

Kuvakielen käsittely transformer-pohjaisissa kielimalleissa

Tietotekniikka
Tietotekniikan laitos, Teknillinen tiedekunta
Kandidaatintutkielma

Laatija:
Heidi Korkeamäki

Joulukuu 2024

Kandidaatintutkielma
Tietotekniikan laitos, Teknillinen tiedekunta
Turun yliopisto

Tutkinto-ohjelma: Tietotekniikka

Tekijä: Heidi Korkeamäki

Otsikko: Kuvakielen käsittely transformer-pohjaisissa kielimalleissa

Sivumäärä: 24 sivua

Päivämäärä: Joulukuu 2024

Kuvakielellä tarkoitetaan kieltä, joka hyödyntää kielikuvia, esimerkiksi metaforia, idiomeja tai metonymiaa. Kuvakieltä käytetään yleisesti kuvaamaan muun muassa abstrakteja konsepteja ja rikastuttamaan viestintää. Ihminen ymmärtää kielikuvia kontekstin ja kulttuurisen kokemuksen perusteella. Tämän tutkielman tavoitteena on tutkia, miten luonnollisen kielen käsittelyn työkaluina käytettävät, transformer-arkkitehtuurille rakentuvat kielimallit erottavat kuvakielisen ilmaisen kirjaimellisesta ja millaisia ongelmia kuvakielen käsittelyyn liittyy. Kuvakielen käsitteleminen on merkittävässä roolissa esimerkiksi automaattisessa tekstin kääntämisessä, tekstin generoinnissa ja tekstin sävyn tunnistamisessa. Tutkielman teoriaosuudessa selvitetään mitä kielimallit ovat ja miten ne koulutetaan, miten transformer-arkkitehtuuri rakentuu ja mitä kuvakielellä tässä tutkielmassa tarkoitetaan. Tutkimusmenetelmänä on käytetty kirjallisuuskatsausta.

Tulokset osoittavat, että transformer-pohjaisten kielimallien kyky käsitellä kuvakieltä perustuu mallin laajalle esikoulutukselle ja laadukkaalla aineistolla tehtävälle hienosäädölle. Esikoulutuksessa malli oppii kielen keskeisiä rakenteita. Kuvakielen käsittelyn tehtäviin malli opetetaan hienosäätämällä sitä runsaasti kuvakielisiä ilmauksia ja niiden kirjaimellisia vastineita sisältävällä aineistolla. Keskeisessä roolissa kuvakielen käsittelyssä ovat myös transformerin itsehuomiomekanismi, kontekstisidonnaisten merkitysten muodostaminen sanoille ja transformer-mallien kyky käsitellä pitkiä aineistoja kerralla.

Transformer-pohjaiset kielimallit ovat ottaneet suuria kehitysaskkeleita, mutta kuvakielen käsittely ja etenkin kuvakielen tuottaminen ovat edelleen haasteellista tehtäviä kielimallille. Keskeisiä haasteita ovat tuotettujen kielikuvien heikko osuvuus, mallien hienosäätämiseen tarvittavien laadukkaiden datasettien vähyys ja kuvakielen kulttuurisidonnaisuutta ajatellen datasettien eri kielivariaatioiden vähyys.

Asiasanat: kielimalli, kuvakieli, metafora, transformer, NLP

Sisällys

1	Johdanto	1
2	Keskeisiä käsitteitä	4
2.1	Kielimalli	4
2.2	Transformer-arkkitehtuuri	7
2.3	Kielimallin kouluttaminen	11
2.4	Kuvakieli	13
3	Kuvakielen käsittely kielimalleissa	15
3.1	Kuvakielen käsittely	15
3.2	Kuvakielen käsittelyn onnistumisia ja ongelmia	18
4	Pohdinta	21
5	Yhteenveto	23
	Lähteet	25

1 Johdanto

“*You shall know a word by the company it keeps!*”

– J.R. Firth, *A Synopsis of Linguistic Theory*, 1957

Kuvakieli tarkoittaa kielikuvia, esimerkiksi metaforia tai idiomeja, hyödyntävää kieltä, jossa sanoja tai ilmauksia käytetään ei-kirjaimellisessa merkityksessä. Aika lentää, elämä on yhtä vuoristorataa ja joku vetää vielä matonkin jalkojen alta. Kuvakielisillä sanoilla ja ilmauksilla voidaan värittää puhetta ja tekstiä, mutta niillä on kielessä myös syvempi ja tärkeämpi merkitys. Teoksessaan *Metaphors We Live By*, Lakoff ja Johnson (1980) väittävät, että metafora ulottuu pelkkää koristeellista kieltä syvemmälle ihmisen ajatteluun ja käsitejärjestelmään. Abstraktien asioiden kuvaamisessa metaforan ja kuvakielen käyttö on lähestulkoon välttämätöntä (Harvilahti ym., 1992; Lakoff & Johnson, 1980).

Vaikka perinteisesti metaforaa ja muita kielikuvia on pidetty lähinnä koristeellisina tai runollisina elementteinä, tutkimustieto osoittaa, että metaforan käyttö on varsin yleistä jokapäiväisessä kommunikaatiossa (Bowdle & Gentner, 2005). Metaforan yleisyydestä kertoo esimerkiksi Steen ym.:n (2010) arvio, jonka mukaan englannin kielessä metaforisesti käytettyjä sanoja esiintyy keskimäärin seitsemän ja puolen sanan välein. Suomen kielestä vastaavaa lukua ei ole saatavilla, mutta metaforan rooli kielenkäytössä on silti kiistämätön¹.

Ihminen ymmärtää metaforat kontekstin ja kulttuurisen kokemuksen perusteella (esim. Mao ym., 2024). Glucksbergin (2001) mukaan metaforisten ilmausten ymmärtäminen kontekstissa ei ole ihmiselle kirjaimellista kieltä vaikeampaa, mutta miten kone selviää tehtävästä? Kuvakielen automaattinen tunnistaminen, tulkitseminen, kääntäminen ja tuottaminen ovat keskeisiä haasteita luonnollisen kielen käsittelyssä (Ge ym., 2023). Kuvakielen käsitteleminen on merkittävässä roolissa esimerkiksi automaattisessa tekstin kääntämisessä ja tuottamisessa, tai tekstin sävyn tunnistamisessa, eli sentimenttianalyysissä (Mao ym., 2022).

Luonnollisen kielen käsittelyssä työkaluina käytetään kielimalleja, jotka pystytään kouluttamaan erilaisiin tehtäviin. Tässä tutkielmassa selvitetään, miten transformer-arkkitehtuurille pohjautuva kielimalli rakentuu ja kuinka kielimalli oppii erottamaan kuvakieliset sanat ja sanonnat kirjaimellisista.

¹ Harvilahti ym.:n (1992) toimittama artikkelikokoelma kertoo kattavasti näkemyksiä metaforasta suomen kielessä.

Tutkimuskysymykset ja menetelmä

Tutkimuskysymykset ovat:

- T1: Miten kielimalli tunnistaa, tulkitsee ja tuottaa kuvakieltä, esimerkiksi metaforia?
- T2: Millaisia ongelmia nykyisillä transformer-pohjaisilla kielimalleilla on kuvakielen tunnistamisessa, tulkitsemisessä ja tuottamisessa?

Tutkielmassa kielimallit rajataan transformer-pohjaisiin, tekstiä tunnistaviin tai generoiviin malleihin. Kuvakielisissä ilmauksissa keskitytään pääasiassa metaforiin, koska metafora vaikuttaa olevan kielikuvista yleisimmin tutkittu verrattuna esimerkiksi metonymiaan tai idiomiin.

Tutkielman menetelmänä on hyödynnetty kirjallisuuskatsausta. Kirjallisuuslähteitä haettiin ensin neljästä eri tietokannasta: Web of Science, ProQuest, Science Direct ja ACM Digital Library. Hakusanojen kieleksi valittiin englanti, jotta relevantteja tuloksia saataisiin mahdollisimman paljon. Jokaisessa tietokannassa kokeiltiin useita hakulauseita, joista parhaaksi valikoituivat seuraavat:

("metaphor processing" OR "metaphor identif*" OR "metaphor interpre*" OR "metaphor understand*" OR "metaphor detect*" OR "figurative language" OR "non-literal language") AND (LLM OR "language model*" OR "deep learning") AND (NLP OR "natural language processing")

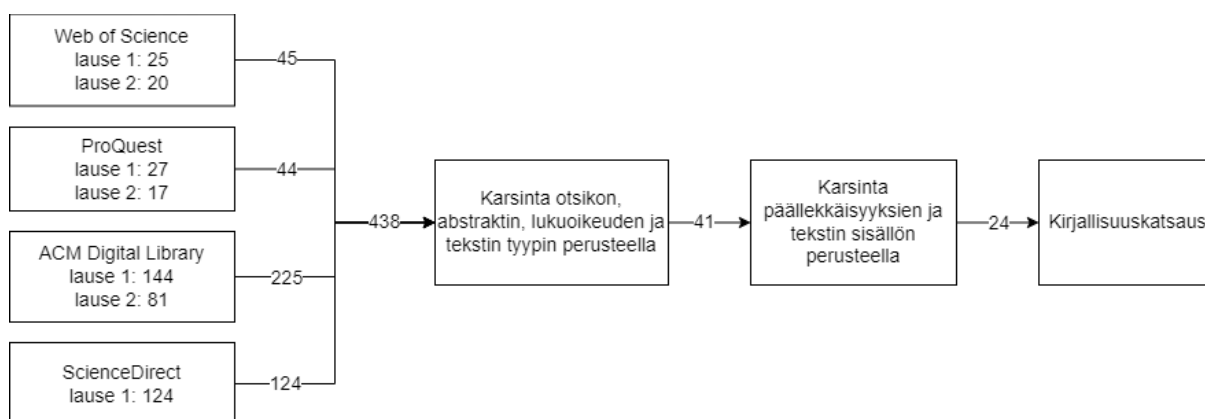
("metaphor generat*" OR "metaphor interpret*" OR "figurative language generat*" OR "figurative language interpret*" OR "non-literal language generat*" OR "non-literal language interpret*") AND (LLM OR "language model" OR transformer OR "deep learning" OR computat* OR automat*)

ja Science Direct -tietokannassa

("metaphor processing" OR "metaphor identification" OR "metaphor interpretation" OR "figurative language" OR "non-literal language") AND (LLM OR "language model" OR "deep learning").

Science Direct -tietokannassa asteriskin käyttö katkaisumerkkinä ei ollut sallittua, ja hakusanojen yhdistämiseen käytettyjä Boolean-operaattoreita sallittiin enintään kahdeksan, jonka vuosi ensimmäistä hakulauseetta lyhennettiin ja katkaistuista sanoista valittiin mahdollisimman perusmuotoinen sana. Toisella hakulauseella Science Direct -tietokannasta ei löytynyt relevantteja tuloksia.

