7th International Conference on Web Research
May 19-20, 2021; Tehran, IRAN ICWR2021

Offensive Context Detection in Search Engines Using Machine Learning

[1] Nima Seyfi, [2] Mahdi Aminian

[1] MSc Student, Computer Engineering, University of Guilan, Rasht, Iran
Nimaseyfi@msc.guilan.ac.ir

[2] Assistant Professor, Computer Engineering, University of Guilan, Rasht, Iran
Mahdi.aminian@guilan.ac.ir

Abstract

Due to the expansion of content in various medias and communication platforms, as well as users access to these facilities, the requirement of shared contents checking is more considered. Specifically, it can be more important in the cultural and social contexts to provide High-quality data for people working in these fields. Detection of offensive contexts is one of important web researches, which is used in textual contents, for example children's contents, cultural, academic, and other subjects. A preprocessed dataset is learned by Machine Learning methods (SVM, Naïve Bayes and KNN), and the final model can detect the possibility of offensive texts received as inputs. The data, we are looking for, is a collection of searches performed on a Persian search engine. In order to increase contents dataset, these queries have been re-searched in Google and added the first page of the results to the dataset, then we determined whether the data is rude or not (labeling). The selected model will learn this data and then the trained model that can detect the possibility that the input data is offensive. The results show that the precision of the Naïve Bayes, SVM and KNN models can be 94.05%, 97.28% and 86.48%, respectively.

Keywords: Machine Learning, Natural Language Processing, Offensive Language Detection, SVM, Naïve Bayes, KNN

7th International Conference on Web Research
May 19-20, 2021; Tehran, IRAN ICWR2021

تشخیص متون توهین‌آمیز در موتورهای جستجو با استفاده از یادگیری ماشین

نیما سیفی[1]، مهدی امینیان[2]

[1] دانشجوی کارشناسی ارشد، گروه مهندسی کامپیوتر، دانشگاه گیلان، رشت، ایران
nimaseyfi@msc.guilan.ac.ir

[2] استادیار، گروه مهندسی کامپیوتر، دانشگاه گیلان، رشت، ایران
mahdi.aminian@guilan.ac.ir

چکیده

با توجه به گسترش محتوا در بسترهای رسانه‌ای و ارتباطی مختلف و همچنین دسترسی کاربران به این امکانات، لزوم بررسی محتوای به اشتراک گذاشته شده به ویژه در ابعاد فرهنگی و اجتماعی به منظور ارائه داده‌های با کیفیت به افراد حاضر در این عرصه‌ها همواره احساس می‌شود. یکی از مسائلی که در محتوای متنی، به خصوص محتوای ویژه کودکان، فرهنگی، دانشگاهی و ... بسیار پر اهمیت است تشخیص متون توهین‌آمیز به کار برده شده است که در این مقاله به آن پرداخته می‌شود. با استفاده از یادگیری ماشین (SVM، Naïve Bayes و KNN) داده‌های پیش‌پردازش شده را به مدل مورد نظر آموزش می‌دهیم و انتظار داریم که خروجی مدلی باشد که با دریافت متن احتمال رکیک بودن محتوا را تشخیص دهد. داده‌های مورد نظر مجموعه‌ای از جستجوهای انجام شده در یک موتور جستجوی فارسی هستند که به منظور افزایش محتوا، دوباره این عبارات را در گوگل جستجو کرده و صفحه اول نتیجه را به داده‌ها اضافه می‌کنیم. سپس تشخیص می‌دهیم که داده مورد نظر رکیک می‌باشد یا خیر (برچسب گذاری). مدل مورد نظر این داده‌ها را یادگیری کرده و پس از آن مدلی داریم که می‌تواند احتمال رکیک بودن داده ورودی را تشخیص دهد. نتایج بدست‌آمده نشان می‌دهد که معیار اندازه گیری صحت (Precision) در مدل های Naïve Bayes، SVM و KNN به ترتیب برابر با ٪۹٤.۰۵ ، ٪۹۷.۲۸ و ٪۸٦.٤۸ خواهد بود.

کلمات کلیدی

یادگیری ماشین، پردازش زبان طبیعی، تشخیص کلمات رکیک، SVM ، Naïve Bayes ، KNN.

۱ – مقدمه

به منظور تشخیص رکیک بودن محتوا به دلایلی مثل وجود نگارش‌های متفاوت از کلمات و همچنین ترکیباتی از کلمات که می‌توانند به تنهایی بی‌ایراد باشند ولی در کنار هم به ایجاد متحوای رکیک منجر شوند، استفاده از یادگیری ماشین می‌تواند کاربرد مؤثری داشته باشد. یادگیری ماشین روشی است که با استفاده از شبکه‌های عصبی به یادگیری داده‌های تعیین شده می‌پردازد و در نهایت مدلی تهیه خواهد شد که با دریافت ورودی مشابه می‌تواند به دسته‌بندی آن بپردازد [2]. بنابراین با توجه به این توضیحات می‌بایست داده‌ها برای یادگیری آماده شده و تغییرات مدنظر برروی آنها اعمال شود. در این مقاله از روش یادگیری بانظارت استفاده می‌شود که ورودی‌های یادگیری دارای برچسب هستند و با توجه به آنها دسته‌بندی انجام می‌شود. داده‌های متنی، به ویژه داده‌های متنی فارسی که می‌توانند دارای نگارش‌های متعددی باشند، نیاز به اصلاحات و یکسان‌سازی‌هایی دارند که در مرحله پیش‌پردازش در این مقاله به آنها می‌پردازیم. از جمله این پیش‌پردازش ها می‌توان به نرمال‌سازی متن، ساخت توکن، حذف کلمات توقف، Stemming و Lemmatizing، تبدیل حروف بزرگ به کوچک (lower case) در کلمات انگلیسی و پردازش TF-IDF اشاره کرد.

