲ این لایه را به صورت ریاضیاتی نشان می‌دهد:

$$z_{i,j}^{2,f} = max(z_{2i-1,2j-1}^{(2,f)}, z_{2i-1,2j}^{(2,f)}, z_{2i,2j-1}^{(2,f)}, z_{2i,2j}^{(2,f)}) \quad (۲۳.۲)$$

در لایه سوم از یک کانولوشن دو بعدی که روی پنجره‌هایی با ابعاد $k_3 \times k_3$ از خروجی لایه ۲ بعدی، استفاده می‌شود:

$$z_{i,j}^{(3,f)} = g(\hat{z}_{i,j}^{(2)})\sigma(w^{(3,f)}\hat{z}_{i,j}^{(2)} + b^{(3,f)}) \quad (۲۴.۲)$$

این ترکیب لایه‌ها می‌تواند با همین توالی ادامه داشته باشد.

بعد از کانولوشن دو بعدی، یک نمایش سطح پایین از دو جمله بدست می‌آید. به کمک این خروجی یک نمایش سطح بالا با نام $z_{i,j}^{(l)}$ بدست می‌آوریم که اطلاعات هر دو جمله را ذخیره می‌کند. کانولوشن دو بعدی به صورت زیر نمایش داده می‌شود:

$$z_{i,j}^{(l)} = g(\hat{z}_{i,j}^{(l-1)})\sigma(w^{(l)}\hat{z}_{i,j}^{(l-1)} + b^{(l,f)}) \qquad (۲۵.۲)$$

در این رابطه $\hat{z}_{i,j}^{(l-1)}$ تابعی است که ورودی‌ها را به یکدیگر متصل کرده می‌کند. عملکرد این لایه با لایه یک بعدی متفاوت است زیرا علاوه بر ترکیب‌های مختلف قسمت‌های جملات، براساس هم‌خوانی محلی [۴۶] نیز عمل می‌کند.این روش مشابه روشی است که در [۳۵] معرفی شده است با این تفاوت که براساس یک ساختار ثابت عمل می‌کند.

این مدل با سه روش مختلف آزمایش شده است که عبارت‌اند از:

۱. پیش‌بینی و تکمیل یک جمله

۲. پاسخ [۴۷] به یک توییت

۳. تشخیص شباهت دو جمله

در این پژوهش، مدل ارائه شده تنها برای تشخیص شباهت جملات استفاده می‌شود و کامل کردن یک جمله و ایجاد پاسخ برای توییت استفاده نمی‌شود، تنها نتایج مربوط به قسمت ۳ در به عنوان روش پایه برای مقایسه استفاده می‌شود که در جدول ۳.۲ نتایج حاصل از پیاده سازی و انجام آزمایش بوسیله پایگاه داده استاندارد MSRP آورده شده است. با توجه به اینکه نتایج مربوط به روش ARC-II بهتر هستند، از این مدل به عنوان روش پایه استفاده می‌کنیم.

۶.۲ نتیجه‌گیری

در این فصل ضمن تشریح روش‌ها و مفاهیم پایه در محاسبه شباهت بین سندها، دیدگاه‌های مختلف بررسی شباهت بین متون نیز مورد بررسی قرار گرفت و چند مورد نیز معرفی شد.

[46] Local matching
[47] Response

جدول ۳.۲: نتایج آزمایش‌ها بر روی پایگاده داده استاندارد MSRP

F1	Accuracy	
80.27	69.6	ARC-I
80.91	69.9	ARC-II

هم چنین روش‌های مبتنی فضای برداری که پایه این تحقیق است بررسی شد و چند مورد دیدگاه‌های بررسی شباهت بین بردارها که در این زمینه مورد استفاده قرار می‌گیرد نیز تشریح شد.

سپس در ادامه برخی از کارهای انجام شده در این زمینه به صورت خلاصه معرفی شدند و دو مورد از آن‌ها که به عنوان روش‌های پایه برای مقایسه انتخاب شده‌اند، شرح داده شدند.

فصل ۳

روش پیشنهادی

در فصل گذشته مفاهیم پایه و برخی از کارهای انجام شده در زمینه اندازه‌گیری شباهت متون مرور شدند. در این فصل ابتدا به پایگاه داده و بردارهای جایگذاری استفاده شده در این تحقیق معرفی می‌شود و سپس روش پیشنهادی و اجزای مختلف آن بسط داده می‌شود.

۱.۳ معرفی پایگاه داده

همانطور که در فصل اول گفته شده در این تحقیق از پایگاه داده MSRP که یک پایگاه داده استاندارد در زمینه اندازه‌گیری شباهت متون است استفاده می‌شود [۳۶]. این پایگاه داده در تحقیقات زیادی مورد استفاده قرار گرفته و برای مقایسه روش‌های مختلف در این زمینه در نظر گرفته می‌شود([۳۷] و [۳۸]).

این پایگاه داده شامل ۵۸۰۰ جفت جمله است. این پایگاه داده در سال ۲۰۰۵ توسط شرکت مایکروسافت ساخته شد. برای ایجاد این پایگاه داده منابع خبری در سطح وب استفاده شده و تضمین شده است که بیشتر از یک جمله از هر مقاله خبری در ساخت آن استفاده نشده‌است.

هر جفت جمله توسط حداکثر سه داور از نظر معنایی مورد بررسی قرار گرفته شده است. ابتدا هر جفت جمله به دو داور داده شده تا آن‌ها با شباهت (۱) و یا عدم شباهت (۰) دو جمله را تعیین کنند. اگر نظر دو داور یکی بود، نتیجه به عنوان برچسب دو جمله انتخاب می‌شود، در غیر این صورت از داور سوم برای نتیجه گیری نهایی استفاده شده است. برای این نشانه گذاری از شرکت‌ها و افراد مستقل از حوزه هوش مصنوعی و پردازش متن استفاده شده است.

بعد از بررسی داوران ۳۹۰۰ مورد از دادگان پایگاه داده (در حدود ۶۷٪) از تمام ۵۸۰۰ داده به عنوان مشابه معنایی [1] تشخیص داده شدند.

برای راحتی در پیاده‌سازی به هرکدام از جملات یک ID اختصاصی داده شده. از ۱۶۵۰ نمونه به عنوان داده تست و ۴۱۵۰ داده باقی مانده به عنوان داده آموزش در نظر گرفته می‌شود. در جدول ۱.۳ چند نمونه از هر دو گروه آورده شده است.

[1] Semantically equivalent

جدول ۱.۳: نمونه داده‌های موجود در پایگاه داده

Quality	#1 ID	#2 ID	#1 String	#2 String
1	1089874	1089925	PCCW's chief operating officer, Mike Butcher, and Alex Arena, the chief financial officer, will report directly to Mr So.	Current Chief Operating Officer Mike Butcher and Group Chief Financial Officer Alex Arena will report to So.
1	3019446	3019327	The world's two largest automakers said their U.S. sales declined more than predicted last month as a late summer sales frenzy caused more of an industry backlash than expected.	Domestic sales at both GM and No. 2 Ford Motor Co. declined more than predicted as a late summer sales frenzy prompted a larger-than-expected industry backlash.
0	1430402	1430329	A tropical storm rapidly developed in the Gulf of Mexico Sunday and was expected to hit somewhere along the Texas or Louisiana coasts by Monday night	A tropical storm rapidly developed in the Gulf of Mexico on Sunday and could have hurricane-force winds when it hits land somewhere along the Louisiana coast Monday night.

۲.۳ بردارهای جایگذاری کلمات

در سال‌های اخیر روش‌های مختلفی برای ایجاد بردارهای جایگذاری معرفی شده‌اند که هر کدام ویژگی و عملکرد خاص خود را دارند. در بین این بردارها دو مورد بیشتر از سایر روش‌های ایجاد بردار جایگذاری مورد استفاده محققان قرار می‌گیرد. یکی از این دو مورد مدل Word2Vec است که در [۳۹] معرفی شده است. این بردار بوسیله یک شبکه عصبی و با دو روش CBOW[۲] و یا Skip gram تولید می‌شود. روش دیگر GloVe[۳][۴۰] است که در سال ۲۰۱۴ معرفی شده است. روش ایجاد بردارها در این مدل مشابه روش LSA است. در ادامه هر یک از این بردارها را معرفی می‌کنیم.

۱.۲.۳ Word۲Vec

الگوریتم Word2Vec یک گروه از مدل‌های مرتبط با پردازش متن است که برای تولید کلمه جاسازی استفاده می‌شود. این مدل‌ها شبکه‌های عصبی هستند که برای آموزش بازسازی مفاهیم زبانی کلمات به کار می‌روند. الگوریتم Word2Vec به عنوان ورودی یک قسمت بزرگ متن را می‌گیرد و یک فضای برداری (به عنوان مثال فضای برداری با صد بعد مختلف) با کلمه منحصربه‌فرد در پیکره متنی که به یک بردار متناظر در فضا اختصاص داده می‌شود، تولید می‌کند.

بردارهای کلمه در فضای بردار قرار می‌گیرند به طوری که کلماتی که زمینه‌های متفاوتی را در زبان‌شناسی پیکره‌ای به صورت مشترک دارند در نزدیکی یکدیگر در فضا قرار دارند. الگوریتم Word2Vec توسط تیمی از محققان به رهبری توماس میکولوف در گوگل ایجاد شد. بردارهایی که با استفاده از الگوریتم Word2Vec ایجاد شده‌اند مزایای زیادی در مقایسه با الگوریتم‌های قبلی مانند آنالیز پنهان مفهومی دارند.

این بردارها به دو روش CBOW (کیسه کلمات) و یا Skip gram ایجاد می‌شوند. در هر دو روش از یک شبکه عصبی ایجاد می‌شود که تنها در اندازه ورودی و خروجی شبکه با یکدیگر تفاوت دارند.

[2] Continuous bag of words
[3] Globa Vector representation of words

۱.۱.۲.۳ کیسه کلمات

در روش کیسه کلمات ابتدا به ازای هر لغت در پیکره یک بردار با طول مشخص و با اعداد تصادفی (بین صفر و یک) تولید می شود. سپس در تمام سندها یک به یک بررسی شده و برای هر کلمه در هر سند به تعداد مشخص کلمه قبل و بعد از آن انتخاب می‌شود (به این تعداد طول پنجره می‌گوییم). در مرحله بعد بردار ایجاد شده برای کلماتی که در دو پنجره قبل و بعد کلمه مورد نظر هستند را به عنوان ورودی در نظر میگیریم و در خروجی کلمه میان دو پنجره را به عنوان مقدار هدف در نظر می‌گیریم. سپس شبکه عصبی که یک شبکه پیشخور [۴] است بوسیله این بردارها آموزش داده می‌شود. بعد از آموزش شبکه عصبی وزن‌های لایه میانی به عنوان بردار کلمات استفاده می‌شود. می‌توان این مدل را به عنوان مدل پیش‌بینی کننده کلمه در نظر گرفت زیرا در ورودی شبکه عصبی از زمینه [۵] استفاده می‌شود تا کلمه در خروجی شبکه ایجاد شود. در شکل ۱.۳ شبکه عصبی در این مدل نمایش داده است.

