

Arkaluonteisten tietojen tunnistaminen digitaalisesta arkistoaineistosta

Juho Laulajainen
Pro gradu -tutkielma
Kieliasiantuntijuuden tutkinto-ohjelma, Digitaalinen kielentutkimus
Humanistinen tiedekunta
Turun yliopisto
Toukokuu 2025

Pro gradu -tutkielma

Kieliasiantuntijuuden tutkinto-ohjelma, digitaalinen kielentutkimus

Juho Laulajainen

Arkaluonteisten tietojen tunnistaminen digitaalisesta arkistoaineistosta

Sivumäärät: 43 sivua, 11 liitesivua

Kansallisarkisto vastaa kansallisesti merkittävien asiakirjojen säilyttämisestä ja niiden käytön edistämisestä. Sen kokoelmat kuvaavat hallinnon, kansalaisyhteiskunnan ja yksilöiden toimintaa eri aikoina ja muodostavat perustan sekä tutkimukselle että yhteiskunnalliselle päätöksenteolle. Sen aineistoista merkittävä osa sisältää henkilötietoja tai erityisen henkilötietoryhmän tietoja, joita suojellaan tietosuojalainsäädännöllä.

Tässä tutkielmassa selvitetään, voiko generatiivisilla kielimalleilla tunnistaa henkilötietoja tai erityisen henkilötietoryhmän tietoja suomenkielisestä digitaalisesta arkistoaineistosta. Tutkielman tavoitteena on tuottaa empiiristä tietoa tekoälyn käyttömahdollisuuksista arkistotoimissa sekä tukea käytännön arkistotyötä tehostavien työkalujen kehittämistä.

Tutkielmassa käytetään seitsemää generatiivista decoder-only transformer -arkkitehtuuriin perustuvaa kielimallia, joita ajetaan paikallisesti Kansallisarkiston laitteistolla. Kielimallien suorituskykyä mitataan osuvuudella, tarkkuudella, kattavuudella, F1-arvolla ja macro F1-arvolla. Tutkielmassa käytettävä luokitteluaineisto sisältää dokumentteja Jahvetin kirjelaatikko -arkistosta sekä Kansallisarkiston antamista seulontapäätöksistä. Kokonaisuudessaan aineisto sisältää 300 digitaalista arkistoasiakirjasivua, jotka on annotoitu joko julkisiksi tai arkaluonteisiksi.

Tutkielman keskeisin tutkimustulos on, että parhaimmillaan generatiiviset kielimallit voivat tunnistaa henkilötietoja ja erityisen henkilötietoryhmän tietoja digitaalisesta arkistoaineistosta erittäin hyvin. Paras tulos saavutettiin Qwen 3 32B -kielimallilla viiden esimerkkidokumentin few-shot-asetelmassa, jolloin F1-arvolla mitattuna sen suorituskyky oli 94,87 %.

Avainsanat: arkistot, tietosuoja, koneoppiminen, tekoäly, kielimallit, tekstiluokittelu

Sisällysluettelo

1	Johdanto	4
2	Tausta	8
2.1	Kansallisarkisto ja henkilötietoja sisältävät aineistot	8
2.2	Koneavusteinen tekstiluokittelu	10
2.2.1	Sääntöpohjaiset menetelmät	10
2.2.2	Koneoppimiseen perustuvat menetelmät	11
2.2.3	Suuriin kielimalleihin perustuvat menetelmät	13
3	Aineisto	18
3.1	Luokitteluaineisto	18
3.2	Esikäsittely	19
4	Menetelmät	22
4.1	Luokittelija	22
4.2	Kielimallit	28
4.3	Arviointi	29
5	Analyysi	31
5.1	Tulokset	31
5.1.1	Luokittelun tulokset ilman optimointia	31
5.1.2	Luokittelun tulokset optimoiduilla kehoitteilla	33
5.2	Virheanalyysi	35
5.3	Johtopäätökset	36
6	Lopuksi	38
	Lähteet	40
	Liitteet	44
	Liite 1 Esimerkki kehoitteesta ja vastauksesta	44

1 Johdanto

Kansallisarkisto vastaa kansallisesti merkittävien asiakirjojen säilyttämisestä ja niiden käytön edistämisestä. Sen aineistot kuvaavat hallinnon, kansalaisyhteiskunnan ja yksilöiden toimintaa eri aikoina ja muodostavat perustan sekä tutkimukselle että yhteiskunnalliselle päätöksenteolle. Viime vuosina digitaalisten arkistoaineistojen määrä on kasvanut nopeasti, ja vuoden 2024 lopussa Kansallisarkiston hallussa oli yli 252 miljoonaa tiedostoa tai kuvaa. Laajenevat digitaaliset kokoelmat ovat tuoneet mukanaan yksityishenkilöiden tietosuojaan liittyviä haasteita.