به طور کلی عبارات توهین‌آمیز را می‌توان در دو دسته عبارات رکیک و سخنان تنفرآمیز دسته‌بندی نمود [20][15]. دسته اول مرتبط با عباراتی هستند که دارای کلمات رکیک و یا بخش‌های ناسزا هستند و

معمولا به شکل عادی قابل تشخیص هستند. دسته دوم عبارات تنفرآمیز هستند که به افراد، اشیا، محتوا}، گروه‌ها، ادیان و... نسبت داده می‌شوند و اکثراً بدون اطلاع از ارتباطات موجود میان موضوع و عبارت تنفرآمیز مورد نظر قابل تشخیص نیستند. تمرکز اصلی این مقاله برروی تشخیص عبارات توهین‌آمیز در بستر اینترنت با توجه به عبارت جستجو شده توسط کاربر است. به همین علت، نوع نگارش بکار برده شده نسبت به متن‌های دیگر، برای مثال توئیت‌های یک کاربر توئیتر و یا نظر یک کاربر در مورد یک موضوع، می‌تواند متفاوت باشد. چرا که یک عبارت جستجو شده می‌تواند فاقد اجزای هر جمله (از جمله فاعل، مفعول و فعل) باشد و حتی شاید در میان کلمات یک عبارت جستجو شده ارتباط مستقیمی برقرار نباشد.

چالش دیگری که در این مقاله با آن روبرو هستیم، کوتاه بودن داده مورد نظر است. به‌طور معمول زمانی که یک جستجو صورت می‌گیرد، عبارت جستجو شده دارای تعداد کلمات محدود و کمی است و نمونه‌های زیادی وجود دارند که ثابت می‌کنند خود این کلمات حاوی مفهوم رکیکی نمی‌باشند، اما نتیجه بدست‌آمده در موتور جستجوی مورد نظر دارای محتوای رکیک و یا توهین آمیز است.

در این مقاله پس از بررسی مختصر اقدامات پیشین انجام شده، به ارائه راهکارهایی به‌منظور رفع چالش‌ها و تشخیص توهین‌آمیز بودن عبارت جستجوشده خواهیم پرداخت و در انتها نتایج این مقاله را با نتایج بدست‌آمده دیگر اقدامات در این موضوع مقایسه می‌کنیم.

۲- کارهای پیشین

با توجه به اینکه در این مقاله از متون فارسی به عنوان ورودی استفاده شده است، مقالاتی که مستقیما به موضوع این مقاله مرتبط باشند، تا به زمان انتشار این مقاله یافت نشده است. اما به دلیل وجود مشابهت در ماهیت رکیک بودن محتوا و تشخیص ناسزا در فرهنگ‌های مختلف می‌توان کارهای مرتبط دیگر را مورد بررسی قرار داد. تشخیص محتوای توهین‌آمیز از موضوعات کاربردی و پرطرفدار تحقیقاتی است به ویژه به منظور استفاده در شبکه‌های اجتماعی. یکی از کارهای انجام شده توسط یین و همکاران [2] انجام شده است که از Ngram و TF-IDF به عنوان ویژگی در یادگیری با نظارت استفاده می‌کند. اشمیت و ویگاند [3] روشی برای تشخیص خودکار سخنان توهین‌آمیز با استفاده از پردازش زبان طبیعی به کار برده‌اند. در این روش از ویژگی‌های سطحی ساده، خلاصه‌سازی کلمات، ویژگی‌های زبان شناختی و ویژگی‌های پایگاه دانش متون استفاده می‌شود. یکی از کارهای مشترک به نام GeramEval در سال ۲۰۱۸ توسط ویگاند و همکاران [4] سازماندهی شد. تمرکز این روش بر تشخیص محتوای رکیک در توییت‌های آلمانی زبان بود که به یادگیری و دسته‌بندی ۸۵۰۰ توییت برچسب گذاری شده می‌پرداخت. دقت نهایی بدست‌آمده از این روش ۷۶/۷۷ درصد بود. کار مشترک دیگری توسط کومار و همکاران [5] انجام شد. مجموعه داده ارائه شده در این کار متشکل از ۱۵۰۰۰ متن و نظرات برچسب شده موجود در شبکه اجتماعی فیسبوک به زبان انگلیسی و هندی بود که در نهایت به دقت دسته‌بندی ۶۴ درصدی در هر دو زبان انگلیسی و هندی رسید.

در ادامه به متدولوژی و پیاده‌سازی موردنظر این مقاله خواهیم پرداخت.

۳- متدولوژی

کاری که در این مقاله انجام شده است به سه بخش تهیه مجموعه داده، پیش‌پردازش و یادگیری ماشین تقسیم می‌شود.

۳-۱- تهیه مجموعه داده

در این بخش به تهیه داده‌های متنی مورد نیاز در زمان یادگیری می‌پردازیم. این داده‌ها از عبارات جستجو شده در پارسی‌جو [13]، که یک موتور جستجوی فارسی است، تشکیل شده‌اند (شکل (۱)) که به منظور ایجاد داده‌های مرتبط معنادار به این عبارات، از جستجوی دوباره این عبارات در موتور جستجوی گوگل استفاده شده است. این داده از جستجو های کاربران واقعی در سال ۲۰۱۸ جمع‌آوری شده اند.

در هر جستجو صفحه اول اطلاعات بدست‌آمده نیز به عبارات مورد نظر اضافه می‌شود. داده‌های اضافه شده شامل لینک، عنوان و توضیحات هر نتیجه بدست‌آمده می‌باشد. به دلیل اینکه در این داده‌ها عبارات انگلیسی نیز یافت می‌شود، می بایست توهین‌آمیز بودن بخش‌های انگلیسی نیز بررسی شود.

۳-۲- پیش‌پردازش

در این بخش داده‌های بدست‌آمده پیش‌پردازش شده و داده‌های مورد نیاز برای یادگیری بوجود آمده و ویژگی‌ها استخراج می‌شود. این پیش‌پردازش-ها شامل نرمال‌سازی کلمات فارسی و انگلیسی، توکن‌سازی از کلمات، حذف کلمات توقف فارسی و انگلیسی و در نهایت اعمال الگوریتم TF-IDF است. در ادامه و در بخش پیاده‌سازی بیشتر به این موارد پرداخته می‌شود.

۳-۳- یادگیری ماشین

در این بخش، داده‌های پیش‌پردازش شده توسط دسته‌بندی کننده‌های SVM، Naïve Bayes و KNN یادگیری شده و دسته‌بندی می‌شوند. مدل‌های نهایی این شبکه‌ها نیز قابل استفاده خواهند بود.