Alan nopean kehityksen vuoksi tuloksia haettiin vuosilta 2020–2024. Yhteensä hakutuloksia oli 438, joista lopulliseen kirjallisuuskatsaukseen valikoitui 24. Ensimmäisellä valintakierroksella tuloksista karsittiin tutkielmat ja lähteet, jotka eivät otsikon ja abstraktin perusteella käsitelleet tutkimuksen aihetta tutkimuskysymyksissä määritellystä näkökulmasta. Lisäksi karsittiin lähteet, joihin ei ollut Turun yliopiston käyttäjätunnuksilla lukuoikeutta. Jäljelle jääneet lähteet käytiin läpi tarkemmin, ja joukosta karsittiin pois eri tietokannoista löytyneet kaksoiskappaleet ja tekstit, joissa ei käsitelty tutkielman tutkimuskysymyksiin liittyviä aiheita riittävällä tarkkuudella. Lopulta kirjallisuuskatsaukseen valikoitui 24 lähdettä, joiden perusteella tutkimuskysymyksiin pyritään vastaamaan. Kirjallisuuskatsauksen avulla kerättyjen lähteiden lisäksi on käytetty täydentäviä lähteitä, joita on löytynyt esimerkiksi kirjallisuuskatsauksen artikkelien lähdeluetteloista sekä aihekohtaisesti hakukoneita ja kirjastoja hyödyntämällä.



Kuva 1 Tutkielman lähteiden hakuprosessi.

Tutkielma jakautuu viiteen lukuun. Johdantoluvun jälkeen tutustutaan tutkielmassa käytettävään sanastoon ja käsitteisiin keskittyen transformer-arkkitehtuuria hyödyntäviin kielimalleihin ja niiden toimintaperiaatteisiin. Kolmannessa luvussa esitellään kirjallisuuskatsauksen tulokset. Neljännessä luvussa keskitytään aiheen pohdintaan, ja viides ja viimeinen luku on yhteenveto.

2 Keskeisiä käsitteitä

2.1 Kielimalli

”Harakka huttua keittää –”, mikä sana tulee seuraavaksi? Suomalaisia lastenloruja tuntevalle tehtävä on varsin helppo, mutta miten tietokone selviää vastaavasta pulmasta? Kielimallit ovat tämän kaltaisiin tehtäviin erikoistuneita kokonaisuuksia, jotka pyrkivät sanojen tai niiden osien todennäköisyyksiä laskemalla ratkaisemaan, mikä on paras mahdollinen seuraava sana, tai millainen lause todennäköisimmin esiintyy tekstissä (Jurafsky & Martin, 2024). Jurafskyn ja Martinin (2024) mukaan monet luonnollisen kielen käsittelyyn liittyvät tehtävät voidaan nähdä sanojen ennustamistehtävinä. Näin ollen, erilaisia kielimalleja voidaan koulutusaineistoa muokkaamalla kouluttaa moniin kielen käsittelyn tehtäviin. Malleja voidaan hyödyntää esimerkiksi konekääntämisessä, tekstin tuottamisessa tai tekstin sävyn tunnistamisessa, eli sentimenttianalysissä (Jurafsky & Martin, 2024).

Ensimmäisiä alkeellisia kielimalleja kehitettiin jo 1950-luvulla (Shannon, 1951), mutta suuren yleisön tietoisuuteen ja arkeen mallit tulivat vasta 2010-luvulla virtuaalisten assistenttien, kuten Sirin ja Alexan myötä. Kielimallien kehitys otti valtavan askeleen vuoden 2022 syksyllä, kun teknologiayritys Open AI julkaisi GPT-kielimallia (Generative Pre-trained Transformer) hyödyntävän ChatGPT-palvelun jokaisen ulottuville (OpenAI, 2022).

Yksinkertaisesti ajateltuna kielimalli on todennäköisyyslaskentaan erikoistunut laskentaväline. Sanojen esiintymistodennäköisyyksien arviointia kielimalli oppii sen koulutuksen aikana, jossa mallille syötetään laajoja tekstiaineistoja analysoitavaksi. Esimerkiksi BERT-kielimallin (Bidirectional Encoder Representations from Transformers) kouluttamiseen on käytetty 800 miljoonan sanan BookCorpus-datasettiä ja 2 500 miljoonan sanan aineistoa englanninkielisestä Wikipediasta (Tuffery, 2022). Koulutuksen aikana kielimalli opetetaan minimoimaan virheet seuraavan sanan tai puuttuvan sanan ennustamistehtävässä (Jurafsky & Martin, 2024). Kielimallien kouluttamiseen palataan tarkemmin tutkielman luvussa 2.3.

Havainnollistavana esimerkkinä sanojen ennustamistehtävästä, ChatGPT-palvelu antaa taulukon 1 mukaiset sanat todennäköisimpinä seuraavina sanoina, kun lauseen alkuna ja kielimallin syötteenä on ”harakka huttua” tai ”harakka huttua keittää”. Esimerkin perusteella GPT-mallin mukaan ”harakka huttua” jatkuu 70 % todennäköisyydellä sanalla ”keittää”, jota seuraa 35 % todennäköisyydellä sana ”ja”.

Taulukko 1 Esimerkki seuraavien sanojen todennäköisyyksistä, kun lauseen alkuna on "harakka huttua" tai "harakka huttua keittää".

Harakka huttua...		Harakka huttua keittää...	
Seuraava sana	Todennäköisyys	Seuraava sana	Todennäköisyys
keittää	0,7	ja	0,35
söi	0,1	kun	0,2
koki	0,05	, (pilkku)	0,15
joi	0,05	niin	0,1
maistoi	0,03	sillä	0,05
pehmentää	0,02	vaan	0,04
tekee	0,02	se	0,03
valmisti	0,01	tahi	0,03
laittaa	0,01	että	0,025
haudutti	0,01	mutta	0,025

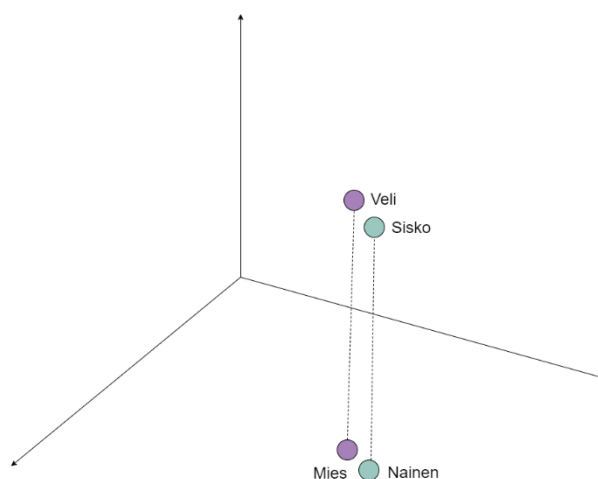
Tokenisointi ja sanaupotukset

Kielimalli ei välttämättä käsittele sille syötteenä annetun tekstin osia kokonaisina sanoina, vaan voi jakaa, eli tokenisoida (engl. tokenise), syötteen eri mittaisiksi merkkijonoiksi ja välimerkeiksi (engl. token) (Jurafsky & Martin, 2024). Näihin yksiköihin viitataan tässä tutkielmassa jatkossa termillä sane. Saneet kielimalli muuttaa laskettavaan numeeriseen vektorimuotoon, jota kutsutaan termillä sanaupotus (engl. (word) embedding) (Jurafsky & Martin, 2024). Kuvitteellinen esimerkki kymmenulotteisista sanaupotuksista esitetään taulukossa 2. Tyypillisesti vektoreilla on joitain satoja ulottuvuuksia. Eri saneiden sanaupotusvektoreiden arvot kielimalli oppii sen koulutuksen aikana, mutta myös valmiiksi muodostettuja vektoreita on mahdollista käyttää (Russell & Norvig, 2021).

Taulukko 2 Kuvitteellinen esimerkki 10-ulotteisista sanaupotuksista.

Sana	Sanaupotus
Harakka	[1.5, 0.3, -1.2, 2.1, -0.7, 0.5, 0.1, -1.0, 0.8, 0.2]
Varis	[1.3, 0.5, -1.5, 1.8, -0.9, 0.7, 0.3, -0.8, 0.9, 0.4]
Mies	[-1.8, 0.6, 2.3, -1.1, 1.7, 0.2, -0.5, 1.4, -0.3, 0.9]
Nainen	[-2.0, 0.8, 2.1, -1.3, 1.9, 0.1, -0.6, 1.2, -0.2, 0.7]
Veli	[0.9, -1.2, 1.8, 0.4, -0.8, 1.3, 2.1, -1.0, 0.6, 0.5]
Sisko	[0.7, -1.0, 2.0, 0.2, -0.6, 1.1, 2.3, -0.9, 0.4, 0.8]
Lintu	[2.3, 1.5, -2.1, 1.9, -1.6, 0.8, 0.5, -2.3, 1.1, 1.0]

Sanaupotusvektori kuvaa saneen merkitystä. Merkitykseltään samankaltaisilla saneilla myös sanaupotukset ovat lähellä toisiaan, ja koordinaatistoon asetettuna vektorit asettuisivat toistensa läheisyyteen (Russell & Norvig, 2021). Kuvassa 2 esitetään, kuinka sanoja *veli*, *sisko*, *mies* ja *nainen* kuvaavat sanaupotusvektorit voisivat asettua koordinaatistoon. Vektorin *nainen* etäisyys vektorista *sisko* on suurin piirtein saman pituinen kuin vektorin *mies* etäisyys vektorista *veli*.



Kuva 2 Esimerkki merkitykseltään samankaltaisten sanaupotusvektoreiden asettumisesta koordinaatistoon.

Neuroverkot ja kielimallin toiminta

Modernit kielimallit sisältävät syviä neuroverkkoja, jotka ovat syötekerroksesta, piilokerroksista ja ulostulokerroksesta rakentuvia matemaattisia malleja (esim. Jurafsky & Martin, 2024; Russell & Norvig, 2021; Tuominen & Neittaanmäki, 2019). Neuroverkon piilokerrokset sisältävät painokertoimia, eli mallin parametrejä, joiden avulla verkolle syötteenä annettusta tiedosta lasketaan kerros kerrokselta aina tarkempi ennuste halutusta tulosteesta.

