Vélrænn úrdráttur á flokkunarrömmum íslenskra sagna

V.FrAME

Námskeið: Sjálfstætt verkefni

Nemandi: Jökull Huxley Yngvason

Leiðbeinandi: Hrafn Loftsson

Efnisyfirlit

Lýsing á verkefni 3
Tól sem notuð voru við útfærslu 4
Útfærsluatriði á útdrætti flokkunarramma 5
Mat á nákvæmni forrits 7
Villugreining 10
Agnarsagnir ..... 11~~12~~
Notkunarleiðbeiningar 13
Lokaorð og vangaveltur um framhald 14
Heimildarskrá 15

Lýsing á verkefni

Flokkunarrammar sagna (e. verb subcategorisation frames) gefa upplýsingar um hvers konar fyllilíði sagnir geta tekið með sér. Þeir eru mikilvægir fyrir ýmis verkefni á sviði máltækni eins og mörkun (e. part-of-speech tagging) og þáttun (e. parsing). Í fyrra tilvikinu er hægt að nota flokkunarramma til að eyða margræðni og í því síðara til að stýra þáttara við byggingu þáttunartrés. Upplýsingar um flokkunarramma sagna geta einnig komið að notum í málvísindum, t.d. við rannsóknir á því hvernig málnotkun breytist á milli tímabila.

Í *Íslenskri setningafræði* eftir Höskuld Þráinsson (1999; bls. 136) eru tekin nokkur dæmi um mismunandi flokkunarramma íslenskra sagna (sögnin skáletruð, fyllilíðurinn feitletraður og ramminn til hægri)

(Hér er NL=nafnliður, FL=forsetningarliður; LL=lýsingarorðsliður og SL=sagnliður)

a. Ísinn hefur *bráðnað* [_]
b. Páll hefur *saknað* **þín** [_ NL]
c. Þeir hafa *fjallað* **um málið** [_ FL]
d. Friðrik hefur alltaf *verið* **grannur** [_ LL]
e. Þeir munu *hafa* **étið útsæðið** [_ SL]
f. Jón mun *lána* **Maríu hring** [_ NL NL]
g. Ég hef *stungið* **peningunum í vasann** [_ NL NL]
h. Hann hefur *málað* **bílinn rauðan** [_ NL LL]

Að setja saman flokkunarramma sagna í höndunum er verulegt átak. Fjöldi íslenskra sagna í gagnagrunni Beygingarlýsingar íslensks nútímamáls (BÍN; http://bin.arnastofnun.is/) eru um 7.700 og talið er að fjöldi sagna í íslensku sé um 9.000. Því er mikilvægt að þróa aðferðir sem geta sett saman flokkunarramma á vélrænan hátt. Upplýsingar um flokkunarramma íslenskra sagna eru til að mjög takmörkuðu leyti á tölvutæku formi hjá Árnastofnun. Í þessu verkefni var gerð tilraun til að draga upplýsingar um flokkunarramma sagna á sjálfvirkan hátt út úr íslenskum textum. Einblínt var á fyllilíði í formi nafnliða og/eða forsetningarliða. Íslenska greingartólið *IceNLP[1]* var notað við greininguna ásamt forritunarmálinu Perl.

Tól sem notuð voru við útfærslu

Fimm markarar voru notaðir til þess að marka málheild sem að samanstendur af einni milljón orða. Mörkunin sjálf átti sér ekki stað í þessu verkefni en hún var gerð síðastliðið sumar [2]. Það sem að átt er við þegar talað er um að marka málheild þá er átt við að merkja orðflokk og beygingarleg einkenni sérhvers orðs í málheildinni. Hér fyrir neðan er dæmi um setningu sem að hefur verið mörkuð.

Þessi faven setning nven hefur sfg3en verið ssg mörkuð sþgven . .

Hér eru rauðu orðin í setningunni svokölluð mörk og segja þau til um orðflokk og beygingarleg einkenni orðsins sem að stendur til vinstri við markið. Nánari útskýring á mörkum má sjá hér [3].