٤- پیاده‌سازی

٤-۱- آماده سازی داده‌های متنی

10.69.59.148	2018-01-13 00:05:10.0	دانلود فیلم با لینک مستقیم	music2pc.ir	http://music2pc.ir/	-1	A33C14AC9D831A5A943D9B65027C9A14	
10.57.251.82	2018-01-13 00:05:11.0	سایت بانک مهر اقتصاد	eft.mehrbank.org	https://eft.mehrbank.org/	106	A33C14AC95FFBB596A51211B02D09A04	
151.235.15.209	2018-01-13 00:05:14.0	فایروال شید forum.soft98.ir	https://forum.soft98.ir/archive/index.php/t-24772.html	-1	A33C14AC8CCC4F5A853DCA6302A29527		
109.162.136.177	2018-01-13 00:05:16.0	digikala	digikala.com	https://www.digikala.com/	-1	A33C14ACEF1B595A873D066402D1012B	
10.57.2.200	2018-01-13 00:05:33.0	الودگی هوا مشهد	khabaronline.ir	http://www.khabaronline.ir/detail/384953	128	A33C14AC2D7B4F5A7E3DBB6302973627	
10.69.59.148	2018-01-13 00:05:40.0	دانلود فیلم با لینک مستقیم	mediayasin.ir	http://mediayasin.ir/	-1	A33C14AC9D831A5A943D9B65027C9A14	
10.122.8.50	2018-01-13 00:05:42.0	pocketgame	pocketgame.ir	http://www.pocketgame.ir/	-1	A33C14ACEE074D5A843D8B630226B025	
151.238.57.93	2018-01-13 00:05:56.0	خلافی خودرو	rahvar120.ir	http://rahvar120.ir/?siteid=1	-1	A33C14AC1E5F7F57D85A6C40028C9E0B	
10.114.83.200	2018-01-13 00:06:02.0	آهنگ مصطفی پاشایی	pop-music.ir	http://pop-music.ir/tag/دانلود-آهنگ-مصطفی-پاشایی/	-1	A33C14AC771A595A803D066302E1B02A	
185.134.97.82	2018-01-13 00:06:04.0	شیپور	sheypoor.com	https://www.sheypoor.com/	-1	A33C14AC3134075A8C3D586402E3710E	
10.114.83.200	2018-01-13 00:06:05.0	آهنگ مصطفی پاشایی	bia4music.org	http://bia4music.org/category//مصطفی-پاشایه/	1-	A33C14AC771A595A803D066302E1B02A	
10.114.96.206	2018-01-13 00:06:34.0	آهنگ کردی iranstyle.ir	http://www.iranstyle.ir/پیدا-شدن-جنازه-دو-نامزد-دانشجو-در-اغوش-ه/	1-	A33C14ACB1817C588D3A043D02FB6503		
10.75.78.182	2018-01-13 00:06:39.0	سامانه همگام مدارس	hamgam.medu.ir	http://hamgam.medu.ir/	365	A33C14AC0FF85959A6076C330229A30B	
2.186.11.113	2018-01-13 00:06:45.0	موبوگرام برای pc	telpedia.ir	http://telpedia.ir//دانلود-رایگان-موبوگرام-برای-کامپیوتر/	1-	A33C14AC5598575A873D066402A9692A	
10.114.96.206	2018-01-13 00:06:46.0	آهنگ کردی iranstyle.ir	http://www.iranstyle.ir/پیدا-شدن-جنازه-دو-نامزد-دانشجو-در-اغوش-ه/	1-	A33C14ACB1817C588D3A043D02FB6503		
10.122.8.50	2018-01-13 00:07:06.0	pocketgame	pocketgame.ir	http://www.pocketgame.ir/	-1	A33C14AC331C595A823D3563026D2D2B	
158.58.112.74	2018-01-13 00:07:12.0	سایا یدک saipayadak.org	http://saipayadak.org/	153	A33C14AC593F545A913DA4650289B129		
2.186.11.113	2018-01-13 00:07:35.0	موبوگرام برای pc	shwebfa.ir	http://shwebfa.ir//موبوگرام-برای-کامپیوتر/	1-	A33C14AC5598575A873D066402A9692A	
10.69.59.148	2018-01-13 00:07:56.0	دانلود فیلم ایرانی	iran-serial.ir	http://iran-serial.ir/film-irani/	-1	A33C14AC9D831A5A943D9B65027C9A14	
83.123.171.240	2018-01-13 00:08:03.0	booking.c0m	samesites.net	http://samesites.net/www/hotel4booking.com	-1	A33C14AC8C9C4F5A8F3D626502E5E226	
217.60.80.94	2018-01-13 00:08:18.0	خرید خانه در مشهد	amlakshargh.ir	http://amlakshargh.ir/	-1	A33C14AC2E98425A953DCF65023B1B22	
10.114.97.9	2018-01-13 00:08:19.0	نمونه سوالات پیام نور	pnunews.com	http://pnunews.com/soalpnu	-1	A33C14ACAA6D435A813DBF630207AB22	

شکل (۱): بخشی از داده‌های مورد استفاده

با توجه به نوع داده‌های ورودی، که مجموعه از جستجوهای کاربران در یک موتور جستجو می‌باشد، نیاز است تا به منظور استخراج داده‌های مورد نیاز که

عبارت‌های جستجو است پردازشی صورت پذیرد. این پردازش شامل مراحل زیر است:

۱- استخراج عبارات جستجو (کوئری‌ها)

۲- جستجو مجدد عبارات در گوگل

۳- جمع‌آوری داده‌های جدید با توجه به جستجو انجام شده به ازای هر کوئری

۴- برچسب گذاری داده‌های بدست‌آمده به منظور تشخیص رکیک بودن عبارات بدست‌آمده

۵- ساخت مجموعه داده با توجه داده‌های برچسب گذاری شده

در ادامه شرح موارد ذکر شده توضیح داده می‌شود.

٤-١-١- استخراج و حذف داده‌های تکراری (کوئری ها)

شکل (۱) بخشی از داده‌های مورد استفاده در مقاله را نمایش می‌دهد.

