# Käännösmuistit ja suomen kieli: kolmen käännösmuistiohjelman vertailua

Ville Martikainen

Pro gradu -tutkielma

Turun yliopisto

Kieli- ja käännöstieteiden laitos

Englanti

Toukokuu 2019

TURUN YLIOPISTO
Kieli- ja käännöstieteiden laitos / Humanistinen tiedekunta
MARTIKAINEN, VILLE:
Käännösmuistit ja suomen kieli: kolmen käännösmuistiohjelman vertailua

Tutkielma, 68 s.

Englannin kääntäminen ja tulkkaus

Toukokuu 2019

Tämä pro gradu -tutkielma käsittelee suomen kielen erikoispiirteiden vaikutusta muistiosumien tunnistamiseen kolmessa käännösmuistiohjelmassa. Kääntäjien teknisiä työkaluja käsittelevissä tutkimuksissa on perinteisesti keskitytty kääntäjien mielipiteisiin tai kokemuksiin käännösmuistien käytöstä, joten niiden käytännön toimivuuden tutkiminen on ollut vähäistä.

Tutkielma toteutetaan käytännönläheisesti: suomen erikoispiirteitä sisältäviä aineistoja ja vertailuaineistoja verrataan kolmessa käännösmuistiohjelmassa, jotka tuottavat prosenttiarvoisia muistiosumia sen perusteella, miten paljon tutkittavan aineiston segmentti muistuttaa vertailuaineiston segmenttiä. Kolmen ohjelman tuloksia vertaamalla voidaan määrittää, mitkä ohjelmista soveltuvat parhaiten suomen kieleen kääntämiseen, mitkä kielen piirteet ovat yleisesti kaikkein ongelmallisimpia ja mitkä aiheuttavat ongelmia yksittäisissä ohjelmissa.

Tulosten perusteella kaikki kolme ohjelmaa soveltuvat ainakin kohtuullisesti suomeen kääntävien käytettäviksi. Erityisesti näistä selvästi tunnetuin SDL Trados Studio vaikuttaisi tunnistavan jo oletusasetuksilla hyvin erilaisia kielen piirteistä johtuvia eroja, mutta vertailun kaksi muuta eivät nekään menesty huonosti. Sanajärjestys tuotti ohjelmilla heikoimmat tunnistustarkkuudet, joten sitä voidaan pitää kielen haastavimpana piirteenä – morfofonologinen vaihtelu sen sijaan vaikutti vähiten haasteita aiheuttavalta, mutta tutkielman rajallisuuden vuoksi tästä ei voida esittää varmoja väitteitä.

Tätä aihetta tulisi ehdottomasti tutkia enemmän, ja erityisesti yksittäisten muutosten, kuten yhden taivutuspäätteen, aiheuttamien vaikutusten tutkiminen olisi mielenkiintoista ja mahdollistaisi kielen piirteiden aiheuttaman vaikutuksen paremman arvioinnin. Toisaalta myös syvempi perehtyminen käännösmuistien ja muisteja käyttävien ohjelmien toimintaan olisi hyödyllistä.

Asiasanat: käännösteknologia, kääntäjän tekniset apuvälineet, käännösmuistit, käännösmuistiohjelmat, suomen kielen ominaispiirteet

# Sisällysluettelo

# 1. Johdanto

Ammattikääntämistä tutkitaan myös Suomessa melko paljon, mutta usein tutkimukset ovat joko kyselytutkimuksia kääntäjien oloista ja toiminnasta tai erilaisia käännöstekniikoita ja -tapoja käsitteleviä tutkimuksia. Kääntäjien työkalut eivät ole perinteisesti saaneet vastaavaa huomiota, mikä johtunee ainakin osittain siitä, että tutkimuksen kannalta mielekkäät tekniset apuvälineet ovat yleistyneet toden teolla vasta viimeisten muutaman vuosikymmenen kuluessa. Käytön yleistyminen on kuitenkin ollut niin nopeaa, että mielestäni myös näitä työkaluja on aika tarkastella seikkaperäisemmin.

Kääntäjien teknisillä työkaluilla tarkoitetaan usein erilaisia konekääntimiä, mukaan lukien monille tuttu Google Translate, tai pääosin käännösmuistien käyttöön perustuvia käännösmuistiohjelmia. Erityisesti konekäännöksiä tuottavista ohjelmistoista on tehty monia mielenkiintoisia tutkimuksia, mutta päivittäisessä käytössä yleisempien käännösmuistiohjelmien osalta tutkimustarjonta on suppeampaa. Osittain tämä johtuu varmasti siitä, ettei käännösmuisti ole teknologiana enää erityisten uutta tai jännittävää toisin kuin esimerkiksi juuri konekääntäminen, jossa uusia menetelmiä ja saavutuksia julkaistaan jatkuvasti. Vanhaakin teknologiaa voi silti parantaa, ja tämän tutkielman tarkoituksena on pohtia vielä tunnistettavia ongelma-alueita.

Tässä tutkielmassa tarkastelen sitä, millaisia ongelmia suomen kielen erilaiset ominaispiirteet aiheuttavat kolmelle tarkasteltavalle käännösmuistiohjelmalle. Tarkastelemiini kielen ominaispiirteisiin lukeutuvat esimerkiksi taivutusjärjestelmä ja sanajärjestys, joiden vaikutusta ohjelmien käännösosumien tunnistustarkkuuteen vertailen käännösmuistien avulla (The Finnish Language in the Digital Age 2012, s. 11). Käytän suomen kielen piirteiden vaikutusten arvioimiseen kolmea eri käännösmuistiohjelmaa, joiden tuloksia vertaamalla saan yleiskäsityksen mahdollisista ongelmista ja pystyn lisäksi vertailemaan kyseisten työkalujen soveltuvuutta erityisesti suomeen päin kääntäville. Käytetyt ohjelmat on valittu edustamaan käännösteknologia-alan eri osa-alueita, sillä en usko, että kolmen suosituimman ohjelman vertailututkimus olisi tuottanut erityisen mielenkiintoisia tuloksia niiden samankaltaisuudesta johtuen.

Aloitan kuitenkin esittelemällä tutkielman kannalta olennaisia käännösteknologian käsitteitä, minkä jälkeen esittelen edellä mainittujen käännösmuistiohjelmien tärkeimmät ominaisuudet – keskittyen erityisesti toimintoihin, jotka erottavat ne kilpailijoistaan – ja hieman niiden historiaa. Työvälineiden esittelemisen jälkeen on luontevaa siirtyä suomen kielen piirteiden kartoittamiseen. Koska aihe on erittäin laaja, on se rajattava niihin piirteisiin, joilla on eniten merkitystä tämän tutkielman kannalta: siis suomen kielen muista – tai ainakin yleisimmistä – kielistä erottaviin piirteisiin sekä niihin ominaisuuksiin, joiden voisi olettaa aiheuttavan eniten ongelmia suomeen päin käännettäessä.

Käännösohjelmien ja tutkittavien kielen piirteiden käsittelyn jälkeen esittelen tutkimusaineiston, jonka sisältämiä tekstejä on muokattu tutkimustarkoitukseen sopiviksi. Itse muistien ja käännösmuistiohjelmien toiminnan tutkimista ei voi suunnitella täydellisesti alusta loppuun, sillä eteen tulee lähes väistämättä tilanteita tai kysymyksiä, joita on tarkasteltava lähemmin ennen lopullisten päätelmien muodostamista. Käsittelen tällaiset tilanteet niiden ilmaantuessa.

Tutkielma päättyy luonnollisesti yhteenvedon ja loppupäätelmien tekemiseen: miten hyvin käännösmuistiohjelmat pystyivät tunnistamaan suomen kielen erikoisuuksia, mitkä niistä aiheuttivat eniten ongelmia, esiintyikö tuloksissa huomattavaa vaihtelua eri ohjelmien välillä ja mitä suomen kieleen kääntävien pitäisi pitää mielessä käännöstyökaluja käytettäessä. Tutkielma herättää varmasti myös mielenkiintoisia lisäkysymyksiä käännösohjelmiin ja suomen kieleen liittyen, joten mainitsen myös niistä tutkielmani päätteeksi.

# 2. Käännösmuisti

## 2.1 Esittely

Käännösmuisti on kääntäjän teknisistä apuvälineistä konekääntämisen ohella kaikkein tunnetuin ja eniten käytetty. Toisin kuin konekääntämisessä, käännösmuistin avulla käännettäessä käännöstyökalun ehdottamat käännökset ovat (ainakin useimmiten) kääntäjän itsensä tai samaa muistia hyödyntävän toisen kääntäjän aikaisemmin tuottamia käännöksiä. Käännösmuisti toimii siis eräänlaisena tietokantana, johon käännösmuistiohjelma tallentaa uusia käännöksiä ja jossa jo olevia käännöksiä se vertaa kääntäjän työstämään tekstiin (Bowker 2002, s. 94). Käännösmuistin tuottamien käännösehdotusten luotettavuus riippuu näin ollen kääntäjän itsensä ammattitaidosta, eikä se juurikaan helpota täysin uudenlaisen tekstin kääntämistä. Itse käännösehdotukset voidaan muistin tyypistä riippuen tuottaa eri menetelmillä, kuten vertailemalla yksinkertaisesti virkkeessä (tai muussa käännössegmentissä) esiintyviä merkkijonoja tai pilkkomalla virkkeen syntaksisiin yksiköihin (Lagoudaki 2006, s. 4). Näistä jälkimmäinen on kuitenkin hyödynnettävissä vain pienellä kieliparien määrällä menetelmän monimutkaisuudesta johtuen (mt.).

Käännösmuisti koostuu käännösyksiköistä eli käännettävän tekstin segmenteistä ja niitä vastaavista käännöksistä. Segmentti on usein yksittäinen virke, mutta monet ohjelmat tukevat myös kappaletason segmentointia (Somers 2003, s. 31). Halutessaan käyttäjä voi asettaa myös ns. katkaisumerkkejä, jotka erottavat peräkkäiset segmentit toisistaan – näin esimerkiksi pilkun tai kaksoispisteen molemmin puolin esiintyvät lauseet voidaan erottaa muistissa erillisiksi käännössegmenteiksi. Käännösmuistin oikeanlainen segmentointi onkin avainasemassa muistista saatavaa hyötyä tarkasteltaessa (Colominas 2008, s. 343). Segmentointi on sitä luonnollisempi, mitä enemmän kieliparin kielet muistuttavat toisiaan rakenteeltaan, minkä vuoksi esimerkiksi suomen ja englannin välillä kääntävät voivat kokea käännösmuistiohjelman oletussegmentoinnin hankalaksi.

Käännösmuistien käytöstä saatavaa hyötyä on tutkittu jonkin verran. Tutkimusten mukaan käännösmuistin käyttäminen parantaa kääntäjän tuottavuutta 10–70 %, mutta niistä saatava hyöty riippuu kääntäjän käännöstyökalutuntemuksen lisäksi myös käytettävän käännösmuistin laadusta (Yamada 2011, s. 63, 69). Lisäksi käännösmuistin käyttö auttaa

pitämään erityisesti johdonmukaisuuteen liittyvät käännösvirheet minimissään (Arenas 2008, s. 16). On tietysti selvää, että myös käännettävän tekstin tyyppi vaikuttaa aiemmista käännöksistä saatavaan hyötyyn. Kaunokirjallisuutta käännettäessä muisteista on todennäköisesti enemmän haittaa kuin hyötyä, sillä esimerkiksi jonkin tietyn lauserakenteen toistuminen segmentistä toiseen ei ole lukukokemuksen kannalta tarkoituksenmukaista. Suurin hyöty saavutetaankin niissä teksteissä, joiden rakenteen on tarkoituskin pysyä samanlaisena tekstistä toiseen. Tällaisia ovat useat tekniset käännökset, kuten käyttöoppaat ja laitteiden turvallisuusohjeet, sekä esimerkiksi patentti- ja käyttöehtotekstit (Biçici & Dymetman 2008, s. 454–455). Muistien haittoihin voitaneen laskea myös yhä yleistyvä käytäntö, jossa kääntäjä ei voi välttämättä käyttää omia käännösmuistejaan, vaan hän joutuu työskentelemään käännöstoimistossa ”esikäännetyn” dokumentin parissa (Garcia 2009, s. 202). Tämän jälkeen käännetty teksti lisätään jälleen toimiston suureen käännösmuistiin, jolloin yhden kääntäjän työtä voidaan hyödyntää muiden kääntäjien töiden hinnoista neuvoteltaessa, sillä osittaisista tai täydellisistä muistiosumista ei usein makseta täyttä korvausta.

Bowkerin (2005, s. 19) käännösopiskelijoilla teettämän tutkimuksen mukaan käännösmuistijärjestelmät paransivat edellä mainitun mukaisesti kääntämisen tuottavuutta, mutta niiden käyttöön liittyi myös joitakin ongelmia. Yksi näistä oli opiskelijoiden kritiikittömyys muistista löytyneitä osumia kohtaa: erityisesti ns. täydellisissä osumissa (*100% match*) olleet virheet päätyivät lähes poikkeuksetta myös uuteen käännökseen (mt.). Toinen ongelma liittyi tyyliin: käännösmuistissa olevat käännösyksiköt ovat usein peräisin eri teksteistä ja tyyliltään erilaisia, joten niiden lisääminen tekstiin muokkaamattomina voi huonontaa tekstin luettavuutta (mt.). Bowker korostaakin kääntäjän vastuuta sekä käännösmuistin tekemisessä että sen käytössä.

Edes muistien huolellinen käyttö ei kuitenkaan ratkaise kaikkia käännösmuisteihin liittyviä haasteita. Koska kääntäjät kääntävät harvemmin täsmälleen identtisiä tekstejä uudelleen ja uudelleen, käännösmuistista ei voi olettaa löytyvän useinkaan täydellisiä osumia. Kääntäjien onneksi nykyaikaiset käännösmuistiohjelmat kykenevät tunnistamaan myös osittaisia osumia (*fuzzy match*), eli käännösyksiköitä, joissa on riittävästi samaa sisältöä käännettävään kohtaan verrattuna, jotta muistista saatavaa osumaa voidaan pitää

hyödyllisenä. Riittävä vastaavuus riippuu käännösmuistista ja käännösmuistiohjelmasta, mutta ennen kaikkea kääntäjän tekemistä valinnoista ohjelman asetuksissa. Asetusten huolellinen määrittäminen on erityisen tärkeää, kun kieliparin toisena kielenä on suomi, mainituista suomen kielen erityispiirteistä johtuen.

Muistista löytyneitä täydellisiä, 100-prosenttisia, osumia pidetään usein kääntäjän kannalta ihanteellisina erityisesti niiden kääntämistä nopeuttavan vaikutuksen vuoksi. Edes kahta sanasta sanaan identtistä lähdetekstin virkettä ei voi kuitenkaan välttämättä kääntää identtisesti, sillä niiden kontekstit voivat olla täysin erilaiset. Jotkin nykyaikaiset käännösmuistiohjelmat helpottavat tämän arvioimista tuottamalla täydellisten osumien lisäksi myös kontekstiosumia ja täydellisiä kontekstiosumia. Kontekstiosumissa (näistä käytetään eri yhteyksissä myös nimityksiä 101-prosenttinen osuma ja *context match*) käännettävä segmentti löytyy sellaisenaan muistista, mutta lisäksi joko sitä edeltävä tai seuraava segmentti on niin ikään identtinen. Täydellisissä kontekstiosumissa, tai 102-prosenttisissa osumissa, sekä edeltävä että seuraava segmentti ovat samat kuin muistissa olevaa segmenttiä ympäröivä konteksti. Näin ollen käännösmuistit voivat parhaimmillaan tuottaa erittäin luotettavia osumia, mutta viimeinen vastuu niiden käytöstä on toki tällöinkin kääntäjällä.

## 2.2 Muistiosumien tunnistaminen

Kuten aiemmin mainittiin, käännösmuisti voi tarjota kääntäjälle sekä täydellisiä että osittaisia osumia. Täydellisten osumien muodostamisperiaate on yksinkertainen: mikäli käännettävänä oleva segmentti löytyy käännösmuistista täysin identtisenä, se voidaan esittää kääntäjälle täydellisenä osumana. Osittaisten osumien tunnistaminen ja niiden arvottaminen on kuitenkin huomattavasti monimutkaisempi operaatio, ja tähän tarkoitukseen onkin kehitetty erilaisia algoritmeja, joiden pohjalta useimpien käännöstyökalujen muistihaut toimivat. Lisäksi jotkin käännösmuistiohjelmat tukevat myös niin sanottua subsegmentaalista tunnistamista (*subsegment matching*), joka tunnistaa segmenttien osia, mutta kyseistä toimintoa ei erikseen tutkita tässä tutkielmassa (Flanagan 2015, s. 65, 74).

Algoritmeja käytetään käännettävän segmentin ja muistissa jo olevien käännösten vertailuun jonkin matemaattisen kaavan pohjalta. Usein vertailu perustuu yksittäisiin merkkeihin, jolloin puhutaan merkkijonovertailusta. Merkkijonometriikalla voidaan kvantifioida kahden merkkijonon välisiä eroja, jolloin tulokseksi voidaan saada esimerkiksi juuri käännösmuisteista tuttuja prosenttilukuja. Käännösmuistiohjelmien hyödyntämiä algoritmeja ei ole kuitenkaan kehitetty ainoastaan kääntämistä varten, vaan niitä hyödynnetään laajalti esimerkiksi hakukoneiden tuottamissa hakuehdotuksissa tai DNA:n mutatoitumisasteen selvittämisessä (Navarro 2001, s. 32).

Matematiikassa metriikalla viitataan pisteiden välisiin etäisyyksiin, mutta etäisyydestä voidaan puhua myös kahta merkkijonoa vertailtaessa: tällöin etäisyys on usein niiden toimenpiteiden määrä, mikä vaaditaan, että merkkijono saadaan muutettua toiseksi merkkijonoksi (Jokinen, Tarhio & Ukkonen 1988, s. 1440). Vaadittujen toimenpiteiden määrää kutsutaan yleisesti myös muokkausetäisyydeksi (Mustonen 2014, s. 13). Kun sallittuja toimenpiteitä ovat merkkien lisääminen, poistaminen ja vaihtaminen ja kun jokaisen toimenpiteen etäisyys on yksi, puhutaan niin sanotusta Levenšteinin etäisyydestä (Navarro 2001, s. 37).

Edellä esitetyn mukaan esimerkiksi merkkijonojen Kala ja Kanat muokkausetäisyys (eli Levenšteinin etäisyys) on kaksi, sillä vaadittavat toiminnot ovat merkin *l* muuntaminen *n*:ksi sekä *t*-merkin lisääminen. Kun Kala-merkkijonon muuttaminen vertailtavaan Kanat-muotoon vaatii kaksi toimenpidettä, on kyseessä 60 %:n muistiosuma eli vastaavuus (1,0 − 2/5 = 0,6). Mitä lyhempi segmentti on kyseessä, sitä enemmän yksittäiset erot luonnollisesti korostuvat – usean sanan virkkeessä, joka on yksittäistä sanaa todennäköisempi todellisessa käännöstoimeksiannossa, tällaiset muutaman merkin eroavaisuudet edustavat vain murto-osaa kokonaisuudesta, jolloin vastaavuusprosentti on selvästi suurempi. Lisäksi kannattaa huomata, että puhtaasti merkkijonojen vertailuun pohjautuva matemaattinen vertailualgoritmi ei osaa huomioida samojen sanojen esiintymistä virkkeen toisessa kohdassa. Täten esimerkiksi luettelojen ”yksi, kaksi, kolme” ja ”kolme, kaksi, yksi” vastaavuusprosentti on huonompi kuin luettelojen ”yksi, kaksi, kolme” ja ”kaksi, kaksi, kolme” vastaava. Käännösmuistiohjelmissa voi olla käytössä myös

muita menetelmiä, jotka tunnistavat samojen sanojen esiintymisen eri järjestyksessä: tätä testataan myöhemmin tutkielman sanajärjestysosiossa.

Useimpien käännösmuistiohjelmien tapauksessa käytetty vertailumekanismi jää arvailujen varaan, sillä sitä ei ymmärrettävästi haluta antaa ilmaiseksi kilpailijoiden käyttöön. Vertailumekanismin ymmärtämisessä parhaimmaksi lähteeksi osoittautuu OmegaT, jonka avoin lähdekoodi on vapaasti jokaisen tarkasteltavissa. OmegaT:n vertailumetodi perustuu sen kehittäjän mukaan Levenšteinin etäisyyteen niin, että OmegaT:ssä algoritmia sovelletaan yksittäisten merkkien sijaan kokonaisiin sanoihin (Briel 2013b). Levenšteinin etäisyyden soveltaminen sanatasolla säästää todennäköisesti jonkin verran laskentatehoa, sillä jokaista sanaparia ei tarvitse vertailla viimeiseen merkkiin asti. Toteutus voi kuitenkin haitata muistiosumien tarkkuutta huomattavasti, kun tarkasteltavana on kieli, jossa taivutuspäätteet liitetään suoraan sanavartaloon: tällaisessa tapauksessa esimerkiksi sanojen Koira ja Koirat vastaavuusprosentti olisi pyöreät nolla, sillä kyseessä ovat kuitenkin eri sanat. Ohjelman kehittäjät näyttävät kuitenkin ymmärtäneen tämän, sillä OmegaT:n uudemmat versiot sisältävät tokenisointityökalun, mikä ainakin teoriassa tunnistaa eri taivutuspäätteitä sisältävät sanat samoiksi sanoiksi. Tokenisoinnin toimivuus käytännössä selviää sekin myöhemmin.

# 3. Käännösmuistiohjelmien esittely

Kuten johdantoluvussa mainittiinkin, tämä tutkielma perustuu suurelta osin käännösmuistiohjelmien vertailuun käytännön kääntämistä mukailevassa tilanteessa. Vertailemalla erilaisia ohjelmia vältetään tilanne, jossa koko tutkielma perustuisi sattumalta esimerkiksi markkinoiden ainoaan erityisen huonosti (tai yhtä lailla erityisen hyvin) toimivaan ratkaisuun. Lisäksi vertaileva tutkimus on jo itsessään mielenkiintoista ja voi tuottaa erityisesti suomalaisille kääntäjille hyödyllistä tietoa sopivimman työkalun valitsemisen helpottamiseksi. Tässä luvussa esittelen tutkielmaan valitsemani kolme käännösmuistiohjelmaa, SDL Trados Studion, CafeTranin ja OmegaT:n, painottaen erityisesti niiden teknisiä ominaisuuksia, mutta hieman myös taustoja. Ensimmäisenä esittelen useimmille kääntäjille tutun jo Tradoksen, mistä on loogista siirtyä kohtuullisen tunnetun ja pitkään saatavilla olleen OmegaT:n kautta käännöstyökalualan kuopuksiin lukeutuvaan CafeTraniin.

## 3.1 SDL Trados Studio 2017

SDL Trados Studiolla, tai lyhyemmin vain Tradoksella, on näistä kolmesta ylivertaisesti pisin historia, johon sisältyy monia vaiheita ja käänteitä. Tradoksen taustoihin on mielenkiintoista perehtyä jo pelkästään siksi, että se peilaa melko hyvin käännösteknologian kiistatonta läpimurtoa alalla viimeisten kymmenien vuosien aikana. Lisäksi Trados lukeutuu kaikkein suosituimpiin käännösmuistiohjelmiin, joten sen sisällyttäminen tutkielmaan on jo siksi perusteltua (Suppanen 2015, s. 72). Trados ei ole kaikkein vanhin käännösmuisteja hyödyntävä ohjelma (joitakin alkeellisia termijärjestelmiä on kehitelty jo 1960-luvulla), mutta pitkään jatkuneen suosionsa vuoksi sen eri vaiheet ovat monille jo entuudestaan tuttuja, mikä tekee Tradoksesta sopivan kiinnekohdan myös itse käännösteknologian historian tarkasteluun (Garcia 2012, s. 453).

Jochen Hummel ja Iko Knyphausen perustivat TRADOS Gmbh:n Stuttgartissa vuonna 1984 (SDL Trados: About Us, n.d.). Yritystä ei perustettu alkujaan kehittämään käännösohjelmistoa, vaan perustajat käyttivät sitä tarjotakseen käännöspalveluita IBM:lle (mt.). Hummel ja Knyphausen tajusivat kuitenkin pian käännösalan kaipaavan teknisiä apuvälineitä alati kasvavien tekstimäärien vuoksi, ja vuonna 1988 he saivatkin valmiiksi ensimmäisen, TED-nimisen käännösmuistiohjelmansa (mt.). TED:n julkaisu johti yrityksen

jakautumiseen niin, että TRADOS keskittyi ainoastaan käännösteknologiatyöhön erillisen INK:n hoitaessa käännöspalveluiden tarjoamisen (mt.).

Vielä nykyäänkin ohjelmistopakettiin kuuluva Multiterm-termityökalu julkaistiin vuonna 1990 (SDL Trados: About Us n.d.). Multiterm oli TRADOS-yrityksen ensimmäinen julkaistu ohjelma (TED toimi ainoastaan pohjana myöhemmälle kehitystyölle), minkä lisäksi siitä neljä vuotta myöhemmin julkaistu 1.5-versio oli yrityksen ohjelmista ensimmäinen, joka hyödynsi tuolloin vielä melko tuoretta *fuzzy matching* -teknologiaa (mt.). Vuonna 1994 julkaistiin myös ensimmäinen Translator's Workbench, joka otettiin pian käyttöön esimerkiksi Euroopan komissiossa (mt.). Ohjelmistojen suosio jatkoi tasaista kasvua uudelle vuosituhannelle, kunnes TRADOS päätyi vuonna 2005 kilpailija SDL:n ostoslistalle (mt.). SDL oli jo aiemmin hankkinut omistukseensa esimerkiksi konekääntämis- ja pilvipalveluosaamista, mikä olikin vahvasti esillä myöhemmin julkaistuissa SDL Trados- ja SDL Trados Studio -ohjelmissa (SDL: SDL History n.d.).

Edellä esitetystä voikin jo päätellä, että kyseessä on monipuolinen ohjelmistokokonaisuus, joka ei rajoitu pelkästään käännösmuistien käsittelyyn ja hyödyntämiseen. Tradoksen monipuolisuus ei ole ainoastaan hyve, sillä ohjelmaa voidaan pitää käännöstyökaluksi verrattain raskaana ja jo sen satojen megatavujen kokoisen asennustiedoston lataaminen voi kestää kauan. Ohjelman koon lisäksi suurta on myös sen hinta: Trados on kääntäjille tarkoitetuista käännösmuistiohjelmista selvästi kalleimmasta päästä, eikä sen tuorein 700 euron hintalappu ole sarjan ohjelmistoissa mitenkään poikkeuksellisen suuri (SDL Trados: Spring Special Offers 2019).