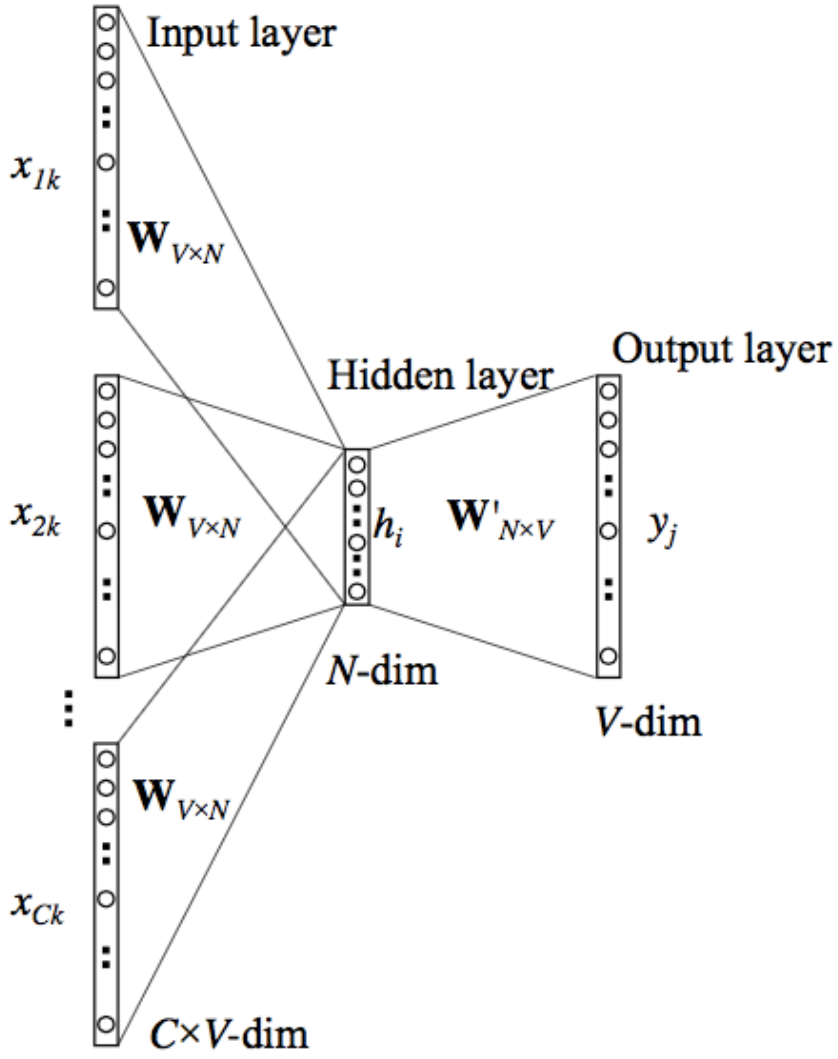
۲.۱.۲.۳ Skip gram

در روش Skip gram کلمات به صورت One hot encoding نشان داده می‌شوند. سپس هر کلمه در متن به عنوان ورودی شبکه عصبی در نظر گرفته می‌شود و کلمات پنجره‌های قبل و بعد از آن در متن به عنوان هدف در شیکه عصبی در نظر گرفته می‌شوند. در این روش نیز از بردارهای لایه میانی به عنوان بردارهای نهایی کلمات استفاده می‌شود. این روش برعکس مدل کیسه کلمات تلاش می‌کند تا به کمک یک کلمه زمینه آن کلمه را پیش بینی کند. ساختار این شبکه نیز در تصویر ۲.۳ نشان داده شده است. در هر دو مورد از وزن‌های لایه ورودی به لایه میانی (W) به عنوان بردارهای جایگذاری استفاده شده است.

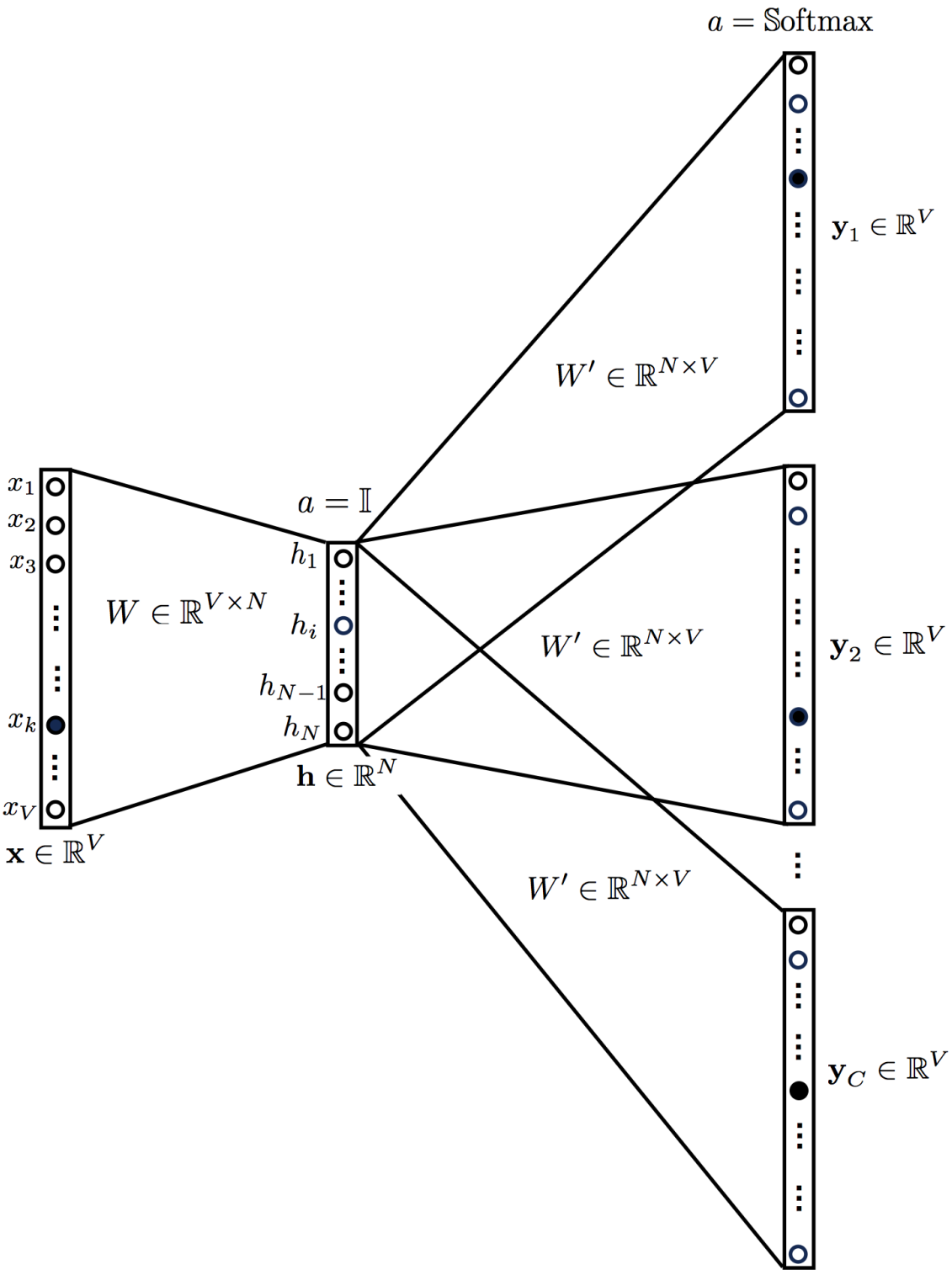
۲.۲.۳ GloVe

GloVe یکی دیگر از مدل‌های بردار جایگذاری است که در سال ۲۰۱۴ توسط آزمایشگاه پردازش متن دانشگاه استنفورد معرفی شده است. در بعضی از مقالات از این روش به عنوان نسخه

[4] Feedforward neural network
[5] Context



شکل ۱.۳: در این تصویر طول پنجره $C/2$ است و تعداد کل کلمات در پیکره برابر V است. تعداد بعدهای بردار کلمات نیز برابر N در نظر گرفته شده است. [۳۹]



شکل ۲.۳: در این تصویر طول پنجره $C/2$ است و تعداد کل کلمات در پیکره برابر V است. تعداد بعدهای بردار کلمات نیز برابر N در نظر گرفته شده است. [۳۹]

جدول ۲.۳: مثالی از ماتریس هم‌رخدادی در روش GloVe

	The	cat	sat	on	mat
The	2	1	2	1	1
cat	1	1	1	1	0
sat	2	1	1	1	0
on	1	1	1	1	1
mat	1	0	0	1	1

دیگری از Word2Vec نام برده می‌شود در صورتی که روش تولید این بردارها کاملا با روش ایجاد Word2Vec متفاوت است. سازندگان این مدل دو ویژگی برای این مدل معرفی کرده اند:

۱. ایجاد بردارهایی که معنای کلمات را در فضای برداری ذخیره کند

۲. استفاده از آمارهای شمارش کلی [۶] به‌جای استفاده از اطلاعات محلی

این روش برخلاف مدل Word2Vec که به صورت جاری شدن [۷] جمله‌ها ایجاد می‌شود، از یک ماتریس هم رخدادی برپایه شمارش استفاده می‌کند که در آن کلمات براساس فاصله وزن دهی می‌شوند. این روش برخلاف Word2Vec ویژگی‌های نحوی را نیز تا حدی در فضای برداری ذخیره می‌کند.

همانطور که گفته شد این روش از یک ماتریس هم رخدادی استفاده می‌کند. این ماتریس با شمارش کلمات ایجاد می‌شود به این صورت که از زمینه محلی [۸] استفاده می‌کند به این شکل که اگر کلمات در در یک بازه مشخص (به نام پنجره) وجود داشتند آنگاه با یکدیگر هم‌رخداد هستند. . به عنوان مثال ماتریس هم‌رخدادی در این روش برای جمله زیر در جدول ۲.۳ نمایش داده شده است:

The cat sat on the mat در این مثال طول پنجره ۲ در نظر گرفته شده‌است. همانطور که در قابل مشاهده است، این ماتریس متقارن است که دلیل آن این است هنگامی که یک کلمه (به عنوان مثال cat) در پنجره کلمه دیگری قرار بگیرد (مانند The) ، برعکس این اتفاق

[6] Global count statistics
[7] Streaming sentences
[8] Local context

جدول ۳.۳: نحوه محاسبه نسبت هم رخدادی در روش GloVe

Probability / Ratio	k = solid	k = gas	k = water	k = fashion
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

نیز می‌افتد. حال با داشتن این جدول می‌توان گفت نسبت [۹] هم‌رخدادی دو کلمه، میزان هم‌معنی بودن آن‌ها را بیان می‌کند. برای توضیح روش بدست آوردن این نسبت از مثالی که پدیدآوردگان این روش در مقاله خود آورده‌اند، استفاده می‌کنیم.

دو کلمه Ice و Steam را در نظر بگیرید. این دو از نظر حالت ماده، دو حالت مختلف دارند در حالی که هردو شکل‌های متفاوتی از آب هستند. در نتیجه می‌توان انتظار داشت کلماتی که با آب (Water) مرتبط هستند در زمینه این دو کلمه وجود داشته باشند (Water و Wet). در حالی که کلماتی مانند Solid و Cold بیشتر در زمینه Ice هستند. در نتیجه با توجه شمارش این کلمات در پیکره مورد استفاده برای آموزش بردارها، می‌توان مقادیر جدول ۳.۳ را به عنوان احتمال هم‌رخدادی این کلمات و نسبت دو کلمه با یکدیگر در نظر گرفت. این احتمالات نشان می‌دهند چه زمانی کلمه k در یک پنجره کنار دو کلمه Ice و Steam آمده‌اند.

به عنوان مثال کلمه Solid را در نظر بگیرید. در سطر آخر جدول مقدار بدست آمده برای این کلمه 8.9 است که به معنای این است که Solid ارتباط معنایی بیشتری با Ice دارد. به عنوان یک مثال دیگر، کلمه Water در سطر آخر مقدار 1.36 دارد که معنای ان این‌است با هر دو کلمه Ice و Steam تقریبا به یک اندازه ارتباط معنایی دارد.

۱.۲.۲.۳ زیرساختارهای خطی

معیارهای شباهتی که برای بررسی شباهت دو بردار استفاده یک عدد تولید می‌کنند که به معنای میزان شباهت دو بردار (در اینجا دو کلمه) است. از آن جایی که کلمات معنی‌های متفاوتی دارند، این سادگی ممکن است باعث مشکلاتی شود. به عنوان مثال کلمه man

[9] Ratio

می‌تواند با کلمه woman در ارتباط باشد زیرا هر دو مفهوم human را بیان می‌کنند (هر دو نوعی از human هستند). از طرفی دیگر هر دو کلمه با مفهوم human نیز در ارتباط هستند. از طرف دیگر این دو کلمه متضاد یکدیگر نیز در نظر گرفته می‌شوند زیرا دو دو گونه متفاوت از human را نشان می‌دهند که با یکدیگر متفاوت هستند.

