Undersøgelse af Natural Language Processing til vurdering af online anmeldelser

Jacob Peter Diesel Nielsen
71317

Luchas Nickolaj Schmidt
71413

William Grynderup Klindt
71347

An image of supervised machine learning painted by John Audubon (Genereret af OpenAI)

Vejledere: Henning Christiansen & Mika Yasuoka Jensen

01/06-2023

Eksamensgruppenr.: S2325625433

Abstract

This study explores the use of Natural Language Processing as a tool for online reviews. By drawing on theories about machine learning, this article investigates how to develop a Natural Language Processing model in Python. The authors present two machine learning models which were developed through an iterative design process and by implementing various approaches from the TRIN-model. This study concludes that machine learning models like ours might not have a significant impact on the future of reviews, but we are optimistic about the potential of Natural Language Processing as a tool for categorizing specific aspects of a company's services. This conclusion is based on the analysis and discussion of the results we gathered from our Natural Language Processing models.

Indholdsfortegnelse

1. Indledning 1
2. Problemfelt 1
3. Problemformulering 2
   3.1 Arbejdsspørgsmål 2
4. Teori og metodeafsnit 3
   4.1 Teori 3
      4.1.1 Supervised- og unsupervised machine learning 3
      4.1.2 Indre mekanismer og processer i Natural Language Processing 4
      4.1.3 Klassifikation 6
      4.1.4 Confusion matrix 6
      4.1.5 Support Vector Machine 7
      4.1.6 Naive Bayes 8
      4.1.7 Lineære modeller 9
      4.1.8 Sentiment analysis 10
   4.2 Validering af data 11
   4.3 Metodeafsnit 12
      4.3.1 TRIN-modellen 12
      4.3.2 Kanban-Board 13
      4.3.3 GitHub 14
5. Analyse af NLP baserede modeller 14
   5.1 Formålet med NLP-modellerne 15
   5.2 Den iterative proces 15
      5.2.1 Første iteration 16
      5.2.2 Anden iteration 16
      5.2.3 Tredje iteration 18
   5.3 Beskrivelse af koden 18
      5.3.1 Opsætning 18
      5.3.2 Scikit-learn 18
      5.3.3 Pandas, NumPy & Matplotlib 19
      5.3.4 Natural Language Tool Kit 19
      5.3.5 Lemmatization 19

5.3.6 Bag-of-Words & N-gram 19
5.4 Databeskrivelse 20
5.5 Data preprocessing og labelling 21
5.6 Data modelling 23
5.7 Resultater fra træningssæt 25
5.8 Modellens resultater 27
6. Diskussion 29
6.1 Udviklingen og evalueringen af vores model 29
6.2 Potentielle problematikker med valg af datasæt 33
6.3 Tilsigtede og utilsigtede effekter ved brugen af NLP 35
7. Konklusion 37

1. Indledning

Før internettets frembrud blev vores opfattelse af begivenheder, services og produkter formet af anmeldelser, der blev udgivet af de traditionelle medier. Anmeldelserne blev blandt befolkningen betragtet som troværdige og pålidelige, da de blev udgivet af etablerede mediehuse. I løbet af de seneste årtier har teknologiens udvikling ændret vores måde at interagere og kommunikere på. Nu til dags har en ny form for anmeldelser ændret måden hvorpå vi tilgår information på, nemlig gennem online anmeldelsesplatforme. Ligesom internettets effekt på anmeldelser, undersøger denne artikel hvordan Natural Language Processing kan benyttes til at evaluere og vurdere anmeldelser på nettet. Artiklen præsenterer to machine learning modeller hvis formål er at kunne evaluere og klassificere en virksomheds online anmeldelser. Modellernes ydeevne bliver analyseret, for at skabe indsigt, i teknologien som et potentielt værktøj for lokale virksomheder.

2. Problemfelt

Anmeldelser er i stigende grad blevet vigtige for forbrugere i vores samfund. Dette gælder især når det kommer til at træffe købsbeslutninger. Mange forbrugere er afhængige af anmeldelser for at få indsigt i kvaliteten og pålideligheden af produkter og tjenester, før de foretager sig et køb. Anmeldelser kan dertil også give forbrugeren værdifulde oplysninger om en virksomheds kundeservice og politikker. Dertil kan anmeldelser også have en direkte effekt på en virksomheds fortjeneste. Positive anmeldelser kan drive salget og øge virksomhedens omsætning, hvorimod negative anmeldelser kan have den modsatte effekt.

Mange forbrugere og serviceudbydere opretter, læser eller svarer på anmeldelser, på anmeldelsesplatforme, såsom Trustpilot og Yelp. Anindya Ghose & Panagiotis G. Ipeirotis (2011), professorer indenfor teknologi fra New York University, beskriver hvordan online anmeldelser kan gøre det sværere for individer at vurdere et produkts sande kvalitet, da en overvældende majoritet af online anmeldelser er tildelt en ekstrem høj eller lav bedømmelse. Dette fører til, at den gennemsnitlige numeriske stjernebedømmelse ikke kan formidle retvisende information til en potentiel køber. De mener at forbrugeren burde læse anmeldelserne for at undersøge det positive eller negative aspekt af et givent produkt. Sangwon Park & Juan Luis Nicolau (2015), henholdsvis opnået en doktorgrad og Ph.d. indenfor business administration, fremlægger i deres undersøgelse om nyttigheden af online anmeldelser, at folk generelt opfatter ekstreme vurderinger, både negative og positive, som mere nyttige end moderate eller neutrale

vurderinger. Undersøgelsen af online anmeldelser af restauranter i henholdsvis London og New York viser dog, at negative anmeldelser bliver vurderet som mere nyttige, end positive anmeldelser (Park & Nicolau, 2015). Anmeldelser er derfor en værdifuld kilde til information for forbrugere, da både positive og negative anmeldelser ofte fremhæver specifikke aspekter af en virksomhed. Disse aspekter kan give indsigt i virksomhedens produkter eller services og deres kvalitet. Dette understreger vigtigheden i at forstå den følelsesmæssige tone og kategorisering af henholdsvis positive, neutrale eller negative aspekter af anmeldelser. Dette kan hjælpe forbrugere med at træffe informerede beslutninger, når de overvejer at købe en virksomheds produkter eller services.

Den teknologiske fremdrift har medført et markant skift af hvordan vi handler, da der er sket en eksponentiel stigning af handler, der foregår via internettet. Dette har haft en stor indflydelse på vigtigheden af anmeldelsesplatforme, da de virker som et medie mellem forbrugere og udbydere. Ifølge Morgan Stanley (u.d.) foregår cirka 23% af alle handler i USA igennem internettet, og det forventes at dette tal vil stige til 31% ved 2026.

Anmeldelsesplatforme, såsom Trustpilot, opfordrer erhvervsdrivende til aktivt at bede deres kunder om at bedømme deres forretning efter en afsluttet handel (Trustpilot u.d.). Derfor vil det være sandsynligt, at der i større og større grad vil være en stigning i antallet af anmeldelser. I takt med stigningen af onlinehandel og anmeldelser, kan det formodes at virksomheder i højere grad skal behandle større mængder af data. Ved at udnytte sofistikeret teknologi, såsom Natural Language Processing, kan arbejdsbyrden mindskes for virksomheder, der ønsker at udlede viden og indsigter fra kundernes anmeldelser. Ved at automatisere denne proces, kan arbejdsbyrden formindskes og eventuelle nye indsigter og adfærdsmønstre, som mennesket ikke nødvendigvis genkender, opdages og bidrage til forbedring af virksomhedens produkter og services.

3. Problemformulering

Kan Natural Language Processing benyttes som et effektivt værktøj, til evaluering af online anmeldelser?

3.1 Arbejdsspørgsmål

1. Hvad definerer Natural Language Processing og hvordan kan det anvendes?
2. Hvilke tilgange medfører udviklingen af effektive Natural Language Processing-modeller?

3. Kan en Natural Language Processing-model demonstrere generaliserbarhed og anvendes i forskellige kontekster?

4. Teori og metodeafsnit

Vi vil i dette afsnit redegøre for vores anvendte teori og metode. Afsnittet er delt op i tre underafsnit, hvor vi i første del vil redegøre for teori, herunder supervised machine learning, klassifikation og diverse relevante supervised algoritmer, for Natural Language Processing. I anden del af afsnittet vil vi præsentere projektets videnskabsteoretiske kobling, herunder hvordan vi validerer datasæt til modellering af machine learning algoritmer. Der vil i tredje del af afsnittet blive redegjort for de metoder, som er blevet benyttet i projektet. I projektet gøres der brug af Kanban board og GitHub, der anvendes til projekt-strukturering, og derefter TRIN-modellen, der danner basis for vores analyse af Natural Language Processing.

4.1 Teori

4.1.1 Supervised- og unsupervised machine learning

Inden for machine learning skelnes der fundamentalt mellem to forskellige tilgange, nemlig supervised- og unsupervised learning. Hui Jiang (2021), professor på York University i Canada, sætter i bogen "Machine Learning Fundamentals – A consise Introduction" ord på hvad disse forskelle udgør.

Supervised learning er kendetegnet ved algoritmer, som trænes ved brugen af datasæt, hvor hvert element har en beskrivelse, også kaldt "label", tilknyttet. Et tænkt eksempel er en algoritme, som skal kunne vurdere hvorvidt billeder viser en hund eller en kat. Træningsdataene vil således bestå af adskillige billeder, hvortil hvert billede tildeles én af følgende labels: "Hund" eller "Kat". Hensigten er efterfølgende, at algoritmen skal kunne genkende billederne, uden på forhånd at vide hvad billedet viser. Uanset hvilken supervised learning algoritme der anvendes, kræves der dog som oftest en menneskelig intervention, hvad angår tildelingen af labels til datasættet (Jiang, 2021). Hertil er det også centralt, at datasættet har et tilfredsstillende omfang, hvilket som oftest kan være en ressourcekrævende proces. Ifølge Goodfellow et al. (2016), forskergruppe indenfor deep learning fra Université de Montréal i Canada, kan opnåelse af store og forskelligartede datasæt være en vanskelig og ressourcekrævende proces. Dette fremhæver vigtigheden af datasættets kvalitet og omfang, i relation til machine learning.

I modsætning til supervised learning, er unsupervised learning kendetegnet ved at blive trænet på datasæt uden labels. Med andre ord er hensigten, at algoritmen genkender mønstre i datasættet uden labels, og heraf tildeler et label. Et tænkt eksempel kunne være et billede af legoklodser i forskellige farver og former. Algoritmen har ingen forhåndsviden om legoklodser, ej heller om farver og former. Men netop disse faktorer udgør forskelle, som algoritmen kan tildele labels ud fra. Fraværet af menneskelig intervention for tildeling af labels i datasættet, er i modsætning til supervised tilgangen mindre ressourcekrævende. Men det medfører samtidigt en anden problemstilling, nemlig hvordan unsupervised algoritmer trænes til at genkende data uden labels. Derfor er det også vanskeligere at skabe en model med denne tilgang (Jiang, 2021).

Vi anser supervised learning som den mest hensigtsmæssige tilgang for vores projekt, da vi ønsker at kunne genkende og klassificerer anmeldelser. Ved at bruge supervised learning kan modellen trænes på det labelled datasæt og deraf identificere mønstre i dataene. Når modellen er trænet, kan den anvendes til at klassificere nye og ikke-labelled data, med en høj grad af nøjagtighed.

Ethem Alpaydin (2014), datalogi professor fra Ozyegin Universitet i Istanbul, forklarer hvordan supervised learning er den mest almindelige type af machine learning til klassificeringsopgaver. Dette skyldes brugen af labelled datasæt til at træne modeller, hvilket generelt fører til bedre ydeevne end unsupervised learning (Alpaydin, 2014).

4.1.2 Indre mekanismer og processer i Natural Language Processing

Dette afsnit definerer Natural Language Processing (NLP), med udgangspunkt i TRIN-modellens første trin, teknologiers indre mekanismer og processer. Denne fremgangsmåde vil hjælpe os med at fremstille klare detaljer om hvordan NLP fungerer.

Da vi ønsker at undersøge og udvikle vores egen model, vil dette danne basis for hvilke muligheder NLP har som teknologi. NLP er et underområde af machine learning, som fokuserer på samspillet mellem teknologi og naturligt sprog. NLP kombinerer feltet inden for lingvistik og datalogi til at afkode sprog struktur, ved at træne computeren til at analysere, fortolke og generere naturligt sprog, i form af tekst og tale, på en måde så det er brugbart for mennesker (Chowdhary, 2020).