Kuten edellä mainittiin, Tradoksesta on moneen muuhunkin kuin käännösmuistien kanssa työskentelyyn. Käännösmuistiohjelmaa kuvaavampi termi voisikin olla tämän vuosituhannen puolella vakiintunut käännösympäristötyökalu (englanniksi *Translation Environment Tool* tai lyhemmin *TEnT)*. Siinä missä käännösmuistiohjelman toiminnot voivat rajoittua nimensä mukaisesti käännösmuisteihin liittyviin operaatioihin, käännösympäristötyökalut sisältävät tämän lisäksi useimmat seuraavista toiminnoista ja ominaisuuksista:

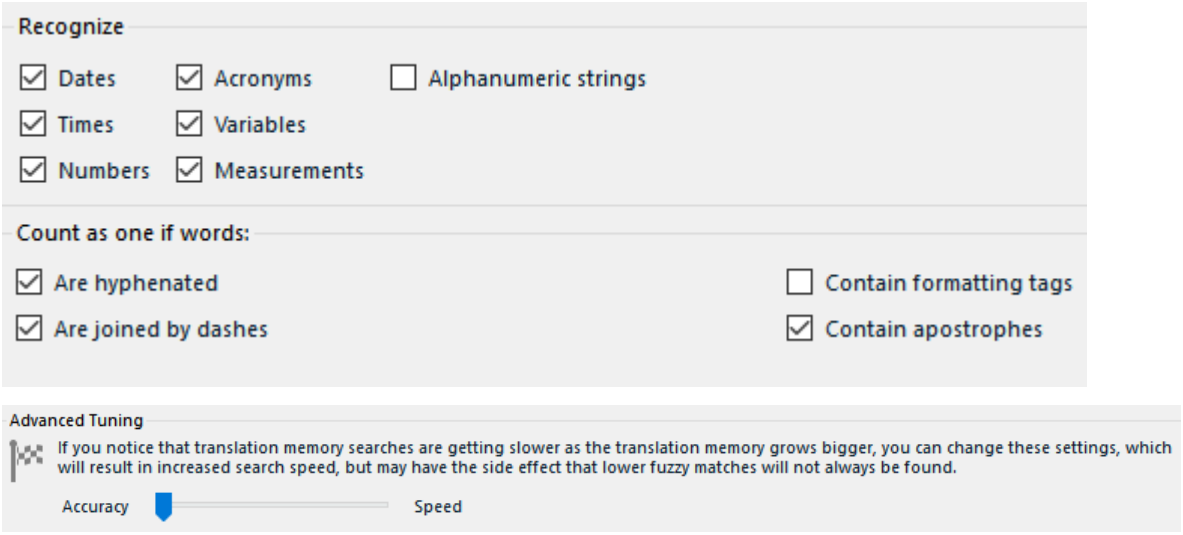
- työkalu termikantojen hallintaan
- oikolukutoiminto
- tekstien kohdistustyökalu (*alignment)*
- tuki yhdelle tai useammalle konekäännöspalvelulle
- konkordanssihaku
- tekstin analysointitoiminto.

Uusin Trados sisältää kaikki edellä luetellut ominaisuudet, joten nimitys käännösympäristötyökalu olisi perusteltu. Käytän kuitenkin tässä tutkielmassa käännösmuistiohjelmaa yläkäsitteenä kaikille tietokoneavusteisen kääntämisen ohjelmille, joiden ominaisuuksiin lukeutuu käännösmuisteista lukeminen ja niihin kirjoittaminen.

Trados tukee käännösmuisteja, termikantoja ja käännösprojekteja varten kehitettyjä avoimia TMX-, TBX- ja XLIFF-tiedostostandardeja vain rajoitetusti. Käytännössä tämä tarkoittaa sitä, että ohjelmaan voi ladata muilla käännöstyökaluilla tehtyjä termi-, muisti- ja käännöstiedostoja, mutta Trados muuntaa ne käytön yhteydessä ohjelman sen omiin tiedostomuotoihin, joiden hyödyntäminen muissa ohjelmissa vaatii tiedostojen muuntamisen esimerkiksi takaisin edellä mainittuihin tiedostomuotoihin. Tämän tutkielman kannalta mielenkiintoisin tiedostotyyppi on käännösmuisti, joka on Tradoksen tapauksessa SDLTM-muotoa. SDLTM ei ole vain uudelleen nimetty TMX-tiedosto, sillä toisin kuin TMX, joka pohjautuu XML-merkintäkieleen, SDLTM:n pohjana on SQLite-tietokantajärjestelmä (SDL Product Help: Creating Translation Memories 2015). Tietokantapohjaisen käännösmuistin tärkeimpinä etuina voidaan pitää sen nopeutta erityisesti silloin, kun käännösmuisti on hyvin suuri, sekä helppoa integroimista SQL-pohjaisiin palvelinmuisteihin (Van Assche 2014). Haittapuolena taas on yleisesti käytetystä standardista poikkeaminen, mikä osaltaan huonontaa eri ohjelmien välistä yhteensopivuutta ja pakottaa kääntäjät valitsemaan leirinsä yhä laajenevilla ja monipuolistuvilla käännöstyökalumarkkinoilla.

Trados antaa kääntäjälle jonkin verran mahdollisuuksia muistiasetusten mukauttamiseen, vaikka se ei olekaan mukautettavuudestaan erityisen tunnettu pienempiin kilpailijoihinsa

verrattuna. Alla on kaksi esitetty kaksi ohjelman muistiasetuskohtaa, joissa näkyy tutkielman kannalta olennaisimmat – sekä muuten mielenkiintoiset – muistiasetukset.




Kuva 1: käännösmuistin asetuksia Trados Studio 2017 -ohjelmassa.

Ensimmäisessä kuvassa Trados voidaan määrittää tunnistamaan tiettyjä tekstin elementtejä ja mahdollisuuksien mukaan lokalisoimaan ne kohdekieliseen muotoon (SDL Product Help 2014). Päiväyksen tunnistamisen mahdollistava *Dates*-kohta on mielestäni erityisen mielenkiintoinen johtuen sen vaikutuksesta käännösmuistiosumien tarkkuuteen. Tradoksen ohjesivun mukaan ohjelma tunnistaa lähtötekstissä esiintyvät päivämäärät ja muuntaa ne automaattisesti kohdekieleen sopivaan muotoon (mt.). Mielenkiintoista tästä tekee erityisesti se, että Trados laskee segmentit, joissa erona on ainoastaan eri päivämäärä, sataprosenttisiksi osumiksi (mt.). On sanomattakin selvää, että tämä antaa sille ainakin lievän etulyöntiaseman teksteissä, joissa esiintyy tunnistettavissa muodoissa olevia päiväyksiä.  Asetusvalikon muut kohdat toimivat vastaavalla tavalla, mutta ainakaan verkko-ohjeen mukaan niitä sisältäviä segmenttejä ei lasketa samalla tavalla sataprosenttisiksi osumiksi.

Alemmassa kuvassa näkyvä *Advanced Tuning* tarkoittaa, kuten sen kuvauksessa sanotaankin, sitä, että ohjelma voidaan määrittää priorisoimaan muistiosumien nopea löytäminen niiden tarkkuuden kustannuksella (SDL Product Help 2014). Näin ollen nopeuden suuntaan optimoidussa käännösmuistissa ne muistiosumat, joiden vastaavuusprosentti on pienempi, voivat jäädä kokonaan näkemättä. Tässä tutkielmassa

tarkoituksenani ei ole mitata muistiosumien nopeutta, vaan ainoastaan niiden tarkkuutta, joten jätän asetukseen kuvassa näkyvän tarkimman mahdolliseen valinnan.

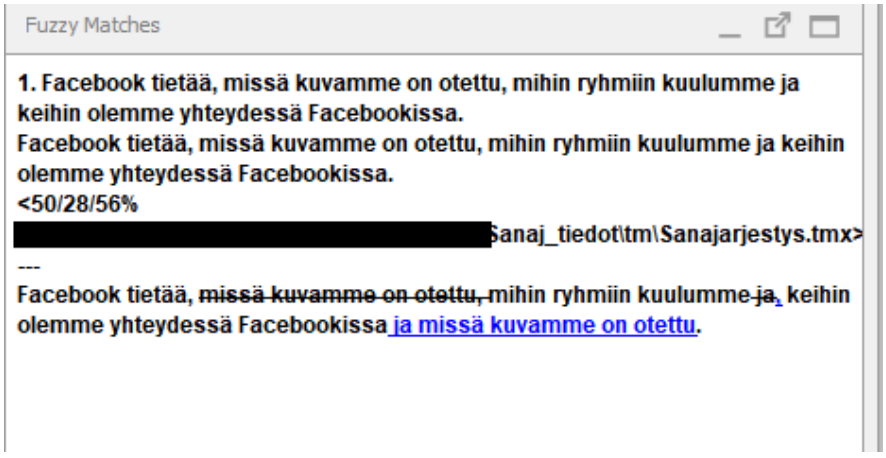
## 3.2 OmegaT

Myös OmegaT on toiminut käännösmarkkinoilla suhteellisen pitkään, vaikka sen historiaa ei voikaan verrata alan pioneereihin lukeutuvaan Tradokseen. Keith Godfreyn alun perin C++-ohjelmointikielellä kehittämän ohjelman ensimmäinen kehitysversio julkaistiin vuonna 2000, ja ohjelman ensimmäinen julkinen versio näki päivänvalon vuotta myöhemmin (Briel 2012, s. 25). Julkisen version myötä OmegaT siirtyi Java-pohjaiseksi ohjelmistoksi, jollaisena se on säilynyt tähän päivään asti (mt.). Javan suurimpia etuja on sen alustariippumattomuus, joten toisin kuin esimerkiksi Tradoksen käyttäjät, OmegaT:n käyttäjät voivat kääntää Windowsin lisäksi myös Linux- ja Mac-tietokoneilla. Koko OmegaT-projekti perustuu avoimeen lähdekoodiin, mikä on jokaisen tarkasteltavissa osoitteessa http://sourceforge.net/p/omegat/code/ci/master/tree/, ja itse ohjelman lataaminen ja käyttäminen on täysin ilmaista.

OmegaT on toimintafilosofiansa lisäksi myös teknisiltä ominaisuuksiltaan lähes täydellinen vastakohta edellä esitellylle Tradokselle – suureksi osin juuri ilmaisuutensa ja avoimuuteen tähtäävän toimintaperiaatteensa vuoksi. Ohjelmassa on monia käännösmuistiohjelmille tyypillisiä ominaisuuksia, mutta kaikkien vastaavista maksullisista ohjelmista tuttujen toimintojen käyttäminen edellyttää kolmannen osapuolen kehittämien, pääsääntöisesti ilmaisten, lisäosien asentamista. Edes nykyisten käännösohjelmien ns. ydintoiminnot, kuten virketason segmentointi tai samalle segmentille lisättävät useat käännösehdotukset, eivät sisältyneet OmegaT:hen sen ensimmäisten vuosien aikana, ja esimerkiksi oikolukuominaisuus lisättiin ohjelmaan vasta kahdeksan vuotta sen ensijulkaisun jälkeen (Briel 2012, s. 25). Toisin kuin Trados, OmegaT ei kuitenkaan kykene käyttämään Microsoftin Office-ohjelmien oikolukutoimintoa, vaan se nojaa tässäkin asiassa ennemmin ilmaiseen ja avoimeen lähdekoodiin perustuvaan Hunspell-oikolukuun. Suomalaisten kääntäjien harmiksi Hunspell ei tue suomea, joten OmegaT:tä käyttävät suomalaiset joutuvat oikolukemaan tekstinsä joko täysin ilman oikolukuohjelmiston apua tai jollain muulla ohjelmalla.

OmegaT-projektin luonteen huomioon ottaen ei ole mikään yllätys, että ohjelman toiminta rakentuu avointen tiedostomuotojen ympärille. Muistit ovat jo tuttua TMX-tiedostomuotoa ja esimerkiksi termit tallennetaan OmegaT:ssä mahdollisimman yksinkertaisesti kolmesta sarkainmerkein erotellusta sarakkeesta koostuvaan tekstitiedostoon, jonka kolmanteen sarakkeeseen voi tallentaa kutakin termiä koskevaa lisätietoa (Smolej 2018, 19.2). Kenties hieman yllättäenkin OmegaT ei hyödynnä lainkaan niin sanottuja kaksikielisiä välitiedostomuotoja, joihin sekä lähde- että kohdeteksti tallennetaan rinnakkaisessa muodossa (OmegaT: Compatibility 2018). Näiden välitiedostomuotojen sijaan OmegaT toimii yksinomaan käännösmuistitiedostojen (jotka toki ovat nekin kaksikielisiä rinnakkaisia esityksiä lähde- ja kohdetekstistä) pohjalta. Tällöin yksittäisen käännösprojektin rakenne on toki mahdollisimman yksinkertainen, mutta toisaalta taas kääntäjä voi joutua turvautumaan mahdollisiin kolmannen osapuolen ohjelmiin, mikäli asiakas haluaa käännettävästä tiedostosta myös kaksikielisen version – kuten alalla usein on tapana.

Tämän tutkielman kannalta OmegaT:n yksi mielenkiintoisimmista ominaisuuksista on sen erityinen osittaisten osumien esitystapa. Toisin kuin useimmissa käännöstyökaluissa, jotka näyttävät kullekin osumalle yhden vastaavuusprosentin, OmegaT:ssä jokainen muistiosuma esitetään kolmen erillisen prosenttiluvun kera, kuten alla olevasta kuvasta käy ilmi.




Kuva 2: OmegaT:n osittaiset osumat.

Näistä prosenttiluvuista ensimmäinen (kuvassa 50 %) kuvaa muistivastaavuuden tarkkuutta silloin, kun ohjelman Tokenizer-lisäosa on käytössä (Smolej 2018, 4.2.2). Tokenizer

määrittää siihen sisällytettyjen kielikohtaisten sääntöjen perusteella kunkin sanan sanavartalon (tätä toimenpidettä nimitetään tokenisoinniksi), ja pyrkii parantamaan käännösmuisteista saatavien osumien tarkkuutta ottamalla huomioon vertailtavissa segmenteissä esiintyvät saman sanan eri taivutusmuodot (Smolej 2018, D.1). Tokenizer sisältyy OmegaT:n uusiin versioihin oletuksena, ja se on käytettävissä myös suomesta käännettäessä. Esittelen seuraavassa alaluvussa Tokenizerin toimintaa hieman syvällisemmin, sillä kyseessä on erityisesti suomalaisille mielenkiintoinen konsepti.

Toisen prosenttiluvun (esimerkkikuvassa 28 %) tulos on saatu vertaamalla lähdesegmentin sanoja muistissa olevan segmentin sanoihin niin, että vastaavien sanojen määrä on jaettu kokonaissanamäärällä (Smolej 2018, 4.2.2). Tässä tapauksessa numeraalit ja esimerkiksi tekstikohtien muotoiluun käytetyt tunnisteet eli tagit on jätetty vertailun ulkopuolelle (mt.). Viimeinen prosenttiluku (56 %) lasketaan muutoin samoin kuin edellinenkin, mutta tässä myös numerot ja tagit sisältyvät segmenttien vertailuun (mt.). Esitellyt arvot näkyvät esimerkkikuvassa parhaimmasta huonoimpaan, sillä (jopa huonosti) tokenisoidun tekstin voi olettaa tuottavan täysin käsittelemätöntä tekstiä parempia osumia erityisesti agglutinoivista kielistä käännettäessä. OmegaT:n kolmen eri vastaavuusprosentin käyttäminen helpottaa erityisesti tokenisointityökalun testaamista, sillä saman ohjelman sisäisiä osumia verratessa eri ohjelmien algoritmeissa olevien eroavaisuuksien aiheuttamaa tulosten vääristymistä ei esiinny.

### 3.2.1 Tokenisointi OmegaT:ssä

OmegaT:ssä suomen kielen tokenisointiin käytettävä sanarunkohaku (engl. *stemming*) perustuu Jacques Savoyn (2004) kehittämään menetelmään, jonka tavoitteena oli parantaa erityisesti saksan, hollannin, ruotsin ja suomen sanarunkohakua. Alla olevalla Java-koodilla, joka on ote OmegaT:n käyttämän Tokenizer-lisäosan FinnishLightStemmer.java-tiedostosta, voidaan yrittää avata yksinkertaistetusti tokenisoinnin toimintatapaa tässä ohjelmassa.

```
if (endsWith(s, len, "ssa")
        || endsWith(s, len, "sta")
        || endsWith(s, len, "lla")
```

```
            || endsWith(s, len, "lta")
            || endsWith(s, len, "tta")
            || endsWith(s, len, "ksi")
            || endsWith(s, len, "lle"))
        return len-3;
```

Esimerkissä ohjelma tutkii siihen syötetyn sanan viimeisiä merkkejä (kuten *endsWith*-nimestä voi päätelläkin). Se vertaa merkkijonon, jossa on *len*-määrä merkkejä, kolmea viimeistä merkkiä luettelossa esitettyihin päätteisiin (ssa, sta, lla jne.). Mikäli pääte on jokin lainausmerkeissä esitetyistä, ehtolause palauttaa sanan pituuden ilman kolmea viimeistä merkkiä, jolloin alkuperäisestä sanasta saadaan taivuttamaton muoto – olettaen, ettei kyseisessä sanassa ole useampaa perättäistä suffiksia. Samaa menetelmää soveltaen kaikista sanoista saadaan, ainakin teoriassa, irrotettua sanavartalot. Kuten esimerkistä voi nähdä, kyseessä ei ole mikään uraauurtava tekoäly, joka kykenisi ymmärtämään luonnollista kieltä ja siinä esiintyvää vaihtelua. Kaikki taivutuspäätteet – ja oletettavasti myös prefiksit ja muut affiksit – pitää lisätä koodiin manuaalisesti, jolloin erilaisten poikkeustapausten kohdalla voisi olettaa olevan tunnistamisvaikeuksia. Ohjelmointitaitoni ei valitettavasti riitä koko ohjelmakoodin ymmärtämiseen, joten esimerkiksi astevaihtelun tunnistaminen täytyy testata käytännössä – olettaisin kuitenkin siinä ilmenevän vähintään lieviä ongelmia, ja todennäköisesti astevaihtelua ei sen suhteellisen harvinaisuuden vuoksi ole tokenisoinnissa huomioitu lainkaan (olen kuitenkin tässä asiassa mielelläni väärässä).

Sanavartalon tunnistamisen lisäksi Tokenizer-lisäosa sisältää niin kutsutun sulkusanaluettelon (*stop word list*). Sulkusanat ovat kielikohtaisia yleisiä sanoja, joiden merkityssisältö on niin pieni, ettei niitä huomioida esimerkiksi tiedonhaussa (Alkula 2000, s. 24). Esimerkiksi Googlen tapa jättää artikkelit ja prepositiot huomioimatta hakuja tehtäessä on hyvä esimerkki sulkusanojen hyödyntämisestä arkikäytössä. Toimintaperiaate on OmegaT:ssä identtinen Google-esimerkkiin verrattuna: sulkusanoiksi tunnistettavat sanat jätetään muistihaun ulkopuolelle, jolloin ne eivät vaikuta osumaprosentteihin lainkaan. Luettelo OmegaT:n suomen kielen sulkusanoista on nähtävissä osoitteessa http://snowball.tartarus.org/algorithms/finnish/stop.txt (tarkistettu 17.3.2019). Yksi

OmegaT:n pääkehittäjistä, Didier Briel, on jopa todennut, että sulkusanojen käyttäminen parantaa muistiosumien laatua enemmän kuin sanarunkohaun käyttäminen (2013a). Kommenttiin liittyen on kuitenkin syytä muistaa, ettei hän välttämättä tarkoita suomen kaltaisia agglutinoivia kieliä, joten mahdollinen päinvastainen tulos testiosuudessa ei mielestäni anna syytä kommentin todenmukaisuuden kyseenalaistamiseen.

Oletan, että tässä tutkielmassa tarkastelemistani käännöstyökaluista OmegaT on ainoa, jossa on hyödynnetty sanarunkohaun kaltaista tekniikkaa. On myös täysin mahdollista, että Tradoksen ja CafeTranin kehittäjät eivät vain ole halunneet paljastaa ohjelmistojensa yksityiskohtia, minkä vuoksi on mielenkiintoista selvittää, onko näiden kolmen työkalun sanavartaloiden tunnistamisessa havaittavissa enemmän eroja vai yhtäläisyyksiä.

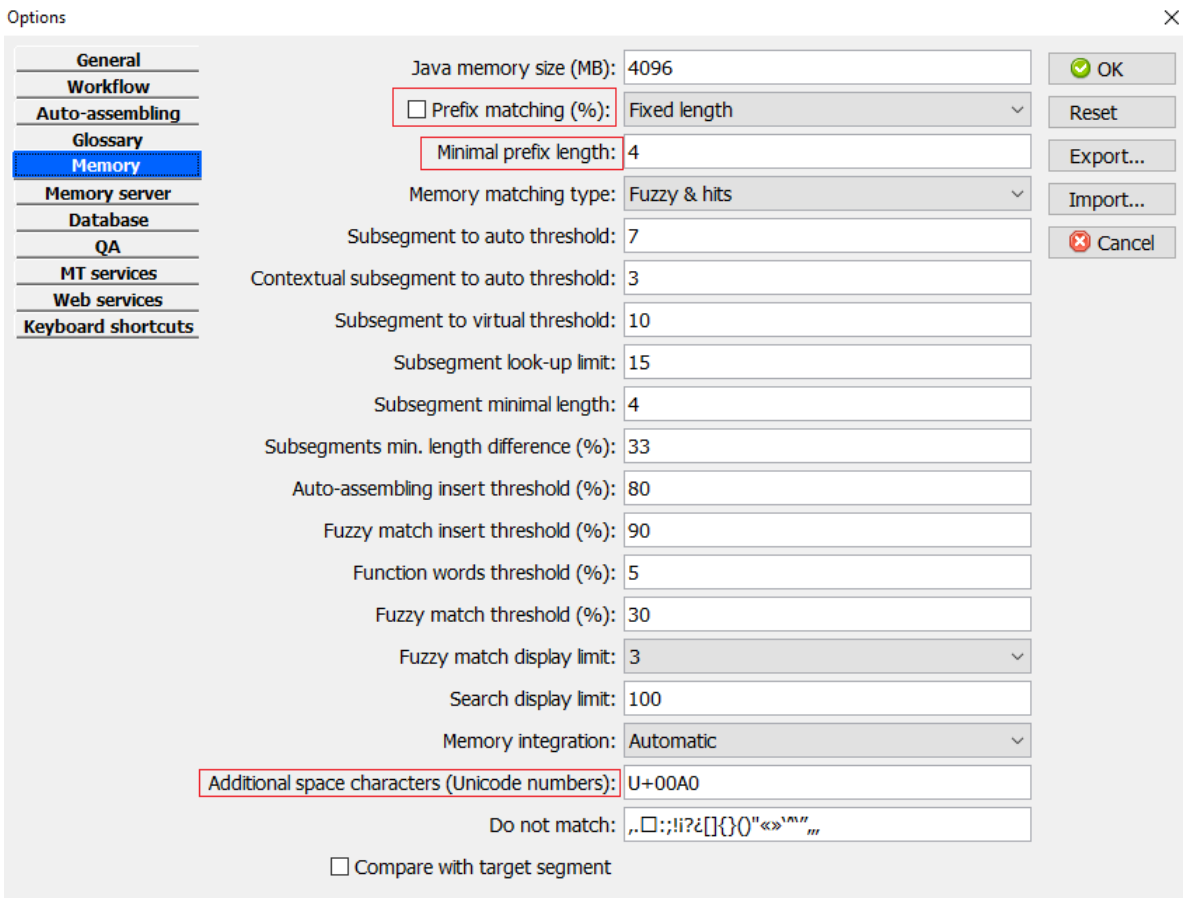
## 3.3 CafeTran

CafeTran on edellä esitellyistä ohjelmista poiketen ainoastaan yhden miehen projekti. Puolalainen Igor Kmitowski kehitti Java-pohjaisen, alun perin TexTran-nimellä julkaistun, ohjelman ensimmäisen version vuonna 2005 (Kmitowski 2012). Kmitowski kertoo kehittäneensä ohjelman avustamaan häntä itseään kääntämisessä ilman, että hänellä oli aavistustakaan siitä, että vastaavia ohjelmia, mukaan lukien Trados, oli jo markkinoilla (mt.). Kenties juuri tämän vuoksi CafeTran onkin käännösmuistiohjelmien joukossa melko poikkeuksellinen ja monin tavoin omaperäinen – eikä vähiten siksi, että useille vakiintuneille ominaisuuksille on omat, valtavirrasta poikkeavat nimensä.

Kuten todettua, CafeTran on OmegaT:n tavoin Java-pohjainen ohjelma, jota voi käyttää kaikilla tietokoneilla käyttöjärjestelmästä riippumatta. CafeTranilla on myös aktiivinen Mac-käyttäjäkunta, ja se yksi harvoista kyseisellä alustalla suoraan toimivista ei-selainpohjaisista käännösmuistiohjelmista. Toinen OmegaT:n kanssa jaettu ominaisuus on ilmaisen Hunspell-oikolukumoottorin suosiminen ilman minkäänlaista tukea esimerkiksi Microsoft Officen oikoluvulle. Suomalaisten kääntäjien onneksi CafeTran kykenee kuitenkin muodostamaan yhteyden ilmaiseen LibreOffice-toimisto-ohjelmistoon, jonka kautta käyttäjät voivat hyödyntää avoimen suomalaisen Voikko-nimisen oikolukulisäosan ominaisuuksia. Oman kokemukseni mukaan Voikko ei häviä Microsoftin oikolukutoiminnolle laadussa juurikaan – jos itse asiassa lainkaan – joten edullisuutensa

(täysversion hinta on 200 €) ja Voikon ansiosta CafeTran voisi olla aidosti varteenotettava vaihtoehto myös niille, jotka eivät ole valmiita sijoittamaan käännösohjelmiinsa monien muiden ohjelmien edellyttämiä summia (hinta tarkistettu osoitteesta https://www.cafetran.com/get-cafetran/ 16.4.2019).

Edellä mainittu LibreOffice-liitettävyys ei ole ainoa CafeTranin OmegaT:stä ja Tradoksesta erottava ominaisuus. Ohjelma sisältää moniin kilpailijoihinsa verrattuna poikkeuksellisen kattavan valikoiman erityisesti muistiosumiin liittyviä asetuksia, joiden avulla osaava käyttäjä saa ainakin periaatteessa optimoitua ohjelman usealle eri kieliparille. Keskityn tässä kuitenkin esittelemään niistä vain tämän tutkielman kannalta olennaisimmat, jotka on korostettu alla olevassa kuvassa.




Kuva 3: CafeTranin muistiasetukset.

Kuvassa ylimpänä korostettu *Prefix matching* tarkoittaa nimensä mukaisesti etuliitteen vastaavuutta. Etuliite on tässä tapauksessa hieman harhaanjohtava käsite, koska sillä ei ole

mitään tekemistä kieliopillisen käsitteen kanssa – lukuun ottamatta sitä, että se esiintyy sanan alussa. Nimen perusteella voisi olettaa, että tämä asetus auttaa karsimaan sanoista etuliitteet, mutta itse asiassa se karsii sanoista kaiken näitä ”etuliitteitä” lukuun ottamatta (Dimitriadis 2019, luku 3). Käytännössä sitä voidaan käyttää suomessa taivutuspäätteiden huomiotta jättämiseen hieman OmegaT:n tokenisoinnin tapaisesti. *Prefix matching* -valikkokohta sisältää seuraavat asetusvaihtoehdot: Fixed length, 10, 20, 30, 40, 50, 60, 70, 80 ja 90 (mt.). Näistä *Fixed length* on oletuksena valittu (mt.).