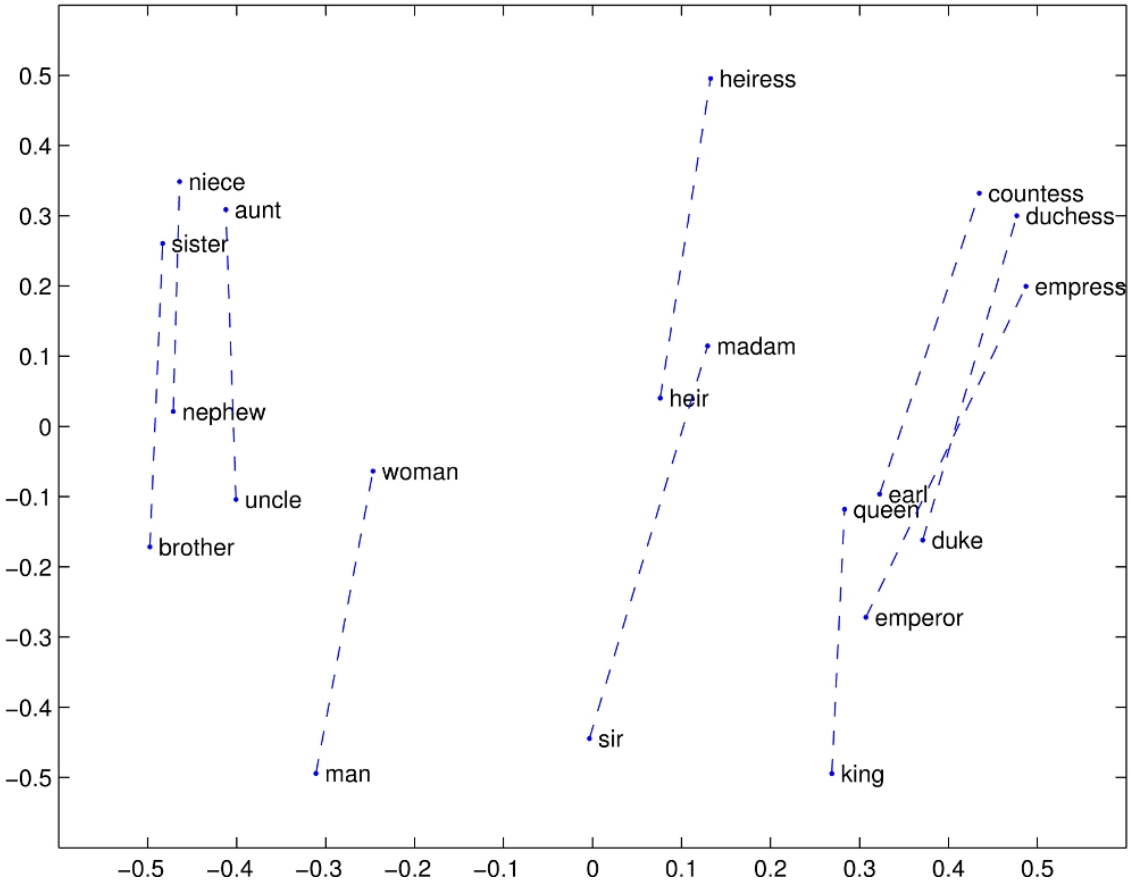
برای بدست آوردن اختلاف جزئی بین دو کلمه man و woman، یک مدل باید بیشتر از یک مقدار عددی برای هر کدام از این کلمات اختصاص بدهد در نتیجه می‌توان کلمات را در فضای برداری به نحوی نشان داد که بتوانند مفاهیم مختلف را بیان کنند. یک روش ساده برای بیان تمایزهای بین دو کلمه که به صورت برداری نشان داده می‌شوند، استفاده از تفاوت برداری آن‌ها است. مدل GloVe به نحوی ایجاد شده‌است که می‌تواند این تفاوت‌ها را با عملیات ریاضی نشان دهد. در ادامه چند نمونه از جداسازی‌هایی که این مدل می‌تواند انجام دهد آورده شده‌است.

در شکل ۳.۳ می‌توان مشاهده کرد که مفهومی که دو کلمه man و woman را از یکدیگر متمایز میکند(جنسیت Gender) توسط سایر کلماتی که این مفهوم را بیان می‌کنند نیز قابل بیان است (نحوه قرار گیری کلمات و طول بردارهای خط چین شده). می‌توان انتظار داشت این خاصیت برای سایر جفت کلماتی که این مفهوم را بیان می‌کنند نیز وجود داشته باشد و طول بردار حاصل از تفریق دو کلمه تقریبا برابر و هم‌جهت بردارهای نشان دهنده جنسیت در این تصویر باشند.

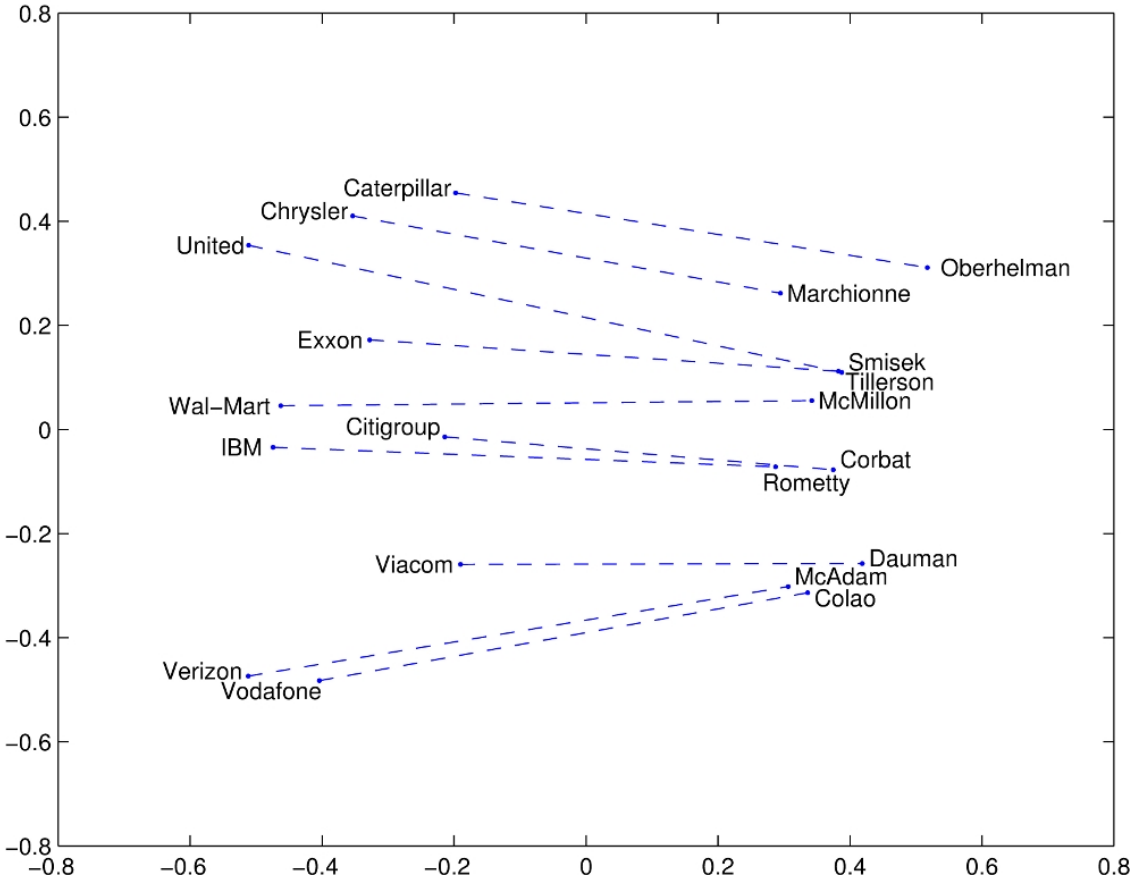
در تصویر ۴.۳ مفهوم مالکیت توسط بردارهای مختلف قابل مشاهده‌است. در این تصویر اسامی شرکت‌ها و محصولات مختلف در چپ و اسامی صاحبان آن‌ها در طرف دیگر قرار دارد.

در تصویر ۵.۳ رابطه مختلف صفت‌ها نشان داده شده‌است. یکی از ویژگی‌های GloVe توانایی نمایش اینگونه روابط است. در این تصویر سه حالت مختلف از چند صفت نشان داده شده‌است. در سمت چپ تصویر شکل عادی صفت آورده شده است (slow)، در قسمت وسط شکل صفت برتری [10] آورده شده است (slower) و در سمت راست شکل عالی [11] صفت آورده شده‌است(slowest).

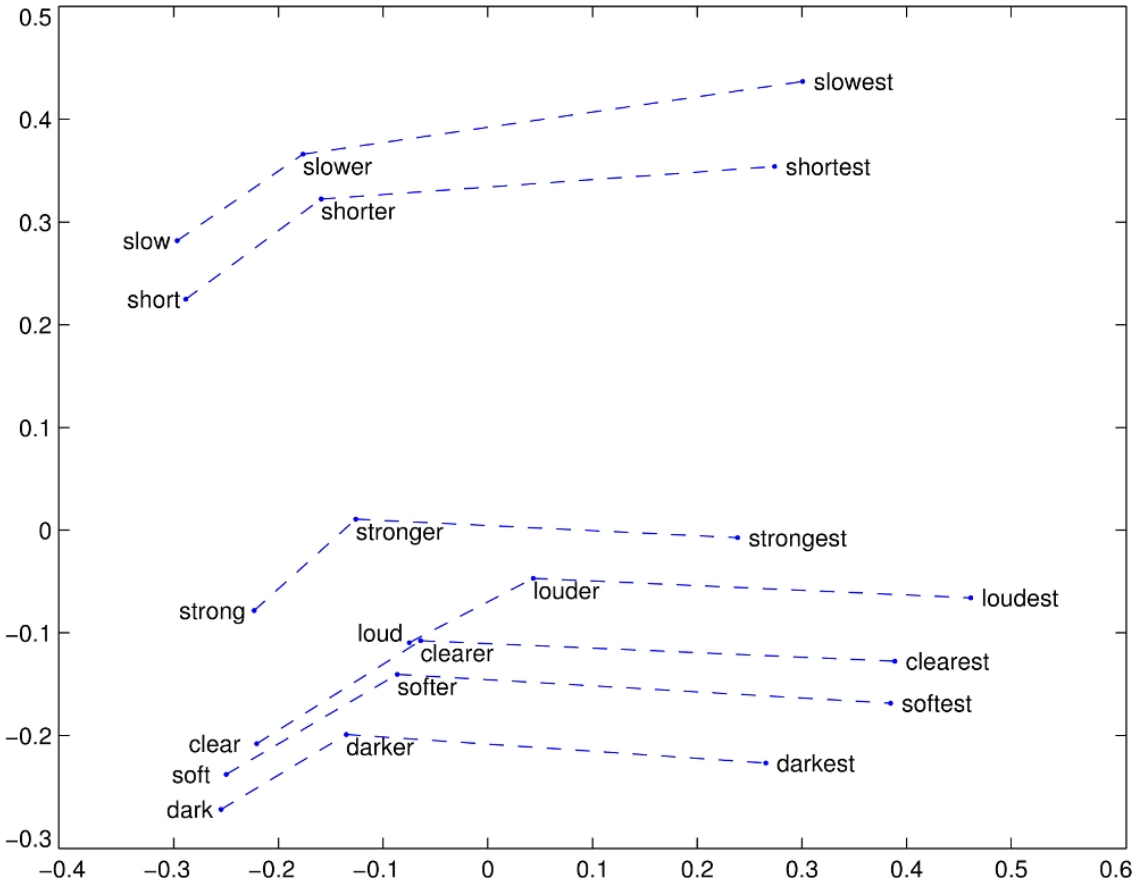
[10] Comparative
[11] Superlative



شکل ۳.۳: در این تصویر کلماتی که مفهوم جنسیت را بیان می‌کنند نشان داده شده‌است



شکل ۴.۳: در این تصویر کلماتی که مفهوم مالکیت را بیان می‌کنند نشان داده شده‌است



شکل ۵.۳: در این تصویر کلماتی حالت‌های مختلف صفت‌ها نشان داده شده‌است

۳.۳ پیش پردازش

در مرحله پیش پردازش داده‌ها ابتدا تمام جملات به کمک کتابخانه NLTK تبدیل به مجموعه‌ای از توکن‌های می‌شوند. این کتابخانه دارای لیستی از Stop word ها است که توسط یک مدل از پیش آموزش داده شده می‌تواند کلمات توقف را تشخیص دهد، بنابراین کلمات توقف موجود در متن نیز به کمک این کتابخانه حذف می‌شوند.