Til þess að auka nákvæmni er hægt að nota fleiri en einn markara og velja svo það mark sem að flestir markararnir völdu og má þannig reikna út líklegasta markið út frá meðaltali. *Combitagger* [4] er tól sem tekur úttak (e.output) sérhvers markara og velur svo algengasta mark orðanna og er það úttakið úr forritinu.

IceTagger [5] er málfræðilegur markari sem að markar íslenskan texta.

Bidirectional tagger [6] , *TnT [7]*, *fnTBL [8]* og *mxpost [9]* eru gagna markarar sem að marka texta óháð tungumáli.

IceParser [10] er tól sem að er að finna í *IceNLP* svítunni (e. *Suite*) sem að tekur markaðan texta sem inntak og þáttar setningar í setningarliði, þar að segja nafnliði, sagnliði o.s.frv., og setningafræðileg hlutverk eins og frumlög og andlög. Flokkunarrammarnir eru svo dregnir út úr úttakinu úr þessu tóli.

Lemmald [11] er tól sem að er að finna í *IceNLP* svítunni og er hlutverk þess að umbreyta sögnum yfir í nafnhátt sagnarinnar.

Perl forritunarmálið var valið til þess að skrifa forritið sem dregur flokkunarrammana úr úttakinu úr *IceParser.* Ástæðan fyrir því að *Perl* varð fyrir valinu er sú að textavinnsla er aðal hönnunarforsenda málsins og reglulegar segðir *(e. Regular expressions)* eru hluti af málinu.

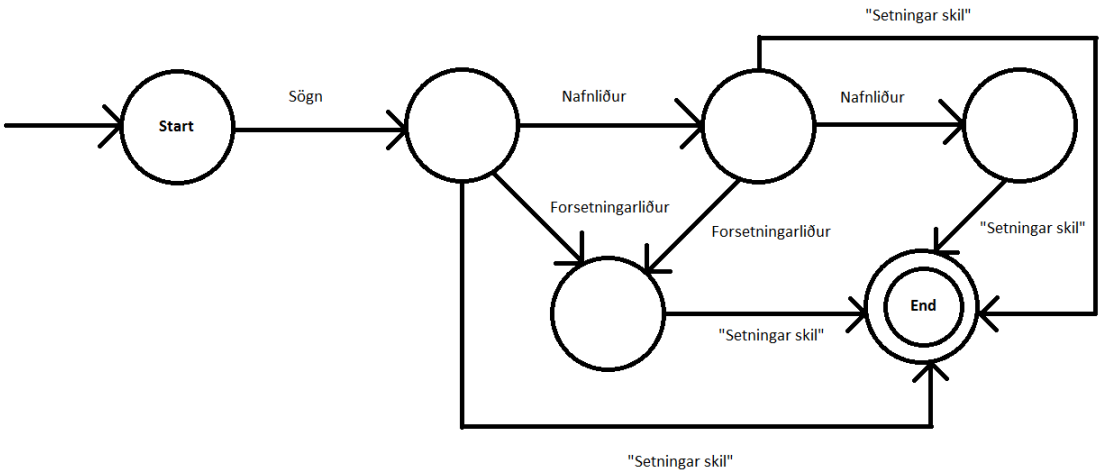
Ubuntu version 9.04 Jaunty Jackalope er stýrikerfið sem að forritið var þróað í. Ástæðan fyrir því að Linux var valið fram yfir Windows er sú að textavinnsla og stuðningur við skeljarforritun er mun auðveldari í Linux en Windows.

Útfærsluatriði á útdrætti flokkunarramma

IceParser þáttarinn er skrifaður sem stöðuvél, þetta þýðir að í hverri stöðu sem að hann er í er hann búin að bæta við upplýsingum við sérhverja setningu, svo sem nafnliði, sagnliði o.s.fr. Hægt er að keyra forritið þannig að sérhver staða er skrifað út í skrá. Þetta nýtti ég mér vegna þess að seinasta staða Iceparsers er notuð til þess að taka út upplýsingar um föll nafnliða en það er mikilvægt að hafa þær upplýsingar þegar ákvarðað er um hvaða falli sögn stýrir.