Tokenisoinnin, sanaupotusten ja kielimallin sisältämien parametrien avulla malli laskee lopulta todennäköisyysjakauman, jonka perusteella malli ennustaa käsittelyssä olevaan tekstiin lisättävän tai siitä puuttuvan sanan. Kielimallin toimintaan vaikuttavat parametrit malli oppii koulutuksen aikana. (esim. Jurafsky & Martin, 2024.) Kielimallin kouluttamisen vaiheisiin palataan tutkielman luvussa 2.3. Nykyisten suurien kielimallien (engl. large language model, LLM) taustalla hyödynnetään lähes poikkeuksetta transformer-neuroverkkoa, jonka nähdään muuttaneen puheen ja kielen koneellisen käsittelyn täydellisesti (esim. Jurafsky & Martin, 2024; Russell & Norvig, 2021). Seuraavaksi tutustutaan transformer-arkkitehtuurin toimintaan.

2.2 Transformer-arkkitehtuuri

Luonnollisen kielen kääntämisen tarkoituksiin alun perin kehitetty transformer-arkkitehtuuri esiteltiin ensimmäisen kerran vuonna 2017 julkaistussa artikkelissa *Attention is All You Need* (Vaswani ym., 2017). Transformeri on neuroverkko, jonka tehtävänä on muunnella sanaupotuksia niin, että upotukset vastaavat kontekstinsa mukaista merkitystä mahdollisimman tarkasti. Jos kielimalli käyttää valmiiksi muodostettuja, staattisia sanaupotuksia, aluksi kahdella homonyymilla, esimerkiksi kuusi (numero) ja kuusi (puu), on samanlainen sanaupotus. Transformerin läpi kuljettuaan sanaupotusten tulisi vastata sanan merkitystä ympäröivässä kontekstissa.

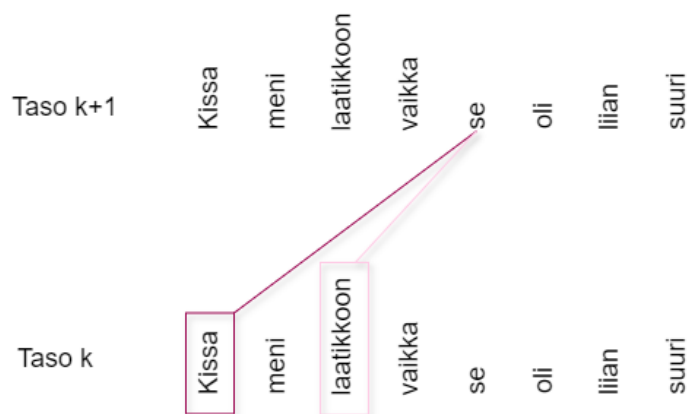
Alkuperäinen transformer-arkkitehtuuri koostuu kooderista (engl. encoder) ja koodinpurkajasta (engl. decoder). Kooderin päätehtävänä on prosessoida sille annettu syöte kontekstiin sopiviksi sanaupotuksiksi. Koodinpurkaja vuorostaan käyttää kooderin ulostuloa ja mallin jo generoimaa tulostetta toivotun tulosteen generoimiseen. Sanaupotusten muuntamiseen kontekstiin paremmin sopiviksi transformer-arkkitehtuuri hyödyntää itsehuomiomekanismia (engl. self-attention), joka on yksi transformereiden menestyksen kulmakivistä. (Vaswani ym., 2017.)

Transformerit ovat kehittyneet sitten ensiesittelyn. Nykyisissä kielimalleista esimerkiksi tekstiä ja muuta sisältöä generoiva GPT käyttää pelkästään transformerin koodinpurkuosaa. Mallin tehtävänä on generoida seuraava sane pelkästään edeltävien saneiden perusteella. BERT-mallissa on käytössä transformerin kooderi-osa, joka käsittelee saneita molemminpuolinen konteksti huomioiden. BERT-malli on käytössä erityisesti tekstin ymmärtämisen ja luokittelun tehtävissä. Kooderin ja koodinpurkajan yhdistelmää hyödynnetään edelleen esimerkiksi tekstin kääntämisen tehtävissä. (Jurafsky & Martin, 2024.) Seuraavaksi tarkastellaan tarkemmin transformerin komponentteja ja niiden toimintaa.

Huomiomekanismi

Monet transformer-arkkitehtuuria edeltävät kielimallit kärsivät lyhyestä muistista, ja yrittivät ennustaa seuraavaa sanaa vain muutaman edeltävän sanan perusteella. Nykyisillä malleilla muisti ulottuu pidemmälle sekä menneisyyteen että tulevaisuuteen, ja itsehuomiomekanismin avulla kyetään käsittelemään lauseita, joissa suhteet ulottuvat koko konteksti-ikkunan (eng. context window) leveydelle. (Jurafsky & Martin, 2024.)

Itsehuomiomekanismin tehtävänä on arvioida saneiden suhteita toisiinsa ja liittää ne ympäröivään kontekstiin. Mekanismi laskee, mitä kuhunkin sanaupotukseen tulisi lisätä, jotta sen merkitys olisi kontekstiinsa sopivampi. Transformereissa yksittäisiä itsehuomiomekanismeja on toiminnassa useita rinnakkain (engl. multi-headed self-attention mechanism) (Vaswani ym., 2017). Informaatiota yhdistelemällä mekanismi oppii tunnistamaan saneiden välisiä yhteyksiä. Käytännössä mekanismi laskee painotettuja summia ja pistetuloja, joiden avulla sanaupotuksia säädetään.



Kuva 3 Itsehuomiomekanismi tunnistaa kunkin sanan kontekstin kannalta olennaiset sanat. Kuva mukaillee Uszkoreitin visualisointia (Uszkoreit, 2017).

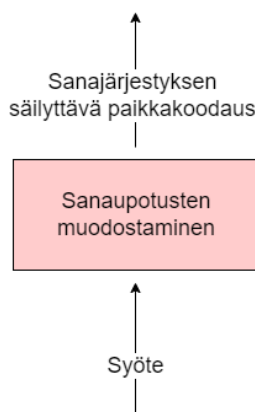
Jokainen sanaupotus toimii huomioprosessin aikana kolmessa eri roolissa. Kysely-roolissa (engl. query) oleva sanaupotus hakee ympäriltään avaimen (engl. key) roolissa olevia upotuksia, jotka olennaisesti liittyisivät kysely-upotukseen. Kuvan 3 esimerkissä sana ”se” liittyy olennaisesti sanaan ”kissa” tai mahdollisesti sanaan ”laatikkoon”. Lisäksi jokaisella saneella on arvovektori (engl. value), jota käytetään itsehuomiomekanismin tulosten laskemiseen. Kyselyvektorin ja avainvektorin pistetulosta saadaan huomiopisteet, joiden perusteella itsehuomiomekanismi pystyy arvioimaan, kuinka paljon kuhunkin sanaupotukseen tulisi kiinnittää huomiota. Huomiomekanismi siis selvittää, minkä saneiden kohdalla kyselyvektori ja avainvektori ovat yhdensuuntaisia, ja siten liittyvät läheisesti toisiinsa. Arvoja normalisoidaan softmax-funktiolla. Sanaupotukset päivitetään painottamalla jokaisen sanan arvovektori huomiopisteillä. Tuloksena saatu vektori lisätään alkuperäiseen sanaupotukseen. (Jurafsky & Martin, 2024; Vaswani ym., 2017.)

Kuten edellä mainittiin, huomiomekanismeja toimii useita rinnakkain. Jokaisella rinnakkaisella huomiomekanismilla on omat kysely-, avain- ja arvomatriisit, joilla laskutoimituksia suoritetaan. Jokainen rinnakkainen huomiomekanismi ehdottaa omia

muutoksiaan sanaupotuksiin, jotka lasketaan yhteen ja lisätään alkuperäiseen upotukseen. Näin saneen kontekstia tarkastellaan monesta eri näkökulmasta. Itsehuomiomekanismin lisäksi transformerin koodinpurkajassa hyödynnetään ristihuomiomekanismia (engl. cross attention), joka itsehuomion tavoin rakentaa kontekstisidonnaista kuvaa saneiden merkityksistä. Ristihuomio mahdollistaa esimerkiksi alkuperäisen tekstin ja sen käännöksen osien yhdistämisen toisiinsa. (Jurafsky & Martin, 2024.)

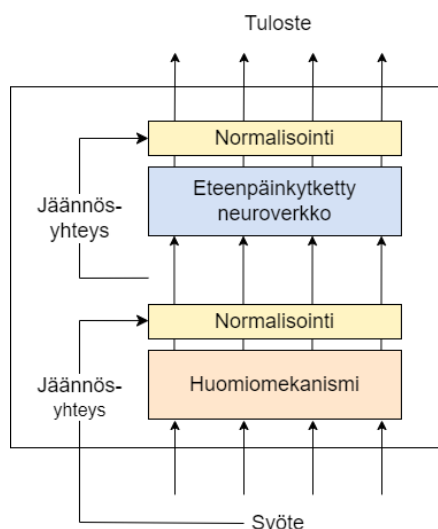
Kooderi

Transformerin kooderi-osan ensimmäinen peruskomponentti muuttaa syötteenä annetut saneet vektorimuotoisiksi sanaupotuksiksi ja lisää sanaupotuksiin paikkakoodauksen (engl. positional encoding), jonka avulla transformeri säilyttää sanajärjestyksen ja tunnistaa saneiden sijainnit ympäröivässä tekstissä (kuva 4).



Kuva 4 Transformer-arkkitehtuurissa syöte muunnetaan sanaupotuksiksi ja jokaiselle saneelle asetetaan sanajärjestyksen säilyttävä paikkakoodaus.

Kooderin toinen pääkomponentti (kuva 5) koostuu useista peräkkäisistä transformer-lohkoista, jotka vuorostaan rakentuvat huomiomekanismista, eteenpäinkytketystä neuroverkosta (engl. Feedforward neural network, FNN) ja normalisoinnista vastaavista kerroksista. Huomiomekanismia ja eteenpäinkytkettyä neuroverkkoa ympäröivät jäännösyhteydet (engl. residual connection) pitävät huolen siitä, että transformerin edeltävissä kerroksissa tehdyt muunnokset säilyvät kerroksesta toiseen: edellisten kerrosten työ ei mene matkan varrella hukkaan, vaan sen päälle rakennetaan aina uutta. Alkuperäisessä transformer-arkkitehtuurissa identtisiä transformer-lohkoja oli kuusi kappaletta. Nykyään lohkojen määrä vaihtelee kielimallin tehtävien mukaan. Esimerkiksi BERT-mallissa kerroksia on 12–24. (Devlin ym., 2018; Vaswani ym., 2017.)



Kuva 5 Transformer-arkkitehtuurin kooderi-osa.

