JÄMFÖRELSE AV MASKININLÄRNINGSMODELLER INOM SPRÅKTEKNOLOGI VID IDENTIFIERING AV RISKFAKTORER FÖR HJÄRT- OCH KÄRLSJUKDOMAR

[Dokumentets underrubrik]

Sammanfattning

[Väck läsarens intresse med en intressant sammanfattning. Det är vanligtvis en kort sammanfattning av dokumentet.
När du vill lägga till innehåll klickar du här och börjar skriva.]

Matilda Wik
[E-postadress]

Referat

Innehållsförteckning

1. Inledning

Språkteknologi har funnits sedan datorer blev vanligare under mitten av 1900-talet [1]. Trots att språkteknologi inte är något nytt fenomen har den utvecklats i snabb takt under de senaste åren. Till exempel har det blivit vanligare med talassistenter, och chattbottar klarar av allt mer avancerade uppgifter. En bidragande faktor till utvecklingen har varit användningen av maskininlärning och speciellt djupinlärning, tillsammans med språkteknologi.

Ett område där språkteknologi används mycket är medicin. Digitaliseringen av medicinska data såsom patientjournaler har medfört att forskare har en bredare tillgång till data att studera. Att gå igenom och analysera sjukdomsrelaterade data såsom elektroniska patientjournaler skulle vara ett väldigt tidskrävande arbete för en människa. För en dator går detta betydligt mycket snabbare med hjälp av språkteknologi. Utöver språkteknologi använder datorer sig också av maskininlärning för att hitta samband mellan data som skulle vara mycket svårt för en människa att hitta. Att använda dessa två tillsammans blir därför ett kraftfullt verktyg när det kommer till att identifiera riskfaktorer och att förutsäga sjukdomar.

Bland sjukdomar är hjärt- och kärlsjukdomar den vanligaste dödsorsaken i världen. Genom att förutse vem som kommer drabbas och tidigt identifiera personer som löper större risk att drabbas kan antalet dödsfall reduceras. I den här avhandlingen kommer jag att jämföra hur väl maskininlärningsmodellerna stödvektormaskiner och återkopplade neurala nätverk fungerar tillsammans med språkteknologi vid förutsägelse av hjärt-och kärlsjukdomar.

2. Språkteknologi

Språkteknologi innebär en dators förmåga att förstå, analysera och producera språk [2]. Språkteknologi är en underkategori till artificiell intelligens och har ett flertal olika användningsområden. Den används bland annat i sökmotorer, stavningskontroller och vid röstigenkänning. För att en dator ska kunna förstå och tillämpa mänskligt språk behöver den bearbeta språket och utföra olika uppgifter (eng. tasks).

2.1 Utmaningar inom språkteknologi

Det finns tusentals språk i världen och de kan skilja sig mycket från varandra beträffande till exempel grammatik, alfabet och läsriktning. Allmänna utmaningar inom språkteknologin är att många ord och meningar har flera betydelser [3]. Till exempel kan ordet val syfta både på ett djur, ett beslut eller ett politiskt val. För en människa är det ofta lätt att förstå innebörden av ordet beroende på sammanhang, men för en dator är detta betydligt svårare. Datorer har även svårigheter att analysera text som innehåller stavfel eller ironi [2], eftersom textens betydelse då är en annan än den som står i texten.

Vid användning av språkteknologi inom medicin tillkommer det ytterligare utmaningar. Det språk som används inom medicin är en minilekt, alltså ett fackspråk, och skiljer sig från vardagligt språk. Detta gör att man inte direkt kan applicera språkteknologi på data som innehåller medicinska termer. Till exempel är man tvungen att ta i beaktande ord, förkortningar, akronymer och annorlunda meningsuppbyggnad. De ord och förkortningar som inte används i andra sammanhang än medicinska behövs ta i beaktande speciellt mycket [4].

2.2 Uppgifter inom språkteknologi

Då språkteknologin bearbetar språk utför den uppgifter (eng. tasks). Vilka uppgifter språkteknologin utför varierar beroende på sammanhanget och syftet med analysen. Eftersom den här avhandlingen fokuserar på språkteknologi inom medicin kommer jag endast nämna de uppgifter som oftast används inom det medicinska området.

