Sentimentanalys av Tweets

DD1418 - Grupp 6

Emil Bergqvist
emib@kth.se
Oscar Danielsson
oscdan@kth.se

Introduktion

Twitter

Twitter är en social nätverkstjänst, mer specifikt en mikroblogg där användare skriver meddelanden, eller tweets, som begränsas av en 280-teckengräns. Det finns flera olika sorters tweets, bland annat, fristående tweets, svar, retweets och citattweets[1]. Denna rapport kommer behandla tweets och svar på tweets. Alla svar innehåller mentions där det konto som användaren svarar på nämns med ett snabel-a följt av användarnamnet. Utöver mentions tillkommer löpande text som utgör själva meddelandet och i vissa fall även hashtags som för användare är klickbara och fungerar som ett sätt att navigera bland tweets med olika teman.

Sentimentanalys

Sentimentanalyser handlar om att avgöra vilken värdering eller stämning en text uttrycker[2]. I detta fall tillämpas sentimentanalysen på tweets och det kommer göras med hjälp av maskininlärningsmetoder. Vad gäller vilka sentiment som uttrycks i tweetsen begränsas det till tre, en tweet kan vara positiv, negativ eller neutral.

Underlag

Datamängd

Det dataset som valdes bestod av tweets riktade mot flygbolag. Träningsmängden bestod av 12000 tweets, valideringmängden 1200 och testmängden 1600 tweets. Varje rad består först av en klassificering, följt av den faktiska tweeten. I den ostädade datan inleds varje tweet med en mention vilket indikeras av ett "@" följt av flygbolagets namn. Några exempel på de tweets som datasetet bestod av innan rensningen följer.

negative	@VirginAmerica you guys messed up my seating.. I reserved seating with my friends and you guys gave my seat away ... ðŸ˜¡ I want free internet
negative	@VirginAmerica status match program. I applied and it's been three weeks. Called and emailed with no response.
negative	@VirginAmerica What happened 2 ur vegan food options?! At least say on ur site so i know I won't be able 2 eat anything for next 6 hrs #fail
neutral	@VirginAmerica do you miss me? Don't worry we'll be together very soon.
negative	@VirginAmerica amazing to me that we can't get any cold air from the vents. #VX358 #noair #worstflightever #roasted #SFOtoBOS
neutral	@VirginAmerica LAX to EWR - Middle seat on a red eye. Such a noob maneuver. #sendambien #andchexmix
negative	@VirginAmerica hi! I just bked a cool birthday trip with you, but i can't add my elevate no. cause i entered my middle name during Flight Booking Problems ðŸ˜¢
neutral	@VirginAmerica Are the hours of operation for the Club at SFO that are posted online current?
negative	@VirginAmerica help, left expensive headphones on flight 89 IAD to LAX today. Seat 2A. No one answering L&F number at LAX!

[1] Wikipedia, "Twitter", 2021
[2] Gupta, S., "Sentiment Analysis: Concept, Analysis and Applications", 2018

Datan städas sedan så att alla mentions, siffror, och specialtecken försvinner. Detta eftersom dessa tecken inte bidrar nämnvärt till att bedöma enskilda tweets sentiment. Mentions kan även störa våra särdrag då de förekommer i varje tweet. Tweetsen består av interaktioner mellan kunder och flygbolags kundtjänster. Föga förvånande är därför att sentimenten av tweetsen till majoriteten är negativa.

Hypotes

En modell baserad på multinomial logistisk regression bör vara bättre på att klassificera tweets än en modell baserad på multinomial Naive Bayes.