Ifølge K. R. Chowdhary (2020), tidligere professor indenfor datalogi ved MBM University i Indien, i bogen "Fundamentals of Artificial Intelligence", bliver NLP blandt andet defineret som:

> "Natural language processing (NLP) is an academic, and technology-based research domain comprising a range of computational techniques for representation and automatic analysis of human languages—a field that is motivated by theory. Automatic analysis of text requires a deep understanding of natural language by machines." (Chowdhary, 2020: 645).

NLP er en række teknikker til at repræsentere og analysere naturligt sprog. Dog kræver sådan en analyse af tekst, en meget stor forståelse af naturligt sprog for computere. For mennesker er det en daglig, og muligvis ubevidst, praksis at udøve NLP. I det at du konverserer med en anden person, ser en film i fjernsynet eller læser denne artikel, lytter og læser du til ord og sætninger som bliver formet, og du vil ved automatik selv skabe en forståelse ud fra det uden besvær. NLP er et fundamentalt aspekt af menneskelig kommunikation og er ofte noget man ikke tænker over. Men for at computere kan udføre effektiv NLP, skal de trænes via enorme og komplekse mængder af sproglige data og tilegne sig viden indenfor en bred række af lingvistiske felter, såsom syntaks, semantik og pragmatik (Chowdhary, 2020). NLP for computere kan yderligere blive delt op i to overlappende underkategorier: Natural Language Understanding (NLU) og Natural Language Generation (NLG).

For at en computer skulle være i stand til at udføre NLP kræver det, at den kan behandle input sprogdata og transformere dem til en form, som kan analyseres og handles ud fra: "To make our computers to understand us, we need to equip them with adequate knowledge." (Chowdhary, 2020: 647). NLU dækker over processen af at omdanne ustruktureret data til struktureret data; en yderst nødvendig proces for at computeren kan forstå naturligt sprog. Dette dækker over en bred vifte af teknikker såsom tekst klassificering, følelsesanalyse (sentiment analysis), syntaks- og grammatik analyse. Samlet set er målet ved NLU at gøre computeren i stand til at kunne forstå og bearbejde naturligt sprog, på samme måde som mennesker. Med det bearbejdede input data er den næste proces at kunne producere et output, som er sammenhængende og naturligt klingende for mennesker at kunne forstå. Dette er målet med NLG, som dækker over mange forskellige teknikker, der gør computeren i stand til at generere dette (Chowdhary, 2020).

Selvom dette projekt udelukkende vil fokusere på NLU, er begge underkategorier essentielle for at kunne danne et helhedsbillede af NLP som teknologi og de dertilhørende indre mekanismer og processer.

4.1.3 Klassifikation

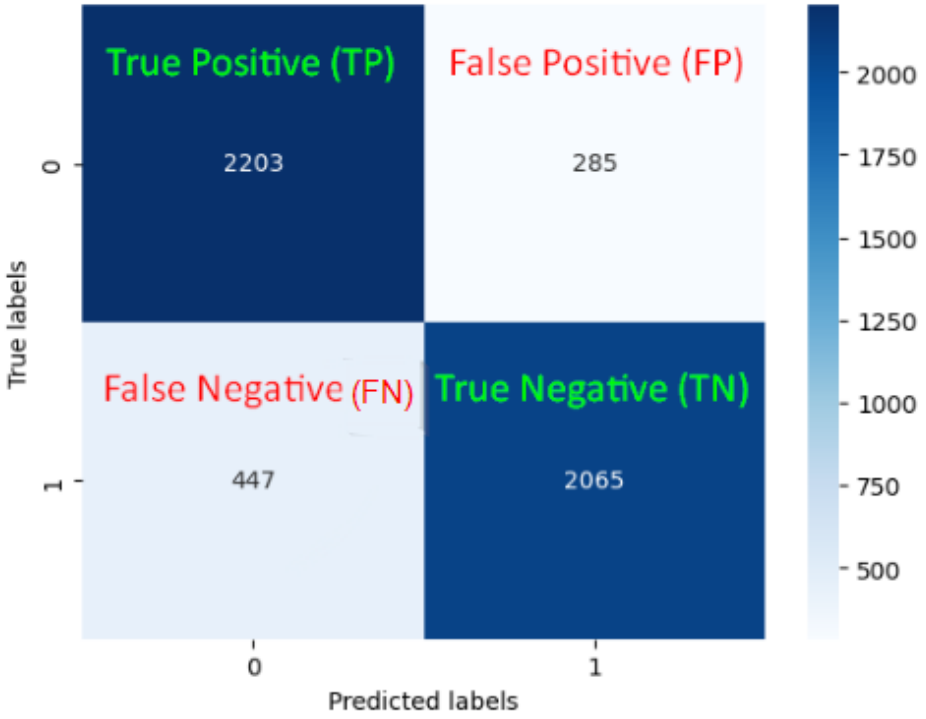
Klassifikation betegner en tilgang inden for supervised machine learning, hvor machine learning algoritmens formål er at genkende og kategorisere ny data med afsæt i modellens træningsdata. Der skelnes samtidigt mellem forskellige typer af klassifikation, eksempelvis binær og multi-klasse, som hver især anvendes med forskellige henseender. Hvor der ved klassifikation med binær kun opdeles i to kategorier (f.eks. 0 eller 1), så opdeles der i multi-klasse i flere end to kategorier (f.eks. Hund, Kat og Papegøje) (Jo, 2021).

Retter vi blikket mod et nyt eksempel, hvor der skal differentieres mellem positive og negative kommentarer, kan begge tilgange til klassificering bruges. Hvor der ved binær blot opdeles i positiv/negativ, opdeles der ved multi-klasse i forskellige underkategorier. Begge tilgange til klassificering, men særligt binær, udfordres dog af sprogets natur, som er mangfoldigt og nuanceret. Forfatteren bag en skrevet kommentar kan for eksempel bruge sarkasme eller slang, hvilket potentielt problematiseres af algoritmens klassificering af kommentaren. Samtidigt er sproget i konstant udvikling, og ords betydning er heraf skiftende (Risch & Krestel, 2020).

Vi anser klassifikation gennem multi-klasse som den mest hensigtsmæssige for vores projekt. Dette begrunder vi i en forventning om at klassificere kommentarer, ud fra forskellige kategorier.

4.1.4 Confusion matrix

En måde at vurdere effektiviteten af en klassifikations model, er ved brug af en confusion matrix. Denne matrix visualiseres gennem en X*X tabel, hvor X indikerer antallet af klasser der skal klassificeres. Ved binær klassificering vil der således være tale om en 2*2 tabel, og ved multi-klasse, med tre kategorier, vil der være tale om en 3*3 tabel. Matricen repræsenterer “true label” (det faktiske label i datasættet), samt “predicted label” (algoritmens forudsigelse for labelled). Matricen sammenfatter med andre ord algoritmens output i form af antallet af korrekte/ikke-korrekte klassificeringer. Helt konkret dækker matrixen over såkaldte false/true positives (FP/TP), samt false/true negatives (FN/TN) (Muller & Guido, 2016).



Figur 1: Eksempel på confusion matrix, for binær klassificering.

I figur 1 ses et eksempel for en confusion matrix, med afsæt i binær klassificering. Resultatet af en sådan matrix, gældende for både binær og multi-klasse, kan sammenfattes gennem flere forskellige udregninger. Dette dækker blandt andet over accuracy, som giver en indsigt i en klassificeringsalgoritmes præstation.

Accuracy er et udtryk for, hvor ofte algoritmen giver det korrekte output. Dette udregnes ved at sammenlægge antallet af TP med antallet af TN, hvortil dette divideres med det samlede antal af klassificeringer. Regnestykket vil se således ud:

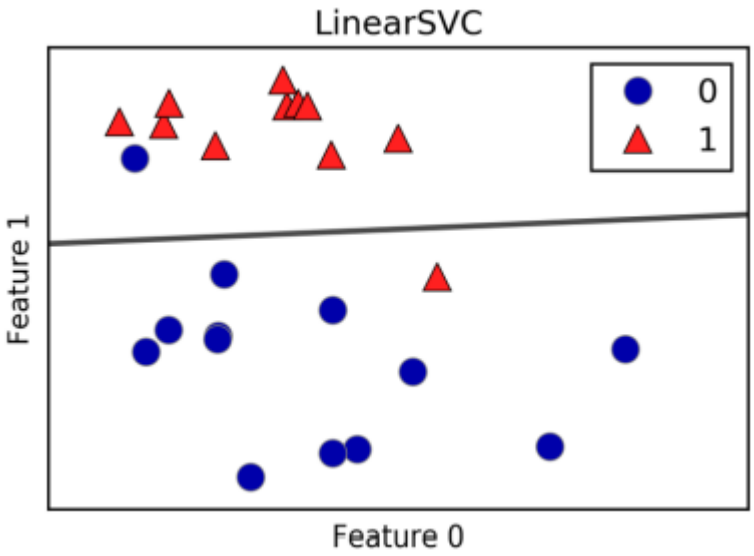
$$\frac{TP + TN}{TP + TN + FP + FN}$$

(Muller & Guido, 2016).

4.1.5 Support Vector Machine

Linear Support Vector Machine (SVM) er endnu en supervised machine learning algoritme, som kan anvendes til klassificering. SVM forsøger at opdele et givent data domæne ved at finde den bedste hyperplan mellem to klasser, hvilket sker ved at maksimere marginen mellem dem. Marginen betegnes som afstanden mellem hyperplanen og de nærmeste datapunkter fra hver

klasse. De to datapunkter bliver også betegnet som support vectors, på dansk understøttende vektorer, hvilket danner basis for algoritmens navn (Suthaharan, 2016).



Figur 2: Visualisering af SVM.

I figur 2 fremstår et eksempel på hvordan SVM har opdelt et data domæne. Illustrationen viser helt præcist, hvordan domænet opdeles af en hyperplan (i form af en sort streg), som adskiller de to respektive klasser (0 og 1). Når den trænede model præsenteres for ny data, vil datapunkter placeret over denne streg blive klassificeret som 1, og derimod som 0, hvis datapunkterne placeres under (Muller & Guido, 2016).

Lineære modeller, herunder SVM, har til fordel, at de ikke er tidskrævende hvad angår træningstid, samt endelig klassificering. Samtidigt egner de sig godt til datasæt med høj dimensionalitet, i form af mange features. Her evner de nemlig effektivt at finde de respektive hyperplaner som adskiller de forskellige klasser (Muller & Guido, 2016). Disse egenskaber anser vi som vigtige, set i lyset af vores datasæt størrelse og kompleksitet.

4.1.6 Naive Bayes

Naive Bayes (NB) er en klassifikationsalgoritme som er bygget op omkring Bayes's teorem, med hensigt på at forudsige sandsynligheden for at noget hænder, baseret på ny information. Bayes' teorem antager at sandsynligheden for at noget sker, givet at noget andet allerede er sket. Bayes kalder det for "conditional probability", som oversat på dansk er betinget sandsynlighed (Chowdhary, 2020).

Bayes' teorem kan eksemplificeres ved at prøve at forudsige om det vil komme til at regne i dag. Der hvor du befinder dig ved du, at i gennemsnittet regner det omkring 1 ud af hver 3. dag. Vejrprognoserne forudsiger dog, at der er 70% chance for at det vil regne der, hvor du befinder dig. Bayes' teorem kan her hjælpe med at justere det oprindelig estimat for sandsynligheden for at det vil regne ved at inddrage den nye information. Bayes' teorem anvender den betinget sandsynlighed sammen med det oprindelige sandsynlighedsestimat (i givet fald, 1/3) til at beregne et nyt estimat for sandsynligheden, som også tager højde for den nye information (Chowdhary, 2020).

Set i kontekst af en NB-klassificering, er antagelsen at alle data er uafhængige af hinanden. Med andre ord, så er de forskellige informationer (features) betinget uafhængige givet en label. Dette betyder at, tilstedeværelsen eller fraværet fra en feature ikke afhænger af tilstedeværelsen eller fraværet af en anden feature, i samme givende label. Selvom denne "naive" antagelse muligvis ikke er sand i virkeligheden, kan klassificeringsalgoritmen være anvendelig i praksis (Muller & Guido, 2016). Denne algoritme bruges ofte til tekst-klassifikation, hvor dataene består af et par tusinde enkeltstående ord. Modellen kræver derfor kun en lille mængde træningsdata og kan køre algoritmen ekstremt hurtigt i forhold til andre sofistikerede modeller (Scikit-learn developers, u.d.). NB kan derfor bruges til at klassificere ordene, der indgår i et træningssæt og forsøge at forudsige om ordene i testsættet har den samme betydning (Chowdhary, 2020).