2.2.1 Informationsutvinning

Informationsutvinning (eng. information extraction) är en kategori av språkteknologiuppgifter som används mycket inom medicin [4]. Dit hör bland annat uppgifterna *namnigenkänning* (eng. named entity recognition), *relation extraction*, *entity linking*, *sentimentanalys* (eng. sentiment analysis) och *händelseutvinning* (eng. event extraction)

Namnigenkänning innebär att orden i en text gås igenom och klassificeras [5]. Exempelvis kan ordet arm kategoriseras som kroppsdel och ordet feber som symtom. Språkteknologiuppgiften relation extraction går ut på att hitta samband mellan dessa klassifikationer [4]. Till exempel kan relation extraction hitta samband mellan en sjukdom och dess symtom. Vid entity linking sammankopplas entiteterna med poster i en databas. Entity linking fungerar speciellt bra för att identifiera synonymer, eftersom ord med lika betydelse kopplas till samma post i databasen [4].

Vid sentimentanalys analyseras texten utgående från vilken värdeladdning texten har och som utdata fås till exempel om texten är positivt, negativt eller neutralt värdeladdad [5]. Till exempel meningen ”Kaffe är gott och uppiggande” skulle klassas som positivt värdeladdad, och följaktligen i meningen ”Fotboll är roligare än ishockey” skulle ordet ”Fotboll” klassas som positivt värdeladdat medan ordet ”ishockey” skulle klassas som negativt värdeladdat.

Händelseutvinning innebär att man försöker hitta händelser i en text. Man analyserar även händelserna utifrån när de äger rum, om de har hänt, kommer att hända eller inte kommer att hända. Till exempel skulle tillämpning av händelseutvinning på meningen ”Du har röstat” klassa ordet ”röstat” som en händelse och sannolikheten för att händelsen inträffar skulle bli 1. [5]

2.2.2 Förprocessering av data

Språkteknologiuppgifterna inom informationsutvinning bygger på ett antal mindre uppgifter som måste utföras för att informationsutvinningsuppgifterna ska lyckas. Några av de vanligaste är tokenisering (eng. tokenization), Part-of-Speech taggning (eng. POS-tagging) och skiftlägesändring (eng. case folding). Dessa uppgifter används också som deluppgifter till andra uppgifter inom språkteknologin än informationsutvinning.

Då man utför tokenisering delar man in texten i mindre delar. I de språk som använder sig av mellanrum mellan orden delas texten in enligt ord och mellanrummen mellan orden indikerar var texten kan delas. I de språk som inte använder sig av mellanrum mellan orden på samma sätt som bland annat de germanska språken är det svårare att utföra tokenisering [6] . Ett exempel på ett språk som inte alltid har mellanrum mellan orden i en text är japanska. I en medicinsk text förekommer det ofta specialtecken, vilket försvårar tokeniseringen [4].

POS-taggning i sin tur innebär att varje ord i en text tilldelas en POS-etikett (eng. POS-tag) enligt vilken ordklass orden hör till [5] . I meningen ”Jag dricker vatten snabbt” skulle ”Jag” tilldelas etiketten pronomen, ”dricker” etiketten verb, ”vatten” etiketteras som substantiv och ”snabbt” som adjektiv. POS-taggning fungerar därför bra för att särskilja samma ord med olika betydelser då de får olika etikett beroende på betydelse i sammanhanget. Ordet ”min” kan till exempel etiketteras både som ett possessivt pronomen och som ett substantiv.

Texten behöver också bearbetas med skiftlägesändring så att alla versaler byts ut mot gemener. Skiftlägesändring kan vara komplicerat i vissa fall. Bland annat kan datorn inte längre lika lätt urskilja namn från resten av texten efter att de stora bokstäverna blivit utbytta [6]. Ett exempel är efternamnet Skog, som också kan betyda ett trädbevuxet område. Utöver detta behöver också emojier och andra specialtecken ofta tas bort eftersom de anses vara onödiga och försvårar datorns förmåga att analysera och förstå texten.

3. Maskininlärning

I likhet med språkteknologi är även maskininlärning en underkategori till artificiell intelligens. Inom maskininlärning tränas ett datorprogram med träningsdata för att utföra uppgifter. Baserat på denna träningsdata kan datorn sedan på egen hand utföra uppgifter med hjälp av indata och ge ut resultatet som utdata [8]. För att träna datorprogrammet användas olika inlärningsmetoder. Exempel på inlärningsmetoder är övervakad inlärning (eng. supervised learning), oövervakad inlärning (eng. unsupervised learning) och djupinlärning (eng. deep learning). Eftersom maskinlärningsmodellerna som undersöks i denna avhandling basera sig på övervakad inlärning och djupinlärning kommer inte oövervakad inlärning tas upp.