Implementation

Multinomial Naive Bayes

Naive Bayes är en deterministisk textklassificeringsalgoritm. Naive Bayes bygger på att man, med hjälp av Bayes sats, beräknar den betingade sannolikheten för en klass givet en text. Då det enbart är maximum-likelihood skattningen som är intressant så kan man bortse från nämnaren i den betingade sannolikheten. Algoritmen tillämpas som så att varje klass får sitt egna korpus. Detta korpus implementeras ofta med hjälp av en bag-of-word vektor som sparar förekomsten av varje ord som ett index. I denna implementationen har dictionaries används för att reducera tidskomplexiteten vid sökning. När varje korpus är fyllt med de ord som hör till respektive kan sannolikheterna för respektive ord beräknas.

Naive Bayes bygger på antagandet om att varje ord eller varje bigram är oberoende av varandra. Detta kan ses som ett naivt antagande då detta oftast inte är fallet och metoden har därav fått sitt namn. Antagandet om oberoende har en tydlig fördel, sannolikheten för att ett ord tillhör en viss klass är den samma som produkten av sannolikheterna att dess komponenter gör det.

$$arg\,max_{y} = P(w_1 \ldots w_n|y) * P(y) \rightarrow arg\,max_{y} = P(w_1|y) \ldots P(w_n|y) * P(y)$$

De enskilda sannolikheterna skattas sedan med hjälp av en maximum likelihood skattning. I unigram-fallet innebär detta att förekomsten av ordet i den aktuella klassen divideras med det totala antalet ord i klassen. Ett problem med detta tillvägagångssätt uppstår om ett givet ord inte finns i korpuset för en klass. Detta medför att täljaren blir noll och i förlängningen att hela produkten blir noll. Ett enkelt sätt att komma runt detta är att implementera Laplace-smoothing. Vilket implementeras genom att göra ett antagande om att alla ord förekommer minst en gång i varje klass. Detta leder till att täljaren för varje sannolikhet måste adderas med ett och nämnaren med totala antalet unika ord vilket illustreras i nedanstående ekvation.

$$P_{Laplace}(W_i \mid y_i) = \frac{C(W_i)+1}{N+V}$$

Då de enskilda sannolikheter tenderar att vara under ett så medför detta att multiplikation mellan dessa sannolikheter resulterar i en liten produkt. Sannolikheterna logaritmeras därför för att komma runt detta.

Logistisk Regression

Till skillnad från Naive Bayes, som är en generativ modell, är Logistisk regression en diskriminativ modell. Att den är diskriminativ innebär i ett textklassificeringssammanhang att den försöker beräkna, eller rättare sagt estimera sannolikheten att en text tillhör en viss klass. Den skiljer sig ytterligare från Naive Bayes genom att vara stokastisk, vilket innebär att resultatet till viss del beror av slumpfaktorer och i sin förlängning att det inte alltid går att återskapa tidigare resultat. Att den logistiska regressionen är multinomial innebär att den försöker predicera bland fler än två klasser[3].

Komponenterna för en multinomial logistisk regression lyder som följande:

- En särdragsrepresentation, X. För varje datapunkt i X finns en komponent för de olika särdragen. Dessa representeras binärt och vilket värde som ges beror på om särdraget uppfylls eller inte
- Ett facit, Y. Representerar den korrekta klassificeringen för alla datapunkter
- En theta-matris som representerar hur de olika särdragen ska viktas
- En softmaxfunktion som omvandlar produkten mellan särdragsrepresentationen och theta-matrisen till en sannolikhetsfördelning.
- En cross-entropy loss-function, en funktion som anger felet i modellens klassificeringar.
- En algoritm för gradient-descent som justerar vikterna för att minimera förlusten. Förlusten minimeras genom att hitta minimipunkten hos den ovannämnda förlustfunktionen med hjälp av gradienten.

En multinomial logistisk regression genomförs i två stadier.

1. Ett träningsstadie med ett träningsset av data där vikterna justeras
2. Ett evalueringsstadie där modellen bedöms utifrån några bestämda kriterier.

