T-61.5020 Luonnollisten kielten tilastollinen käsittely

Vastaukset 7, ke 14.3.2007, 12:15–14:00 — Sanojen merkitysten erottelu
Versio 1.0

Varsinaiset laskaritehtävät

1. Aloitetaan Bayesin kaavasta:

$$P(s_k|c) = \frac{P(c|s_k)P(s_k)}{P(c)}$$

Nyt meitä kiinnostaa vain todennäköisyyksien järjestys, eikä absoluuttinen arvo, joten voimme unohtaa normalisointitekijän $P(c)$:

$$\begin{aligned} s' &= \operatorname*{argmax}_{s_k} \frac{P(c|s_k)P(s_k)}{P(c)} \\ &= \operatorname*{argmax}_{s_k} P(c|s_k)P(s_k) \end{aligned}$$

Yhtälössä jälkimmäinen termi on sanan merkityksen prioritodennäköisyys, joka voidaan estimoida esimerkiksi laskemalla kuinka suuri osa opetusjoukon sanoista on esiintynyt merkityksessä s_k. Keskitytään tässä tarkastelemaan termiä $P(c|s_k)$.

Kontekstiksi valitaan vaikkapa sanaa ympäröivät 10 sanaa:

$$c = (\ w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9\ )$$

Tässä siis sana, jonka merkitystä arvioidaan olisi $w_{4.5}$ merkintöjen helpottamiseksi. Tässä siis sanojen järjestyksellä on väliä, sitä voidaan merkitä laittamalla sanat kaarisulkuihin. Tällaisilla piirrevektoreilla tunnistimen opettaminen on käytännössä mahdotonta, koska kahta täysin samaa 10 sanan kontekstia tuskin löytyy opetus- ja testijoukosta. Approksimoidaan tätä mallia olettamalle, että sanojen järjestyksellä ei ole väliä (aaltosulut):

$$c = \{\ w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9\ \}$$

Nyt meillä on siis:

$$P(c|s_k) = P(\{\ w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9\ \}|s_k)$$

Helpotetaan todennäköisyysjakauman estimointia ja oletetaan että sanat esiintyvät toisistaan riippumatta:

$$\begin{aligned} & P(\{\ w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9\ \}|s_k) \\ = \ & P(w_0|s_k)P(w_1|s_k)\ldots P(w_9|s_k) \\ = \ & \prod_{i=0}^{9} P(w_i|s_k) \end{aligned}$$

Kirjoitetaan vielä kaava auki

$$\begin{aligned} s' &= \operatorname*{argmax}_{s_k} P(c|s_k)P(s_k) \\ &= \operatorname*{argmax}_{s_k} \left(P(s_k) \prod_{i=0}^{9} P(w_i|s_k) \right) \\ &= \operatorname*{argmax}_{s_k} \left(\log P(s_k) + \sum_{i=0}^{9} \log P(w_i|s_k) \right) \end{aligned}$$

Viimeisellä rivillä kaava on kirjoitettu logaritmimuotoon. Tämähän voitiin tehdä, koska logaritmin otto ei vaihda lukujen suuruusjärjestystä. Riippuu tilanteesta onko suora vai logaritminen muoto kätevämpi.

Kannattaa huomata, että mikään matkan varrella tehdyistä approksimaatioista ei ole täysin oikein. Karkein virhe tehdään ehkä arvioidessa kontekstin sanat riippumattomiksi. Näin saadaan kuitenkin käyttökelpoinen menetelmä.

2. Käytetään edellisessä tehtävässä johdettua naiivin Bayestunnistimen kaavaa:

$$\begin{aligned} s' &= \operatorname*{argmax}_{s_k} P(c|s_k)P(s_k) \\ &= \operatorname*{argmax}_{s_k} \left(P(s_k) \prod_{i=0}^{N} P(w_i|s_k) \right) \end{aligned}$$

missä w_i on konteksissa esiintyneet sanat.