3.1 Övervakad inlärning

Övervakad inlärning är en inlärningsmetod där datorn förses med träningsdata innehållandes både indata och utdata. Då en modell som använder sig av övervakad inlärning tränas tilldelas modellen det väntade värdet på utdata. Målet är att skillnaden mellan det väntade värdet på utdata som förses vid träning av modellen och det beräknade värdet på utdata som modellen själv beräknar ska vara så liten som möjligt. Övervakad inlärning används ofta vid klassificering (eng. classification) och regression. [9]

Klassificering innebär att datorn analyserar en datamängd och delar in den i klasser [9]. Till exempel om datamängden innehåller bilder på bananer och äpplen analyserar datorn varje bild och avgör om motivet föreställer en banan eller ett äpple. Om det är en banan på bilden klassificerar datorn den som banan och om det är ett äpple som ett äpple.

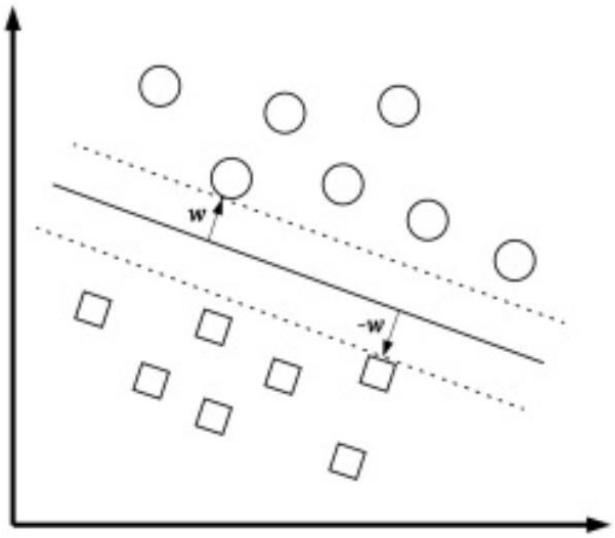
Vid regression förutspår datorn ett värde baserat på en datamängd. Figur 9 visar hur processen går till då ett regressionsproblem löses. Regressionsmodellen tar in data i vektorform, analyserar den och ger ett kontinuerligt värde som utdata. Ett exempel på ett regressionsproblem är förutsägelse av värdet på en aktie. [9]

Figur 1. Regressionsprocess. Översatt från [9].

3.1.1 Stödvektormaskin

En maskininlärningsalgoritm som använder sig av övervakad inlärning är stödvektormaskin (eng. support vector machine), vilket också är en av maskininlärningsalgoritmerna som undersöks i denna avhandling. Stödvektormaskiner används både vid klassificering och regression men oftare används den vid klassificering [10].

Stödvektormaskiner använder sig av hyperplan för att dela in data i två klasser både vid klassifikationsproblem och regressionsproblem. Figur 2 illustrerar ett hyperplan som har placerats i datamängden så att det uppstår en maximal marginal mellan de datapunkter som är närmast varandra i de skilda klasserna [4]. Stödvektormaskiner fungerar såväl på data som är linjärt separerbara (eng. linearly separable) som på icke-linjärt separerbara data (eng. non-linearly separable). Om datapunkterna inte är linjärt separerbara behöver stödvektormaskinerna hitta en kärnfunktion (eng. kernel function) som ombildar planet till en högre dimension. Efter planet omvandlats till en högre dimension kan man igen använda linjär klassificering. Några av de vanligaste kärnfunktionerna är linjär kärnfunktion (eng. linear kernel), polynom kärnfunktion (eng. polynomial kernel), RBF-kärnfunktion (eng. radial basis functions kernel), Gaussisk kärnfunktion och Sigmoid kärnfunktion. [11]



Figur 2 Exempel på hyperplan som stödvektormaskiner använder. Från [4]