Asetusvaihtoehdoissa olevat luvut merkitsevät prosenttiosuuksia 10–90 %, jolloin esimerkiksi alimman arvon valitseminen tarkoittaa, että ohjelma analysoi sanan pituudesta ensimmäiset 10 prosenttia ja jättää loput huomioimatta (Dimitriadis 2019, luku 3). Käytännössä alimmasta asetuksesta ei ole juurikaan iloa, sillä 20 merkkiä pitkästä sanasta huomioitaisiin tällöin vain kaksi ensimmäistä merkkiä. Tämän esimerkin mukaisesti sana *lohikäärmeyhdyskunta* olisi 100-prosenttinen osuma sanalle *logaritmitaulukkojen* tai mille tahansa muulle vähintään 20-kirjaimiselle *lo*-alkuiselle sanalle. Oletan, että käytännön kääntämisessä suurin etu saadaan prosenttiarvoilla 40–70, mutta tämä selviää vasta kokeilemalla tutkielman käytännön osiossa. Oletuksena valittu *Fixed length* on enemmän tai vähemmän itsestään selvä sekin: tällöin sanasta otetaan huomioon kiinteä määrä kirjaimia sen todellisesta pituudesta riippumatta.

Seuraavassa kohdassa *Minimal prefix length* valitaan vähimmäismäärä merkkejä minkä tahansa sanan etuosassa (Dimitriadis 2019, luku 3). Ilman tätä asetusta esimerkiksi kaksikirjaimisen sanan ensimmäinen kirjain riittäisi täyden vastaavuuden säilyttämiseen, vaikka *Prefix matching* -arvoksi olisi asetettu 50 %. Mikäli *Prefix matching* -asetus on jätetty oletusarvoonsa, *Minimal prefix length* määrittää kiinteän *Fixed length* -pituuden (Dimitriadis 2019, luku 3). CafeTran sisältää erikoisuutena myös mahdollisuuden merkitä sanojen taivutuspäätteitä käännösmuistiin manuaalisesti, vaikka jokin edellä mainituista automaattisista asetuksista olisikin käytössä. Tämä tehdään käyttämällä merkkiä ”|” siinä kohdassa sanaa, josta se halutaan katkaista (CafeTran 2018). Esimerkiksi sanan *saippuakauppiaiden* taivutuspäätteet voidaan jättää tallentamatta merkintätavalla *saippuakauppia|iden*. Tällöin ohjelma tunnistaa myös muodot *saippuakauppias* ja *saippuakauppiaat* samaksi sanaksi ja osaa ehdottaa niihin soveltuvaa käännöstä. Tämän

hyödyntäminen käytännössä vaatii kuitenkin huomattavaa vaivannäköä, sillä jokainen sana olisi lisättävä muistiin yksittäin. Käytännössä toiminnosta on kuitenkin apua esimerkiksi termikantoja luotaessa, mikäli siinä on termejä vain vähän.

CafeTran sisältää lisäksi mielenkiintoisen *Additional space characters (Unicode numbers)* -asetuksen. Kohtaan voidaan lisätä Unicode-merkkejä, joita ohjelma kohtelee välilyönnin tavoin (Dimitriadis 2019, luku 1). Tämä tarkoittaa, että tietyllä merkillä erotetut sanat myös tunnistetaan erillisiksi sanoiksi, jolloin niihin löytyy todennäköisemmin osuma käännösmuistista. Sitä voidaan käyttää suomen kielessä esimerkiksi kahden yhdysmerkillä erotetun sanan (*Unicode-merkki, satama-alue* jne.) tunnistamiseen erikseen. Huomioitavaa on kuitenkin, että tässä tapauksessa CafeTran ei tunnista yhdysmerkillä erotettuja sanoja yhdeksi termiksi, mikä voi olla ei-toivottavaa monissa tilanteissa.

# 4. Suomen kielen ominaispiirteet

## 4.1 Johdanto

Suomen kieli on maailman, ja jopa Euroopan, kielten joukossa paitsi vähän puhuttu myös suhteellisen omalaatuinen – tai tällainen on ainakin vallitseva yleinen käsitys (Dahl 2008, s. 545). Mikäli näin on, on suomi jo lähtökohdiltaan huonommassa asemassa moniin muihin kieliin verrattuna käännösmuisti- ja muiden kieliteknologioiden näkökulmasta. Etulyöntiasema sen sijaan on paljon puhutuilla valtakielillä tai niitä läheisesti muistuttavilla kielillä, sillä erilaisten kaupallisten ohjelmistojen kehittäjät haluavat luonnollisesti tuotteilleen mahdollisimman laajan käyttäjäkunnan. Tällöin kehittäjien on järkevää panostaa eniten juuri kaikkein yleisimpiin kieliin – näihin lukeutuvat muun muassa englanti, espanja ja kiina, joskin kiinan kielen tuki on ainakin useimmissa länsimaissa kehitetyissä ohjelmistoissa huonohko muihin edellä mainittuihin kieliin verrattuna. Tässä luvussa käsittelen suomen erikoispiirteitä vertaamalla sitä pääasiassa englantiin ja painottamalla erityisesti niitä seikkoja, joiden oletan aiheuttavan eniten ongelmia käännöstyökaluissa.

Morfologisessa typologiassa kielet jaetaan perinteisesti neljään luokkaan: isoloiviin, agglutinoiviin, polysynteettisiin ja flekteeraaviin kieliin (Dahl 2008, s. 547). Minna Sunia lainaten isoloivat kielet voidaan määritellä kieliksi, joissa ”kieliopilliset suhteet ilmaistaan syntaksin ja funktiosanojen avulla”, kun taas agglutinoivat kielet perustuvat sanojen taivuttamiseen (2008, s.38). Polysynteettisissä kielissä sen sijaan ”yksi sana ilmaisee useampia leksikaalisia ja/tai kieliopillisia merkityksiä ja myös leksikaalinen merkitys voidaan ilmaista sidonnaisella morfeemilla” (Pajunen 2010, s. 481). Tällaisessa kielessä yksi sana voi siis vastata toisen kielen kokonaista lausetta. Flekteeraavassa kielessä erottavana tekijänä toimii sanojen vartalonsisäinen vaihtelu eikä sanan päätettä ole välttämättä mahdollista erottaa sen vartalosta (Lehečková 2012, s. 60).

Englanti on pääosin isoloiva kieli useine prepositioineen, mutta siinäkin esimerkiksi monikkoa ilmaiseva s-liite liitetään suoraan siihen liittyvän sanan perään, joten täysin isoloivasta kielestä ei voida puhua. Englannissa sanojen alkuun tai loppuun liitettäviä prefiksejä ja suffikseja, tai yleisemmin affikseja, on kuitenkin niin vähän, ettei niillä ole

tässä tutkielmassa merkitystä (riittävän pieni affiksimäärä on myös helppo lisätä ohjelman sisältämään tietokantaan, jolloin ne voidaan ottaa huomioon muistissa oleviin segmentteihin vertailtaessa). Suomi sen sijaan mielletään pääasiassa agglutinoivaksi kieleksi, mutta suomen kielen asema morfologisen typologian jatkumolla ei ole kuitenkaan aivan itsestään selvä.

Östen Dahl käsittelee artikkelissaan suomen kielen eksoottisuutta *World Atlas of Language Structures* -järjestelmästä (WALS) saatavien tietojen pohjalta (2008). Artikkelista käy ilmi, että suomea typologisesti kaikkein lähinnä olevat kielet vaikuttaisivat olevan hieman yllättäen flekteeraavia indoeurooppalaisia kieliä ja suomen tavoin uralilaisiin ja agglutinoiviin kieliin lukeutuva unkari on suomea lähinnä olevien kielten luettelossa vasta sijalla 11 (mt., s. 546). Dahl viittaa artikkelissaan tuolloin vielä julkaisemattomaan Martin Haspelmathin tutkimukseen, jonka tulosten mukaan suomi muistuttaisi edellä mainitun mukaisesti, ainakin tiettyjä parametreja käyttäen, enemmän flekteeraavia kuin agglutinoivia kieliä (mt., s. 548). Haspelmath kyseenalaistaa tutkimuksessaan myös koko morfologiseen typologiaan perustuvan kielten jakamisen tarkoituksenmukaisuuden (mt.). Ottamatta sen enempää kantaa Haspelmathin näkökantaan ja typologiaerojen luotettavuuteen, suomi ja englanti ovat kuitenkin nähdäkseni kielten jatkumolla riittävän kaukana toisistaan, jotta niiden käsittelyä käännösmuistiohjelmissa olisi mielekästä vertailla.

Käännösmuisteissa kielen morfologisella tyypillä on oletettavasti suuri vaikutus hyvien osumien löytämisessä. Toisin kuin ihminen, tietokone ei kykene vaivattomasti tunnistamaan tietyn sanan eri taivutusmuotoja (tosin kohtuullisiin tuloksiin voidaan päästä esimerkiksi myöhemmin esiteltävillä *wildcard*-ratkaisulla ja tokenisoinnilla) tai esimerkiksi yhdyssanan eri osia. Näin ollen isoloivien kielten voisi olettaa olevan etulyöntiasemassa muihin kielityyppeihin verrattaessa, sillä täydellisesti isoloivassa kielessä sanat esiintyvät tekstissä aina samassa muodossa. Tietokoneelle vaikeimpia tyyppejä ovat flekteeraavat kielet. Täysin säännölliset agglutinoivat kielet olisivat toiseksi helpoimpia ja säännölliset polysynteettiset kielet kolmanneksi, mutta kuten hyvin tiedetään, säännöllisyys ei ole montaakaan kieltä määrittävä piirre vaan erilaiset poikkeukset ovat yleisiä. Suomen kielestä puhuttaessa esimerkiksi voidaan nostaa astevaihtelu, joka vaikeuttaa taivutettujen sanojen tunnistamista saman sanan eri muodoiksi. Astevaihtelu itsessään ei ole suuri ongelma, sillä

käännösmuistiohjelman voi ohjelmoida tunnistamaan säännönmukaisen astevaihtelun. Ongelma on aiemmin mainittu epäsäännöllisyys, sillä astevaihtelu ei koske esimerkiksi kaikkia lainasanoja tai erisnimiä ja toisissa tapauksissa taas sanojen molemmat muodot kelpaavat (VISK § 44).

Seuraavissa alaluvuissa käyn tarkemmin läpi niitä pääpiirteitä, jotka tekevät suomen kielestä niin haastavan käännösmuistisovelluksille moniin muihin kieliin verrattaessa. Painotan tässäkin erityisesti käännösmuistiosumien tarkkuuteen vaikuttavia ominaisuuksia, mutta esittelen niiden ohessa myös tämän tutkielman kannalta vähemmän olennaisia mutta yleisesti suomen kieltä leimaavia piirteitä.

## 4.2 Yhdyssanat, sananmuodostus ja sanavartalot

Pitkät yhdyssanat ovat yksi ensimmäisistä asioista, joilla suomea vieraana kielenä opiskelevia tavallisesti pelotellaan, sillä sanojen yhdistäminen ei tunnetusti rajoitu vain kahteen sanaan. Yhdistämisen ohella uusia sanoja voidaan tuottaa myös johtamalla, ja yhdessä nämä kaksi muodostavatkin suomen kielen sananmuodostuksen perustan. *Suomen kieli digitaalisella aikakaudella* -raportin mukaan ”sanakirjojen hakusanoista perussanoja [sanoja, joita ei voida enää jakaa pienempiin yksiköihin] on noin 10–15 %, johdoksia noin 20–30 % ja yhdyssanoja noin 60–70 %” (2012, s. 11). Tämän perusteella on siis selvää, että käännösmuistien onnistunut hyödyntäminen suomesta toiseen kieleen käännettäessä edellyttää ohjelman selviytyvän vähintään kohtalaisesti muidenkin kuin vain perussanojen tunnistamisesta – tavalla tai toisella. Seuraavassa esittelen tarkemmin suomen sananmuodostuksen päämenetelmät mutta niitä ennen erityisesti johtamiseen läheisesti liittyvät sanavartalot.

Sanavartalo on se, mitä jää jäljelle, kun sanoista karsitaan kaikki päätteet ja tunnukset (Verkkokielioppi, 2.1.4). Esimerkiksi sanan *saappaaseen* sanavartalo saadaan poistamalla sanasta illatiivin *-seen*-pääte, jolloin sanavartaloksi siis muodostuu *saappaa*. Sanavartaloita on kahta tyyppiä: vokaali- ja konsonanttivartalot. Vokaalivartalo esiintyy jokaisella suomen sanalla, mutta lisäksi osalla sanoista on myös konsonanttivartalo, joten näitä sanoja kutsutaan kaksivartaloisiksi sanoiksi (mt.). Vokaalivartalo päättyy nimensä mukaisesti vokaaliin – kuten yllä esimerkkinä käytetty *saappaa* – ja astevaihtelun alaisen konsonantin

sisältämä vokaalivartalo voi olla sijamuodosta riippuen joko heikko tai vahva (astevaihtelu ja vahvat ja heikot asteet on esitelty seuraavassa luvussa).

Kaksivartaloisten sanojen konsonanttivartalot ovat heikkoasteisia aina, kun vartalossa esiintyy yleensä astevaihtelua (Verkkokielioppi, 2.1.4). Kaksivartaloisia nomineja ovat ne, joiden yksikön nominatiivi päättyy konsonanttiin (*allas : allas+ta*), jäännöslopukkeeseen (*kaste*$^{X}$ *: kastet+ta*) tai *i*-kirjaimeen niin, että sanavartalossa on *h*, *l*, *m*, *n*, *r*, *s* tai *t*, jota seuraa *e*-kirjain (*saari : saare+n : saar+ta*) (mt.). Vastaavasti verbeistä kaksivartaloisia ovat ne, joiden kaksi- tai kolmitavuinen vokaalivartalo päättyy *AA*:han tai *VA*:han (*tervaa+n : tervat+koon*), ne joiden kaksitavuinen vokaalivartalo päättyy *e*:hen *l*:n, *n*:n, *r*:n tai *s*:n jälkeen (*mene+n : men+köön*) ja ne, joiden vokaalivartalossa on enemmän kuin kaksi tavua ja joiden vokaalivartalo päättyy *e*:hen (*naureskele+e : naureskel+koon*) (mt.).

Kuten johdantokappaleessa mainittiinkin, sanoja voi muodostaa suomessa pääosin johtamalla ja yhdistämällä. Kumpikin näistä nojaa vahvasti kielessä jo valmiiksi esiintyviin sanoihin ja leksikaalisiin aineksiin, ja esimerkiksi uusien sanojen lainautuminen muista kielistä tai täysin uusien, muista lekseemeistä riippumattomien perussanojen syntyminen on mainittuihin menetelmiin verrattuna melko harvinaista (VISK § 146). Johtamisen ja yhdistämisen lisäksi suomessa esiintyy myös joitakin muita sananmuodostuksen keinoja, mutta ne jäävät yleisyydessä selvästi jälkeen edellä mainituista, joten esittelen ne tässä luvussa vain pikaisesti.

Johtaminen eli derivaatio on nimensä mukaisesti uusien sanojen muodostamista liittämällä jo olemassa olevaan lekseemiin yksi tai useampi johdin. Johdin on sanavartalon perään liitettävä suffiksi, jonka sijainti sanassa on suomen kielessä taivutuspäätteiden ja liitepartikkeleiden edellä (Verkkokielioppi, 2.7.2.1). Koska johtaminen on niin yleistä, myös erilaisia johtimia on luonnollisesti monia: nomininjohtimia on kaikkiaan noin 140, verbinjohtimia noin 60 ja partikkelinjohtimia noin 20 (mt.). Johtamisesta tekee erityisen joustavan sananmuodostuskeinon se, ettei kantasanan – eli sanan, johon johdos tai johdokset liitetään – tarvitse suinkaan olla perussana, vaan jo johdettu sana voi toimia perussanana uudelle johdokselle muodostaen ns. johtoketjuja (esim. *sana* > *sanasto* > *sanastollinen*) (VISK § 155).

Johdosten erottaminen perussanoista, tai edes yhdyssanoista, ei ole aina aivan itsestään selvää. Vartalon ja johtimen välinen raja voi ajan myötä hämärtyä, jolloin johdokset voivat muuttua perussanoiksi (VISK § 155). Edellä esiteltyä ilmiötä kutsutaan leksikaalistumiseksi, ja se on havaittavissa helposti esimerkiksi *opettajan* ja *nielun* (*nieleminen*) kaltaisissa sanoissa (VISK § 166). Yhdyssanojen ja johdosten välistä rajaa hämärtävät erilaiset yhdysosamaiset kielenainekset, kuten *-lainen* ja *-moinen*, jotka voidaan nähdä johtimina mutta jotka samanaikaisesti muistuttavat yhdysosia siinä, että ne liittyvät sanan genetiivimuotoon eivätkä mukaile vokaalisointua (VISK § 155).

Koska johdosten ja muiden sanojen erottaminen toisistaan voi tuottaa ongelmia ihmisillekin, on oletettavaa, ettei tietokoneen tekemä käännösmuistista hakeminen pysty erottelemaan niitä sitäkään vähää. Tällöin voi miettiä sitä, miten tärkeää on, että käännösmuisti pystyy erottamaan perussanan ja sen johdoksen erillisiksi sanoiksi – monissa tapauksissa niillä on kuitenkin niin paljon yhteistä, että jo toisen esittäminen osumana voi auttaa kääntäjää käännöstyössään. Toisaalta taas kaikkien mahdollisten johdos- ja yhdyssanojen esittäminen edes osittaisina osumina voi tuottaa tulosluetteloon niin monia tuloksia, ettei ohjelman käyttäjä pysty enää nopeasti erottelemaan hyödyllisiä täysin turhasta osumien silpusta.

Yhdistäminen on toinen suomen sananmuodostuksen pääkeinoista, ja sen tuloksena on yhdyssanoja, jotka voivat olla joko määritys- tai summamuotoisia (VISK § 398). Määritysyhdyssanat koostuvat yhdysosista, jotka voivat olla joko määrite- tai perusosia (Verkkokielioppi, 2.7.1). Määriteosa – jälleen nimensä mukaisesti – määrittää sitä seuraavaa perusosaa, kuten esimerkissä *eläinkauppa*, jossa määriteosa (*eläin-*) kertoo tarkemmin, millainen perusosa (*-kauppa*) on kyseessä. Tällöin koko yhdyssana on siis perusosansa hyponyymi. Yhdysosien rakennetta ei ole kielessä juuri rajoitettu, vaan ne voivat olla joko perussanoja (esim. *maa+pallo*), johdoksia (esim. *kirjoitus+pöytä*), muita yhdyssanoja (esim. *rautatie+liikenneverkko*) tai näiden erilaisia yhdistelmiä (mt.).

Summayhdyssanoissa (tai kopulatiivisissa yhdyssanoissa) sanan vähintään kaksi yhdysosaa ovat semanttisesti samanarvoisessa suhteessa, edustavat samaa sanaluokkaa ja jakavat

yhteisen merkityskentän (VISK § 432). Esimerkiksi yhdyssanalla *kirjailija-kääntäjä* kuvailtava henkilö edustaa sanan kumpaakin yhdysosaa samanarvoisesti: henkilö on siis sekä kirjailija että kääntäjä eikä esimerkiksi ainoastaan kirjallisuuden kääntäjä. Summayhdyssanojen väliin sijoitetaan usein yhdysmerkki selkeyttämään osien rinnasteisuutta, mutta ne voidaan kirjoittaa myös yhteen (esim. *mustavalkoinen*) (mt.). Erityisesti yhdysmerkillä kirjoitettujen summayhdyssanojen voisi kuvitella olevan jonkinlainen poikkeus oletukseen, jonka mukaan suomen kielen yhdyssanat, erityisesti moniosaiset sellaiset, tuottavat käännösmuistiohjelmille ongelmia. Tämä johtuu siitä, että useat ohjelmat voi säätää tunnistamaan yhdysmerkillä, tai oikeastaan millä tahansa merkillä, erotetut osat eri sanoiksi, jolloin jäljelle jäävä osa ei aiheuta interferenssiä tunnistetun yhdysosan osumatarkkuuden laskemisessa. Oletusarvoisena tämä asetus ei kuitenkaan ole tavallisesti käytössä.

Eräänlaisia sananmuodostuksen keinoja ovat myös lyhennesanat ja takaperoisjohto. Lyhennesanat ovat lekseemejä, joista puuttuu vähintään yksi lähtösanan osa (VISK § 167). Lyhennesanat jaetaan typiste-, kirjain- ja koostesanoihin, ja ne ovat selvästi yleisempiä puhekielisessä tekstissä kuin normien mukaisessa kirjakielessä. Typistesanat muodostetaan useimmiten niin, että sanasta poistetaan sen loppuosa, jolloin esimerkiksi *opettajasta* tulee *ope* ja *informaatiosta info* (mt.). Vaikka typistesanat ovatkin usein substantiiveja, voivat ne olla verbejä lukuun ottamatta myös muiden sanaluokkien lekseemejä (mt.). Typistesanoja, ja muitakin lyhennesanoja, esiintyy useissa kielissä (vrt. englannin *info* < *information* tai *teach* < *teacher*), joten niitä ei voida pitää suomen kielen erikoispiirteinä, vaikka esittelenkin myös ne tässä lyhyesti.

Typistesanojen luonteesta johtuen esimerkiksi yhdyssanat, joilla on sama alkuosa, voidaan typistää samaan muotoon (VISK § 167). Tämä on ainakin teoriassa ongelmallista käännösmuistiohjelmien kanssa työskenneltäessä, sillä huolimattoman kääntäjän tekstiin voi päätyä täysin väärä, mutta silti ohjelman mukaan täydellinen, osuma (esim. sana *hovi* voi olla muodostettu sekä *hovimestarista* että *hovioikeudesta*, minkä lisäksi se voi viitata myös kuninkaalliseen hoviin). Käytännössä ongelma ei kuitenkaan liene kovin yleinen, sillä käännettävät tekstit ovat useimmiten kirjakielisiä. Poikkeuksia voi esiintyä erityisesti

kaunokirjallisuuden kääntämisessä, mutta toisaalta käännösmuistien käyttökään ei ole siinä yhtä yleistä.

Kirjainsanat muodostetaan sanayhtymien sanojen ensimmäisistä kirjaimista (*TV* < *televisio, alv* < *arvonlisävero*), ja ne äännetään vakiintuneen asun mukaan (teevee, aaälvee) (VISK § 169). Kirjainsanojen haastavuus käännösteknologiassa johtuu pääasiassa niiden lyhyydestä ja siitä, että sama sana voi esiintyä samassa tekstissä myös täysimittaisessa kirjoitusasussaan, eikä käännösmuistiohjelma pysty tunnistamaan niitä saman sanan eri muodoiksi erityistapauksia lukuun ottamatta.

Koostesanat muodostetaan nekin sanayhtymien sanoista, mutta toisin kuin kirjainsanat, jotka muodostetaan ensimmäisistä kirjaimista, koostesanat voidaan muodostaa ensimmäisen kirjaimen lisäksi sanojen muistakin osista (VISK § 170). Niitä muodostettaessa tavoitteena on saada aikaan lyhennetty sana, joka on helposti äännettävissä ja taivutettavissa suomen kielen sääntöjen mukaan (mt.) Esimerkkejä koostesanoista ovat *luomu (luonnonmukainen)* ja *Kela (Kansaneläkelaitos)*.

Takaperoisjohto on tämän tutkielman kannalta hyvin mielenkiintoinen, sillä se on omiaan aiheuttamaan sekaannuksia käännösmuisteissa. Takaperoisjohdolla tarkoitetaan johdosta muistuttavan sanan karsimista niin, että suffiksaalinen aines poistetaan, jolloin tuloksena on ”lyhyempi ja morfologisesti yksinkertaisempi sana, joka edustaa yleensä eri sanaluokkaa kuin lähtösana” (VISK § 168). Tämän mukaisesti esimerkiksi verbistä *tarrata* voidaan muodostaa substantiivi *tarra* tai päinvastoin substantiivista *pakkolunastus* saadaan verbi *pakkolunastaa* (mt.). Erityisesti esimerkistä *tarrata* on helppo huomata, miten se saattaa aiheuttaa ongelmia käännettäessä. Kuten edellä on mainittu, jotkin käännösmuistiohjelmat voivat pyrkiä olemaan huomioimatta sanojen suffikseja tavoitteenaan parantaa käännösmuistin osumatarkkuutta. Tuloksena saatava sana voi olla pahimmassa tapauksessa täsmälleen sama kuin takaperoisjohtamalla saatu sana, sillä esimerkiksi CafeTran-ohjelman *Prefix matching* -asetuksen *Fixed length* -arvolla 5 *tarrata* lyhentyy sopivasti muotoon *tarra*, jolloin käännösmuistista saatava tulos olisi luonnollisesti virheellinen.

Lopputulokseltaan takaperoisjohto muistuttaa huomattavasti nollajohtoa, jossa uusi sana muodostetaan ilman uutta morfologista ainesta (VISK § 171). Nollajohto on kääntämisen kannalta vielä edellistäkin ongelmallisempi, sillä siinä lähtösana ja siitä ”johdettu” sana ovat muodoltaan identtisiä (esim. *paini, tihku, tahto*), vaikka niillä onkin eri merkitys (mt.). Nollajohdossa substantiivi mielletään verbin johdokseksi, ja se ”perustuu rinnakkaisiin, semanttisesti yhtäläisiin teonnimijohdostapauksiin”, kuten *tihku-a* < *tihku-minen* (mt.). Nollajohdon aiheuttamien mahdollisten väärien käännösehdotusten välttäminen on mahdollista ainoastaan kontekstin huomioivissa käännösmuisteissa, joissa huomioidaan myös käännettävää sanaa edeltävä tai sitä seuraava sana tai joissakin tapauksissa molemmat. Tietysti olennainen kysymys on myös se, tulisiko johdettuja sanoja edes huomioida muistiosumissa, sillä kyseessähän ei ole varsinaisesti saman sanan eri taivutusmuoto. Mielestäni niiden tunnistaminen on kuitenkin toivottavaa, sillä johdokset ovat merkitykseltään usein lähellä kantasanaansa ja tällöin näidenkin tunnistaminen voi auttaa kääntäjää työssään. Tietokonepohjaiselta ohjelmalta on – ainakin toistaiseksi – turhaa odottaa kykyä tunnistaa sanojen merkityksiä ja sitä, miten lähellä johdoksen merkitys on kantasanansa merkitystä.

## 4.3 Morfofonologinen vaihtelu

Morfofonologisessa vaihtelussa morfeemin, joko sanavartalon tai sanan suffiksin, äännesegmentti muuttuu sen ympäristön mukaan (VISK § 40). Vaihtelua esiintyy johdosten sanavartaloissa, astevaihteluna sekä taivutustunnuksia edeltävien sanavartaloiden lopussa vokaalivaihteluna (mt.). Tässä luvussa käydään lyhyesti läpi nämä kolme morfofonologisen vaihtelun mallia.