همانطور که دیده می‌شود هر خط از این فایل مربوط به اطلاعات جستجو یک کوئری است که با ترتیب زیر (راست به چپ) در فایل قرار داده شده‌اند :

کد هش – کد خروجی سرور– لینک کلیک شده – وبسایت انتخاب شده – کوئری – زمان جستجو – آدرس آی‌پی کاربر

با توجه به اینکه از هر خط داده تنها به کوئری نیاز است، باقی اطلاعات باید حذف شوند. همچنین برای جلوگیری از تکرار در هنگام استخراج عبارات جستجو، کوئری‌های تکراری نباید در نظر گرفته شوند.

٤-١-٢- جستجو عبارات در گوگل و ساخت فایل splitted

پس از اینکه کوئری ها را بدست آوردیم نیاز است تا اطلاعات بیشتری به کوئری‌ها اضافه کنیم تا بدانیم که هر عبارت جستجو شده برای موتورهای جستجو به چه معنی است و اطلاعات تکمیلی بیشتری داشته باشیم، بنابر این پس از اینکه هر عبارت را در گوگل جستجو کردیم و صفحه اول نتیجه را دریافت کردیم، اطلاعات را به صورت جدول (۱) بخش بندی می‌کنیم. نتایج بدست‌آمده، مجموعه‌ای از موارد مشابه شکل (۲) خواهند بود

همچنین، نمونه ای از یک فایل ایجاد شده به صورت جدول (۱) است.

همانطور که در جدول (۱) دیده می‌شود ستون اول این فایل کوئری مورد نظر است و ستون titles شامل عنوان‌های نتایج صفحه اول جستجو می‌باشد. ستون links شامل لینک‌های نتایج و descs توضیحات زیر هر نتیجه است.

٤-١-٣- تشخیص کلمات رکیک فارسی و انگلیسی به کمک ابزارها (برچسب گذاری)

پس از جمع‌آوری داده‌ها نوبت به تشخیص رکیک بودن متون می رسد که برای اطمینان از چند طریق انجام می‌شود:

۱ – تشخیص با استفاده از ابزار تشخیص کلمات رکیک فارسی (سایت text-mining.ir [12])

۲ – استفاده از better_porfanity [11]که یک ماشین یادگیری شده برای تشخیص متون رکیک انگلیسی است.

۳ – استفاده از کلمات رکیک موجود در دیکشنری مورد نظر و اعلام رکیک بودن در صورت وجود در متن

شکل (۲): نمونه یک جستجو

جدول (۱): بخشی از یک نمونه فایل splitted

querie	titles	links	descs
آپارات	آپارات – سرویس اشتراک ویدیو برنامه آپارات – دانلود | کافه بازار برنامه آپارات کودک – دانلود | کافه بازار آپارات – ویکی‌پدیا، دانشنامهٔ آزاد آپارات – زومیت آپاراتYJC –	www.aparat.com cafebazaar.ir › app › com.aparat cafebazaar.ir › app › com.aparat.kids fa.wikipedia.org › wiki › آپارات www.zoomit.ir › aparat www.yjc.ir › tags › آپارات	در آپارات وارد شوید تا ویدیوهای و کانال‌های بهتری بر اساس سلیقه شما پیشنهاد شود وارد شوید ۰ آپارات در موبایل Free - Android Free - Android آپارات از سرویس‌های اشتراک‌گذاری ویدئو در ایران است. و...

پس از ارسال متن به سرور، در پاسخ به ازای کلماتی که رکیک تشخیص داده می‌شوند یک تگ ارسال می‌شود که به صورت زیر تقسیم می‌شوند:

StrongSwearWord : کلمه رکیک قطعی

MildSwearWord : کلمه رکیک احتمالی

پس از دریافت نتیجه با شمارش تعداد کلمات رکیک و مقایسه با Strictness تصمیم‌گیری می‌شود متن رکیک اعلام شود یا خیر. هر چه Strictness کمتر باشد احتمال تشخیص رکیک بودن کلمات بالا می‌رود و حداقل می‌تواند مقدار 3 داشته باشد.

برای استفاده از ابزار better_porfanity نیاز است تا مدل یادگیری شده این ابزار در کد پیاده‌سازی شده مقاله بارگزاری شود و با ارسال متن به این مدل می توان خروجی مربوط به کلمات توهین‌آمیز موجود در متن را دریافت نمود.

در مورد روش دوم از یک دیکشنری به نام offensive_dictionary، که شامل یک کلمه رکیک در هر خط است، استفاده می‌شود. در صورتی که متن مورد نظر دارای کلماتی که در دیکشنری قرار گرفته است بود، آن متن را رکیک تشخیص می‌دهد. در دیکشنری استفاده شده در این مقاله ۱۷۳ عبارت توهین‌آمیز (فارسی) ذخیره شده است.

٤-١-٤- ساخت مجموعه داده

در این مرحله با توجه به داده‌های جمع‌آوری شده، مجموعه داده مورد نیاز برای یادگیری آماده می‌شود. در این مرحله داده‌های موجود در فایل splitted گردآوری شده و به صورت یک متن با عنوان text در می‌آیند. سپس text به روش توضیح داده‌شده در بخش قبلی برچسب‌گذاری شده و مجموعه داده را تشکیل می‌دهند.

سپس این متن به taggerهای توضیح داده شده ارسال می‌شود تا برچسب رکیک بودن یا نبودن هر یک از متن‌ها تعیین شود. در صورتی که یکی از هرکدام از ۳ روش (استفاده از text-mining.ir ، استفاده از ابزار better_porfanity و تشخیص کلمات موجود در دیکشنری) موفق به تشخیص رکیک بودن متن شود، داده دارای برچسب "1" و در غیر این صورت "0" خواهد بود.

٤-۲- پیش‌پردازش

پس از اینکه مجموعه داده داده‌های مورد نیاز آماده شد، نیاز است تا داده‌ها پیش‌پردازش شده و برای یادگیری ماشین آماده شوند. همانطور که گفته شد داده‌های موردنظر این مقاله از میان جستجوهای انجام شده در یک موتور جستجو (پارسی جو) گزینش شده‌اند که برای افزایش محتوا و اطمینان از ارتباط متن با محتوای رکیک ، کوئری‌ها در گوگل جستجو شده و اطلاعات بیشتری استخراج شده است. هرچه داده‌های بیشتری به این مجموعه داده اضافه شود، یادگیری دقیقتر خواهد بود و نتیجه حاصله اطمینان بخش‌تر می‌شود بنابراین می‌توان مجموعه داده را بسط داد و داده‌های بیشتری به آن اضافه نمود.