Func_SUBJ2 er staðan sem notuð var til þess að draga flokkunarramma úr. Þessi staða var valin því að allar upplýsingar varðandi föll á nafnliðum er að finna í þessari stöðu.

Eftirfarandi algrím var notað til að draga flokkunarramma sagnanna úr stöðunni (algrímið hefur verið einfaldað aðeins hérna fyrir neðan, meðal annars eru atviksliðir og annað tekið út til þess að draga ekki athyglina frá aðal efninu sem er hvernig flokkunarrammarnir eru myndaðir.)



Ástæðan fyrir því að setningarskilin eru höfð í gæsalöppum er sú að hér er ekki endilega átt bara við punkt. Setning getur nefnilega samanstaðið af fleiri en einni einfaldri setningu. Dæmi um slíka setningu er til dæmis: *Jón fór út í búð og þar verslaði hann í matinn.* Þessa setningu gætum við skrifað upp á annan máta. *Jón fór út í búð. Hann verslaði þar í matinn.* Í fyrri setningunni er 'og' merkt sem setningarskil og þá náum við báðum sögnunum sem að koma fyrir í setningunni.

Eins og sést á DFA (e. Deterministic finate-state automata) myndinni hér fyrir ofan sést að algrímið fer í gang þegar að fyrsta sögnin í setningunni finnst. Ef t.d. nafnliður finnst fyrir aftan sögnina er sögnin merkt með ramma sem að inniheldur nafnlið og fallið sem að nafnliðurinn er merktur með. Þegar að setningarskil finnast þá er farið í stöðuna **End** og ramminn vistaður fyrir sögnina. Eftir að

forritið hefur farið yfir alla málheildina er algengasti ramminn fyrir sérhverja sögn valinn sem rammi sagnarinnar. Ekki eru þó allar sagnir sem fundist hafa í keyrslunni hafðar með. Sögn er einungis merkt með ramma ef að einn tiltekinn rammi hefur að minnsta kosti komið fyrir eins oft og Filter 1 og Filter 2 segir til um (sjá -f1 og -f2 flag í notkunarleiðbeiningum). Þetta er gert til þess að minnka villur í forritinu en kemur aftur á móti niður á því hversu margar sagnir verða merktar með römmum. Ef að upp kemur að tveir mismunandi rammar eru jafn algengir þá er sá styttri valinn (sjá -s flag í notkunarleiðbeiningum) vegna þess að hann þykir líklegri rammi. Leyfilegir rammar fyrir sagnir eru eftirfarandi.

Tómur rammi:

Sögnin stýrir ekki falli og tekur ekki með sér forsetningarlið (þ.e. sögnin er áhrifslaus). Dæmi um slíka sögn er sögnin að anda.

Dæmi: Ég anda léttar.

Tómur rammi er táknaður sem [_] í úttaki forritsins.

Einn nafnliður:

Sögnin tekur með sér einn nafnlið í einhverju falli. Nafnliðurinn er andlag sagnarinnar. Dæmi um slíka sögn er að aflýsa.

Dæmi: Það þurfti að aflýsa fótboltaleiknum (sögnin stýrir þágufalli).

Rammi með einn nafnið er táknaður sem [_ NP<fall>] í úttaki forritsins.

Tveir nafnliðir:

Sögnin tekur með sér tvo nafnliði (beint og óbeint andlag) sem eru í sitt hvoru fallinu. Dæmi um slíka sögn er sögnin að gefa.

Dæmi: Hann gaf henni hnetur (sögnin stýrir þágufalli og svo þolfalli)

Rammi með tvo nafnliði er táknaður sem [_ NP<fall> NP<fall>] í úttaki forritsins.

Einn forsetningarliður.

Sögnin tekur með sér einn forsetningarlið. Dæmi um slíka sögn er sögnin að skríða.

Dæmi: Þau skriðu undir sumarbústaðinn.

Rammi með einn forsetningarlið er táknaður sem [_ PP] í úttaki forritsins.

Einn nafnliður ásamt forsetningarlið.