3.2 Djupinlärning

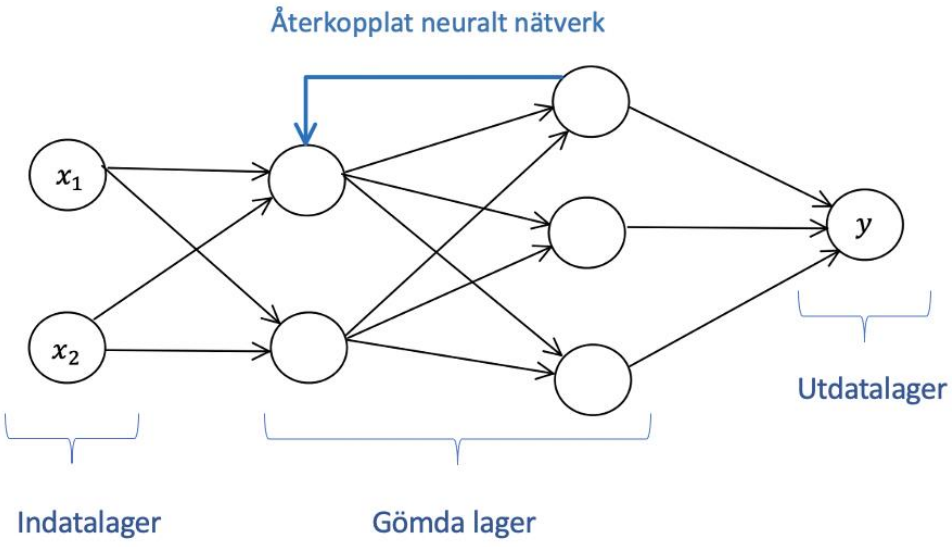
Djupinlärning är en inlärningsmetod som försöker efterlikna människans tänkande [7]. För att försöka uppnå detta använder djupinlärningen sig av neurala nätverk (eng. neural networks). I dessa neurala nätverk finns gömda lager som består av neuroner med funktioner som gör beräkningar på indata. Då en neuron har beräknat indata skickar den vidare resultatet till neuroner i nästa gömda lager som också gör beräkningar [12].

3.2.1 Återkopplade neurala nätverk

Återkopplade neurala nätverk (eng. recurrent neural networks) är en typ av neurala nätverk och den andra maskininlärningsalgoritmen som jag undersöker i avhandlingen. Det speciella med återkopplade neurala nätverk jämfört med andra neurala nätverk är att de kan minnas tidigare beräknade indata och ta resultatet i beaktande vid beräkning av nästa indata [12] .

Figur 3 illustrerar hur ett återkopplat neuralt nätverk kan se ut. I indatalagret förses modellen med indata som sedan förs vidare till neuroner i de gömda lagren där varje neuron är en funktion som gör beräkningar på indata [12]. Hur många gömda lager som finns varierar beroende på uppgiften som det återkopplade neurala nätverket ska utföra. I figur 3 kan man också se hur modellen återkopplar från det andra

gömda lagret tillbaka till det första gömda lagret. Det är då resultatet från tidigare beräkningar som skickas tillbaka och kan användas vid beräkningarna i det första gömda lagret för nästa indata [13].



Figur 3. Återkopplat neuralt nätverk. Översatt från [13].

4. Tillämpning av maskininlärningsmodeller inom språkteknologi

Inom språkteknologi används olika typer av metoder för att utföra språkteknologiuppgifter. Regelbaserade metoder (eng.rule based methods), statistiska metoder (eng. statistical methods), samt maskininlärningsmetoder (eng. machine learning methods) och djupinlärningsmetoder (eng. deep learning methods). Regelbaserade metoder baserar sig på olika regler och reguljära uttryck. Dessa är dock inte lika effektiva som till exempel maskin- och djupinlärningsmetoder, vilka är de metoder som används oftast nuförtiden. [4]

Användningen av maskininlärning inom språkteknologi möjliggör att språkteknologiuppgifterna som nämndes i kapitel två kan utföras både med ett bättre resultat och flera uppgifterna kan utföras som en enda uppgift [5], vilket också resulterar i att uppgifterna utförs snabbare.

I denna avhandling har jag valt att jämföra maskininlärningsmodellerna stödvektormaskiner och återkopplade neurala nätverk eftersom de är två av de vanligaste maskininlärningsmodellerna som används tillsammans med språkteknologi, även inom medicin. Jag kommer också undersöka hur en modell som tränats med övervakad inlärning skiljer sig i resultat jämfört med en modell som tränats med djupinlärning.