سپس هر توکن در تمام مجموعه‌های جایگذاری کلمات مورد استفاده، جست‌وجو می‌شوند. در صورتی که توکن پیدا نشود این عملیات را با حذف علائم نگارشی استفاده شده مانند ”.“، در توکن تکرار میکنیم.

بعد از بدست آوردن مجموعه توکن‌های موجود در لیست‌های کلمات جایگذاری و با توجه مشخص شدن هر جمله به کمک یک عدد یکتا، جملات را در پیاده سازی‌ها به شکل زوج‌های کلید_مقدار نشان می‌دهیم که کلید هر جمله شناسه یکتا آن جمله و مقدار آن نیز مجموعه توکن‌های بدست آمده است.

۴.۳ سیستم پیشنهادی

در این روش، ما قصد داریم تا حد امکان از کمترین منابع دانش خارجی مانند شبکه‌هایی معنایی، درخت‌ها تجزیه و منابع دانش ساخت یافته استفاده کنیم. هدف ما ایجاد یک مدل عمومی است، که نیازمند دانش پیشین از زبان طبیعی (مانند درختان تجزیه) و منابع خارجی اطلاعات معنایی ساختاری نیست.

پیشرفت‌های اخیر در معانی توزیع‌شده، به‌ویژه روش‌های مبتنی بر شبکه عصبی مانند Word2Vec نیاز به مقدار زیادی از داده‌های متنی بدون برچسب دارند. این داده‌ها برای ایجاد یک فضای معنایی استفاده می‌شوند. واژه‌ها در این فضای معنایی به شکل بردارهایی نمایش داده می‌شوند که به آن‌ها کلمه جاگذاری شده می‌گویند. در اثبات‌شده است که خواص هندسی این فضا به لحاظ معناشناختی و نحوی معنی‌دار است، یعنی کلماتی که ازلحاظ معنایی یا نحوی مشابه هستند در این فضا نزدیک می‌شوند.

یک چالش استفاده از کلمات جاگذاری شده در تعیین شباهت معنایی متون کوتاه، تغییر

سطح معانی از سطح کلمه به متن کوتاه است. این مشکل در طول چند سال گذشته موردبررسی قرارگرفته است. در این تحقیق ما از معانی در سطح کلمه بالاتر رفته و شباهت معنایی را در به کمک بردارهای جاگذاری شده در سطح متن بررسی می‌کنیم. یکی از ویژگی‌های این تحقیق این است که تعداد دلخواه مجموعه‌های کلمات جاگذاری شده می‌توانند بدون توجه به جسم [۱۲] مورداستفاده برای آموزش، الگوریتم آموزش، پارامترهای آن یا ابعاد بردارهای کلمه استفاده شوند.

یک ویژگی دیگر این روش این است که به منابع دانش خارجی مانند شبکه‌های معنایی، تجزیه‌گرهای لغوی [۱۳] ، منابع ساخت‌یافته دانش مانند ویکی‌پدیا [۱۴] و موارد مشابه که معمولا تنها برای یک یا چند زبان خاص توسعه داده شده‌اند، نیاز ندارد.

برای محاسبه شباهت معنایی بین دو متن ما از یک روش یادگیری ماشین با ناظر استفاده می‌کنیم. الگوریتم ۱ شبه کد مرحله آموزش را نشان می‌دهد. ورودی این الگوریتم جفت جمله‌ها و برچسب هر کدام است که بیانگر شباهت یا عدم شباهت هر جفت جمله است. مجموعه کلمات جاگذاری متفاوتی که با استفاده متن‌های مختلف و با الگوریتم‌های متفاوت آموزش داده‌شده‌اند می‌توانند استفاده شوند. چندین تابع استخراج ویژگی به کمک بردارهای جاگذاری شده ویژگی‌های مختلفی از جملات استخراج می‌کنند. این توابع در این ادامه توضیح داده می‌شود. در هنگام آموزش ما برای تمام جفت جملات (خط ۲) و تمام مجموعه کلمات جاگذاری شده(خط ۴) به کمک توابع استخراج ویژگی، چندین ویژگی استخراج کرده و یک بردار ویژگی می‌سازیم(خط ۶).

در ادامه چهار تابع استخراج ویژگی که هر کدام مزایای خاص خود را دارد، را شرح می‌دهیم.

۱.۴.۳ توابع استخراج ویژگی

یکی از مزایای این روش این است که می‌توان توابع مختلف استخراج ویژگی که در سایر زمینه‌ها استفاده شده و نتایج بسیار خوبی نیز بدست آورده با مجموعه‌های مختلف کلمات جایگذاری شده استفاده کرد. در ادامه چهار تابع استفاده شده در آزمایشات شرح داده می‌شود.

[12] Corpus
[13] Syntax parser
[14] Wikipedia

input : List of sentence pairs $((s_{1,1}, s_{1,2}), (s_{2,1}, s_{2,2}), ..., (s_{n,1}, s_{n,2}))$

input : List of associated labels L = $[l_1, l_2, ..., l_n]$

input : Sets of word embeddings $[WE_1, WE_2, ..., WE_m]$

input : Multiple feature extractors $[fe_1, fe_2, \ldots, fe_l]$

output: A trained prediction model M

F= empty feature matrix;
for $i \leftarrow 1$ ***to*** n **do**
    $\vec{f} = <>$;
    for $j \leftarrow 1$ ***to*** m **do**
        for $k \leftarrow 1$ ***to*** l **do**
            $\vec{f} = \text{concat}(\ \vec{f},\ fe_k((s_{i,1},\ s_{i,2}),\ WE_j)\ )$;
        end
    end
    $F[i] = \vec{f}$;
end
M = *trainModel(F, L)*

Algorithm 1: الگوریتم روش ارائه شده

۱.۱.۴.۳ SWSN[15]

ما می خواهیم راهی برای در نظر گرفتن توزیع اصطلاحات در یک متن کوتاه در فضای معنایی نسبت به توزیع اصطلاحات در متن دیگر داشته باشیم. البته همه اصطلاحات به یک اندازه مهم نیستند. کلماتی مانند ضمیرها و یا حروف ربط که فراوانی بیشتری دارند مانند کلماتی که فراوانی کمتری دارند تاثیرگذاری زیادی در معنا ندارند. عکس فراوانی سند [16] معمولا با فراوانی اصطلاح [17] ترکیب می‌شود. تابعی که برای محاسبه شباهت متن استفاده می‌کنیم به صورت زیر است:

$$f_{sts}(S_l, S_s) = \sum_{w \in S_l} IDF(w) \frac{sem(w, S_s)(k_1 + 1)}{sem(w, S_s) + k_1(1 - b + b\frac{|S_s|}{avglength})} \qquad (۱.۳)$$

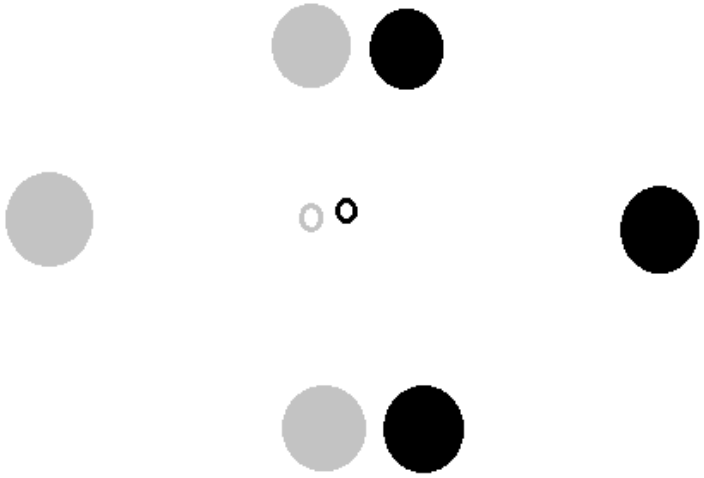
در ۱.۳ S_l جمله طولانی‌تر و S_s جمله با طول کمتر است. $avglength$ میانگین طول جملات در بین تمام جملات مجموعه آموزش است. شباهت کلمه w نسبت به جمله s از طریق معادله ۲.۳ بدست می‌آید:

$$sem(w, s) = \max_{w' \in s} f_{sem}(w, w') \qquad (۲.۳)$$

تابع f_{sem} شباهت بین دو کلمه را محاسبه می‌کند. همانطور که در رابطه ۱.۳ مشخص است، همیشه از جمله بزرگتر به عنوان منبع [18] برای محاسبه شباهت استفاده می‌کنیم. این کار دو دلیل دارد. اول اینکه با این روش نتیجه حاصل متقارن است. دلیل دوم این است که نمی‌خواهیم که کلمات نادیده گرفته بشوند. به عنوان مثال فرض کنید دو جمله داریم به طوریکه جمله کوچکتر از بعضی از کلمات جمله بزرگتر تشکیل شده‌است. اگر از جمله کوچکتر به عنوان منبع برای محاسبه شباهت استفاده کنیم نتیجه حاصل شباهت بسیار بالا خواهد بود. حال اگر جمله بزرگتر را به عنوان منبع در نظر بگیریم، ناسازگاری [19] بین دو جمله اثر خود را در نتیجه نهایی گذاشته و شباهت به مقدار اصلی خود نزدیکتر می‌شود.

از طرفی می‌توان رابطه ۱.۳ را اینگونه تفسیر کرد که قابلیت محاسبه شباهت غیر لفظی را نیز دارا است. در واقع از آنجایی که نتیجه شباهت به کمک IDF وزن‌دهی می‌شود، کلمات در

[15] Saliency-weighted semantic network
[16] Inverse Document Frequency
[17] Term Frequency
[18] Reference
[19] Incongruity



شکل ۶.۳: فضای نمونه از نمایش بردارهای جایگذاری کلمه در فضای دو بعدی برای دو جمله که هر کدام ۳ کلمه دارد. دایره‌های بزرگ مربوط به کلمات و دو دایره کوچک مربوط میانگین بردار جایگذاری کلمات هر جمله است.

صورتی نتیجه قابل توجهی در شباهت دارند که، مقدار شباهت آنها زیاد باشد، در واقع تمام کلمات در شباهت نهایی تاثیر دارند اما کلمات متشابه تاثیر خیلی بیشتری دارند.

یک توضیح دیگر درباره رابطه ۱.۳ این است که کلمات برجسته را در یک شبکه معنایی وزن‌دهی می‌کند. به عنوان مثال در تصویر ۶.۳ اگر هر کلمه را راس گراف در نظر بگیریم، عملگر max در ۲.۳ یال بین راس‌ها را رسم کرده و سپس این یال به کمک ضریب IDF وزن‌دهی می‌شود. در نتیجه عدم تطابق بین دو کلمه مانند دو دایره در سمت چپ و راست تصویر ۶.۳ در صورتی که مقدار IDF باشد تاثیر خیلی کمی‌دارد و در صورتی که مقدار IDF بیشتر باشد این تاثیر بیشتر می‌شود. از آنجایی که ما کلمات را به صورت بردار نشان می‌دهیم یک انتخاب برای این تابع می‌تواند فاصله کسینوسی باشد. آزمایشات فصل بعدی نشان داده‌است که تعداد خروجی مناسب برای این تابع ۳ است که می‌توان این خروجی‌ها را به صورت مقادیر کیفی کاملا مشابه، مشابه و متفاوت در نظر گرفت.

۲.۱.۴.۳ USN[۲۰]

برای انتقال اطلاعات به اندازه کافی به طبقه‌بند نهایی، ما همچنین یک شبکه معنایی بدون وزن نیز ایجاد می‌کنیم. برای یک جفت متن کوتاه (s_1، s_2) ، شباهت کسینوسی در فضای معنایی بین همه اصطلاحات در متن کوتاه s_1 و تمام اصطلاحات در s_2 محاسبه می شود. این کار به ما یک ماتریس شباهت بین اصطلاحات در s_1 و s_2 می دهد. از این ماتریس، دو مجموعه از ویژگی ها را محاسبه می کنیم.