Sögnin tekur með sér andlag ásamt forsetningarlið. Dæmi um slíka sögn er sögnin að sækja.

Dæmi: Ég sótti hana út á flugvöll.

Rammi með nafnlið og forsetningarlið er táknaður sem [_ NP<fall> PP] í úttaki forritsins.

Mat á nákvæmni forrits

Til þess að geta metið nákvæmni forritsins var stuðst við svokallaðan gullstaðal sem fengin var hjá fyrirtækinu Friðrik Skúlason ehf. Gullstaðall er málheild sem að hefur verið búin til í höndunum, eða yfirfarinn í höndunum eftir að forrit hafa farið yfir málheildina. Þetta á bæði við flokkunarramma og mörkun texta ásamt fleiru sem að viðkemur náttúrulegum tungumálum. Gullstaðallinn sem að notast var við inniheldur ríflega 4000 sagnir sem að hafa verið yfirfarnar í höndunum. Hver sögn í gullstaðlinum hefur fleiri en einn ramma.

Dæmi um færslur í gullstaðlinum:

NF fjarlægja ÞF

NF fjarlægja ÞF af ÞGF

NF fjarlægja ÞF úr ÞGF

NF fjarlægjast ÞF

NF FT fjarlægjast (ATV)

Þetta er annað sniðmát en forritið notar en forritið umbreytir því yfir í samskonar sniðmát og úttakið úr forritinu. Þar sem að við höfum aðeins áhuga á föllunum/forsetningunum sem að sagnirnar taka með sér þá höfum við ekki áhuga á einstökum orðum og eru þau ekki höfð með þegar sniðmátinu er breytt (orðin merkt með rauðu hér fyrir ofan). Hins vegar ef að fall stendur fyrir aftan orð eins og hér fyrir ofan þá er um forsetningu að ræða og verður sögnin merkt með forsetningarramma. Einnig höfum við ekki áhuga á sögnum sem ekki taka með sér nefnifallsfrumlög þannig að aðeins sagnir sem að eru merktar með NF fyrir framan eru notaðar, einnig eru aðeins sagnir sem að eru í eintölu notaðar (sagnir í fleirtölu eru merktar með FT).

Sagnir merktar með (ATV) eru merktar með tómum ramma. Rammarnir eftir sniðmátinu hefur verið breytt fyrir þessar sagnir eru hér fyrir neðan.

fjarlægja [_ NPa]:[_ NPa PP]

fjarlægjast [_ NPa]

Taka skal samt fram að gullstaðallinn er ekki tæmandi listi yfir sagnir í íslensku og einnig má það teljast mjög ólíklegt að listinn af römmum fyrir þær sagnir sem að annað borð eru í gullstaðlinum sé tæmandi. Þegar forritið var keyrt á málheildina fundust sagnir sem að ekki var að finna í gullstaðlinum og í þeim tilfellum voru þær sagnir ekki hafðar með í útreikningum á nákvæmni forritsins. Nákvæmin var mæld á eftirfarandi hátt. Rammi fyrir sérhverja sögn var borin saman við alla ramma sagnarinnar í gullstaðlinum. Svo voru útreikningarnir eftirfarandi,

(fjöldi réttra ramma í úttaki forrits / heildarfjöldi ramma í úttaki forrits) * 100

Útreikningurinn byggir eingöngu á þeim sögnum sem finnast á annað borð í gullstaðlinum. M.ö.o. rammi telst réttur ef sögn í úttaki forrits finnst í gullstaðlinum og ramminn fyrir sögnina í úttakinu passar við rammann í gullstaðlinum. Heildarfjöldi ramma í úttaki forrits er þá heildarfjöldi ramma fyrir þær sagnir sem á annað borð finnast í gullstaðlinum.

Við þetta fæst prósenta sem að gefur hugmynd um nákvæmni forritsins.

