NyS

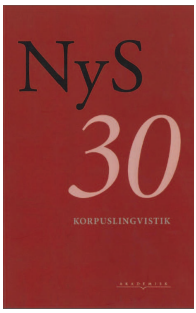
Titel: Dansk grammatikkontrol med transformation-based learning

Forfatter: Daniel Hardt

Kilde: *NyS – Nydanske Sprogstudier* 30. *Korpuslingvistik*, 2002, s. 89-99

Udgivet af: Akademisk Forlag A/S

URL: www.nys.dk



Søgbarhed

Artiklerne i de ældre NyS-numre (NyS 1-36) er skannet og OCR-behandlet. OCR står for 'optical character recognition' og kan ved tegngenkendelse konvertere et billede til tekst. Dermed kan man søge i teksten. Imidlertid kan der opstå fejl i tegngenkendelsen, og når man søger på fx navne, skal man være forberedt på at søgningen ikke er 100 % pålidelig.

Dansk grammatikkontrol med transformation-based learning

DANIEL HARDT

1. INTRODUKTION

Vi beskriver en metode til at bygge grammtikkontrolprogrammer automatisk ved hjælp af Brill-taggeren Brill94. For en given type grammatikfejl er fejl genereret på en systematisk måde, med nye tags der markerer de korrekte og ukorrekte former. Taggeren er trænet sådan at den lærer kontekster hvor fejl kan identificeres. Vi bruger *Contextual Rule Learning*-systemet fra Brilltaggeren uændret. Dette system kan lære kontekstregler der kigger på tre ord foran, tre ord bagefter, samt taggene.

Vi har anvendt metoden til to typer grammatikfejl på dansk: artikel-substantiv-inkongruens og kommaplacering. Det trænede system til kongruenskontrol opnår 95% præcision, og mange af de resterende fejl skyldes mangler i leksikonet. Kommaplacering er meget vanskeligere, men dette system opnår alligevel 91% præcision.

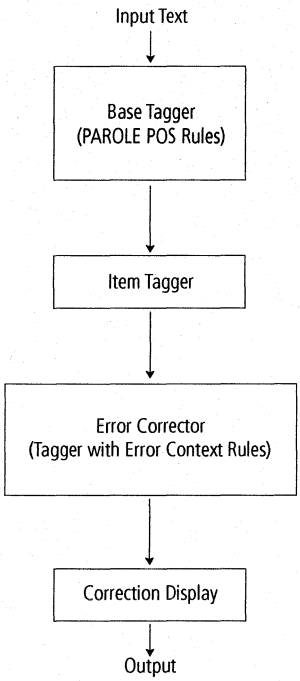
2. GENEREL METODE

Vi starter med det manuelt taggede danske PAROLE-korpus, som indholder ca. 290.000 ord. Vi har trænet Brill-taggeren på PAROLE-korpuset, og den trænede tagger kalder vi Base Tagger. Man kan definere et grammatikkontrolproblem som et valg blandt en mængde leksikalske enheder, som kaldes *Confusion Set*[1]. For at gøre det enkelt, bruger vi Confusion Sets med højst to elementer, men det kunne nemt generaliseres. Vi beskriver således et grammatikkontrolproblem som et valg mellem *lexitem1* og *lexitem2*. Vi tagger hver forekomst af lexitem1 med en unik tag ITEM1, og vi tagger hver forekomst af lexitem2 med en unik tag ITEM2. Så indsætter vi systematisk fejl i korpuset ved at erstatte nogle forekomster af lexitem1 med lexitem2, og vice versa.

Derefter producerer vi *training data* til Brill-taggerens læringssystem, *Contextual-Rule-Learn*. Læringssystemet får to filer som input, kaldet *Truth* og *Dummy*. *Truth* er tagget korrekt, *Dummy* ukorrekt. Systemet prøver at finde frem til regler som kan bruges til at få *Dummy* til at ligne *Truth* så meget som muligt. Vi starter med at lave to kopier af det taggede korpus. *Truth* filen er uændret[2], mens vi laver ændringer af følgende form til *Dummy*:

lexitem1/ITEM1 -> lexitem2/ITEM2
lexitem2/ITEM2 -> lexitem1/ITEM1

FIGUR 1. Training the Error Corrector System



Korrekte tags er altså beholdt i *Truth*, men ikke i *Dummy*. Bemærk at vi går ud fra at alle forekomster i det originale korpus er korrekte, og hvert ændring er ukorrekt.

