Opinion mining

# سطوح تحلیل

در زمان توسعه­ی Opinion mining در ابتدا تحلیل متون تنها در سطح داکیومنت انجام می شد و هدف آن پیدا کردن گرایش کلی داکیومنت بود. سپس این فرایند در سطح جمله انجام شد و در نهایت در سطح entity و aspect ها. [1]

سطح داکیومنت: عقیده کاوی در این سطح تحلیل تلاش می کند تا یک داکیومنت را به کلاس های مثبت یا منفی دسته بندی کند. کاربرد این سطح معمولا تحلیلreview (مانند بررسی فیلم، نظرات محصول، توییت و بلاگ ها) است.

وظیفه ی این مرحله استخراج گرایش احساس کلی متن است. و فرض می کند که تنها یک target(هدف عقیده) در یک داکیومنت بررسی شده. [2]

سطح جمله: این سطح معادل مورد قبلی است چون می توان یک جمله را یک داکیومنت کوتاه در نظر گرفت.

وظیفه سطح جمله مشابه سطح داکیومت ، مسئله ی کلاس بندی است اما این سطح بر روی هرجمله­ی داکیومت تمرکز می کند و مشخص می کند که یک جمله کدام یک از گرایش های مثبت منفی را بیان می کند.

سطح ویژگی: این سطح ریزدانه ترین سطحی است که در آن عقیده کاوی انجام می شود. در اینجا، وظیفه تنها پیدا کردن گرایش عقیده نیست بلکه یافتن target (موجودیت ، جنبه یا هردو) نیز هست بنابراین 5تایی عقیده کاملا پیدا می شود.

عقیده کاوی در سطح جمله و در سطح داکیومنت هنگامی که متن مورد بررسی تنها شامل یک موجودیت و aspect(ویژگی) باشد به خوبی عمل می کنند ولی وقتی بیشتر از یک مورد وجود داشته باشد دچار اشتباه می شوند. عقیده کاوی در سطح ویژگی تلاش می کند تا این مشکل را با یافتن هر aspect یا ویژگی که در متن ذکر شده و ارتباط دادن آن با یک عقیده حل کند [1]

برای مثال در عقیده کاوی در سطح ویژگی، برای به دست آوردن پنجتایی عقیده در نظرات مربوط به یک محصول، نام موجودیت به طور صریح مشخص است(برابر نام محصول) و چندین aspect این موجودیت در عقیده ها ذکر شده اند. هدف در این مرحله پیدا کردن target عقیده ها(aspect خاص موجودیت) و گرایش احساس متناظر با آن است.

پیچیدگی مسائل از سطح داکیومنت تا سطح ویژگی افزایش میابد. نبود داده های برچسب دار در سطح های ریزدانه تر آن را پیچیده­تر نیز می کند.

تحلیل هایی که ریز دانه­ترند می توانند تجمیع شده تا سطوح بالاتر را بسازند. برای مثال با عقیده کاوی aspect-based می­توان مقدار متوسط احساسات را در یک جمله برای به دست آورد. [1]

# فرایند عقیده کاوی

1)جمع آوری داده ها

گام اول برای فرایند عقیده کاوی جمع آوری داده ها است و شامل به دست آوردن مجموعه داده هایی است که می خواهیم برای یافتن عقیده ها کاوش کنیم. به طور کلی دو روش برای به دست آوردن داده ها از وب سایت ها وجود دارد:

1- از طریق Application Programming Interface سایت ، مانند API توییتر

2- استفاده از web crawler ها برای به دست آوردن داده ها از سایت های مورد نظر

هردو این روش ها فواید و مشکلاتی دارند و trade-off هایی برای استفاده از هرکدام وجود دارد

با استفاده از روش مبتنی بر API پیاده سازی آسان­تر است. اما این روش ممکن است محدودیت هایی داشته باشد. مثلا سرعت خواندن داده ها توسط کلاینت درAPIهای توییتر محدود است. البته این روش نیاز به وجود API های مورد نظر در آن وب سایت دارد و تمامی وب سایت ها آن ها را فراهم نمی کنند و حتی در صورت وجود آن ممکن است همه ی کاربرد های مورد نیاز ما، موجود نباشد.