Forritið býður upp á tvær síur til þess að auka nákvæmni en kemur niður á fjölda sagna sem að forritið setur ramma á. Önnur sían (sem héðan af verður vísað í sem sía 1) virkar þannig að aðeins þær sagnir sem að koma *x* sinnum fyrir eru notaðar. Hin sían (sem héðan af verður vísað í sem sía 2) virkar þannig að aðeins sagnir sem að eru með sama rammann *y* sinnum eru notaðar. Þessar síur virka vel að því leiti að sagnir sem að hafa aðeins einn ramma gætu verið sagnir sem að markararnir hafa valið ranga og eins er hin sían gagnleg því að ef að sami ramminn kemur fyrir oftar er einu sinni er líklegra að þar sé réttur rammi á ferð.

Hér fyrir neðan eru niðurstöður mælinga þar sem mismunandi síur voru notaðir.

Sía 1: 0 Sía 2: 0	<= Gildi á síum. Fyrsta sían þýðir að sögn má minnst hafa 0 ramma og sú seinni þýðir að rammi verður að minnsta kosti að koma oftar fyrir en 0 sinnum.
GOLD: TotalHits 967	<= Heildarfjöldi ramma í úttaki forrits (fyrir sagnir sem á annað borð finnast í gullstaðli).
GOLD: Incorrect matches 306	<= Fjöldi ramma sem að ekki fundust í gull staðlinum
GOLD: Correct matches 661	<= Fjöldi réttra ramma.
GOLD: Accuracy 68.4%	<= Hlutfall réttra ramma
Total number of verbs found: 2921	<= Heildarfjöldi sagna sem að var í úttaki forrits

Sía 1: 1	*Sía 2: 1*	*Sía 1: 3*	*Sía 2: 1*
GOLD: TotalHits	588	GOLD: TotalHits	540
GOLD: Incorrect matches	122	GOLD: Incorrect matches	112
GOLD: Correct matches	466	GOLD: Correct matches	428
GOLD: Accuracy	79.2%	GOLD: Accuracy	79.3%
Total number of verbs found:	1231	Total number of verbs found:	1088

Sía 1: 4	*Sía 2: 1*	*Sía 1: 5*	*Sía 2: 1*
GOLD: TotalHits	478	GOLD: TotalHits	434
GOLD: Incorrect matches	99	GOLD: Incorrect matches	86
GOLD: Correct matches	379	GOLD: Correct matches	348
GOLD: Accuracy	79.3%	GOLD: Accuracy	80.2%
Total number of verbs found:	938	Total number of verbs found:	835

Sía 1: 5	*Sía 2: 2*	*Sía 1: 5*	*Sía 2: 3*
GOLD: TotalHits	401	GOLD: TotalHits	352
GOLD: Incorrect matches	72	GOLD: Incorrect matches	58
GOLD: Correct matches	329	GOLD: Correct matches	294
GOLD: Accuracy	82.0%	GOLD: Accuracy	83.5%
Total number of verbs found:	757	Total number of verbs found:	648

Sía 1: 5	*Sía 2: 4*	*Sía 1: 5*	*Sía 2: 5*
GOLD: TotalHits	315	GOLD: TotalHits	281
GOLD: Incorrect matches	50	GOLD: Incorrect matches	43

GOLD: Correct matches	265	GOLD: Correct matches	238
GOLD: Accuracy	84.1%	GOLD: Accuracy	84.7%
Total number of verbs found:	562	Total number of verbs found:	489

Eins og sést hér fyrir ofan má auka nákvæmni forritsins með því að beita síunum, en fjöldi sagna sem fá ramma fækka einnig við notkun síanna.

Villugreining

Forritið er ekki villulaust eins og algengt er þegar unnið er með náttúruleg tungumál. Hugsanlegar villur má gróflega flokka í þrjá flokka. Fyrsti flokkurinn er þegar algrímið merkir sagnir með röngum ramma. Í annan flokk mætti setja orð sem að eru annað hvort merkt sem sagnir sem að eru í raun ekki sagnir (villur í mörkun) eða að IceParser gerir villu. Í þriðja flokknum eru svo orð sem að markararnir, IceParser og V.FrAME merkja rétt og sögnin finnst í gullstaðlinum en ramminn (sem að er réttur) er ekki til í gullstaðlinum.

