CLT131 Korpusten käsittely

Nicholas Volk

Yleisen kielitieteen laitos, Helsingin yliopisto

HELSINGIN YLIOPISTON
YLEISEN KIELITIETEEN LAITOS

Syöterivien järjestäminen (sort)

- `sort`-komento järjestää syöterivin (aakkosjärjestykseen)
- Alla olevan esimerkin perl-komennon tulostaa numerot 11, 5 ja 30 omille riveilleen. Perl-komentoa ei tarvitse ymmärtää.

```
$ perl -e 'print "11\n5\n30\n";'
11
5
30
$ perl -e 'print "11\n5\n30\n";' | sort
11
30
5
$
```

- Eli normaalisti `sort` järjestää numeerisen syötteen ensimmäisen merkin mukaan.

Numeerinen järjestäminen (sort -n)

- `sort`-komennon option `-n` avulla peräkkäiset numeromerkit käsitetään samaan lukuun kuuluviksi:

```
$ perl -e 'print "11\n5\n30\n";' | sort -n
5
11
30
$
```

- Eli normaalisti `sort` järjestää numeerisen syötteen ensimmäisen merkin mukaan.

Käänteinen järjestäminen (sort -nr)

- `sort`-komennon option `-r` tulostus tulee käänteisessä järjestyksessä
- Eli aakkosilla ’z’ tulee ennen ’a’:ta ja numeroilla ’9’ ennen ’1’:tä
- Yhdistämällä `-n` ja `-r` saadaan siis suurin luku ensin

```
$ perl -e ’print "11\n5\n30\n";’ | sort -nr
30
11
5
$
```

Duplikaattien poisto (uniq)

- Komento `uniq` tulostaa peräkkäisistä identtisistä merkkijonoista duplikaatit:

```
$ perl -e 'print "a\nb\na\n";' | uniq
a
b
a
$ perl -e 'print "a\na\nb\n";' | uniq
a
b
$
```

- Eli järjestetään ensin:

```
$ perl -e 'print "a\nb\na\n";' | sort | uniq
a
b
$
```

Duplikaattien määrä (uniq -c)

- `uniq`-komennon optio `-c` kertoo peräkkäisten duplikaattien määrän:

```
$ perl -e 'print "a\nb\na\n";' | sort | uniq
a
b
$ perl -e 'print "a\nb\na\n";' | sort | uniq -c
      2 a
      1 b
$
```

- Yllä esitettyjä ja edellisellä kerralla tavattuja komentoja käyttämällä voi tehdä frekvenssilistan...
- ...joka on tämän kurssin tärkein asia säännölisten lausekkeiden ohella!

Frekvenssilistaa (1)

- Komennolla `tr ' ' '\n'` sai sanat omat rivilleen.
- Komennolla `sort` saadaan nämä aakkosjärjestykseen.
- Komento `head` tulostaa oletusarvoisesti 10 ensimmäistä riviä syötteestään.

```
$ tr -s ' ' '\n' < yhdestoista.txt | sort | head
"Enkä
"Suotta
Ahti
Ahti
Ahtia
En
Inkerelle,
Inkereltä:
Kaloin
Kasvoi
```

Frekvenssilistaa (2)

- Komento `uniq` tulostaa peräkkäisistä identtisistä merkkijonoista duplikaatit.
- Sen option `c` avulla saa laskettua sananmuodon peräkkäisten esiintymien määrän:

```
$ tr -s ' ' '\n' < yhdestoista.txt | sort | uniq -c | head
      1 "Enkä
      1 "Suotta
      2 Ahti
      1 Ahtia
      1 En
      1 Inkerelle,
      1 Inkereltä:
      1 Kaloin
      1 Kasvoi
      1 Kauan
```

Frekvenssilistaa (3)

- `sort`-komennon `r`-optio (reverse) järjestää syötteen käänteiseen järjestykseen (z tulee ennen a:ta).
- `n`-optio tulkitsee peräkkäiset numerot yhdeksi luvuksi.
- Nämä yhdistämällä saadaan numerot järjestettyä frekvenssilistaan, yleisin ensin:

```
$ tr -s ’ ’ ’\n’ < yhdestoista.txt | sort | uniq -c |
sort -nr | head
      8 on
      5 nälkä,
      4 eip’
      4 Saaren
      3 poiallehen:
      3 mennyt
      3 luona
      3 Viron
      3 Kosi
      2 öitä
```