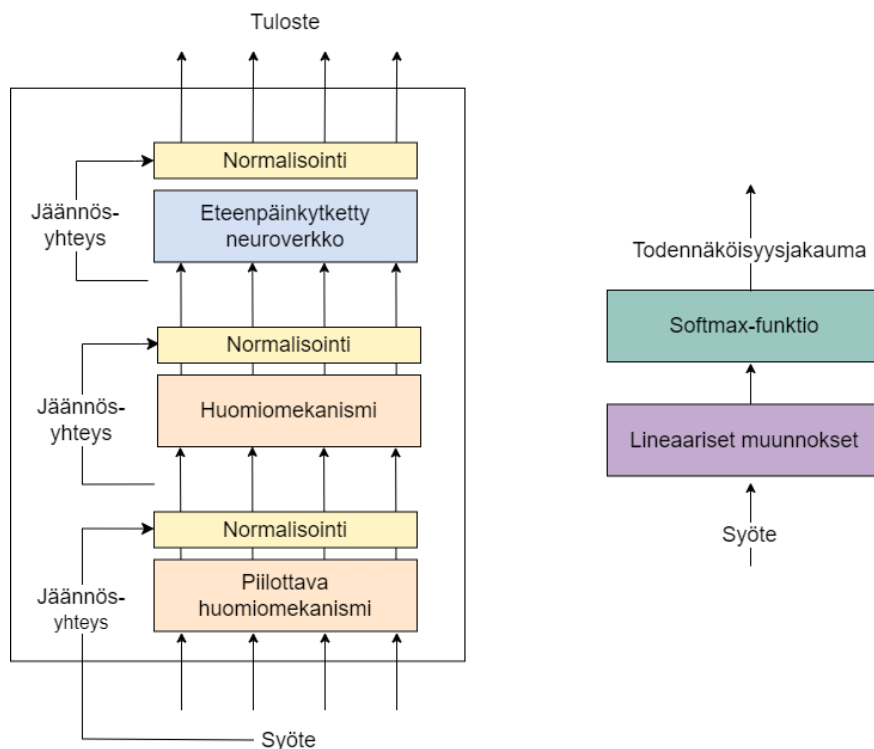
Yksittäinen transformer-lohko muuntaa (engl. map) edellisen komponentin tuottaman sanaopetusvektorin seuraavan komponentin syötevektoriksi. Peräkkäiset transformer-lohkot käsittelevät kerralla kokonaisen konteksti-ikkunan. Kerroksissa muunnokset ovat käytännössä matriisien ja vektorien välisiä pistetuloja, vektoreiden lukuja kerrotaan koulutuksen aikana opituista parametreista koostuvien matriisien kertoimilla. (Jurafsky & Martin, 2024.)

Koodinpurkaja

Kuten kooderissa, myös koodinpurkajassa (kuva 6) on useita identtisiä transformer-lohkoja. Koodinpurkajan lohkot sisältävät huomiomekanismin, normalisointikerrosten, eteenpäinkytketyn neuroverkon ja jäännösyhteyksien lisäksi piilottavan huomiomekanismin, jonka tehtävänä on varmistaa, että sanoja generoidessaan kielimalli ei huomioi myöhemmin tulevia sanoja ennen aikojaan, vaan keskittyy generoinnin kohteena olevaan sanan ennustamiseen sitä edeltävien sanojen perusteella. (Vaswani ym., 2017.)

Seuraavan tai puuttuvan sanan todennäköisyysjakauman muodostamiseksi kooderista ja koodinpurkajasta saatu tuloste käy läpi vielä lineaarisen muunnoksen, jonka jälkeen lopullinen todennäköisyysjakauma muodostetaan softmax-funktiolla.

Todennäköisyysjakaumassa jokaisen arvon tulee olla nollan ja yhden väliltä, ja arvojen summan tulee olla yksi. Aikaisempien kerrosten tuottamat luvut eivät kuitenkaan ole todennäköisyysjakaumaan kelpaavia. Softmax-funktio normalisoi syötteenään saamansa vektorin listaksi, jonka osien summa on yksi. Funktio muuttaa suurimmat arvot lähemmäs yhtä ja pienimmät lähemmäs nollaa. Tuloksena saadaan todennäköisyysjakauma. (Vaswani ym., 2017.)



Kuva 6 Transformer-arkkitehtuurin koodinpurkuosa (vas.) ja todennäköisyysjakauman muodostamisesta vastaava osa (oik.).

2.3 Kielimallin kouluttaminen

Kielimallin kouluttaminen on syväoppimiseen perustuva prosessi, jossa malli opetetaan käsittelemään luonnollista kieltä. Syväoppiminen puolestaan on koneoppimisen suuntaus, joka perustuu syviin, vähintään kaksi piilokerrosta sisältäviin neuroverkkoihin (esim. Goodfellow ym., 2016; Russell & Norvig, 2021; Tuffery, 2022). Tässä tutkielmassa koneoppimisen ja syväoppimisen tarkempi kuvailu sivuutetaan², ja keskitytään kielimallien kouluttamiseen yleisemmällä tasolla. Kouluttamisprosessi voidaan jakaa kahteen vaiheeseen: esikoulutukseen (engl. pre-training) ja hienosäätöön (engl. fine-tuning), joihin tutustutaan seuraavissa kappaleissa.

Esikoulutus

Esikoulutusvaiheessa kielimalli oppii kielen yleisiä rakenteita laajojen tekstikorpusten avulla. Koulutusmenetelmänä käytetään usein ohjaamatonta oppimista (engl. unsupervised learning), jossa malli oppii itsenäisesti koulutusdatasta, eikä dataa tarvitse annotoida tai muuten jäsentää

² Lisätietoa kone- ja syväoppimisesta tarjoavat esimerkiksi Goodfellow ym. (2016), Russel ja Norvig (2021) tai Tuffery (2022).

käsityönä etukäteen. Suurten kielimallien kouluttamiseen on käytetty esimerkiksi Wikipedian sisältöä. (Esim. Jurafsky & Martin, 2024; Russell & Norvig, 2021; Tuffery, 2022.)

Koulutusmateriaalina käytettävä aineisto siistitään tarpeen mukaan ja syötetään mallille. Koulutuksen aikana GPT-mallia pyydetään ennustamaan seuraava sana. BERT-mallin kouluttamisessa syötteestä piilotetaan sanoja, ja malli koulutetaan ennustamaan puuttuvat sanat. Malli opetetaan minimoimaan virheet seuraavan tai puuttuvan sanan ennustustehtävässä. Virhettä mitataan häviöfunktion avulla (engl. loss function) ja gradientteja laskemalla määritellään, miten kielimallin parametrejä tulee säätää virheiden minimoimiseksi. (Jurafsky & Martin, 2024.)

Koulutuksen jälkeen mallin toimintaa voidaan kokeilla testausaineistolla, johon kielimalli ei ole aikaisemmin päässyt tutustumaan. Näin varmistetaan, että mallin parametrien säätämisessä on onnistuttu ja malli kykenee yleistämään oppimansa, eikä toista koulutusaineiston sisältöä. Esikouluttaminen monipuolisella ja laajalla korpuksella opettaa mallille yleispätevää tietoa kielen rakenteista, syntaksista ja riippuvuuksista, joita malli pystyy soveltamaan sille annetuissa tehtävissä (Radford ym., 2018).

Hienosäätö

Esikoulutuksen jälkeen kielimalli voidaan hienosäätää suorittamaan tiettyä tehtävää, esimerkiksi tekstin kääntämistä tai kielikuvien tunnistamista. Tämä vaihe voidaan rinnastaa ihmisen täydennyskoulutukseen tai ammatilliseen erikoitumiseen.

Hienosäätöprosessissa käytetään tehtäväkohtaista, esikoulutusta huomattavasti suppeampaa aineistoa, joka on usein ihmisten manuaalisesti annotoimaa, tai muulla tavoin tarkemmin tiettyä tehtävää varten kuratoitua. Tehtäväkohtainen koulutusaineisto syötetään esikoulutusvaiheen tapaan mallille, ja malli säätää parametrejään edelleen tarkemmiksi kyetäkseen suorittamaan tiettyä tehtävää. Esikoulutuksesta poiketen hienosäätövaiheessa ei enää ole tarpeen säätää kaikkia mallin parametrejä. Hienosäätäminen voidaan tehdä esimerkiksi ensin opettamalla mallille tietyn alan sanastoa, ja seuraavaksi opettamalla malli tiettyyn tehtävään, esimerkiksi vastaamaan kysymyksiin. (Russell & Norvig, 2021.)

Keskusteluun kykenevä kielimalli, esimerkiksi ChatGPT tarvitsee koulutusvaiheessa lisäksi esimerkkejä siitä, kuinka kysymyksiin vastataan. Kielimalli siis koulutetaan ennustamaan sana kerrallaan, kuinka syötteenä saatuun kysymykseen tulisi vastata (engl. Reinforcement learning with human feedback). (OpenAI, 2022.)

Datsettien tuottaminen

Esikoulutetun kielimallin hienosäätämiseen tarvittavia tehtäväkohtaisia datasettejä voidaan tuottaa manuaalisesti asiantuntijoiden käsityönä tai esimerkiksi joukkoistamalla (engl. crowd sourcing), jossa datasetin laatimiseen valjastetaan tarvittava määrä henkilöitä.

Joukkoistamalla tuotettujen datasettien laatua voidaan tarvittaessa tarkastaa asiantuntijoiden toimesta. (Chakrabarty ym., 2022; Kabra ym., 2023; E. Liu ym., 2022; Su ym., 2025; Zayed ym., 2020.)

Kielikuvien käsittelyn datasettien laatimisprosessia voidaan automatisoida esimerkiksi hyödyntämällä verkkosanakirjoja ja korpuksia, joista löytyneet kuvakieliset ilmaukset on kielimalleja hyödyntämällä muunneltu kirjaimellisiksi versioiksi. Lopputuloksena saadaan datasetti, jossa on rinnakkain kuvakielinen ilmaus ja sen kirjaimellinen vastine. (Chakrabarty ym., 2020, 2021, 2022; Zayed ym., 2020.)

2.4 Kuvakieli

Kielellä ajatellaan perinteisesti olevan kahdenlaisia merkityksiä: kirjaimellinen merkitys ja kuvainnollinen merkitys. Kirjaimellisen ja kuvainnollisen kielen määritelmistä on monenlaisia näkemyksiä³, mutta tämän tutkielman tarpeisiin kuvakielellä tarkoitetaan kieltä, joka sisältää kielikuvia. Kielikuvia, eli trooppeja, ovat esimerkiksi metafora, metonymia, idiomi, hyperbola ja synekdokee. Kirjaimellinen kieli puolestaan tarkoittaa sanoja ja lauseita, joka voidaan ymmärtää suoraan niiden sanallisessa merkityksessä ilman erillistä tulkintaa. Kuvakielisissä ilmauksissa on yleistä kielen odottamattomuus ja ristiriitaisuus (Potamias ym., 2021), josta voidaan johtaa, että kirjaimellinen kieli puolestaan on verrattain suoraviivaista ja yksiselitteistä.

