AN ALGORITHM FOR END-OF-SENTENCE DETECTION IN TEXT

Robert Eklund*
Janne Lindberg*

Internal Report, June 1993

1 Introduktion

Automatisk detektering av meningsslut i text är ett desideratum inom ett flertal lingvistiska tillämpningar. Dels kan det sägas vara en förutsättning för automatisk parsning av text, dels är det behjälpligt inom text-till-tal-tillämpningar. Även om text är en mycket "hanterligare" källa än en talström när det gäller att hitta satsgränser är uppgiften långt ifrån problemfri. Även om interpunktion visserligen utgör ett reglerat system är bruket av interpunktionstecken inte helt konsistent eftersom samma tecken kan användas på flera olika sätt avhängigt kontexten.

I denna uppsats presenteras en algoritm för automatisk detektering av meningsslut i text. Dessutom beskrivs en av Liberman & Church (1989) tidigare anvisad metod som legat till grund för den alternativa metod som vi presenterar i denna uppsats. En jämförelse mellan metoderna tillhandahålls.

2 Algoritmer för automatisk detektering av meningsslut

2.1 Liberman & Church

En algoritm för automatisk detektering av meningsslut presenteras av Liberman & Church (1989). Deras fullt utbyggda algoritm har följande utseende:

(? ! .) (" ' ‘)* (_ ¬)+ (" ' ‘)* (versal)

...där:

_	=	mellanslag
¬	=	tabb
*	=	noll eller flera förekomster
+	=	en eller flera förekomster

De parenteser som inte är markerade innebär *en* förekomst av något i parentesen inkluderat tecken.

Detta enkla mönster, menar Liberman & Church "will catch nearly all the sentence-ending periods". Vad algoritmen missar är meningar som börjar med en siffra - något som Liberman och Church funnit vara extremt ovanligt - och rena typografiska fel som utelämnade mellanslag. Ett större problem är övergenereringar såsom förkortade titlar (av olika slag) och initialer framför personnamn, som i t ex **Mr. Miller**, **Robert E. Lee** och dylikt.

De resultat Liberman & Church redovisar uppvisar låga felprocent, både vad gäller över- och undergenereringar. Även om de uppvisar imponerande låga siffror för "felträffar" - lägre än en promille i somliga fall - så kvarstår faktum att de negligerat vissa, om än sällsynta, förekommande meningsslut och meningsbörjan. Förutom att de inte inkluderat meningar som börjar med siffra (t ex årtal eller nominalfraskvantifierare) så har de inte inkluderat meningar som börjar eller slutar med parentes. (Detta är antagligen inte särskilt ymnigt förekommande, men *denna* mening börjar och slutar med parentes.) Ett annat fenomen, och betydligt vanligare, är meningar som slutar med kolon: (Föregående mening slutar med kolon!) Huruvida kolon kan sägas vara meningsavgränsare eller inte varierar beroende på kontext. (Se vidare Diskussion.)

---

* Även Institutionen för lingvistik, datorlingvistik, Stockholms universitet. (Email: robert@ling.su.se och beb@ling.su.se)

I följande stycke presenterar vi en algoritm som inkluderar detta och annat.

2.2 Ett förslag till en utbyggd algoritm

Den algoritm vi här presenterar innefattar Liberman & Church' metod som en strikt delmängd. Den algoritm vi föreslår formaliseras sålunda:

(? ! . :) ( )] })? (" ' ‘)? (_ ¬ - \n)+ (" ' ‘)? (([{)? (" ' ‘)? (versal/siffra)

...där:

?	=	noll eller en förekomst
¬	=	tabb
-	=	tankstreck
\n	=	radbrytning
_	=	mellanslag

Omarkerad parentes kräver *en* förekomst av något i parentesen ingående tecken.

Som synes inkluderar denna algoritm fler möjliga meningsslut/meningsbörjan än Liberman & Church genom att tillåta att meningar både kan sluta och börja med parentes, men även börja med tankstreck. Meningar tillåts dessutom sluta med kolon (vilket kommer att visa sig vara den förändring som svarar för den största skillnaden vad rör respektive algoritms prestanda). Vi har även inkluderat siffror som tillåten meningsbörjan.

Initialt inkluderade vi även den restriktionen att det första skiljetecknet inte fick föregås av mellanslag. Vi accepterade även multipla skiljetecken:

˜(_) (? ! . :)+ (osv)

Orsaken till det föregående var att vi bland våra texter fann ett par innehållsförteckningar där ett stort antal punkter åtskilda av mellanslag förekom. För att inte få meningsslut utplacerade vid dessa lade vi på restriktionen att ett skiljetecken inte får föregås av mellanslag. Detta ledde emellertid att vi missade typografiska fel (dvs, "extra" mellanslag före skiljetecken), varför restriktionen plockades bort. Vi sysslar visserligen inte med en korrektur-algoritm, men texter innehåller oundvikligen fel, och om en metod kan göras robustare genom att ta hänsyn till dessa så kan det ju inte vara av ondo. Huruvida det skulle vara fruktsamt att tillåta multipla skiljetecken eller ej medgav inte vårt material att avgöra, men en kvalificerad gissning ger vid handen att det skulle vara kunna vara lämpligt...