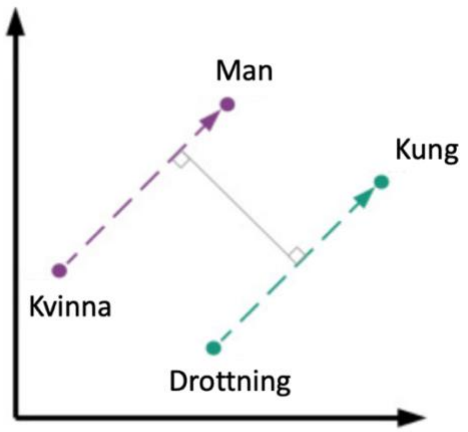
4.1 Vektorisering

För att använda en maskininlärningsmodell behöver data som förs in i modellen vara i sifferform [4]. Inom språkteknologi representerar man oftast texten i sifferform med hjälp av vektorer. Några av de vanligaste teknikerna för vektoriseringen är Bag of Words (BoW), tf-idf och ordinbäddningar (eng. word embeddings).

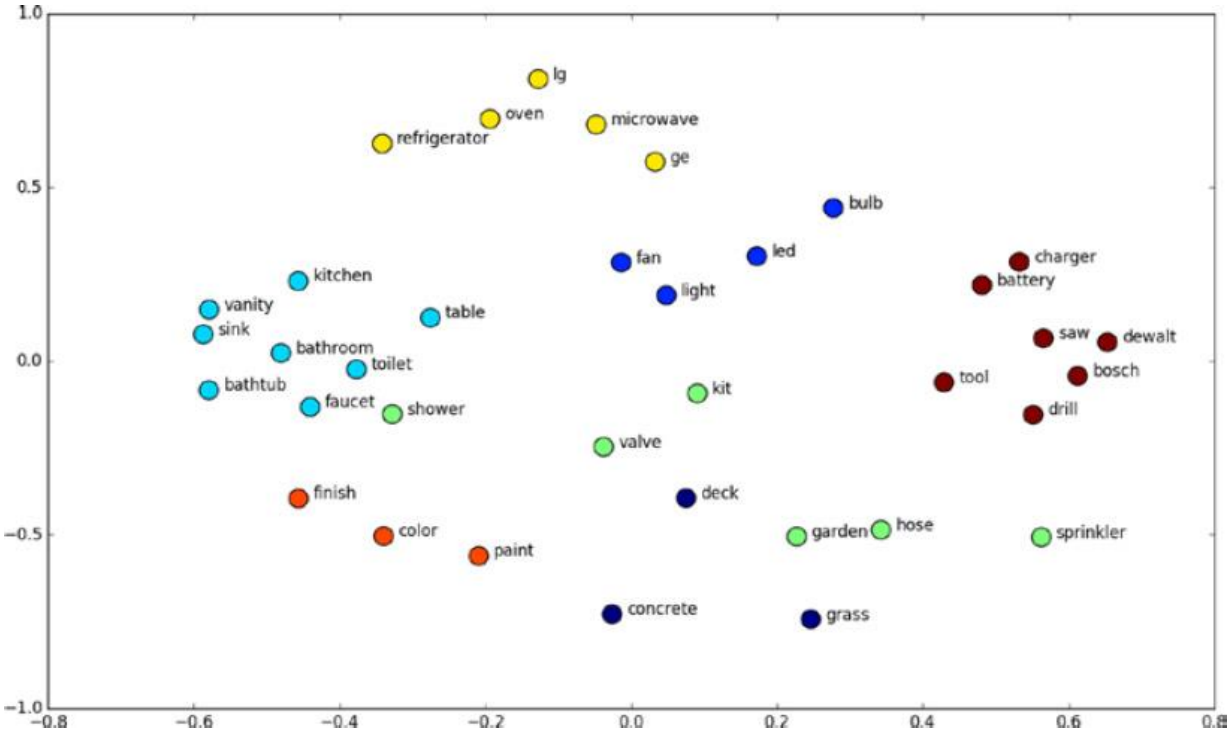
Den enklaste tekniken *BoW* gör texten till en vektor som innehåller siffror på hur många gånger ett ord dyker upp i en text. För att göra detta utgår den från en ordlista och om ett ord som finns i ordlistan inte finns i texten som analyseras representeras det ordet som en nolla i vektorn. Om det ordet finns en gång i texten representeras det som en etta och så vidare. En nackdel med BoW är att den inte tar ställning till vilken ordning orden förekommer i texten. [6] Till exempel om meningen ”Fina blommor” jämförs med ordlistan ”Många fina blommor på ängen” fås vektorn [0,1,1,0,0].