4.1.7 Lineære modeller

Lineære modeller omfatter en bred vifte af machine learning modeller, som alle har det fælles formål, at de anvender et lineært forhold mellem input features og output for at lave forudsigelser. Input features omfatter med andre ord de karakteristika, som danner grundlag for modellens output, altså de endelige forudsigelser. Lineære modeller anvendes til hhv. regression og klassifikations- opgaver. Ved regression sker forudsigelsen af en værdi gennem en lineær kombination af input features. Består datasættet af flere end én feature, så sker forudsigelsen som en vægtet sum af de samlede input features. Ved klassifikation sker forudsigelsen af en kategori ligeledes gennem en lineær kombination af input features, hvortil en hyperplan afgøre opdelingen af dataene i forskellige klasser (Muller & Guido, 2016).

Lineær regression, også kendt som Ordinary Least Squares (OLS), er en lineær supervised machine learning algoritme. Som navnet antyder anvendes den til regressions opgaver, hvis

formål er at forudsige en kontinuerlig værdi med afsæt i de givne input features. Dertil er det én af de mest simple tilgange når det kommer til regression. OLS har nemlig ingen parametre, som kan ændres og heraf er det ej muligt at ændre på modellens kompleksitet. Som med øvrige lineære algoritmer, såsom Linear SVM, så antager lineær regression også at der forefindes en lineær relation mellem input features og modellens output. Hensigten for algoritmen er at finde det bedste lineære forhold for dataene, hvilket sker ved at minimere den såkaldte mean squared error (MSE) værdi. MSE betegnes som gennemsnittet af de kvadrerede forskelle blandt de forudsagte værdier, samt de faktiske værdier. Grundet OLS's simple natur, er der risiko for at modellen overtrænes. Dette gør sig gældende i tilfælde af datasæt med et stort antal features. I sådanne tilfælde bør der opvejes, hvorvidt øvrige modeller vil kunne effektuere et bedre resultat (Muller & Guido, 2016).

4.1.8 Sentiment analysis

Sentiment analysis er en underkategori af NLP, der bliver brugt til at identificere den følelsesmæssig tone eller udtryk i en given tekst. Anvendelsen af sentiment analysis bliver hyppigt brugt til klassificering af holdninger på sociale medier, undersøgelser af anmeldelser, marketing mm. Her bliver holdninger og tekst ofte klassificeret som positive eller negative (Muller & Guido, 2016).

Det er vigtigt at pointere, at sentiment analysis ikke er en fuldkommen videnskab, og de samme ord og sætninger kan fortolkes forskelligt afhængigt af individets perspektiv og konteksten ordene bliver anvendt i. Hvad der er positivt for én person, kan være negativt for en anden. Sentiment analysis kan have udfordringer ved at konkludere den nøjagtige nuance af ord og sætninger, og derved vil det ikke altid reflektere den rigtige følelsesmæssige tone. Det er derfor vigtigt at overveje eventuelle begrænsninger og den mulige partiskhed, når man fortolker resultaterne ved sentiment analysis. Da dette projekt udfører sentiment analysis via machine learning modeller, hvor der bliver anvendt et labelled datasæt, er det væsentligt at forholde sig kritisk til datasættet. Det er vigtigt at være opmærksom på potentiel partiskhed i dataene. Dette kan opstå fra forskellige kilder, herunder processen ved dataindsamlingen, valg af labels og forfatterens subjektive fortolkning af dataene (Muller & Guido, 2016).

4.2 Validering af data

Validering af data er afgørende for vores undersøgelse, da datasættets validitet kan have indflydelse på de konklusioner, vi drager ud fra vores algoritmiske resultater. I dette afsnit vil vi benytte en videnskabsteoretisk tilgang, til at diskutere validering af data i relation til NLP og data science.

Den videnskabsteoretiske tilgang, positivismen, fremhæver vigtigheden af, at den viden vi indsamler understøtter det positive, der i positivismen betyder det faktiske eller konkrete (Holm, 2011). Den anvendte data antager vi er objektivt, målbart data, hvilket ifølge de positivistiske briller er den eneste form for videnskab, der er sand. Denne tilgang tvinger os til at verificere datakilden og undersøge dataene for eventuelle problematikker, for at sikre dataenes anvendelighed i projektet.

Validitet og pålideligheden af de data vi bruger til at træne vores NLP-model, er af stor betydning, da det er disse data, der lærer modellen at skelne mellem positive, negative og neutrale fraser. Det har derfor været afgørende for os at vælge vores data med omhu, for at undgå træning af modellen på bias, diskriminerende eller racistisk data.

En grundlæggende videnskabsteoretisk antagelse i denne sammenhæng er, at dataene skal være valide og pålidelige, da de danner grundlaget for vores analyser og konklusioner. Vi vil derfor undersøge, hvordan vi kan validere vores datasæt og vurdere deres pålidelighed i forhold til de videnskabelige standarder og kriterier for datakvalitet.

Vi har anvendt data fra Kaggle.com til at træne vores modeller. Kaggle er en online platform, der henvender sig til data science og machine learning entusiaster og giver brugere adgang til et bredt udvalg af offentlige datasæt, der er designet til konkurrencer, kurser og projekter (Moltzau, 2021). Vores datasæt består af 568.000 produktanmeldelser, som er indsamlet fra internethandelssiden Amazon (Amazon Product Reviews, u.d.).

I vores projektgruppe antager vi, at dataene stammer fra ægte anmeldelser, og at stjernebedømmelserne afspejler brugernes anmeldelser. Vi har gennemgået en mindre del af datasættet for at validere om bedømmelsesscorerne afspejler brugernes anmeldelser. Selvom vi er enige i de fleste tilfælde, har vi også bemærket, at der er få tilfælde af, hvor anmeldelser og dens dertilhørende bedømmelsesscore ikke er retvisende. Vi har derfor valideret dataene ud fra

vores egne holdninger. Vi mener, at datasættet er relevant og pålideligt, og vi har derfor besluttet at træne vores model med disse data.

Tager vi udgangspunkt i den fænomenologiske videnskabsteoretiske tilgang, antyder vi, at al viden skabes ud fra en filosofisk undersøgelse af, hvordan mennesker opfatter og erfarer verden omkring dem. Fænomenologi undersøger også hvordan disse erfaringer kan beskrives og analyseres (Olesen, 2021.).

I relation til projektet og det valgte datasæt, kan fænomenologi benyttes som en videnskabsteoretisk tilgang til at undersøge, hvordan vi oplever og tolker sproget og kommunikationen. Fænomenologi tager udgangspunkt i, at vores oplevelse af verden er subjektiv og individuel, og at vores erfaringer og opfattelser af verden er dybt forankrede i vores personlige baggrund, kultur, sprog og kontekst (Olesen, 2021.).

Når vi arbejder med NLP, er det vigtigt at tage højde for, hvordan dem der har klassificeret datasættet, opfatter og tolker fraserne. Fænomenologi kan hjælpe os med at forstå, at mennesker har forskellige perspektiver og oplevelser af sproget. Dette kan give os indsigt i, hvordan vi kan udvikle bedre algoritmer, der tager højde for fortolkningen af sproget i forskellige kulturer, aldersgrupper, samfundslag og demografiske områder. Derfor kan en fænomenologisk tilgang give os en mere nuanceret og dybtgående forståelse af, hvordan sproget fungerer i praksis og hvordan vi kan udvikle bedre NLP-algoritmer, på baggrund af vores fortolkning af datasættet.

4.3 Metodeafsnit

Vi vil i dette afsnit fremlægge de forskellige metoder, som er blevet anvendt i projektet. Metoderne der præsenteres i afsnittet, har hjulpet os med at anvende den indsamlede teori. Derudover kan nogle af de nævnte metoder også anses som værende brugbare værktøjer, da vi skulle strukturere vores udviklingsproces. Vi har benyttet os af GitHub og Kanban board for at planlægge vores udviklingsproces og ajourfører vores forskellige versioner af koden. Dertil inddrager vi TRIN-modellen som analyseværktøj af teknologien NLP.

4.3.1 TRIN-modellen

TRIN-modellen vil hjælpe os med at analysere, med hovedvægt på teknisk-videnskabelige aspekter, og begribe NLP som en teknologi (Jørgensen, 2020). TRIN-modellen består af seks forskellige trin:

1. Teknologiers indre mekanismer og processer
2. Teknologiers artefakter
3. Teknologiers utilsigtede effekter
4. Teknologiske systemer
5. Modeller af teknologier
6. Teknologier som innovation

I dette projekt vælger vi at tage udgangspunkt i to af de ovenstående seks trin, da det er dem vi finder mest relevante for vores projekt. Vi vælger at anvende trin 1. Teknologiers indre mekanismer og processer, og trin 3. Teknologiers utilsigtede effekter.

Trin 1 bliver beskrevet som "De centrale principper ved en teknologi, som bidrager til at opfylde teknologiens formål" (Jørgensen, 2020: 6). Her ønsker man at undersøge den pågældende teknologis centrale processer og indre mekanismer, der tilsammen udgør teknologien. Vi vil bruge trin 1 til at definere NLP-teknologien og opnå en forståelse af hvilke elementer, der er vigtige i teknologien. Særligt med henblik på at opnå en forståelse for, hvilke kerneelementer NLP skal indeholde, for at teknologien skaber et godt nøjagtigt output.

Trin 3: Teknologiers utilsigtede effekter, henviser til de elementer af en teknologi, der vurderes til at have negative konsekvenser for teknologien (Jørgensen, 2020). Vi ønsker særligt at benytte os af dette trin i diskussionen, for at undersøge om der er nogle utilsigtede effekter ved NLP-teknologien. Vi ønsker dog hertil også at undersøge teknologiens tilsigtede effekter, for at undersøge teknologiens potentialer.

4.3.2 Kanban-Board

Karl Cox (2021), professor på University of Brighton, giver i bogen "Business Analysis, Requirements and Project Management – A Guide for Computing Students" et overblik over Kanban boards, som metode til projekthåndtering.

Kanban board er en agil metode, som har til hensigt at visualisere og strukturere et projekt. Boarded illustrerer forskellige stadier af arbejdsprocessen, samt de dertilhørende arbejdsopgaver. Når en opgave afsluttes, flyttes den fra ét stadie til et andet. Heraf skabes der et fælles overblik over, hvor langt i projektet man er. Metoden forudsætter ikke, at der laves nøjagtige forudsigelser for

tidsrammen, når arbejdsopgaverne tages i betragtning. I stedet for, præsenterer boardet et samlet overblik over hvilke opgaver, der skal udføres (og som er udført) (Cox, 2021).

Vores Kanban board består af fire forskellige stadier: "To do", "In progress", "Roadblocks" og "Done". Disse fire stadier skal bidrage til, at vi kan nå at udføre så mange arbejdsopgaver som muligt. Vi forventer derfor, at boardet kommer til at bidrage positivt, når det kommer til at overholde deadlines for projektet. Blandt projektgruppens medlemmer er der enighed om, at Kanban boards i tidligere projekter har bidraget til at holde fokus på de centrale aspekter af opgaven. Derfor påtænker vi også at placere eventuelle ikke-væsentlige arbejdsopgaver i stadiet "Roadblocks", såfremt vi ikke kan løse dem, og de samtidigt ikke er kritiske for vores problemfelt.

4.3.3 GitHub

Da vi påbegyndte udviklingsfasen, anskuede vi nødvendigheden i at kunne dele vores kode blandt gruppemedlemmerne, således at vi alle kan tilgå den nyeste version. Derfor besluttede vi at gøre brug af Github. Github er en gratis online platform til deling af kode og understøtter samtidigt også versionsstyring. Som bruger af Github, kan man lave et såkaldt repository, hvortil man kan uploade ens kode og filer. Det er samtidigt muligt at lave branches, hvilket kan beskrives som forskellige grene af ens repository. Som udgangspunkt består et repository af en main-branch, og det er her at gruppens medlemmer primært arbejder. Ønsker man at eksperimenterer med koden, har man mulighed for at lave sub-branches. Her kan man lave ændringer i koden, uden det gør sig gældende for de øvrige medlemmer. Bidrager ændringerne til forbedret kode, kan man vælge at uploade den til ens main-branch (Hello World - GitHub Docs, n.d.).