Derefter kører vi *Contextual-Rule-Learn* med Truth og Dummy som input. Systemet prøver at lære *patterns* hvor Dummy kan ændres sådan

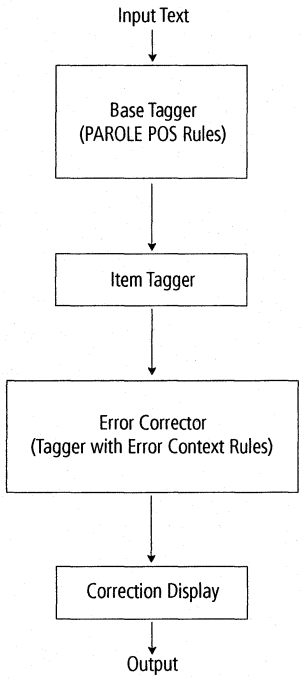
at den ligner Truth: I dette tilfælde er de eneste forskelle hvor ITEM1 tag burde ændres til ITEM2, eller omvendt. Resultatet bliver en ordnet liste af regler som beskriver kontekster hvor disse ændringer skal udføres. Vi kalder disse regler *Error Context Rules* (se figur 1).

Nu kan man køre taggeren med *Error Context Rules*, og vi kalder dette system Error Corrector. Vi har nu et system der kan rette fejl i almindelig tekst. Systemet består af følgende processingsfaser (se figur 2):

1. Base Tagger: Tagger som har ordklasseregler dannet ved træning på det danske PAROLE-korpus
2. Item Tagger: Annoterer lexitem1 med taggen ITEM1, lexitem2 med ITEM2
3. Error Corrector: Tagger med Error Context Rules 4. Udskrivning

Med denne metode kan vi udvikle en Error Corrector for et arbitrært grammatikkontrolproblem uden at ændre i Brill-taggerens software.

FIGUR 2. Error Correction System



3. KONGRUENS MELLEM ARTIKEL OG SUBSTANTIV

På dansk skal artikler og adjektiver have samme køn som substativer. Her fokuserer vi på kongruens mellem indefinit artikel og substantiv, dvs. *et* og *en*.[3]

For at lære regler om artikel-substantiv-kongruens laves først to kopier af det manuelt taggede PAROLE-korpus, hvor hver linie gentages tre gange. Den første kopi kaldes *Truth*. Den anden kopi, som hedder *Dummy*, er modificeret på flg. måde: den første linie er uændret. I den anden linie er alle forekomster af "en" ændret til "et". I den tredje linie er alle forekomster af "et" ændret til "en". Vi tagger hver forekomst af "et" med taggen ET og hver forekomst af "en" med taggen EN. Betragt følgende konstruerede eksempel (irrelevante tags er fjernet).

Truth

En/EN mand har et/ET hus.
En/EN mand har et/ET hus.
En/EN mand har et/ET hus.

Dummy

En/EN mand har et/ET hus.
En/ET mand har et/ET hus.
En/EN mand har et/EN hus.

Bemærk i den anden line af Dummy at en EN-tag er blevet ændret til en ET-tag, og i den tredje line at en ET-tag er blevet ændret til en EN-tag. Hver sætning i det 290.000 ord store korpus er blevet behandlet på denne måde. Derefter køres *Contextual-Rule-Learn* fra Brill-taggeren. Dette producerer *Error Context Rules* til artikel-substantiv kongruens. Herunder ses de første ti regler der er blevet lært:

1. ET → EN if one of the three following tags is N(common-sing)
2. EN → ET if one of the three following tags is N(neuter-sing)
3. ET → EN if one of the three following tags is Adj(common-sing)
4. ET → EN if the next tag is N(common-sing)
5. EN → ET if one of the two following tags is N(neuter-sing-genitive)
6. EN → ET if one of the two following tags is N(neuter-plural)

7. EN → ET if the next tag is Adj(neuter-sing)
8. ET → EN if the next tag is N(common-sing-genitive)
9. EN → ET if the next tag is N(neuter-sing)
10. ET → EN if one of the two following tags is Pronoun(common-sing)

Disse regler beskriver tilfælde hvor den først tag skal erstattes med den anden.