Träningsstadiet tillämpas på två olika sätt genom rapporten. Den ena metoden, Batch gradient descent, kommer, för varje gång gradienten ska beräknas, gå igenom alla datapunkter. Den andra metoden, Stochastic gradient descent, kommer istället slumpa en datapunkt varje gång gradienten ska beräknas. Tidigare nämndes att logistisk regression är stokastisk. Det beror dels på att vikterna till en början slumpas. Stochastic gradient descent introducerar ytterligare ett element av slumpmässighet genom att variera vilka datapunkter som används.

Implementationen predicerar en av tre klasser, positiv, negativ eller neutral och det med hjälp av 12 olika särdrag. Särdagen är följande:

- Positivt ord
- Negativt ord

[3] Jurafsky, D. Martin, J., “Speech and Language Processing”, 3rd edition, 2021.

- Nettosentiment (+ och neutral)
- Längd (antal ord)
- Positiva och negativa superlativ
- Positiva och negativa adverb
- Positiva och negativa adjektiv
- Emojis

Bortsett från Nettosentiment och Längd tillämpas alla särdrag genom slå upp varje tweets enskilda ord i olika korpus. Så länge som minst ett ord finns i respektive korpus kommer motsvarande särdrag uppfyllas. Nettosentiment beräknas genom att subtrahera antalet ord (i tweeten) som återfinns i det negativa korpuset från antalet ord i det positiva. Längden har en brytpunkt på 25 ord. Alla tweets som har fler ord uppfyller särdraget.

Modellen har implementerats i Python med hjälp av biblioteken Pandas och Numpy. Koden består till största del av egna implementationer, speciellt vad gäller gradient descent-algoritmen. Ett fåtal hjälpfunktioner är hämtade annanvart. Softmaxfunktionen är tagen från biblioteket Scipy. Metoderna för plottning av förlustfunktionen är tagen från laboration 5. Evalueringen genomförs med hjälp av Scipy-funktioner för att beräkna confusion matrix samt precision och recall.

Resultat och Evaluering

Samtliga metoder evalueras med hjälp av samma evalueringsmetoder. Evalueringen mellan körningar genomfördes med hjälp av valideringsmängden och inte förrän samtliga förändringar var genomförda kördes testmängden vars resultat evaluerades med hjälp av följande metoder.

Confusion matrix: Kolumnerna representerar de sanna klasserna och raderna de förutspådda. Huvuddiagonalen representerar gångerna då den förutspådda klassen och den verkliga överensstämmer vilket kan betecknas som True-positive. De värden som utgör kolumnerna är de sanna värdena. Recall defineras som den andel från en specifik klass som är rätt klassade. Detta representeras i confusion-matrix för en specifik klass som True-positive värdet genom summan för hela kolumnen.

Actual / Predicted	Neutral	Positive	Negative
Neutral	8	4	213
Positive	1	17	148
Negative	39	41	1141

Recall för Neutral blir i detta fall: $\frac{8}{8+1+39}$

Precision defineras som den andel datapunkter, som klassats till en viss klass, som faktiskt tillhör den klassen. Detta representeras i confusion-matrisen för en specifik klass som true-positive värdet genom summan av hela raden.

Actual / Predicted	Neutral	Positive	Negative
Neutral	8	4	213
Positive	1	17	148
Negative	39	41	1141

Precision för Neutral blir i detta fall: $\frac{8}{8+4+213}$

F-score används även för att få ett mer rättvist medelvärde mellan precision och recall. Ett vanligt medelvärde missar nämligen ifall någon av faktorerna är väldigt låga vilket F-score tar höjd för. F-score defineras som $\frac{2PR}{P+R}$ där P står för precision och R för recall.