5. Analyse af NLP baserede modeller

Dette afsnit har til formål at præsentere udviklingsprocessen og formålet med vores modeller. Afsnittet vil fremlægge de forskellige iterationer, modellerne har været igennem, samt evaluering heraf. Derefter vil modellens opbygning og struktur blive præsenteret som resultat af de forskellige iterationer. Her vil vi uddybe de forskellige funktioner og libraries, som vi har anvendt. Afslutningsvis vil vi analysere og evaluere resultaterne for henholdsvis NB og SVM som klassificeringsalgoritmer. Her vil der også blive inddraget et nyt datasæt, som vil bruges til at evaluere modellerne.

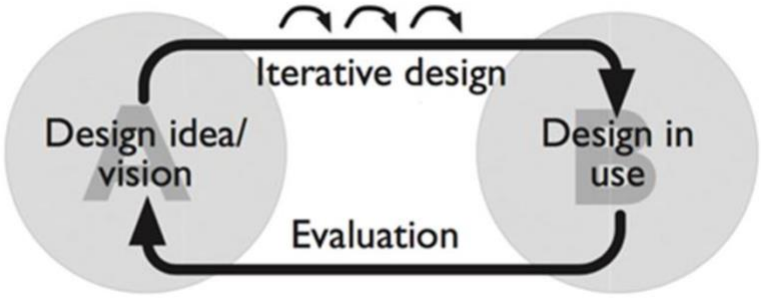
5.1 Formålet med NLP-modellerne

Dette afsnit beskriver formålet med vores NLP-modeller og deres potentiale i en anvendelsessammenhæng. Vores modeller er designet til at analysere anmeldelser fra lokale virksomheder og kategorisere deres følelsesmæssige tone. Vi håber, at vores nuværende modeller vil have evnen til at kategorisere anmeldelser. Modellerne skal derfor kunne evaluere en anmeldelse ud fra brugernes beskrivelser, i stedet for at skulle evaluere ud fra en bedømmelsesscore fra 1-5. Vores håb er derfor, at vores model kan give virksomheder en bedre forståelse af, hvordan kunderne oplever deres produkter eller serviceydelser.

Hvis vi havde mere tid, flere ressourcer og en forbedret forståelse af machine learning, vil vores mål være at videreudvikle modellen til at kunne identificere den følelsesmæssige tone i anmeldelser, af særlige produkter og serviceområder. På denne måde vil modellen kunne kategorisere hvilke produkter, eller serviceområder, en kunde anmelder. Ved at identificere den følelsesmæssige tone af et produkt eller service, kan virksomheder bedre målrette deres indsats for at forbedre et specifikt produkt eller serviceområde. Virksomheden vil dermed kunne tiltrække flere kunder og skalere virksomheden.

5.2 Den iterative proces

Idéen om hvordan vores model skulle udforme sig, har ikke været tydelig fra starten af vores projekt. Vi vidste dog på forhånd, at vi ville arbejde med NLP. Dette skyldes, at der var en stor motivation og nysgerrighed for teknologien. Den store undren var hvilken vinkel på projektet, som vi skulle vælge. Vi har igennem en iterativ proces skabt erfaringer og tilegnet os ny viden, hvorpå vi løbende kunne forbedre vores modeller. Man kan påpege, at designprocessen kører i ring, da vi konstant får nye idéer og inputs til vores model og derved bliver ved med at forbedre den (Schön, 1988). Nedenstående figur illustrerer vores tilgang til designprocessen herom (Se figur 3). Dette afsnit vil derved beskrive vores iterative udviklingsproces af modellerne og de dertilhørende udfordringer.



Figur 3: Den iterative proces.

5.2.1 Første iteration

Det første der vakte interesse, var anvendelsen af NLP til at identificere hadefulde ytringer på online anmeldelsesplatforme, og online mobning på sociale medier. På baggrund af vores litteratur fra undersøgelsesfasen og rådgivning fra vores vejleder, valgte vi at benytte Stanfords CoreNLP; en open-source NLP-toolkit, udviklet af Stanford University's Natural Language Processing Group (Stanford Natural Language Processing Group, u.d.) (Se bilag 1). CoreNLP udbyder effektive værktøjer som lemmatization og sentiment analysis, hvilket vi ville benytte til at afprøve detektionen af hadefulde og negative beskeder. Modellen kunne vise os om en frase eller et ord var enten: online mobning, negativt ladet eller neutralt. Vi fandt dette interessant, da idéen var at det vil kunne hjælpe en platform med at registrere hadefulde ytringer og give dem mulighed for at tage handling i realtid. Dertil var vi også nysgerrige på, om en algoritme kunne identificere online mobning på sociale medier. Vores algoritme, der benyttede sig af CoreNLP, blev udviklet i programmeringssproget, Java, da vi alle havde stort kendskab til sproget.

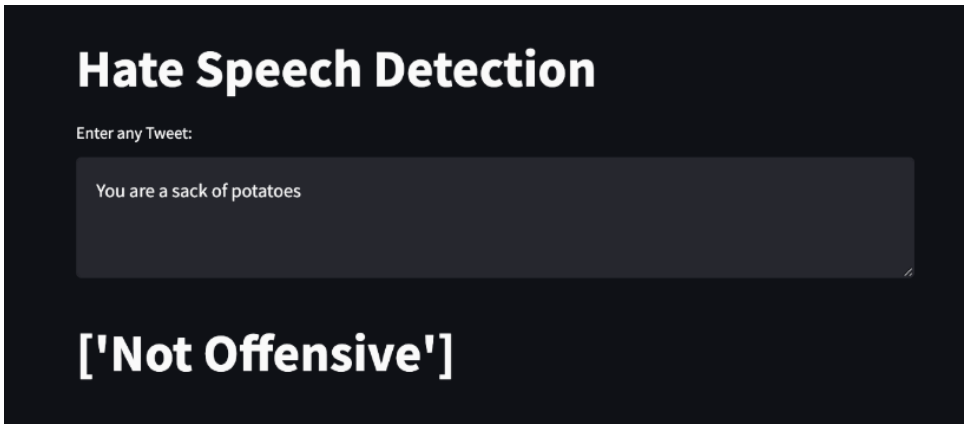
Under den indledende iteration blev det tydeligt for os, at Java var mindre effektiv til udvikling af vores NLP-algoritme, grundet dens begrænsede tilgængelighed af relevante libraries. Vi evaluerede flere programmeringssprog som vi fandt relevante, hvoraf vi fandt Python brugbar. Dette skyldes Pythons bredere udvalg af libraries inden for NLP og generelle bedre ydeevne indenfor feltet.

5.2.2 Anden iteration

Python introducerede os til en række nye muligheder, i form af de mange libraries vi kunne tage i brug. Dette førte til anden iteration af modellen, og markerede overgangen fra Java til Python.

Dette betød, at vi i stedet for Stanfords CoreNLP, anvendte et decision tree klassificeringsalgoritme library til at træne og teste vores model. Vi trænede modellen ud fra et datasæt af Twitter kommentarer, labelled som enten værende offensivt sprog, ikke offensivt sprog eller hadefuldt sprog. Dette betød også, at vi valgte at fokusere udelukkende på hadefulde anmeldelser og kommentarer.

For at teste modellen med nyt input, importerede vi Streamlit library; et open-source library med formål at udvikle interaktive brugergrænseflader til machine learning projekter (Streamlit, u.d.). Figur 4 viser brugergrænsefladen vi udviklede for at teste vores model, hvor brugeren kan indtaste tekst i input feltet. Modellen ville derefter klassificere teksten ud fra én af de tre labels.



Figur 4: Billede af Streamlit brugergrænsefladen fra anden iteration.

Brugergrænsefladen gjorde det nemt for os at teste vores trænede model, med en bred vifte af forskellige og nye inputs, hvilket gav os ny indsigt i vores models mangler. Vores model var ofte misvisende, da den havde tendens til at klassificere input som det "forkerte", set i vores øjne. Selvom vi prøvede at give den input, som vi mente hældte mere mod én bestemt klasse, resulterede modellen i en af de to andre. Vores misvisende model gjorde at vi blev nødt til at evaluere decision tree som klassificeringsalgoritme og eventuelt undersøge andre mulige algoritmer. Dette ledte os også til spørgsmålet om hvornår sprog egentligt er offensivt eller ikke offensivt, og hvad forskellen mellem offensivt sprog og hadefulde ytringer er. Dette mente vi var udenfor projektets undersøgelsesfelt.

5.2.3 Tredje iteration

For tredje iteration valgte vi at undersøge potentialet for to algoritmer, SVM og NB, med samme udgangspunkt i at klassificere online kommentarer som offensivt sprog, ikke offensivt sprog eller hadefuldt sprog. Dertil valgte vi ikke at anvende Streamlit for afprøvning og evaluering af modellen, men i stedet at beregne accuracy ud fra de respektive algoritmer. Det var også i denne del af udviklingsfasen, at vi fandt ud af, at der var en stor mangel på data om hadefulde ytringer i anmeldelser, hvilket gjorde at vi anvendte det aktuelle datasæt. Dette ledte os til at afvige fra målet om at klassificere hadefulde ytringer i anmeldelser, til kun at fokusere på at klassificere anmeldelser som positive, negative eller neutrale.

Den tredje iteration af vores model repræsenterede en betydelig genovervejelse. Efterfølgende versioner har bygget oven på dette grundlag, hvilket resulterede i flere mindre iterationer med små ændringer og justeringer.

5.3 Beskrivelse af koden

Som ovenstående afsnit nævnte, har vores machine learning modeller været udviklet igennem en iterativ proces. En række af genovervejelser og ændring af indgangsvinkel, har ført os til vores nuværende modeller. Dette afsnit vil præsentere vores machine learning modeller og deres opbygning (Se bilag 2). Afsnittet vil starte med at redegøre for de forskellige anvendte libraries. Herefter vil der blive redegjort for vores udvalgte data, samt hvordan vi har forbedret dataene, for at sikre en bedre ydeevne. Til sidst gennemgås hvordan vi har trænet og testet vores modeller.

5.3.1 Opsætning

Machine learning modellerne er udviklet ved hjælp af en række libaries til Python-programmeringssproget. Disse libraries har spillet en afgørende rolle for udviklingen af vores modeller, og vi vil i det følgende afsnit forklare, hvordan vi har anvendt de forskellige libraries.

5.3.2 Scikit-learn

Scikit-learn er et open source Python library, som gør det nemt at udvikle og analysere machine learning modeller. Det tilbyder forskellige machine learning algoritmer, som let kan implementeres i koden, og er derfor et populært værktøj til data science projekter. Én af Scikit-learns fordele er dens brugervenlighed og dokumentation, hvilket har gjort det muligt for os at udvikle machine learning modeller (Muller & Guido, 2016).

5.3.3 Pandas, NumPy & Matplotlib

Pandas, Matplotlib og Numpy er alle open source Python-libraries. Pandas benyttes primært til håndtering og analyse af importeret datasæt. Pandas tilbyder datastrukturen DataFrame, der gør den kompatibel med filformater, såsom CSV og Excel. Dette library har været afgørende for projektet, da det hjalp os med at manipulere og skabe indsigt i vores datasæt (Muller & Guido, 2016).

Matplotlib bliver i denne model anvendt til visualisering af data, da værktøjet tilbyder en række forskellige diagramtyper, såsom histogrammer, søjlediagrammer og linjegrafer. Dette library har været nyttigt til udviklingen af confusion matrices, som vi senere benytter os af til at analysere machine learning modeller.

NumPy er et nyttigt library for NLP-modeller, da det tillader manipulation af datasæt og understøtter multidimensionelle arrays. Vi anvender NumPy, da vi arbejder med store mængder numeriske data og vektorer (Muller & Guido, 2016).

5.3.4 Natural Language Tool Kit

Machine learning modellerne benytter sig af Natural Language Tool Kit (NLTK), som er en Python-pakke der indeholder flere nyttige tekstbehandlingsalgoritmer såsom lemmatization, bag-of-words og N-gram.

5.3.5 Lemmatization

Lemmatization er en teknik inden for machine learning, som har til formål at fjerne ords bøjninger i datasættet. (Muller & Guido, 2016). Bøjninger af ord kan udfordrer modellen da dette medfører støj, hvilket øger chancen for at modellen er tilbøjelig for overfitting.

5.3.6 Bag-of-Words & N-gram

Fælles for NLP-algoritmer er, at de kun kan trænes med numeriske data, hvorved der er en nødvendighed i at omdanne teksten til numeriske værdier. To tilgange til dette er henholdsvis Bag-of-Words og N-gram (Muller & Guido, 2016).