Tässä tutkielmassa selvitetään, voiko generatiivisilla kielimalleilla tunnistaa henkilötietoja tai erityisen henkilötietoryhmän tietoja suomenkielisestä digitaalisesta arkistoaineistosta.

Tutkimustehtäväni on myös muotoiltavissa kahdeksi suoraksi tutkimuskysymykseksi:

- 1) Kuinka luotettavasti esikoulutetut generatiiviset kielimallit tunnistavat henkilötietoja ja erityisen henkilötietoryhmän tietoja suomenkielisestä arkistoaineistosta?
- 2) Onko eri kielimallien luotettavuudessa merkittäviä eroja, ja mitkä kielelliset, kontekstuaaliset tai tekniset tekijät selittävät niitä?

Tutkielman tavoitteena on tuottaa uutta, empiiristä tietoa tekoälyn käyttömahdollisuuksista arkistotoimissa sekä tukea käytännön arkistotyötä tehostavien työkalujen kehittämistä.

Lisäksi tutkielma tuottaa tietoa siitä, kuinka hyvin yhdellä grafiikkasuorittimella ajettavat kielimallit ymmärtävät suomenkielisiä tekstejä ja niiden kontekstuaalisia piirteitä.

Henkilötietojen ja erityisen henkilötietoryhmän tietojen tunnistaminen suomenkielisestä arkistoaineistosta generatiivisilla kielimalleilla on tutkimusaiheena uusi ja tutkimaton.

Tämän tutkielman tutkimustehtävä ja tavoitteet ovat syntyneet tarpeesta. Yhtäältä

Kansallisarkiston lakisääteinen tehtävä on edistää aineistojensa käyttöä ja saatavuutta. Käytön edistämisen osana Kansallisarkistossa kehitetään aktiivisesti tekoälyyn perustuvia

sisällöntunnistus- ja analyysimenetelmiä. Tässä tutkielmassa tehdyt tekniset ratkaisut ovat osa tätä kehitystyötä. Toisaalta kesällä 2024 Kansallisarkistolle syntyi tarve tunnistaa, kuinka iso

osa Jahvetin Kirjelaatikko -arkiston dokumenteista sisältää henkilötietoja tai erityisen

henkilötietoryhmän tietoja ja missä laajuudessa se voitaisiin avata käyttöön. Arkiston suuren

koon vuoksi sen manuaalista analysointia ei nähty toteuttamiskelpoisena vaihtoehtona, vaan

ainoaksi mahdolliseksi toteuttamistavaksi todettiin koneavusteinen tekstiluokittelu. Päätös on ohjannut tämän tutkielman aineistovalintaa.

Tutkielmassa käytettävä aineisto sisältää dokumentteja Valtioneuvoston kanslian Jahvetin kirjelaatikko -arkistosta sekä Kansallisarkiston antamista seulontapäätöksistä.

Kokonaisuudessaan aineisto sisältää 300 digitaalista asiakirjasivua, jotka on ositettu kahteen osakokonaisuuteen. Molemmat osiot sisältävät käyttörajoitettuja asiakirjoja, jotka saa toistaiseksi käyttöönsä vain Kansallisarkiston tutkijasaleissa. Käytäntö on voimassa olevan arkistolain (831/1994) ohjaama. Tutkielman teossa on noudatettu Tutkimuseettisen neuvottelukunnan (TENK)¹ ja Kansallisarkiston ohjeita, ja sen tekninen toteutus on tehty Kansallisarkiston virkatyönä. Tämän tutkielman tuotokset eli ohjelmakoodi² ja kirjallinen osuus eivät sisällä käyttörajoitettuja tai muuten salassa pidettäviä tietoja.

Tutkielmassa on kolme keskeistä, lainsäädännön määrittelemää käsitettä. *Henkilötiedolla* tarkoitetaan tietoa, joka mahdollistaa yksityishenkilön suoran tai välillisen tunnistamisen. Henkilötietoja ovat esimerkiksi nimi, puhelinnumero, kotiosoite, auton rekisterinumero tai tiedot esimerkiksi iso- tai isoisovanhempien perinnöllisistä sairauksista. *Erityisten henkilötietoryhmien tiedolla* tarkoitetaan henkilöön yhdistettäviä arkaluonteisia tietoja, joiden käsittely voi aiheuttaa huomattavia riskejä henkilön perusoikeuksille ja -vapauksille ja joita on suojeltava erittäin tarkasti. Molempien käsitteiden määrittely perustuu yleiseen tietosuojasetukseen (Euroopan unionin yleinen tietosuojasetus 2016/679, jäljempänä GDPR) ja kansalliseen tietosuojalakiin (tietosuojalaki 1050/2018). Tilanteissa, joissa viitataan molempiin termeihin mutta niiden erottaminen toisistaan ei ole tarpeellista, käytetään sateenvarjotermiä *arkaluonteiset tiedot*.