از اقدامات انجام شده می‌توان به حذف داده‌های null و تبدیل حروف بزرگ به کوچک (در کلمات انگلیسی) اشاره کرد. همچنین در ادامه، باقی پیش‌پردازش‌ها که نیاز به توضیح دارند شرح داده شده است.

٤-۲-۱- نرمال‌سازی کلمات فارسی

در نگارش کلمات فارسی این امکان وجود دارد که برخی کلمات دارای املاهای مختلف باشند، برای مثال کلمه «می‌شود» را می‌توان به شکل‌های روبرو نیز نگارش کرد: «میشود، می شود»

به دلیل اینکه تمامی این کلمات یک معنی دارند، نیاز است تا همه کلمات این چنینی را به یک تحریر درآوریم تا در پردازش‌های بعدی هرکدام به صورت کلمات جداگانه تشخیص داده نشوند. ابزارهای مختلفی برای این عمل موجود هستند که در این جا از Normalizer() متعلق به کتابخانه hazm استفاده می‌شود.

٤-۲-۲- ساخت توکن

در این مرحله نیاز است که هر سطر از ورودی به کلمات سازنده تقسیم شده و به توکن‌ها تبدیل شود. این کار توسط یک Word Tokenizer از کتابخانه nltk انجام می‌شود و متن را با توجه به کلمات موجود در آن به صورت توکن‌هایی در می آورد که هر کدام نشان‌دهنده یک کلمه باشند.

٤-۲-۳- عنوان‌سازی و ریشه‌یابی و حذف کلمات توقف فارسی و انگلیسی (StopWords)

در هر زبانی برخی کلمات توقف وجود دارند که در جمله تاثیر مستقیمی در معنی مورد نظر ندارند. به علاوه اینکه جملات هر سطر در مرحله قبل به توکن‌ها تبدیل شده و هر توکن نشان‌دهنده یک کلمه است به همین دلیل می‌توانیم کلمات توقف انگلیسی و فارسی را حذف کرده تا از بار پردازشی بیهوده جلوگیری کنیم.

ابتدا نیاز است لیستی از کلمات توقف فارسی (بدست آمده از [14]) به تعداد ۱۳۱۶ کلمه را به لیست کلمات توقف انگلیسی موجود در کتابخانه nltk اضافه کنیم. سپس با گذشتن از توکن درصورتی که توکن مورد نظر یک کلمه توقف بود آن توکن حذف خواهد شد. نمونه کلمات توقف فارسی را در شکل (۳) مشاهده می‌کنید.

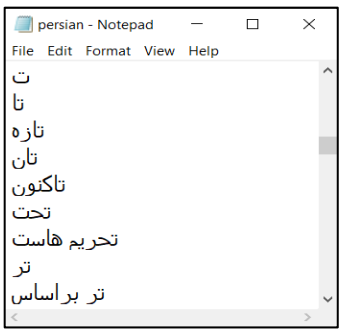
مفهوم بعدی ریشه‌یابی (Stemming) می‌باشد که در پیاده‌سازی مقاله بر روی کلمات فارسی اعمال شده است. Stemming به معنی ریشه‌یابی لغات است که با حذف پیشوند و یا پسوندهای کلمات مختلف آنها را به ریشه (stem) خود می‌رساند. نکته قابل توجه در مورد ریشه‌یابی این است که تنها پیشوند و پسوند را از کلمه حذف می‌کند و کلمه را بدون تغییر در نظر می‌گیرد. بنابراین این احتمال وجود دارد که کلمه ریشه‌یابی شده به کلمه اصلی خود بازگردانی نشود که البته با توجه به قواعد ساخت کلمات در زبان فارسی این احتمال محدود است.

به عنوان مثال کلمه «بانوان» پس از ریشه‌یابی به «بانو» تبدیل خواهد شد. در صورتی که کلمه‌ای نیازی به ریشه‌یابی نداشته باشد به همان شکل باقی می‌ماند.

مفهوم دیگر Lemmatizing یا عنوان‌سازی است که مانند ریشه‌یابی عمل می‌کند با این تفاوت که بجای اینکه کلمه را به ریشه مورد نظر برگرداند، آن را به ریشه اصلی تبدیل می‌کند. با توجه به ویژگی‌های زبان انگلیسی، این ابزار را برای کلمات انگلیسی موجود در متن به کار می‌بریم. به عنوان مثال، کلمه Studies پس از Stemming به studi، و پس از Lemmatizing به study تبدیل می‌شود.

٤-۳- پردازش TF-IDF

با توجه به اینکه داده‌های متنی معنایی برای یک شبکه عصبی ندارند، نیاز است آنها را به اعدادی قابل درک برای سیستم تبدیل کنیم. TF-IDF روشی در پردازش زبان طبیعی است که با توجه به تکرار هر یک از کلمات



شکل (۳): بخشی از کلمات توقف فارسی

در تمام متن‌ها، به هریک از کلمات عددی را اختصاص می‌دهد [6]. روش فراوانی وزنی کلمات با توجه به مدل‌های SVM، Naïve Bayes و KNN استفاده شده در مقاله کاربرد مناسبی دارد.

روش کار این الگوریتم به‌کار بردن فرمول‌های (۱)، (۲) و (۳) به ازای هر کلمه و اختصاص عدد خروجی به آنها است:

$$tfidf(t,d,D) = tf(t,d).idf(t,D) \qquad (۱)$$

به صورتی که:

$$tf(t,d) = \log(1 + freq(t,d) \qquad (۲)$$

$$idf(t,D) = log\frac{N}{count(d \in D: t \in d)} \qquad (۳)$$

Term frequency (tf) : تعداد تکرار هر کلمه در یک document (فرمول (۲)).