Bag-of-Words tilgangen omdanner teksten til numeriske værdier, ved først at lave et ordforråd over alle de respektive ord som indgår i datasættet. Består datasættet af mange forskellige sætninger, vil hvert enkeltstående ord i datasættet få en numerisk værdi, ud fra antallet af gange det opstår i datasættet. Hvis et ord opstår fem gange i datasættet, får det den numeriske værdi af 5. Tilgangen er dog udfordret af, at der ikke tages højde for sætningens struktur. Eksempelvis vil

sætningerne “Jeg smiler fordi eksamen er afbrudt” og “Eksamen er afbrudt fordi jeg smiler” med Bag-of-Words repræsenteres på samme måde. Dette skyldes at antallet af de respektive ordene i begge sætninger, er identiske. Sætningerne har dog ikke den samme betydning og der er derved risiko for, at noget af semantikken går tabt (Muller & Guido, 2016).

N-gram tilgangen står i forlængelse af Bag-of-Words, men har i stedet til hensigt at tage højde for den semantiske struktur af en given tekst. Navnet N-gram udspringer netop af muligheden for at definere længden af sætninger, som skal repræsenteres. Modsat Bag-of-Words hvor tekst udelukkende repræsenteres som enkeltstående ord, er det med N-gram samtidigt også muligt at repræsentere tekst som sammenslutninger af forskellige ord. Eksempler herpå er unigram og bigram. Unigram repræsenterer en sætning i form af enkeltstående ord, ligesom Bag-of-Words. Derfor deler unigram en sætning op i individuelle ord. Bigram opdeler en sætning i kombinationer af to sammenhænge ord. Eksempelvis vil sætningen “Jeg elsker lange ferier” blive opdelt som, “Jeg elsker”, “elsker lange” og “lange ferier”. Styrken ved tilgangen findes netop i muligheden for at repræsentere teksten som N-grams, altså af forskellige længder. Dog er det væsentligt at tage højde for hvor stor en N-gram der vælges. En større N-gram medfører flere features, og det er altså nødvendigt at fastslå om dette medfører overfitting (Muller & Guido, 2016).

5.4 Databeskrivelse

Vi har benyttes os af et datasæt (Amazon datasættet), som oprindeligt bestod af 568.000 produktanmeldelser, som er indsamlet fra internethandelssiden Amazon. Anmeldelserne var skrevet af forskellige Amazon-brugere og omhandler forskellige produkter eller services, på tværs af en række forskellige kategorier. Grundet begrænsninger i tid og ressourcer forbundet med træningen af vores machine learning modeller, har vi valgt at reducere Amazon datasættet til omtrent 10.000 anmeldelser (Se bilag 3). Begrænsningerne herom omfatter hovedsageligt de computerressourcer som var til rådighed. Vores nuværende computere er muligvis ikke optimalt udstyret til at kunne håndtere komplekse machine learning modeller, der skal bearbejde datasæt bestående af 568.000 datapunkter. I tilfælde af vi valgte at træne en model på det oprindelige datasæt, vil det kræve enorm lang beregningstid.

Det oprindelige datasæt blev valgt på grund af dens variation, samt relevans for vores emne og undersøgelsesfelt. Amazon datasættet bestod oprindeligt af følgende kolonner, hvortil vi tog til overvejelse hvilke, der var relevante til træning af vores model:

1. Id
2. ProductId
3. UserId
4. ProfileName
5. HelpfulnessNumerator
6. HelpfulnessDenominator
7. Score
8. Time
9. Summary
10. Text

Dette resulterede i, at vi udelukkende beholdt kolonnerne Score og Text. Text refererer til selve anmeldelsen som en bruger har skrevet om et givent produkt. Score refererer til brugerens bedømmelsesscore og måles på en skala fra 1-5, hvor 1 anses som det laveste og 5 som det højeste.

For at reducere størrelsen af Amazon datasættet, gennemførte vi en udvælgelsesproces bestående af tilfældig sampling. Heraf filtrerede vi dataene så der var 2000 datapunkter fra hver bedømmelsesscore, altså 2000 datapunkter med en score lig 1, 2000 datapunkter med en score lig 2, osv. Dette gjorde det muligt for os at bevare en repræsentativ del af dataene, samtidig med at vi skabte en betydelig reducering af beregningstid og omfanget af træning. Amazon datasættet bestående af 10.000 produktanmeldelser, består stadig af forskellige variationer af produktanmeldelser med forskellige længder, stilarter og udtryksformer af Text kolonnen. Vi har forsøgt at bevare et repræsentativt datasæt med en rimelig variation, som kan generaliseres til det oprindelig datasæt på 568.000 datapunkter.

5.5 Data preprocessing og labelling

Data preprocessing er et af de indledende trin når man skal arbejde og analysere på store datasæt. Det refererer til en række forskellige applikationer som kan udføres på et datasæt, for at få en ny repræsentation af dataene:

"Learning a new representation of the data can sometimes improve the accuracy of supervised algorithms, or can lead to reduced memory and time consumption."
(Muller & Guido, 2016:132).

Som nævnt tidligere, har vi anvendt et lemmatization library inden for machine learning. Lemmatization er en preprocessing metode, som ofte bliver anvendt når man arbejder med NLP. Det dækker over en række teknikker, som fjerner ords bøjninger for at reducerer ord til deres grundform. Ordet bliver derfor præsenteret på samme måde, som hvis et ord bliver slået op i en ordbog (Muller & Guido, 2016). Nedenstående tabel eksemplificerer hvordan lemmatization fungerer (Se figur 5). Venstre kolonne, Text, viser de rå anmeldelser som da de blev indsamlet fra anmeldelsesplatformen. Højre kolonne, lemma_text, viser resultatet af lemmatization. Det er tydeligt at se, at lemmatization fjerner bøjningerne og samtidigt også fjerner irrelevante ord fra sætninger, såsom "my", "the" og "as".

	Text	Score	label	lemma_text
0	Product arrived labeled as Jumbo Salted Peanut...	1	0.0	Product arrive label Jumbo Salted Peanuts pean...
1	My cats have been happily eating Felidae Plati...	1	0.0	cat happily eat Felidae Platinum two year get ...
2	The candy is just red No flavor Just plan...	1	0.0	candy red flavor plan chewy would never buy
3	This oatmeal is not good Its mushy soft I d...	1	0.0	oatmeal good mushy soft like Quaker Oats way go
4	Arrived in days and were so stale i could no...	1	0.0	Arrived day stale could eat bag
...	...	...	...	...
9995	This was my first purchase of this water It ...	5	2.0	first purchase water taste great bottle thick ...
9996	Water tastes good I typically prefer only pur...	5	2.0	Water taste good typically prefer purified wat...
9997	Always glad to see this water delivered to our...	5	2.0	Always glad see water deliver doorstep drop sp...
9998	This water is really good It is reasonab...	5	2.0	water really good reasonably price mountain sp...
9999	i thot i was satisfied with the spring water i...	5	2.0	thot satisfy spring water buy elsewhere buy br...

Figur 5: Eksempel på før (Text) og efter (lemma_text) anvendelsen af lemmatization.

Udover preprocessing af vores data, har vi også selv labelled Amazon datasættet. Da vores datasæt ikke indebar labels til de respektive anmeldelser, i form af positiv, neutral og negativ, har vi som følge automatiseret en fremgangsmåde til at give anmeldelserne labels. Logikken er baseret på vores egne overvejelser om, hvad der vil være en rimelig måde at klassificere den følelsesmæssige tone i anmeldelserne.

	Score	Text	label
0	5	I have bought several of the Vitality canned d...	2
1	1	Product arrived labeled as Jumbo Salted Peanut...	0
2	4	This is a confection that has been around a fe...	2
3	2	If you are looking for the secret ingredient i...	0
4	5	Great taffy at a great price. There was a wid...	2
...	...	...	...
10109	3	I did not find the smell to be anything but in...	1
10110	5	Unlike microwave popcorn, you cannot hear the ...	2
10111	2	But the taste got sickening quite fast. It als...	0
10112	5	I like spicy snacks and this product fills the...	2
10113	5	This is a great alternative to microwave popco...	2

Figur 6: Udsnit af Amazon datasættet med labels.

Kolonnen Score består som nævnt af numeriske værdier fra 1-5, og fungerer som udgangspunktet for de labels vores model skal trænes på. Da vores hensigt er at klassificere anmeldelserne som positiv, neutral eller negativ, har vi udviklet en funktion, der gennemgår alle værdier i Score kolonnen. Anmeldelser med scoren 1 eller 2, tildeles værdien 0 (Set som værende negativ). Har anmeldelsen en score på 3, bliver værdien 1 tildelt (Set som værende neutral). Ligeledes får anmeldelser med en score større end 3 værdien 2 (Set som værende positiv). Dette gemmes i en ny kolonne, labels, og hver række i Amazon datasættet består nu hhv. af den oprindelige bedømmelsesscore, selve anmeldelsen, og den tildelte label. Dette kan ses i den ovenstående tabel (Se figur 6).

5.6 Data modelling

Vi starter indledningsvist med at udvælge de features fra Amazon datasættet, som skal indgå i træningen og test af modellen. Kolonnen lemma_text udgør input data (X-værdien), teksten der er blevet preprocessed, og kolonnen label udgår klassifikationen (y-værdien) som er de respektive anmeldelsers labels.

Samtidigt har vi også hentet en række Trustpilot-anmeldelser fra Bilka (Se bilag 4), som de endelige modeller efterfølgende evalueres mod. Derfor vælger vi også input data fra dette datasæt (Xx-værdien), samt dertilhørende labels (yy-værdien) (Se figur 7).

```
X = np.array(data["lemma_text"]) ## input data, object (string)
y = np.array(data["label"]) ## label, int64
.........
Xx = np.array(datany["review_text"]) ## input data, object (string)
yy = np.array(datany["label"]) ## label, int64
```

Figur 7: Valg af features.

Dernæst opdeler vi Amazon datasættet til henholdsvis et trænings- og testdatasæt, hvoraf sidstnævnte udgør 20% af vores data til test.

Da vi arbejder med to forskellige algoritmer, har vi lavet funktionen data_modelling som har to parametre. Den første parameter er modeller, som består af listen over de algoritmer som skal trænes. Den anden parameter er vect, hvilket er den anvendte metode til repræsentation af ordene som numeriske værdier. I vores tilfælde gør vi brug af N-gram tilgangen, og hertil opdeler vi sætningerne som hhv. unigrams og bigrams.

Funktionen itererer i forhold til antallet af modeller, i vores tilfælde to gange. Hver respektive model trænes på baggrund af anmeldelsen, som er omdannet til numeriske værdier, samt de dertilhørende labels.

De trænede modeller evalueres først mod de 20% af Amazon datasættet. Samtidigt evalueres modellerne efterfølgende mod det førnævnte Bilka datasæt bestående af Trustpilot anmeldelser. Dette resulterer i en tabel, som måler modellernes præstation. (Se figur 8 & 9)

```
def data_modelling(models,vect):
    result_table=[]
    result_table2=[]
    for i in range(len(models)):
        X_train_vect = vect.fit_transform(X_train)
        X_test_vect = vect.transform(X_test)
        model = models[i]
        #model_name = type(model).__name__
        model.fit(X_train_vect, y_train)
        y_pred_class = model.predict(X_test_vect)

        Xx_test_vect = vect.transform(Xx)
        yy_pred_class = model.predict(Xx_test_vect)

        data={'Accuracy score on TrainTest data':metrics.accuracy_score(y_test, y_pred_class)}
        dataTest={'Accuracy score on another dataset':metrics.accuracy_score(yy, yy_pred_class)}

        result_table.append(data)
        result_table2.append(dataTest)

    df = pd.DataFrame(result_table, index =['Naive Bayes', 'Support Vector Machine'])
    df2 = pd.DataFrame(result_table2, index =['Naive Bayes', 'Support Vector Machine'])
    df_combined = pd.concat([df, df2], axis=1)
    return df_combined
```

Figur 8: Funktion til træning og test af modeller.

5.7 Resultater fra træningssæt

I dette afsnit præsenterer vi modellernes resultater og evne til klassificering af træningsdataene. Vi fremhæver modellernes accuracy og forskellene i deres klassifikationsevner, med udgangspunkt i confusion matrices.

	Accuracy (Train/Test)	Accuracy (Bilka Dataset)
Naive Bayes	0.7095	0.488889
Support Vector Machine	0.6960	0.533333

Figur 9: Tabel over modellernes accuracy-scorer, både for trænings- og valideringsdataene.