3.1 DET TRÆNEDE SYSTEM

Som beskrevet i afsnit 2 har vi bygget en *kongruens-tjekker* på en meget enkel måde: inputtet er først tagget med Base Tagger. Så køres Item Tagger som tagger alle forekomster af "en" med EN og tagger alle forekomster af "et" med ET. Bagefter er fejlene blevet rettet med *Error Corrector*, og til sidst bliver fejl vist – det vil sige de tilfælde hvor en forekomst af "en" bærer en ET tag, eller omvendt. *Error Display* viser fejl i dette format: en(et) for "en" rettet til "et" – eller et(en) for "et" rettet til "en".

Her er en tekstprøve fra en testkørsel:

Input

En mand har et hus.
Et mand har et hus.
En mand har en hus.

Output

En mand har et hus .
et(en) mand har et hus .
En mand har en(et) hus .

3.2 RESULTATER OG ANALYSE

Systemet blev testet på en fil med 162.876 ord, kopieret fra Bergenholtz korpus Bergenholtz88. Filen indholdt 527 forekomster af "et" og 1098 af "en". Der blev først lavet to nye filer, begge bestående af to konkatenerede kopier af originalfilen. Den ene nye fil kaldtes Truth, og var uændret.

Der blev genereret fejl i den anden fil på følgende måde: i første halvdel blev alle forekomster af "en" ændret til "et", og i anden halvdel blev alle forekomster af "et" ændret til "en". I alt var der 1625 forskelle (dvs. fejl). Testen resulterede i 1454 forslåede rettelser fra "et" til "en" eller fra "en" til "et". Af disse var 1375 korrekte og 79 ukorrekte. Det giver en præcision på 95%. Der var i alt 1625 fejl, og det betyder at *recall* er på 85%.

Mange ukorrekt placerede tags kan forbindes med mangler der stammer fra det lille træningskorpus (det danske PAROLE-korpus på 290.000 ord). Som led i en træningssession bygger taggeren et leksikon bestående af hvert ord der forekommer i træningskorpuset, samt en liste af alle mulige ordklasser for hvert ord.

Dette er en afgørende information for kongruenstjekkeren, fordi taggeren kan finde frem til substantivets køn ved at kigge på listen af ordklasser for et givet substantiv. Den hyppigste fejlårsag er tilsyneladende at substantivet ikke findes i leksikonet. Følgende er en liste af de første 5 ukorrekt markerede fejl:[4]

- et(en) vidunder
- et(en) øjebliks
- et(en) statsapparat
- et(en) ordentligt møgfald
- et kvæk : et(en) kvæk

Substantiverne i alle disse fejltilfælde (*vidunder, øjebliks, statsapparat, møgfald, og kvæk*) mangler i systemleksikonet. Derfor er det sandsynligt at et større træningskorpus ville forbedre systemet, fordi det ville udvide leksikonet. På den anden side er der nogle fejl som har at gøre med stærkt idiosynkratiske konstruktioner, som f.eks. følgende: "et provokerende 'husets-herre-venter-på-at-blive-opvartet'-attitude"

Her forbliver "et" ukorrekt uændret af systemet (der var „en" i originalteksten). Systemet søger efter udtrykket *husets-herre-venter-på-at-blive-opvartet* i leksikonet, uden held.

Alt i alt antyder analysen at metoden, trods sin enkelhed, kan udvikle systemer der løser kongruensproblemet med ret stort held. Mange fejl vil formentlig kunne undgås alene ved at arbejde med et nogenlunde komplet leksikon. Andre fejl stammer fra idiosynkratiske konstruktioner, som måske overhovedet ikke kan behandles med de nuværende metoder.

Formålet her er at bygge et system der kan finde forkerte kommaer, dvs. kommaer der skal slettes. Vi anvender samme metode som til kongruensproblemet, dog er nogle detaljer i træningsmaterialet anderledes, som beskrevet herunder.

Træningsfilen blev lavet ved at tagge en tekst på 600.000 ord kopieret fra Bergenholtz-korpuset, med brug af Base Tagger. Vi konverterede herefter taggene til det reducerede PAROLE-tagset (Haltrup 2002), med det formål at lette indlæringen af generaliseringer som f.eks. "intet komma mellem en præposition og et substantiv". I det originale PAROLE-tagset er der 23 forskellige tags til appellativer (fællesnavne), for at dække forskelle i tal, køn, osv. I det reducerede PAROLE-tagset er der kun to: N_GEN (genitiv) og N (alle øvrige). Andre ordklasser har også reducerede antal tags. Bemærk at vi ikke kunne bruge det reducerede tagset til kongruensproblemet, hvor information om køn var nødvendig.