De båda regressionerna utvärderas även genom att plotta cross-entropy loss funktionen. Förlustfunktionen ger ett mått på hur nära prediktionen är det verkliga värdet. Genom att köra Förlustfunktionen i takt med att theta-matrisen uppdateras borde modellen bli bättre på att predicera det verkliga värdet. Förlusten borde med andra ord minska i takt med att vår modell blir bättre på att förutspå det faktiska värdet. Minskningen bör vara som störst i början då den euklidiska normen av gradienten är proportionerlig mot lutningen av förlustfunktionen. Förlusten kommer därför i takt med att den euklidiska normen av gradienten minskar att konvergera mot ett värde. Hur snabbt förlusten konvergerar beror både på hur bra modellen är samt hur hög learning-rate som används. För att kunna jämföra Stochastic och Batch konvergerar är learning-rate konstant.

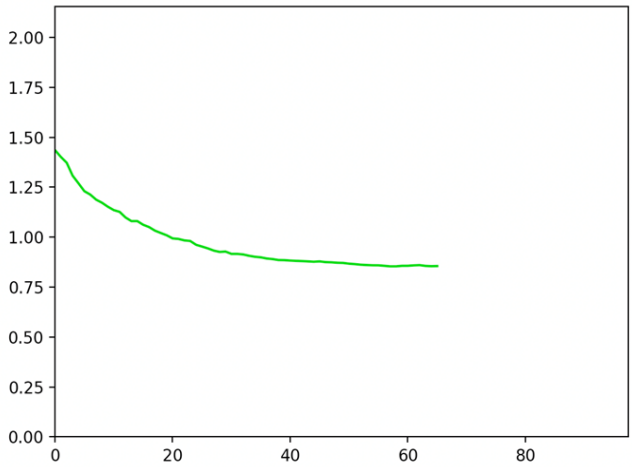
Det sista måttet som används för att bedöma de olika modellerna är accuracy. Accuracy kan tolkas som andelen rätt klassificerade datapunkter och illustreras i confusion-matrisen som summan av huvuddiagonalen genom totala antalet datapunkter.

Actual / Predicted	Neutral	Positive	Negative
Neutral	8	4	213
Positive	1	17	148
Negative	39	41	1141

Logistisk Regression

Stochastic Gradient Descent

Cross-entropy Loss-function



Confusion matrix

Actual / Predicted	Neutral	Positive	Negative
Neutral	124	46	55
Positive	28	102	36
Negative	322	161	738