Tarvitsemme laskun suorittamiseen kahta estimaattia, todennäköisyyttä $P(w_j|s_k)$ että kontekstin sana w_j esiintyy merkityksen s_k kanssa ja merkityksen prioritodennäköisyyttä $P(s_k)$. Koska näytejoukossamme on yhtä monta esintymää merkitykselle *sataa=sade* ja *sataa=luku*, voimme ainoastaan asettaa prioritodennäköisyydeksi 0.5. Kirjan laskuissa sovelletaan järjestään ML-estimointia (suurimman uskottavuuden estimointi). Tehtävässä kuitenkin pyydettiin käyttämään prioreita, joten määritellään todennäköisyydelle $P(w_j|s_k)$ pieni priori, että kaikki sanat ovat yhtä todennäköisiä kaikissa konteksteissa ja lisätään seuraaviin estimaatoreihin $\lambda = 0.5$. Suuremman λ:n valinnalla voidaan korostaa prioriuskon merkitystä ja vähäinen todistus opetusjoukossa ei vielä suuremmin hetkauta tuota uskomusta. Tätä tapaa kutsutaan MAP (Maksimi A Posteriori) -estimoinniksi. Se voidaan ajatella vaikka niin, että kuvitellaan jo etukäteen nähdyksi opetusjoukon, jossa jokainen tunnettu sana on esiintynyt 0.5 kertaa molemmissa konteksteissa.

$$P(w_j|s_k) = \frac{C(w_j, s_k) + \lambda}{C(s_k) + N\lambda}$$

Tässä tunnettujen sanojen lukumäärä $N = 85$.

a) Lasketaan ensimmäisen lauseen merkityksen luokitteluun tarvittavat estimaattorit:

$$\begin{aligned}
P(\text{“koirasusitarha”}|\text{”sataa”=sade}) &= \frac{0.5}{6+0.5\cdot 85} = \frac{1}{97} \\
P(\text{“vieraili”}|\text{”sataa”=sade}) &= \frac{1}{97} \\
P(\text{“pari”}|\text{”sataa”=sade}) &= \frac{1}{97} \\
P(\text{“ihminen”}|\text{”sataa”=sade}) &= \frac{1}{97}
\end{aligned}$$

Huomataan, että ensimmäiseen merkitykseen ei saada kuin priorin aiheuttamaan todennäköisyysmassaa. Vertailuluvuksi (normalisoimattomaksi todennäköisyydeksi) saadaan

$$0.5\cdot\frac{1}{97}\cdot\frac{1}{97}\cdot\frac{1}{97}\cdot\frac{1}{97} = 5.6\cdot 10^{-9}$$

Lasketaanpa sama merkitykselle sataa=luku:

$$\begin{aligned}
P(\text{“koirasusitarha”}|\text{”sataa”=luku}) &= \frac{0.5}{6+0.5\cdot 85} = \frac{1}{97} \\
P(\text{“vieraili”}|\text{”sataa”=luku}) &= \frac{1}{97} \\
P(\text{“pari”}|\text{”sataa”=luku}) &= \frac{2+0.5}{6+0.5\cdot 85} = \frac{5}{97} \\
P(\text{“ihminen”}|\text{”sataa”=luku}) &= \frac{5}{97}
\end{aligned}$$

Vertailuluvuksi saadaan

$$0.5\cdot\frac{1}{97}\cdot\frac{1}{97}\cdot\frac{5}{97}\cdot\frac{5}{97} = 1.4\cdot 10^{-7}$$

Mallin mukaan merkitys sata=luku on siis todennäköisempi.

b) Kuten edellä lasketusta huomaamme, voimme jättää sanat, joita ei ole havaittu kummankaan sanan kontekstissa huomiotta, sillä ne eivät vaikuta vertailulukujen järjestykseen, vain ainoastaan niiden suuruuteen. Käytetään muunninta, joka muuntaa numerot muotoon “num” (tässä tapauksessa kolmantena=”num”).

Lasketaan siis 2. kohdassa todennäköisyydet ainoastaan seuraaville:

$$\begin{aligned}
P(\text{“räntää”}|\text{”sataa”=sade}) &= \frac{1.5}{6+0.5\cdot 85} = \frac{3}{97} \\
P(\text{“tai”}|\text{”sataa”=sade}) &= \frac{3}{97} \\
P(\text{“lunta”}|\text{”sataa”=sade}) &= \frac{7}{97} \\
P(\text{“num”}|\text{”sataa”=sade}) &= \frac{5}{97} \\
\\
P(\text{“räntää”}|\text{”sataa”=luku}) &= \frac{0.5}{6+0.5\cdot 85} = \frac{1}{97} \\
P(\text{“tai”}|\text{”sataa”=luku}) &= \frac{1}{97} \\
P(\text{“lunta”}|\text{”sataa”=luku}) &= \frac{1}{97} \\
P(\text{“num”}|\text{”sataa”=luku}) &= \frac{5}{97}
\end{aligned}$$