Modellerne viser en accuracy på omkring 70% på testdataene og træningsresultaterne tyder på, at NB-algoritmen opnår bedre ydeevne end SVM-algoritmen, da NB-algoritmen korrekt kan klassificere flere anmeldelser. I figur 9 præsenteres modellens træningsresultater, der viser en forskel på 1,35% i accuracy-score til fordel for NB-algoritmen, sammenlignet med SVM-algoritmen.

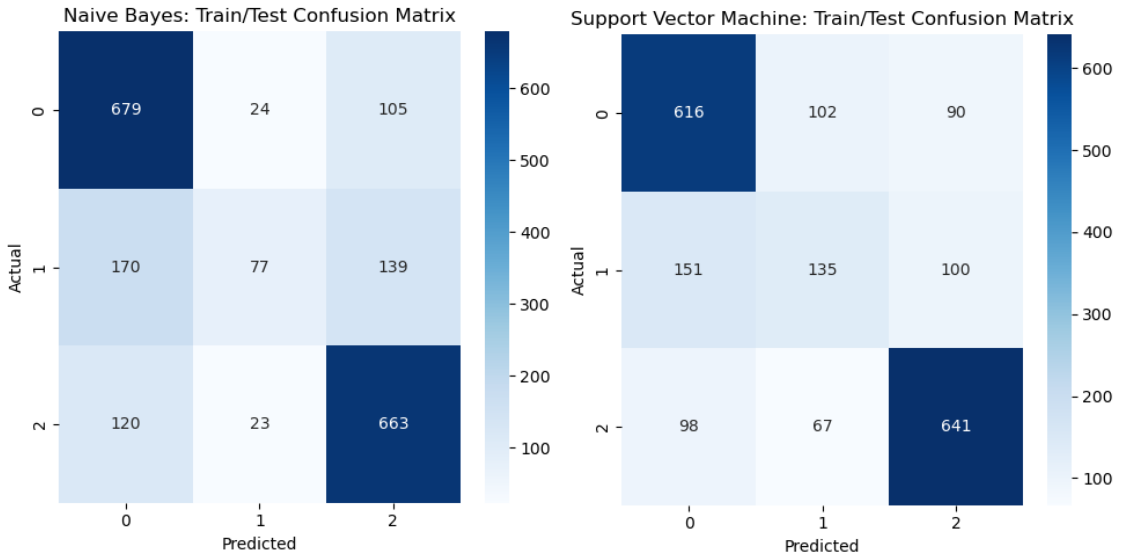
I en undersøgelse foretaget af Sida Wang & Christopher D. Manning (2012), forskere indenfor datalogi fra Standford University i USA, blev NB og SVM's ydeevne analyseret, og deres fordele og ulemper diskuteret, samt i hvilke anvendelsesscenarier det kan være fordelagtigt at benytte disse algoritmer. Artiklen konkluderer, at SVM generelt er overlegen til klassifikationen af datasæt, der består af lange sætninger. Derimod er NB bedre til klassifikation af mindre datasæt med korte sætninger eller uddrag.

```
In [26]: avg_num_words = data['Text'].apply(lambda x: len(x.split())).mean()

         print('Average number of words:', avg_num_words)

         Average number of words: 89.1065
```

Figur 10: Gennemsnit af ord per anmeldelse.

Amazon datasættet, vi benytter til vores træningsmodel, består af ca. 89 ord per anmeldelse (Se figur 10). Vores testresultater understøtter derfor ikke Wang & Mannnig's (2012) konklusion om, at SVM-algoritmen er fordelagtig til klassifikation af lange sætninger.

Man kan argumentere for, at en forskel på 1,35% i accuracy-score måske ikke udgør ikke en betydelig forskel i modellernes evne til at klassificere anmeldelsernes bedømmelse. Derimod kan selv små forskelle i modellernes klassifikationsevner have indflydelse på deres ydeevne i praksis.



Figur 11: Confusion matrix for hver model over træningsdataene.

I figur 11 præsenteres en confusion matrix for hver model, der har klassificeret testdataene. Tabellerne viser distributionen af modellens evne til at klassificere anmeldelserne i Amazon datasættet. Vi har beregnet dataene fra begge confusion matrices, og udviklet en model, der viser den isolerede accuracy fra hver klasse af anmeldelser.

Performance (Test/Train Datasæt)	Positive Anmeldelser	Neutrale Anmeldelser	Negative Anmeldelser
Naive Bayes Model	$(663/808) * 100 = 82,05\%$	$(77/386)*100 = 19.90\%$	$(679/806) * 100 = 84,24\%$
Support Vector Machine Model	$(641/806) * 100 = 79,52\%$	$(135/386) * 100 = 34,97\%$	$(616/808) * 100 = 76,23\%$

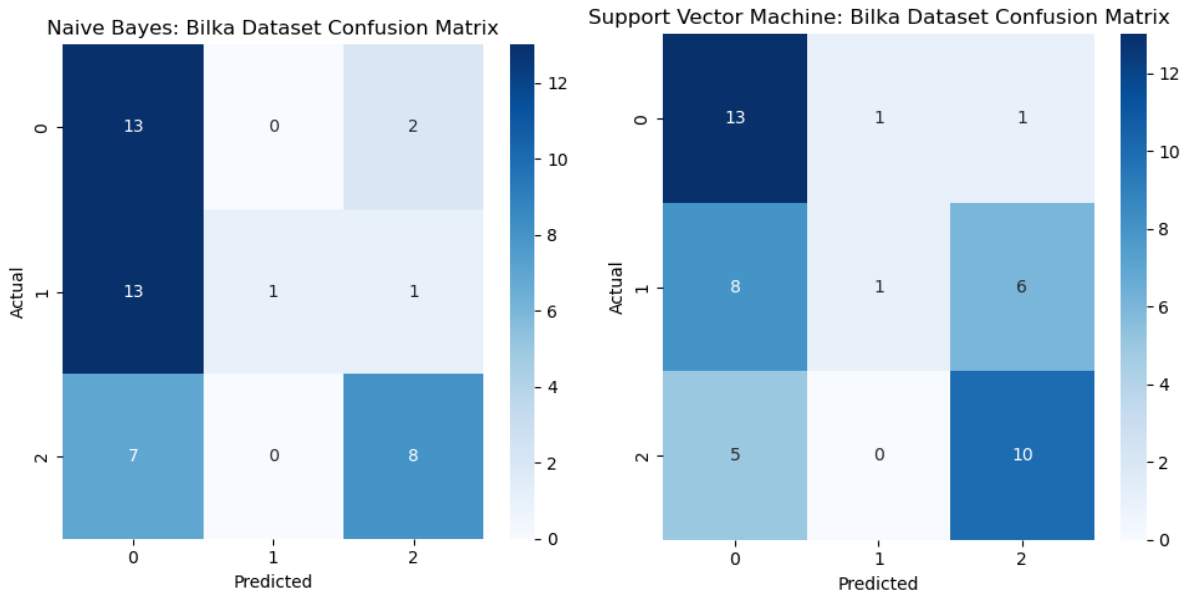
Figur 12: Procentvis fordelingen af klassifikation over træningsdataene.

Figur 12 viser at begge modeller præsterer bedst i klassificering af positive og negative anmeldelser. Derimod klarer modellerne sig dårligere til klassificeringen af neutrale anmeldelser. NB-modellen opnår kun en accuracy-score på 19.90%, mens SVM-modellen opnår en accuracy-score på 34,97% i klassificeringen af neutrale anmeldelser. Som følge af de foreløbige resultater, er vi derfor usikre på modellernes egenskaber til klassificering af virkelige anmeldelser, da testdataene viser, at modellerne ikke er egnet til klassificering af neutrale anmeldelser.

5.8 Modellens resultater

Accuracy-scoren for de trænede modeller, er som nævnt tæt på identisk. Anderledes ser det dog ud når modellerne evalueres mod Bilka datasættet (Se figur 9). Her opnår vores NB-model en accuracy på 48,88%, hvilket er 22,07% lavere end resultatet fra Amazon datasættet. Vores SVM-model opnår en 53,33% accuracy på Bilka datasættet, hvilket er 16,27% lavere end resultatet fra Amazon datasættet. Dog er SVM-modellens accuracy-score 4,44% højere på Bilka datasættet, i sammenligning med NB-modellen.

I figur 13 præsenterer vi én confusion matrix for hver model, som viser distributionen af modellens evne til at klassificere anmeldelserne i Bilka datasættet. De respektive klasser i Bilka datasættet består hver af 15 elementer. På baggrund af begge confusion matrices har vi igen udviklet en model, som viser den isolerede accuracy for hver kategori af anmeldelser i Bilka datasættet.



Figur 13: Confusion matrix for hver model over valideringsdataene.

Retter vi først blikket mod NB kan vi se, at modellen korrekt har klassificeret 13 ud af 15 anmeldelser som værende negative. De resterende 2 anmeldelser blev klassificeret som værende positive, hvoraf ingen blev klassificeret som værende neutrale. Kigger vi isoleret på klassifikation af negative tekster, opnår modellen en accuracy-score på 86,66% (Se figur 14). Resultatet indikerer at NB-modellen i høj grad er i stand til at genkende negative tekster.

Af de 15 neutrale tekster bliver 13 klassificeret som negativ og 1 som værende positiv. Det er således kun én af teksterne, som korrekt bliver klassificeret som neutral, hvilket giver en isoleret accuracy-score på 6,66% (Se figur 14). NB-modellen har med andre ord en stærk hældning mod at klassificere neutrale tekster som negative.

Af de 15 positive tekster bliver 8 korrekt klassificeret, hvoraf de resterende 7 klassificeres som negativ. Dette resulterer i en isoleret accuracy-score på 53,33%. Modellen er således langt bedre til at klassificere positive tekster, sammenholdt med de neutrale. Den samlede lave accuracy-score er dog et resultat af to faktorer. Modellen er ikke i tilstrækkelig grad i stand til at klassificere neutrale tekster, og ved positive tekster klassificeres der kun korrekt i omtrent halvdelen af tilfældene. Med andre ord evner NB-modellen kun tilfredsstillende at genkende negative tekster.

Performance (Bilka Datasæt)	Positive Anmeldelser	Neutrale Anmeldelser	Negative Anmeldelser
Naive Bayes Model	$(8/15) * 100 = 53,33\%$	$(1/15) * 100 = 6,66\%$	$(13/15) * 100 = 86,66\%$
Support Vector Machine Model	$(10/15) * 100 = 66,66\%$	$(1/15) * 100 = 6,66\%$	$(13/15) * 100 = 86,66\%$

Figur 14: Procentvis fordelingen af klassifikation over valideringsdataene.

Selvom SVM-modellens samlede accuracy-score er 4,44% højere, indikerer confusion matricen tilsvarende problemstillinger (Se figur 14). Modellen klassificerer ligeledes korrekt 13 af de 15 tekster som værende negative, hvortil de 2 sidste klassificeringer falder på hhv. neutral og positiv. Der er her igen tale om en isoleret accuracy-score på 86,66% for klassificeringen af negative tekster. Udfordringerne opstår igen ved modellens evne til klassifikation af neutrale og positive tekster. Af de 15 neutrale tekster, bliver kun én korrekt klassificeret. Hertil bliver 8 klassificeret som negativ, og 6 som værende positiv. Modellen opnår ligeledes en isoleret accuracy-score på 6,66%, og har tilsvarende med NB derfor en meget lav evne til at genkende neutrale tekster. Dog er der her en mere ligelig fordeling blandt de negative og positive klassificeringer.

Som det fremgår i figur 14, opnår modellens klassifikation af de positive tekster en isoleret accuracy-score på 66,66%, hvor 10 af 15 tekster korrekt bliver klassificeret som værende

positive, og hvor de resterende klassificeres som værende negative. Selvom dette er en forbedring sammenholdt med NB-modellen, er den procentvise genkendelse fortsat lav.

Vores resultater indikerer at begge modellers evne til at generalisere kan betvivles. Dette skal ses i lyset af modellernes væsentligt lavere samlede accuracy-score, når der evalueres på vores Bilka-datasæt. Figur 14 fremhæver den fortsatte udfordring med at klassificere neutrale tekster, hvor der for begge modeller som nævnt kun opnås en isoleret accuracy på 6,66%. Modellerne har samtidigt en markant højere tendens til at klassificere negative tekster korrekt, hvortil de positive tekster i væsentligt lavere grad bliver genkendt. Dette står i kontrast til resultaterne fra evalueringen mod det oprindelige datasæt, og underbygger derved modellernes utilstrækkelige generaliserbarhed.