Inverse document frequenvy (idf) : نشان‌دهنده اینکه یک کلمه در یک document چقدر معمول و یا نادر است. هرچه عدد به صفر نزدیک‌تر باشد، کلمه معمول‌تر و هرچه به ۱ نزدیک‌تر باشد کلمه کمیاب‌تر است. در فرمول مورد نظر N تعداد تمام کلمات و count تعداد تکرار کلمه مورد نظر است (فرمول (۳)).

٤-٤- یادگیری ماشین

با توجه به اینکه طریقه ایجاد مجموعه داده و عملیات پیش‌پردازش مجموعه داده توضیح داده شد، به چگونگی انجام عملیات یادگیری ماشین می‌پردازیم. در این مقاله از سه مدل SVM، Naïve Bayes و KNN استفاده شده است که در ادامه توضیحات لازم داده می‌شود [7].

٤-٤-١- یادگیری Naïve Bayes

دسته‌بندی کننده بیز ساده یک مدل یادگیری ماشین بر اساس احتمال وقوع می‌باشد که از قضیه بیز استفاده می‌کند. با توجه به اینکه در بخش (۳-۴) با استفاده از TF-IDF داده‌ها تبدیل به اعداد احتمال وقوع آنها شد، به راحتی می‌توان از این مدل استفاده نمود.

٤-٤-٢- یادگیری SVM

ماشین بردار پشتیبانی (Support vector machines - SVMs) مدلی برای طبقه بندی و تشخیص الگو می‌باشد که از ماتریس الگو استفاده می‌کند. یکی از مزایای این مدل یادگیری ساده و سریع است که با توجه به حجم پردازش مورد توجه قرار می‌گیرد. این مدل به منظور یادگیری داده‌های پیچیده ابتدا داده‌ها را با تابع phi محاسبه کرده و به یک تابع هسته چندجمله ای و یا سیگموید می‌رساند. سپس داده‌ها را یادگیری کرده و طبقه‌بندی می‌کند.

٤-٤-٣- یادگیری KNN

الگوریتم K نزدیک‌ترین همسایه یکی از معروف ترین الگوریتم‌های دسته‌بندی است که براساس فاصله K عنصر نزدیک به یکدیگر عمل می‌کند و الگوریتم محاسبه فاصله و مقدار K می‌تواند در نتیجه آن مؤثر باشد. در این مقاله K=3 در نظر گرفته شده است.

با توجه به اینکه یادگیری داده‌ها عملی زمان بر است و هزینه پردازشی زیادی در پی دارد این امکان وجود دارد که پس از اینکه هر یک از مدل‌ها یادگیری شد، از مدل یادگیری شده یک خروجی گرفته شود و در دفعات بعدی بارگزاری و استفاده شود.

٥- نتایج پیاده‌سازی

٥-١- معیارهای اندازه‌گیری

به منظور آزمایش و مقایسه نتایج بدست‌آمده از پیاده‌سازی، از معیارهای استاندارد اندازه‌گیری در روش‌های دسته‌بندی متون [8,9,10] استفاده شده است. این اندازه‌گیری‌ها شامل صحت (Precision)، پوشش (Recall) و F-score می‌باشد. Precision درصد متن‌هایی است که توهین‌آمیز تشخیص داده شده‌اند. Recall نشان‌دهنده صحت کلی دسته‌بندی می‌باشد. به عبارتی این معیار نشان‌دهنده تعداد واقعی متن‌های توهین‌آمیز که به درستی تشخیص داده شده‌اند است. از معیارهای دیگری که مورد استفاده قرار گرفته‌اند، FP (False Positive) و FN (False Negative) را می‌توان در نظر گرفت.FP نرخ متن‌هایی است که به اشتباه توهین‌آمیز تشخیص داده شده‌اند و FN نرخ متن‌هایی است که توهین‌آمیز بوده‌اند ولی تشخیص داده نشده‌اند. همچنین F-score میانگین وزن دار Precision و Recall است که به صورت فرمول (۴) تعریف می‌شود.

$$\mathrm{F-score} = \frac{2\times(precision \times recall)}{precision+recall} \qquad (۴)$$

٥-٢- مجموعه داده

مجموعه داده بدست‌آمده با توجه به توضیحات داده شده در بخش‌های قبل ۲۲۹۴ داده متنی شامل عبارت جستجو، لینک‌های جستجو شده، عنوان لینک‌های بدست‌آمده و توضیحات لینک‌های بدست‌آمده در جستجوی عبارت مورد نظر در موتور جستجوی گوگل می‌باشد که با استفاده از توضیحات موجود در بخش (۴) پیش‌پردازش و برچسب گذاری شده‌اند. از این تعداد داده ۱۰۹۴ داده توهین‌آمیز و ۱۲۰۰ داده طبیعی هستند.

۵-۳- یادگیری و تست

پس از اینکه مجموعه داده مورد نظر تهیه شد، نوبت به یادگیری مجموعه داده توسط مدل‌های توضیح داده شده است. ۲۰ درصد مجموعه داده برای انجام تست و ۸۰ درصد برای انجام یادگیری مورد استفاده قرار می‌گیرد. نتایج بدست‌آمده در سه مدل یاد شده به شرح جدول (۲) است (اعداد نشان‌دهنده درصد هستند).

با توجه به نتایج بدست‌آمده می‌توان در چند بخش مدل ها را با یکدیگر مقایسه نمود. هرکدام از معیارهای مورد آزمایش نشان‌دهنده نقاط مؤثری از مدل نهایی بدست‌آمده هستند. همانطور که پیش تر توضیح داده شد معیار صحت (Precision) نشانگر این است که درصد تشخیص‌های توهین‌آمیز درست (TP) مدل یادگیری شده نسبت به کل متن‌هایی که توهین‌آمیز تشخیص داده شده، شامل تشخیص درست و غلط (FP) چقدر است (فرمول (۵)).