Taulukko 3 Esimerkkejä kielikuvista.

Kielikuva	Esimerkki
Metafora	Julia on aurinko
Metonymia	Luin tänään Kilpeä
Idiomi	Erkki veti herneet nenäänsä
Hyperbola	Olen pyytänyt ainakin sata kertaa
Synekdokee	On minulla sentään katto pään päällä

³ Ks. esim. Evans & Green, 2006; Gibbs, 1994.

Metaforassa sanaa tai lause kuvaa jotain muuta, kuin mitä se kirjaimellisesti ymmärrettynä tavallisesti kuvaa (Hosiaislouma, 2003): taulukon 3 esimerkissä Julia ei kirjaimellisesti ajateltuna ole aurinko, mutta hänellä on ominaisuuksia, joita myös aurinkoon voidaan liittää. Metonymiassa sana korvataan siihen liittyvällä ilmaisulla (Hosiaislouma, 2003): En kirjaimellisesti lukenut Kilpeä, vaan Volter Kilven tuotantoa. Idiomi on esimerkiksi tietylle kielelle tai alueelle ominainen ilmaus (Hosiaislouma, 2003): herneet nenäänsä vetänyt on suuttunut pikkuasiasta, eikä hänen nenästään kirjaimellisesti löydy herneitä. Hyperbola on liioitteleva ilmaus (Hosiaislouma, 2003): olen pyytänyt sinua viemään roskat ainakin sata kertaa tarkoittaa, että olen todennäköisesti pyytänyt enemmän kuin kerran, etkä vieläkään ole toiminut. Synekdokee kuvaa kokonaisuutta sen osalla (Hosiaislouma, 2003): katto pään päällä ei ole pelkkä katto, vaan sillä kuvataan koko asumusta.

Metafora vaikuttaa olevan kielikuvista eniten tutkittu luonnollisen kielen käsittelyn aihepiirissä, jonka vuoksi myös tässä tutkielmassa painottuu metafora. Mao ym. (2022) huomauttavat, että eivät artikkelissaan erota metaforaa muista kielikuvista, joka on yleinen toimintatapa myös monissa korpuksissa. Metaforan nähdään olevan kenties ainoita tapoja kuvata abstrakteja asioita, ja ihminen myös ajattelee osin metaforisesti (esim. Harvilahti ym., 1992; Lakoff & Johnson, 1980). Metafora ja muut kielikuvat ovat keskeisiä elementtejä ihmisen kielenkäytössä, jonka vuoksi myös luonnollisen kielen käsittelyyn tarkoitettujen järjestelmien tulisi kyetä käsittelemään kuvakieltä.

Kuvakielen automaattinen tunnistaminen, tulkitseminen, kääntäminen ja tuottaminen ovat keskeisiä haasteita luonnollisen kielen käsittelyssä (esim. Ge ym., 2023). Kuvakielen käsitteleminen on merkittävässä roolissa esimerkiksi automaattisessa tekstin kääntämisessä ja tuottamisessa, tai tekstin sävyn tunnistamisessa, eli sentimenttianalyysissä (esim. Mao ym., 2022).

3 Kuvakielen käsittely kielimalleissa

Aristoteleen runousopin mukaan metaforan hallitseminen osoittaa runoilijan lahjakkuuden, sillä sitä ei voi muilta oppia (Aristoteles, 1997). Nykytiedon valossa näin ei välttämättä enää ole, mutta metafora on joka tapauksessa keskeinen keino monimutkaisten konseptien ymmärtämisessä ja ilmaisemisessa. Sen ymmärtämiseen vaaditaan kehittyneitä kognitiivisia kykyjä, kuten esimerkiksi asioiden välisten suhteiden hahmottamista, abstraktia ajattelua ja johtopäätösten tekemistä (esim. Bowdle & Gentner, 2005; Holyoak & Stamenković, 2018).

Verrattuna kirjaimelliseen kieleen, luonnollisen kielen käsittelyyn rakennetut kielimallit eivät saavuta yhtä hyviä tuloksia, kun käsittelyssä on kuvakielistä tekstiä (Potamias ym., 2019). On kuitenkin havaittu, että suuret kielimallit kykenevät tulkitsemaan jopa uusia kaunokirjallisuuden ja runouden metaforia (Ichien ym., 2024). Edellisessä luvussa tutustuttiin kielimallien ja transformereiden toimintaperiaatteisiin. Seuraavaksi tutkitaan, mihin kielimallien kyky käsitellä kuvakieltä perustuu. Tässä tutkielmassa kysymystä tutkitaan pääasiassa metaforan kautta, joka on erilaisista kuvakielistä ilmauksista eniten tutkittu (Lai & Nissim, 2022). Kuvakielen käsittely näyttää kirjallisuuskatsauksen perusteella nojaavan samoihin periaatteisiin kielikuvan tyypistä riippumatta.

3.1 Kuvakielen käsittely

Kuvakielen käsittely jaetaan luonnollisen kielen käsittelyn piirissä yleensä kahteen aliluokkaan: tunnistamiseen (engl. detection tai identification) ja tulkitsemiseen (engl. interpretation tai understanding) (Mao ym., 2024; Yu & Wan, 2019). Kielimallien ottamien suurten harppausten myötä myös kolmas osa-alue, kuvakielen tuottaminen (engl. generation) on saanut enemmän huomiota osakseen. Kolmesta osa-alueesta kielikuvien tunnistaminen on melko laajasti tutkittu (Babieno ym., 2022; Lai ym., 2023; Potamias ym., 2019; Ptiček & Dobša, 2023; Su ym., 2020; Tong ym., 2021). Tulkitsemista ja tuottamista on tutkittu vähemmän, mutta viime vuosina kiinnostus näitäkin osa-alueita kohtaan on kasvanut (Lai & Nissim, 2024; Su ym., 2025; Tong ym., 2021; Yu & Wan, 2019; Zayed ym., 2020). Vaikka automaattinen kielikuvien tuottaminen on jäänyt vähemmälle tutkimukselle, tutkimusta tarvitaan, jotta ihmistasoisen kielen tuottaminen voisi olla mahdollista (Tong ym., 2021).

Kielikuvien tunnistaminen

Metaforan tunnistaminen on luokittelutehtävä, jossa kokonainen lause tai sen sisältäviä sanoja tai saneita luokitellaan merkitykseltään metaforisiksi (Babieno ym., 2022; Ge ym., 2023; Lai ym., 2023; Su ym., 2020). Sama periaate pätee myös muihin kielikuviin.

Su ym. (2020) jaottelevat metaforan tunnistamisen kolmeen päämenetelmään: sanasto- ja sääntöpohjaiset menetelmät, korpuksiin perustuvat tilastolliset mallit ja syväoppimista hyödyntävät mallit. Sanasto- ja sääntöpohjaisissa menetelmissä kielimalli muodostetaan ihmisen määrittelemien, kielen osaamiseen perustuvien sääntöjen avulla. Korpuksiin perustuvat tilastolliset mallit rakentuvat tunnettujen metaforien pohjalle, eivätkä ne siten tunnista harvinaisempia metaforia, joita ei löydy mallin muodostamisessa käytetyistä korpuksista. Syväoppimista hyödyntäviin malleihin kuuluvat esimerkiksi transformer-mallit GPT, BERT ja RoBERTa (A Robustly Optimized BERT Pretraining Approach). (Su ym., 2020.) Mao ym. (2025) toteavat, että esikoulutettu transformer-kielimalli on nykyisin todennäköisimmin käytetty ratkaisu metaforan tunnistamistehtävään.

Tunnistamistehtävää varten koulutettavien kielimallien hienosäätämiseen käytetään datasettejä, joiden sisältämiin tekstiaineistoihin on merkitty, onko kukin sana tai lause merkitykseltään kuvaannollinen vai kirjaimellinen (Ge ym., 2023). Yksi tunnetuimmista kielikuvien käsittelyyn keskittyvistä dataseiteistä englannin kieltä kuvaavasta kansalliskorpuksesta (British National Corpus) kerätty VU Amsterdam -metaforakorpus. Korpus sisältää 10 567 lausetta kielitieteilijöiden manuaalisesti annotoimaa tekstiä, johon on merkitty MIP-prosessia⁴ (Metaphor Identification Procedure) hyödyntäen, onko kunkin sanan merkitys metaforinen vai kirjaimellinen (Steen ym., 2010).

Aghazadehin ym. (2022) tutkimuksen perusteella esikoulutetuilla kielimalleilla metaforan tunnistustehtävä voi olla opittu jo esikoulutusvaiheessa kouluttamiseen käytetyistä laajoista tekstiaineistoista. Esikoulutetut suuret kielimallit eivät tämän tutkimuksen perusteella välttämättä tarvitsisi hienosäätöä metaforan tunnistustehtävän suorittamiseksi.

⁴ Steen ym. (2010) avaa tarkemmin MIP-prosessia. Ptíček ja Dobša (2023) kokoavat yhteen muita käytössä olevia annotointimenetelmiä.

Kielikuvien tulkitseminen

Metaforan tunnistamistehtävällä luokiteltiin lauseet tai yksittäiset sanat tai saneet merkitykseltään metaforisiksi. Metaforan tulkitsemistehtävä vuorostaan liittyy metaforisen lauseen tai sen osan merkityksen ymmärtämiseen (Chakrabarty ym., 2022; Ge ym., 2023; Kabra ym., 2023; E. Liu ym., 2022; Zayed ym., 2020). Metaforaa koskeva määritelmä on yleistettävissä koskemaan myös muita kielikuvia.

Tulkitsemistehtävää voidaan lähestyä Zayed ym.:n (2020) mukaan leksikaalisella korvaamisella (engl. lexical substitution), jossa metaforinen sana tai sanonta korvataan sen kirjaimellisella vastineella, parafrasien tuottamisella (engl. paraphrase generation), jossa kokonainen lause, metaforinen sana tai sanonta mukaan lukien, muutetaan kirjaimellisempaan muotoon, tai määritelmän tuottamisella (engl. definition generation), jossa metaforiselle sanalle tai sanonnalle annetaan kirjaimellinen merkitys. Tong ym. (2021) toteavat, että suurin osa tulkitsemistehtävää koskevasta tutkimuksesta keskittyy parafrasien tuottamiseen.