Tutkielman tutkimustehtävä on pohjimmiltaan kielimalleilla suoritettava binäärinen tekstiluokittelutehtävä, jossa arkistoasiakirjat luokitellaan kahteen mahdolliseen luokkaan: *julkinen* tai *arkaluonteinen tieto*. Luokittelussa käytetään seitsemää generatiivista decoder-only transformer -arkkitehtuuriin perustuvaa kielimallia, joita ajetaan paikallisesti Kansallisarkiston laitteistolla. Mallit on julkaistu avoimeen käyttöön, ja niiden toimintaa ohjataan Ollama-ohjelmistokehyksellä ja DSPy-ohjelmointikehyksellä. Syy mallien paikalliseen ajamiseen perustuu yksityishenkilöiden yksityiselämän suojaamiseen, minkä takia tässä tutkielmassa ei ole käytetty kaupallisten palveluntarjoajien ylläpitämiä tekoälypalveluita tai kielimalleja.

¹ https://tenk.fi/sites/default/files/2021-01/Ihmistieteiden_eettisen_ennakkoarvioinnin_ohje_2020.pdf

² <https://github.com/juvilau/DIKI/>

Tutkielma etenee johdannosta teorialukuun ja aineisto-, menetelmä- sekä analyysilukujen kautta aina päätäntöön. Ensimmäisessä käsittelyluvussa eli järjestykseltään toisessa luvussa käsitellään tutkielman käytännöllistä ja teoreettista taustaa. Ensimmäisessä alaluvussa avataan Kansallisarkiston toimintaa ja syitä, miksi sen arkistoinnissa aineistoissa on runsaasti henkilötietoja sisältäviä asiakirjoja. Samalla täsmennetään henkilötiedon ja erityisen henkilötietoryhmän tiedon käsitteitä sekä avataan arkistoalaa ohjaavaa tietosuojalainsäädäntöä. Toisessa alaluvussa puolestaan käsitellään koneavusteista tekstiluokittelua ja esitellään tälle tutkielmalle keskeistä tutkimusta ja teoriataustaa. Käsittely jakaantuu edelleen kolmeen osioon, jotka ovat sääntöpohjaiset tekstiluokittelumenetelmät, koneoppimiseen perustuvat tekstiluokittelumenetelmät sekä suuriin kielimalleihin perustuvat tekstiluokittelumenetelmät.

Kolmannessa luvussa käsitellään tässä tutkielmassa käytetyn luokitteluaineiston muodostamista ja esikäsittelyä. Jahvetin kirjelaatikko -arkiston asiakirjat muodostavat tässä tutkielmassa käytetyn luokitteluaineiston rungon, ja sen esittely muodostaa kolmannen luvun ensimmäisen alaluvun keskeisen asiasisällön. Kolmannen luvun toisessa alaluvussa puolestaan käsitellään luokitteluaineiston varsinaista koostamista sekä menetelmiä, joilla Jahvetin kirjelaatikko -arkiston digitoitu kirjeaineisto muunnettiin koneluettavaan muotoon, esikäsiteltiin ja ositettiin kahdeksi osakokonaisuudeksi. Samalla eritellään osakokonaisuuksiin sisältyvien dokumenttien määrää, pituuksia ja niiden jakautumista eri luokkiin.

Neljännän luvun keskeinen asiasisältö koostuu luokitteluohjelman teknisen toteutuksen dokumentaatiosta. Ensimmäisessä alaluvussa esitellään tässä tutkielmassa kehitettyä luokittelijaa ja avataan sen tekniselle toteutukselle keskeistä DSPy-ohjelmointikehystä ja sillä suoritettua optimointiprosessia. Toisen alaluvun asiasisältö muodostuu sen sijaan henkilötietojen tunnistamisessa kokeilluista kielimalleista ja niiden teknisten tietojen erittelystä. Kolmannessa alaluvussa avataan tässä tutkielmassa käytettyjä arviointimittareita eli metriikoita, joilla eri kielimallien suorituskykyä ja luokittelijan toimintaa voidaan arvioida.