$$recall = \frac{TP}{TP+FN} \qquad (۵)$$

با مقایسه این معیار در دسته‌بندی‌های انجام شده در سه مدل SVM، Naïve Bayes و KNN می‌توان دید که مدل SVM با ۹۷.۲۸٪ دارای Precision بهتری نسبت به دو مدل دیگر است. به این معنا که این مدل متن‌های بیشتری را به درستی توهین‌آمیز تشخیص داده است. از طرفی با بررسی معیار پوشش (Recall)، که نشان‌دهنده نسبت تعداد متن‌های توهین‌آمیز که به درستی تشخیص داده شده‌اند (TP) به مجموع TP و تعداد متن‌هایی که به غلط طبیعی تشخیص داده شده‌اند (FN) است (فرمول (۶))، می‌توان دید که مدل KNN با ۹۴.۴۶٪ پوشش بهتری نسبت به باقی مدل‌ها دارد. تفاوت این دو معیار زمانی قابل لمس خواهد بود که تمرکز هرکدام از این معیارها در نظر گرفته شود. در معیار صحت می‌توان دید که میزان تشخیص متن‌های توهین‌آمیز بر اساس خطاهایی که در تشخیص متون توهین‌آمیز به وجود آمده‌اند در نظر گرفته می‌شود؛ به عنوان مثال در مدل SVM در صورتی که متنی توهین‌آمیز تشخیص داده شده باشد، به احتمال ۹۷.۲۸٪ این تشخیص درست بوده است و به تبعات این موضوع که اگر این تشخیص کاملاً غلط باشد توجهی ندارد. در صورتی که در معیار پوشش، غلط بودن تشخیص یک متن به عنوان یک متن غیر توهین‌آمیز(FN) به عنوان یک پارامتر مهم در نظر گرفته می‌شود. در این صورت زمانی که تشخیص غلط یک متن غیر توهین‌آمیز (FN) فاکتوری مؤثرتر از تشخیص غلط یک متن توهین‌آمیز (FP) باشد، باید معیار Recall درنظر گرفته شود.

با توجه به توضیحات داده شده، می‌توان معیار دیگری را بررسی نمود که معیار f-score نام دارد. اگر به فرمول (۴) نگاهی داشته باشیم، می‌بینیم که f-score تابعی برحسب Precision و Recall است و به عنوان میانگین وزن‌دار این دو پارامتر درنظر گرفته می‌شود. در صورتی که تعادلی از هردو معیار صحت و پوشش مدنظر باشد، می‌توان از f-score استفاده نمود. این پارامتر در زمانی قابل استفاده است که تفاوت False Positive(FP) و False Negative(FN) زیاد باشد و همچنین تعداد متن‌های طبیعی در مجموعه داده بیشتر از متن‌های توهین‌آمیز باشد. با توجه به جدول (۲) می‌توان دید که از لحاظ معیار f-score مدل Naïve Bayes با ۹۴.۷۰٪ بهترین نتیجه را ارائه می‌کند.

با توجه به نتایج بدست‌آمده و توضیحات داده شده می‌توان مشاهده کرد که انتخاب بهترین مدل، با توجه به طراحی انجام شده در این مقاله، وابسته به نحوه تصمیم گیری در مورد داده‌های ورودی می‌باشد به این معنی که هروقت صحت (Precision) دردستور کار باشد مدل SVM، زمانی که پوشش (Recall) مهم باشد KNN و زمانی که میانگین هر دو معیار قبلی یعنی f-score مورد توجه باشد مدل Naïve Bayes مؤثر عمل خواهند کرد. در جدول (۲) می‌توان دو معیار fn rate و fp rate را نیز مشاهده کرده که به ترتیب نرخ تشخیص غلط متن غیر توهین‌آمیز و نرخ تشخیص غلط متن توهین‌آمیز هستند. هر چه این دو معیار مقدار کمتری داشته باشند، مدل مورد نظر دقیق‌تر عمل می‌کند. در صورتی که معیار صحت را برای انتخاب مدل در نظر بگیریم fp rate و در صورتی که معیار پوشش را در نظر بگیریم fn rate مورد توجه قرار می‌گیرد. در ادامه به مقایسه نتایج بدست‌آمده در این مقاله با دیگر مقالاتی که مسیر تحقیقاتی مشابهی با این کار داشته‌اند می‌پردازیم.

۵-٤- مقایسه نتایج

در این بخش نتایج بدست‌آمده در این بخش (۵) را با نتایج موجود در مقالات دیگر مقایسه می‌کنیم. یکی از کارهای قابل بررسی توسط کومار و همکارانش [6] انجام شده است که سه مدل FastText(FT)، google universal sentence encoder (GVE) و dynamic mode decomposition (DMD) را به همراه Random Kitchen Sink (RKS) پیاده‌سازی کرده است. این مدل ها برای تشخیص توئیت‌های رکیک مورد استفاده قرار گرفته‌اند. در کار دیگری که توسط چن و همکارانش [1] انجام شده است، یک مدل مبتنی بر ویژگی‌های نحوی واژگان (LSF) طراحی شده است که در دو فاز بررسی محتوای رکیک و یافتن کاربری که این محتوا را به کار می‌برد پیاده‌سازی شده است. این مدل برروی مجموعه داده‌ای از نظرات چند ویدیو برتر یوتیوب با موضوعات مختلف اعمال شده است و نتایج فاز اول این کار را با این مقاله مقایسه می‌کنیم. جدول (۳) نشان دهنده مقایسه انجام شده است (اعداد برحسب درصد هستند).

در این مقاله نتایج مدل‌های Bag of Words (BoW)، N-Gram و Appraisal، که روش‌هایی برای تشخیص محتوای توهین‌آمیز (بدون به‌کارگیری یادگیری ماشین) هستند، نیز آورده شده که به منظور مقایسه از این نتایج استفاده می‌کنیم.