Tf-idf är en vektoriseringsteknik vars förkortning står för ”term frequency – inverse document frequency”. Denna vektoriseringsteknik är mer avancerad än BoW eftersom den också tar ställning till hur viktiga orden är i en text är. Tf-idf är produkten av värdena från två andra metoder, tf och idf. Tanken med tf-idf är att de ord som förekommer ofta och i många olika dokument, får ett lågt värde. Till exempel ord som ”och”, ”är” och ”men”. På samma sätt får ord som förekommer ofta, men endast i ett eller några dokument ett högre värde eftersom de anses ha en större betydelse för texten. [6]

Den mest avancerade vektoriseringstekniken är *ordinbäddningar*. Ordinbäddningar gör varje ord till en egen vektor, vilket gör att man kan urskilja hur orden hänger ihop med andra ord. Till exempel kan man urskilja motsatsord och synonymer [6]. Figur 4 visar hur orden ”man”, ”kvinna”, ”kung” och ”drottning” avbildas som vektorer i ett plan. Eftersom vektorerna är parallella i vektorplanet kan datorn utgående från dem förstå att orden har ett samband med varandra [4]. I figur 5 kan man se hur ord i en större datamängd vektoriseras med ordinbäddning. Ord som har liknande betydelse eller hör till samma kategori tilldelas vektorer med liknande värden och placeras därför närmare varandra i vektorplanet. Till exempel är de ord i figur 5 som kan kategoriseras som verktyg nära varandra [14].



Figur 4. Ordvektorer skapade med ordinbäddning. Översatt från [4].



Figur 5. Ordvektorer skapade med ordinbäddningar. Från [14].

4.2 Tillämpning av stödvektormaskiner

Stödvektormaskiner är den vanligaste maskininlärningsmetoden som används för att förutsäga hjärtsjukdomar [13]. Eftersom en stor mängd medicinska data som kan användas som data vid förutsägelse av sjukdomar är i textform är det även nödvändigt att använda språkteknologi tillsammans med stödvektormaskiner för att kunna identifiera riskfaktorer från texten.

I en studie av Buchan et al. använder de sig av språkteknologi tillsammans med stödvektormaskiner för att automatiskt förutsäga vilka som kommer drabbas av kranskärlssjukdomar bland patienter som lider av diabetes. Eftersom diabetes redan är en riskfaktor för hjärt-och kärlsjukdomar och personer med diabetes ofta även har andra riskfaktorer för kranskärlssjukdomar är det en utmaning att kunna identifiera vilka som kommer att drabbas. [15]

Materialet de använder sig av i studien är elektroniska medicinska register (EMR). I studien använder de sig förutom av stödvektormaskiner även av två andra maskininlärningsmodeller för att förutsäga vilka personer som kommer drabbas,

Naive Bayes och MaxEnt. De undersöker både en stödvektormaskin som använder linjär kärnfunktion och en stödvektormaskin med RBF-kärnfunktion. [15]

I studien testades modellerna med olika variabler och stödvektormaskinen med linjär kärna fick F1 värdet 0.768 och stödvektormaskinen med RBF-kärnfunktion fick F1 värdet 0.755. De andra maskininlärningsmodellerna fick också liknande F1 värden. F1 är en skala som mäter maskininlärningsmodellers precision i skalan 0-1, där 1 är det högsta värdet. Skalan mäter alltså hur många gånger modellen gjorde rätt förutsägelse på hela datamängden.

En begränsning som ofta finns i studier om förutsägelser av sjukdomar är att det är svårt att mäta exakt hur många fall som modellerna lyckas förutsäga rätt eftersom patienterna kan drabbas av kranskärlssjukdomar längre in i framtiden än studien undersökte. [15]

En annan studie som använder sig av stödvektormaskiner tillsammans med språkteknologi är Rochefort et al. [16]. I denna studie vill de identifiera venös tromboembolism (VTE) hos patienter baserat på elektroniska hälsoregister. Till venös tromboembolism hör också lungemboli och djup ventrombos som är vanliga hjärt- och kärlsjukdomar. I studien undersöker de tio olika stödvektormaskiners prestationer vid identifiering av dessa sjukdomar. Vektoriseringstekniken som används i studien är BoW. [16]

AUC är ett värde som kan mäta hur väl en maskininlärningsmodell presterar. Liksom F1 värdet antar AUC ett värde mellan 0–1 beroende på hur väl modellen klarar av att klassificera. Exempelvis hur väl modellen klarar av att särskilja patienter med sjukdom och icke sjukdom.