Dæmi um villur í flokki 1: *(ATH: í öllum dæmunum hér fyrir neðan er sía 1 = 1 og sía 2 = 1)*

segja:[_ NPn]

Það er spurning með þennan ramma. Ef að maður tekur til dæmis dæmi um setningu þar sem að þessi sögn kemur fyrir:

Ég sagði *hestinum* sögu (hér tekur sögnin með sér þágufall).

Ástæðan fyrir því að algrímið í VFrAME velur nefnifalls ramman er sú að þegar sögnin að segja er í nafnhætti þá kemur nefnifall á eftir sögninni.

Dæmi úr málheildinni:

Þannig túlka ég að minnsta kosti myndbandið af hundunum og köttunum að læra að segja mamma (hér er mamma í nefnifalli).

helga:[_]

Samkvæmt gullstaðlinum tekur sögnin að helga ekki með sér tóman ramma og mér dettur ekki nein setning í hug þar sem að sögnin að helga myndi hafa tóman sagnramma.

Dæmi um villur í flokki 2:

it:[_] Hér er enska orðið it merkt sem sögn, sem að leiðir af sér að orðið kemur fyrir í úttaki VFrAME.

but: [_ NPn] Sama vandamál og með it.

Dæmi um villur í flokki 3:

dofna:[_ PP] Þessi sögn er í gullstaðlinum en ramminn finnst ekki. Dæmi þar sem þessi rammi á við: *Liturinn dofnaði í prentun*.

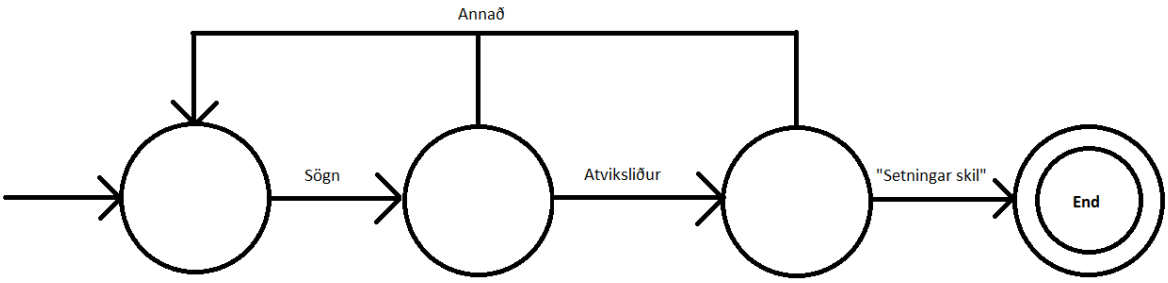
vinda:[_ NPd PP] Dæmi úr málheildinni: *Hann vatt sér inn í bílinn.*

dreyma:[_ PP] Mig dreymir í binary. (*ATH: Þessi rammi er til í gullstaðlinum en hann er ekki tekin með þegar gullstaðlinum er komið í sama sniðmát og forritið notar þar sem að sniðmátið á sögninni dreyma er frábrugðið öðrum sögnum í gullstaðlinum)*

Þetta eru aðeins örfá dæmi um villur sem að eru að finna fyrir þessa tileknu málheild. Til að bæta nákvæmni forritsins og losna við villur úr flokki 1 mætti skoða sagnir sem að taka með sér sagnliði sérstaklega. Hvað varðar flokk tvö þá er stöðugt verið að endurbæta markarana og IceParser en að vonast eftir að þessi tól nái 100% nákvæmni er óraunhæft. Ástæðan fyrir því er sú að íslenska er mjög flókið mál málfræðilega og mikið af undantekningum í málinu. Einnig mætti skoða hvernig tekið er á erlendum orðum, þar sem að heilu ensku setningarnar slettast oft í málfar fólks nú á dögum. Til þess að fækka villum í flokki 3 þarf einfaldlega að vera með stærri gullstaðal. Hafa skal það samt í huga að stærri gullstaðall eykur ekki nákvæmni forritsins heldur gefur aðeins nákvæmari mælingu á hversu nákvæmt forritið er. VFrAME gæti til dæmis nýst til þess að búa til nýjan gullstaðal með því að láta forritið fara yfir mjög stóra málheild og fara svo yfir niðurstöðurnar í höndunum.