در ابتدا شباهت‌ها را به سه دسته متفاوت، نسبتا شبیه و کاملا شبیه تقسیم می‌کنیم [۲۱] . بازه هرکدام از این دسته‌ها با توجه به داده‌های آموزش مشخص می‌شود. دوم، حداکثر شباهت برای هر کلمه محاسبه می‌شود و سطل‌ها [۲۲] از این حداکثر مقادیر ساخته می‌شوند.

تفاوت این تابع با تابع قبلی در این است که در قسمت قبل به صورت ضمنی شباهت بین کلمات به صورت وزن‌دار محاسبه می‌شد و برای کلماتی که ارتباط معنایی بیشتری با یکدیگر داشتند (به عنوان مثال کلماتی که یک رنگ را نشان می‌دهند مانند آبی، قرمز و ...) وزن بیشتری در نظر گرفته می‌شود. این تابع دو دسته خروجی دارد. دسته اول سه مقدار میان بیشترین شباهت‌ها در ماتریس شباهت بدست آمده‌است و دسته دوم سه مقدار از بین تمام شباهت‌های موجود در ماتریس شباهت‌ها که باعث می‌شود خروجی این تابع ۶ ویژگی باشد.

۳.۱.۴.۳ DBVM[۲۳]

همان طور که قبلا ذکر شد، یک روش استاندارد برای استخراج معنا متن‌های بزرگ، محاسبه میانگین بردار کلمات متن است. با این کار برای هر جمله یک بردار بدست می‌آوریم.

برای محاسبه ابتدا میانگین بردار جایگذاری تمام کلمات هر جمله را محاسبه میکنیم (تمام بردارها با هم جمع می‌شوند و سپس بر تعداد کلمات تقسیم شده) و شباهت دو بردار حاصل را اندازه‌گیری می‌کنیم. در این تحقیق فاصله بین دو بردار جمله که به روش بالا ایجاد می‌شود با دو روش فاصله کسینوسی و فاصله اقلیدسی محاسبه می‌شود در نتیجه این تابع دو ویژگی ایجاد می‌کند.

[20] Unweighted semantic network
[21] binning
[22] bins
[23] Distance between vectors means

۴.۱.۴.۳ BoD [۲۴]

شباهت کسینوسی بین دو بردار می‌تواند به عنوان یک تجمع بر روی تفاوت در هر بعد تفسیر شود. به این ترتیب، تمام اطلاعات مربوط به شباهت‌ها یا تفاوت بین دو بردار ضبط نمی‌شود [۲۵] . به عنوان مثال، مقایسه شباهت کسینوسی بین دو بردار که در بسیاری از ابعاد بسیار شبیه هستند و در تعداد کمی متفاوت هستند، می تواند نتایج مشابهی با در نظر گرفتن شباهت کسینوسی بین دو بردار که در هر بعد کمی متفاوت هستند، منجر شود. برای مثال تصویر ۷.۳ را در نظر بگیرید.

برای رفع این مشکل بردارهای میانگین s_1 و s_2 در چهار دسته که دو دسته اول بیانگر مشابه و دو دسته دوم بیانگر مشابهت بسیار بالا است، دسته بندی می‌شود.

دو تابع اول استخراج ویژگی را می‌توان به عنوان توابع استخراج ویژگی سطح کلمه در نظر گرفت زیرا محاسبه شباهت را بین دو کلمه (توکن) انجام می‌دهند و از طرف دیگر دو تابع دیگر را به عنوان توابع استخراج کننده ویژگی‌های سطح متن در نظر گرفت زیرا این کار را براساس انجام عملیات مختلف روی بردارهای میانگین هر جمله انجام می‌دهند.

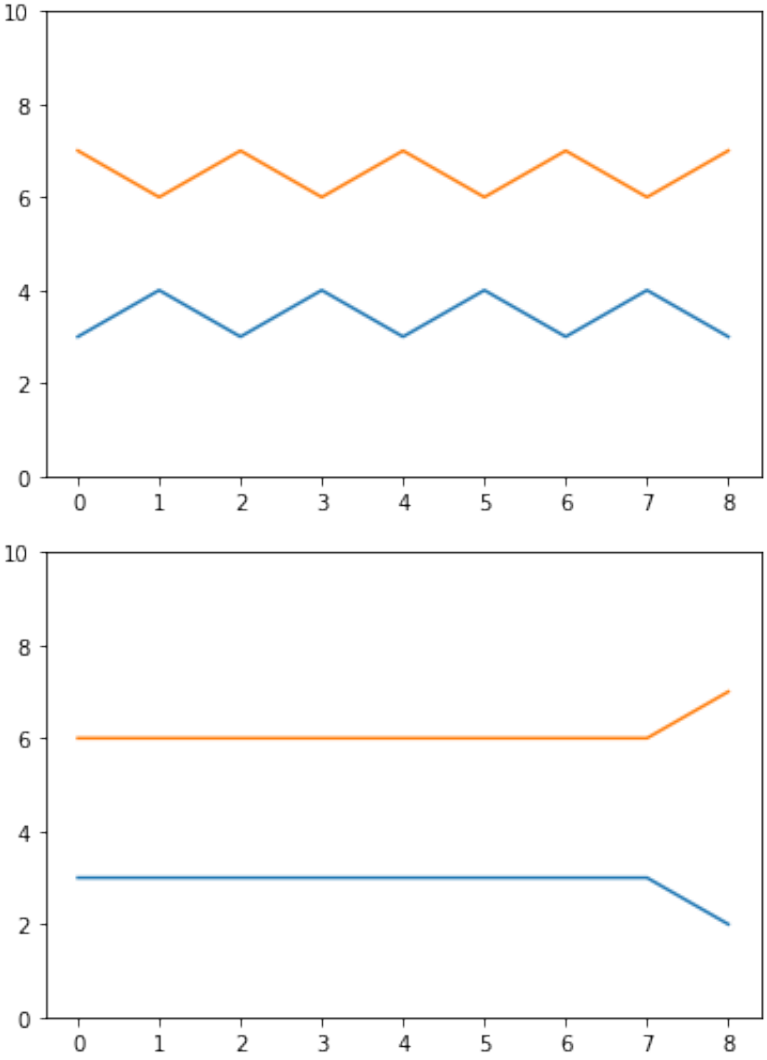
با توجه به موارد گفته شده درباره توابع استخراج ویژگی، در انتها برای هر مجموعه از مدل جایگذاری کلمات که در هنگام آزمایش‌ها استفاده شده، ۱۵ ویژگی داریم. به عنوان مثال اگر از چهار مجموعه جایگذاری کلمات Word2Vec و GloVe برای انجام آزمایشات استفاده کنیم (از هر کدام دو مجموعه با منابع آموزش مختلف) در نهایت ۴ * ۱۵ = ۶۰ ویژگی بدست می‌آوریم.

۲.۴.۳ موتور استنتاج

تحقیقات مختلفی که در زمینه شباهت بین متون و با استفاده از مدل‌های جایگذاری کلمات استفاده شده، عملکرد الگوریتم‌های مختلف یادگیری ماشین بر روی داده‌های استخراج شده به کمک مدل‌های جایگذاری ماشین بررسی شده‌اند.

از آن‌جایی که ما در این پژوهش از پایگاه‌داده MSRP استفاده می‌کنیم و این پایگاه داده Binary است (شباهت یا عدم شباهت) مقالات مختلفی که از این پایگاه داده استفاده کرده‌اند بررسی شده است. در [۴۱] که روشی مشابه این روش استفاده کرده است، آزمایش‌هایی روی

[24] Bag of Dimension
[25] Capture



شکل ۷.۳: دوبردار مختلف با شباهت کسینوسی برابر ۰/۹

مدل‌های مختلف پرکاربرد در زمینه پردازش متن از جمله شبکه‌های عصبی مصنوعی [۲۶] ، درخت‌های تصمیم [۲۷] و دسته‌بند ماشین بردار پشیتبان [۲۸] انجام داده‌است. با توجه به نتایج این تحقیق مدل SVC با تابع هسته RBF برای داده‌های باینری عملکرد بهتری را داشته است، از این رو ما در این تحقیق از این مدل به عنوان موتور استنتاج الگوریتم و ابزاری برای دسته‌بندی جملات استفاده می‌کنیم. پارامترهای مرتبط با این مدل در فصل بعدی بیان می‌شود.

۵.۳ نتیجه گیری

در این بخش ابتدا پایگاه‌داده مورد استفاده معرفی و بخش‌هایی از آن به عنوان نمونه نمایش داده‌شد. شنحوه پیش‌پردازش داده‌ها و مراحلی که برای آماده شدن نیاز است توضیح داده شد. در ادامه الگوریتم کلی روش ارائه شده بیان شد و مزایای این روش نسبت به سایر روش‌ها بیان گردید. در ادامه سپس بخش‌های مختلف روش که شامل توابع استخراج ویژگی و دسته‌بند استفاده شده در روش است شرح داده شد.

[26] Artificial Neural Networks
[27] Decision Tree
[28] Support Vector Machine Classifier (SVC)

فصل ۴

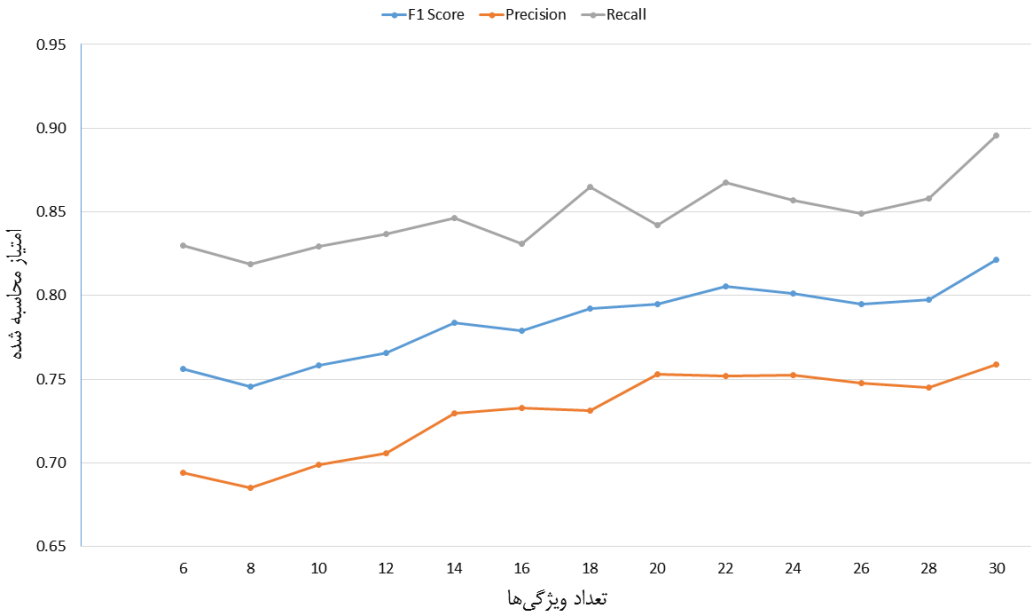
جزئیات پیکربندی و پیاده‌سازی

۱.۴ پیش پردازش

همانطور که گفته شد ٪۷۱ از پایگاه داده برای آموزش و ٪۲۹ نیز برای ارزیابی استفاده شده‌اند. این پایگاه داده به صورت یک فایل نصبی مخصوص سیستم‌عامل‌های ویندوز از پرتال اینترنتی شرکت مایکروسافت قابل دریافت است. بعد از نصب این فایل داده‌های پایگاه‌داده در سه فایل قابل دسترسی است.

در یک فایل تمام جمله‌ها و شناسه هر جمله موجود است. دو فایل دیگر به داده‌های آموزش و ارزیابی را که توسط سازندگان پایگاه داده تفکیک شده است، دارا است. همانطور که در بخش مربوط به پیش‌پردازش در فصل قبل گفته شد، تمام جملات ابتدا به کمک کتابخانه NLTK در زبان برنامه‌نویسی پایتون پردازش می‌شوند تا علائم نگارشی و حروف توقف حذف شوند و توکن‌های قابل استفاده در برنامه بدست بیاید.