### 4.3.1 Astevaihtelu

Astevaihtelu näkyy sanavartalon muutoksina klusiileissa *p*, *t* ja *k* sekä marginaalisemmin *b* ja *g* (VISK § 41). Astevaihtelua esiintyy, kun klusiili seuraa soinnillista äännettä, ja se voi olla joko kvantitatiivista tai kvalitatiivista ja lisäksi esiintyä vahvana tai heikkona asteena (Verkkokielioppi, 2.2.3). Esimerkkejä astevaihtelusta on *katti*-sanan taipuminen muotoon *katin* tai *henki*-sanan muotoon *hengen*. Astevaihtelu on suomessa verrattain yleistä, ja tuhannesta yleisimmästä sanasta sitä esiintyy lähes joka kolmannessa (VISK § 41). Näin

ollen myös astevaihtelun vaikutuksella käännösmuistihakujen tarkkuuteen on suomesta käännettäessä epäilemättä suuri merkitys.

Astevaihtelun asteen vahvuus riippuu klusiilia seuraavasta tavusta, jolloin vahva aste edeltää useimmiten avotavua ja heikkoaste umpitavua, mutta on huomattava, että tähän sääntöön sisältyy lukuisia poikkeuksia (VISK § 43). Vahvaa astetta edustavat esimerkiksi geminaattaklusiilit (*kk*, *tt*) sekä *p* ja *t*, kun taas heikkoa edustavat vastaavat yksittäisklusiilit *k* ja *t* sekä *v* ja *d* (VISK § 41). Astevaihteluun kuuluvien klusiilien ja konsonanttiyhtymien vahvat ja heikot asteet on esitetty tarkemmin alla olevassa astevaihtelutaulukossa.

| KVANTITATIIVINEN ASTEVAIHTELU | | | |
|---|---|---|---|
| 1 | pp : p | tt : t | kk : k |
| 2 | mpp : mp | ntt : nt | ŋkk : ŋk |
| 3 | lpp : lp | ltt : lt | lkk : lk |
| 4 | rpp : rp | rtt : rt | rkk : rk |
| 5 | bb : b | | gg : g |
| KVALITATIIVINEN ASTEVAIHTELU | | | |
| 6 | mp : mm | nt : nn | ŋk : ŋŋ |
| 7 | lp : lv | lt : ll | lk : l ~ lj |
| 8 | rp : rv | rt : rr | rk : r ~ rj |
| 9 | | ht : hd | hk : h ~ hj |
| 10 | p : v | t : d | k : – (~ v) |

Kuva 4: astevaihtelu suomessa (VISK § 41).

Taulukon rivit 1–5 kuvaavat kvantitatiivista astevaihtelua, jolloin sanavartalossa siis esiintyy muutoksia geminaatta- ja yksittäisklusiilien välillä. Ensimmäinen rivi kuvaa tilannetta, jossa geminaatta seuraa vokaalia, toisessa se vastaavasti seuraa nasaalia ja kolmannen ja neljännen rivin esimerkeissä likvidoja. Viidennellä rivillä esitetty soinnillisten klusiilien astevaihtelu on harvinaista, ja sitä esiintyykin lähinnä slangisanoissa, kuten *diggaa : digata* (VISK § 43). Riveillä 6–10 kuvatut muutokset puolestaan edustavat kvalitatiivista astevaihtelua. Kvalitatiivisessa astevaihtelussa yksittäinen klusiili joko vaihtuu toiseksi konsonantiksi tai jää kokonaan pois, jolloin puhutaan kadosta.

Yksittäisklusiilia voi edeltää nasaali (rivi 6), likvida (rivit 7 ja 8), h (rivi 9) tai vokaali (rivi 10) (mt.).

Astevaihtelu voi olla lisäksi sanan perusmuodon mukaan joko suoraa tai käänteistä. Suora astevaihtelu tarkoittaa, että ”sanan perusmuoto on vahvassa asteessa ja heikkoasteisia muotoja ovat nomineilla esim. yksikön genetiivi, verbeillä esim. indikatiivin 1. ja 2. persoonan muodot” (VISK § 42). Vastaavasti taas käänteisessä astevaihtelussa sana esiintyy perusmuodossaan heikkoasteisena ja useimmissa muissa muodoissa vahvassa asteessa (mt.). Astevaihtelun suoruudella tai käänteisyydellä ei ole tämän tutkielman tuloksiin mitään vaikutusta, sillä eri asteet aiheuttavat samanlaisia eroja riippumatta siitä, esiintyykö esimerkiksi vahva aste käännösmuistissa tai käännettävässä tekstissä. Astevaihtelusta, ja erityisesti suomen kielen joustavuudesta, puhuttaessa tämäkin asia on silti mielestäni esille nostamisen arvoinen.

Aiemmin kuvattujen sääntöjen lisäksi astevaihtelussa on – kielten ominaisuuksille tyypillisesti – myös poikkeuksia. Poikkeuksia on nykysuomessa kahta tyyppiä: joko vahva aste esiintyy säännönvastaisesti myös umpitavun edellä tai heikko aste vastaavasti avotavun edellä (Verkkokielioppi, 2.2.3). Ensimmäistä poikkeustyyppiä on kolmessa tapausryhmässä: ”supistumalla syntyneen kaksoisvokaalin ja diftongin edellä”, ”vahva aste omistusliitteen edellä” (esim. *äidin,* vrt. *äitinne*) ja ”vahva aste *is*-loppuisissa konsonanttivartaloissa useimmiten” (esim. *henkistä : henkinen*, vrt. *hengen*) (mt.). Supistumalla syntyneellä kaksoisvokaalilla tai diftongilla tarkoitetaan tilannetta, jossa vokaalien välinen konsonantti ja tavuraja on hävinnyt: tällöin esimerkiksi *apu* taipuu muotoon *apuun* eikä, murrekäyttöä lukuun ottamatta, suinkaan muotoon *apuhun* (mt.).

Toinen poikkeustyyppi ilmenee sekin kolmessa tapausryhmässä. Heikko aste voi olla avotavun edessä ns. jäännöslopukkeen edellä, ”tapauksissa, joissa klusiilin sisältävä tavu on vuoroin avonainen, vuoroin umpinainen seuraavalla tavunrajalla olevan geminaattaklusiilin astevaihtelun tähden” (esim. *avuton : avuttoman,* vrt. *apu)* ja ”*i*-loppuiseen diftongiin päättyvän sivupainollisen tavun edellä” tietyissä tapausryhmissä (Verkkokielioppi, 2.2.3).

Edellä esitetystä voi huomata, että astevaihtelu vaikuttaa vain pieneen osaan sanaa. Siitä tekee kuitenkin erityisen mielenkiintoisen se, että muutos ei ilmene vasta sanan lopussa: näin ollen on odotettavaa, että astevaihtelulla olisi suurempi vaikutus käännösmuistityöskentelyyn kuin esimerkiksi taivutuspäätteillä.

### 4.3.2 Vartalonloppuinen vokaalivaihtelu

Suffiksit voivat aiheuttaa äännevaihtelua sanavartalon lopun vokaaleihin *a, e, i* ja *ä* sekä diftongeihin ja pitkiin vokaaleihin (*aa, ee* jne.) (VISK § 45). Seuraavassa käsitellään viittä vaihtelutyyppiä: *A : O ~ Ø, A : e, i : e, e : Ø* ja *VV : V*.

*A : O ~ Ø* tarkoittaa joko *a*- tai *ä*-vokaalin katoamista tai muuttumista vokaaliksi *o* tai *ö* monikkoa tai imperfektiä edustavan *i*-tunnuksen edellä (VISK § 46). Sama vaihtelu toteutuu sekä verbeissä että nomineissa, joten se on suomen kielessä erittäin yleinen (mt.). Vaihtelussa on eroja sen mukaan, onko kyseessä kaksitavuinen vartalo, monitavuinen vartalo vai vaihtelu konditionaalissa ja superlatiivissa, joten käsittelen niistä kunkin tässä erikseen.

| | VERBIT | NOMINIT |
|---|---|---|
| *a* : ∅ | sula- : sul-i | muna : mun-i-ssa |
| | muutta- : muutt-i | kova : kov-i-ssa |
| | soitta- : soitt-i | tuima : tuim-i-ssa |
| *a* : *o* | aja- : ajo-i | kana : kano-i-ssa |
| | saatta- : saatto-i | kiva : kivo-i-ssa |
| | laula- : laulo-i | reuna : reuno-i-ssa |
| *ä* : ∅ | elä- : el-i | määrä : määr-i-ssä |
| | kylvä- : kylv-i | hyvä : hyv-i-ssä |

Kuva 5: kaksitavuisten verbien ja nominien *A : O ~ Ø*-vaihtelu (VISK § 46).

Kuten yllä olevasta taulukosta käy ilmi, *a*-loppuvokaalin käyttäytyminen riippuu siitä, mikä vokaali, tai diftongi, sanavartalon ensimmäisessä tavussa esiintyy monikkoa tai imperfektiä edustavan *i*:n edellä (VISK § 46). Ensitavussa esiintyvä pyöreä vokaali *u* tai *o* johtaa loppuvokaalin katoamiseen, kun taas siinä esiintyvä lavea vokaali (*a, e, i*) puolestaan korvaa sen *o*:lla (mt.). Loppuvokaali *ä* katoaa aina monikkoa tai imperfektiä edustavan *i*:n

edellä (mt.). Ensitavun pyöreän vokaalin sisältävän diftongin vaikutus loppuvokaaliin *a* tai *ä* puolestaan riippuu diftongin ensimmäisestä osasta: jos ensimmäinen osa on *u* tai *o*, loppuvokaali katoaa (*soitta- : soitt-i*), jos se on *a, e* tai *i*, loppuvokaali korvataan vokaalilla *o* (*laula- : laulo-i, reuna : reuno-i-ssa*) (mt.).

Monitavuisten verbien vartaloiden vaihtelussa *A : O ~ Ø* imperfektin *i*-tunnus saa *A*:n katoamaan aina, ja nomineissa se joko katoaa tai muuttuu *O*:ksi sanaluokan tai vartalon viimeisen tavurajan konsonanttien mukaan (VISK § 47). Substantiivien ja adjektiivien vokaalivaihtelu esitetään seuraavassa taulukossa.

| Vartalon loppuääntеistö | Sanaluokka | *A : O* | *A : Ø* |
|---|---|---|---|
| -lA | subst.<br>adj. | kynttilä : kynttilö-i- | <br>matala : matal-i- |
| -rA | subst.<br>adj. | makkara : makkaro-i- | <br>ankara : ankar-i- |
| -nA | subst.<br>adj. | lakana : lakano-i- | (harv. omena : omen-i-)<br>ihana : ihan-i- |
| -mA | subst.<br>partis. | panoraama : panoraamo-i- | kuolema : kuolem-i-<br>tekemä : tekem-i- |
| -vA | subst.<br>adj.<br>partis. | | kanerva : kanerv-i-<br>terävä : teräv-i-<br>kouluttava : kouluttav-i- |
| -jA | subst. | | opettaja : opettaj-i-<br>remontoija : remontoij-i- |
| | | armeija : armeijo-i- | |
| -CijA | subst. | kävelijä : kävelijö-i- | |
| -(k)kA | subst. | harakka : hara(k)ko-i-<br>matematiikka : matematiikko-i-<br>karahka : karahko-i- | |
| | adj. | punakka : puna(k)ko-i- | (harv. jämäkkä : jämäkk-i-) |
| -ttA | subst. | navetta : nave(t)to-i-<br>derivaatta : derivaatto-i- | |
| -ppA | subst. | ulappa : ula(p)po-i- | |
| -tsA | subst. | kurpitsa : kurpitso-i- | |
| -eA, -OA | subst.<br>adj. | idea : ideo-i- | lipeä : lipe-i-<br>kapea : kape-i- |
| -iA, -UA | subst. | astia : astio-i- | |
| -isA | adj. | | valoisa : valois-i- |
| Komparatiivi | adj. | | suurempa- : suuremp-i- |
| Superlatiivi | adj. | | suurimpa- : suurimp-i- |

Asetelmaan eivät sisälly kaikki mahdolliset sananloput.

Kuva 6: monitavuisten substantiivien ja adjektiivien *A : O ~ Ø*-vaihtelu (VISK § 46).

Vaihtoehtoja on tässäkin runsaasti, mikä vahvistaa edelleen käsitystä siitä, että suomen sananmuodostuksen sääntöjä on käytännössä liki mahdotonta ohjelmoida luotettavasti tunnistettaviksi.

### 4.3.3 Äännevaihtelu johdoksissa

Aiemmin johtaminen mainittiin yhtenä suomen kielen sananmuodostuksen keinona. Johtamisessa sanavartalona esiintyy useimmiten jokin sen kantasanan taivutusmuodoissa esiintyvistä vartaloista, mutta ei kuitenkaan aina (VISK § 159). Toisinaan johdoksena kantavartalo on sellainen, mitä kantasanan taivutusmuodoissa ei ole, jolloin niitä kutsutaan morfofonologisesti poikkeaviksi (mt.). Seuraavassa kuvassa olevassa taulukossa on esitetty tällaiset johdoksen kantavartalon äännevaihtelutyypit.

| | |
|---|---|
| 2-TAV. *U*-VARTALOT: | kys-äise- < kys*y*-, suut-ahta- < suutt*u*-, puh-e < puh*u*- |
| 2-TAV. *O*-VARTALOT: | seis-ahta- < seis*o*-, toiv-e < toiv*o*- |
| 2-TAV. *I*-VARTALOT: | pyrk-y < pyrk*i*-, puss-ukka < puss*i* |
| 3-TAV. *A*-NOMINIT: | madal-ta- < matal*a*, hämär-tä- < hämär*ä* |
| *EA*-NOMINIT: | kork-uinen < kork*ea*, sup-ista- < supp*ea* |
| *ITSE*-VERBIT: | merki-ntä < merk*it(se)*-, häiri-inty- < häir*it(se)*-, tilk-e < tilk*it(se)*- |
| *AISE*-VERBIT: | ilma-us < ilma*is(e)*- |
| *ISE*-VERBIT: | jyli-nä ~ jyl-y < jyl*is(e)*- |
| *AA*-VERBIT: | naukk-u < naukk*aa*-, ryypp-y < ryypp*ää*-, lep-o < lep*ää*- |
| *OI*-VERBIT: | luettel-oi- < luettel*o* + -Oi- |

VARTALOTYYPISSÄÄN YKSITTÄISTAPAUKSIA: kev-ene- < kev*yt* (: kev*ye*-) | lämmi-ttä- < lämm*in* : lämp*imä*-, hapa-tta- < hap*an* : happ*ama*- | ater-in : ater-ime-t < ateri*oi*- ~ ateri*a* | jutt-u (vrt. jutt*ele*-) | kyk-y (vrt. kyk*ene*-), pak-o (vrt. pak*ene*-)

Kuva 7: taivutusasustaan poikkeavat kantavartalotyypit johdoksissa (VISK § 159).

Kuten edellä olevasta voi huomata, sanavartalon muutokset eivät ole johdoksissa välttämättä samanlaisia kuin taivutustunnusten aiheuttavat muutokset (VISK § 159). Huomattava ero on johdinta edeltävän sanavartalon pyöreän vokaalin katoaminen monissa tapauksissa (mt.). Tällöin esimerkiksi *lausu-a* muuttuu muotoon *laus-ahtaa*, jolloin pyöreä *u* häviää johtimen edeltä, toisin kuin taivutustunnuksella muodostettavassa *lausu-taan*-muodossa (mt.). Sanavartaloon voi liittyä myös yksittäistapauksissa lisäainesta, mikäli kyseessä on yksitavuinen sanavartalo: *suu**t**-ele-, vyö**hy**-ke* (mt.).

Kaikki sanan kantavartaloon tapahtuvat muutokset ovat omiaan huijaamaan käännösmuistiohjelmia, sillä niiden ennakoiminen on käytännössä mahdotonta ilman, että kaikkien suomen kielen sanojen kaikkia muotoja syötetään ohjelmaan esimerkiksi tietokantana. Se puolestaan on käytännössä mahdotonta valtavan koon vuoksi ja siksi, ettei kyseistä tietokantaa ole, tai edes voi olla, olemassakaan. Mahdottomuutta korostavat johdosten sanavartaloihin lisäainesta aiheuttavien yksittäistapausten kaltaiset poikkeukset, joita esiintyy runsaasti.

## 4.4 Sanajärjestys

Suomen sanajärjestyksestä kuulee usein sanottavan, että se on vapaa ja ettei sanojen järjestyksellä juurikaan tai jopa lainkaan merkitystä. Tämä ei tietenkään pidä paikkaansa, mutta moniin muihin kieliin verrattuna suomi on toki melko joustava, suureksi osaksi siksi, että sanojen merkityssuhteita ilmaistaan suureksi osaksi taivutuspäätteillä. Sanajärjestyksiä kutsutaan tunnusmerkillisiksi tai tunnusmerkittömiksi niiden käyttöasteen mukaan: tunnusmerkittömät lauseet ovat tavallisia, kuten *subjekti – verbi – objekti* tai *subjekti – verbi – predikatiivi*, ja tunnusmerkilliset puolestaan suppeammin käytettäviä, kuten *objekti – subjekti – verbi* (*auton minä ostin*) (VISK § 1366).

Tunnusmerkkisten lauseiden epätavallisilla sanajärjestyksillä voidaan haluta luoda kuva sen tehtävästä kontekstissa (VISK § 1366). Esimerkiksi edellisen kappaleen esimerkin *auton minä ostin* tunnusmerkkinen sanajärjestys voi selittyä sillä, että se on vastaus puhujalle esitettyyn kysymykseen (*Mitä ostit?*). Referentiaaliseen merkitykseen, eli siihen, miten lause viittaa tekstinulkoiseen maailmaan, saman lauseen eri sanajärjestysvaihtoehdot eivät vaikuta (mt.). Muut merkitykset riippuvat kontekstista.

Vapaa sanajärjestys määritellään siten, että rakenteellisen seikat eivät rajoita sitä (VISK § 1367). Rakenteellisilla seikoilla voidaan viitata esimerkiksi tehtävään lauseenjäsenenä, sanaluokkaan tai lauseketyyppiin (mt.). Vapaassa sanajärjestyksessä, toisin kuin kiinteässä, sanan tai lausekkeen paikka ei ole välttämätön ilmauksen kielenmukaisuuden tai referentiaalisen merkityksen kannalta (mt.). Näiden välimaastoon sijoittuvat lauseet, joissa järjestys on kieliopillisessa mielessä vapaa, mutta joiden sanajärjestys vaikuttaa kuitenkin sen informaatiorakenteeseen tai diskurssin tehtävään (mt.).

”Lauseen alku on järjestykseltään osittain kiinteä, koska eräät elementit kertovat lauseen syntaktisesta ja semanttisesta tyypistä” (VISK § 1368). Näitä elementtejä ovat esimerkiksi useimmat konjunktiot sekä kaikki relatiivipronominit, jotka ilmaisevat suhteen toiseen lausekkeeseen: esimerkiksi kysymyslauseen määrittävä interrogatiivipronomini (*kuka, mikä, kumpi*) ja aines, johon kysymyspartikkeli *-kO* sisältyy, ovat aina lauseen alussa (mt.). Vaikka nämä ovatkin lauseen alussa, ne eivät välttämättä ole lauseen ensimmäisiä. Vuoronalkuiset partikkelit, kuten dialogipartikkelit, huomionkohdistimet ja lausumapartikkelit, joiden tehtävänä on ilmaista vuoron asemaa keskustelussa, voivat aina edeltää lauseenalkuisia aineksia (VISK § 1027). Relatiivipronominien kanssa vastaavassa asemassa ovat liitepartikkelit, jotka kuuluvat aina lauseen ensimmäiseen jäseneen – tosin mahdollisesti vasta konjunktion perään (VISK § 1368).

Myös verbien järjestys lauseessa on suhteellisen kiinteä, kun predikaatti sisältää useamman kuin yhden verbin (VISK § 1368). Vähintään kahdesta verbistä koostuvia predikaatteja ovat liittomuodot, verbiliitot, verbiketjut, koloratiivirakenteet ja *ja*-kiteymät (VISK § 450). Näistä liittomuodoissa, verbiliitoissa ja verbiketjuissa, ei-finiittisessä muodossa olevaa pääverbiä täydentää finiittinen apuverbi, joka riippuu predikaatin tyypistä (mt.). Liittomuodoissa tämä apuverbi on joko *olla* tai *ei* (*olen tehnyt*, *en ole nähnyt*), ja se ”ilmaisee yhdessä pääverbin kanssa kiellon, tempuksen tai molemmat” (mt.). Liittomuodon apuverbi kongruoi subjektin luvun ja persoonan kanssa yksinkertaisen persoonamuotoisen verbin tavoin, kun sitä vastoin pääverbi kongruoi ainoastaan luvun mukaan (mt.). Liittomuodot ovat useimmiten finiittisiä, joskin myönteisten liittomuotojen muodostaminen on mahdollista (mt.).

Verbiliitto muistuttaa liittomuotoja siinä, että myös sen apuverbi on useimmiten *olla*, mutta myös *tulla*-verbin tai muun verbin käyttö on mahdollista (VISK § 450). Verbiliitot voivat esiintyä liittomuodoissa (*ei ollut tehtävissä*) tai infiniittisissä rakenteissa (*ei halua olla näkevinään*) (mt.). ”Verbiketjussa modaaliseen tai muuhun abstraktiin verbiin liittyy infiniittinen verbi, jonka ei katsota muodostavan verbin täydennyksenä toimivaa lauseketta. Verbiketjuja ovat esim. *alkoi itkeä, saattavat lähteä, sattuu olemaan*” (mt.).

Koloratiivirakenne koostuu deskriptiivistä pääverbistä ja sitä edeltävästä A-infinitiivin perusmuodossa olevasta verbistä, joka kuvaa samaa toimintaa neutraalilla tavalla: *juosta vilistää, nauraa hohottaa* (VISK § 450). Toisin kuin kolmessa edellä mainitussa rakenteessa, koloratiivirakenteessa verbien välissä ei voi esiintyä mitään muita lauseenjäseniä (mt.).

## 4.5 Taivutus

Taivuttaminen on sanajärjestyksen ohella yksi helpoiten huomattavista suomen kieltä leimaavista piirteistä. Nämä kaksi liittyvätkin toisiinsa varsin läheisesti, sillä suomen jokseenkin vapaa sanajärjestys vaatii toimiakseen kattavan taivutusjärjestelmän. Taivutus eli fleksio tarkoittaa lyhyesti sanottuna taivutustunnusten liittämistä nominien ja verbien sanavartaloihin. Tässä luvussa nominien taivuttaminen eli deklinaatio ja verbien taivuttaminen eli konjugaatio käsitellään erillisinä kokonaisuuksina.

### 4.5.1 Nominien taivutus

Nomineja, joihin lukeutuvat substantiivit, adjektiivit, numeraalit ja pronominit, voidaan taivuttaa kahdessa muodossa: luvussa ja sijassa (VISK § 78). Lisäksi omistusliitteillä eli possessiivisuffikseilla voidaan ilmaista (lähinnä substantiiveissa) persoonaa (Verkkokielioppi, 2.5.1.3). Yhdessä nominin taivutusmuodossa voi esiintyä kaikkia kolmea taivutuspäätettä: tällöin sanavartaloa lähinnä on nominin luvun taivutustunnus, sitten sijapääte ja viimeisimpänä possessiivisuffiksi (esim. *leip-i-ä-ni*) (VISK § 78). Tämän jälkeen sanaan voi liittyä vielä liitepartikkeli, mutta se ei ole enää taivutuspääte (mt.).

Nominin luvun taivutusta helpottaa käännösmuistisovellusten osalta se, että nominin yksikkö on aina tunnukseton (VISK § 79). Monikon tunnuksia ovat *t, i* ja *j*, joista *t* toimii samalla myös nominatiivin sijatunnuksena (mt.). Muiden kuin nominatiivimuotoisten monikkojen tunnuksissa *i* ja *j* esiintyy vokaalivaihtelua, jolloin sanan pitkä loppuvokaali voi korvautua lyhyellä vokaalilla (esim. *maa : ma-i-*) tai lyhyt vokaali korvautua toisella vokaalilla (*kuppila : kuppilo-i-*) tai jäädä kokonaan pois (*asema : asem-i-*) (VISK § 80). Kuten arvata saattaa, *i*- ja *j*-tunnuksellisten nominien variaatio on omiaan aiheuttamaan ongelmia sanojen tunnistamisessa, joten myös nominien luvun taivutuksen voidaan olettaa olevan käännösohjelmille ongelmallista yksiköiden tunnuksettomuudesta huolimatta.

Kuten aiemmin mainittiin, nominit taipuvat luvun lisäksi myös sijassa. Nominien sijoja on 15, ja ne on esitetty seuraavassa olevassa taulukossa.

| Sija | Yksikkö | Pääte | Monikko | Pääte |
|---|---|---|---|---|
| Nominatiivi | tuttu, tuote | – | tutu-t, venee-t | *-t* |
| Genetiivi | tutu-n, tuottee-n | *-n* | tuttu-j-en, poik-i-en, paperi-en ~ papere-i-den, tuotte-i-den ~ tuotte-i-tten, nais-ten, vanho-j-en ~ vanha-in | *-en, -den ~ -tten, -ten, -in* |
| Partitiivi | tuttu-a, maa-ta, tuote=t=ta, toin-ta ~ toime-a, tärkeä-ä ~ tärkeä-tä | *-A, -(t)tA* | tuttu-j-a, poik-i-a, tuotte-i-ta, palvelu-i-ta ~ palvelu-j-a, fyysiko-i-ta ~ fyysikko-j-a | *-A, -tA* |
| Akkusatiivi | minu-t | *-t* | meidä-t | *-t* |
| Essiivi | tuttu-na, tuottee-na | *-nA* | tuttu-i-na, tuotte-i-na | *-nA* |
| Translatiivi | tutu-ksi, tuottee-ksi | *-ksi* | tutu-i-ksi, tuotte-i-ksi | *-ksi* |
| Inessiivi | tutu-ssa, tuottee-ssa | *-ssA* | tutu-i-ssa, tuotte-i-ssa | *-ssA* |
| Elatiivi | tutu-sta, tuottee-sta | *-stA* | tutu-i-sta, tuotte-i-sta | *-stA* |
| Illatiivi | tuttu-un, tuottee-seen, maa-han, essee-seen ~ essee-hen | *-Vn, -hVn, -seen* | tuttu-i-hin, tuotte-i-siin, poik-i-in, korke-i-siin ~ korke-i-hin | *-hin, -siin, -in* |
| Adessiivi | tutu-lla, tuottee-lla | *-llA* | tutu-i-lla, tuotte-i-lla | *-llA* |
| Ablatiivi | tutu-lta, tuottee-lta | *-ltA* | tutu-i-lta, tuotte-i-lta | *-ltA* |
| Allatiivi | tutu-lle, tuottee-lle | *-lle*$^{x}$ | tutu-i-lle, tuotte-i-lle | *-lle*$^{x}$ |
| Abessiivi | tutu-tta, tuottee-tta | *-ttA* | tutu-i-tta, tuotte-i-tta | *-ttA* |
| Komitatiivi | | | tuttu-i-ne, tuotte-i-ne- + POS | *-ine*$^{(x)}$ |
| Instruktiivi | | | tutu-i-n, tuotte-i-n | *-in* |

Kuva 8: nominien sijamuodot (VISK § 81)

Taivutusmuotojen määrä on suuri useimpiin muihin kieliin verrattuna, minkä lisäksi vokaaliharmonia lisää entisestään mahdollisten taivutusmuotojen lukumäärää (Verkkokielioppi, 2.5.1.2). Kuten taulukosta voi huomata, yksikössä ja monikossa on usein identtiset päätteet – poikkeuksia ovat genetiivi ja illatiivi, joiden päätteissä on eroja.