6. Diskussion

I dette afsnit ønsker vi at diskutere de utilsigtede effekter ved udviklingen af en NLP-model, til at evaluere anmeldelser. Denne diskussion vil være baseret på de resultater vi har fået ud fra vores analyse, som er baseret på vores fremgangsmåde. Derudover vil vi diskutere validering af Amazon datasættet, fra et videnskabsteoretisk perspektiv. Afslutningsvis vil vi diskutere projektets fejlkilder, samt hvordan vores model vil have præsteret, såfremt vi havde rettet op på de nævnte mangler.

6.1 Udviklingen og evalueringen af vores model

Analysen af de to klassificeringsalgoritmer, præsenterer modellernes evne til at klassificere træningsdata og ny data. Begge modellers evne til at klassificere anmeldelser kan betvivles, baseret på de samlede accuracy-scorer. For selvom NB og SVM kun kan skelnes mellem en snæver margin, bliver modellerne stadig udfordret ved nyt input. Hvordan kunne en anden tilgang have gavnet evaluering af modellerne? Og kunne sådan en tilgang have gjort modellerne mere generaliserbar?

Som vist i analysen fremhæver confusion matricerne, at der er en fordeling af modellernes klassifikation af anmeldelserne i datasættet. Begge klarer sig bedst i klassifikationen af positive og negative anmeldelser, men har svært ved at klassificere neutrale anmeldelser. Dette rejser tvivl om modellernes evne til at klassificere virkelige anmeldelser, da de ikke egner sig godt til, identificerer neutrale anmeldelser. Dette bringer spørgsmålet om, hvorvidt en binær

klassifikation kunne have været mere hensigtsmæssig i udviklingen af modellen, fremfor multi-klassifikation. Binær klassifikation vil klassificere anmeldelser som enten positive eller negative. Anmeldelser med en bedømmelsesscore lig 3, vil derfor blive fjernet fra datasættet, da de ikke vil passe ind i hverken positiv eller negativ. Dette vil have betydet, at vi skulle ændre fremgangsmåden ved vores data labelling proces ligeledes:

- 1-2 i Score: Negativ
- 4-5 i Score: Positiv

Denne fremgangsmåde vil have fungeret godt, hvis formålet var at skelne mellem en positiv eller negativ følelsesmæssig tone i anmeldelser. Set i lyset af vores nuværende fremgangsmåde med multi-klassificering, vil binær klassificering ikke bidrage med en særlig nuanceret klassifikation eller fremhæve situationer hvor der er en neutral følelsesmæssig tone. I dette tilfælde, hvor anmeldelser har brugeres tilfredshed på en score fra 1 til 5, vil multi-klassificering fange en bredere vifte af udtrykte følelser og en mere nuanceret klassificering. En anden relevant tilgang vi kunne have overvejet, var at anvende bedømmelsescoren mere nyttigt. For netop at identificere flere udtrykte følelser i anmeldelser, kunne vi udvide antallet af følelsesmæssige klasser fra tre til fem. I givet fald, vil vores data labelling proces se således ud:

- 1 i Score: Meget negativ
- 2 i Score: Negativ
- 3 i Score: Neutral
- 4 i Score: Positiv
- 5 i Score: Meget positiv

Denne fremgangsmåde vil kunne påpege en bredere vifte af udtrykte følelser og give en mere nuanceret klassificering. Ved at inkludere "Meget negativ" og "Meget positiv" vil dette bidrage til en mere detaljeret forståelse for brugernes anmeldelser. Derimod vil dette dog kræve endnu flere ressourcer at implementere i vores modeller, hvilket vi som nævnt ikke har adgang til. Dette vil samtidigt også vække bekymring for vores nuværende modellers præstation. Med de resultater, der blev vist fra analysen, kan man diskutere hvorvidt vores modeller er tilstrækkelig nok til at kunne give repræsentative klassificeringer af neutrale anmeldelser. Hvilket derfor også fremmer tvivlen om modellernes egenskab til at kunne skelne mellem flere klasser end tre. Der

er nemlig udfordringer med klassifikation af neutrale anmeldelser, hvor begge modeller kun opnår en isoleret accuracy-score på 6,66%. Implementeringen af binær klassificering vil have simplificeret klassifikationen og vil muligvis have reduceret kompleksitet og beregningstiden for modellerne. Dette vil også have været fordelagtigt, set i kontekst af de begrænsninger og udfordringer vi har nævnt.

Efter at have overvejet fordelene og ulemperne ved henholdsvis binær- og multi-klassificering, rettes fokus nu mod spørgsmålet om vores modeller, og resultaterne herom, kan være påvirket af andre faktorer. For hvad hvis resultaterne fra den nuværende fremgangsmåde, er baseret på en for simpel- eller overspecialiseret model? Vi har i afsnittet om NLP's indre mekanismer og processer klargjort teknologiens formål, altså dens tilsigtede effekter. Vi har dog ikke drøftet de utilsigtede virkninger ved udviklingen og brugen af NLP-modeller, såsom over- eller underfitting af en model.

> "Hovedsigtet med punktet er identifikation og analyse af en teknologis utilsigtede effekter, og især de uønskede blandt disse." (Jørgensen, 2020: 45).

Den muligvis største uønskede effekt af at udvikle en NLP-model, er hvis modellen resulterer i at blive overfitted. Overfitting opstår når man tilpasser en model, for tæt til træningssættes data. Dette vil resultere i en model, der er højeffektiv på træningssættets data, men fejler når den prøver at generalisere til nyt data (Muller & Guido, 2016). Den bliver altså overspecialiseret i de underliggende mønstre og udsving af træningssættet. Den primære risiko ved dette er, at den mangler evnen til at generalisere dens ydeevne til nye og usete data.

Dette resulterer i at modellen ikke vil være i stand til effektivt at klassificere forskelligt input. Overfitting vil også lede til en høj varians i modellens forudsigelser, hvor at resultaterne vil være usammenhængende eller ustabile når den udsættes for nyt data. Dette vil medvirke til en stor upålidelighed når modellen anvendes i praksis (Muller & Guido, 2016). Selvom vores analyse af modellernes ydeevne på træning- og ny data påpeger, at modellerne kan være overfitted, er det dog ikke garanteret. Det er vigtigt at pointere, at de ovenstående indikatorer til overfitting ikke er afgørende. Der er ligeledes også mange andre faktorer, som skal tages i mente for en videreanalyse herom.

Derimod kan der også være risiko for, at vores modeller er underfitted. Underfitting opstår hvis en model er for simpel, da den ikke kan lære sammenhængen mellem input og output, i dette

tilfælde lemma_text og label. Muller & Guido (2016) beskriver og eksemplificere underfitting således:

> "On the other hand, if your model is too simple - say, 'Everybody who owns a house buys a boat' - then you might not be able to capture all the aspects of and variability in the data, and your model will do badly even on the training set." (Muller & Guido, 2016: 28).

Effektive NLP-modeller har brug for evnen til at kunne analysere hele sætninger, bestående af naturligt sprog. Som nævnt tidligere, er det naturlige sprog yderst komplekst og dækker over en bred vifte af lingvistiske felter. Bekymringen for, at vores modeller er underfitted opstår ved vores valg af klassifikationsalgoritmer; Naive Bayes og Support Vector Machine. Disse to algoritmer kan muligvis anses som værende for "simple", og har svært ved at bearbejde data i form af naturligt sprog, som kan være meget komplekst. Dette skyldes da NB antager, at der er en uafhængighed mellem features, og kan blive udfordret ved at finde kontekstuelle relationer og afhængigheder i det naturlige sprog. SVM vil muligvis også udstå, da den finder lineære hyperplaner, men muligvis fejler når den forsøger at finde sammenhænge ved ikke-lineære hyperplaner. Det kan altså påpeges, at de valgte algoritmer kan blive udfordret ved at klassificere data i form af naturligt sprog, i henseende til, at de er begrænset ved at tillærer sig kompleksiteten ved det naturlige sprog.

Det kan pointeres, at over- og underfitting er utilsigtede effekter og designfejl, med stor risiko for at en model er ubrugelig. I sidste ende er det ultimative mål at kunne generalisere en maskinlæringsmodel til ny data, så den kan blive anvendt i virkelige verdenssituationer, hvor der skal laves forudsigelser og klassificering af data. Men kan vores modeller generalisere effektivt til nye data?

Som nævnt i vores analyse, har vores modeller en accuracy-score på 53%, når den bliver udsat for nyt input. Dette indikerer at vores modellers præsentation kun fremviser en marginal forbedring, sammenlignet med tilfældigt gætværk. Isoleret set, henviser en accuracy-score på 53% nødvendigvis ikke at en model er ineffektiv ved generalisering, men det giver dog anledning til bekymring om modellens evne herom. Trods for, at accuracy-scoren ikke er betydelig høj, er det vigtigt at påpege, at modellerne muligvis har tillært sig, mønstrer fra Amazon datasættet. Dette er dog ikke ensbetydende med, at de kan generaliseres til forskellige og nye datapunkter. Det er derfor udfordrende at vurdere, hvorvidt vores modeller kan

generaliseres til nyt data. Vores manglende test af modellerne på forskelligartede datasæt begrænser vores vurdering af deres generaliseringsevne. Som nævnt, har vi kun valideret modellerne ud fra Bilka datasættet, men ikke med andre eksterne eller varieret datasæt. Det er derfor vigtigt at udføre yderligere analyse og undersøgelse, såsom Cross-Validation, for at få mere indsigt i modellerne og derved en sikker konklusion om hvorvidt modellerne kan generalisere til nyt data (Muller & Guido, 2016).

6.2 Potentielle problematikker med valg af datasæt

Vores valg af datasæt, herunder valg af features, italesætter en anden utilsigtet effekt for NLP-modeller. Som nævnt er teknologiens formål at være højt betonet af generaliserbarhed. Dette kræver netop, at modellen formes efter virkeligheden, og derfor er det væsentligt at datasættet er fundamenteret heraf. Datasættet har således til formål at repræsentere virkeligheden, hvilket danner grundlag for modellens beslutningstagen. Uoverensstemmelse mellem datasættets repræsentation, og den faktiske virkelighed, resulterer i en klassifikationsmodel, som fejlagtigt træffer beslutninger.

Vi har tidligere redegjort for, hvordan vores modeller er trænet på følgende labels:

- 1-2 i Score: Negativ
- 3 i Score: Neutral
- 4-5 i Score: Positiv

Anmeldelsernes sentimentale karakter, er alene blevet fastlagt ud fra bedømmelsesscoren. Med andre ord baseres beslutningsprocessen entydigt på den antagelse, at scoren afspejler den faktiske virkelighed; nemlig, at en vred tone f.eks. udelukkende er at finde i anmeldelser med en score på 1 eller 2. Selvom det formodningsvist ofte er tilfældet, bringer det dog med sig en stor usikkerhed.

Den positivistiske tilgang er som nævnt udgangspunktet for vores valg af datasæt og features. Heraf opstår der et interessant spørgsmål: Kan der alene på baggrund af kvantificerbare målinger og objektive data skabes grobund for sand viden, når en teksts sentimentale karakter skal fastslås? Med afsæt i vores fremgangsmåde, er svaret nej. Vores fremgangsmåde negligerer nemlig fuldstændigt, at sproget er betonet af kompleksitet og nuanceringer. Ved at reducere

valget af features til én bedømmelsesscore, risikerer disse elementer at udstå fra den endelige sentimentale karakter.

Den fænomenologiske tilgang præsenterer en interessant vinkel på problemstillingen. Hvor vi med den positivistiske fremgangsmåde vægtede objektive data, i form af bedømmelsesscoren, vil fænomenologiens verdenssyn bidrage til en subjektiv anerkendelse af dataene. Sproget er som nævnt ikke entydigt, og derfor vil en kontekstafhængig tilgang højne resultatet. Derfor er det væsentligt at tage højde for det subjektive individs fortolkninger og oplevelser af sproget. En given tekst kan anses som værende positiv i én sammenhæng, og samtidigt være negativ i anden sammenhæng. Dette understreger nødvendigheden i at indarbejde en bredere forståelse af konteksten, herunder brugernes intentioner, for på den måde at opnå en mere korrekt validering af tekstens sentimentale karakter.