Vertailuluvuiksi saadaan

$$\begin{aligned}
\text{sataa=sade} &: \quad 0.5\cdot\frac{3}{97}\cdot\frac{3}{97}\cdot\frac{7}{97}\cdot\frac{3}{97} = 1.1\cdot 10^{-6} \\
\text{sataa=luku} &: \quad 0.5\cdot\frac{1}{97}\cdot\frac{1}{97}\cdot\frac{1}{97}\cdot\frac{5}{97} = 2.8\cdot 10^{-8}
\end{aligned}$$

Eli nyt ilmeisesti sana tarkoittaa sadetta.

c) Kolmannelle lauseelle saadan

$$\begin{aligned}
P(\text{“noin”}|\text{”sataa”=sade}) &= \frac{3}{97} \\
P(\text{“num”}|\text{”sataa”=sade}) &= \frac{5}{97} \\
\\
P(\text{“noin”}|\text{”sataa”=luku}) &= \frac{7}{97} \\
P(\text{“num”}|\text{”sataa”=luku}) &= \frac{5}{97}
\end{aligned}$$

Vertailuluvuiksi saadaan

$$\begin{aligned}
\text{sataa=sade} &: \quad 0.5\cdot\frac{3}{97}\cdot\frac{5}{97} = 8.5\cdot 10^{-8} \\
\text{sataa=luku} &: \quad 0.5\cdot\frac{7}{97}\cdot\frac{5}{97} = 2.0\cdot 10^{-7}
\end{aligned}$$

Eli nyt ilmeisesti sana tarkoittaa lukua.

d) Viimeisen lauseen sanojen todennäköisyyksiä ei annettu opetusdata muokkaa mihinkään suuntaan ja koska priorien mukaan molemmat merkitykset ovat yhtä todennäköisiä, ei malli pysty tekemään mitään päätöstä tässä tilanteessa.

3. Etsitään kaikille lauseen sanoille sanakirjamääritelmät. Verrataan näitä sanakirjamääritelmiä haettavan sanan kummankin merkityksen sanakirjamääritelmään. Kumman merkityksen selityksessä on enemmän yhteisiä sanoja lauseen sanojen selitysten kanssa paljastaa, kumpi merkitys on oikea.

   Tässä tapauksessa ampumista tarkoittavan merkityksen selostuksesta löytyy sanat "harjoitella" ja "varusmies", jotka löytyvät suoraan annetusta lauseesta. Sana "sarjatuli" löytyy sanan "kivääri" selityksestä, joten ampumista tarkoittavalle merkitykselle 3 pistettä.

   Lehmän ammunta tarkoittavan merkityksen selostuksesta löytyy sana "niityllä", joka löytyy myös suoraan lauseesta. Tälle merkitykselle 1 piste.

   Ilmeisesti siis nyt on kyseessä ampuminen (3>1).

Sivuhuomautus: Vaikka kirjan menetelmissä käytetään runsaasti sanaa Bayes, mitkään estimaatit siellä eivät ole Bayesiläisiä vaan perinteisiä maksimiuskottavuusestimaatteja.

Tietokonelaskarit

4. Katsotaanpa, kuinka monta osumaa Google antaa.

prices	go up	111 000
price	goes up	88 100
		199 100

prices	slant	58
prices	lean	2 520
prices	lurch	21
price	slants	1
price	leans	63
price	lurches	114
		2 777

   Tämän äänestyksen voittaa selvästi *kallistua* sanan merkitys *"go up"*, nousta.