2.3 Metod

För att testa respektive algoritms prestanda beslutade vi köra dem på ett antal textfiler. Metoderna implementerades som reguljära sökuttryck i AWK (Dougherty 1991), vilka sedan konverterades till PERL (Wall & Schwartz 1991). Textfilerna tillhör Stockholm-Umeå-korpusprojektet (Ejerhed et al 1992), och hårdvaran var en SUN-station under UNIX.

Vid sammanställningen av resultaten räknades antalet hittade meningsslut av olika slag. I de fall en hittad mening innehöll både kolon och siffra (dvs, två företeelser som Liberman & Church inte hittar) så räknades varannan till den ena gruppen, varannan till den andra gruppen, detta för att undvika procenttal större än 100. Innehållsförteckningar räknades inte alls med.

En brist i undersökningen har varit att vi inte kunna tillgå en vad rör meningsslut oambiguöst taggad text eller korpus. Därför har det varit svårt att kontrollera vad båda algoritmerna "missar" i texter. För att i viss mån kunna kontrollera detta excerperades förkortningar ur textfilerna med hjälp av ett program skrivet i stränghanteringsspråket PCBETA (Brodda forthcoming, Lindberg 1993, Malkior & Carlvik 1990).

3 Resultat

Nedan redovisas resultaten av de bägge algoritmernas prestanda på textfilerna.

TABELL 3.1 - *Totalt antal meningsslut rapporterade med respektive algoritm, separat för varje fil*
E & L = Eklund och Lindberg
L & C = Liberman and Church

	L & C tot	E & L tot	diff tot
g_helen	12 151	12 354	+ 203
kmilj1	888	901	+ 13
kmilj2	3 314	3 440	+ 126
ny-teknik	3 000	3 022	+ 22
kocksgatan	14 902	15 006	+ 104
pc_jersild	10 592	10 763	+ 171
aik-nytt	1 353	1 366	+ 13
Σ	46 200	46 751	+ 652

TABELL 3.2 - *Mönstrens relativa bidrag i procent till skillnaden i täckningsgrad mellan L & C:s och E & L:s detektering av meningsslut, totalt över alla filer.*

:_Övr	85.5 %
:_Uppräkn.	1.5 %
Tankstreck	0.5 %
[.?!]_(	2.5 %
[.?!]_0-9	8 %
[.?!])_	1 %
Övriga	1 %
Σ	100 %

TABELL 3.3 - *Mönstrens frekvens / relativa bidrag i procent till skillnaden i täckningsgrad mellan L & C:s och E & L:s detektering av meningsslut, separat för varje fil.*

	g_helen	kmilj1	kmilj2	ny_teknik	kocksg	pc_jersild	aik-nytt	Σ
:_Övr.	176 / 87%	10 / 77%	86 / 68%	16 / 73%	92 / 88%	170 / 99.5%	6 / 46%	556
:_Uppräkn.	8 / 4%	-	-	1 / 4.5%	1 / 1%	-	-	10
Tankstreck	3 / 1.5%	-	-	-	-	-	1 / 8%	4
[.?!]_(	1 / 0.5%	-	12 / 9.5%	1 / 4.5%	-	-	1 / 8%	15
[.?!]_0-9	14 / 6.5%	3 / 23%	22 / 17.5%	3 / 13.5%	6 / 6%	1 / 0.5%	3 / 23%	52
[.?!])_	-	-	3 / 2.5%	1 / 4.5%	3 / 3%	-	-	7
Övriga	1 / 0.5%	-	3 / 2.5%	-	2 / 2%	2 / 15%	-	8
Σ	203	13	126	22	104	171	13	652

3.1 Parentes

Det kanske mest uppenbara förbiseendet i Liberman & Churchs detektering är att meningar både kan börja och sluta på parentes.

...kulturhistorisk utredning. (Meddelanden från...
Norrland under stenåldern och bronsåldern. (Studier i...
...på din linje fullt ut.) I två år...
...med hjälp av ett lexikon för sådana. (Endast relativt otvetydiga...

Som synes representerar dessa tre exempel två olika bruk av parenteser. I de två första exemplen rör det sig om parenteser i referenser, dvs titeln nämns i brödtexten, och källan anges inom parentes. De två senare exemplen visar satsslut respektive satsbörjan på meningar helt inneslutna inom parenteser. Alla dessa fall missar Liberman & Church. I våra exempeltexter har vi dock endast funnit ett mindre antal sådana (se TABELL 3.3). Möjliga övergenereringar här är om en förkortning föregår en referens.