Mao ym.:n (2025) mukaan myös tulkitsemistehtävässä esikoulutetut kielimallit ovat avainasemassa. Tulkitsemistehtävää varten kielimallia hienosäädetään datasetillä, joka sisältää esimerkiksi kuvakielisten ilmausten ja niiden kirjaimellisten parafrasien pareja. Esimerkiksi Fig-QA-metaforadatasetti sisältää 10 256 ihmisten kirjoittamaa metaforaa ja niiden kirjaimellista parafrasia, datasetin laatimiseen hyödynnettiin joukkoistamista (Liu ym., 2022).

Kielikuvien tuottaminen

Metaforan tuottamisella tai generoimisella tarkoitetaan kirjaimellisen lauseen uudelleenkirjoittamista metaforiseksi (Chakrabarty ym., 2020, 2021; Lai & Nissim, 2022, 2024; Stowe ym., 2021; Su ym., 2025). Yleistettynä kuvakielen generoiminen on annetun tekstin uudelleenmuotoilemista sisältämään haluttu kielikuva, esimerkiksi metafora tai hyperbola, ilman että tekstin alkuperäinen merkitys muuttuu (Lai & Nissim, 2024).

Onnistunut generoiminen sisältää halutun kielikuvan, merkitys ja konteksti säilyy, ja teksti säilyttää sujuvuuden ja luettavuuden.

Tuottamistehtävässä on aiemmin käytetty jäykempiä sääntöihin ja kielelliseen osaamiseen perustuvia lähestymistapoja, mutta viime vuosina transformer-arkkitehtuurin yleistyttyä mallintaminen on siirtynyt neuroverkkoja hyödyntäviin menetelmiin. Hyviä tuloksia on saatu hienosäätämällä ja hyödyntämällä esikoulutettuja malleja (Lai & Nissim, 2022). Lai ym.:n

(2024) mukaan tuottamistehtävän toteuttamiseen on useampia lähestymistapoja: kielimallin kouluttaminen tyhjästä, esikoulutetun kielimallin hienosäätäminen tai kehoitteilla ohjaaminen (engl. prompt learning).

Tulkitsemistehtävän tapaan tuottamistehtävään koulutettavan mallin hienosäätämiseen voidaan käyttää datasettiä, joka sisältää kielikuvien ja niiden kirjaimellisten parafrasien pareja. Datasetti voi sisältää myös selityksen kielikuvan merkityksestä (Lai & Nissim, 2024). Shou ym. (2024) näkevät tulkitsemistehtävän tuottamistehtävän vastakohtana ja kehittivät mallin, joka suoriutuu sekä metaforan tulkitsemistehtävästä että tuottamistehtävästä.

Hienosäätämisen asemesta voidaan myös hyödyntää valmiiksi koulutettua, esimerkiksi yleiseen käyttöön tarkoitettua kielimallia, jota ohjailaan mallille annettavia kehoitteita (engl. prompt) muokkaamalla. Hyvin suunniteltujen ja muotoiltujen kehoitteiden avulla pyritään hyödyntämään mallin esikoulutusvaiheessa oppimia kielen yleisiä rakenteita mahdollisimman syvällisesti (Zhao ym., 2023). Tarpeen mukaan mallille voidaan tarjota myös esimerkkejä toivotun kaltaisesta vastauksesta. Syötteiden muotoiluun käytettäviä menetelmiä ovat zero-shot, jossa mallille annetaan hyvin muotoiltu kehote ilman esimerkkejä, one-shot, jossa mallille annetaan yksi esimerkki tehtävästä, ja few-shot jossa mallille annetaan muutamia esimerkkejä tehtävästä (esim. Jurafsky & Martin, 2024).

3.2 Kuvakielen käsittelyn onnistumisia ja ongelmia

Kaikissa kolmessa kielikuvien käsittelytehtävissä parhaiten toimivat mallit on saatu aikaan hienosäätämällä transformer-arkkitehtuuria hyödyntävää kielimallia (Tong ym., 2021; Lai ym., 2024). Suosittuja malleja vaikuttavat olevan etenkin BERT, RoBERTa ja erilaiset kooderia ja koodinpurkajaa yhdistävät mallit (engl. seq2seq), esimerkiksi BART (Bidirectional and Auto-Regressive Transformers). Transformeria edeltävien mallien haasteina tunnistettiin esimerkiksi lyhyt muisti, joka ei yltänyt muutamaa sanaa kauemmas (Jurafsky & Martin, 2024). Lyhyen muistin seurauksena malli saattoi unohtaa pitkän lauseen alun ennen lauseen loppua, joka haittaa sanojen sitomista kontekstiin. Transformer-mallit kykenevät käsittelemään kokonaisen konteksti-ikkunan. Myös sääntöpohjaisten mallien laaja käsityönä tehtävä määrittelytyö jää transformer-malleissa pois (Su ym., 2020), kun malli oppii kielen rakenteet ja säännöt omatoimisesti laajasta esikoulutusmateriaalista.

BERT-malli onnistuu hyvin kontekstiin sidottujen sanaupotusten luomisessa, ja siten kykenee tunnistamaan sanojen eri merkityksiä kontekstissa. Malli on siten soveltuva mm.

tunnistamistehtävään (Lai ym., 2023; Potamias ym., 2019). RoBERTa-mallia on hyödynnetty tunnistamistehtävän lisäksi myös tulkitsemistehtävässä (Babieno ym., 2022; Chakrabarty ym., 2022; Y. Liu ym., 2019; Su ym., 2020). BART-malli suoriutuu erityisesti kääntämistä ja parafrasien tuottamista vaativista tehtävistä, jolloin se soveltuu hyvin myös kuvakielen tulkitsemiseen ja tuottamiseen (Chakrabarty ym., 2020, 2021; Lai & Nissim, 2022; Stowe ym., 2021; Su ym., 2025). Suuret GPT-mallit pystyvät käsittelemään kuvakieltä ilman erillistä hienosäätämistä niiden monipuolisen ja laajan koulutusdatan vuoksi (Ichien ym., 2024), mutta Mao ym.:n (2025) mukaan kyseiset mallit eivät yllä suorituksessaan erityisesti kuvakielen käsittelyyn hienosäädettyjen mallien tasolle.

Kuten tutkielman luvussa 2.3 kuvattiin, kielimalli esikoulutetaan suurella määrällä dataa ohjaamattoman oppimisen menetelmällä. Malli oppii itsenäisesti kielen perusrakenteet, jonka jälkeen malli voidaan hienosäätää haluttuun tehtävään. Tällä tavoin vältetään mallin kouluttaminen täysin tyhjästä ja minimoidaan myös koulutusdatan esikäsitteily. Potamias ym.:n (2019) mukaan huomiomekanismia hyödyntävät kielimallit vaikuttavat toimivan varhaisempia malleja tehokkaammin. Keskeistä mallien koulutuksen ja etenkin hienosäädön onnistumiselle on tehtävään soveltuvan datasetin käyttäminen koulutusmateriaalina.

Esimerkiksi Lai ja Nissim (2022), Mao ym. (2024), Su ym. (2025) ja Zayed ym. (2020) mainitsevat kuvakieleen keskittyvien datasettien vähyyden keskeisenä ongelmana ja esteenä metaforan tai yleisemmin kuvakielen käsittelemisen kehitykselle. Lai ja Nissim (2022) sekä Su ym. (2025) nostavat datasettien vähyyden lisäksi esiin myös datasettien laadun kuvakielen tuottamisen tarpeisiin. Osasyynä datasettien puutteelle on prosessiin vaadittavien resurssien vähyys: käsin annotoitujen datasettien laatiminen on kallista ja paljon aikaa vievää (Potamias ym., 2019; Su ym., 2020).

Metaforien tuottamiseen tarkoitettut datasetit ovat pääasiassa englanninkielisiä, eikä olemassa olevilla dataseiteillä luonnollisesti voida ratkaista muiden kielten datasettitarpeita (Kabra ym., 2023; Lai & Nissim, 2022). Lai ja Nissim (2022) mainitsevat, että tuottamistehtävän datasettejä on saatavilla esimerkiksi kiinaksi ja saksaksi. Kielikuvien tunnistamisen dataseiteistä muilla kuin englannin kielellä toteutettuja ovat esimerkiksi kiinankielinen hyperboladatasetti HYPO-cn, ja monikielinen ID10M-datasetti, joka on tarkoitettu idiomien tunnistamiseen saksan, italian, espanjan, venäjän ja farsin kielillä (Lai ym., 2023; Tedeschi ym., 2022). Omia datasettejä tarvittaisiin myös lisää eri kielikuville, esimerkiksi hyperbolalle ja metonymialle (Lai & Nissim, 2022). Mao ym. (2024) mainitsevat datasettien ongelmien

lisäksi yhteisten arviointikriteerien puutteen, jonka seurauksena eri kuvakielen käsittelyn menetelmien toimivuutta ei voida arvioida riittävällä tasolla.

Vaikka on todettu, että suuret kielimallit kykenevät tulkitsemaan jopa uusia kaunokirjallisuuden ja runouden metaforia (Ichien ym., 2024), metaforan tuottamisessa ongelmia esiintyy. Su ym. (2024) toteavat, että tuotetut metaforat voivat olla kielimallille annetun syötteen kaltaisia, eikä siten kovin luovia. Toisaalta tuotetut metaforat voivat olla myös liian luovia, jolloin niiden merkitys ei enää aukea niitä lukevalle ihmiselle. Su ym. peräänkuuluttavat lisää tutkimusta etenkin metaforien tuottamisen osa-alueelle.

4 Pohdinta

Tutkielman johdannossa esitetty englantilaisen kielitieteilijä J.R. Firthin lainaus ”you shall know a word by the company it keeps!”, vapaasti suomennettuna ”sanan tuntee seurastaan”, kuvaa oivasti myös kielimallien toimintaa. Transformer-arkkitehtuuria hyödyntävien kielimallien avain menestykseen näyttää laajan koulutusaineiston lisäksi olevan transformerin itsehuomiomekanismi, jonka elämäntehtävänä on löytää sanojen seuralaiset ja niiden perusteella rakentaa kattava kuva sanojen kontekstisidonnaisesta merkityksestä. Kuten on todettu, kuvakieli on syvällä ihmisen ajatusmaailmassa, ja siten myös laajasti edustettuna suurien kielimallien esikoulutuksessa käytetyssä materiaalissa. Synonyymit esiintyvät samanlaisissa ympäristöissä (esim. Harris, 1954), jonka perusteella voisi ajatella, että myös kielikuvien kontekstit vastaavat kirjaimellisten vastineidensa konteksteja.