Sijat voidaan jakaa funktionsa nojalla kolmeen ryhmään: kieliopillisiin, konkreettisiin ja muihin sijoihin (Verkkokielioppi, 2.5.1.2). Kieliopilliset sijat, joihin kuuluvat genetiivi, partitiivi, akkusatiivi ja nominatiivi, ilmaisevat nimensä mukaisesti pääasiassa kieliopillisia suhteita (Verkkokielioppi, 2.5.1.2.1). Mainituista sijamuodoista mielenkiintoisin – ainakin käännösteknologian kannalta – on akkusatiivi, joka esiintyy tavallisesti genetiivin tai nominatiivin näköisenä, eikä näin ollen itse asiassa monimutkaista suomen sanojen tunnistamista lainkaan (mt.).

Konkreettisiin sijoihin luetaan inessiivi, elatiivi, illatiivi, adessiivi, ablatiivi ja allatiivi (Verkkokielioppi, 2.5.1.2.2). Näitä kutsutaan yleisesti myös paikallissijoiksi: nimitys johtuu siitä, että niitä käytetään suunnan ja paikan ilmaisemiseen, mutta myös muita käyttötarkoituksia on (kuten adessiivin käyttö ajanilmauksessa *talvella*) (mt.). Paikallissijat jakautuvat ulko- (adessiivi, ablatiivi, allatiivi) ja sisäpaikallissijoihin (inessiivi, elatiivi, illatiivi) sekä lisäksi olo- (inessiivi, adessiivi), ero- (elatiivi, ablatiivi) ja tulosijoihin (illatiivi, allatiivi) (mt.).

### 4.5.2 Verbien taivutus

Verbien taivutus, konjugaatio, on jokseenkin yksinkertaisempaa kuin nominien, mutta silti kaukana yksinkertaisesta – ainakin tietokoneiden näkökulmasta. Verbien taivutustyyppeihin lukeutuvat finiittiset persoonamuodot, infiniittiset nominaalimuodot, tempus- ja modustaivutusmuodot, myöntö- ja kieltomuodot sekä aktiivi- ja passiivitaivutusmuodot (VISK § 71). Kaikilla verbeillä, tai laajemmin kaikilla suomen kielen sanoilla, on vokaalivartalo, eli sanan vartalo päättyy vokaaliin (Verkkosuomi, 2.1.4). Tämän lisäksi osalla verbeistä on myös konsonanttivartalo, jolloin verbiä kutsutaan kaksivartaloiseksi (VISK § 71).

Persoonat ovat suomen kielessä puhujaan ja tämän viiteryhmään viittaava 1. persoona, kuulijaan ja tämän viiteryhmään viittaava 2. persoona ja kaikkiin muihin viittaava 3. persoona, minkä lisäksi olemassa on myös yksipersoonainen passiivi (VISK § 106). Verbien persoonamuotoja käytetään kuitenkin myös muilla tavoin, sillä esimerkiksi niin sanottua nollapersoonaa, joka on yksikön 3. persoonan erikoistapaus, voidaan käyttää yleisesti tai viittaamaan puhujaan ja toisaalta yksikön 2. persoonasta on olemassa puhekielinen ”sinä-passiivi” (*sen jälkeen sä käännyt vasemmalle*) (mt.).

Verbien finiittimuodot ovat luultavasti helpoiten verbeiksi tunnistettavia sanoja niiden tekijää ilmaisevien persoonapäätteiden vuoksi (Verkkokielioppi, 2.5.2.1). Finiittiverbi sisältää näin ollen itsessään tiedon subjektin persoonasta, jolloin subjektia ei ole tarpeen erikseen ilmoittaa – kuten esimerkeissä *(minä) näen sinut* ja *(sinä) tulet huomenna esiintymään* – mikä toisaalta helpottaa ja nopeuttaa tekstin tuottamista ja lukemista, mutta

toisaalta taas vaikeuttaa sanojen, ja tässä erityisesti subjektien, tunnistamista tietokoneavusteisesti (VISK § 107). Persoonapääte liittyy finiittiverbien myöntömuodoissa suoraan verbin vartaloon tai tempuksen tai moduksen tunnukseen ja kielteisissä muodoissa kieltoverbiin, kuten seuraavasta kuvasta käy ilmi (mt.).

| PERSOONA | INDIKATIIVI, KONDITIONAALI, POTENTIAALI | PÄÄTE | IMPERATIIVI | PÄÄTE |
|---|---|---|---|---|
| YKS. 1. | kerro-n, kertoisi-n, kertone-n | *-n* | – | |
| YKS. 2. | hyppää-t, hyppäisi-t, hypänne-t | *-t* | kerro$^{x}$, hyppää$^{x}$ | – |
| YKS. 3. | kerto-o, syö, söisi, syöne-e | *-(V)* | kertoko-on, hypätkö-ön | *-On* |
| MON. 1. | kerro-mme, söisi-mme, antane-mme | *-mme* | kertokaa-mme | *-mme* |
| MON. 2. | kerro-tte, söisi-tte, antane-tte | *-tte* | kertokaa, hypätkää | – |
| MON. 3. | kerto-vat, söisi-vät, hypänne-vät | *-vAt* | kertoko-ot | *-Ot* |
| PASSIIVI | kerrota-an, syötäisi-in, annettane-en | *-Vn* | kerrottako-on | *-On* |

Kuva 9: verbien persoonapäätteet moduksissa (VISK § 107)

Tempuksella ja moduksella kuvataan verbin tekemisen tapaa tai aikaa. Moduksella, joita ovat suomessa imperatiivi, potentiaali, konditionaali ja indikatiivi, tilanne suhteutetaan vallitsevaan todellisuuteen, eli jonkin tilanteen voidaan mainita olevan esimerkiksi toivottava tai todennäköinen (VISK § 111). Tempuksella kohteena olevan tilanteen ajankohta voidaan suhteuttaa kyseisen hetken ajankohtaan (mt.). Suomessa morfologisia tempuksia on kaksi, imperfekti ja preesens, minkä lisäksi ns. liittotempusten, perfektin ja pluskvamperfektin, muodostaminen on mahdollista apuverbeillä (mt.).

Verbien infiniittisten nominaalimuotojen, joita ovat infinitiivit ja partisiipit, taivutukset muistuttavat jossain määrin nomineja, sillä niissä ei ole modusta tai tempusta, minkä lisäksi persoona voidaan joissakin tilanteissa ilmaista persoonapäätteen sijaan possessiivisuffiksilla (VISK § 119). Partisiipit muistuttavat näistä kahdesta enemmän nomineja, sillä niillä on lisäksi sija- ja lukutaivutus (mt.). Infinitiiveillä sen sijaan ei ole monikkoa lainkaan, ja myös niiden sijataivutus on epätäydellinen (mt.). Infiniittiset muodot muistuttavat nomineja myös siinä, mitä tehtäviä niillä voi lauseissa olla: ne voivat toimia subjekteina, objekteina, attribuutteina ja adverbiaaleina (Verkkokielioppi, 2.5.2.2).

Kuten edellä mainittiin, infinitiivit toimivat lauseessa nominien – tarkemmin sanottuna substantiivien – tapaan (Verkkokielioppi, 2.5.2.2.1). Ne eroavat kuitenkin substantiiveista

siinä, ettei niissä käytetä kuin muutamia sijoja eikä lukua ilmaista lainkaan (mt.). Infinitiivityyppejä on kolme: A-, E- ja MA-infinitiivit (VISK § 120). Näistä kahdessa ensimmäisessä esiintyy seuraavassa esitettyä morfologista vaihtelua verbin vartalon mukaan.

| Vartalon loppu | A-infinitiivi | Tunnus | E-infinitiivi | Tunnus |
|---|---|---|---|---|
| 1. *V* | kerto-a, etsi-ä, luke-a, rakasta-a | *A* | kerto-e-, etsi-e-, luki-e-, rakasta-e- | *e* |
| 2. *VV* | saa-da, tupakoi-da | *dA* | saa-de-, tupakoi-de- | *de* |
| 3. *s, l, r, n* | nous-ta, tul-la, luistel-la, sur-ra, men-nä | *tA, lA, rA, nA* | nous-te-, tul-le-, luiste-le-, sur-re-, men-ne- | *te, le, re, ne* |
| 4. *=t=* | hypä=t=ä, häiri=t=ä, vanhe=t=a | *(t)A* | hypä=t=e-, häiri=t=e-, vanhe=t=e- | *(t)e* |

Kuva 10: A- ja E-infinitiivien taivutukset (VISK § 120)

Tunnusten sijainti verbissä riippuu sen vartalosta: yksivartaloisissa tunnukset liittyvät vokaalivartaloon, kun taas kaksivartaloisissa ne liittyvät konsonanttivartaloon (VISK § 120). Mikäli vartalo päättyy lyhyeen vokaaliin, tunnuksena on *A* tai *e*, ja mikäli vartalon lopussa on pitkä vokaali, tunnuksena on vastaavasti joko *dA* tai *de* (mt.). Mikäli yksivartaloisessa verbissä on kuitenkin *e*-loppuinen vartalo, tunnusta edeltää *i* eikä *e* (*luki-e-, kylpi-e-*) (mt.). Konsonanttivartaloon liittyessään ”tunnuksen konsonantti on vartalon s:n jäljessä t-aineksinen, muuten se assimiloituu vartalon lopun konsonantin kanssa ja on l-, r- tai n-aineksinen”, kuten edellä olevan kuvan kohdasta 3 näkyy (mt.). Kohdan 4 verbeillä vartalon ja tunnuksen välinen raja on epäselvä (mt.).

MA-infinitiivi esiintyy vain yksivartaloisten verbien yhteydessä, ja sillä on kuusi mahdollista sijamuotoa: inessiivi, elatiivi, illatiivi, adessiivi, abessiivi ja instruktiivi (VISK § 121). Näistä instruktiivi on tosin erittäin harvinainen, sillä sitä esiintyy vain *pitää*-verbin yhteydessä *Mitä pitikään tekemän* -kaltaisissa ilmauksissa (mt.). MA-infinitiivin muodot on esitetty seuraavassa kuvassa olevassa taulukossa.

| VARTALO | TUNNUS | SIJA |
|---|---|---|
| kerto-, etsi-, luke-, rakasta-<br>saa-, tupakoi-<br>nouse-, tule-, sure-, mene-<br>hyppää-, tarjoa-, häiritse-, pimene- | *mA* | INE kerto-ma-ssa, ELA ui-ma-sta,<br>ILL sure-ma-an, ADE hyppää-mä-llä,<br>ABE häiritse-mä-ttä, INS sano-ma-n |

Kuva 11: MA-infinitiivin taivutukset (VISK § 121)

Partisiipit ovat infinitiivien tavoin verbien infiniittisiä nominaalimuotoja. Siinä missä infinitiivit toimivat substantiivien tavoin, partisiippeja käytetään adjektiivien tavoin substantiivien etumääritteinä, substantiivilausekkeen edussanana tai itsenäisesti predikatiivina sekä joissakin tapauksissa myös liittomuotojen osissa tai muissa verbiliitoissa (VISK § 122). Partisiippityyppejä on neljä: NUT-partisiipit, VA-partisiipit, agenttipartisiipit ja kieltopartisiipit (mt.).

| PARTISIIPPI | TUNNUS | PASSIIVIMUODOSSA |
|---|---|---|
| NUT-partisiippi | *nUt, lUt, rUt, sUt, (nnUt)* | *(t)tU*$^{(x)}$ |
| VA-partisiippi | *vA* | *(t)tAvA* |
| Agenttipartisiippi | *mA* | |
| Kieltopartisiippi | *mAtOn* | |

Kuva 12: partisiipit (VISK § 122)

NUT-partisiipin tunnus liittyy aktiivimuodossa yksivartaloisissa verbeissä niiden vahvaan, eli astevaihtelullisen konsonantin sisältävään, vokaalivartaloon ja kaksivartaloisissa konsonanttivartaloon (VISK § 122). Taivutuksen osalta ne kuuluvat *ee*:n vokaalivartaloissa sisältävien nominien ryhmään (kuten *lukenut : lukenee*-) (mt.). Passiivimuodossa ilmenevä TU liittyy yksivartalollisissa verbeissä heikkoon vokaalivartaloon ja kaksivartaloisissa konsonanttivartaloon muiden passiivimuotojen tavoin (mt.). VA-partisiipin aktiivin ja passiivin tunnukset sekä agenttipartisiipin ja kieltopartisiipin tunnukset puolestaan liittyvät vahvaan vokaalivartaloon (mt.).

Verbeillä on suomen kielessä yhteensä 264 paradigmamuotoa eli taivutusmuotoa (VISK § 124). Näistä finiittisten osuus on 34, kieltomuotojen osuus 10, infinitiivien osuus 28, partisiippien osuus 156 ja muiden infiniittisten osuus 36, joten partisiippien selkeästi suuremmasta lukumäärästä huolimatta niitä kaikkia voidaan olettaa esiintyvän kielessä

melko tasapainoisesti (mt.). Tämän perusteella on selvää, että jo yhden verbin kaikkien paradigmamuotojen ”koodaaminen” ohjelmistoon vaatii huomattavaa vaivannäköä, joten muiden käsiteltyjen aiheiden mukaisesti myös kaikkien suomen verbien lisäämistä voitaneen pitää mahdottomana tehtävänä – varsinkin, kun uusia sanoja tulee kieleen jatkuvasti.

Tilanne ei ole kuitenkaan täysin toivoton: suomi voisi nimittäin olla huomattavasti hankalampaakin. Ymmärtämistä helpottavana seikkana voidaan pitää esimerkiksi sitä, että suomessa taivutusaines sijoittuu aina sanan tai sanavartalon loppuun, minkä lisäksi ne ovat aina samassa järjestyksessä (Pajunen, s. 566). Näin ollen suomen kielen sanat on periaatteessa mahdollista jakaa osiin, joilla on kullakin oma merkityksensä (Pajunen, s. 567). Lisäksi suomi ei synteettisenä kielenä ole kovinkaan kompleksinen tai muutenkaan ”kovin erikoinen kieli” (Pajunen, s. 569). Tämä antaa lisätoivoa seuraavassa esiteltävään tutkimusosioon.

# 5. Tutkimuksen toteuttaminen

Toteutan käännösmuistiohjelmien vertailun empiirisesti hyödyntämällä sitä varten valmistelemiani testausaineistoja. Testiaineistot koostuvat lyhyistä tekstiotteista, joita on muokattu niin, että tutkielmassa esiteltyjen kysymysten ja hypoteesien testaaminen olisi mahdollisimman suoraviivaista ja tulosten havaitseminen helppoa: käytännössä tämä tarkoittaa lyhyitä ja rakenteeltaan yksinkertaisia virkkeitä. Käytetty aineisto koostuu julkaistuista teksteistä ja niistä valmistelluista muokatuista versioista. Testiaineistona olisi voinut toimia myös yksi pitkä teksti, mutta mielestäni monen lyhyen tekstin käyttäminen helpottaa testaamisen kohdistamista haluttuihin aiheisiin ilman, että tekstin muut osat vaikuttavat liikaa tulosten tarkkuuteen. Lisäksi eri tekstien käyttäminen voi paljastaa käännösmuistiohjelmien toiminnasta jotain, joka jäisi yhden tekstin useaan kertaan muokattua versiota tarkasteltaessa havaitsematta.

Ensimmäisenä muodostan alkutekstistä ja sen käännöksestä käännösmuistin kohdistamalla, eli lisäämällä käännösmuistiin alkutekstin ja käännöksen segmentit rinnakkain virke kerrallaan (European Commission Directorate-General for Translation 2017, s. 14). Käännös voi olla sanana hieman harhaanjohtava, sillä en käytä kohdistuksessa tekstien virallisia tai edes itse kääntämiäni käännöksiä. Tämä johtuu yksinkertaisesti siitä, ettei niille ole mitään tarvetta: käännösmuistiohjelman kannalta segmenttien analysoinnin tulos on sama riippumatta siitä, mitä kohdesegmentissä lukee, sillä se vertailee ainoastaan lähtötekstin segmenttejä toisiinsa. Tässä tapauksessa tekstejä ei ole käännetty mitenkään, vaan samoja virkkeitä on käytetty sekä lähtö- että kohdekielessä. Tämä johtuu yksinkertaisesti siitä, ettei kohdekieltä voi jättää monissa ohjelmissa tyhjäksikään. Pelkästään tulosten kannalta kohdesegmentit olisi yhtä hyvin voitu jättää myös tyhjiksi tai korvata esimerkiksi lorem ipsumin kaltaisella täytetekstillä.

Käytin käännösmuistin muodostukseen memoQ 8.2 -ohjelmaa minimoidakseni riskin, että jokin tutkimukseen sisällytetyistä ohjelmista olisi saanut samalla ohjelmalla kohdistetusta käännösmuistista etua muihin vertailtaviin ohjelmiin verrattuna. Etua voisi muodostua esimerkiksi siitä, että eri ohjelmat käyttävät omia muististandardejaan – kuten Tradoksessa käytössä oleva SDLTM-tiedostomuoto – ja tallentavat tekemiinsä muisteihin erilaisia metatietoja esimerkiksi tunnisteiden muodossa, jolloin yhden ohjelman tallentamat

tunnistetiedot voivat esiintyä toisessa ohjelmassa ylimääräisinä merkkeinä ja heikentää näin ollen muistiosumien tarkkuutta. Myös memoQ käyttää käännösmuisteissa omaa tiedostomuotoaan, jota muut ohjelmat eivät kykene lukemaan. Muistitiedostot voi kuitenkin viedä alan standardina toimivaan TMX-muotoon, jota kaikki tässä käyttämäni käännösmuistiohjelmat tukevat (tosin Trados muuntaa aina avatun TMX-tiedoston omaan SDLTM-tiedostomuotoonsa ennen käyttöä).

Toisessa vaiheessa alkutekstiä muokataan sen mukaan, mitä kielen osa-aluetta halutaan tutkia: tämä voi tarkoittaa esimerkiksi lauseen sanajärjestyksen muuttamista tai sanojen taivuttamista. Tämän jälkeen vertaan muokattuja tekstejä muokkaamattomasta tekstistä muodostettuun käännösmuistiin nähdäkseni, miten paljon muutokset vaikuttavat tekstin tunnistustarkkuuteen. Muutosten on oltava melko tarkkaan rajoitettuja, jotta mahdollisten erojen johtuminen muista syistä voidaan sulkea pois. Käännösmuistiohjelmien antamat prosenttiarvot ovat suoraan vertailukelpoisia, joten tuloksia ei tarvitse tämän jälkeen juurikaan käsitellä.

Testausaineistojen avulla tutkittavat piirteet voidaan jakaa seuraaviin osioihin: taivutuspäätteet, morfofonologinen vaihtelu (joka liittyy myös edelliseen) ja sanajärjestys. Jakaminen olisi toki mahdollista monella muullakin tavalla, mutta jokaisen suomen kielen erikoispiirteen tarkastelu erikseen ei ole tämän tutkielman laajuuden kannalta mielekästä. Aloitan kunkin osion esittelemällä lyhyesti siihen sisältyvät kielelliset piirteet ja osion piirteiden tutkimiseen käytettävän testausmateriaalin. Tämän jälkeen tarkastelen testausmateriaalin käännösmuistiosumia erikseen kullakin käännöstyökalulla (Trados, CafeTran, OmegaT) ja vertailen tuloksia ennen seuraavaan osioon siirtymistä.

Käytännössä tarkastelu toteutetaan niin, että luon kulloinkin testattavalla käännösohjelmalla ensin uuden käännösprojektin, johon lisään alkuperäisestä, muokkaamattomasta tekstistä luodun käännösmuistin, minkä jälkeen aloitan muokatun tekstin kääntämisen alkuperäisen käännösmuistin pohjalta. Mikäli johonkin ohjelmaan sisältyy mielenkiintoisia ominaisuuksia, joiden voidaan olettaa vaikuttavan tutkimustuloksiin – kuten OmegaT:n tokenisointi tai CafeTranin muistiasetukset – saatan perehtyä vielä niihin pikaisesti erikseen. Niiden tarkastelu ei kuitenkaan ole tutkielman kannalta välttämätöntä.

## 5.1 Taivutuspäätteet

Taivutusmuotojen tunnistamisen tarkastelu on ainakin teoriassa yksinkertaista: riittää, että lausetta, joka sisältää ainoastaan tai ainakin enimmäkseen taivuttamattomia muotoja, verrataan samaan lauseeseen niin, että osa sanoista on taivutettu. Periaatteessa tähän tarkoitukseen riittää pelkkä sanaluettelo, joka ei muodosta järkevää kokonaisuutta. Haluan kuitenkin vertailla ohjelmien toimintaa aitoja käännöstilanteita mukailevassa kontekstissa, joten käytän tässä suomen kieliopin mukaisia virkkeitä, joita muokkaan taivutusmuotoja poistamalla ja toisaalta lisäämällä. Pyrin välttämään taivutuspäätteitä lisätessäni morfofonologista vaihtelua parhaani mukaan, sillä sen testaaminen liittyy tutkimusosion seuraavaan vaiheeseen.

Taivutusmuotojen tutkimiseen käyttämäni testiaineisto on helposti ymmärrettävällä selkokielellä kirjoitetulla *Selkosanomat*-verkkosivustolla 14.3.2018 julkaistu Maria Österlundin kirjoittama uutinen ”Laki valinnanvapaudesta eteni eduskuntaan”. Valitsin selkokielisen uutisartikkelin siksi, että siinä virkkeet ovat lyhyitä ja sisältävät tavallista vähemmän erikoissanastoa, kuten lainasanoja, joiden taivuttaminen voi olla suomenkielisille sanoille epätyypillistä. Lyhyissä virkkeissä voi olla ongelmana se, että pienetkin muutokset virkkeen sisällössä saattavat aiheuttaa suuria prosentuaalisia muutoksia tekstin samankaltaisuuden tunnistamisessa. Pahimmassa tapauksessa virke voi olla niin lyhyt, että tehtyjen muutosten jälkeen käännösmuistiohjelma ei enää tunnista sitä lainkaan samankaltaiseksi segmentiksi. Pyrin välttämään tätä muuttamalla kutakin virkettä kohtuudella – koko tekstin muutosten määrällähän ei ole tässä tapauksessa vaikutusta, sillä ohjelma tarkastelee sitä irrallisina segmentteinä. Uutisessa on 83 sanaa, mikä tekee siitä sopivan mittaisen tutkielman yhden osa-alueen tarkasteluun. Uutisteksti on kokonaisuudessaan esitetty alla.

> Hallitus jätti eduskunnalle lakiehdotuksen valinnanvapaudesta viime viikolla. Sote eli sosiaali- ja terveydenhuollon uudistus etenee. Soteen kuuluu myös lakiehdotus valinnanvapaudesta. Valinnanvapaus tarkoittaa sitä, että kansalaisilla on oikeus valita, missä he haluavat saada hoitoa. Tätä varten eduskunnan täytyy päättää yhteensä 40 uudesta laista ja lainmuutoksesta. Oppositio eli puolueet hallituksen ulkopuolella arvostelevat sotea. Opposition mielestä sote on kallis ja huono. Pääministeri Juha Sipilä uhkasi, että hallituksen täytyy erota, jos eduskunta ei hyväksy sotea. Hallituksen suunnitelma on, että suomalaiset saavat jo syksyllä 2019 valita, missä heitä hoidetaan (Österlund 2018a).

Taivutusmuotojen lisäämiseen ei ole kaavaa tai säännönmukaisuutta, mikä olisi tällaisessa vertailussa ehdottomasti eduksi. Näin ollen kävin tekstin virke virkkeeltä läpi lisäten taivutuspäätteitä sanoihin, joissa niitä ei esiinny, ja muuttaen niitä sanoissa, joissa taivutuspäätteitä jo on. Kaikkien sanojen muuttaminen ei ole mielestäni järkevää, sillä tällöin vastaavuusprosentit voivat jäädä niin pieniksi, ettei niitä ohjelmissa edes näytetä. Lopuksi merkitsin kussakin muistisegmentissä – eli käytännössä lähes aina kussakin virkkeessä – esiintyvät muutokset, mikä helpottaa myöhempää vertailua. Seuraavassa on esitetty tekstiin tekemäni muutokset, jotka on korostettu alleviivauksella. Hakasulkeet merkitsevät kieliopin vastaista muotoa tai rakennetta ja kaarisulkeet ilmaisevat, että sanasta on poistettu yksi tai useampi kirjain ilman, että sitä on korvattu muulla.

> Hallituksemme jätti eduskunnalle lakiehdotuksia valinnanvapaudesta viime viikolla. Sotea eli sosiaali- ja terveydenhuollon uudistusta [etenee]. Soteen kuuluvat myös lakiehdotukset valinnanvapaudesta. Valinnanvapaudella tarkoitetaan sitä, että kansalaisillakin on oikeus valita, missä he haluavat saada hoitoa. Tätä varten eduskunnan täytyy päättää yhteensä [40:stä] uudesta laista ja lainmuutoksesta. Oppositio eli puolueet hallituksen ulkopuolella arvostelevat sotea. Oppositionkin mielestä sote olisi kallis ja huono. Pääministerinä Juha Sipilä uhkaa, että hallituksen täytyy erota, jos eduskunta ei hyväksyisi sotea. Hallituksen suunnitelma oli, että suomalainen saa() jo syksyllä 2019 valita, missä heitä hoidetaan.