En mængde nye kommaer blev sat ind på tilfældige steder i træningsfilen. Disse ekstra kommaer ansås for fejl og fik taggen BC (bad comma) i *Truth* filen, mens de original kommaer fik taggen GC (good comma). *Dummy* filen var identisk med *Truth*, bortset fra at alle kommaer var tagget GC. Opgaven for systemet var altså at lære at identificere de kontekster hvor et kommas tag skulle ændres fra GC til BC og markere dem som fejl. Læringen består, som før beskrevet, i at systemet udvikler en ordnet liste af regler der beskriver i hvilken kontekst et kommatag skal ændres. Det er vigtigt at bemærke at disse regler er *ordnede*, sådan at ændringer indført af en regel tidligere på listen i nogle tilfælde bliver korrigeret af en regel senere på listen (Haltrup 2002.)

I alt blev der udviklet 166 *Error Context Rules*. De først 12 regler var følgende:

1. GC → BC if one of the three following tags is End-of-sentence
2. GC → BC if one of the two previous tags is Beginning-of-sentence
3. GC → BC if the next tag is Preposition
4. GC → BC if one of the two following tags is Verb(Infinitive)
5. GC → BC if the previous tag is Conjunction
6. BC → GC if the previous tag is Interjection

7. GC → BC if the previous tag is Preposition and the following tag is N
8. GC → BC if one of the two previous tags is Subordinating Conjunction
9. GC → BC if the previous tag is Pronoun and the following tag is N
10. GC → BC if the previous tag is Verb(past) and the following tag is Pronoun(personal)
11. BC → GC if one of the next two tags is Subordinating Conjunction
12. GC → BC if the previous word is *er* (is)

De først to regler siger at et komma er markeret forkert ("BC") hvis det forekommer efter et af de 3 sidste ord i sætningen, eller efter et af de første to ord i sætningen. Disse regler blev lært fordi der er relativt få korrekte kommaer i disse omgivelser i Truth, mens mange ukorrekte kommaer forekommer i disse omgivelser. Senere lærer systemet at disse to regler er for generelle. For eksempel udtrykker den sjette regel at et komma er korrekt hvis det følger en interjektion (INTERJ). Det sker typisk omkring starten eller slutning af sætningen, som f.eks. i:

```
Naa/INTERJ ,/GC I/PRON_PERS sidder/V_PRES stadig/RGU
og/CC hygger/V_PRES jer/PRON_PERS ./XP
```

Regel 7 forbyder kommaer mellem præpositioner og en substantiver, og Regel 8 forbyder kommaer lige efter begyndelsen af en ledsætning. Dette hænger sammen med at kommaer typisk forekommer netop før en ledsætning på dansk. Samme fænomen er afspejlet i Regel 11, som tillader kommaer lige foran en underordnende konjunktion. Regel 9 forbyder kommaer mellem pronominer og substantiver. I PAROLE-tagsettet findes der ingen tag for ordklassen *artikel*, og ord som "den" og "en" blevet tagget som pronominer.

4.1 DET TRÆNEDE SYSTEM

Vi bygger et kommarettelsesystem efter den generelle metode: Base Tagger køres, og derefter køres Error Corrector, som er taggeren med Error Context Rules til kommaproblemet. Kommafejl er tagget med BC i outputtet. Herunder er vist en prøvekørsel af systemet, med forskellig kommatering af sætningen *Det er godt at du kom*:

Input

Det er godt, at du kom.
Det er godt at, du kom.
Det er godt at du, kom.
Det, er godt at du kom.
Det er, godt at du kom.

Output

Det er godt , at du kom .
Det er godt at ,/BC du kom .
Det er godt at du ,/BC kom .
Det ,/BC er godt at du kom .
Det er ,/BC godt at du kom .

Kun den først af de fem forskellige kommateringer er acceptabel (efter normen *grammatisk komma*). Systemet tagger alle alternativer som ukorrekt (BC).