پیاده سازی روش های crawler-based سخت تر است اما این روش ها مزیت نامحدود بودن را دارند. [1]

# preprocess

معمولا داده ی خام جمع آوری شده از منابع مختلف پیش از انجام تحلیل کامل نیاز به پیش پردازش دارد. برخی گام های پیش پردازش عبارت است از:

* Tokenization(نشانه گذاری)

برای تبدیل کردن جمله­ها به کلمات و عبارات و نماد ها یا دیگر توکن های معنادار استفاده می شود و در زبان انگلیسی با جداکردن لغات به وسیله ی space به آسانی انجام می شود. [2]

* حذف stopwordها

فرایند حذف کلماتی که برای ساختاردهی زبان استفاده می شوند اما در محتوای آن نقشی ندارند. مانند a, are, the, was و will

* Stemming(ریشه یابی)

پروسه ی تبدیل کردن کلمه به حالت ریشه ای آن با حذف پسوند و پیشوندهای کلمه و تبدیل به شکل اصلی یا ریشه ی آن

برای مثال person, person’s, personify ,personification تبدیل بهperson می شوند.

* parts of speech (POS) tagging (برچسب گذاری نقش کلمه)

این مرحله برای تشخیص نقش کلمات در متن انجام می شود و در عقیده کاوی بسیار کمک کننده است چون کلمات بیان کننده ی عقیده ها معمولا صفت بوده و و target عقیده ها(موجودیت ها و aspectها) معمولا اسم یا ترکیبی از اسامی هستند و این برچسب­ها می توانند برای استفاده به عنوان features برای فرایند machine learning استفاده شوند [1]

نقش کلمات مانند:adjective, noun, verb, adverb and preposition

همه ی فرایندهای گفته شده همیشه ضروری نیستند و باید بر اساس کاربرد عقیده کاوی برای مسئله انتخاب شوند

روش های unsupervised ممکن است به نقش کلمه ها در جمله برای ساختن قوانین وابستگی داشته باشند که بعدا در فرایند اصلی عقیده کاوی استفاده می شود در نتیجه stopword ها حذف نمی شوند [3]

راهبرد های مختلف:

دو روش اساسی برای انجام فرایند اصلی عقیده کاوی وجود دارد .

روش unsupervised lexicon-based :

نام دیگر آن semantic-based است و این روش، تلاش می کند تا بر اساس قوانین وابسته‌ به‌ زبان‌ شناسي‌ گرایش متن را استخراج کند. [1]

این روش بر اساس منابع دانش خارجی عمل می‌کند و برخلاف روش­های supervised شامل مرحله ی آموزش نیست.

گام های این روش به طور کلی به صورت زیر است:

1. استخراج کلمات یا عباراتی که گرایش احساسی متن را بیان می کنند.
2. تعیین گرایش این عبارات استخراج شده.
3. تعیین گرایش متن با اجتماع گرایش تک تک کلمات. [4]

روش supervised learning-based یا machine-learning-based که در آن اطلاعات زیربنایی را از داده های آموزشی یاد می گیرد که به آن امکان کلاس بندی داده های بدون برچسب جدید را می دهد.

این روش معمولا شامل مهندسی feature ها برای نمایش object ی که باید کلاس آن تعیین بشود و سپس استفاده از این نمایش به عنوان ورودی الگوریتم می باشد. برخی از feature هایی که در عقیده کاوی به کار می رود عبارت است از:

تعداد تکرار کلمات، نقش کلمات، کلمات یا عبارات احساسی و...

همچنین تعداد فزاینده ای از مطالعات بر روی ترکیب موفق این دو روش وجود داشته است. [1]

برای تشخیص عقیده، معمولا روش های supervised بهتر از روش های unsupervised عمل می کنند اما کمبود منابع با داده های برچسب دار، تحقیقات را به سمت توسعه ی روش های semisupervised یا unsupervised متمایل کرده است. [2]

الگوریتم ها

یک روش لغت نامه محور

این الگوریتم یک روش لغت نامه محور است و در سطح جمله،گرایش عقیده را تشخیص می دهد. این روش با داشتن موضوع( به عنوان ورودی) و مجموعه ای از متن ها راجع به موضوع، احساسات اظهار شده در مورد موضوع در هر متن را به دست می آورد و افرادی که هر احساس را بیان کرده اند را تشخیص می دهد. احساسات می تواند مثبت یا منفی باشند. در این روش از از کلمات به سمت جمله حرکت می کنیم. هر کلمه را به عنوان واحد نشان دهنده­ی احساسات در نظر گرفته و در ابتدا کلاس هر کلمه را بر اساس احساسات آن دسته بندی می کنیم.