Viidennessä luvussa käsitellään varsinaisia tuloksia. Ensimmäisen alaluvun polttopisteenä on, kuinka hyvin monikieliset generatiiviset kielimallit tunnistivat arkaluonteisia tietoja suomenkielisestä arkistoaineistosta. Vastaavasti toisen alaluvun ydin on virheanalyysi, jossa pohditaan syitä virheellisille tai muuten epäonnistuneille luokitteluille. Viidennen luvun kolmannessa alaluvussa vastataan tämän tutkielman tutkimuskysymyksiin. Lopulta Kuudennessa luvussa lopulta tiivistetään tutkielman keskeiset havainnot yhteen, pohditaan

tutkielman tieteellistä ja käytännöllistä merkitystä sekä hahmotellaan mahdollisia jatkotutkimusaiheita.

2 Tausta

Tämän tutkielman tutkimustehtävä on pohjimmiltaan arkistoalaan liittyvä tekstiluokittelutehtävä, jossa luokittelijana toimii generatiivinen kielimalli. Tämä muodostaa rungon myös tutkielman taustaa ja tutkimuksellisia kiinnekohtia käsittelevälle luvulle. Ensimmäisessä alaluvussa avataan Kansallisarkiston toimintaa ja arkistoalaa ohjaavaa tietosuojalainsäädäntöä. Samalla täsmennetään henkilötiedon ja erityisen henkilötietoryhmän tiedon käsitteitä. Toisessa alaluvussa puolestaan käsitellään koneavusteista tekstiluokittelua. Alaluku jakaantuu edelleen kolmeen osioon, jotka käsittelevät sääntöpohjaista tekstiluokittelua, koneoppimiseen perustuvaa tekstiluokittelua sekä kielimalleilla tehtävää tekstiluokittelua.

2.1 Kansallisarkisto ja henkilötietoja sisältävät aineistot

Suurimmassa osassa maailman maita on valtiollinen arkistolaitos. Niiden olemassaolo liittyy tavallisesti muiden julkisoikeudellisten organisaatioiden toimintaan. Jokainen ihminen asioi eri viranomaisten kanssa elämänsä eri vaiheissa ja tarvitsee julkisia palveluita tavalla tai toisella. Syntymisestä, elämisestä ja kuolemasta muodostuu asiakirjoja, joista osa arkistoidaan ensisijaisen käyttötarkoituksensa päätyttyä. Suomessa tätä tehtävää hoitaa Kansallisarkisto, ja sen aineistokokonaisuuksista valtaosa onkin viranomaisarkistoja. Lakisääteisenä tehtävänä se vastaanottaa viranomaisten luovuttamia asiakirjoja, hankkii yhteiskunnallisesti tärkeitä tietoaineistoja sekä varmistaa asiakirjallisen kulttuuriperinnön säilymisen ja tutkimuksellisesti arvokkaiden tietoaineistojen saatavuuden. (Litzen, 1994; Lybeck et al., 2006.)

Suurin osa Kansallisarkiston säilyttämistä asiakirja-aineistoista on lähtökohtaisesti julkisia. Asiaa säätelee laki viranomaisten toiminnan julkisuudesta (621/1999) eli niin sanottu julkisuuslaki. Osaan aineistoista voi kuitenkin liittyä luovutus sopimuksen tai lainsäädännön mukaisia käyttörajoituksia. Käyttörajoituksen syynä voi olla esimerkiksi asiakirjojen huono kunto tai salassa pidettäväksi määräytyvät tiedot. Jälkimmäiset liittyvät tavallisesti henkilötietoihin ja yksityishenkilöiden yksityiselämään, joita suojellaan EU:n tietosuoja-asetuksen (GDPR) ja kansallisen tietosuojalain (1050/2018) velvoittamina.³

³ Julkisen hallinnon tietoaineistojen arkistointia ohjaavasta lainsäädännöstä enemmän ks. Rasimus, 2019.

EU:n tietosuoja-asetuksen (GDPR 4 artikla 1 kohta) mukaan henkilötietoja ovat kaikki sellaiset tiedot, joiden avulla henkilö voidaan tunnistaa suoraan tai epäsuorasti. Esimerkiksi yksityishenkilön nimi, henkilötunnus, sijaintiedot, verkkotunnisteet tai tunnusomaiset fysiologiset, geneettiset, psyykkiset, taloudelliset, kulttuurilliset tai sosiaaliset tekijät ovat tällaisia. Myös esimerkiksi puhelinnumero, kasvokuva, sormenjälki, kotiosoite, auton rekisterinumero, lemmikin eläinlääkärیتiedot tai jopa tiedot iso- ja isoisovanhempien perinnöllisistä sairauksista voivat tietyissä tilanteissa täyttää henkilötiedon määritelmän.⁴ Koska valtaosa Kansallisarkiston aineistosta on syntynyt viranomaistoiminnassa, niihin sisältyy runsaasti edellä mainittuja tietoja. Kuten mainittu, jokainen ihminen asioi eri viranomaisten kanssa syntymästä kuolemaan.