جدول (۲): نتایج بدست‌آمده از یادگیری ماشین

Measurement	SVM	Naïve Bayes	KNN
Precision	97.28	94.05	86.48
Recall	92.61	93.35	94.46
fp rate	1.21	2.78	6.96
fn rate	3.48	3.13	2.61
f-score	93.89	94.70	90.29

جدول (۳): مقایسه نتایج

Model	*Precision*	*Recall*	*f-score*
SVM	*97.28*	*92.61*	*93.89*
Naïve Bayes	*94.05*	*93.35*	*94.70*
KNN	*86.48*	*94.46*	*90.29*
GEV-SVM Linear[6]	*81.13*	*72.63*	*75.1*
GEV-NB[6]	*68.44*	*69.58*	*68.92*
GEV-LR[6]	*81.71*	*72.45*	*75.4*
FT-SVM Linear[6]	*82.99*	*68.29*	*70.94*
FT-NB[6]	*58.4*	*59.78*	*51.89*
FT-LR[6]	*82.99*	*68.29*	*70.94*
GEV+RKS1000[6]	*90.36*	*74.17*	*81.46*
FT+RKS1000[6]	***99.58***	***98.75***	***99.16***
FT-SVM RBF[6]	*79.88*	*54.68*	*51.38*
GEV-SVM RBF[6]	*36.05*	*50*	*41.89*
BoW[1]	*90*	*68*	*76*
2-Gram[1]	*94*	*33*	*50*
3-Gram[1]	*93*	*46*	*63*
5-Gram[1]	*92*	*63*	*73*
Appraisal[1]	*98*	*67*	*79*
LSF[1]	*97*	*91*	*93*

با توجه به نتایج جدول (۳) می‌بینیم که بهترین عملکرد مرتبط با مدل FT+RKS1000 (مدل FastText با RKS با عمق ۱۰۰۰) می‌باشد که در تمام معیارها برتری دارد.

لازم به ذکر است LR به معنی Logistic Regression و NB معادل Naïve Bayes می‌باشند. همچنین سه نتیجه اول مربوط به این کار و باقی مربوط به منابع ذکر شده هستند.

٦- نتیجه گیری

با توجه به اهمیت موضوع تشخیص متون توهین‌آمیز و رکیک در عرصه‌های مختلف، لزوم استفاده از روش‌های بروز و معتبر در این حیطه از الزامات پیشرفت صحیح به ویژه در شبکه‌های اجتماعی به‌حساب می‌آید. در این مقاله با ارائه روشی جدید و بررسی عملکرد این روش با مدل‌ها و نتایج دیگر دیدیم که می‌توان با اطمینان قابل قبولی در کنار برترین راهکارهای موجود از این روش استفاده نمود. به ویژه زمانی که استفاده از عبارات مورد جستجو در شبکه جهانی اینترنت مدنظر باشد. چرا که ادبیات به کار رفته در زمان جستجو می‌تواند با داده‌های متنی دیگر متفاوت باشد. در این مقاله با توجه به نتایج بدست‌آمده می‌توان مشاهده نمود که مدل‌های یادگیری شده کارایی مناسبی در کنار کارهای برتر انجام شده تابه‌حال دارند و از میان سه مدل یادگیری شده در این کار می‌توان مدل SVM را به عنوان برترین مدل معرفی نمود چرا که بالاترین صحت (Precision)، پوشش (Recall) قابل قبول و درصد پایین fn rate و fp rate را دارا می‌باشد. همچنین زمانی که معیارهای دیگر مدنظر باشد، می‌توان مدل Naïve Bayes را نیز هم‌رده با SVM در نظر گرفت. با این حال در میان مقایسه‌های انجام شده برترین مدل FT+RKS1000 (مدل FastText با RKS با عمق ۱۰۰۰) می‌باشد.

مراجع

[1] Y. Chen, Y. Zhou, S. Zhu and H. Xu, "Detecting Offensive Language in Social Media to Protect Adolescent Online Safety," *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, Amsterdam, Netherlands, 2012, pp. 71-80, doi: 10.1109/SocialCom-PASSAT.2012.55.

[2] Yin, Dawei & Xue, Zhenzhen & Hong, Liangjie & Davison, Brian & Edwards, April & Edwards, Lynne. (2009). Detection of harassment on Web 2.0.

[3] Schmidt, Anna & Wiegand, Michael. (2017). A Survey on Hate Speech Detection using Natural Language Processing. 1-10. 10.18653/v1/W17-1101.

[4] Wiegand, M., Siegel, M., & Ruppenhofer, J. (2018). Overview of the germeval 2018 shared task on the identification of offensive language.

[5] Kumar, R., Ojha, A. K., Malmasi, S., & Zampieri, M. (2018, August). Benchmarking aggression identification in social media. In Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018) (pp. 1-11).

[6] Mt, Vyshnav & Kumar S, Sachin & Kp, Soman. (2020). Offensive Language Detection: A Comparative Analysis.

[7] Agarwal B., Mittal N. (2014) Text Classification Using Machine Learning Methods-A Survey. In: Babu B. et al. (eds) Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28-30, 2012. Advances in Intelligent Systems and Computing, vol 236. Springer, New Delhi.

[8] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," In EMNLP'02: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, pp. 79-86, 2002.

[9] P. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of

reviews," In Proceedings of the Association for Computational Linguistics (ACL), pp. 417-424, 2002.
[10] Q. Ye, W. Shi, and Y. Li, "Sentiment classification for movie reviews in Chinese by improved semantic oriented approach," in HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on System Sciences, 2006, pp. 53b-53b
[11] https://pypi.org/project/better-profanity/ accessed at 2021.03.22
[12] text-mining.ir accessed at 2021.03.22
[13] parsijoo.ir accessed at 2021.03.20
[14] https://github.com/kharazi/persian-stopwords accessed at 2021.03.20
[15] PITENIS, Zeses; ZAMPIERI, Marcos; RANASINGHE, Tharindu, "Offensive language identification in greek." *arXiv preprint arXiv:2003.07459*, 2020.
[16] Mubarak, H., Kareem Darwish and Walid Magdy. "Abusive Language Detection on Arabic Social Media." *ALW@ACL* , 2017.
[17] Çöltekin, Çağrı. "A Corpus of Turkish Offensive Language on Social Media." *LREC* , 2020.
[18] Safaya, Ali, Moutasem Abdullatif and Deniz Yuret. "KUISAIL at SemEval-2020 Task 12: BERT-CNN for Offensive Speech Identification in Social Media." *ArXiv* abs/2007.13184 , 2020.
[19] Razavi, A. H., D. Inkpen, Sasha Uritsky and S. Matwin. "Offensive Language Detection Using Multi-level Classification." *Canadian Conference on AI* , 2010.
[20] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated Hate Speech Detection and the Problem of Offensive Language", *ICWSM*, vol. 11, no. 1, 2017.