هر جمله ممکن است حتی بیانگر نظرات چند نفر باشد به همین دلیل هنگام ترکیب احساسات در سطح کلمات، ابتدا برای نظرهنده های مختلف بخش مرتبط با او را در جمله مشخص می کنیم و سپس از مدل های مختلف برای ترکیب احساسات کلمات استفاده می کنیم.

با داشتن موضوع و مجموعه ای متن ها سیستم در چهارگام این عملیات را انجام می دهد. اول جمله هایی که شامل عبارت موضوع و کاندیدا های افرادی که نظر را داده اند را انتخاب می کند. سپس مرز ناحیه ی مبتنی بر نظر دهنده را مشخص می کند. پس از آن طبقه بند احساسات کلمات گرایش تک تک کلمات دارای احساسات را محاسبه می کند و در آخر سیستم آن ها را برای تولید احساسات نظر دهنده برای کل جمله را ترکیب می کند.

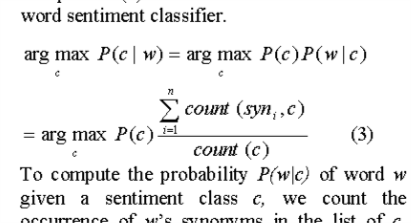
word sentiment classification

در روش اصلی برای به دست آوردن احساسات کلمات ، مجموعه ی کوچکی از کلمات را به صورت دستی جمع آوری می کنیم و بر اساس گرایش به دو دسته تقسیم می کنیم.(مثبت ومنفی) سپس این مجموعه را با اضافه کردن کلمات مترادف و متضاد آن ها از WordNet گسترش می دهیم با فرض اینکه اکثر مترادف های یک کلمه مثبت هم مثبت بوده و متضاد های آن منفی هستند . (در صورتی که کلمه ای بیشتر از یکبار تکرار شد، تعداد آن نیز ذخیره می شود)

WordNet یک مرجع آنلاین لغات است که در آن اسم ها و فعل ها و صفت های انگلیسی در گروه های مترادف قرار دارند. [5]

برخی کلمات مانند great, strong , take چندین با در دسته های مثبت و منفی دید شدند که نشان می دهد نیاز به معیار قدرت احساسات گرایش هر کلمه داریم. این روش به ما امکان این را می دهد که کلماتی که از لحاظ احساسی مبهم هستند را دور انداخته و تنها کلماتی که قدرت آن ها از مقدار آستانه بیشتر است را نگه داریم.

با این روش می توانیم قدرت احساسی کلماتی که تا کنون مشاهده نشده اند را نیز تعیین کنیم و به این صورت عمل می کنیم که با استفاده از WordNet مجموعه کلمات مترادف آن کلمه جدید را تعیین می کنیم و به صورت زیر عمل می کنیم:



که در آن C دسته بندی احساسات است(مثبت و منفی) و w کلمه ی جدید است. synn کلمات مترادف w استخراج شده از WordNet اند.

در این روش برای محاسبه ی قدرت کلمه ی w با داشتن کلاس c (مقدارP(w|c)) ، تعداد رخ داد مترادف های w را در لیست c می شماریم ( ایده اصلی این است که هرچه بیشتر مترادف ها در این کلاس باشند، با احتمال بیشتری این کلمه عضو این کلاس است). برای هرکلمه قدرت مثبت و منفی را محاسبه می کنیم و قدر مطلق آن ها را مقایسه می کنیم.

مثال:

abysmal : NEGATIVE

[+ : 0.3811][- : 0.6188]

adequate : POSITIVE

[+ : 0.9999][- : 0.0484e-11]

afraid : NEGATIVE

[+ : 0.0212e-04][- : 0.9999]

ailing : NEGATIVE

[+ : 0.0467e-8][- : 0.9999]

amusing : POSITIVE

[+ : 0.9999][- : 0.0593e-07]

answerable : POSITIVE

[+ : 0.8655][- : 0.1344]

apprehensible: POSITIVE

[+ : 0.9999][- : 0.0227e-07]

averse : NEGATIVE

[+ : 0.0454e-05][- : 0.9999]

blame : NEGATIVE

[+ : 0.2530][- : 0.7469]

Sentence Sentiment Classifier

ما در جستجوی احساساتی هستیم که نظر دهنده ابراز کرده است. تحلیل ها نشان داده اند که این احساسات به طور قابل اطمینان تری در جمله نزدیک به نظردهنده به دست می آیند. به همین دلیل در این مراحل ، گام هایی برای یافتن موضوع با جستجوی خود کلمه در متن و هر نظردهنده ی ممکن ،اضافه شده است. سپس نزدیک هر نظردهنده، منطقه ای که احساسات آن باید در نظر گرفته شود را مشخص کنیم.