Osa Kansallisarkiston säilyttämistä asiakirja-aineistosta voi sisältää myös niin sanottuun erityiseen henkilötietoryhmään kuuluvia tietoja. Tällaisia ovat esimerkiksi tiedot, joista ilmenee henkilön rotu tai etninen alkuperä, poliittinen mielipide, uskonnollinen tai filosofinen vakaumus, ammattiliiton jäsenyys, terveyttä koskevia tietoja, seksuaalinen suuntautuminen tai käyttäytyminen taikka geneettisiä ja biometrisiä tietoja, joiden avulla henkilö voidaan tunnistaa. (GDPR 9 artikla 1 kohta.)⁵ Erityisiin henkilötietoryhmiin kuuluvien tietojen käsittely on lähtökohtaisesti kiellettyä, mutta kieltoon voidaan säätää poikkeuksia. Yleisen edun mukainen arkistointitarkoitus on yksi tällaisista (Lönnblad, 2025).

Arkistoinnin kannalta keskeistä on, että tietosuojalainsäädäntöä sovelletaan vain elävien henkilöiden henkilötietojen tai erityisen henkilötietoryhmän tietojen käsittelyssä. Toisin sanoen henkilötietoja ja erityiseen henkilötietoryhmään kuuluvia tietoja suojataan tietosuojalainsäädännöllä vain, jos henkilö on elossa. Kuolleita sitä vastoin suojataan julkisuuslailla, jonka 31.2§:n mukaan asiakirja, joka on määrätty salassa pidettäväksi yksityiselämän suojaamiseksi, pysyy salassa pidettynä 50 vuotta henkilön kuolemasta. Mikäli kuolinajankohta ei ole tiedossa, salassapitoaika on 100 vuotta. (Lönnblad, 2025.) Kansallisarkistossa noudatetaan usein jälkimmäistä suoja-aikaa, sillä tietoa henkilön mahdollisesta kuolemasta ei tavallisesti ole.

Tässä tutkielmassa henkilötietojen käsittely on toteutettu Kansallisarkiston ohjeistuksen ja EU:n tietosuoja-asetuksen (GDPR 9 artikla 5) mukaisesti. Tutkielman luokitteluaineistoa on käsitelty asianmukaisella huolellisuudella ja siinä esiintyvien henkilöiden ihmisarvoa,

⁴ Ks. myös <https://tietosuoja.fi/henkilotietojen-kasittely>

⁵ Ks. myös <https://tietosuoja.fi/erityisten-henkilotietoryhmien-kasittely>

yksityisyyttä ja muita oikeuksia kunnioittaen. Aineistoa on käsitelty vain Kansallisarkiston hyväksymissä tietojenkäsittely-ympäristöissä ja Kansallisarkiston hyväksymillä laitteistolla. Käsittelystä aiheutuvia riskejä on pyritty arvioimaan mahdollisimman kattavasti, ja esimerkiksi tämän tutkielman tuotokset eli kirjallinen osuus ja ohjelmakoodi eivät sisällä luokitteluaineistossa esiintyviä henkilötietoja. Tutkielman tekemisessä on noudatettu Tutkimuseettisen neuvottelukunnan (TENK) ohjeita, ja sen peruseriaatteena tai ohjenuorana on ollut luokitteluaineistossa esiintyville henkilöille aiheutuvien riskien tai haittojen minimoiminen.

2.2 Koneavusteinen tekstiluokittelu

Henkilötietojen ja erityisen henkilötietoryhmän tietojen tunnistaminen digitaalisesta arkistoaineistosta on pohjimmiltaan luokittelutehtävä, jossa syötteelle ennustetaan kategorinen luokka havaintojen perusteella. Siinä käytetyt menetelmät voidaan yleistää kahteen erilaiseen lähestymistapaan: sääntöpohjaisiin ja koneoppimiseen (Li, Q. et al., 2020; Minaee et al. 2021).