   Entäpä toinen esimerkkimme? Jos teemme käännöksen ja haun noudattaen annettua sanajärjestystä, emme saa yhtään osumaa (pl. tämän laskaritehtävän edellisvuosilta). Kokeillaan siis etsiä dokumenttejä, joissa sanat esiintyvät missä tahansa järjestyksessä:

want	shin	hoof	liver	or	snout	260
like	shin	hoof	liver	or	snout	304
covet	shin	hoof	liver	or	snout	219
desire	shin	hoof	liver	or	snout	243
						1 026
want	kick	poke	cost	or	suffer	43 500

Huomataan, että sanojen verbimerkitykset voittavat tässä, vaikkakin tämä merkitys on ilmeisesti väärä. Kaikkia hakuja ei tarvitse edes suorittaa, koska jo ensimmäinen haku tuottaa enemmän osumia kuin toisten merkitysten haut yhteensä. Lisäksi suurin osa ensimmäisen 4 haun palauttamista osumista oli sanakirjoja. Huomataan, että koska merkitykset *shin, hoof, liver* ja *snout* ovat paljon harvinaisempia kuin verbimuodot, niitä myös löytyy suhteessa paljon vähemmän. Tässä tilanteessa pitäisi hakua varmaankin normalisoida jollain tavoin. Hakua vaikeuttaa myös se, että annettu lause ei ole kiinteä ilmaisu, kuten ensimmäisessä kohdassa.

5. Tehtävässä pitäisi siis arvioida merkityksen s_k todennäköisyys, kun tiedetään konteksti c_i.

$$P(s_k|c_i) = \frac{P(c_i|s_k)P(s_k)}{\sum_{k'=1}^{K} P(c_i|s_{k'})P(s_{k'})}$$

Käytetään aikaisemmin esitetty naiivin Bayesluokittiemen oletusta, että kontekstin w_j sanat eivät riipu toisistaan:

$$P(c_i|s_k) = \prod_{w_j \in c_i} P(w_j|s_k)$$

Alustus

Alustetaan EM-algoritmissa tarvittavat suureet:

– Asetetaan kaikki sanat yhtä todennäköisiksi kummallekin lähteelle ja lisätään tasajakaumaan hieman kohinaa σ. Ilman kohinaa algoritmi ei tule konvergoimaan, koska kaikki tapahtumat ovat yhtä todennäköisiä.

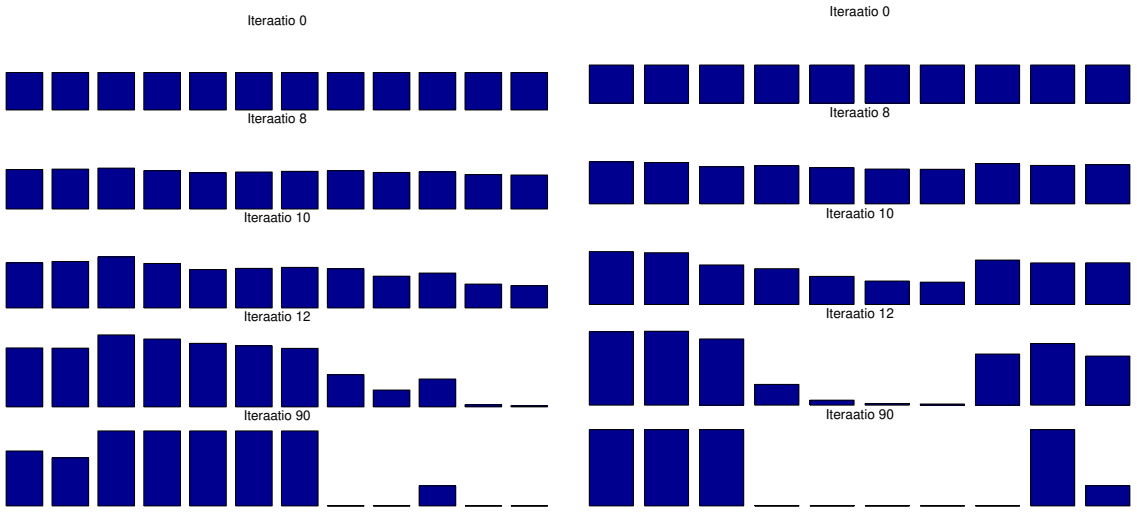
$$P(w_j|s_k) = \frac{1}{J} + \sigma$$

Tässä J on tunnettujen sanojen määrä.

– Asetetaan kaikki merkitykset yhtä todennäköiksi

$$P(s_k) = \frac{1}{K}$$

Tässä K on erilaisten merkitysten määrä.