%	Neutral	Positive	Negative
Precision	26	33	89
Recall	55	61	60
F-score	35	43	72

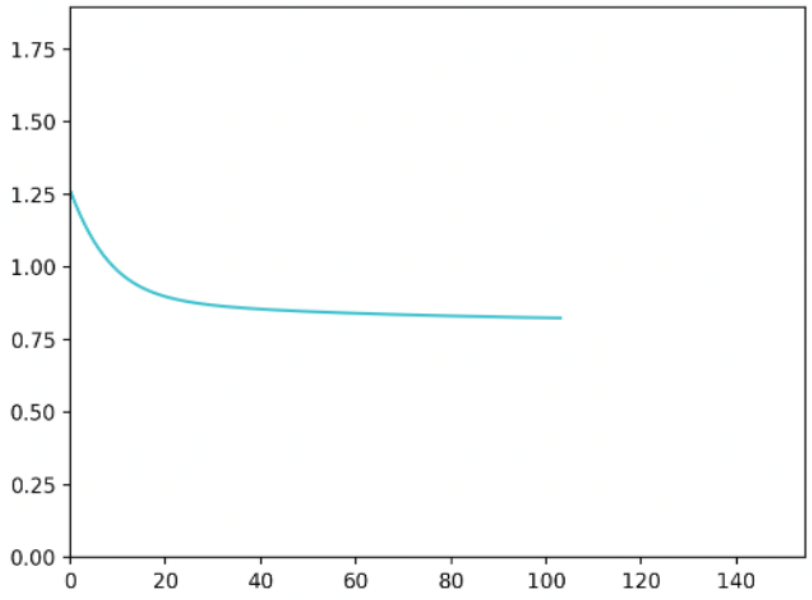
Accuracy : 72%

Vikter

	Deadweight	Neg word	Pos word	Pos sum	Neut sum	Length	Pos Super-latives	Neg Super-latives	Pos Adverb	Neg Adverb	Pos Adj	Neg Adj	Emojis
0	-0.90	-0.39	-0.63	-0.56	0.28	-0.66	0.96	-0.26	0.93	-0.25	0.65	0.32	-0.99
+	-0,80	-1.05	0.45	-0.77	-0.49	-0.94	0.93	-0.72	0.13	0.45	0.59	-0.36	-0.19
-	0.21	0.68	-0.76	0.05	-0.36	-0.67	0.74	-0.01	-0.02	-0.47	0.65	0.46	-0.28

Batch Gradient Descent

Cross-entropy Loss-function



Confusion matrix

Actual / Predicted	Neutral	Positive	Negative
Neutral	8	4	213
Positive	1	17	148
Negative	39	41	1141

%	Neutral	Positive	Negative
Precision	17	27	76
Recall	4	10	93
F-score	6	15	84

Accuracy: 60%

Vikter

	Deadweight	Neg word	Pos word	Pos sum	Neut sum	Length	Pos Super-latives	Neg Super-latives	Pos Adverb	Neg Adverb	Pos Adj	Neg Adj	Emojis
0	-0.19	-0.46	-0.21	-0.12	-0.03	0.69	0.74	0.65	0.63	-0.12	-0.40	0.42	-0.15
+	-0,95	-0.49	1.09	0.25	-0.004	0.18	0.55	0.78	0.53	0.20	-0.70	-0.43	-0.09
-	0.30	1.24	-0.36	0.74	-0.09	0.35	-0.30	-0.28	0.07	0.89	0.07	0.44	-0.61

Naive Bayes

Confusion matrix

Actual / Predicted	Neutral	Positive	Negative
Neutral	66	9	150
Positive	10	96	60
Negative	33	14	1174

%	Neutral	Positive	Negative
Precision	61	81	85
Recall	29	58	96
F-score	40	67	90

Accuracy: 83%

Diskussion

Vikter

Att analysera vikterna kan ge en fingervisning av vilka särdrag som hjälper modellen att särskilja klasser och vilka som inte gör det. Ett bra särdrag är ett sådant där vikterna skiljer sig åt nämnvärt för olika

klasser. För att minimera förlustfunktionen är det önskvärt att sannolikheten för den korrekta klassen maximeras. Till exempel, särdraget för positiva och negativa ord korrelerar mycket väl med respektive klass. En annan intressant aspekt av dessa särdrag är hur korrelationen är negativ för den neutrala klassen. Detta tyder på att de tweets som klassas som neutrala, till stor del, saknar värderande ord. Andra särdrag som uppvisar önskvärt beteende är den för neutralt nettosentiment, alltså att summan av alla värderande ord är noll. Här har den neutrala klassen en positiv korrelation medan både den negativa och positiva klassen har negativa vikter. Resterande särdrag bidrar inte med särskilt mycket för att predicera klasser. Till största delen beror troligtvis detta på att särdragen är specifika och de korpus som används är nämnvärt mindre än de för positiva och negativa ord. Det finns helt enkelt inte tillräckligt många datapunkter för någon klass som uppfyller särdraget och det leder till att vikterna inte är optimerade.

Exempel på klassificeringar

"@AmericanAir bad weather happens but lack of preparation is inexcusable. Depending on good weather is not a business model."

Predicerad: negativ

Faktisk: negativ

"@USAirways you're the only airline that's not open sooner. It's not a confidence booster."

Predicerad: positiv

Faktisk: negativ

Ovanstående tweets är klassificerade av Stochastic gradient descent.