Den stödvektormaskinversionen som fick bäst resultat vid identifiering av sjukdomen djup ventrombos var en stödvektormaskin med RBF-kärnfunktion där parametrarna hade optimerats. Den lyckades i medeltal identifiera 80 % av de rapporter där patienten hade djup ventrombos. Den genererade också i medeltal 11% falska positiva värden. Medeltalet på den bäst presterande modellens AUC värde var 98%. Stödvektormaskinerns prestationer varierade rätt mycket beroende

på parameterval. Den sämst presterande modellen var en stödvektormaskin med linjär kärnafunktion vars parametrar inte hade optimerats. Den lyckades i medeltal identifiera endast 65% av rapporterna som hade djup ventrombos. Medeltalet på modellens AUC värde var 94 %. Stödvektormaskinernas prestationer varierade dock relativt lite beroende på vilken kärnfunktion de använde. [16]

4.3 Tillämpning av återkopplade neurala nätverk

De senaste åren har det blivit allt vanligare att använda sig av djupinlärning och neurala nätverk i stället för traditionella maskininlärningsmetoder tillsammans med språkteknologi vid identifiering av riskfaktorer och förutsägelser av sjukdomar [13]. Ett neuralt nätverk som lämpar sig bra för att lösa språkteknologiska uppgifter är återkopplade neurala nätverk tack vare dess förmåga att minnas tidigare beräknade indata [10]. Vid tillämpning på text kan återkopplade neurala nätverk till exempel ta i beaktande tidigare ord och baserat på det beräkna vilket ord som kommer därefter.

Chokwijitkul et al. [17] undersöker hur väl de neurala nätverken återkopplade neurala nätverk och faltningsnätverk (eng. convolutional neural network) lyckas identifiera riskfaktorer för hjärtsjukdomar. Förutom traditionella återkopplade neurala nätverk använde de också tre andra tillämpningar på återkopplade neurala nätverk. Vektoriseringstekniken som används i studien är ordinbäddningar och de använder elektroniska medicinska register som data. 60 % av registren användes vid träning av modellerna och de resterande 40 % användes för att testa modellerna. [17]

Deras studie visade att alla de återkopplade neurala nätverken presterade något bättre än faltningsnätverk vid identifiering av hjärtsjukdomar baserat på elektroniska medicinska register. Återkopplade neurala nätverk fick F1 värdet 0.890 medan faltningsnätverket fick värdet 0.879. Den modellen som presterade bäst var bidirectional long short-term memory (BLSTM) som är en version av återkopplade neurala nätverk. BLSTM fick ett F1 värde på 0.908. De jämförde även resultaten med tidigare studier där både regelbaserade system och system

baserade på olika maskininlärningsalgoritmer använts och drog slutsatsen att neurala nätverk kan prestera på ungefär samma nivå som de bäst presterande av dessa system. [17]

Återkopplade neurala nätverk används även i en studie av Choi et al. [18]. I studien vill de undersöka om användningen av återkopplade neurala nätverk fungerar bättre än traditionella maskininlärningsmetoder för att förutsäga hjärtsvikt. De använder sig av modellen gated recurrent unit (GRU) som är en typ av återkopplat neuralt nätverk. [18]

De tränade även traditionella maskininlärningsmodeller varav en var stödvektormaskiner för att jämföra hur väl GRUn presterar. Modellerna tränades på data från elektroniska hälsoregister som var baserade på en 3-12 månader lång tidsperiod. Vid en tolv månader lång tidsperiod fick modellerna alltså tillgång till en större mängd data. GRUn presterade bättre än de traditionella maskininlärningsmodellerna oberoende på längden av tidsperioden med ett AUC värde på 0.776 vid den längsta tidsperioden medan stödvektormaskinen hade ett AUC värde på 0.744 vid den längsta tidsperioden. [18]

I denna studie presenterades även hur lång tid per patient det tog för dessa modeller att förutsäga hjärtsvikt som visas i figur 6. GRU:n presterade bättre tog det mycket längre tid för modellen att komma fram till sitt resultat jämfört med de modeller som baserar sig på traditionell maskininlärning.

Performance metric	Logistic regression	SVM	MLP	KNN	GRU
Prediction time (seconds)	0.000002	0.000034	0.000259	36.66	0.020408

Figur 6. Tabell över tiden det tog för modellerna att göra en förutsägelse per patient. Från [18].

5. Jämförelse och resultat

TODO

Vilken av modellerna som är bäst att använda vid identifiering och förutsägelse av hjärt-och kärlsjukdomar beror på bland annat typen av problem, storleken på datamängden man utgår ifrån och även datamängdens innehåll.