در تمام آزمایشات از دو مدل جایگذاری Word2Vec [۳۹] و GloVe[۴۰] استفاده شده است. این مدل‌ها به ترتیب توسط شرکت گوگل و دانشگاه آزمایشگاه پردازش زبان طبیعی دانشگاه



شکل ۱.۴: در این نمودار محور افقی تعداد ویژگی‌ها در هر مرحله را نشان می‌دهد

استنفورد آموزش داده شده‌اند و به صورت عمومی در دسترس محققان قرار گرفته‌اند. با توجه به این نکته تمام توکن‌های بدست آمده در هر دو مدل جست و جو می‌شوند. بعد از اجرای برنامه، تمام توکن‌های موجود در داده‌های آموزش و ارزیابی در این مدل موجود بودند.

۲.۴ پارامترها

در فصل قبل چهار تابع استخراج ویژگی پیشنهاد شده، معرفی و شرح داده‌شدند. هر کدام از این توابع ابتدا با شرایط گفته شده برای هر کدام، شباهت‌ها را محاسبه کرده و نتایج بدست را به عنوان ویژگی برمی‌گردانند. خروجی هر تابع به صورت کاملا مشابه، مشابه و متفاوت در نظر گرفته شده است (به جز تابع DBVM که دو عدد به عنوان میانگین فاصله برمی‌گرداند). در مرحله اول هر تابع تمام ویژگی‌های درنظر گرفته شده را برمی‌گرداند و سیستم با این تعداد ویژگی ارزیابی می‌شود، سپس در مرحله بعد از این تعداد یک ویژگی کم می‌شود و دوباره سیستم با تعداد جدید ویژگی‌ها ارزیابی می‌شود و این روند ادامه پیدا میکند تا هر تابع تنها یک ویژگی بازگرداند (به جز تابع DBVM که دو ویژگی با توجه به دو معیار فاصله کسینوسی و اقلیدسی برمی‌گرداند). در شکل ۱.۴ نتایج با تعداد ویژگی‌های مختلف آورده شده است.

در آزمایش‌های صورت گرفته تعداد ابتدایی ویژگی‌ها (اجرای بار اول) برابر ۱۵ است که با توجه به اینکه از دو مجموع مختلف بردار کلمه جاگذاری شده استفاده می‌کنیم تعداد کل ویژگی‌ها برابر ۳۰ می‌شود.

همانطور که از شکل ۱.۴ مشخص است بهترین تعداد ویژگی ۱۵ ویژگی در مجموع است. با توجه به اینکه تعداد خروجی مناسب هر تابع بدست آمدهه است، با دسته بندی مجموع خروجی‌های هر تابع به تعداد مناسب هر خروجی (تعداد خروجی هر تابع در فصل ۳ آورده شده‌است) بازه‌های مناسب برای دسته‌بندی ویژگی‌های هر تابع بدست آمده‌است.

برای تابع SWSN بازه‌هایی که سه کیفیت مشابه، نسبتا مشابه و متفاوت را نمایش می‌دهند به ترتیب از چپ به راست مطابق زیر است:

$$[0, 0.15), [0.15, 0.4), [0.4, \infty)$$

برای تابع USN بازه‌هایی که سه کیفیت مشابه، نسبتا مشابه و متفاوت را نمایش می‌دهند به ترتیب از چپ به راست مطابق زیر است:

$$[-1, 0.45), [0.45, 0.8), [0.8, \infty)$$

برای تابع BoD چهار دسته انتخاب شده عبارتند از:

$$[-\infty, 0.001), [0.001, 0.01), [0.01, 0.02), [0.02, \infty)$$

همانطور که در توضیحات مربوط به تابع SWSN گفته شد، مقدار IDF نقش بسیار تعیین کننده‌ای در دقت این تابع دارد. از آنجایی که کلمات در پایگاه‌داده مورد استفاده بسیار محدود است، نتایج پیاده‌سازی محاسبه IDF به کمک جملات MSRP تمام کلمات مقادیر تقریبا برابر بدست آوردن که باعث می‌شود عملکرد این تابع به‌شدت کاهش پیدا کند. برای رفع این مشکل از یک پیکره استاندارد به نام INEX 2013[۴۲] برای محاسبه IDF استفاده شده‌است که تاثیر بسیار زیادی در نتایج نهایی داشته است.

۳.۴ معیارها

در این قسمت معیارهای دقت مختلف که برای ارزیابی استفاده شده است بررسی می‌شوند. همانطور که قبلا گفته شد، پایگاه‌داده مورد استفاده در آزمایشات باینری است، معیارهای [1]

[1] Metircs

مناسب برای ارزیابی عبارتند از: دقت [۲] ، Precision ، Recall و F1. در روابطی که در ادامه می‌آیند، هر کدام از نمادها به صورت زیر تعریف شده‌است:

- TP: مخفف کلمه True Positive و به معنای نمونه‌ای است که برچست آن به درستی مثبت تعیین شده‌است (در اینجا تشخیص صحیص شباهت دو نمونه است).
- TN: مخفف کلمه True Negative و به معنای نمونه‌ای است که برچست آن به درستی منفی تعیین شده‌است (در اینجا به معنی تشخیص صحیح عدم‌شباهت دو نمونه است).
- FP: مخفف کلمه False Positive و به معنای نمونه‌ای است که برچست آن به اشتباه مثبت تعیین شده‌است (در اینجا به معنی تشخیص شباهت اشتباه دو نمونه است).
- FN: مخفف کلمه False Negative و به معنای نمونه‌ای است که برچست آن به اشتباه منفی تعیین شده‌است (در اینجا به معنای تشخیص عدم شباهت دو نمونه است که با یکدیگر شباهت دارند).

در ادمه هر معیار را به صورت کوتاه شرح می‌دهیم.

۱.۳.۴ معیار دقت

این معیار نسبت نمونه‌های درست تشخیص داده (مثبت یا منفی) به کل داده‌ها را بیان می‌کند:

$$Accuracy = \frac{TP + TF}{TP + TF + FP + FN} \tag{۱.۴}$$

۲.۳.۴ معیار precision

این معیار، نسبت تعداد نمونه‌های درست نسبت داده شده را به کل نمونه‌های نسبت داده شده به همان کلاس نشان می‌دهد. این معیار درجه صداقت [۳] الگوریتم را بیان می‌کند.

$$Precision = \frac{TP}{TP + FP} \tag{۲.۴}$$

[2] Accuracy
[3] Soundness

۳.۳.۴ معیار Recall

این معیار نسبت تعداد نمونه‌هایی که به درستی در یک کلاس قرار گرفته‌اند را به کل نمونه‌هایی که باید در آن کلاس قرار بگیرند را بیان می‌کند. این معیار درجه تمامیت [۴] الگوریتم نیز نامیده می‌شود.

$$Recall = \frac{TP}{TP + FN} \tag{۳.۴}$$

۴.۳.۴ معیار F_1

این معیار از ترکیب دو معیار بالا بدست می‌آید. دلیل استفاده از این معیار این است که دو معیار بالا به تنهایی برای ارزیابی سیستم موثر نبوده و استفاده مجزا از هر کدام می‌تواند باعث ارزیابی نادرست از سیستم شود. شیوه محاسبه کلی این معیار به شکل زیر است:

$$F_\beta = \frac{(\beta^2 + 1) * Recall \times Precision}{\beta^2(Recall + Precision)} \tag{۴.۴}$$

در رابطه بالا β ضریب برابری

Precision و Recall است که معمولا برابر ۱ در نظر گرفته می‌شود. با توجه به این مسئله این رابطه به شکل زیر درمی‌آید:

$$F_1 = \frac{2 * Recall \times Precision}{Recall + Precision} \tag{۵.۴}$$

۴.۴ نتایج آزمایش

در دو فصل گذشته پارامترهای آزمایش‌ها و معیارهای ارزیابی ارائه شدند. در ادامه نتایج اجرا الگوریتم روی پایگاه داده استاندارد MSRP با روش‌های پایه مقایسه می‌شود. همانطور که از نتایج در جدول ۱.۴ مشخص است این روش تقریبا در تمام موارد نسبت به دو روش دیگر بهتر عمل کرده است. یکی از نقاط قوت این روش نسبت دو روش دیگر این است که مقدار Accuracy به میزان قابل توجهی از دو مورد دیگر بیشتر است. دلیل این مورد این است که

[4] Completeness

جدول ۱.۴: مقایسه نتایج حاصل از پیاده سازی با روش‌های پایه

معیار / روش	Accuracy	Precision	Recall	F_1
ARC-II	0.69	-	-	0.80
SVSM	0.71	0.71	0.95	0.81
روش معرفی شده	0.76	0.78	0.90	0.83

دو روش پایه تمرکز بیشتری در تشخیص شباهت داشتند (مقدار precision تقریبا برابر با یک در روش اول) ، این مساله باعث می‌شود در هنگام استخراج اطلاعات مورد نیاز برای ایجاد ویژگی‌ها، اطلاعاتی که عدم شباهت را بیان می‌کنند نادیده گرفته شود.

مزیت دیگر این روش این است که دو نوع ویژگی از هر متن استخراج می‌کند. یک دسته ویژگی‌های سطح متن و دیگری ویژگی‌های لغوی است (در روش ARC-II تنها به ویژگی‌های سطح متن اهمیت داده می‌شود و بردارها نهایی از ترکیب بردارهای کلمه ایجاد می‌شوند).

نکته دیگر در مورد این روش این است که تنها به اطلاعات معنایی موجود در متن اتکا ندارد و منابعی که معنای کلمات را به خوبی در فضای چند بعدی نشان می‌دهند استفاده می‌کند در صورتی که در دو روش دیگر تنها از معنای قابل استخراج از پیکره استفاده می‌شود در نتیجه در صورتی که کلمه‌ای که از نظر معنایی تاثیری در نتیجه گیری نهایی ندارد به تعداد خیلی کمی در پیکره تکرار شده باشد(به عنوان مثال فعلی که در متن‌های امروزی کاربرد خیلی کمی دارد و معمولا از مترادف‌های آن استفاده می‌شود)، اهمیت زیادی در بردارهای ویژگی دارد.

در ادامه ابتدا تاثیر ویژگی‌های مختلف را بررسی می‌کنیم، سپس تاثیر ویژگی‌های لغوی متن مانند تعداد کلمات مشترک و طول جمله‌ها را در دقت این روش بررسی می‌کنیم.

۱.۴.۴ اهمیت ویژگی‌ها

در این قسمت تاثیر مجموعه‌های مختلف از ویژگی‌ها را بررسی می‌کنیم. برای بررسی تاثیر هر ویژگی از شیوه قطع کردن [۵] استفاده می‌کنیم به این شکل که در هر مرحله ویژگی‌های استخراج

[5] Ablation study

شده از هر یک از توابع را کنار گذاشته و الگوریتم را روی سایر ویژگی‌ها انجام می‌دهیم. در جدول ۲.۴ نتایج آزمایشات اورده شده‌است. همانطور که از نتایج مشخص است، ویژگی‌های

جدول ۲.۴: بررسی تاثیر هر مجموعه از ویژگی‌ها

تابع استخراج ویژگی	Accuracy	Precision	Recall	F_1
SWSN	0.741	0.768	0.874	0.818
BoD	0.746	0.767	0.886	0.822
USN	0.747	0.763	0.898	0.825
DBVM	0.759	0.778	0.892	0.831

حاصل از تابع DBVM کمترین تاثیر را دارند که قابل پیش‌بینی است. این روش، روش پایه استفاده از مدل جایگذاری کلمات برای شباهت است و معمولا برای رفتن از سطح کلمه به سطح متن استفاده می‌شود و همانظور که در فصل قبل روش ایجاد این ویژگی شرح داده‌شده، ویژگی‌های کلی هر دو نمونه را در نظر می‌گیرد و اهمیت و نقش کلمات مختلف در جمله و تفاوت‌های معنایی آن‌ها را به خوبی دریافت نمی‌کند.