Agnarsagnir

Forritið býður einnig upp á að finna svokallaðar agnarsagnir. Agnarsagnir er sögn ásamt atviksorði sem að sögnin tekur með sér. Þetta var ekki útlistað sem hluti verkefnisins en þetta var mjög einfalt að útfæra þar sem að samnýta mátti kóðan sem kominn var til þess að finna flokkunarrammana til þess að finna agnarsagnirnar. Eftirfarandi algrím var notað til að finna agnarsagnirnar.



Við hverja sögn er skoðað hvort að atviksliður standi fyrir aftan og ef svo er hvort að setningarskil séu þar fyrir aftan. Ef svo er þá er sögnin ásamt atviksorðinu vistuð sem sagnögn.

Dæmi um setningu í málheidinni þar sem að agnarsögn finnst.

Börnin í hverfinu voru við flugvallargirðinguna og *fylgdust með*.

Notkunarleiðbeiningar

Eins og áður hefur komið fram þá notast forritið við stöðuna Func_SUBJ2 úr *IceParser*. Í þessu ferjaldi eru fleiri en einn setningarliður í hverri línu. Keyra skal skriftuna(e.script) cleanParseSub2.sh til þess að koma Func_SUBJ2 ferjaldinu yfir í sniðmát sem að V.FrAME skilur. cleanParseSub2.sh tekur tvö viðfaung, annars vegar inntaksskrá (sem að á að vera Func_SUBJ2 ferjald) ásamt nafn á úttaksskrá. Dæmi um keyrslu

./cleanParseSub2.sh <inntaksskrá> <úttaksskrá>

V.FrAME getur tekið eftirfarandi fána.

-s

Stendur fyrir shortest match. Ef að tveir mismunandi rammar finnast jafn oft fyrir eina sögn er sá styttri valin. Ef þessum fána er sleppt er fyrsti ramminn valinn.

-g <STRING>

Ber úttak forrits saman við gullstaðal. Gullstaðlinum er breytt í sniðmát sem að V.FrAME notar. Forritið prentar svo niðurstöðuna út á skjá.

-d

Stendur fyrir debug. Ef þessi fáni er notaður þá skrifar forritið út hash töflurnar sem að notaðar eru til þess að ákvarða ramma sagnanna út í skrár. Þessar skrár eru eftirfarandi.

<úttaksskrá>.debug.lem	Hash tafla eftir að Lemmald hefur verið keyrt á sagnirnar.
<úttaksskrá>.debug.verb	Hash tafla áður en Lemmald hefur verið keyrt á sagnirnar.
<úttaksskrá>.debug.resultHash	Hash tafla sem að inniheldur sagnir eftir lemmun og eftir að algengasti ramminn hefur verið valin.
<úttaksskrá>.debug.gold	Hash tafla sem að inniheldur ramma gullstaðarins á sama sniðmáti og V.FrAME notar.

-sa

Ef þessi fáni er hafður með þá finnur forritið agnarsagnir í inntaki forritsins og skrifar þær í skrá sem að heitir <úttaksskrá>.advpv

-f1 <NO>

Þessi fáni stendur fyrir filter 1. Hann segir til um hversu oft sögn þar að hafa komið fyrir í inntaki til þess að sögnin verði merkt með ramma.

-f2 <NO>

Stendur fyrir filter 2. Þessi fáni segir til um hversu oft sögn þarf að hafa verið merkt með sama rammanum til þess að hún verði höfð með í úttaki forrits.

-i <STRING>

Inntak forritsins. Þetta skal vera úttak cleanParseSub2.sh scriptunar.

-o <STRING>

Nafn á úttaki forritsins.