Valmiihko frekvenssilista

- Komentorivillä voi luettavuuden parantamiseksi vaihtaa riviä kirjoittamalla kenoviivan \ ja rivinvaihdon peräkkäin
- Välimerkit tms. saattavat vääristää tulosta, joten otetaan ne pois (siis ennen järjestämistä ja laskemista):

```
$ tr -s ' ' '\n' < yhdestoista.txt | tr -dc 'A-Za-zäöÄÖ \n' | \
sort | uniq -c | sort -nr | head
      8 on
      7 nälkä
      4 eip
      4 Saaren
      3 poiallehen
      3 neiti
      3 mennyt
      3 lähe
      3 luona
      3 kasvoi
```

Merkkijonon poimiminen syötteestä (fgrep)

- `fgrep` (eli `grep -F`) on `grep`-perheen kuvausvoimaltaan heikoin lenkki.
- Komennolla saa poimittua haluttuja merkkijonoja syötteestä.
  Löydettyä riviä kutsutaan osumaksi.
- Etsitty merkkijono annetaan ensimmäisenä argumenttina:

```
$ fgrep neiti yhdestoista.txt
Kylli oli Saaren neiti, Saaren neiti, Saaren kukka.
eip’ on neiti mennytkänä; itse vasten vastaeli:
```

Täydellinen osuma (fgrep -x)

- Optio x tarkoittaa, että "osuma" kattaa koko rivin:

```
$ tr ’ ’ ’\n’ < yhdestoista.txt |fgrep neiti
neiti,
neiti,
neiti
$ tr ’ ’ ’\n’ < yhdestoista.txt |fgrep -x neiti
neiti
$
```

fgrep -i

- `i`-valitsin ei tee eroa ison ja pienen kirjaimen välille.
- Muiden kuin välillä `a-z` olevien kirjainten (mm. ääkköset) toiminnan varaan ei kannata laskea. Maakohtaiset asetukset (locale) eivät välttämättä ole kohdallaan...
- `uniq`-komennolla on samanniminen vastaava optio.
- `sort`-komennolla on `f`-optio, joka ajaa saman asian.

```
$ tr ' ' '\n' < yhdestoista.txt |fgrep -xi "en"
En
en
$ tr ' ' '\n' < yhdestoista.txt |fgrep -x "en"
en
$
```

Ympäröivien rivien mukaanotto

- Optio B *n* kertoo montako osumaa edeltänyttä riviä halutaan tulostaa
- A *n* tekee saman sanan jälkeen tuleville:

```
$ tr -s ' ' '\n' < yhdestoista.txt | fgrep -B2 -A2 -x tuli
kelpoavi;
vaan
tuli
vähän
vialle,
--
kuului:
kaukoa
tuli
kosijat
neien
```

Alkeellinen korkordanssi

Jo opittuja komentoja yhdistelemällä voi rakentaa alkeellisen konkordanssin:

```
$ tr -s ’ ’ ’\n’ < yhdestoista.txt | fgrep -B2 -A2 -x on | tr -s
’\n’ ’ ’ | tr -s ’-’ ’\n’
runo Vika on Ahtia sanoa,
 Saarelainen, tuo on lieto Lemmin
 poiallehen: eip’ on mennyt Päivälähän
 poiallehen: eip’ on mennyt Kuutolahan
 poiallehen: eip’ on mennyt Tähtelähän
 Inkereltä: eip’ on neiti mennytkänä;
 pänkerelle; siell’ on nälkä, kaiken
 nälkä." Tuop’ on lieto Lemminkäinen, $
```

Huomaa promptin paikka! Miksi se on samalla rivillä?

fgrep-komennon rajoitteet

- `fgrep` pystyy poimimaan halutun merkkijonon, mutta voidaan haluta muutakin kun etukäteen määriteltyjä merkkijonoja:
- Merkkijonot, joiden pituus on n-m merkkiä
- Tuntemattoman merkkijonon metsästys: *as X as*
- Suomen kielen tavurakenteen mukaiset sanat
- Sanat, joissa on takavokaaleja
- ...
- Näiden ongelmien ratkaisuun voidaan käyttää säännöllisiä lausekkeita...