(a) $P(w_j|s_0)$: Todennäköisyys, että merkitys s_0 tuottaa sanan w_j

(b) $P(s_0|c_i)$: Todennäköisyys, että konteksti c_i esiintyy merkityksen s_0 yhteydessä

Kuva 1: *EM algoritmi. Kuva esittää algoritmin konvergoitumisen. Sanat vasemmalta oikealle lueteltuna: yksi, kaksi, kolme, neljä, viisi, seitsemän, kahdeksan, mänty, leppä, haapa, koivu, kataja. Lauseet ovat samassa järjestyksessä kuin kysymyksessä.*

E- eli odotusarvoaskel

– Lasketaan kunkin merkityksen todennäköisyys kaikille konteksteille:

$$P(s_k|c_i) = \frac{\prod_{w_j \in c_i} P(w_j|s_k)P(s_k)}{\sum_{k'=1}^{K} \prod_{w_j \in c_i} P(w_j|s_{k'})P(s_{k'})}$$

M- eli maksimointiaskel

– Arvioidaan uudet sanatodennäköisyydet E-askeleessa arvioitujen lausetodennäköisyyksien avulla:

$$P(w_j|s_k) = \frac{\sum_{c_i : w_j \in c_i} P(s_k|c_i)}{\sum_{k'=1}^{K} \sum_{c_i : w_j \in c_i} P(s_{k'}|c_i)}$$

– Päivitetään prioritodennäköisyydet:

$$P(s_k) = \frac{\sum_{i=1}^{I} P(s_k|c_i)}{\sum_{k'=1}^{K} \sum_{i=1}^{I} P(s_{k'}|c_i)}$$

Kuvassa 1 on esitetty algorimin konvergointi, kun E- ja M-askelta iteroidaan vuorotellen. Tässä tapauksessa prioritodennäköisyydet $P(s_k)$ pidettiin $\frac{1}{2}$:ssa ensimmäiset

15 iteraatiota, mikä paransi algoritmin stabiilisuutta. Huomataan, että algoritmi kykenee pomimaan numerot ja puulajit erilleen. Lauseille 8 ja 9 malli ylioppii ja sijoittaa ne varmasti jompaan kumpaa merkitykseen. Datan määrän kasvaessa nämäkin estimaatit varmaan asettuisivat paremmin kohdalleen.

Käytännössä aivan samaa algorimia voidaan käyttää jakamaan dokumenttikokoelma eri aihepiireihin. Silloin kontekstina on koko dokumentti.

6. Tehtävän ratkaisu vaihe vaiheelta. Tärkeimmät kohdat, jossa on tehty mielivaltainen päätös, jonka voi aiheuttaa epätarkkuutta menetelmään ja jonka voisi helposti tehdä toisin on merkitty *kursiivilla*.

   1) Ensimmäinen tehtävä on siivota datasta kaikki ylimääräiset otsikot, tägit ja merkit pois. Sitten haluamme erottaa lauseet erilleen. *Pidetään kunkin sanan konteksina koko lausetta, missä se esiintyy.* Muutetaan jotkut kaksi sanaa yhdeksi sanaksi, esim. sanat "sade" ja "komissio". Merkitään myös muistiin oikeat vastaukset, vaikka ei käytetäkään niitä opetuksessa.

   2) Muutetaan kaikkien kontekstien sanat vektorimuotoon. Tässä voitaisiin käyttää binäärisiä indikaattorivektoreita, mutta *approksimoidaan niitä asettamalla joka sanalla satunnainen, yhden pituinen 200-ulotteinen vektori.* Jos vektori on tarpeeksi suuriulotteinen, se on suurinpiirtein ortogonaalinen kaikkien muiden sanavektoreiden kanssa ja approksimaatio ei aiheuta suurta virhettä.

   3) Oletetaan, että kontekstin *sanojen järjestys ei vaikuta.* Lasketaan kunkin sanan konteksti *summaamalla sitä ympäröivien sanojen vektorit yhteen ja jakamalla summa kontekstin sanojen määrällä.*

   4) Klusteroidaan kontekstivektorit, tässä *SOM-algoritmilla.* Päätetään sopiva klustereiden määrä *kokeilemalla.* Pienestä määrästä klustereita voi nopeammin silmämääräisesti arvioida menetelmän onnistumista, iso määrä klustereita voi antaa hienojakoisemman erottelun.