Underlag

Två huvudsakliga brister har identifierats i underlaget. Den ena är att facit också är predicerat av en modell och därmed är alla klassificeringar inte korrekta. Detta ger upphov till skevhet i vår modell i form av att den tränas fel. Den andra bristen, eller snarare svårigheten, är att Twitter som medium, på grund av dess teckenbegränsning, uppmuntrar till informellt språk. Datan innehåller många felstavningar och en hel del slang.

"@VirginAmerica classiq, luv Virgin America. Greetingz"

Ovanstående exempel utgör ett problem för båda modeller, men kanske framförallt för den logistiska regressionen som använder sig av flera korpus för att slå upp ord. Tweeten är utan tvivel positiv men eftersom de positiva värdeorden "love" och "greetings" är inkorrekt stavade kommer inga positiva särdrag att uppfyllas eftersom uppslagningen i korpusen returnerar false.

Modeller

Vad gäller brister i modellerna har fyra problem identifierats. Följande brister gäller för både Naive Bayes och den Logistiska regressionen.

Kontext: Ingen av modellerna fångar kontext särskilt väl. Ett grundläggande antagande i Naive Bayes är att orden är oberoende av varandra. Vad gäller den logistiska regressionen ligger problemet i att positivt laddade ord kan användas i en negativ kontext och vice versa. Se följande exempel:

"@AmericanAir wish you had a better mobile app. You should look at the app from @united as it is much more seamless to check in

Negationer: Ett mer specifikt fall av kontextproblemet. Utgör ett påtagligt problem, en negation kan omvandla ett positivt laddad ord till ett negativt och vice versa. Ingen av modellerna fångar detta särskilt väl.

Sarkasm: Enligt Poe's lag är det svårt, nästintill omöjligt att avgöra om en text är sarkastisk utan en tydlig indikation på att den är det. Eftersom människor har svårt att klassa sarkasm kommer säkerligen en modell ha ännu större problem med det. Varken förekomsten av värdeord eller kontext är tillräckligt för att identifiera sarkasm. Snarare krävs en intuitiv förståelse för språkliga konventioner och framförallt kring kulturella normer om humor. Kort sagt, sarkasm är alldeles för abstrakt för att kunna fångas i de tillämpade modellerna[4].

"@VirginAmerica Waited 2h four a flight that got cancelled, best 1300$ I've ever spent!!"

Skev fördelning av data: Ett problem som främst berör båda modellenaa.. Eftersom träningsdatan var snävt fördelat, med uppemot 70% av tweetsen klassificerade som negativa, hade den logistiska regressionen en tendens att också klassificera det mesta som negativt. F-score var påtagligt lägre för den positiva och den neutrala klassen. Ett alternativ, som dock inte undersöktes, hade varit att ta fram träningsdata som hade jämnare fördelning för att sedan köra ett testset med den ursprungliga fördelningen. Detta kan potentiellt ge bättre resultat då modellen inte blir övertränad på negativa ord. Naive Bayes lider av liknande problem. Eftersom priorisannolikheten avgörs av hur stor andel av tweetsen som tillhör vilket sentiment så kommer denna skevhet i fördelningen medföra att priorisannolikheten för den negativa klassen blir betydligt högre än vad den blir för övriga klasser.

Modellerna prövades även på en datamängd med jämnare fördelning (RandomTweets) men som dessvärre var sämre klassificerad. Detta resulterade i en jämnare prestanda mellan de båda regressionerna samt mellan regressionerna och Naive Bayes. Man kan även se att resultaten för Naive Bayes blev betydligt sämre vilket indikerar på att priorisannolikheten vägde tyngre än conditional probabilites för att avgöra vilken klass som var mest sannolik. Men det är svårt att dra några slutsatser då det komplementerande

[4] Wikipedia, "Poes Lag", 2021

datasetet led av betydligt fler felklassificeringar än det huvudsakliga datasetet gjorde. Vilket var anledningen till att datasetet med flygtweets valdes över det med slumpmässiga tweets trots sin skeva fördelning.