تشخیص نظردهنده

در اینجا از ابزاری برای مشخص کردن Named entity ها برای تشخیص نظردهندگان یک عقیده استفاده شده است و فقط فرد یا سازمان را به عنوان نظردهنده های ممکن در نظر گرفته است. در جمله هایی با بیشتر از یک نظر دهنده، موردی که به عبارت موضوع نزدیکتر بود را انتخاب می کند.

Named entity یک مفهوم در دنیای واقعی است مانند افزاد مکان ها سازمان ها محصولات و..که می تواند با یک نام خاص نشان داده شود وNamed-entity recognition در یک متن بدون ساختارمحل Named entity ها را یافته و آن ها را در کلاس های از قبل مشخص شده، کلاس بندی می کند .

Sentiment Region

به صورت های مختلفی می توان ناحیه ای که احساسات مورد نظر در آن بیان شده را مشخص کرد. برای مثال پنجره های زیر را می توان درنظر گرفت.

پنجره 1: کل جمله

پنجره 2: لغات بین نظردهنده وعبارت موضوع

پنجره 3: همان پنجره2 ± دوکلمه

پنجره 4: همان پنجره 2 تا پایان جمله

مدل های کلاس بندی:

برای انتساب یک کلاس احساسی به جمله ی داده شده می توان از راه های زیر استفاده کرد:

مدل 1:

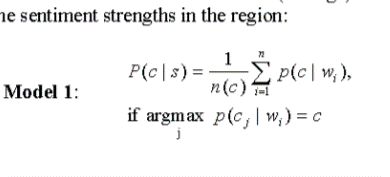
Π (signs in region)

ایده ی این مدل براساس این است که منفی در منفی برابر مثبت است. در این مدل کلماتی مانند not و neverمی تواند گرایش احساسی را وارونه کند. این روش به این دو جمله یک احساس را نسبت می دهد:

"The California Supreme Court ***agreed*** that the state’s new term-limit law was ***constitutional***”

“the California Supreme Court ***disagreed*** that the state’s new term-limit law was ***unconstitutional***”

مدل دوم میانگین قدرت احساسات در جمله است



در این مدل n( c) تعداد کلمات در ناحیه است که دسته بندی احساسی آن برابر c است.در صورتی که جمله ای دارای کلمات مثبت قوی­تری باشد ، جمله مثبت خواهد بود. [5]

مثال:

For that reason and others, the Constitutional

Convention unanimously rejected term limits

and the ***First Congress*** soundly defeated two

subsequent ***term-limit*** proposals.

TOPIC : term limit

HOLDER : First Congress

OPINION REGION: soundly, defeated ,two,subsequent, term-limit,proposals.

SENTIMENT\_POLARITY: negative

Naive Bayes Classifier

Naive Bayes یکی از ساده ترین و معمول ترین classifierها برای عقیده کاوی است و روشی supervised بوده که در سطح داکیومنت گرایش را تشخیص می دهد. این مدل احتمال پسین هر کلاس(مثبت یا منفی بودن عقیده) را بر اساس پراکندگی کلمات در داکیومنت به دست می آورد و براساس روش استخراج ویژگی bag of words کار می کند.

مدل bag of words روشی برای نمایش است که در پردازش زبان طبیعی استفاده می شود که در آن متن به صورت مجموعه ای از کلمات بدون ترتیب نمایش داده می شود و قوانین گرامری و ترتیب کلمات را نادیده می گیرد.

کلاس داکیومت برابر برچسب مربوط به بیشترین احتمال زیر است.

P(label|document)=

با فرض مستقل بودن رخ دادن کلمات(ویژگی ها) از یکدیگر، عبارت بالا به صورت زیر تبدیل می شود:

P(label|document)=

P(document) تاثیری در انتخاب label ندارد پس می توان آن را حذف کرد. [6]

مرحله ی آموزش محاسبه ی P(label) :

P(label)=

P(fi|label)=

در محاسبه ی P(fi|label) برای کلماتی که تا کنون مشاهده نشده اند مقدار 0 به دست می آید که باعث می شود محاسبات با مشکل مواجه شود پس برای رفع این مشکل از فرمول زیر برای محاسبه ی این احتمال استفاده می کنیم. با این راه (با نام add-one smoothing ) به کلماتی که تا کنون مشاهده نشده اند احتمالی اختصاص داده می شود و جمع احتمال همه ی کلمات در کلمات یک برچسب همچنان یک باقی خواهد ماند. [7]

روش محاسبه ی P(fi|label) با استفاده از به شکل زیر خواهد بود:

P(fi|label)=

مثال

برای آموزش سیستم داکیومنت هایی مانند نظرات راجع به محصولات (در این مثال بازی کامپیوتری)که دارای امتیاز نیز هستند انتخاب می شود و امتیاز های بالا(4و5) به برچسب مثبت و امتیاز های پایین(1و2) به برچسب منفی تبدیل شد Review ها با امتیاز 3 به عنوان نمونه های خنثی حذف شد.