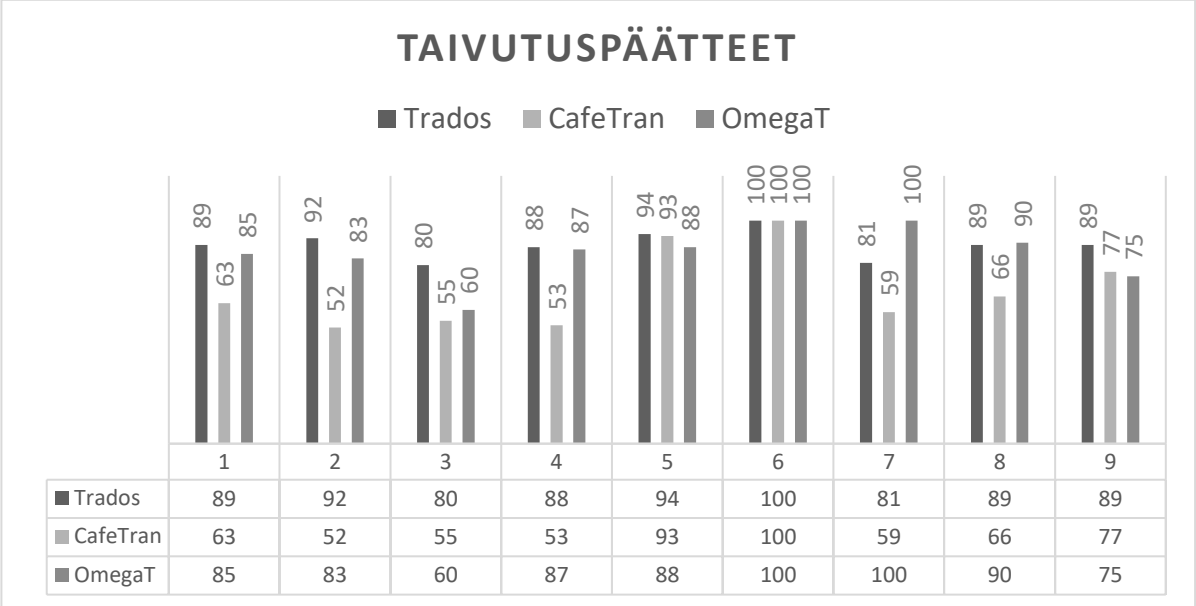
### 5.1.1 Tulokset

Käännösmuistiohjelmien tuloksissa ilmeni yllättävän paljon eroja. Niitä esiintyi jokaisen käännösmuistiohjelman välillä kaikissa segmenteissä, lukuun ottamatta segmenttiä 6, joka oli ainoa muutoksia sisältämätön segmentti (muuttamattoman segmentin avulla on mahdollista vahvistaa, ettei käytetty formaatti itsessään aiheuta eroja, vaan ne kaikki johtuvat kielellisestä aineksesta). Kuten aiemmin OmegaT:tä käsittelevässä osiossa mainitsin, ohjelma esittää jokaiselle segmentille kolme eri vastaavuusprosenttilukua. Ohjelmien vertailutuloksissa olen huomioinut niistä ainoastaan ensimmäisen, eli sen, jossa on hyödynnetty ohjelman sisältämää tokenisointitoimintoa, elleivät tulokset anna syytä tarkastella myös muita vaihtoehtoja.

| | | | |
|---|---|---|---|
| 1 | ~~Hallitus~~Hallituksemme jätti eduskunnalle ~~lakiehdotuksen~~lakiehdotuksia valinnanvapaudesta viime viikolla. | 89% | Hallitus jätti eduskunnalle lakiehdotuksen valinnanvapaudesta viime viikolla. |

Kuva 13: Esimerkki Tradoksessa esitetystä muistiosumanäkymästä.

Tradoksen tarjoamat muistiosumien tarkkuudet olivat kolmesta ohjelmasta parhaat kuudessa kahdeksasta (mikäli muokkaamaton segmentti jätetään huomiotta) virkkeestä, ja OmegaT ylsi kärkeen kahdessa segmentissä. CafeTran tarjosi kahta muuta selvästi heikompia osumia yli puolessa segmenteistä eikä yleisesti vakuuttanut tunnistustarkkuudellaan. Muita heikompi suoritustaso voi selittyä sillä, että sen Prefix matching (%) -asetus ei ollut ohjelmassa oletusarvoisesti käytössä. Tämän vuoksi toistan testin CafeTranin osalta vielä uudelleen pyrkien hyödyntämään asetuksen oletettua etulyöntiasemaa erityisesti suomen kaltaisia kieliä käännettäessä.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| Trados | 89 | 92 | 80 | 88 | 94 | 100 | 81 | 89 | 89 |
| CafeTran | 63 | 52 | 55 | 53 | 93 | 100 | 59 | 66 | 77 |
| OmegaT | 85 | 83 | 60 | 87 | 88 | 100 | 100 | 90 | 75 |

Kuva 14: Taivutuspäätetestin tulokset. Vaaka-akselilla esitetään segmentin numero.

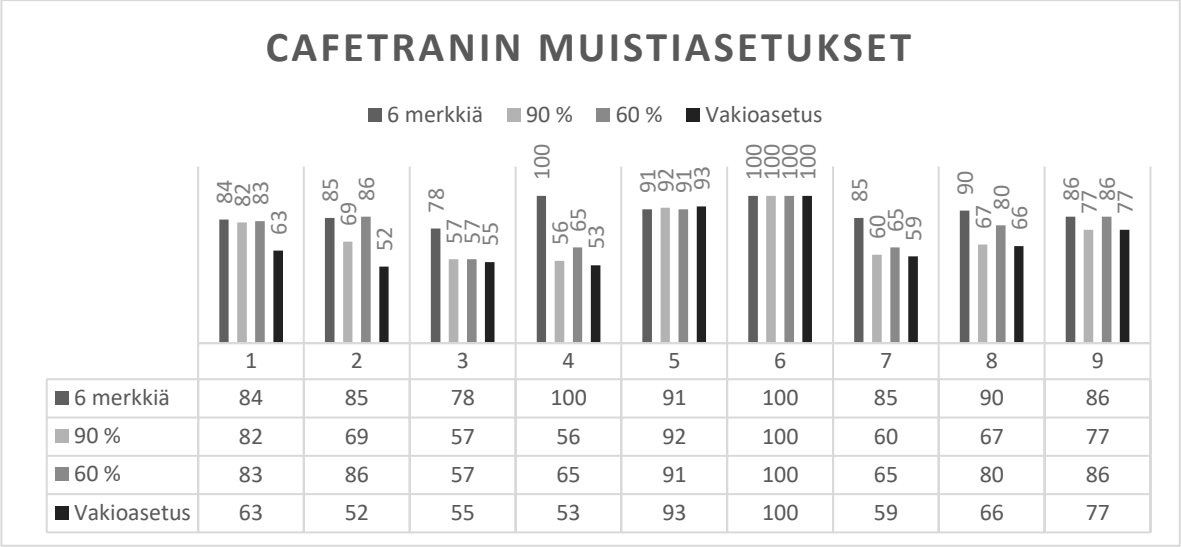
Suurin ero huonoimman ja parhaimman vastaavuuden välillä ilmeni segmentissä seitsemän – ”Oppositionkin mielestä sote olisi kallis ja huono” – jossa OmegaT ilmoitti muistiosuman tarkkuudeksi 100 prosenttia CafeTranin jäädessä 59 prosenttiin. Tämä on lähes varmasti esimerkki OmegaT:n tokenisoinnista, joka on onnistunut tunnistamaan muodot *Opposition* ja *Oppositionkin*, minkä lisäksi *on* ja *olisi* esiintyvät sen sulkusanaluettelossa, joten ne on jätetty vastaavuutta laskettaessa huomiotta. Samalla se herättää kuitenkin kysymyksen siitä, miksi OmegaT ei ole pärjännyt yhtä hyvin testin muissa segmenteissä. Yksi mahdollinen

selitys on se, että seitsemäs segmentti oli näistä ainoa, jossa kaikki taivutusmuodot, tai -päätteet, sisältyivät tokenisointityökalun sanatietokantaan ja jossa jäljelle jääneet sanat (tai tässä tapauksessa sana) laskettiin sulkusanoiksi. Tulos on joka tapauksessa osoitus siitä, ettei OmegaT:n suomen kielen tokenisointi ole aivan vielä parhaalla mahdollisella tasolla.

Parhaiten tunnistuksessa onnistuneen Tradoksen ja toiseksi parhaiten onnistuneen OmegaT:n välinen ero on useimmissa tapauksissa enintään viisi prosenttiyksikköä ja vain kahdessa segmentissä yli kymmenen prosenttiyksikköä. Suurimpia eroja aiheuttaneet virkkeet ovat edellisessäkin kappaleessa mainittu Oppositio-virke, jossa ero on 19 prosenttiyksikköä OmegaT:n eduksi, sekä virke ”Soteen kuuluvat myös lakiehdotukset valinnanvapaudesta”, jossa Trados on tarkempi 20 prosenttiyksikön turvin. On vaikea sanoa, miksi näiden kahden, keskimäärin melko pitkiä muokattuja sanoja (*oppositionkin, kuuluvat, lakiehdotukset*) sisältävän segmentin tunnistustuloksissa niin paljon eroa. Se voi selittyä sillä, että ensimmäinen segmentti sisältää suoraan alkutekstin sanan loppuun lisätyn *-kin*-liitteen ja *olla*-verbin taivutusmuodon, joka on voitu lisätä tokenisointityökaluun sen yleisyyden vuoksi, kun taas jälkimmäisessä segmentissä muokattujen sanojen alkuperäiset taivutuspäätteet eivät ole enää nähtävissä. Mikäli tämä pitää paikkansa, OmegaT:n pitäisi tuottaa täydellisiä osumia aina, kun sanat muuttuvat ainoastaan niiden perään lisättävien taivutuspäätteiden verran. Tämä hypoteesi oli mielestäni niin mielenkiintoinen, että päätin testata sen käytännössä luomalla uuden käännettävän tekstin, jossa alkutekstin sanoista ei ole poistettu mitään. Vaikka tulokset olivatkin tässä melko hyviä, eivät ne olleet kuitenkaan kaikki sataa prosenttia, joten tokenisointi ei toiminut odottamallani tavalla.

Toinen tekemisen arvoinen lisätesti on CafeTranin testaaminen muilla kuin ohjelmassa oletuksena valituilla asetuksilla, sillä niillä tulokset jäivät selvästi muita heikommiksi. Koska oletin ongelmien johtuvan enimmäkseen siitä, että CafeTranin Prefix matching ei ole käytössä, toteutin uusintatestin, jossa vertailin ohjelmansisäisiä tuloksia asetuksen kolmella eri arvolla: kiinteä merkkimäärä (*fixed length*), suurin sallittu prosentuaalinen arvo 90 % (CafeTran jättää huomiotta sanan viimeiset 10 %) ja hieman puolenvälin yläpuolta edustava 60 % (CafeTran jättää huomiotta sanan viimeiset 40 %). Koska EU:n tekstien pohjalta luodussa niin sanotussa Parole-korpuksessa suomenkielisen sanan keskipituuden on

havaittu olevan 8,5 merkkiä ja koska taivutuspäätteiden pituus vaihtelee, ainakin enimmäkseen, yhden ja neljän merkin välillä, päätin valita sanavartalon ja taivutuspäätteen rajaa edustavaksi kiinteäksi merkkimääräksi kuusi (Heikkinen, Lehtinen & Lounela 2001). Testissä parhaan tuloksen saisi asettamalla minkä tahansa edellä esitetyistä kolmesta arvosta mahdollisimman matalaksi, mutta käytännössä tulokset olisivat vähemmän imartelevia, kun selvästi erilaisetkin segmentit näkyisivät käyttäjälle lähes täydellisinä vastaavuuksina. Esimerkiksi kiinteällä yhden merkin prefix matching -asetuksella ohjelma tunnistaisi kaikki samalla kirjaimella alkavat sanat samoiksi sanoiksi, mikä vääristäisi testin tuloksia ja olisi käytännön työssä käyttökelvoton ratkaisu.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 6 merkkiä | 84 | 85 | 78 | 100 | 91 | 100 | 85 | 90 | 86 |
| 90 % | 82 | 69 | 57 | 56 | 92 | 100 | 60 | 67 | 77 |
| 60 % | 83 | 86 | 57 | 65 | 91 | 100 | 65 | 80 | 86 |
| Vakioasetus | 63 | 52 | 55 | 53 | 93 | 100 | 59 | 66 | 77 |

Kuva 15: Taivutuspäätetestin tulokset CafeTranin eri muistiasetuksilla.

Kuten yllä olevasta kuvasta voi nähdä, muistiasetusten säätämisellä on CafeTranin tapauksessa erittäin suuri tarkkuutta parantava vaikutus. Oletin 90 %:n asetuksen tuottavan oletusasetusten jälkeen heikoimpia tuloksia, ja näin tulosten mukaan myös kävi. Tämä johtuu yksinkertaisesti siitä, että kyseisellä asetuksella jo pienikin vaikutus sanan loppuun heikentää osumatarkkuutta. Kuuden merkin kiinteä sanavartaloraja vaikutti tuottavan parhaat tulokset – itse asiassa niin hyvät, että kyseisellä asetuksella CafeTran kykeni samaan tulokseen Tradoksen kanssa. Testin segmenttien keskimääräinen osumatarkkuus oli kiinteän merkkimäärän CafeTranilla 89 prosenttia (vrt. oletusasetusten 69 prosenttia), kun taas Tradoksella sama oli 89 prosenttia ja OmegaT:llä 85 prosenttia.

Mielenkiintoisin on viides segmentti (”Tätä varten eduskunnan täytyy päättää yhteensä [40:stä] uudesta laista ja lainmuutoksesta”), jossa CafeTranin oletusasetukset tuottivat parhaan tuloksen. Segmentin ainoa muutos käännösmuistin segmenttiin on taivutuspäätteen lisääminen lukuun kaksoispisteen avulla. Käsittääkseni kaksoispisteen pitäisi toimia tässä joko tavallisen merkin tavoin tai sanarajaa ilmaisevana merkkinä. Kummassakin tapauksessa vakiomuistiasetuksen pitäisi nähdäkseni tuottaa huonompia tuloksia kuin taivutusmuodot huomioivien tulosten, minkä lisäksi en keksi järkevää syytä siihen, että 60 prosentin asetus tuottaa yhtä prosenttiyksikköä huonomman tuloksen kuin suuremman samankaltaisuuden edellyttävä 90 prosentin asetus. Pitäisihän sanan lopusta suuremman osuuden huomiotta jättävän asetuksen olla kaikissa tilanteissa 90 %:n vertailuasetusta tarkempi kaikissa tilanteissa, kun muut asetukset ovat identtiset.

Riippumatta siitä, mitä muistiasetuksia ohjelmissa (tai tässä tapauksessa ainoastaan CafeTranissa) käytetään, SDL Trados Studio vaikuttaa selviävän suomen kielen taivutuspäätteistä kahteen muuhun verrattuna hyvin. Oletusasetuksia käytettäessä OmegaT on sekin kohtuullisen tarkka, kun sitä vastoin CafeTran tarjoaa käyttäjilleen keskimäärin huonompia osumia. Näiden kahden osalta asetelma kääntyy päälaelleen, kun CafeTranin asetukset mukautetaan suomiystävällisempään muotoon, jolloin sen osumatarkkuus vastaa Tradosta. Huomion arvoista on toki sekin, että myös OmegaT:n muistiasetuksissa on jonkin verran säätövaraa muun muassa tokenisoinnin eri versioiden muodossa, sillä käyttäjä voi halutessaan vaihtaa käyttöön myös työkalun vanhemman version. En kuitenkaan testannut niiden vaikutusta osumatarkkuuteen tässä tutkielmassa, sillä oletan oletuksena käytössä olevan uusimman version tuottavan edeltäjiään paremmat tulokset.

## 5.2 Morfofonologinen vaihtelu

Morfofonologisen vaihtelun testissä voidaan todeta taivutusmuotoja helpommin, tunnistaako käännösmuistiohjelma sanojen lopun samankaltaisuuden, mikäli sanassa esiintyy aiemmin eroavaisuus, vai katkeaako tunnistus ensimmäiseen havaittavaan eroon. Tämä vaihe olisi ollut helppoa yhdistää myös edelliseen, jolloin kaikkien taivutusmuotojen vaikutus olisi voitu testata kerralla. Halusin kuitenkin selvittää erikseen sekä taivutuspäätteiden että morfofonologisen vaihtelun vaikutuksen tuloksiin. Erityisen kiinnostavaa on sanavartalon morfofonologinen vaihtelu, sillä en usko taivutuspäätteiden

morfofonologisella vaihtelulla olevan tässä juurikaan merkitystä. Tämä johtuu siitä, että ne muistuttavat, ainakin tietokoneohjelmien näkökulmasta, rakenteeltaan tavallisia taivutuspäätteitä.

Käytin tässä osiossa niin ikään *Selkosanomat*-verkkosivustolla 14.3.2018 julkaistua uutista ”Sauna Unescon luetteloon” (Österlund 2018b), josta valitsin testattavaksi vain sen ensimmäisen osuuden liian pitkän aineiston välttämiseksi. Testiaineiston valmistelu onnistuu edellisen vaiheen periaatteella sillä erotuksella, että morfofonologisen vaihtelun aineistossa lähtötekstin muokkaaminen sanoja lisäämällä tai vaihtamalla on tarpeen, sillä morfofonologista vaihtelua ei esiinny aineistossa muutoin riittävästi mielekkäiden tulosten saamisen kannalta. Alkuteksti on esitetty seuraavassa.

> Unesco on YK:n eli Yhdistyneiden kansakuntien *halpa* järjestö, joka edistää kasvatusta, tiedettä ja kulttuuria koko maailmassa. Unescon maailmanperintö-luettelossa luetellaan *villinä* eri maiden arvokasta kulttuuriperintöä. Luettelossa on esimerkiksi perinteitä, tansseja, seremoinoita [sic], leikkejä, lauluja*, vesi* ja ruokalajeja. Ensi vuonna Suomi *siirtää* omaa kulttuuriperintöään Unescon luetteloon. Ehdotuksesta päättää opetus- ja kulttuuriministeriö. *Kylmän* Suomen ehdotus valitaan suomalaisten omasta luettelosta. Suomalaisten luettelossa on 50 erilaista suomalaista perinnettä ja tapaa, esimerkiksi tällaisia perinteitä: joulukynttilän sytyttäminen haudoille, pääsiäiskokko, tervanpoltto, suomalainen tango, romanien lauluperinne*, puku* ja latotanssit (Österlund 2018b).

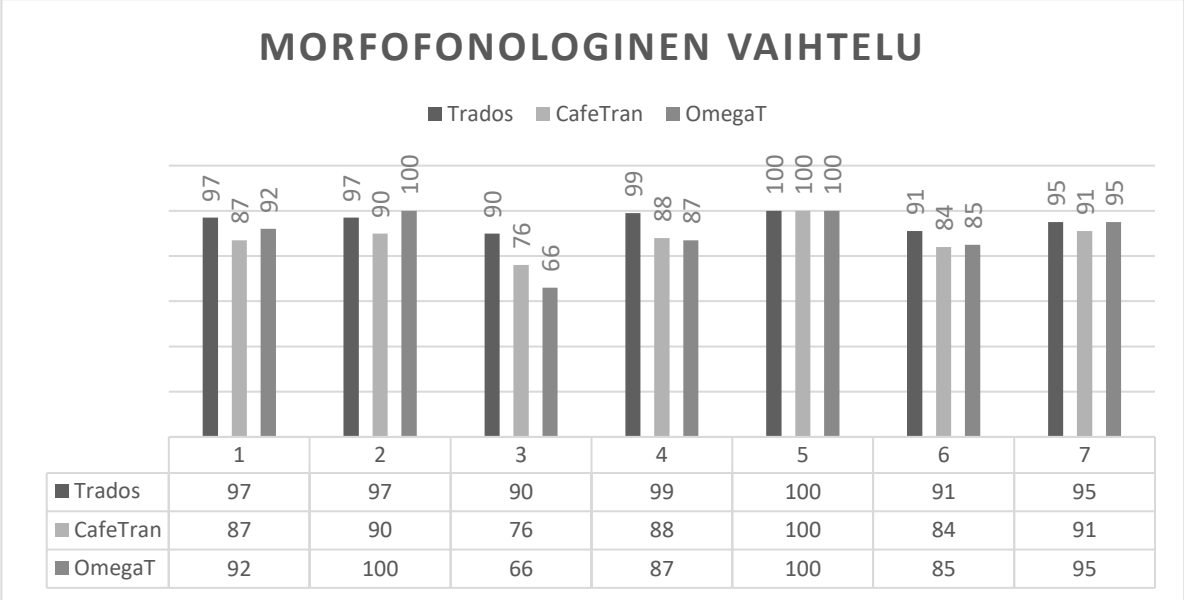
Esimerkkejä alkuperäisen tekstin sanoista, joissa esiintyy morfofonologista vaihtelua sen jossain muodossa, ovat *leikkejä* (*leikit*), *tanssi* (*tansseja*) ja *haudoille* (*hauta*). Tekstissä jo olevien lisäksi päätin lisätä sinne esimerkkejä tapauksista, joissa morfofonologinen vaihtelu vaikuttaa sanaan rakenteeseen tavallista enemmän: tämän pitäisi taata mahdollisten erojen näkyminen näinkin lyhyessä tekstikokonaisuudessa. Lisätyt sanat on merkitty yllä olevaan tekstiin kursivoinnilla. Seuraavassa on edellisen osion tavoin muokattu teksti. Alkutekstin muutetut ja sinne lisätyt morfofonologista vaihtelua edustavat sanat on korostettu alleviivaamalla. Kuten huomata saattaa, muokattu alkuteksti tai niin sanottu vertailuteksti eivät kumpikaan edusta erityisen hyvää suomea. Tämä on kuitenkin tarkoituksellista, sillä virkkeiden muuttaminen kieliopillisesti oikeaan muotoon vaatisi niin paljon muutoksia, ettei mahdollisten erojen johtaminen morfofonologiseen vaihteluun olisi enää mahdollista.

> Unesco on YK:n eli Yhdistyneiden kansakuntien halvin järjestö, joka edistää kasvatusta, tiedettä ja kulttuuria koko maailmassa. Unescon maailmanperintö-luettelossa luetellaan villeinä eri maiden arvokasta kulttuuriperintöä. Luettelossa on esimerkiksi perinteitä, tanssit, seremoinoita, leikit, lauluja, vettä ja ruokalajeja. Ensi vuonna Suomi siirsi omaa kulttuuriperintöään Unescon luetteloon. Ehdotuksesta päättää opetus- ja kulttuuriministeriö. Kylmimmän Suomen ehdotus valitaan suomalaisten omasta luettelosta. Suomalaisten luettelossa on 50 erilaista suomalaista perinnettä ja tapaa, esimerkiksi tällaisia perinteitä: joulukynttilän sytyttäminen haudoille, pääsiäiskokko, tervanpoltto, suomalainen tango, romanien lauluperinne, puvut ja latotansseja.

Myös tässä tekstissä on yksi muokkaamaton virke, jonka avulla voin varmistaa, ettei käännösmuistin muodostuksessa tai tekstien tallennuksessa ole tapahtunut esimerkiksi huonosti yhteensopivasta tiedostomuodosta johtuvia virheitä.

### 5.2.1 Tulokset

Morfofonologisen testin tuloksissa on oletusasetuksilla vähemmän vaihtelua taivutuspäätteisiin verrattuna, mikä on hieman yllättävää. Tunnistamisen korkeat prosentit selittyvät kuitenkin erityisesti sillä, että tässä testissä muuttujia oli selvästi vähemmän edelliseen verrattuna. Prosenttiluvut siis eivät tässä tapauksessa kerro siitä, että käännösmuistiohjelmat olisivat jotenkin etevämpiä tunnistamaan morfofonologista vaihtelua – asia voi olla hyvin päinvastoin. Joka tapauksessa tuloksista voidaan kuitenkin päätellä, ettei morfofonologinen vaihtelu ole käännösmuisteille ja -ohjelmille merkittävästi taivutuspäätteitä ongelmallisempaa – tällöin eroja olisi pitänyt näkyä, vaikka muuttujien määrä olisikin pienempi. Vähäinen vaihtelu eri ohjelmien välillä johtuu sekin osittain muuttujien suhteellisen vähäisestä määrästä, mutta ennen kaikkea siitä, että CafeTran tuotti tällä kertaa kelvollisia tuloksia myös ohjelman vakioasetusten ollessa käytössä. Sen tulokset vakioasetuksilla olivatkin itse asiassa paremmat tai yhtä hyvät kuin muilla edellisen osion asetusvaihtoehdoilla kokeiltaessa.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Trados | 97 | 97 | 90 | 99 | 100 | 91 | 95 |
| CafeTran | 87 | 90 | 76 | 88 | 100 | 84 | 91 |
| OmegaT | 92 | 100 | 66 | 87 | 100 | 85 | 95 |

Kuva 16: Vaihtelutestin tulokset.

Kuten jo yllä olevaa kuvaa vilkaisemalla on ilmeistä, Trados suoriutui tästäkin testistä muita puhtaammin. Sen keskimääräinen osumatarkkuus oli 96 % eli 8 prosenttiyksikköä parempi kuin CafeTranilla ja 7 prosenttiyksikköä parempi kuin OmegaT:llä. Yhteenlaskettujen tulosten erot ohjelmien välillä ovat itse asiassa merkittävästi suuremmat kuin taivutuspäätetestissä, joskin on muistettava, että CafeTran osoitti siinä hyvää tarkkuutta vasta muistiasetuksia viilaamalla ja OmegaT:n keskiarvoa nostaa 19 prosenttiyksikköä parempi – joskin ansaittu – tarkkuus Tradoksen verrattuna testin seitsemännessä segmentissä.

OmegaT tuotti testissä sekä parhaan että huonoimman tuloksen, mikäli muokkaamatonta segmenttiä ei oteta huomioon. Parhaiten se onnistui toisessa segmentissä (”Unescon maailmanperintö-luettelossa luetellaan villeinä eri maiden arvokasta kulttuuriperintöä”), joka oli mielestäni myös testimateriaalin mielenkiintoisin siksi, että siinä ainoa muuttunut sana muuttui sanavartalon lopusta vain yhdellä lisätyllä kirjaimella sanan loppuosan säilyessä sellaisenaan. Sataprosenttinen osuma ei kuitenkaan tarkoita tässä tapauksessa täydellistä tulosta, sillä virke sisältää selvästi muuttuneen sanan, joka myös muuttaa hieman virkkeen merkitystä. Huolimaton kääntäjä voi jättää tällaisen osuman tekstiin sellaisenaan, mikä johtaa luonnollisesti väärään käännökseen. CafeTranin 90 prosentin vastaavuus viittaa puolestaan siihen, että (lähes) koko sana on jätetty pois laskuista prosentuaalista

vastaavuutta laskettaessa. Tradoksen 97 prosentin vastaavuus lienee näin ollen lähinnä hypoteettista ”todellista” vastaavuutta.

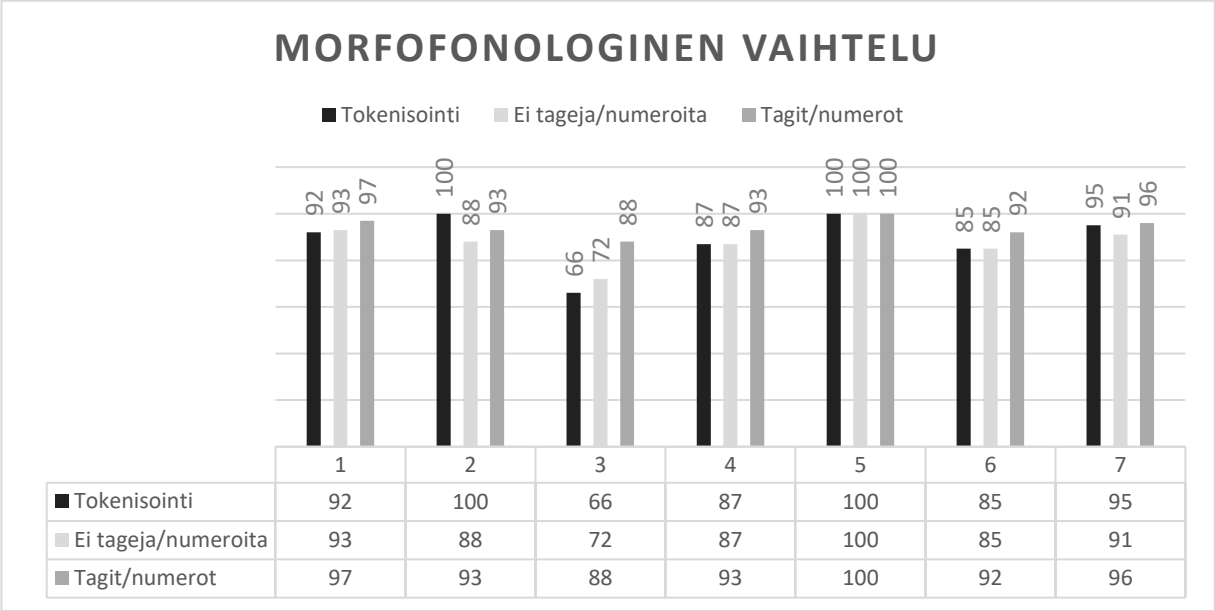
Myös edellistä esimerkkiä suoraan seuraava virke on mielenkiintoinen, sillä siinä Tradoksen ja CafeTranin välinen ero on 14 prosenttiyksikkö ja CafeTranin ja OmegaT:n välinen ero lähes yhtä suuri 10 prosenttiyksikköä. Kyseissä virkkeessä (”Luettelossa on esimerkiksi perinteitä, tanssit, seremoinoita, leikit, lauluja, vettä ja ruokalajeja.”) on peräti kolme muuttunutta sanaa, mikä voi johtaa yksinkertaisen tunnistusvirheen kertautumiseen, minkä lisäksi jokainen muuttunut sana muuttuu pituuteensa suhteutettuna melko paljon (*tansseja > tanssit, leikkejä > leikit, vesi > vettä*). CafeTranin tulos on tässä melko yhdenmukainen sen olettamuksen kanssa, että se laskee jokaisen sanan lähes nollaosumaksi: nollaosumina vastaavuusprosentti olisi 73 (suunnilleen, esimerkiksi tekstiin lisätty pilkku todennäköisesti laskisi tätä hieman), kun se nyt on 76.