4.2 RESULTATER OG ANALYSE

Systemet blev testet med en fil af tekst fra Bergenholtz-korpuset (strengt adskilt fra træningsteksten). Testfilen indholdt 14.044 ord og 869 kommaer. Der blev indsat 389 yderligere kommaer på tilfældige steder, som fejl. Systemet markerede 327 kommaer som fejl, af hvilke 299 faktisk var fejl. Det giver en præcision på 91,4% og en recall på 76,9%.

Her er de først 10 eksempler hvor systemet markerer for fejl, selv om kommaet er korrekt:

1. Hulgaard/EGEN ,/BC Århus/EGEN
2. mener/VPRES ,/BC vi/PRONPERS
3. mener/VPRES ,/BC han/PRONPERS
4. menneskemassen/VPRES ,/BC der/UNIK
5. 17-13/NUM ,/BC Norris-Paulsen/N
6. morderiske/VPRES ,/BC psykopatiske/VINF
7. Sørensen/EGEN ,/BC Århus/EGEN
8. nabokommunen/N ,/BC på/SP
9. systemet/N ,/BC kan/VPRES
10. de/PRONDEMO aktive/ADJ ,/BC servicefunktionerne/N

I nummer 1 og 7 var der en forkert lineskift lige foran teksten. Nummer 4 og 6 rummer forkerte ordklassetags: "menneskemassen" og "morderiske" er begge substantiver, men er tagget som verber.

Nummer 10 er interessant: "De aktive, servicefunktionerne". Kommaet er bedømt ukorrekt på grund af følgende regel:

- GC → BC if the previous tag is ADJ and the next tag is N

Reglen er egentlig fornuftig: kommaer forekommer normalt ikke mellem et adjektiv og et substantiv. Men her forekommer "De aktive" og "servicefunktionerne" som to selvstændige NP'er.

5. KONKLUSIONER OG FREMTIDSPLANER

Vi har vist at Brill-taggeren kan bruges til at konstruere *grammar checkers* automatisk, inden for to vigtige problemområder i dansk grammatik: artikel-substantiv-kongruens og kommaplacering. Metoden er generel og kan anvendes til vilkårlige grammatikkontrolproblemer der kan beskrives som et valg blandt en mængde leksikalske enheder. Traditionelle grammatikbøger (f.eks. Jacobsen & Jørgensen1991) indholder lange lister af sådanne emner: vi har identificeret mindst 12 grammatikkontrolproblemer på dansk som metoden kan anvendes på. Vi har også planer om at anvende metoden til problemer i engelsk grammatik.

Vi formoder at mange andre grammatikproblemer ligner artikel-substantiv-kongruensproblemet og derfor med held kan behandles med vores metode. Vi er i gang med at undersøge hvordan Brill-taggeren kan ændres til at blive et mere effektivt værktøj til at udvikle grammatikkontrolsystemer, f.eks. ved at modificere indlæringsalgoritmen. Brill-taggeren lærer på en *grådig* (greedy) måde idet den altid maksimerer den overordnede succes-procent. Der kan dog være grund til at vægte *præcision* højere i vores applikationer.

Daniel Hardt
Institut for Datalingvistik,
Handelshøjskolen i København
email: dh@id.cbs.dk

NOTER

1. Golding & Schabes 1996 beskriver et lignende arbejde.
2. For kommasystemet er *Truth*-filen ikke uændret, som beskrevet i det følgende. Her indfører vi fejl i begge filer. Fejlforekomster er tagget som fejl i Truth, men ikke i Dummy. Metoden kræver at alle tags er korrekte i Truth, mens visse tags der er relevante i forhold til problemområdet, er ukorrekte i Dummy.
3. Der er selvfølgelig andre former for artikel-substantiv-kongruens som vi ikke dækker her, f.eks. med definitte artikler og demonstrativer.
4. Disse er tilfælde hvor systemet lavede en ukorrekt ændring fra "et" til "en".

LITTERATUR

Bergenholtz, H. (1988): Et korpus med dansk almensprog. *Hermes.*

Brill, E. (1994): A report of recent progress in transformation-based error-driven learning. *DARPA Workshop.*

Golding, A.R. & Schabes, Y. (1996): Combining trigram-based and feature-based methods for context-sensitive spelling correction. *Proceedings of the 34th Annual meeting of the Associaltion for Computational Linguistics.*

Hansen, Dorte Haltrup (2002): To ressourcer. *NyS 30* (dette nummer)

Jacobsen, H.G. & Jørgensen, P.S. (1991): *Politikens Håndbog i Nudansk.* Politikens Forlag.