Ett exempel skulle kunna vara

Diderichsen, P. (1962). Elementaer...

...där vår utbyggda algoritm skulle övergenerera till skillnad från Liberman & Church.

3.2 Kolon

Den största skillnaden mellan Liberman & Church och den här presenterade metoden är inkluderandet av kolon. Som framgår av TABELL 3.2 så utgörs 87 procent av skillnaden mellan de båda algoritmernas prestanda av kolon. Huruvida detta är av godo eller ondo kan diskuteras. Kolon har ett flertal olika användningar, och huruvida man kan säga att ett kolon utgör en meningsavgränsare eller inte varierar på en skala från nästan otvetydiga meningsavgränsare till fall där kolon har en helt annan fuktion än meningsavgränsande.

Det vanligaste fallet är där kolon anger direkt anföring, t ex:

...svarade min far: "En kan inte...
...gjorde han kommentaren: "Om man funderar...
...med sin givna formulering: "Redaktören är då...
...det gamla talesättet: "Det är möjligt...
Min slutreplik blev: "Nästa vecka är...

Dessa fall är de minst problematiska av kolonfallen. Här är det som följer på kolonet en relativt otvetydig mening medan meningen före i någon mening är oavslutad samt får fortsättningston. Man skulle emellertid även kunna betrakta den underliggande representationen för meningen efter som en bisats med bibehållen huvudsatsordföljd och struket infinitivmärke:

...svarade min far att "en kan inte...

...eller ännu klarare vid indirekt anföring:

...svarade min far att man inte kan...

Därvid tvangs man konstatera att mönstret ovan inte utgör meningsslut. Till de mer problematiska fallen hör:

Reportern i direktsändning: "Vi sitter hemma hos...

Här har strängen till vänster om kolonet en funktion som närmar sig bruket i teatermanus, enligt mönstret:

KALLE: *Vad ska vi göra ikväll?*

LISA: *Vi kan väl gå på bio!*

I dessa fall är strängen till vänster om kolonet ett slags "rubrik", och i ett text-till-tal-system måste antagligen i någon mån fortsättningston tillfogas. Klart är att kolonet inte markerar ett klart meningsslut!

Kolon används också för att inleda uppräkningar av olika slag. Ett exempel som fanns i våra texter var:

...bröderna: Amorgos, Anafi, Sikinos, Folegrandos, Milos...

I samtliga förekommande fall av uppräkningar i denna uppräkning rörde sig det om personnamn, vilka således inleddes med versal. Uppräkningar av andra slag, t ex ingredienser, vilka börjar med gemener borde betraktas på samma sätt som uppräkningen ovan, varför algoritmen i sådana fall bör även acceptera mönstret KOLON-MELLANSLAG-GEMEN.

3.3 Siffror

En iögonenfallande "brist" hos Liberman & Church är exkluderandet av meningsinitiala siffror. Detta motiverar de med att meningar som börjar med siffra är så ovanliga att siffror inte behöver inkluderas. Bland 248 000 meningar från *Wall Street Journal* hittade Liberman & Church endast två meningar som började med siffra. Här bör emellertid nämnas att en texts stil kan vara avgörande för vad man finner. Således har Liberman & Church bland en miljon tecken ur *Wall Street Journal* inte funnit ett enda utropstecken.

Såsom framgår av tabellerna ovan så innehöll texterna ett stort antal meningar vilka började med siffra:

...blivit en lagteknisk term. 52 utpekade områden i...
ålderdomligaste bygden. 75 % av befolkningen hade...
...det för faktor? 4:an?
...ackorden till Alla fåglar kommit ren. 4/4-dels...

En källa till övergenerering i vår föreslagna algoritm var siffror i samband med förkortade ord:

...möjligt kringgå kap. 4 i abortlagen, den om...

Dessutom kunde siffror förekomma i referenslistor:

Fahlgren, Karl, Skellefteå stads historia. 1945.

I detta fall övergenererar vår algoritm, medan Liberman & Church inte sätter ut meningsslut i detta fall.

3.4 Tankstreck

Meningar som inleds med tankstreck förekommer kanske främst i skönlitteratur vid repliker, alltså i stället för citationstecken. Exempel:

...finns redan i våra led. - Men ett lag...
...just den motsägelsefulla osäkerheten. - Det...

Så länge kravet på versal efter tankstrecket kvarstår är risken för övergenering liten. Ett vanligt bruk av tankstreck är annars som ett slags parenteser inuti meningar, som till exempel:

...kunde mitt rykte - och fars! - skadas...
Jag själv hade - och har! - vad...

Om det som följer tankstrecket i dessa fall skulle vara ett personnamn skulle algoritmen övergenerera.