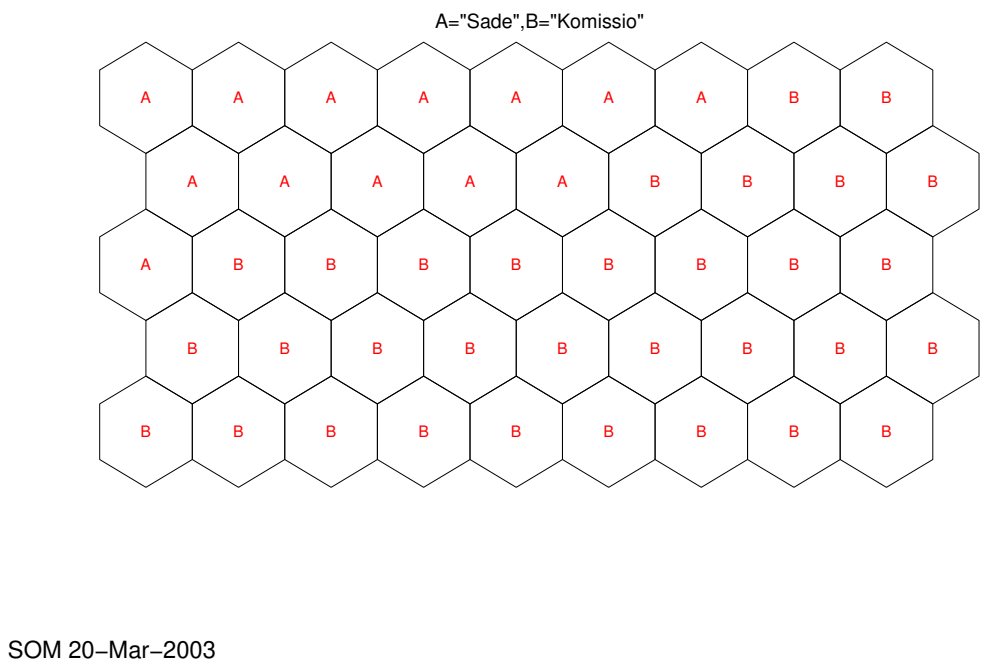
   5) Nyt pitäisi arvioida annetun klusteroinnin hyvyys. Ohjaamattomilla menetelmillä tämä on joskus hieman hankalaa, mutta tässä tapauksessa voidaan menetellä seuraavasti: Katsotaan ensin opetusjoukolla, menivätkö erimerkityksiset sanat siististi eri klustereihin. Tämähän ei sinänsä vielä todista paljoa: Jos valitaan yhtä monta klusteria kuin opetusjoukon sanoja, saadaan automaattisesti täysin oikea tulos. Käytetään kuitenkin näitä opetusjoukon näytteitä merkitsemään, joka klusteriin, mitä sanaa se edustaa. Siis se klusteri, jossa on enemmän merkitystä A (*suhteessa kummankin merkityksen opetusjoukon kokoon*) väittää, että kaikki sen lähelle tippuvat näytteet nyt kuuluvat varmasti merkitykseen A. Kokeillaan seuraavaksi testijoukkoa näitä merkityksiä vasten ja katsotaan, kuinka paljon merkityksistä menee oikein.

   Menetelmää käyttäen saadaan taulukossa 1 annetut tulokset. Tässä käytettiin 9×5 kokoista karttaa. Jos oikeita vastauksia ei ole saatavilla, on silmämääräisesti helpompi

arvioida tulokset vähemmästä määrästä ryhmiä. Esim. sanoilla “sade” ja “komissio”, 2×3 kartan tulokset olivat 59 % ja 98%. Kuvassa 2 on annettu sanojen “sade” ja “komissio” ryhmittyminen 9×5 kartalle.

Taulukko 1: *Tulokset*, 9×5 *kartta*

w_1	w_2	opetus		testi	
		w_1 oikein %	w_2 oikein %	w_1 oikein %	w_2 oikein %
Lappi	Pariisi	63	55	61	53
sade	komissio	66	93	66	92
Venäjä	tammikuu	80	60	78	60
Halonen	TPS	62	74	63	70
leijona	ydinvoima	70	55	75	48



Kuva 2: 9×5 *kartta*