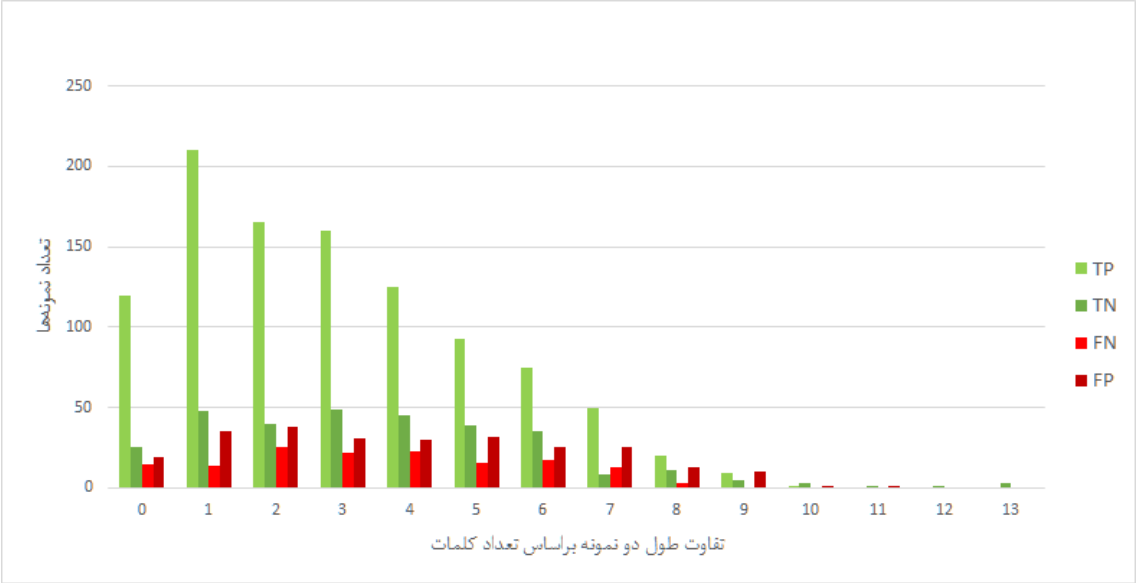
از طرف دیگر بدترین عملکرد این روش زمانی است که تابع استخراج ویژگی SWSN حذف می‌شود. این تابع به صورت ضمنی قابلیت تشخیص مترادف و یا متضاد بودن، شباهت و یا عدم شباهت و از این دست ویژگی‌های لغوی دو کلمه را بدلیل وزنی‌دهی مقدار محاسبه‌شده برای شباهت به کمک IDF داراست و می‌توان با توجه به داده‌های نشان داده‌شده در جدول ۲.۴ نتیجه گرفت که مهمترین تابع استخراج ویژگی در این روش است.

۲.۴.۴ بررسی خطا

در ادامه عملکرد این روش را که مبتنی بر معنا است را به کمک دو ویژگی قواعد نحوی بررسی می‌کنیم: طول جمله‌ها و هم‌پوشانی کلمات دو جمله.

۱.۲.۴.۴ طول جمله

در تصویر ۲.۴ نتایج پیاده سازی مربوط به این بخش آورده شده‌است. همانطور که انتظار



شکل ۲.۴: نتایج اجرا که براساس طول جملات تفکیک شده‌است. معیار طول جملات تعداد توکن‌ها است.

می‌رود جملاتی مشابهی که در پارامتر طول نیز با هم مشابه هستند راحتتر تشخیص داده می‌شوند. البته این نتیجه گیری برای تشخیص صحیح جملات نامشابه خیلی واضح نیست (نمودار مربوط به نرخ TN قوس کمتری دارد) که دلیل این امر نبودن داده منفی کافی در داده‌های ارزیابی است.

یک برداشت دیگر از این نمودار این است در نیمه سمت چپ، مقدار FN علی‌رغم کاهش تعداد نمونه‌ها ثابت است، یعنی افزایش نسبی نرخ این پارامتر با توجه به افزایش تفاوت طول دو جمله در حالی این تفسیر برای FP برعکس است و این یعنی با افزایش تفاوت طول جملات، روش ارائه شده قادر است به عنوان ابزار کارآمدی در تشخیص عدم شباهت استفاده شود. مهم‌ترین برداشت از این نمودار این است که این روش تقریبا در تمام حالت‌ها، برچسب هر داده آموزشی را به درستی پیش‌بینی میکند.

۲.۲.۴.۴ هم‌پوشانی لغوی

در این مرحله میزان هم‌پوشانی توکن‌های هر جفت جمله را محاسبه کردیم و سپس نتایج حاصل از آزمایش‌ها را به صورت نموداری براساس هم‌پوشانی‌ها در شکل ۳.۴ نشان دادیم. همان‌طور که انتظار می‌رود هنگامی که هم‌پوشانی لغوی بسیار زیاد است (سه بازه آخر در

سمت راست تصویر) این روش در اکثر مواقع دو جمله را از نظر معنایی مشابه ارزیابی می‌کند که دلیل این امر وجود کلمات یکسان زیاد در هر سه مورد است. این امر باعث می‌شود نرخ FP در این بازه‌ها زیاد باشد (دو پارامتر دیگر در این بازه‌ها تقریبا صفر هستند).

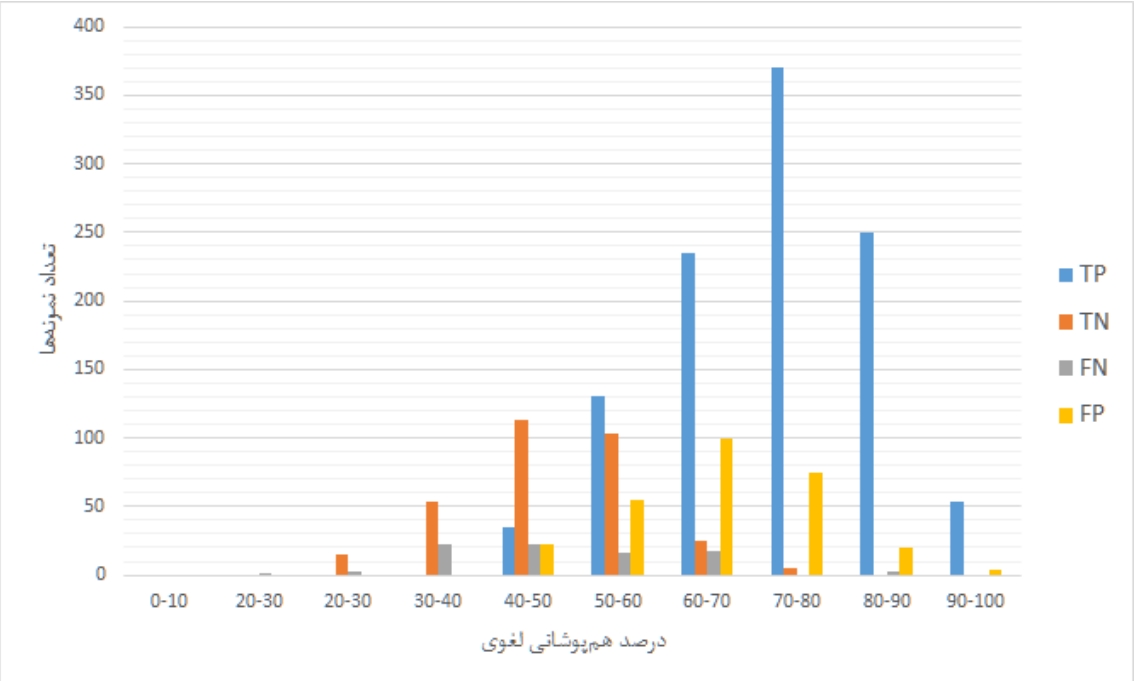
یک نکته جالب آخرین FP در سمت راست هست که بیانگر ۳ مورد FP در اجرا بر روی داده‌های ارزیابی است که سازندگان پایگاه‌داده MSRP این سه مورد را مشابه در نظر نگرفته‌اند اما روش پیشنهادی آنها را مشابه تشخیص داده است. به عنوان مثال در زیر یک مورد از این ۳ مورد آورده شده‌است:

Air Canada, the largest airline in Canada and No. 11 in the world, has been under court protection from creditors since April 1

و

The No. 11 airline in the world, Air Canada has been under court protection from creditors since April 1.

یک نکته دیگر در میانه نمودار (بازه ۵۰ـ۶۰%) است. در این بازه روش ارائه شده در بعضی از موارد نمی‌تواند درست عمل کرده عدم شباهت را به اشتباه تشخیص می‌دهد (نرخ بالای



شکل ۳.۴: نتایج اجرا برای بررسی تاثیر هم‌پوشانی لغوی بر دقت روش.

FN در این بازه) ولی با این وجود باز هم در بسیاری از موارد به درستی عمل کرده شباهت یا عدم شباهت را تشخیص می‌دهد. نکته دیگر این است که حتی زمانی که تعداد زیادی از کلمات تشکیل دهنده دو جمله یکسان نیست (دو مورد ۳۰ـ۴۰٪ و ۴۰ـ۵۰٪) این روش باز هم می‌تواند عدم شباهت را به درستی تشخیص دهد. در کنار تمام این موارد،در تمام بازه‌ها روش ارائه شده در بیشتر موارد تشخیص درستی داده است. این دو مورد نشان می‌دهد این روش بیشتر از شباهت معنایی بهره می‌برد تا شباهت‌های ظاهری.

۵.۴ نتیجه گیری

در این فصل ابتدا مراحل مختلف پیش‌پردازش داده‌ها به صورت کامل توضیح داده‌شد. سپس مقادیر پارامترهای و معیارهای استفاده شده در آزمایش‌ها شرح داده شد. در ادامه عملکرد این روش در مقایسه با روش‌های معرفی شده در فصل دوم، بررسی شد و نقات قوت این روش نسبت آن‌ها بیان شد.

سپس هر یک از توابع استخراج ویژگی به تنهایی مورد آزمایش قرار گرفت و عملکرد هر کدام بررسی شد. در انتهای عملکرد روش پیشنهادی با توجه تفاوت‌های ویژگی‌های لغوی پایگاه‌داده مورد استفاده نیز بررسی شده و نشان داده‌شده که این روش در تشخیص شباهت معنایی متون به خوبی عمل می‌کند.

فصل ۵

نتیجه گیری و پیشنهادات

۱.۵ یافته‌های تحقیق

در این پژوهش روشی پیشنهاد شد که می‌تواند شباهت معنایی بین دو جمله یا پاراگراف را با دقت بالایی تشخیص دهد. یکی از ویژگی‌های این روش این است که تفاوت‌های لغوی و ظاهری جمله‌ها و پاراگراف‌ها تاثیر چشم‌گیری در عملکرد آن نداشته و می‌تواند شباهت یا عدم‌شباهت را در بیشتر موارد با توجه به نتایج بدست آمده از آزمایشات بدرستی تشخیص دهد.

یکی دیگر از ویژگی‌های این روش این است که به منابع دانش خارجی مانند درخت‌های تجزیه، شبکه‌های معنایی و موارد مشابه نیاز ندارد و باز آنجایی که از مدل‌های جایگذاری کلمات برای استخراج ویژگی استفاده می‌کند و با توجه در دسترس بودن این مدل‌های برای زبان‌های مختلف، می‌توان از این روش در سایر زبان‌ها استفاده کرد.

علاوه براین، این روش قابلیت این را دارد که با هر تعداد مدل جایگذاری کلمات، علاوه بر دو مدل استفاده شده در آزمایشات و با ابعاد مختلف استفاده شود. یکی دیگر از ویژگی‌های این

روش نیازهای پایین به منابع محاسباتی مانند حافظه و پردازنده است که در مقایسه روش‌های برپایه شبکه‌های عصبی و یا درخت‌های تجزیه منابع کمتری نیاز دارد. به عنوان مثال ابزار درخت تجزیه معرفی شده در [۴۳] برای تجزیه یک جمله با ۴۰ کلمه نیاز به 500MB حافظه RAM دارد. این نکته در کنار دقت بالای این روش در تشخیص شباهت و یا عدم شباهت، باعث می‌شود این روش برای استفاده در کاربردهای مختلف مناسب باشد.