3.5 Förkortningar

Ett stort problem vid meningsslutsdetektering utgörs av förkortningar. Jämför de två (här för ändamålet skapade) meningarna:

*Musikstycket inleddes med lilla **f.** **K**ristjinskij ansåg att...*
*Detta ägde rum århundraden **f.** **K**r. men spreds...*

En algoritm som inte "ser" längre än ett tecken till höger om mellanslaget kan omöjligt skilja vad som är meningsslut och vad som inte är det om inte algoritmen känner igen förkortningen. En lösning här är ju givetvis att alla föreslagna meningsslut jämförs med en förkortningslista, eller åtmistone alla föreslagna förkortningar med vissa "misstänkta" drag, som t ex ett *f.*

4 Övriga iakttagelser

Förutom de ovan beskrivna fenomenen så uppträdde ett fåtal andra, vilka här skall redovisas.

Vi inkluderade initialt en restriktion i metoden, nämligen att skiljetecknet (**? ! : .**) inte fick föregås av mellanslag. Detta orsakades av att vi bland våra texter hade innehållsförteckningar där löplinjerna utgjorden av en rad punkter åtskilda av mellanslag, vilka ledde till övergenerering om inte denna restriktion pålades formalismen. Vi upptäckte emellertid att detta ledde till att vi missade typografiska fel i texterna, där skiljetecknet i en mening föregicks av ett mellanslag. Exempel:

Etapp 2 . Inventering och...
Voila ! Grundkursen i...

Sådana fall klarar Liberman & Church av, medan en algoritm med ovannämnda restriktion missar det. Inkluderandet av restriktionen måste begrundas utifrån aktuella behov, men styrkan att kunna detektera även felaktiga meningsslut i löptext (som i exemplet ovan) torde väga tyngre än undvikandet av viss övergenerering i innehållsförteckningar eller andra, mer "udda" bruk av punkter.

Ett annat, lustigare exempel är följande:

I det ögonblicket föddes Emigrantinstitutet. (...) Jag...

"Meningen" *(...)* klarar Liberman & Church inte alls av, medan den här presenterade algoritmen klarar av att placera ett meningsslut efter, men inte före, exemplet. Det kan givetvis diskuteras vilken status "meningar" av ovanstående typ kan sägas ha, och att missa sådana kan måhända inte betraktas som någon katastrof. Framför allt kan man fråga sig hur t ex ett text-till-tal-system skulle begagna sig av informationen att *(...)* utgör en mening i egen rätt! I detta fall rör det sig måhända om en utelämnad del i ett citat. Andra sätt att ange utelämningar kan t ex också vara med snedstreck:

"Detta syntes vara ett stort problem. /.../ Vi beslöt emellertid fortsätta."

Det naturligaste är antagligen att helt negligera sådana mönster, eller att läsa ut strängen sådan den står.

5 Diskussion

Vad som framkommit i denna undersökning är att Liberman & Church' algoritm i all sin enkelhet faktiskt klarar av att hitta merparten meningsslut i text. Vårt inkluderande av siffror, parenteser och kolon har procentmässigt inneburit procentuellt små förbättringar (se TABELL 3.1.). Däremot har denna utbyggnad inte lett till några svårjusterbara övergenereringar, varför utbyggnaden kan sägas vara motiverad.

En annan slutsats vi dragit rörande respektive algoritms prestanda är textavhängigheten. Som framgår av TABELL 3.3 så är distributionen av olika meningsavgränsare spridd. Det säger sig självt att texter som inkluderar repliker i någon form också innehåller fler mönster av typen KOLON-MELLANSLAG-CITATIONSTECKEN-VERSAL än rena faktatexter, t ex.

Referenser

Brodda, B. Forthcoming.

Church, Kenneth W och Liberman, Mark Y. 1989. *Text Analysis and Word Pronunciation in Text-to-speech Analysis*. DARPA WORKSHOP ON SPEECH AND NATURAL LANGUAGE.

Dougherty, Dale. 1991. *Sed & awk*. O'Reilly & Associates, Inc. Sebastopol, California.

Ejerhed, Eva, Källgren, G, Wennstedt, O och Åström, M. 1992. *The linguistic annotation system of the Stockholm-Umeå corpus project*. Umeå universitet.

Lindberg, J. (1993). *Kompendium i PCBETA-programmering*, Stockholms Universitet 1993.

Malkior, S och Carlvik, M. (1990). *PC BETA Reference* Ver. 90-06 Institute of Linguistics, Stockholm University 1990.

McAllister, Mike. 1989. *The Problems of Punctuation Ambiguity in Fully Automatic Text-to-Speech Conversion*. Proceedings of Eurospeech.

Wall, Larry och Schwartz, Randall. 1991. *Programming Perl*. O'Reilly & Associates, Inc. Sebastopol, California.