OmegaT:n tulos edellisen esimerkin segmentissä on sekin enimmäkseen selitettävissä. Kuten edellä mainitsin, mikäli ohjelma ei tunnistaisi muutettuja sanoja lainkaan samankaltaisiksi, vastaavuuden pitäisi noin 73 prosenttia. OmegaT:n tapauksessa se on kuitenkin vain 66 prosenttia, mikä johtuu kahdesta huomiotta jätetystä sulkusanasta *on* ja *ja*. Tarkastelin kuitenkin kyseistä segmenttiä ohjelmassa tarkemmin ja huomasin, että OmegaT:n tarjoamat muut vastaavuudet olivat tokenisoinnilla saatavaa tulosta parempia. Tulos, jossa samoiksi tunnistettujen sanojen lukumäärä jaetaan kokonaissanamäärällä, tuotti ennalta arvattavan 72 prosentin vastaavuuden, mutta sama menetelmä niin, että myös numerot ja tunnisteet otetaan huomioon, tuotti yllättäen 88 prosentin vastaavuuden.

Yksinkertaisin, eli useimmiten myös oikea, selitys olisi se, että sekä muokatussa tekstissä että käännösmuistissa on näkymättömiä muotoilutunnisteita tai muita tageja. Käännösmuistin muodostamiseen käytetty alkuteksti ja sen muokattu versio ovat kuitenkin yksinkertaisia .txt-tekstitiedostoja juuri tämän ongelman välttämiseksi. Lisäksi tässä tapauksessa tunnisteiden pitäisi aiheuttaa ongelmia, ja visuaalisesti näkyä, myös muissa ohjelmissa, mutta tulosten nojalla näin ei vaikuttaisi olevan. Kokeilin tästä huolimatta tallentaa tiedoston eri merkistöillä siltä varalta, että alkujaan käyttämäni UTF-8 olisi jostakin syystä huonosti yhteensopiva OmegaT:n kanssa, mutta tulokset olivat identtisiä

käytetystä merkistöstä riippumatta. Epätodennäköistä on sekin, että muistin luomiseen käytetty memoQ olisi lisännyt muistisegmentteihin sinne kuulumattomia tageja, sillä tällöinhän osumatarkkuuden pitäisi olla huonompi tagien ja numeroiden tunnistuksen ollessa käytössä. Varmistin tagien poissaolon lopullisesti käyttämällä OmegaT:n projektiasetuksissa olevaa Remove Tags -työkalua, joka nimensä mukaisesti poistaa käännettävästä tekstistä tunnisteet, mutta edes tällä ei ollut vaikutusta tuloksiin.

Syystä riippumatta ongelma koskee vain OmegaT:tä, joten en usko testausmenetelmässä olevan vikaa. Perinteisesti pelkissä tekstitiedostoissa on vähemmän muotoilutietoja kuin esimerkiksi Microsoftin .doc- tai .docx-tiedostoissa, joten hyvin todennäköisesti ongelma olisi vain korostunut muilla tiedostomuodoilla. Tällaisissa ohjelmistoja ja niiden toimintaa sisältävissä tutkimuksissa kaikkien epäjohdonmukaisuuksien selittäminen on liki mahdotonta, ellei ohjelma ole testaajan itsensä kehittämä tai hyvin yksinkertainen avoimeen lähdekoodiin perustuva ohjelma, jolloin vastaus voisi löytyä ohjelmointikoodin kätköistä. OmegaT:n muiden tunnistustapojen tulos käsittelemässäni ongelmasegmentissä oli kuitenkin niin mielenkiintoinen, että päätin uusia testin keskittyen pelkästään OmegaT:n kolmen tunnistusmenetelmän vertailuun. Testin tulokset näkyvät alla olevassa kaaviossa.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Tokenisointi | 92 | 100 | 66 | 87 | 100 | 85 | 95 |
| Ei tageja/numeroita | 93 | 88 | 72 | 87 | 100 | 85 | 91 |
| Tagit/numerot | 97 | 93 | 88 | 93 | 100 | 92 | 96 |

Kuva 17: Vaihtelutestin tulokset OmegaT:n eri tunnistusmenetelmillä.

Kolmas segmentti osoitti jo, että tulokset voivat tässä testissä hyvinkin yllättää, ja niin myös kävi. Tokenisointiin perustuva tunnistus osoittautui lähes jokaisessa virkkeessä lähes yksinkertaisinta mahdollista tunnistustapaa – jossa siis tunnistettujen sanojen määrä, tagit ja numerot huomioiden, jaetaan kokonaissanamäärällä – huonommin toimivaksi. Yksinkertaisin menetelmä osoittautui itse asiassa yllättävänkin toimivaksi: sen keskimääräinen 94 prosentin osumatarkkuus jäi Tradoksen vastaavasta ainoastaan kaksi prosenttiyksikköä.

## 5.3 Sanajärjestys

Sanajärjestys on tutkittavista suomen kielen piirteistä viimeinen ja kenties myös mielenkiintoisin. Mielenkiintoisen siitä tekee erityisesti se, että sanoihin itseensä ei tehdä tässä lainkaan muutoksia, joten periaatteessa sekä sataprosenttiset osumat että hyvin huonot tunnistustarkkuudet ovat yhtä mahdollisia riippuen siitä, millaiset tunnistusalgoritmit ohjelmissa on. Sataprosenttinen tarkkuus tosin edellyttäisi todennäköisesti jonkinlaista erityisesti suomelle ja suomea läheisesti muistuttaville kielille räätälöityä menetelmää, jota en usko näistä ohjelmista löytyvän. Trados menestyi kahdessa aikaisemmassa testissä muita paremmin tai vähintään yhtä hyvin, joten oletan sen olevan tarkka myös tässä. OmegaT:llä ja CafeTranilla on kummallakin ollut omat ongelmansa, joten niiden osalta ennustuksia on vaikea tehdä.

Sanajärjestystä testattaessa aineistolla ei ole juurikaan merkitystä: sanojen pituus on yhdentekevä eikä virkkeiden pituudellakaan ole muuta väliä kuin mahdollisten tunnistustarkkuuden erojen korostaminen, mikäli useamman sanan järjestystä vaihdetaan. Pitäydyn kuitenkin hyväksi havaituissa *Selkosanomien* artikkeleissa ja valitsen tähän testiin 28.3.2018 julkaistun artikkelin ”Facebookia arvostellaan”, joka on esitetty alla.

> Britannian viranomaiset tutkivat tapausta, jossa 50 miljoonan amerikkalaisen Facebook- käyttäjän tietoja on kerätty ja myyty. Käyttäjille oli kerrottu, että tietoja käytetään tutkimukseen. Nyt näyttää siltä, että tiedot myytiin brittiläiselle Cambridge Analytica -yritykselle. Tietoja on käytetty muiden muassa Donald Trumpin kampanjassa Yhdysvaltain presidentinvaaleissa. Britanniassa tutkitaan myös, onko tiedoilla yritetty vaikuttaa äänestykseen, jossa britit päättivät erota EU:sta. Helsingin Sanomien mukaan monet yritykset voivat kerätä tietoja Facebookin käyttäjistä. Näin kertoo Sandy Parakilas, joka on työskennellyt Facebookissa. – Monet yritykset voivat kerätä tietoja. Sitä ei valvota ollenkaan.

> Yritykset voivat tehdä tiedoilla mitä ne haluavat, sanoi Parakilas. Jussi Parikka on Turun yliopiston dosentti ja Southamptons Universityn professori. Hän sanoo Helsingin Sanomissa, että Cambridge Analytican tapaus ei varmasti ole ainoa. Facebookin liikeidea on kerätä monenlaista tietoa käyttäjistä ja myydä sitä edelleen. Tietoja käytetään mainonnassa, jota suunnataan tietyille ryhmille. Facebook tietää, missä kuvamme on otettu, mihin ryhmiin kuulumme ja keihin olemme yhteydessä Facebookissa (Österlund 2018c).

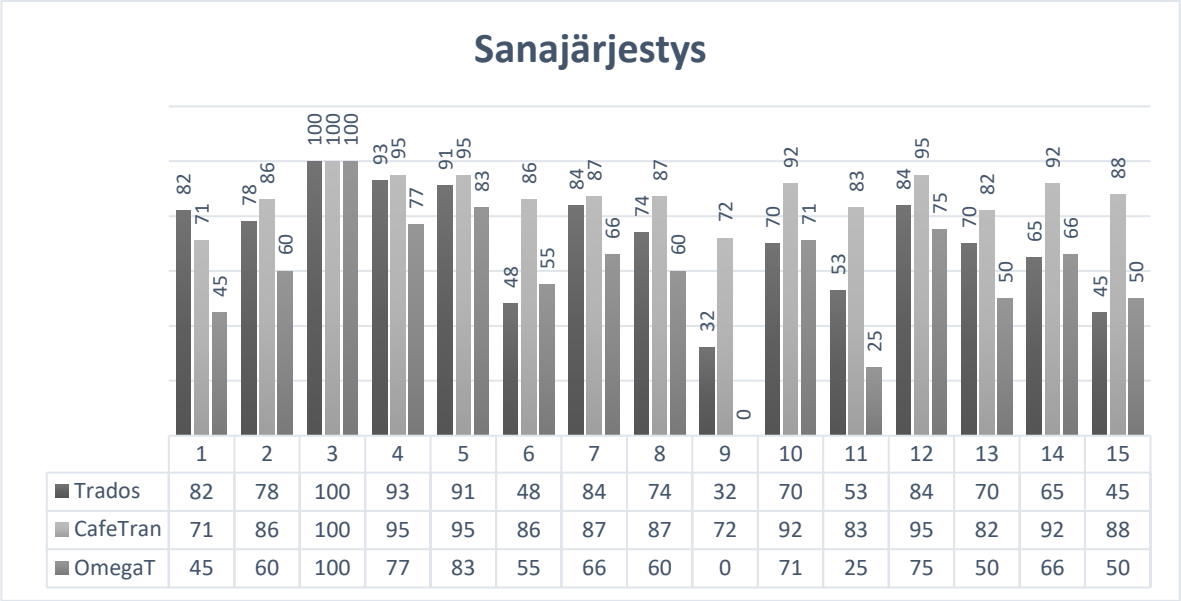
Valitsin tähän hieman pidemmän tekstin siksi, että voin helpommin tarkastella sekä virkkeen yksittäisten muutosten että useiden muutosten vaikutusta tuloksiin. Kiinnostavia esimerkkejä ovat muun muassa yksittäisen sanan paikan siirtäminen, kahden sanan paikkojen vaihtaminen, suurempien kokonaisuuksien, kuten kokonaisen lauseen, paikan vaihtaminen ja virke, jossa kaikkien tai ainakin lähes kaikkien sanojen paikat vaihtavat paikkaa. Tehdyt muutokset on merkitty artikkelin muokattuun versioon niin, että siirretyt osat ovat hakasulkeissa.

> Tapausta, jossa 50 miljoonan amerikkalaisen Facebook- käyttäjän tietoja on kerätty ja myyty, [tutkivat] [Britannian viranomaiset]. [Kerrottu] oli [käyttäjille], että tietoja käytetään tutkimukseen. Nyt näyttää siltä, että tiedot myytiin brittiläiselle Cambridge Analytica -yritykselle. Tietoja on [muiden muassa Donald Trumpin kampanjassa Yhdysvaltain presidentinvaaleissa] käytetty. Britanniassa [myös] tutkitaan, onko tiedoilla yritetty vaikuttaa äänestykseen, jossa britit päättivät erota EU:sta. Monet yritykset voivat kerätä tietoja Facebookin käyttäjistä [Helsingin Sanomien mukaan]. Näin kertoo Sandy Parakilas, joka [Facebookissa] [työskennellyt] [on]. – Monet yritykset voivat [tietoja] kerätä. [Ollenkaan] [ei] [sitä] [valvota]. Yritykset voivat tehdä, [mitä ne haluavat], tiedoilla, sanoi Parakilas. Jussi Parikka on [Southamptons Universityn professori] ja [Turun yliopiston dosentti]. Hän sanoo Helsingin Sanomissa, että Cambridge Analytican tapaus ei [ole] [ainoa] [varmasti]. Facebookin liikeidea on [käyttäjistä] [monenlaista tietoa] kerätä ja sitä edelleen [myydä]. Tietoja käytetään mainonnassa, jota [tietyille ryhmille] suunnataan. Facebook tietää, [mihin ryhmiin kuulumme], [keihin olemme yhteydessä Facebookissa] ja [missä kuvamme on otettu].

### 5.3.1 Tulokset

Viimeisen testin tulokset olivat eniten yllättäviä ja myös vaihtelua esiintyi selvästi enemmän kuin muissa testeissä. Eroja on paljon paitsi eri käännösmuistiohjelmien suoritustasossa myös samojen ohjelmien suoritustasossa eri segmenteissä. Yleisesti sanajärjestyksen voi todeta aiheuttavan ohjelmille enemmän ongelmia päätteisiin ja morfofonologiseen vaihteluun verrattuna, mikä on mielenkiintoista siksi, että samojen

sanojen tunnistamisen virkkeen eri kohdissa voisi luulla olevan huomattavasti helpompaa kuin sanojen eri taivutusmuotojen tunnistamisen. Testin tulokset on esitetty alla olevassa kaaviossa.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Trados | 82 | 78 | 100 | 93 | 91 | 48 | 84 | 74 | 32 | 70 | 53 | 84 | 70 | 65 | 45 |
| CafeTran | 71 | 86 | 100 | 95 | 95 | 86 | 87 | 87 | 72 | 92 | 83 | 95 | 82 | 92 | 88 |
| OmegaT | 45 | 60 | 100 | 77 | 83 | 55 | 66 | 60 | 0 | 71 | 25 | 75 | 50 | 66 | 50 |

Kuva 18: Sanajärjestystestin tulokset.

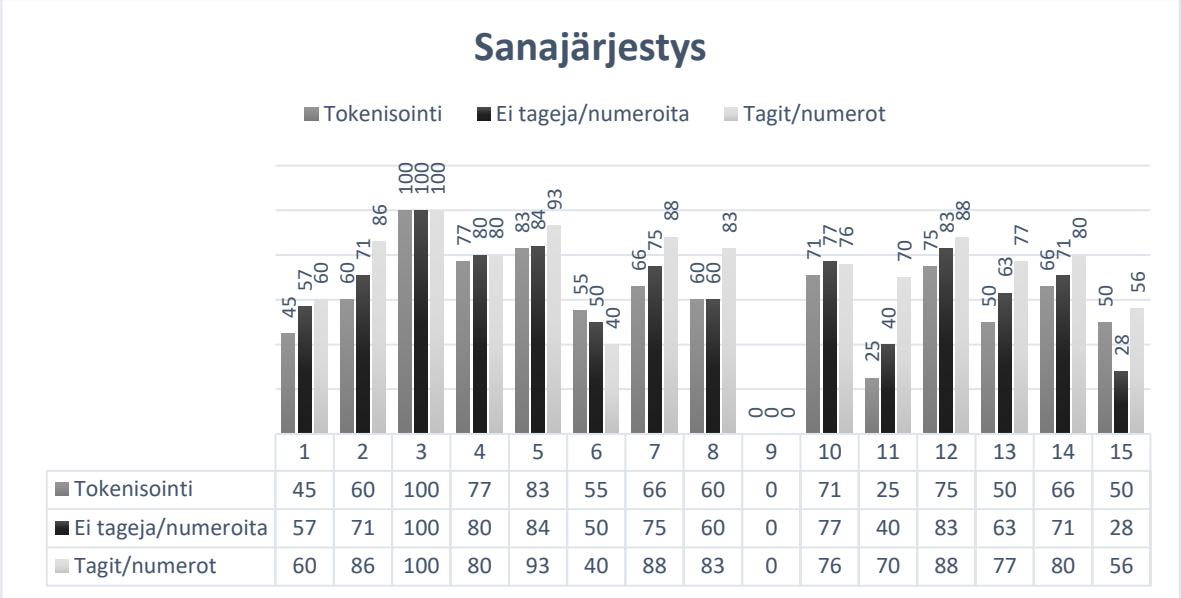
Trados, joka oli tähän asti vertailun selvästi paras, ei tunnistanut sanajärjestyksen muutoksia kovinkaan hyvin. Testin viidestätoista – tai käytännössä neljästätoista, sillä kolmannessa segmentissä ei ole lainkaan muutoksia – segmentistä se tarjosi huonoimman tunnistustarkkuuden jopa neljässä. Huonoin tarkkuus ei toki automaattisesti tarkoita huonoa tarkkuutta, mikäli jokainen testattu ohjelma saavuttaa testissä hyvän tuloksen. Tässä tapauksessa se ei kuitenkaan päde, sillä Tradoksen tunnistustarkkuus oli alle 50 prosenttia kolmessa segmentissä ja alle 75 peräti kahdeksassa. Vertailu ei mairittele myöskään muihin ohjelmiin vertailtaessa: Tradoksen tarkkuus oli kolmesta ohjelmasta paras vain yhdessä segmentissä.

OmegaT puolestaan saa Tradoksen näyttämään lähes loistavalta, sillä sen keskimääräinen tunnistustarkkuus oli vain 59 prosenttia, mikä on 12 prosenttiyksikköä huonompi kuin Tradoksen vastaava. OmegaT jäi joukon hännille kymmenessä segmentissä, ja yhdessä niistä sen tulokseksi jäi pyöreä nolla. Nolla ei ole välttämättä, eikä edes todennäköisesti, tarkkaa tunnistustarkkuutta edustava luku, mutta OmegaT ei suostunut näyttämään

kyseiselle segmentille (”[Ollenkaan] [ei] [sitä] [valvota].”) minkäänlaista osumaa millään muistiasetuksilla, minkä vuoksi muun kuin nollan käyttöä ei voi näiden tietojen pohjalta oikeuttaa. OmegaT osoitti lisäksi yhdennessätoista segmentissä kykenevänsä näyttämään tuloksia 25 prosenttiin asti, joten tyhjäksi jääneen segmentin tunnistustarkkuus ei voi olla ainakaan tätä suurempi. Kyseinen segmentti, jossa jokaisen sanan paikka on muuttunut käännösmuistissa olevaan virkkeeseen verrattuna, tuotti ongelmia myös muille testin osapuolille. CafeTran oli kolmikosta ainoa, jolla se ei ollut huonoimman tunnistustarkkuuden tuottava segmentti, ja silläkin eroa sen huonoimpaan tulokseen oli vain yhden prosenttiyksikön verran.

Mikäli OmegaT:n kolmen käännösmuistimenetelmän vertailutulokset olivat morfofonologisessa testissä hämmentäviä, olivat ne sanajärjestyksen osalta jo lähes huolestuttavia. Siinä missä viime testissä eri menetelmät tuottivat enemmän tai vähemmän toistensa kaltaisia tuloksia niin, että tagit ja numerot huomioiva yksinkertainen tunnistus oli vain hieman muita parempi, tällä kertaa ero parhaan ja toiseksi parhaan menetelmän välillä oli peräti yhdeksän prosenttiyksikköä. Suurin yksittäisen segmentin ero oli yhdennessätoista segmentissä, jossa tunnisteet ja numerot tuottivat 30 prosenttiyksikön eron näitä huomioimattomaan menetelmään ja 45 prosenttiyksikön eron tokenisoivaan tunnistamiseen verrattuna. Kyseisessä segmentissä (”Jussi Parikka on [Southamptons Universityn professori] ja [Turun yliopiston dosentti]”) esiintyy poikkeuksellisen paljon suuria kirjaimia, mikä herättää luonnollisesti seuraavan kysymyksen: merkitseekö OmegaT suuret kirjaimet (muualla kuin sanan alussa) käyttämällä näkymättömiä tunnisteita? Tämä voisi selittää tulosten eroja näennäisesti identtisissä virkkeissä, joten päätin tutustua asiaan tarkemmin löytääkseni syyn näille niin sanotuille haamutageille.

Vertailin OmegaT:n kolmen tunnistusmenetelmän tuloksia morfofonologisessa vaihtelutestissä tekemäni vertailun tavoin keskittyen erityisesti niihin segmentteihin, joissa numerot ja tagit huomioiva tunnistus oli muita menetelmiä selvästi parempi sekä niihin, joissa eroja ei juurikaan tai lainkaan esiintynyt. Kyseisen testin tulokset on esitetty alla olevassa kaaviossa.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Tokenisointi | 45 | 60 | 100 | 77 | 83 | 55 | 66 | 60 | 0 | 71 | 25 | 75 | 50 | 66 | 50 |
| Ei tageja/numeroita | 57 | 71 | 100 | 80 | 84 | 50 | 75 | 60 | 0 | 77 | 40 | 83 | 63 | 71 | 28 |
| Tagit/numerot | 60 | 86 | 100 | 80 | 93 | 40 | 88 | 83 | 0 | 76 | 70 | 88 | 77 | 80 | 56 |

Kuva 19: Sanajärjestystestin tulokset OmegaT:n eri tunnistusmenetelmillä.

Nopeasti on nähtävissä, että tagit/numerot-menetelmä tuottaa muita keskimäärin parempia osumia ja toisaalta että tokenisointi vaikuttaisi olevan menetelmistä kaikkein tehottomin. Tästä poikkeaa selvästi ainoastaan segmentti 15, jossa tagit ja numerot huomioimatta jättävä menetelmä menestyy selvästi kahta muuta heikommin. Kyseinen segmentti (”Facebook tietää, [mihin ryhmiin kuulumme], [keihin olemme yhteydessä Facebookissa] ja [missä kuvamme on otettu].”) ei päällisin puolin kerro, mistä poikkeava tulos johtuu – kolme (lähes) perättäistä muokkausta esiintyvät myös segmenteissä 7 ja 12, mutta näissä tulokset ovat linjassa muiden segmenttien kanssa. Suurin ero edellä mainittuihin on se, että 15. segmentissä jokainen tekstin osa, jonka paikkaa on vaihdettu, koostuu useammasta sanasta. Tämä puolestaan pitää paikkansa myös segmentin 11 (”Jussi Parikka on [Southamptons Universityn professori] ja [Turun yliopiston dosentti].”) osalta, joten sekään ei nähdäkseni selitä 15. virkkeen tunnistettavuuden poikkeavuutta.

Myöskään edellä esitettyyn kysymykseen liittyen isojen kirjainten käsittelyyn tunnisteina ei saatu vastausta. Tunnisteet huomioiva menetelmä on muita selvästi parempi segmenteissä 2, 8 ja 11, ja näistä vain 11. segmentti sisältää isoja kirjaimia muualla kuin virkkeen alussa. Näiden kolmen välillä ei vaikuttaisi olevan mitään muutakaan yhteistä, joka erottaisi ne muista. Kyseessä vaikuttaisikin olevan jälleen yksi esimerkki OmegaT:n sisäisen toiminnan mysteereistä.

Siinä missä muut ohjelmat olivat vaikeuksissa, CafeTran vaikutti loistavan. Sen keskimääräinen tarkkuus oli 87 prosenttia, ja yksittäisissä segmenteissä CafeTranin tunnistus toimi parhaiten jopa 13:ssa vertailukelpoisessa segmentissä (muokkaamaton pois lukien) ja jäljessä jääneessäkin sen tarkkuus oli toiseksi paras. Tämän perusteella CafeTranin on käsiteltävä sanajärjestystä muista poikkeavalla ja selvästi ylivertaisella tavalla.

# 6. Loppupäätelmät

Vaikka tulosten osalta ei ollut juurikaan odotuksia, onnistuivat ne osittain silti yllättämään. Kenties suurin yllätys oli eräänlainen johdonmukaisuuden puuttuminen: vaikka käännöstyökalut olikin mahdollista asettaa paremmuusjärjestykseen kussakin yksittäisessä testissä, esiintyi erityisesti OmegaT:ssä ja CafeTranissa huomattavaa virkekohtaista vaihtelua. Virkkeiden ja niihin tehtyjen muutosten erilaisuuden vuoksi eroja oli toki odotettavissa, mutta samojen ohjelmien merkittävästi erilaiset tulokset toisiaan vastaavien segmenttien tunnistamisessa olivat tästä huolimatta yllättäviä. Tämä vaikuttaisi olevan merkki siitä, ettei käännösmuistiohjelmien toiminnan johdonmukaisuus – ainakaan suomen kielen osalta – ole vielä parhaalla mahdollisella tasolla. Toisaalta tulokset voivat viitata myös testausmenetelmän epäjohdonmukaisuuteen: tarkemmin valikoidun ja valmistellun aineiston myötä myös tulokset olisivat saattaneet olla yhtenäisempiä. Tarkimman mahdollisen analysoinnin mahdollistava aineisto olisi kuitenkin melko keinotekoista ja myös kauimpana luonnollisesta, todellisen elämän käännöstehtävästä, joten tällainen tutkimus voisi soveltua paremmin käännösmuistien ja niitä käyttävien ohjelmien tekniseen tarkasteluun kuin niiden tarkasteluun kääntäjien näkökulmasta.