2.2.1 Sääntöpohjaiset menetelmät

Sääntöpohjaisten luokittelijoiden perusideana on tunnistaa avainsanoja, joiden läsnäolon tai puuttumisen perusteella syötteelle ennustetaan jokin ennalta määritelty luokka. Varsinainen luokittelija koostuu yksittäisten ehtojen kokoelmasta ja niitä vastaavista seuraamuksista. Ehdot tarkastetaan yksi kerrallaan, kunnes jokin täyttyy, jonka seuraamuksena syöte luokitellaan sitä vastaavaan luokkaan. (Minaee et al. 2021; Han et al. 2012.) Sääntöpohjaisten menetelmien toimintaa kuvaa alla oleva pseudokoodi:

```
if henkilötunnus in text:
    label = henkilötieto
else if puhelinnumero in text:
    label = henkilötieto
else if sähköpostiosoite in text:
    label = henkilötieto
else:
    label = julkinen
```

Sääntöpohjaisten luokittelijoiden tunnusmerkkinen piirre on niiden suoraviivainen toiminta. Ne ovat tehokkaita ja verrattain tarkkoja tehtävissä, joissa luokittelun perusteena on esimerkiksi henkilötunnusten, puhelinnumeroiden, sähköpostiosoitteiden tai muiden tietyn kaavan mukaan muodostettujen merkkijonojen sisältyminen syötteeseen. (Marciano et al., 2018.) Sääntöpohjaisia menetelmiä onkin hyödynnetty erityisesti roskapostisuodattimissa (Minaee et al., 2021), ja niitä on myös ehdotettu yhdeksi menetelmäksi, jolla voidaan tunnistaa henkilötietoja digitaalisista arkistoaineistoista (Border & Baron, 2016). Esimerkiksi Marciano et al. (2018) ovat osoittaneet, että hyvin laadituilla sääntökokoelmilla voidaan tunnistaa arkaluonteisia tietoja arkistoaineistoista, jos ne ovat rakenteellisessa muodossa ja niihin sisältyvien arkaluonteisten tietojen laatu ja muoto ovat ennalta tiedossa.

Sääntöpohjaisissa luokittelijoissa on myös heikkoutensa, kuten Yang, J. et al. (2023) ovat huomauttaneet. Heidän mukaansa niiden ylläpitäminen tuotantokäytössä on työlästä ja kallista ja siten arkistoalalla useimmiten mahdotonta. Sääntöjen päivittäminen on usein manuaalista työtä, eikä tietylle aineistokokonaisuudelle laadittu sääntökokoelma välttämättä yleisty uuteen aineistoon. Myös Jurafsky & Martin (2025) ovat esittäneet samansuuntaisia huomioita. Heidän mukaansa riittävän kattavien sääntökokoelmien laatiminen on useimmiten hyvin haastavaa ja ihmistyövoimaa vaativaa, eivätkä ihmiset ole kovin hyviä keksimään yleispäteviä sääntöjä. Siksi viime vuosikymmeninä koneoppimiseen perustuvat ratkaisut ovatkin olleet sääntöpohjaisia verrokkejaan suositumpia (Li, Q. et al. 2021).

2.2.2 Koneoppimiseen perustuvat menetelmät

Koneoppiminen voidaan yleistää kahteen eri kategoriaan. Ohjatussa koneoppimisessa (*supervised learning*) malli opetetaan ennustamaan uusia tapauksia valmiiksi luokitellun opetusaineiston avulla. Opetusaineistona toimii korpus, joka koostuu syötteiden ja niihin liitettyjen oikeiden vastausten muodostamista opetusesimerkeistä. Sitä vastoin ohjaamattomassa koneoppimisessa (*unsupervised learning*) malli oppii ennalta tuntemattomia rakenteita tai säännönmukaisuuksia etukäteen luokittelemattomasta tekstimassasta. Mallille syötettävä aineisto ei siis sisällä opetusesimerkkejä, vaan se koostuu pelkistä syötteistä, jotka malli oppii jakamaan ryhmiin niiden sisältämien piirteiden perusteella.

Jurafskyn ja Martinin (2025) mukaan suurin osa tekstiluokittimista perustuu tätä nykyä ohjattuun koneoppimiseen. Niiden perusidea voidaan ilmaista täsmällisesti niin, että funktio f on malli, joka ennustaa, mihin ennalta tunnettuun luokkaan y syöte x kuuluu:

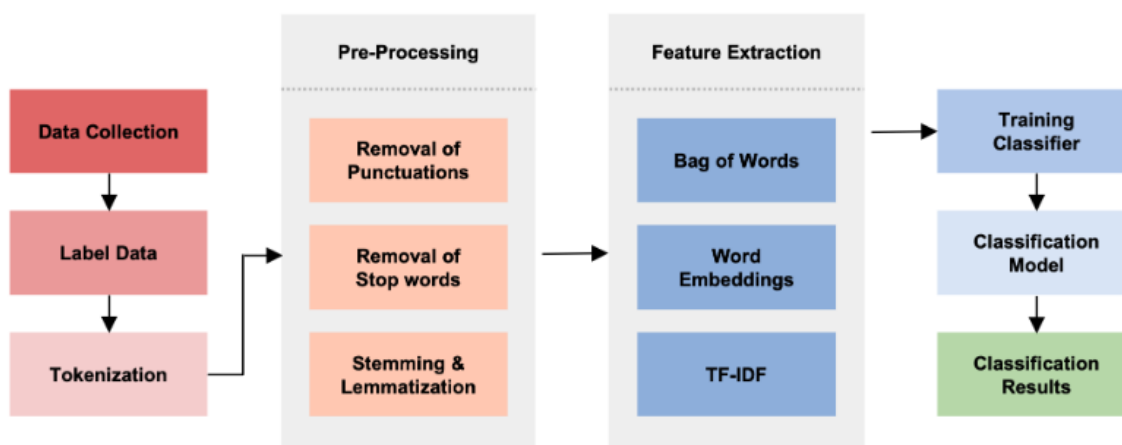
$$f: x \rightarrow y, \quad y \in Y = \{y_1, y_2, \dots, y_n\}$$

Varsinaisessa ennustamisessa voidaan käyttää useita erilaisia algoritmeja, jotka voidaan karkeasti jakaa neljään kategoriaan: todennäköisyyslaskentaan perustuviin, geometriapohjaisiin, tilastotieteellisiin ja syväoppimiseen perustuviin.⁶ (Wang et al. 2023.) Algoritmista riippumatta malli koulutetaan lopullista luokittelutehtävä edustavalla annotoidulla opetusaineistolla, joka koostuu syötevektorin x ja vastaavan tulosvektorin y muodostamista pareista eli opetusesimerkeistä:

$$\{(x_1, y_1), (x_2, y_2), \dots, (x, y_N)\}$$

Ohjatussa koneoppimisessa opetusaineisto jaetaan tavallisesti kolmeen toisistaan erilliseen osaan. Näitä ovat opetus-, validointi-, ja testiaineistot (*trainset*, *validation set*, *testset*). Opetusaineiston avulla algoritmi oppii mallin parametrit, kun taas validointiaineistoa käytetään hyperparametrien säätämiseen. Testiaineistoa puolestaan käytetään mallin suorituskyvyn arviointiin, eli siihen kuinka hyvin malli yleistyy uusiin ja ennen näkemättömiin syötteisiin. (Jurafsky & Martin, 2025.) Perinteiseen koneoppimiseen perustuvan tekstiluokitteluprosessin vuokaavio on nähtävissä alla kuvassa 2.1.

Kuva 2.1: Perinteiseen koneoppimiseen perustuvan luokittelijan tekstiluokittelu (Wang et al., 2023)



Perinteiseen koneoppimiseen perustuvat mallit ovat laskennallisesti tehokkaita ja oppivat luokittelemaan syötteitä verrattain suppean opetusaineiston perusteella. Luokittelutehtävää hyvin edustavalla, laadukkaalla ja huolellisesti esikäsitellyllä aineistolla opetettuna ne voivat

⁶ Esimerkiksi Markovin piilomalli (HMM) ja naiivi Bayes variaatioineen perustuvat todennäköisyyslaskentaan. Tukivektorikone (SVM) perustuu geometriaan, kun taas k-lähimmän naapurin menetelmä (KNN) tilastolliseen laskentaan. Syväoppimiseen perustuvia luokittelijoita ovat esimerkiksi konvoluutioverkot (CNN), LSTM-verkot sekä RNN-verkot (Wang et al. 2023).

myös yleistyä uusiin syötteisiin kohtalaisen hyvin. (Jurafsky & Martin, 2025.) Perinteistä koneoppimista onkin pyritty hyödyntämään henkilötietojen tunnistamisessa arkistoaineistosta, ja esimerkiksi Hutchinson (2018) on tutkinut naiiviin Bayesiin perustuvan luokittelijan käyttämistä arkistoitujen sähköpostiaineistojen luokittelussa.

Perinteisten koneoppimismenetelmien keskeisenä heikkoutena on, että ne edellyttävät edustavia, laadukkaita ja huolellisesti esikäsiteltyä aineistoja toimiakseen hyvin (Minaee et al., 2021). Luokiteltava aineisto on myös muunnettava piirrevektoreiksi eli koneluettavaan muotoon ennen luokittelua, ja valittujen piirteiden esikäsitteily vaikuttaa voimakkaasti lopputulokseen. Perinteiseen koneoppimiseen perustuvat luokittimet toimivatkin yleensä vain aineistoilla, joihin ne on nimenomaisesti koulutettu, eivätkä yleisty helposti uusiin luokittelutehtäviin ilman kokonaan uutta tai laajennettua opetusaineistoa (Yang et al., 2023).