۲.۵ پیشنهادات و کارهای آینده

همانطور که گفته شد یکی از زمینه‌هایی که تشخیص شباهت معنایی جملات در آن بسیار اهمیت دارد، پرسش و پاسخ خودکار است. سیستم‌های پرسش و پاسخ خودکار از روش‌های مختلف تشخیص شباهت به عنوان موتور استنتاج استفاده می‌کنند. با توجه ویژگی‌های مختلف که برای این روش بیان شد و با توجه به نتایج مناسب آزمایش‌ها، می‌توان از این مدل به عنوان موتور استنتاج سیستم پرسش و پاسخ استفاده کرد.

البته با توجه به ماهیت پایگاه داده استفاده شده که در آن در تمام موارد، دو جمله یا پاراگراف مورد مقایسه از یک نوع بودند (هر دو از نظر دستور زبان، جمله یا پاراگرافی خبری بودند) نمی‌توان به طور قطعی در مورد مناسب بودن توابع استخراج ویژگی برای این منظور اظهار نظر کرد.

یکی دیگر از تحقیقاتی که در ادامه این پژوهش می‌توان در آینده انجام داد، بهبود عملکرد مدل برای در رابطه با مواردی است که ویژگی‌های لغوی متفاوتی دارند و هم‌پوشانی لغات و یا تعداد کلمات مشترک در آن‌ها کم است.

با توجه به این‌که قسمت‌های مختلف الگوریتم استخراج ویژگی، به یکدیگر برای محاسبه مقادیر بردارهای ویژگی وابستگی ندارند، می‌توان الگوریتم ارائه شده در فصل سوم را به صورت موازی اجرا کرد تا زمان استخراج ویژگی‌ها کاهش یابد و سیستم برای استفاده در کاربردهای روزمره مناسب‌تر شود.

مراجع

[1] Hall, Patrick AV and Dowling, Geoff R. “Approximate string matching.” **ACM computing surveys (CSUR)**, 12(4):381–402, 1980.

[2] Jaro, Matthew A. “Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida.” **Journal of the American Statistical Association**, 84(406):414–420, 1989.

[3] Jaro, Matthew A.“Probabilistic linkage of large public health data files.” **Statistics in medicine**, 14(5-7):491–498, 1995.

[4] Barrón-Cedeno, Alberto, Rosso, Paolo, Agirre, Eneko, and Labaka, Gorka. “Plagiarism detection across distant language pairs.” **Proceedings of the 23rd International Conference on Computational Linguistics**, pp. 37–45. Association for Computational Linguistics, 2010.

[5] Jiang, Jay J and Conrath, David W. “Semantic similarity based on corpus statistics and lexical taxonomy.” **arXiv preprint cmp-lg/9709008**, 1997.

[6] Sebastiani, Fabrizio. “Machine learning in automated text categorization.” **ACM computing surveys (CSUR)**, 34(1):1–47, 2002.

[7] Apté, Chidanand, Damerau, Fred, and Weiss, Sholom M. “Automated learning of decision rules for text categorization.” **ACM Transactions on Information Systems (TOIS)**, 12(3):233–251, 1994.

[8] Yih, Wen-tau, Toutanova, Kristina, Platt, John C, and Meek, Christopher. "Learning discriminative projections for text similarity measures." **Proceedings of the fifteenth conference on computational natural language learning**, pp. 247–256. Association for Computational Linguistics, 2011.

[9] Winkler, William E. "String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage.", 1990.

[10] Lintean, Mihai and Rus, Vasile. "Measuring semantic similarity in short texts through greedy pairing and word semantics." **Twenty-Fifth International FLAIRS Conference**, 2012.

[11] Mihalcea, Rada, Corley, Courtney, Strapparava, Carlo, et al. "Corpus-based and knowledge-based measures of text semantic similarity." **AAAI**, vol. 6, pp. 775–780, 2006.

[12] Barrington, Luke, Chan, Antoni, Turnbull, Douglas, and Lanckriet, Gert. "Audio information retrieval using semantic similarity." **2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07**, vol. 2, pp. II–725. IEEE, 2007.

[13] Angell, Richard C, Freund, George E, and Willett, Peter. "Automatic spelling correction using a trigram similarity measure." **Information Processing & Management**, 19(4):255–261, 1983.

[14] Yamamoto, Tetsuo, Matsushita, Makoto, Kamiya, Toshihiro, and Inoue, Katsuro. "Measuring similarity of large software systems based on source code" correspondence. **International Conference on Product Focused Software Process Improvement**, pp. 530–544. Springer, 2005.

[15] Lin, Jimmy and Katz, Boris. "Question answering from the web using knowledge annotation and knowledge mining techniques." **Proceedings of the twelfth in-**

ternational conference on Information and knowledge management, pp. 116–123. ACM, 2003.

[16] Gupta, Vishal, Lehal, Gurpreet S, et al. "A survey of text mining techniques and applications." **Journal of emerging technologies in web intelligence**, 1(1):60–76, 2009.

[17] Bilenko, Mikhail and Mooney, Raymond J. "Adaptive duplicate detection using learnable string similarity measures." **Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining**, pp. 39–48. ACM, 2003.

[18] Yujian, Li and Bo, Liu. "A normalized levenshtein distance metric." **IEEE transactions on pattern analysis and machine intelligence**, 29(6):1091–1095, 2007.

[19] Cohen, William W, Ravikumar, Pradeep, Fienberg, Stephen E, et al. "A comparison of string distance metrics for name-matching tasks." **IIWeb**, vol. 2003, pp. 73–78, 2003.

[20] Brown, Peter F, Desouza, Peter V, Mercer, Robert L, Pietra, Vincent J Della, and Lai, Jenifer C. "Class-based n-gram models of natural language." **Computational linguistics**, 18(4):467–479, 1992.

[21] Nakov, Preslav. "Improving english-spanish statistical machine translation: experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing." **Proceedings of the third workshop on statistical machine translation**, pp. 147–150. Association for Computational Linguistics, 2008.

[22] Barrón-Cedeno, Alberto, Eiselt, Andreas, and Rosso, Paolo. "Monolingual text similarity measures: A comparison of models over wikipedia articles revisions." **Proceedings of the ICON: 7th International Conference on NLP**, pp. 29–38. Citeseer, 2009.

[23] Barzilay, Regina and Elhadad, Noemie. “Sentence alignment for monolingual comparable corpora.” **Proceedings of the 2003 conference on Empirical methods in natural language processing**, pp. 25–32. Association for Computational Linguistics, 2003.

[24] Nelken, Rani and Shieber, Stuart M. “Towards robust context-sensitive sentence alignment for monolingual corpora.” **11th Conference of the European Chapter of the Association for Computational Linguistics**, 2006.

[25] McKeown, Kathleen, Klavans, Judith L, Hatzivassiloglou, Vasileios, Barzilay, Regina, and Eskin, Eleazar. “Towards multidocument summarization by reformulation: Progress and prospects.” “1999.

[26] Shrestha, Prajol. “Corpus-based methods for short text similarity.” **Rencontre des Étudiants Chercheurs en Informatique pour le Traitement automatique des Langues**, vol. 2, p. 297, 2011.

[27] Hearst, Marti A. “Automated discovery of wordnet relations.” **WordNet: an electronic lexical database**, pp. 131–153, 1998.

[28] Deerwester, Scott, Dumais, Susan T, Furnas, George W, Landauer, Thomas K, and Harshman, Richard. “Indexing by latent semantic analysis.” **Journal of the American society for information science**, 41(6):391–407, 1990.

[29] Han, Jiawei, Kamber, Micheline, and Mining, “Data. Concepts and techniques.” **Morgan Kaufmann**, 340:94104–3205, 2001.

[30] Abdel-Hamid, Ossama, Mohamed, Abdel-rahman, Jiang, Hui, and Penn, Gerald. “Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition.” **2012 IEEE international conference on Acoustics, speech and signal processing (ICASSP)**, pp. 4277–4280. IEEE, 2012.

[31] Lewis, David D, Yang, Yiming, Rose, Tony G, and Li, Fan. "Rcv1: A new benchmark collection for text categorization research." **Journal of machine learning research**, 5(Apr):361–397, 2004.

[32] Dahl, George E, Sainath, Tara N, and Hinton, Geoffrey E. "Improving deep neural networks for lvcsr using rectified linear units and dropout." **2013 IEEE international conference on acoustics, speech and signal processing**, pp. 8609–8613. IEEE, 2013.

[33] Bordes, Antoine, Glorot, Xavier, Weston, Jason, and Bengio, Yoshua. "A semantic matching energy function for learning with multi-relational data." **Machine Learning**, 94(2):233–259, 2014.

[34] LeCun, Yann, Bottou, Léon, Bengio, Yoshua, Haffner, Patrick, et al. "Gradient-based learning applied to document recognition." **Proceedings of the IEEE**, 86(11):2278–2324, 1998.

[35] Caruana, Rich, Lawrence, Steve, and Giles, C Lee. "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping." **Advances in neural information processing systems**, pp. 402–408, 2001.

[36] Quirk, Chris, Brockett, Chris, and Dolan, William. "Monolingual machine translation for paraphrase generation." **Proceedings of the 2004 conference on empirical methods in natural language processing**, pp. 142–149, 2004.

[37] Islam, Aminul and Inkpen, Diana. "Semantic text similarity using corpus-based word similarity and string similarity." **ACM Transactions on Knowledge Discovery from Data (TKDD)**, 2(2):10, 2008.

[38] Hu, Baotian, Lu, Zhengdong, Li, Hang, and Chen, Qingcai. "Convolutional neural network architectures for matching natural language sentences." **Advances in neural information processing systems**, pp. 2042–2050, 2014.

[39] Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg S, and Dean, Jeff. "Distributed representations of words and phrases and their compositionality." **Advances in neural information processing systems**, pp. 3111–3119, 2013.

[40] Pennington, Jeffrey, Socher, Richard, and Manning, Christopher D. "Glove: Global vectors for word representation." **Empirical Methods in Natural Language Processing (EMNLP)**, pp. 1532–1543, 2014.

[41] Sokolova, Marina and Lapalme, Guy (2009). "A systematic analysis of performance measures for classification tasks" **Information Processing & Management**, 4,45 ,427–437.

[42] Bellot, Patrice, Doucet, Antoine, Geva, Shlomo, Gurajada, Sairam, Kamps, Jaap, Kazai, Gabriella, Koolen, Marijn, Mishra, Arunav, Moriceau, Véronique, Mothe, Josiane, et al. "Report on inex 2013." **ACM SIGIR Forum**, vol. 47, pp. 21–32. ACM, 2013.

[43] Klein, Dan and Manning, Christopher D. "Accurate unlexicalized parsing." **Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1**, pp. 423–430. Association for Computational Linguistics, 2003.

Aabstract

Nowadays, given the increasing amount of information and documentation in various fields, quick access to information is of particular importance to each individual. Hence, in addition to information retrieval techniques, summarization and categorization techniques can also be helpful in increasing the speed of users access to their documents. . The construction of a system that can effectively identify the similarity between the two terms has been the subject of many studies. Determining the distance between the two words can be done through a similarity between words or by machine learning methods.

In this research, a method is proposed that, in terms of the meaning of words in each sentence, identifies the similarity between sentences. To get the meaning of each word, we use word placement models. One of the features of these models is that they represent each word in a multi-dimensional space, so that with the various operations of vectors such as the dual-welded collections, one can obtain the meaning of the neighborhood of two words. In the following, with the help of the four function extraction functions, the properties of the terms are extracted, then these attributes are used in a category to identify the similarity or non-identity of the two sentences.

In making such a system, one of the most important components of the ability to recognize the similarity between sentences and paragraphs of the texts is the subject of much research. This method can recognize the semantic similarity between the two sentences in spite of their lexical similarity. In addition to recognizing similarity, this method is also effective in detecting the lack of similarity between the two expressions, so that after doing the tests, this method accurately categorized 83% of the data tested, which performs better than the introduced methods.

Faculty of Computer Engineering

MSc Thesis in Artificial Intelligence Engineering

Sentence and paragraph similarity using word embeding models

By: Morteza Allahpour

Supervisor:

Morteza Zahedi

Advisor:

Hoda Mashayekhi

July 2019