Edellä mainitusta epäjohdonmukaisuudesta huolimatta joidenkin johtopäätösten tekeminen on mahdollista. Yksi näistä on se, että yleisimmin käytetty SDL Trados Studio toimii oletusasetuksilla melko hyvin, eivätkä mitkään suomen kielen erityispiirteet aiheuta sille huomattavia haasteita, vaikka sanajärjestyksen vaihtuminen vaikuttaakin tuloksiin selvästi muita kielen ominaispiirteitä enemmän. Kaikkien kolmen testattavan osa-alueen keskimääräinen tunnistustarkkuus oli Tradoksella 85 %, CafeTranilla 81 % ja OmegaT:llä 78 %. Rajan vetäminen hyvien, keskinkertaisten ja huonojen tulosten välille on vaikeaa, kun testattavia ohjelmia on ainoastaan kolme eikä olemassa ole kaavaa ns. oikeiden tulosten laskemiseen, jolloin kutakin ohjelmaa voisi verrata kyseiseen arvoon. Ero parhaan ja huonoimman välillä on kuitenkin vain seitsemän prosenttiyksikköä, joten minkään tietyn ohjelman käyttö ei aseta kääntäjää merkittävästi kollegoita heikompaan tai parempaan asemaan. Luultavasti erilaisten tarkistustyökalujen olemassaolo tai vain käytön helppous (joka on toki subjektiivista) vaikuttaa kääntämisen nopeuteen ja tarkkuuteen enemmän, kuin tässä tutkielmassa havaitut erot tunnistustarkkuudessa. Vaikka Trados siis osoittautuikin luotettavaksi apuvälineeksi, näissä testeissä havaitut erot

tunnistustarkkuudessa eivät mielestäni riitä tekemään siitä ehdotonta valintaa käännösmuistiohjelman hankkimista harkitseville – erityisesti, kun ohjelmien väliset huomattavat hintaerot otetaan huomioon.

Lisäksi on huomattava, etteivät ohjelmien oletusasetuksilla saadut tarkkuudet kerro koko totuutta. OmegaT:n ja CafeTranin vaihtoehtoisten muistiasetusten hyvät tulokset osoittavat, etteivät ohjelmien oletusasetukset ole useinkaan täydellisiä, vaan kääntäjän on oltava valmis kokeilemaan, mikä niistä sopii tämän työkieliin parhaiten. Ennen kaikkea ne osoittavat kuitenkin sen, ettei näitä ohjelmia ole optimoitu suomen kaltaisten kielten kääntämiseen. Toisaalta OmegaT:n tokenisointiin sisältyvä sulkusanatoiminto, jonka tarkoitus oli parantaa tarkkuutta, vaikutti päinvastoin heikentävän niitä. Tämä selittyy nähdäkseni sillä, että monet sulkusanaluettelossa olevat sanat ovat harvoja suomen kielen taipumattomia sanoja: kun ne poistetaan tekstistä, jäljelle jäävillä eroilla on suurempi vaikutus kokonaistarkkuuteen. Tulokset olisivat varmasti melko erilaisia selvästi tavanomaisemmassa englannista suomeen käännettäessä, mutta tällöin kaikki suomen kielen mielenkiintoiset piirteet pitäisi jättää huomiotta, sillä kohdekielellähän ei ole käännösmuisteissa ja niiden tuloksissa merkitystä. Tästä huolimatta kääntäjien, oikolukijoiden ja muiden näitä ohjelmia käyttävien olisi järkevää tutustua edes pintapuolisesti niiden asetuksiin, sillä jo muutaman asetuksen muuttaminen voi selvästi parantaa tunnistustarkkuutta. Valitettavasti kääntäjien, ja erityisesti iäkkäämpien kääntäjien, keskuudessa esiintyy jonkin verran tietotekniikkakielteisyyttä, mikä voi johtaa siihen, ettei ohjelmistojen toimintaan tutustuta pakollista enempää (Suppanen 2015, s. 69).

Vaikka huolellinen asetusten hienosäätäminen näyttäisikin tuottavan paremman tunnistustarkkuuden, tuloksia tarkasteltaessa myös täysin vastakkainen näkökulma nostaa päätään. Erityisesti OmegaT:n toisinaan hämmentävät tulokset herättävät kysymyksen siitä, onko hienostunutta, taivutuspäätteiden tunnistamiseen pohjautuvaa menetelmää tai vaivalla koottua sulkusanaluetteloa edes mielekästä kehittää, jos kaikkein yksinkertaisimmalla mahdollisella metodilla saavutetaan sitä parempia tuloksia. Käyttäjien ja kehittäjien kannalta mahdollisimman hyvin mahdollisimman yksinkertaisesti toimiva kokonaisuus on luonnollisesti paras vaihtoehto. OmegaT:n tulokset herättävät hieman skeptisyyttä alati uudistuvia algoritmeja ja uusia muistiosumien ja kielenainesten tunnistamistekniikoita

kohtaan, mutta toivottavasti nämä ovat kuitenkin vain kasvukipuja, joista päästään tokenisoinnin ja muiden menetelmien kehittyessä.

Tulosten pohjalta voidaan vetää joitakin varovaisia johtopäätöksiä myös suomen ominaispiirteiden vaikutuksesta käännösmuistien käyttöön. Tärkein näistä on se, että sanajärjestys vaikuttaisi olevan eniten suomeen kääntävien työkaluiltaan saamaan apuun vaikuttava tekijä. CafeTranin hyvä tulos sanajärjestyksen osalta on osoitus siitä, etteivät kaikki ohjelmat suinkaan käsittele segmenttejä identtisellä tavalla. Tämä on mielestäni hyvä uutinen, sillä se voi toimia kannustimena kehittää algoritmeja myös suomen kaltaisten harvinaisempien kielten kannalta paremmiksi. Yhtä mieleenpainuvia eroja ei esiintynyt testin muissa osioissa, sillä esimerkiksi taivutuspäätteiden ja morfofonologisen vaihtelun erot selittyvät suurelta osin testien sisältämien muutosten välisestä erosta. Näiden kahden samankaltaisuuden vuoksi oletan silti taivutuspäätteiden vaikutuksen tunnistustarkkuuteen olevan vähäisempi siksi, että ne esiintyvät sanassa myöhemmin eivätkä aiheuta yksinään muutoksia sanavartaloihin.

Käännösteknologiaa tulisi joka tapauksessa ehdottomasti tutkia lisää myös Suomessa, sillä olemassa ei vaikuta olevan monia englanninkielisiä, saati sitten suomenkielisiä, kattavia tutkimuksia tai artikkeleita niiden toiminnan arvioimisesta käytännössä. Usein on keskitytty käännösteknologian käsitteisiin ja ominaisuuksiin teoreettisella tasolla tai kartoittamaan kääntäjien mielipiteitä teknologiasta tai heidän tietoteknistä osaamistaan. Väitän, että monien muiden alojen tavoin myös kääntämisessä erilaiset tekniset apuvälineet tulevat vain yleistymään tulevaisuudessa, joten on tärkeää, ettei aihetta sivuutettaisi myöskään kääntämistä tutkittaessa. Tätä tutkielmaa voi käyttää pohjana tulevalle tutkimustyölle, jossa voidaan keskittyä tarkemmin esimerkiksi käännösmuistien ja käännösohjelmien tekniseen toimintaan tai esimerkiksi sanajärjestyksen vaikutuksen tarkempaan analysointiin. Vaikka ihminen on toivon mukaan tulevaisuudessakin käännösprosessin tärkein yksittäinen toimija, ei prosessin muitakaan tekijöitä kannata jättää täysin huomiotta.

# Lähteet

## Tutkimusaineisto

Österlund, M. (2018a). Laki valinnanvapaudesta eteni eduskuntaan. Selkosanomat 14.3.2018. Uutinen. Osoitteessa: https://selkosanomat.fi/kotimaa/laki-valinnanvapaudesta-eteni-eduskuntaan/. Viitattu 17.4.2019.

Österlund, M. (2018b). Sauna Unescon luetteloon. Selkosanomat 14.3.2018. Uutinen. Osoitteessa: https://selkosanomat.fi/kotimaa/sauna-unescon-luetteloon/. Viitattu 17.4.2019.

Österlund, M. (2018c). Facebookia arvostellaan. Selkosanomat 28.3.2018. Uutinen. Osoitteessa: https://selkosanomat.fi/ulkomaat/facebookia-arvostellaan/. Viitattu 17.4.2019.

## Kirjallisuuslähteet

Alkula, R. (2000). Merkkijonoista suomen kielen sanoiksi. Tampereen yliopisto. Väitöskirja. Osoitteessa: http://urn.fi/urn:isbn:951-44-4886-3. Viitattu 17.4.2019.

Arenas, A. G. (2008). Productivity and quality in the post-editing of outputs from translation memories and machine translation. *Localisation Focus* 7(1), 11–21.

Biçici, E., & Dymetman, M. (2008). Dynamic translation memory: Using statistical machine translation to improve translation memory fuzzy matches. *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, Berliini, 454–465.

Bowker, L. (2002). Computer-aided translation technology: a practical introduction. University of Ottawa Press, Ottawa, 92–127.

Bowker, L. (2005). Productivity vs Quality? A Pilot Study on the Impact of Translation Memory Systems. Artikkeli. Osoitteessa: https://www.localisation.ie/sites/default/files/publications/Vol4_1Bowker.pdf. Viitattu 17.4.2019.

Briel, D. (2012). OmegaT. Dublin Computational Linguistic Research Seminars. Diaesitys. Osoitteessa: http://www.didierbriel.com/downloads/omegatdclrs.pdf. Viitattu 17.4.2019.

Briel, D. (2013a). Re: [OmTdev] Stemming with Hunspell. Keskustelualueen kommentti. Osoitteessa: http://sourceforge.net/p/omegat/mailman/message/30527768/. Viitattu 17.4.2019.

Briel, D. (2013b). Re: [OmTdev] Technical Request. Keskustelualueen kommentti. Osoitteessa: https://sourceforge.net/p/omegat/mailman/message/31144657/. Viitattu 17.4.2019.

CafeTran (2018). Multilingual Glossaries. Verkkokäyttöopas. Osoitteessa: https://cafetran.freshdesk.com/support/solutions/articles/6000113541-multilingual-glossaries. Viitattu 17.4.2019.

Colominas, C. (2008). Towards chunk-based translation memories. *Babel* 54(4), 343–354.

Dahl, Ö. (2008). Kuinka eksoottinen kieli suomi on? *Virittäjä* 112(4), 545–559.

Dimitriadis, J. (2019). TheCafeTranFiles. Verkkokäyttöopas. Osoitteessa: https://github.com/idimitriadis0/TheCafeTranFiles. Viitattu 17.4.2019.

European Commission Directorate-General for Translation (2017). Translation Tools and Workflow. Esite. Osoitteessa: https://publications.europa.eu/en/publication-detail/-/publication/00e51a8e-9c50-11e6-868c-01aa75ed71a1/language-en. Viitattu 17.4.2019.

Flanagan, K. (2015). Subsegment recall in Translation Memory — perceptions, expectations and reality. *The Journal of Specialised Translation* 23, 64–88.

Garcia, I. (2009). Beyond Translation Memory: Computers and the professional translator. *The Journal of Specialised Translation* 12, 199–214.

Garcia, I. (2012). Machines, translations and memories: language transfer in the web browser. *Perspectives* 20(4), 451–461.

Heikkinen, V., Lehtinen, O. & Lounela, M. (2001). Kuvia kirjoitetusta suomesta. *Kielikello* 3/2001. Osoitteessa: https://www.kielikello.fi/-/kuvia-kirjoitetusta-suomesta. Viitattu 17.4.2019.

Jokinen, P., Tarhio, J. & Ukkonen, E. (1996). A Comparison of Approximate String Matching Algorithms. *Softw: Pract. Exper.* 26, 1439–1458.

Kmitowski, I. (2012). History of CafeTran + questions about basic architecture. Keskustelualueen kommentti. Osoitteessa: https://groups.google.com/forum/#!msg/cafetranslators/kuX8gMmrxl4/57I_sAHSj1IJ. Viitattu 17.4.2019.

Korpela, J. (2014). A short introduction to the Finnish language. Osoitteessa: http://jkorpela.fi/finnish-intro.html. Viitattu 17.4.2019.

Lagoudaki, E. (2006). Translation memories survey 2006: Users' perceptions around TM use. Imperial College, Lontoo. Osoitteessa: http://www.mt-archive.info/Aslib-2006-Lagoudaki.pdf. Viitattu 17.4.2019).

Lehečková, H. (2012). Kielitypologia ja neurolingvistiikka. *Puhe ja kieli* 32(2), 59–67.

Mustonen, A. (2014). Pääosin ohjaamaton sanaston poiminta rakenteettomasta tekstistä. Oulun yliopisto. Diplomityö. Osoitteessa: http://jultika.oulu.fi/files/nbnfioulu-201404081248.pdf. Viitattu 17.4.2019.

Navarro, G. (2001). A guided tour to approximate string matching. *ACM Comput. Surv.* 33(1), 31–88.

OmegaT (2018). Compatibility. Osoitteessa: https://omegat.org/en/howtos/compatibility.html. Viitattu 17.4.2019.

Savoy, J. (2004). Report on CLEF-2003 Monolingual Tracks: Fusion of Probabilistic Models for Effective Monolingual Retrieval. Toim. Peters C. & al. *Comparative Evaluation of Multilingual Information Access Systems. CLEF 2003. Lecture Notes in Computer Science, vol 3237*. Springer, Berliini, 322–336.

SDL (n.d.). SDL History. Osoitteessa: https://web.archive.org/web/20170630102815/https:/www.sdl.com/about/sdl/history/. Viitattu 17.4.2019.

SDL Product Help (2014). Translation Memory Settings Dialog Box > Fields and Settings. Osoitteessa: http://producthelp.sdl.com/sdl%20trados%20studio/client_en/Ref/O-T/TM_Settings/TM_Setup_Dialog_Box_Fields_and_Settings.htm. Viitattu 17.4.2019.

SDL Product Help (2015). Creating Translation Memories. Osoitteessa: http://producthelp.sdl.com/SDK/TranslationMemoryApi/3.0/html/7e2a9a58-ce96-4e6a-9b50-20b8429d79e5.htm. Viitattu 17.4.2019.

SDL Trados (2019). Spring Special Offers! Osoitteessa: https://www.sdltrados.com/store/?target_audience=translator. Viitattu 17.4.2019.

SDL Trados (n.d.). About Us. Osoitteessa: https://www.sdltrados.com/about/history.html. Viitattu: 17.4.2019.

Smolej, V. (2018). OmegaT 3.5 - User's Guide. Verkkokäyttöopas. Osoitteessa: https://omegat.sourceforge.io/manual-latest/en/index.html. Viitattu 17.4.2019.

Somers, H. (2003). Translation Memory Systems. Toim. H. Somers. *Computers and Translation: A Translator's Guide*. John Benjamins, Amsterdam/Philadelphia, 31–46.

Suni, M. (2008). Toista kieltä vuorovaikutuksessa: kielellisten resurssien jakaminen toisen kielen omaksumisen alkuvaiheessa. Jyväskylän yliopisto. Väitöskirja. Osoitteessa: http://urn.fi/URN:ISBN:978-951-39-3209-1. Viitattu 17.4.2018.

Suomi, K., Toivanen, J. & Ylitalo, R. (2008). Finnish sound structure: phonetics, phonology, phonotactics and prosody. Oulun yliopisto, Oulu.

Suppanen, O. (2015). ”Freelancerin nyt pitää sitten osata itse oikeastaan kaikki.” Kääntäjien tietoteknisten taitojen kartoitusta kyselytutkimuksella. Tampereen yliopisto. Pro gradu -tutkielma. Osoitteessa: http://urn.fi/URN:NBN:fi:uta-201510292387. Viitattu 17.4.2019.

The Finnish Language in the Digital Age (2012). Toim. Rehm, G. & Uszkoreit, H. Valkoinen kirja. Osoitteessa: http://www.meta-net.eu/whitepapers/e-book/finnish.pdf. Viitattu 17.4.2019.

Van Assche, G. (2014). Why using SQLite for processing an XML format like TMX? Osoitteessa: https://web.archive.org/web/20140923065636/http://www.datamundi.be/cms/index.php/why-using-sql-on-an-xml-format-like-tmx. Viitattu 17.4.2019.

Verkkokielioppi (2001). Suomen kielen äänne-, muoto- ja lauseoppia. Finn Lectura & Savolainen, E., Helsinki. Osoitteessa: https://fl.finnlectura.fi/verkkokielioppi/aloitus.htm. Viitattu 17.4.2019.

VISK = Hakulinen, A., Vilkuna, M., Korhonen, R., Koivisto, V., Heinonen, T. R. & Alho, I. (2004). Iso suomen kielioppi. Verkkoversio. Suomalaisen Kirjallisuuden Seura, Helsinki. Osoitteessa: http://scripta.kotus.fi/visk. Viitattu 17.4.2019.

Yamada, M. (2011). The effect of translation memory databases on productivity. *Translation research projects* 3, 63–73.

## Summary in English

### Introduction

The purpose of this thesis is to analyze how the defining characteristics of Finnish impact the accuracy in translation memory matches when translating into Finnish using translation tools. Translation technology is, as a field, constantly evolving and growing and the number of tools is on a rise. However, most of the research is still focused on the attitudes and experiences translators have had instead of the actual inner workings of translation memories and the programs producing and making use of those. This thesis attempts at filling that void.

### Tools & technologies

This thesis focuses on the performance of three software tools: SDL Trados Studio, CafeTran, and OmegaT. All of these are examples of what are called CAT tools or computer aided translation tools. As the name suggests, these tools are designed to facilitate the translation process by keeping track of the translated segments – which usually refers to sentences – and letting the user know when a segment is similar to one that has already been translated. Another key term in this process is translation memory. Translation memories can be explained as databases for translated segments, as they contain both the source segments and the target segment as well as often some additional information, including the translation date and time, languages, and possibly other information, like the project name.

Even though CAT tools use and generate translation memories, the latter are usually not integrated in the former, which makes it possible to share translation memories between different systems and, more importantly, to adopt another CAT tool without losing the valuable translation database. This feature is also important in making this comparison possible, because it makes it easy to use a single translation memory in all three programs. A single memory helps to eliminate some of the variables that are always present when using different pieces of software. The translation memories used in this thesis are in the standard TMX file format. Some programs, like Trados Studio, convert these to another, proprietary format, but all three support TMX at least to some extent.

The three programs, as mentioned, are SDL Trados Studio, CafeTran, and OmegaT. Out of these, Trados Studio is by far the best known and most widely used (Suppanen 2015, p. 72). It is developed by SDL, a well-known giant in the translation industry, and uses many proprietary technologies, file formats and so on. CafeTran, on the other hand, is developed by a single person and is not that well known. It is, however, a cross-platform program (meaning it works in Windows, Linux, or Mac computers) and has some interesting differences when compared to the more mainstream CAT tools. One of these differences is the ability to fine tune the way the translation memory matching works. This is also tested in this thesis.

OmegaT is the only one of these that is completely free. Like CafeTran, it is also a cross-platform CAT tool and has some interesting features, such as the Tokenized add-on, which is enabled by default (Smolej 2018, D.1). The Tokenizer works by trying to separate the word stem, which makes it possible, at least theoretically, to recognize different forms as same words (ibid.). Another difference has to do with showing matches. Instead of only showing one match and leaving the rest hidden under settings and options, OmegaT shows three different matches at once: a match percentage with tokenizer, a match where the number of matched words is divided by the word count while the tags and numbers are ignored, and a match, which is like the previously mentioned, except that tags and number are not ignored (Smolej 2018, 4.2.2). In addition, OmegaT includes a built-in list of so-called stop words – these are common words which are ignored when computing the match percentage (Alkula 2000, s. 24).

## Defining characteristics of Finnish

This thesis focuses on three key characteristics of Finnish that may cause issues when translating Finnish texts and which help define the language and separate it from other languages. These three are: suffixes, morphophonological alternation, and word order. There are also many other aspects of Finnish that could be analyzed, but this thesis focuses on the aforementioned three, which will be briefly introduced in this chapter.

As a synthetic language, Finnish relies on suffixes for grammatical relations and for forming new words (Korpela 2014). Suffixes are affixes that are added after the stem of the word and thus cannot be easily detected using computational algorithms unlike, for example, prepositions in English. This makes it rather likely that they hinder the recognition of translated words, even if the actual meaning only changes a little. As noted earlier, OmegaT includes a tool that has been added just for this purpose: to recognize the stem of the word and – as a result – to recognize inflected words as perfect matches.

The suffixes are numerous in Finnish: they can be used in the conjugation of verbs and the declension of adjectives, nouns, numerals, and pronouns. The grammatical categories, which can be expressed using suffixes, include for example person, tense, number, mood, and case (VISK § 53). As noted earlier, suffixes can also be used in the creation of new words in a process called derivation (VISK § 155). Derivation means that a new word is created by adding an affix to the stem of an existing word (ibid.). Derived words are more complex than their root words and because derived words can be further derived by adding new affixes, the resulting words can end up being extremely challenging to recognize (ibid.). This so-called chain derivation raises the question whether derived words should be shown as matches for their root words. In my opinion, they should. The reasoning is that many of the derived words have similar meanings to their root words, and any possible matches can only be of use to the translator. Based on all of the above, it is evident that suffixes play a major role in Finnish and the way the translation tools deal with them, could be a great factor in matching accuracy.

Morphophonological alternation is in a way related to suffixes, as they both cause changes within the word. In morphophonological alternation, either a phonetic segment of the stem or a phonetic segment of the suffix changes due to its environment (VISK § 40). Morphophonological alternation is characteristic to Finnish and it can affect both consonants and vowels (Suomi, Toivanen & Ylitalo 2008, p. 4, 9). The consonant alternation is called consonant gradation, and it affects the plosives *p*, *t*, and *k* changing them to *pp*, *tt*, and *kk* in certain environments (ibid.). Other forms of morphophonological alternation include changes to the word stem before suffixes – affecting vowels *a*, *e*, *i*, *ä*, diphthongs, and double vowels – and alterations in derivations (VISK § 45, § 159).

Because Finnish is a synthetic language, the word order is relatively free when compared to languages like English. There are, of course, some restraints: not all words orders are possible and the word order can be influenced by the information structure, for example (VISK § 1366). The referential meaning, which is the meaning referring to things outside of language, is still identical between differently ordered versions of the same sentence (ibid.). Phrases can be divided into two categories based on how free the word order is: grammatically free or grammatically fixed (VISK § 1367).

Grammatically free word order is not restrained by structural matters, such as word classes or clause types (VISK § 1367). In fixed word orders, changes in word order lead to ungrammatical phrases or changes in phrase's referential meaning (ibid.). In addition, word order can be grammatically free but otherwise fixed, if a certain order affects the information structure and discourse function of the phrase (ibid.).

## Method

The analysis is done by creating reference translation memories from three articles written by Maria Österlund (2018a, 2018b & 2018c) and published in *Selkosanomat* website, which publishes texts written in simplified Finnish, and then editing the aforementioned Finnish characteristics – suffixes, morphophonological alternation, and word order – into the texts and comparing how well the programs recognize the similarities in the two versions. Simplified language is preferred, because it is less likely to have many loan words or other extremely complex or long words, which could impact the results in a negative way. The match percentage – which is the level of similarity when comparing a segment, that is to be translated, to its closest match in translation memory – makes it easy to compare the performance of each translation tool and notice which elements seem to be the most difficult ones for these tools to recognize.

As both the amount and the degree of changes varies from text to text, it is not possible to say exactly how the language elements rank against each other in terms of problems caused for CAT tools and translation memories. It is, however, possible to conclude how the

elements generally affect the matching process and whether any of the studied characteristics make any difference or a very noticeable difference in matching accuracy.

## Results

The results are overall quite interesting and raise some questions as well. In suffix test, the average matching accuracy was 89% for Trados, 85% for OmegaT, and 69% for CafeTran. It is evident that adding or removing suffixed does not affect the matching in the first two too much, but the way CafeTran handles those seems to differ. One explaining fact is the program's Prefix matching setting, which is not enabled automatically. This setting treats the selected percentage or fixed length as a prefix, or in the case of Finnish as a word stem, and ignores everything following that. When tested again with this function enabled, the results were much better. The best practical result, 89% accuracy, was achieved using the fixed length of six characters, which is based on the average length of Finnish words minus 2.5 characters for estimated average suffix length (Heikkinen, Lehtinen & Lounela 2001).

The test results for morphophonological alternation were more uniform. The best matching was again achieved using Trados (96% average accuracy), but the other two provided almost identical results: CafeTran with 88% average and OmegaT with 89% average. The higher match percentage is partly explained by the fact that the number of edits was smaller than in the suffix test. In this test, OmegaT produced surprising results in more than one way. The first interesting case is a sentence where a one-letter change in the word stem resulted in a 100% match. This can, however, be explained by the tokenizer detecting one word and the other being included in the stop word list.

Another somewhat puzzling segment is the following: "Luettelossa on esimerkiksi perinteitä, tanssit, seremoinoita, leikit, lauluja, vettä ja ruokalajeja". This segment includes three edited words (underlined), and the default match OmegaT offers is 66%. This can be explained with stop words: if OmegaT treated each edited word as a zero match, the result would be a roughly 73% match. However, the two ignored words, *on* and *ja*, lower the word count just enough for a 66% match. The puzzling part is the fact that the two other matching methods provide better results than the supposedly best tokenizer-based matching, with the tags and numerals counting method being the best of the bunch. The

easiest explanation would be that the testing material includes so-called invisible tags, which cannot be seen but which may affect the translation results. This is, however, very unlikely for two reasons. Firstly, both the edited and the unedited test materials have been comprised using plain text files and two different character encodings were tested to make sure it did not cause these issues. Secondly, OmegaT has a build-in tool to remove all tags, but even it made no difference. In the end, this result remained a mystery that would require thorough analysis to fully unravel.

Word order is the third and final test segment. This had the potential to provide the most interesting results, because it is possible to have either very good or very bad results depending on the algorithms used. The results are certainly interesting, and give a hint regarding which language element might the most difficult for CAT tools to handle. Trados, which was up to this point very convincing, fell a little flat when it came to detecting the changes in word order. The average match was 71%, which is clearly below that of the other too tests. OmegaT had similar, though a magnitude bigger, problems, as its average match fell to 59%. Out of the three, only CafeTran seemed to handle word order well, with its average match of 87%.

Out of the three match percentages that OmegaT shows, the one which divides the number of matched words, including tag and numerals, by the total word count provides much better results. And the difference is not small either: 72% average when compared to the default 52%. This is another inexplicable result, for there is only one numeral in the whole text and no tags at all.

## Conclusions

One of the biggest takeaways of these results is the lack of consistency: even if the average match percentages were quite consistent throughout the three tests, there was great variation between individual segments. This applies especially to CafeTran and OmegaT, as Trados was relatively consistent in all three tests – even when it fell a little off in the word order test. Different results are, of course, to be expected when every segment is different in terms of length and the number of edits, but the results did differ between very similar segments as well. This indicates that CAT tools are not yet as consistent as possible,

at least when it comes to languages like Finnish. Some of this inconsistency could be explained by the test material, but a more uniform material would also be farther from natural language and would not fit this particular thesis.

The average match percentages across all tests were as follows: Trados 85%, CafeTran 81%, and OmegaT 78%. Still, it cannot be said that language elements are handled in an identical manner in every program. Both Trados and OmegaT struggled with word order, and CafeTran could not handle suffixes well with its default settings but, on the other hand, performed excellently in the word order test. This suggest that at least word order is handled differently in different programs and also that it is a prime candidate for the most difficult aspect of Finnish when it comes to translation memories and CAT tools – though more research is definitely needed to confirm that.

Another interesting observation relates to the effect of customizability. Two of the tested CAT tools include options other than default ones, which help to provide significantly better matching results. Therefore, it is in translators' best interest to familiarize themselves with the program of their choosing and its various settings and to try and find the ones that work best for them. Further research is necessary to keep up with the ever-growing popularity and availability of CAT tools and translation memories. The focus could be on the more technical side of these tools or even on the matching peculiarities highlighted in this thesis.