Dæmi um keyrslu:

perl vframe.pl –s –d –sa –g output_ma.txt –i all.parsed.out –o all.frames.out

Lokaorð og vangaveltur um framhald

Í þessu verkefni var aðeins einn rammi merktur fyrir hverja sögn, en hver sögn getur haft fleiri en einn ramma. Framtíðar endurbætur á forritinu gætu meðal annars falist í því að velja fleiri en einn ramma fyrir hverja sögn. Hafa skal þá í huga að slíkt getur dregið mjög úr nákvæmni forritsins þar sem að rammar sem að koma sjaldan fyrir geta bent til þess að ramminn sé einfaldlega rangur. Einnig mætti prófa að velja ramma af handahófi í stað þess að velja þann ramma sem að er algengastur fyrir sögnina. Í [12] var bent á að nákvæmnin batnaði um 2% við að nota handahófs valin ramma í stað þess að nota ramma sem að komu oftast fyrir, en ekki er tekið fram af hverju niðurstöðurnar urðu betri. Einnig komst ég að því að til þess að forritið geti gefið betri niðurstöður þarf að nota málheild sem að er mjög stór, þar sem að aðeins 3000 sagnir fundust ef að hvorugur fillterinn var notaður og en færri ef að þeir voru notaðir. Árnastofnun er að vinna í að setja saman málheild sem að á að samanstanda af 25 miljón orðum en nú er notast við miljón orð. Upplagt væri að keyra forritið á þá málheild til að fá enn betri niðurstöðu.

Heimildarskrá

[1] Loftsson, H. and Rögnvaldsson, E. 2007. IceNLP: A Natural Language Processing Toolkit for Icelandic. *Proceedings of InterSpeech 2007, Special session: "Speech and language technology for less-resourced languages"*. Antwerp, Belgium.

[2] Yngvason, J. 2009. Mörkun og leiðrétting nýrrar málheildar. Verkefni styrkt af Nýsköpunarsjóði námsmanna.

[3]J. Pind, F. Magnússon, and S. Briem. 1991. Íslensk orðtíðnibók [The Icelandic Frequency Dictionary]. Orðabók Háskólans, Háskóli Íslands, Reykjavik.

[4] Verena Henrich, Timo Reuter and Hrafn Loftsson. 2009. CombiTagger: A System for Developing Combined Taggers. In *Proceedings of the 22nd International FLAIRS Conference, Special Track: "Applied Natural Language Processing"*. Sanibel Island, Florida, USA.

[5]IceTagger: Loftsson, H. 2008. Tagging Icelandic text: A linguistic rule-based approach. Nordic Journal of Linguistics, **31(1)**, 47-72.

[6] Dredze, M. and Wallenberg, J. 2008. Icelandic Data Driven Part of Speech Tagging. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 33–36*. Morristown, NJ, USA: Association for Computational Linguistics.

[7] Brants, T. 2000. TnT: A statistical part-of-speech tagger.In *Proceedings of the 6th Conference on Applied natural language processing, 224–231*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

[8] Ngai, G., and Florian, R. 2001. Transformation-based learning in the fast lane. In *Proceedings of the 2nd meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies 2001*, 1–8. Morristown, NJ, USA: Association for Computational Linguistics.

[9] Ratnaparkhi, A. 1996. A Maximum Entropy Model for Part-Of-Speech Tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 133–142. Morristown, NJ, USA: Association for Com-putational Linguistics.

[10] Loftsson, H. and Rögnvaldsson, E. 2007. IceParser: An Incremental Finite-State Parser for Icelandic. In J. Nivre, H-J. Kaalep, K. Muischnek and M. Koit (eds.), *Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA-2007)*. Tartu, Estonia.

[11] Ingason, A., Helgadóttir, S., Loftsson, H. and Rögnvaldsson, E. 2008. A Mixed Method Lemmatization Algorithm Using a Hierarchy of Linguistic Identities (HOLI). In B. Nordström and A. Ranta (eds.), *Advances in Natural Language Processing, 6th International Conference on NLP, GoTAL 2008, Proceedings*. Gothenburg, Sweden.

[12] Sarkar, A. and Zeman, D. 2000. Automatic Extraction of subcategorization Frames for Czech. In *Proceedings of the 18th conference on Computational linguistics*.Saarbrücken, Germany