Det kan samtidigt diskuteres i hvor stor grad, at bedømmelsesscoren skal afspejle en given teksts sentimentale karakter. En alternativ fremgangsmåde til validering kunne være brugen af labels med større nuance. Dette kunne eksempelvis udforme sig gennem kvalitative beskrivelser af teksternes sentimentale karakter. I stedet for blot at forholde sig til om teksten er positiv, negativ eller neutral, vil denne fremgangsmåde i større grad tage forbehold for sprogets mangfoldighed. Hertil vil ekspertviden i forlængelse af teori kunne bidrage til en større nøjagtighed. Bedømmelsesscoren vil ikke nødvendigvis udestå, men vil i stedet fungere som en supplering til udviklingen af modellerne.

Træningsdatasættets omfang, har dog også en betydningsfuld karakter. I takt med at datasættets størrelse vokser, øges mængden af datapunkter, der skal bearbejdes under træningen af modellen. Store datasæt medfører ikke blot længere træningstider for modellerne, men opretholder samtidigt et større menneskeligt tidsforbrug, når datapunkterne skal tildeles labels.

Derfor kan der argumenteres for, at en overvejelse af datasættets størrelse er nødvendig for at finde en tilfredsstillende accuracy-score. På den ene side kan større datasæt bidrage til forøgede ressourcer, men på den anden side kan større datasæt resultere i modeller med høj generaliserbarhed.

Det menneskelige tidsforbrug, i henhold til at give datapunkter labels, har som resultat af den automatiserede tildeling af labels, ikke været en udfordring. Ved at automatisere denne proces,

bliver store datasæt mere håndgribelige, men der stilles samtidigt også krav til, at processen foregår med hensyn til modellens formål. Her kan man måske stille sig selv spørgsmålet: Afspejler den automatiserede tildeling af labels den virkelighed, som datasættet repræsenterer? Set i lyset af vores projekt, vil spørgsmålet være om bedømmelsesscoren, for en anmeldelse, rent faktisk afspejler en teksts sentimentale karakter. De fleste vil nok medgive, at forbrugere, som oftest afspejler bedømmelsesscoren, med teksten i anmeldelsen. Dette er dog ikke nødvendigvis altid tilfældet. Individet kan eksempelvis afgive en anden bedømmelsesscore, end hvad hensigten var. Som nævnt kan det også diskuteres, om en bedømmelsesscore fra 1-5 alene kan skabe fundament for anskuelsen af tekstens sentimentale karakter. Den sentimentale karakter af sætningen: "Musikken var høj og bassen kunne mærkes", afhænger af konteksten. Er man til koncert, er der formodentligt tale om en positiv oplevelse. Ændres konteksten derimod til en indkøbstur i supermarkedet, kan det formodes at oplevelsen var negativ.

6.3 Tilsigtede og utilsigtede effekter ved brugen af NLP

Som nævnt er vores modeller svækket af begrænsede computerkraft, hvilket har medført, at vi måtte vælge et mindre datasæt, end først tiltænkt. Endnu en fejlkilde udspringer af det valgte datasæt, i henhold til vores evaluering af modellernes nøjagtighed. I stedet for kun at evaluere vores Bilka datasæt, kunne det være hensigtsmæssigt at inddrage en variation af forskellige datasæt. Et nærliggende spørgsmål vil således være, om den entydige evaluering af Bilka datasættet giver et retvisende resultat, af de trænede modellers nøjagtighed? Det havde således været interessant at evaluere vores modeller med anmeldelser fra øvrige detailvirksomheder, eksempelvis Netto eller andre danske supermarkedskæder. Samtidigt havde det også været interessant at evaluere præstationen med brugen af forskellige datasæt, som afviger fra træningsdataene. Dette kunne eksempelvis udforme sig ved, at vi inddrog Twitter kommentarer under testen. Da hensigten fortsat vil være identifikationen af tekstens sentimentale karakter, vil en sådan test kunne give indsigt i hvorvidt vores modeller, i for høj grad, er betonet af en kontekst afhængighed.

Evalueringen af modellerne er alene baseret på deres accuracy-score. Vi har således kun opnået indsigt i modellernes nøjagtighed, på baggrund af de valgte datasæt. Og netop valget af datasæt, herunder hvordan vi tildeler labels, har fungeret som udgangspunktet for analysens resultater. Spørgsmålet er, om resultaterne alene kan begrundes herpå? Gennem projektets forløb har vi

nemlig ikke haft til overvejelse, om modellernes parametre i Python kunne have en betydning. Derfor vil en oplagt tilgang være brugen af Cross-Validation, hvilket gør det muligt at evaluere modellerne med forskellige parametre og heraf få indsigt i hvilke der opnår den bedste accuracy. Eksempelvis kunne vi have afprøvet, hvor stor en procentvis af datasættet, som modellerne skulle trænes på. Vil en større mængde medføre bedre resultater, eller er 20% af datasættet det mest optimale? Som nævnt omdannes teksten som modellen skal trænes med, til numeriske værdier i form af unigrams og bigrams. Giver dette de mest nøjagtige resultater, eller vil det være bedre at omdanne teksterne i form af unigrams? Begge eksempler, har vi på nuværende tidspunkt ikke indsigt i, hvilket understreger nødvendigheden i at undersøge hvilke parametre, der giver det bedste resultat.

En af vores overvejelser vedrørende fejlkilder, omhandler vores test af modellen i en praktisk sammenhæng. Det kunne have gavnet vores indsigt for relevansen af projektet at afprøve modellen på fokusgrupper eller virksomheder.

Hvis vi havde opfyldt de præsenterede kriterier og rettet op på fejlkilderne, er det muligt, at vores modeller var blevet bedre og kunne anvendes i en praktisk sammenhæng. Hvis det var lykkedes os at udvikle en model, der kunne klassificere anmeldelser korrekt med en relativ høj nøjagtighed, ville vi have haft mulighed for at videreudvikle på vores idé og skabe en mere avanceret og specifik model.

Det kunne have været spændende at arbejde videre med vores idé, hvor modellen skulle kunne være i stand til at kategorisere anmeldelser, ved at genkende særlige keywords. Ved at genkende specifikke ord og deres synonymer ville modellen have været i stand til identificere, om en anmeldelse omhandlede emner som levering, support, service eller en bestemt produktgruppe. En virksomhed kunne benytte modellen til at identificere hvilke områder, inden for deres virksomheds drift, der fungerer godt og hvilke, der ikke fungerer godt.

Vi forestiller os, at den videreudviklede model vil være relevant i praksis. Dog er vi er ikke overbeviste om, at vores nuværende model er tilstrækkelig relevant. Selvom vi optimerer vores nuværende modeller til at kunne evaluere anmeldelser korrekt, er vi usikre på, om teknologien vil have nogen indvirkning på, hvordan virksomheder eller individer vurderer anmeldelser. Bedømmelsescoren fungerer allerede godt, da den kan give en mere nuanceret indsigt i en virksomheds præstation end anmeldelser, der kun består af tre klassifikationer: neutral, negativ

eller positiv. Som tidligere nævnt har vi ikke afprøvet modellen i en praktisk sammenhæng, eller opnået indsigt gennem kvalitative- eller kvantitative analyser. Vi tvivler dermed om, hvorvidt vi har skaffet tilstrækkelig empiri og betvivler de nuværende modellers relevans, som et værktøj i samfundet.

7. Konklusion

I dette projekt har vi undersøgt, hvordan man kan benytte Natural Language Processing som et teknologisk værktøj, til evaluering af online anmeldelser. Artiklen understøttes af teorier om machine learning i Python, samt anvendelsen af TRIN-modellen. Gennem en iterativ design proces, har vi reflekteret over udviklingen af en Natural Language Processing model, til vurdering af online anmeldelser. Gennem vores erfaringer har vi opdaget, at selvom valget af datasæt og tildeling af labels er vigtige faktorer, er det er nødvendigt at gøre sig overvejelser om hvilke parametre, i relation til træning af modellen, der giver den bedste accuracy-score. Selvom resultaterne for vores modeller ikke er tilfredsstillende, har det givet os en væsentlig indsigt i udviklingsprocessen af Natural Language Processing-modeller, til evaluering af online anmeldelser. Artiklen konkluderer, at det er tvivlsomt om en lignende Natural Language Processing-model vil være et effektivt værktøj, til evaluering af online anmeldelser i praksis. Bedømmelsescore fungerer allerede godt, da de i forvejen giver en nuanceret indsigt i en virksomheds præstation. Vi er dog optimistiske med hensyn til potentialet for Natural Language Processing som et værktøj inden for anmeldelsesfeltet, til at detektere og kategorisere specifikke aspekter af en virksomheds ydelser. Vi anerkender derfor teknologiens evne til at analysere og forstå naturligt sprog, hvilket kan skabe en dybere indsigt i, hvordan brugere oplever forskellige aspekter af en virksomheds ydelser.

Litteraturliste

Amazon Product Reviews. (u.d.). Kaggle. Tilgået d. 26. maj, 2023 fra https://www.kaggle.com/datasets/jillanisofttech/amazon-product-reviews

Anmeldelsesinvitationer - Trustpilot Business. (u.d.). Trustpilot. Tilgået d. 26. maj, 2023 fra https://dk.business.trustpilot.com/features/review-invitations

Alpaydin, E. (2014). Introduction to machine learning. MIT Press.

Chowdhary, K. R. (2020). Fundamentals of Artificial Intelligence (1st ed. 2020.). Springer India.

Cox, K. (2021). Business Analysis, Requirements, and Project Management: a guide for computing students. CRC Press. https://doi.org/10.1201/9781003168119

Ghose, A., & Ipeirotis, P. G. (2011). Designing ranking systems for consumer reviews: The impact of review subjectivity on product sales and review quality. Information Systems Research, 22(4), 784-801. https://doi.org/10.1287/isre.1100.0325

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.

Hello World - GitHub Docs. (u.d.). GitHub Docs. Tilgået d. 11. april, 2023 fra https://docs.github.com/en/get-started/quickstart/hello-world

Holm, A. (2011). Videnskab i virkeligheden: en grundbog i videnskabsteori. Samfundslitteratur.

Jiang, H. (2021). Machine learning fundamentals : a concise introduction. Cambridge University Press. https://doi.org/10.1017/9781108938051

Jo, T. (2021). Simple Machine Learning Algorithms. Springer EBooks, 69–90. https://doi.org/10.1007/978-3-030-65900-4_4

Jørgensen, N. (2020). Digital signatur. En eksemplarisk analyse af en teknologis indre mekanismer og processer. Roskilde Universitet.

Moltzau, A. (2021). What is Kaggle? - DataSeries - Medium. Medium. Tilgået d. 21. april, 2023 fra https://medium.com/dataseries/what-is-kaggle-4751e384e916

The Surprising Case for Stronger E-commerce Growth - Morgan Stanley. (u.d.). Morgan Stanley. Tilgået d. 26. maj, 2023 fra https://www.morganstanley.com/ideas/global-ecommerce-growth-forecast-2022

Muller, A. C., & Guido, S. (2016). Introduction to machine learning with Python. O'Reilly.

Olesen, F., Bille, M., & Riis, S. (2021). Postfænomenologi. I P. Danholt, & C. Gad (red.), Videnskab, teknologi og samfund: En introduktion til STS (s. 121-140). Hans Reitzels Forlag.

Park, S. & Nicolau, J. L. (2015). Asymmetric effects of online consumer reviews. Annals of Tourism Research, 50, 67–83. https://doi.org/10.1016/j.annals.2014.10.007

Risch, J., & Krestel, R. (2020). Toxic Comment Detection in Online Discussions. SpringerLink. https://link.springer.com/chapter/10.1007/978-981-15-1216-2_4

Schön, D. A. (1988). Designing: Rules, types and worlds. Design Studies, 9(3), 181–190. https://doi.org/10.1016/0142-694X(88)90047-6

Scikit-learn developers (u.d.). 1.9 Naive Bayes. Tilgået d. 02. maj, 2023 fra https://scikit-learn.org/stable/modules/naive_bayes.html

Stanford Natural Language Processing Group. (u.d.). Stanford NLP software. Tilgået d. 09. april, 2023 fra https://nlp.stanford.edu/software/

Streamlit, Inc. (u.d.). Streamlit documentation. Tilgået d. 12. maj, 2023 fra https://docs.streamlit.io/

Suthaharan, S. (2016). Machine Learning Models and Algorithms for Big Data Classification. Springer EBooks. https://doi.org/10.1007/978-1-4899-7641-3

Wang, S., & Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. Proceedings of the 50th Annual Meeting of the Association for