برای آماده سازی document ها علامت NOT\_ به کلماتی که بین عبارت منفی کننده مانند not, isn’t , didn’t و اولین علامت نگارشی پس از عبارت منفی کننده، اضافه شد سپس tokenization انجام شد.

مثال اعمال منفی کننده در جمله:

This place is not OK.

Features={ This ,place, NOT\_OK)

برای حفظ سادگی این راه و بررسی تاثیر مستقیم Naïve Bayes ، از موارد پیش پردازش دیگر مانند بردن به حالت ریشه ای و حذف برخی کلمات خودداری شده است.

برای مرحله ی آموزش، جملات زیر را بررسی می کنیم:

1. This is a fun game 4
2. This game is wonderful 5
3. started crashing for no reason. 2
4. I have to restart everytime before new game. 1

برای هر جمله، در ابتدا امتیاز هر کدام تبدیل به برچسب مثبت و منفی شد وبه جملات 1و2 برچسب مثبت وبه جملات 3و4 برچسب منفی داده شد.

بعد از انجام tokenization و اعمال منفی کننده ها( مانند NO\_reason ) به کلمات جملات مثبت برچسب مثبت زده و به کلمات جملات منفی برچسب منفی می زنیم. حالا با استفاده از اطلاعات به دست آمده جدول زیر را ایجاد می کنیم که در هر سطر تعداد دفعات تکرار برچسب آن ستون برای آن کلمه ذخیره شده است.

Table 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | This | Is | A | Fun | game | wonderful | Started | crashing | for | NO\_reason | I | Have | to | restart | everyTime | Before | new |  |
| positive | 2 | 2 | 1 | 1 | 2 | 1 |  |  |  |  |  |  |  |  |  |  |  | 9 |
| negative |  |  |  |  | 1 |  | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 12 |

برای مشاهده ی روند کلی این فرایند گرایش جمله ی زیر را بررسی می کنیم

Document= Everytime I have started the game it is crashing.

بر اساس تعداد بار تکرار کلمات در جدول شماره 1 ، برای جمله تست ، جدول زیر را تشکیل می دهیم:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Everytime | I | have | started | the | game | it | is | crashing |
| positive | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 2 | 0 |
| negative | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

مقدار عبارت P(label|document) را برای برچسب های مثبت و منفی محاسبه می کنیم.

P(pos)= =

P(Everytime |document)= =

P(game|document)= =

P(positive|document)= = =.0.004

P(neg)= =

P(the|neg)= =

P(Everytime | neg)= =

P(neg|document)= ==0.038

به دلیل اینکه مقدار عبارت برای برچسب منفی بیشتر است، برچسب جمله ی تست بر اساس کلاسیفایر naïve bayes منفی است.

# References

|  |  |
| --- | --- |
| [1] | Balazs, Jorge A. , Juan D. Velásquez. 2016. “Opinion Mining and Information Fusion: A survey.” Information Fusion 27:95–110. |
| [2] | Sun, Shiliang, Chen Luo, و Junyu Chen. 2017. “A review of natural language processing techniques for opinion mining systems.” Information Fusion 36:10–25. |
| [3] | -Ravi, Kumar و Vadlamani Ravi. 2015. “A survey on opinion mining and sentiment analysis: Tasks, approaches and applications.” Knowledge-Based Systems 89:14–46. |
| [4] | "Ontology-Supported Polarity Mining". |
| [5] | "S.-M. Kim , E. Hovy , Determining the sentiment of opinions, in: Proceed- ings of the 20th". |
| [6] | "W. Medhat et al., Sentiment analysis algorithms and applications: a survey,". |
| [7] | "B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? Sentiment classification using machine learning techniques, Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, vol. 10, Association for Computational Linguistics, 2002, p". |
| [8] | "Miller, G.A., R. Beckwith, C. Fellbaum, D.Gross, and K. Miller. 1993. Introduction to WordNet: An On-Line Lexical Database. http://www.cosgi.princeton.edu/~wn.". |