Kirjallisuuskatsaukseen poimitut artikkelit eivät erikseen nostaneet esiin itsehuomiomekanismin roolia kielikuvien tunnistamisessa, tulkinnassa ja tuottamisessa. Ottaen kuitenkin huomioon itsehuomiomekanismin kyky kerätä tietoa ympäröivästä kontekstista ja liittää kontekstietoa yksittäisiin saneisiin, voidaan johtaa, että kyseisellä mekanismilla täytyy olla vaikutusta siihen, kuinka hyvin kielimalli oppii kuvakieltä käsittelemään. Tong ym. (2021) arvelevat, että olemassa olevien syväoppimismallien piilotilat (engl. hidden states) saattavat jo sisältää tietoa metaforisista merkityksistä.

Kirjallisuuskatsauksen perusteella voidaan todeta, että kuvakielen käsitteleminen tehtävänä ei mallin koulutuksen osalta eroa muista luonnollisen kielen käsittelyn luokittelutehtävistä. Parhaimpiin tuloksiin on päästy esikouluetulla mallilla, joka on hienosäädetty laadukkaalla, tehtäväkohtaisella datasetillä. Kuvakielen käsittelyn tapauksessa datasetit voivat sisältää esimerkkejä kuvakielisen ja sitä vastaavan kirjaimellisen ilmaisun pareista, tai tekstiä, jonka sanat on luokiteltu merkitykseltään kuvaannollisiksi tai kirjaimellisiksi. Keskeisimmät ongelmat toimivien mallien kehittämisessä näyttävät liittyvän kielimallien hienosäätämiseen tarvittaviin datasetteihin.

Mao ym.:n (2025) mukaan nykyiset yleiskäyttöiset suuret kielimallit eivät yllä suorituksessaan erityisesti kuvakielen käsittelyyn hienosäädettyjen mallien tasolle. Laajalla koulutusaineistolla saadaan aikaan monipuolisuutta, mutta samalla mallin kykyä selviytyä erikoistuneista tehtävistä huononee. Pienimuotoisena empiirisenä kokeena esitetään vielä yleismalli ChatGPT:n näkemyksiä kuvakielisistä lauseita. Pyydettyessä tuottamaan

kymmenen esimerkkiä suomenkielisistä kuvakieltä sisältävistä lauseista ja niiden kirjaimellisista vastineista, ChatGPT tuotti taulukossa 4 esitellyn tuloksen. Ichien ym. (2024) ovat todenneet, että GPT-4-kielimalli kykenee tuottamaan hyvinkin järkeviä tulkintoja jopa uusista metaforista. Jää kuitenkin epäselväksi, kuinka malli tähän oikeastaan pystyy. Kyseinen malli on myös kykenevä metaforien tuottamiseen. Kirjoittajat epäilevät kuitenkin, että tuotetut metaforat ovat kielimallin koulutusmateriaalista peräisin olevia metaforia, jotka ovat alun perin ihmisen tuottamia.

Taulukko 4 Esimerkki ChatGPT-palvelun tuottamista kuvakielisistä lauseista ja niiden kirjaimellisista vastineista.

Kuvakielinen lause	Kirjaimellinen vastine
Hän on kultakalan kokoinen.	Hän on hyvin pieni.
Hänellä on kultaiset kädet.	Hän on erittäin taitava käsistään.
Aika lentää kuin siivillä.	Aika menee nopeasti.
Hän on kuin avaruusalus.	Hän on hyvin erikoinen ja outo.
Hän meni kylmää vettä juomaan.	Hän meni huonoon tai epämiellyttävään tilanteeseen.
Tämä on kuin myrskyn silmässä oleminen.	Tämä tilanne on erittäin sekava ja epävakaa.
Hänellä on pää täynnä pilviä.	Hän on hajamielinen ja unelmoiva.
Se on sulaa kultaa.	Se on erittäin arvokasta ja ihailtavaa.
Hänellä on avara sydän.	Hän on hyvin ystävällinen ja antelias.
Hänellä on rotta ruumiissaan.	Hän on vilkas, kekseliäs ja juonitteleva.

ChatGPT selkeästi kykenee hahmottamaan kuvakielen ja kirjaimellisen kielen eron, jonka malli todennäköisesti on oppinut laajasta esikoulutusmateriaalista. Tuotettujen kielikuvien laatu vaikuttaisi kuitenkin olevan ainakin osittain varsin huono. Osa tuotetuista lauseista on yleisesti suomen kielessä käytettyjä ilmauksia, kuten esimerkiksi ”aika lentää kuin siivillä”. Suurin osa lauseista ei kuitenkaan aukea sellaisenaan ilman kirjaimellista vastinetta, esimerkiksi ”hänellä on rotta ruumiissaan” tai ”hän meni kylmää vettä juomaan”, josta voidaan tulkita, että tuotetut lauseet eivät ole riittävän osuvia ollakseen käyttökelpoisia. Juuri ilmausten osuvuus (engl. aptness) nähdään yhtenä laadukkaan metaforan tunnusmerkeistä (Ichien ym., 2024), ja jokapäiväiseen kielenkäyttöön siirtyvät kielikuvat ovat usein varsin osuvia (Jones & Estes, 2006). Kuvakieliset ilmaukset eivät ole samanlaisia kielestä toiseen, vaan ne ovat vahvasti kulttuuri- ja yhteiskuntasidonnaisia (esim. Kabra ym., 2023). Toisessa kulttuurissa ja yhteiskunnassa osuva ilmaus ei välttämättä lainkaan sovi toisenlaiseen kulttuuriin. Epäselväksi jää, millainen kyky kielimalleilla on sovittaa tuotettua sisältöä eri kulttuurien kontekstiin.

5 Yhteenveto

Kuvakiellellä tarkoitetaan kieltä, joka hyödyntää kielikuvia, esimerkiksi metaforia, idiomeja tai metonymiaa. Kuvakieltä käytetään kuvaamaan muun muassa abstrakteja konsepteja ja rikastuttamaan viestintää. Sen käyttö on yleistä jokapäiväisessä kommunikaatiossa.

Kielimallit ovat sanojen todennäköisyyksiä ennustavia laskentamalleja, joiden avulla pystytään ratkaisemaan erilaisia luonnollisen kielen käsittelyn tehtäviä, joita ovat esimerkiksi automaattinen kielen kääntäminen ja sentimenttianalyysi. Suuri osa nykyisistä kielimalleista perustuu transformer-arkkitehtuurille, jonka erityispiirteenä on sanojen kontekstisidonnaisen merkityksen tunnistava itsehuomiomekanismi. Transformer-arkkitehtuuria hyödyntävät esimerkiksi GPT-, BERT- ja BART-kielimallit. Vaikka transformer-pohjaiset kielimallit ottavat jatkuvasti suuria kehitysaskelaita, kuvakielen käsittely on edelleen malleille haasteellista. Tämän tutkielman tavoitteena oli löytää vastaukset seuraaviin tutkimuskysymyksiin:

T1: Miten kielimalli tunnistaa, tulkitsee ja tuottaa kuvakieltä, esimerkiksi metaforia?

T2: Millaisia ongelmia nykyisillä transformer-pohjaisilla kielimalleilla on kuvakielen tunnistamisessa, tulkitsemisessä ja tuottamisessa?

Transformer-kielimallit käyvät läpi kaksivaiheisen koulutuksen. Ensimmäisessä vaiheessa mallille syötetään suuri määrä yleistä tekstiaineistoa, jonka avulla malli oppii esimerkiksi kielen yleisiä rakenteita. Malli voidaan tämän jälkeen hienosäätää tehtäväkohtaisella aineistolla. Hienosäätämässä käytetään usein manuaalisesti tai osittain automatisoiden koostettuja aineistoja. Kuvakielen kannalta olennaista hienosäätöaineistossa on runsas määrä esimerkkejä kuvakielisistä ilmauksista ja niiden kirjaimellisista vastineista. Tutkielman tulokset osoittavat, että kielimallin kyky tunnistaa, tulkita ja tuottaa kuvakieltä perustuvat kielimallin laajaan esikoulutukseen ja laadukasta datasettiä hyödyntävään hienosäätöprosessiin. Transformerin itsehuomiomekanismin ja kontekstisidonnaisten sanaoputusten avulla kielimalli kykenee muodostamaan sanoille ja lauseille kontekstiin sopivan merkityksen. Keskeistä on myös transformer-mallien kyky käsitellä verrattain suuria määriä materiaalia kerralla, jolloin pitkissäkin lauseissa malli ei unohda käsittelyssä olevia sanoja edeltävää tekstiä liian aikaisin. Kuvakielen tunnistamis- ja tulkitsemistehtävissä hyviä tuloksia on saatu käyttämällä esimerkiksi BERT- ja RoBERTa-malleja. Kuvakielen tuottamistehtävässä BART-malli on suosittu.

Kuvakielen tuottamistehtävässä keskeiseksi haasteeksi tunnistettiin, että tuotetut kielikuvat eivät aina ole riittävän osuvia, eivätkä siten ole aina helposti ymmärrettäviä tai ihmisen tuottaman kielen veroisia. Yksi syy tähän voi olla kuvakielen kulttuurisidonnaisuus. Vaikka kielikuvia voi jossain määrin kääntää ymmärrettävästi toisille kielille, jokaisella kielellä ja yhteiskunnalla on omia juuri kyseiseen kulttuuriin soveltuvia kielikuvia, jotka eivät ole pelkällä käännöksellä selitettävissä. Alan tutkimuksen kannalta hidasteena on laadukkaiden hienosäätöön tarvittavien datasettien vähyys ja eri kielivariaatioiden puute. Kielimalleilla on edelleen ongelmia tunnistaa monikielisiä ja kulttuurisidonnaisia kielikuvia.

Kuvakieleen ja kielimalleihin liittyviä jatkotutkimuskohteita ovat esimerkiksi tehokkaampien keinojen kehittäminen laadukkaiden datasettien laatimiseen mallien hienosäätämistä varten, kehoitteiden muotoilu ja mallien esikoulutuksen potentiaalin hyödyntäminen. Myös kielimallien sisältämän kulttuurisen kontekstin tuntemus on mielenkiintoinen tutkimuskohde. Onko kulttuurisidonnaisuus ylipäättään opetettavissa kielimallille? Kuinka paljon kielimallien kasvava käyttö tulee vaikuttamaan kielten kehitykseen ja ihmisen kielenkäytön kehitykseen jatkossa? Kuvakieli vaihtelee kielestä ja kulttuurista toiseen, jonka vuoksi esimerkiksi englanninkielisillä aineistoilla koulutettujen mallien käyttäminen suomen kielen osaamista vaativissa tehtävissä ei tuota haluttuja tuloksia, tai ainakaan pysty käyttämään hyväksi kaikkia kielen sisältämiä hienovaraisia vivahteita.