Slutsats

Vi kan konstatera att hypotesen inte stämmer då Naive Bayes presterar bättre än både Stochastic-gradient descent och Batch-gradient descent eftersom Naive Bayes har högre F-score än de båda logistiska regressionerna, högre accuracy, samt är betydligt snabbare. Detta gäller även när metoderna testas på den inferiöra datamängden (RandomTweets). Även fast Naive Bayes i detta fall inte har lika stor fördel av priorisannolikheterna, då datan är mer jämt fördelad mellan klasserna.

Skevheten i datan medför att modellen kommer klassa tweets som negativa fast de inte är det. Kollar man på F-score för de positiva och neutrala klasserna ser man att detta sker i ännu större utsträckning för de logistiska regressionerna. Utöver överträning kan detta även bero på att särdragen i sig är bristfälliga och inte innehåller tillräckligt mycket data eller att de inte korrelerar bra med de positiva tweetsen. Vi tror även att orsaken till att Naive Bayes presterar så bra är på grund av att enbart ord som är unika eller är överrepresenterade för en specifik ordklass kommer att bidra till att sannolikhetsmassan ökar nämnvärt. Vanliga ord som “the, be to” kommer att bidra till samtliga klassers sannolikhetsmassa men det är vid klassunika ord sannolikheterna skiljer sig. Således medför det att även om priorisannolikheten skiftar så att den negativa klassen får mer sannolikhetsmassa, så kommer våra conditional probabilites medföra att tweets som är laddade med flertalet positiva ord fortfarande klassas som positiva.

Det finns även skillnader mellan de två logistiska regressionerna. Vid första anblick kan man tro att batch har presterat bättre då den har högre accuracy. Om man istället kollar på metodernas F-score ges en mer nyanserad bild. Batch har bättre accuracy eftersom den predicerar mycket fler tweets som negativa än vad Stochastic gör. Detta medför att F-score för klasserna positiv och neutral blir lidande då flertalet tweets som faktiskt var positiva samt neutrala felaktigt prediceras som negativa. I och med att både batch-gradient-decent och stochastic-gradient-decent är generativa modeller så kommer resultaten att variera från gång till gång. Resultaten för stochastic varierar mer än för batch, då den utöver att slumpa fram den initiala theta-matrisen även slumpar den datapunkt som modellen tränas på. Generellt gäller dock att Stochastic genererar bättre resultat. Vad gäller förlustfunktionen så konvergerar den för Batch och Stochastic ungefär lika snabbt även om förlusten för Stochastic initialt varierar mer till följd av den slumpmässigt valda datapunkten.

Genom att jämföra Stochastic Gradient Descent och Naive Bayes så kan man se att Naive Bayes har betydligt bättre precision. Medan Stochastic Gradient Descent har bättre recall (för den positiva och neutrala klassen). Även fast Naive Bayes har högre F-score så behöver det inte per automatik betyda att den är bättre. Eftersom olika ändamål ställer olika krav på vad som prioriteras mellan precision och recall. Om ett flygbolag exempelvis är ute efter att hitta alla positiva tweets utan att missa en enda, då är hög recall viktigt. Om ett flygbolag istället är ute efter att hitta vissa positiva tweets men vill vara säker på att de tweets som klassificeras som positiva faktiskt är positiva så är precision det viktigaste. Självklart vill man att båda ska vara så hög som möjligt men det är ofta en avvägning vad som ska prioriteras.

Källförteckning

Jurafsky, D. Martin, J., “Speech and Language Processing”, 3rd edition, 2021.
https://web.stanford.edu/~jurafsky/slp3/

Wikipedia, “Poes Lag”, 2021
https://sv.wikipedia.org/wiki/Poes_lag

Wikipedia, “Twitter”, 2021
https://sv.wikipedia.org/wiki/Twitter

Gupta, S., “Sentiment Analysis: Concept, Analysis and Applications”, 2018
https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17