Förutom vilken maskininlärningsmodell är också förprocessingen av data väldigt viktigt och kan inverka på modellens prestation. Även vilken vektoriseringsteknik som används påverkar modellens prestation. En mer avancerad teknik resulterar i en bättre presterande modell. Förprocesseringen av data spelar också en roll i hur modellerna presterar, speciellt inom språkteknologi.

I studien av Rochefort et al. [16] varierade resultatet ganska mycket på den bäst presterande modellen och den sämst presterande modellen trots att båda modellerna byggde på stödvektormaskiner.

En faktor som kan påverka resultatet vid förutsägelser av sjukdomar är hur långt fram i tiden man förutser.

5.1 Val av data

-Rätt data viktigare än modellen (AI, ML and DL)

5.2 Etik och Patientsäkerhet

Eftersom ingen av dessa maskininlärningsmodeller ger ett perfekt resultat förekommer det både falska positiva värden och falska negativa värden. Detta är något som kan vara väldigt känsligt vid förutsägelse av sjukdomar. Att berätta till en patient att den förmodligen kommer drabbas av någon typ av hjärt- och kärlsjukdom kan skapa mycket oro hos patienten.

6. Sammanfattning

Källor

[1] R. Dale, H. Moisl och H. Somers, Handbook of natural language processing, Marcel Dekker, 2000.

[2] D. Khurana, A. Koli, K. Khatter och S. Singh, ”Natural language processing: state of the art, current trends and challenges,” *Multimedia Tools and Applications,* pp. 3713-3744, 14 July 2022.

[3] H. Sharma, ”Improving Natural Language Processing tasks by Using Machine Learning Techniques,” i *5th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, 2021.

[4] J. Vasilakes, S. Zhou och Z. Rui, ”Chapter 6 - Natural language processing,” i *Machine Learning in Cardiovascular Medicine*, London, Academic Press, 2021, pp. 123-148.

[5] Y. Zhang och Z. Teng, Natural Language Processing: A Machine Learning Perspective, Cambridge University Press, 2021.

[6] O. Campesato, Natural Language Processing Fundamentals for Developers, Mercury Learning & Information, 2021.

[7] K. P. Murphy, Machine learning a probabilistic perspective, Cambridge: MIT Press, 2012.

[8] E. Alpaydin, Introduction to Machine Learning, MIT Press, 2014.

[9] J. Taeho, Machine learning foundations : supervised, unsupervised, and advanced learning, Cham: Springer, 2021.

[10] O. Campesato, Artificial Intelligence, Machine Learning, and Deep Learning, Dulles, Virginia ; Boston, Massachusetts ; New Delhi: mercury learning and information, 2020.

[11] B. H. Boyle, Support Vector Machines: Data Analysis, Machine Learning and Applications : Data Analysis, Machine Learning and Applications, Nova Science Publishers, 2011, pp. 81-102.

[12] J. D. Kelleher, Deep learning, Cambridge: MIT Press, 2019.

[13] H. Essam H, M. Rehab E och A. Abdelmgeid A, ”Machine Learning Techniques for Biomedical Natural Language Processing: A Comprehensive Review,” *IEEE Access,* pp. 140628-140653, 2021.

[14] A. Kulkarni och S. Adarsha, Natural language processing recipes : unlocking text data with machine learning and deep learning using Python, Apress, 2021.

[15] K. Buchan, M. Filannino och U. Özlem, ”Automatic prediction of coronary artery disease from clinical narratives,” *Journal of Biomedical Informatics,* vol. 72, pp. 23-32, August 2017.

[16] C. M. Rochefort, A. D. Verma, T. Eguale, T. C. Lee och D. L. Buckeridge, ”A novel method of adverse event detection can accurately identify venous thromboembolisms (VTEs) from narrative electronic health record data,” *Journal of the American Medical Informatics Association,* vol. 22, nr 1, pp. 155-165, October 2014.

[17] T. Chokwijitkul, A. Nguyen, H. Hassanzadeh och S. Perez, ”Identifying Risk Factors For Heart Disease in Electronic Medical Records: A Deep Learning Approach,” i *Proceedings of the BioNLP 2018 workshop*, Melbourne, 2018.

[18] E. Choi, A. Schuetz, W. F. Stewart och J. Sun, ”Using recurrent neural network models for early detection of heart failure onset,” *Journal of the American Medical Informatics Association,* vol. 24, nr 2, pp. 361-370, Mars 2017.