Lähteet

- Aghazadeh, E., Fayyaz, M., & Yaghoobzadeh, Y. (2022). Metaphors in Pre-Trained Language Models: Probing and Generalization Across Datasets and Languages. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1, 2037–2050. <https://doi.org/10.18653/v1/2022.acl-long.144>
- Aristoteles. (1997). *Runousoppi* (P. Saarikoski, Käänt.). Kustannusosakeyhtiö Otava.
- Babieno, M., Takeshita, M., Radisavljevic, D., Rzepka, R., & Araki, K. (2022). MIss RoBERTa WiLDe: Metaphor Identification Using Masked Language Model with Wiktionary Lexical Definitions. *Applied Sciences* 2022, Vol. 12, Page 2081, 12(4), 2081. <https://doi.org/10.3390/APP12042081>
- Bowdle, B. F., & Gentner, D. (2005). The Career of Metaphor. *Psychological Review*, 112(1), 193–216. <https://doi.org/10.1037/0033-295X.112.1.193>
- Chakrabarty, T., Choi, Y., & Shwartz, V. (2022). It's not Rocket Science: Interpreting Figurative Language in Narratives. *Transactions of the Association for Computational Linguistics*, 10, 589–606. https://doi.org/10.1162/TACL_A_00478
- Chakrabarty, T., Muresan, S., & Peng, N. (2020). Generating similes effortlessly like a Pro: A Style Transfer Approach for Simile Generation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 6455–6469. <https://doi.org/10.18653/V1/2020.EMNLP-MAIN.524>
- Chakrabarty, T., Zhang, X., Muresan, S., & Peng, N. (2021). MERMAID: Metaphor Generation with Symbolism and Discriminative Decoding. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4250–4261. <https://doi.org/10.18653/v1/2021.naacl-main.336>
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1, 4171–4186. <https://arxiv.org/abs/1810.04805v2>
- Evans, V., & Green, M. (2006). *Cognitive Linguistics*. Edinburgh University Press.
- Ge, M., Mao, R., & Cambria, E. (2023). A Survey on Computational Metaphor Processing Techniques: From Identification, Interpretation, Generation to Application. *Artificial Intelligence Review*, 56, 1829–1895. <https://doi.org/10.1007/s10462-023-10564-7>
- Gibbs, R. (1994). *The Poetics of Mind*. Cambridge University Press.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. <https://www.deeplearningbook.org/>

- Harris, Z. S. (1954). Distributional Structure. *WORD*, 10(2–3), 146–162.
<https://doi.org/10.1080/00437956.1954.11659520>
- Harvilahti, L., Kalliokoski, J., Nikanne, U., & Onikki, T. (Toim.). (1992). *Metafora: Ikkuna kieleen, mieleen ja kulttuuriin*. Suomalaisen kirjallisuuden seura.
- Holyoak, K. J., & Stamenković, D. (2018). Metaphor Comprehension: A Critical Review of Theories and Evidence. *Psychological Bulletin*, 144(6), 641–671. <https://doi.org/10.1037/bul0000145>
- Hosiaislouma, Y. (2003). *Kirjallisuuden sanakirja*. WS Bookwell Oy.
- Ichien, N., Stamenković, D., & Holyoak, K. J. (2024). Large Language Model Displays Emergent Ability to Interpret Novel Literary Metaphors. *Metaphor and Symbol*, 39(4), 296–309.
<https://doi.org/10.1080/10926488.2024.2380348>
- Jones, L. L., & Estes, Z. (2006). Roosters, robins, and alarm clocks: Aptness and conventionality in metaphor comprehension. *Journal of Memory and Language*, 55(1), 18–32.
<https://doi.org/10.1016/j.jml.2006.02.004>
- Jurafsky, D., & Martin, J. H. (2024). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models* (3. p.). <https://web.stanford.edu/~jurafsky/slp3/>
- Kabra, A., Liu, E., Khanuja, S., Aji, A. F., Winata, G. I., Cahyawijaya, S., Aremu, A., Ogayo, P., & Neubig, G. (2023). Multi-lingual and Multi-cultural Figurative Language Understanding. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 8269–8284. <https://doi.org/10.18653/V1/2023.FINDINGS-ACL.525>
- Lai, H., & Nissim, M. (2022). Multi-Figurative Language Generation. *29th International Conference on Computational Linguistics*, 5939–5954. <https://aclanthology.org/2022.coling-1.519>
- Lai, H., & Nissim, M. (2024). A Survey on Automatic Generation of Figurative Language: From Rule-based Systems to Large Language Models. *ACM Computing Surveys*, 56(10).
<https://doi.org/10.1145/3654795/ASSET/44280426-3879-4586-893C-A480BAECDEB8/ASSETS/GRAPHIC/CSUR-2023-0284-F03.JPG>
- Lai, H., Toral, A., & Nissim, M. (2023). Multilingual Multi-Figurative Language Detection. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 9254–9267. <https://doi.org/10.18653/V1/2023.FINDINGS-ACL.589>
- Lakoff, G., & Johnson, M. (1980). *Metaphors We Live By*. University of Chicago Press.
- Liu, E., Cui, C., Zheng, K., & Neubig, G. (2022). Testing the Ability of Language Models to Interpret Figurative Language. *NAACL 2022 - 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, 4437–4452. <https://doi.org/10.18653/v1/2022.naacl-main.330>
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., & Allen, P. G. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. <https://arxiv.org/abs/1907.11692v1>

- Mao, R., Ge, M., Han, S., Li, W., He, K., Zhu, L., & Cambria, E. (2025). A survey on pragmatic processing techniques. *Information Fusion*, *114*, 102712.
<https://doi.org/10.1016/j.inffus.2024.102712>
- Mao, R., He, K., Beth Ong, C., Liu, Q., & Cambria, E. (2024). MetaPro 2.0: Computational Metaphor Processing on the Effectiveness of Anomalous Language Modeling. *Findings of the Association for Computational Linguistics ACL 2024*, 9891–9908.
<https://doi.org/10.18653/V1/2024.FINDINGS-ACL.590>
- Mao, R., Li, X., Ge, M., & Cambria, E. (2022). MetaPro: A computational metaphor processing model for text pre-processing. *Information Fusion*, *86–87*, 30–43.
<https://doi.org/10.1016/J.INFFUS.2022.06.002>
- OpenAI. (2022). *Introducing ChatGPT*. <https://openai.com/index/chatgpt/>
- Potamias, R. A., Siolas, G., & Stafylopatis, A. (2021). A Robust Deep Ensemble Classifier for Figurative Language Detection. *Communications in Computer and Information Science*, *1000*, 164–175. https://doi.org/10.1007/978-3-030-20257-6_14
- Potamias, R. A., Siolas, G., & Stafylopatis, A. G. (2019). A Transformer-based Approach to Irony and Sarcasm Detection. *Neural Computing and Applications*, *32(23)*, 17309–17320.
<https://doi.org/10.1007/s00521-020-05102-3>
- Ptiček, M., & Dobša, J. (2023). Methods of Annotating and Identifying Metaphors in the Field of Natural Language Processing. *Future Internet*, *15(6)*, 201. <https://doi.org/10.3390/FI15060201>
- Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving Language Understanding by Generative Pre-Training*. <https://api.semanticscholar.org/CorpusID:49313245>
- Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4. p.). Pearson Education.
- Shannon, C. E. (1951). Prediction and Entropy of Printed English. *Bell System Technical Journal*, *30(1)*, 50–64.
- Shou, X., Huang, X., & Xi, W. (2024). Conceptual Metaphor Theory Guides GANs for Generating Metaphors and Interpretations. *IEEE Access*, *12*, 55357–55366.
<https://doi.org/10.1109/ACCESS.2024.3389071>
- Steen, G. J., Dorst, A. G., Herrmann, J. B., Kaal, A., Krennmayr, T., & Pasma, T. (2010). *Method for Linguistic Metaphor Identification: From MIP to MIPVU* (1. p.). John Benjamins Publishing Company.
- Stowe, K., Chakrabarty, T., Peng, N., Muresan, S., & Gurevych, I. (2021). Metaphor Generation with Conceptual Mappings. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 6724–6736. <https://doi.org/10.48550/arXiv.2106.01228>
- Su, C., Fukumoto, F., Huang, X., Li, J., Wang, R., & Chen, Z. (2020). DeepMet: A Reading Comprehension Paradigm for Token-level Metaphor Detection. *Proceedings of the Second*

- Workshop on Figurative Language Processing*, 30–39.
<https://doi.org/10.18653/V1/2020.FIGLANG-1.4>
- Su, C., Wang, X., Chang, Y., Wu, K., & Chen, Y. (2025). Metaphor generation based on noval evaluation method. *Neurocomputing*, 611. <https://doi.org/10.1016/j.neucom.2024.128651>
- Tedeschi, S., Martelli, F., & Navigli, R. (2022). ID10M: Idiom Identification in 10 Languages. *Findings of the Association for Computational Linguistics: NAACL 2022*, 2715–2726.
<https://doi.org/10.18653/v1/2022.findings-naacl.208>
- Tong, X., Shutova, E., & Lewis, M. (2021). Recent Advances in Neural Metaphor Processing: A Linguistic, Cognitive and Social Perspective. *NAACL-HLT 2021 - 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, 4673–4686.
<https://doi.org/10.18653/V1/2021.NAACL-MAIN.372>
- Tuffery, S. (2022). *Deep Learning: From Big Data to Artificial Intelligence with R*. John Wiley & Sons.
- Tuominen, H., & Neittaanmäki, P. (2019). *Tekoälyn perusteita ja sovelluksia*.
<http://urn.fi/URN:ISBN:978-951-39-7796-2>
- Uszkoreit, J. (2017). *Transformer: A Novel Neural Network Architecture for Language Understanding*. <https://research.google/blog/transformer-a-novel-neural-network-architecture-for-language-understanding/>.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010.
- Yu, Z., & Wan, X. (2019). How to Avoid Sentences Spelling Boring? Towards a Neural Approach to Unsupervised Metaphor Generation. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 1*, 861–871. <https://doi.org/10.18653/V1/N19-1092>
- Zayed, O., McCrae, J. P., & Buitelaar, P. (2020). Figure Me Out: A Gold Standard Dataset for Metaphor Interpretation. *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 5810–5819. <https://aclanthology.org/2020.lrec-1.712>
- Zhao, B., Jin, W., Zhang, Y., Huang, S., & Yang, G. (2023). Prompt learning for metonymy resolution: Enhancing performance with internal prior knowledge of pre-trained language models. *Knowledge-Based Systems*, 279, 110928.
<https://doi.org/10.1016/J.KNOSYS.2023.110928>