2.2.3 Suuriin kielimalleihin perustuvat menetelmät

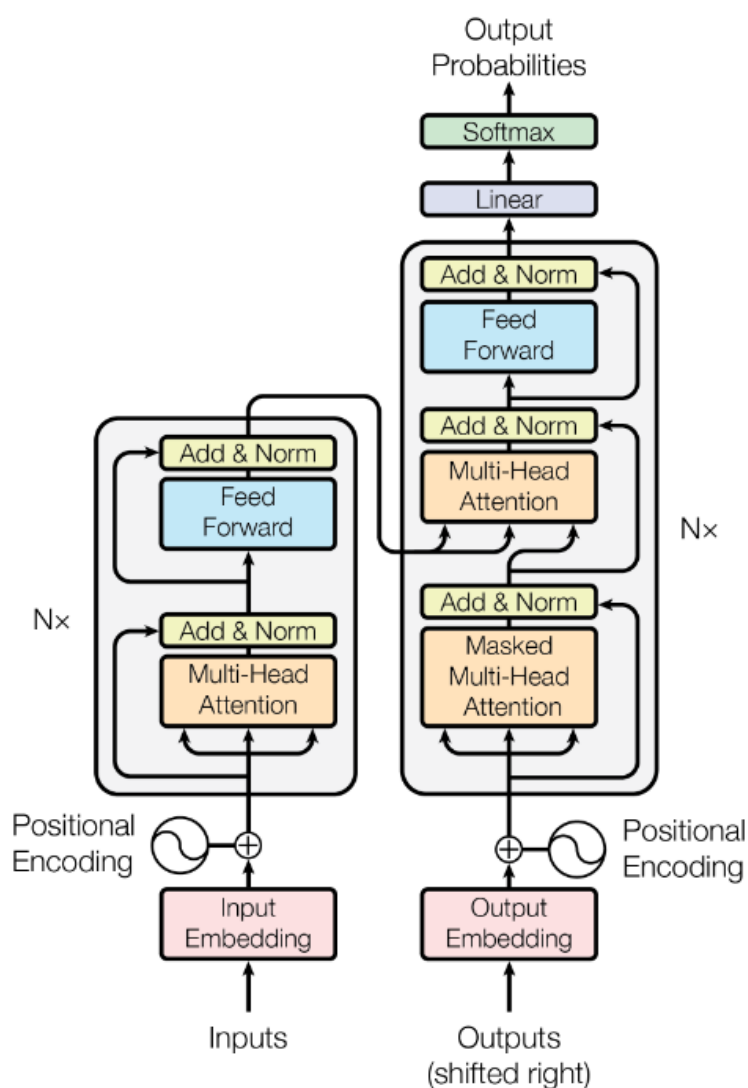
Suuret kielimallit ovat viime vuosikymmeninä yleistyneet keskeisiksi luonnollisen kielen käsittelyn työkaluiksi (Wang et al. 2017). Ne voivat oppia monimutkaisia kieliopillisia, semanttisia ja tilastollisia kielen rakenteita, minkä lisäksi ne yleistyvät uusiin tehtäviin ja aineistoihin erinomaisesti (Jurafsky & Martin, 2025). Niillä on saavutettu loistavia tuloksia erilaisia tekstiluokittelutehtävissä, kuten sentimenttianalyysissä, kielen rekisterin tunnistamisessa sekä nimettyjen entiteettien tunnistamisessa (Repo et al. 2024; Laurer et al., 2024; La Mela & Vats 2023).

Tätä nykyä lähes kaikki suuret kielimallit perustuvat transformer-arkkitehtuuriin. Tiivistetysti transformer on kielen todennäköisyysjakaumaa mallintava syvä neuroverkko, joka muodostaa saamastaan syötteestä kyselyjä (*queries*), avaimia (*keys*) ja arvoja (*values*) lineaarisilla projektiomatriiseilla. Näiden perusteella jokaiselle tokenille eli sanankaltaiselle yksikölle lasketaan painokerroin, joka kuvaa sen semanttisia ja kontekstuaalisia suhteita kaikkiin muihin tokeneihin. Tämän jälkeen painotetut arvot yhdistetään abstraktiksi esitykseksi ja syötetään eteenpäin seuraaviin kerroksiin. Lopulta kielimallin sanastossa oleville sanoille muodostetaan todennäköisyysjakauma lineaarisella projektiolla ja softmax-funktiolla sekä valitaan tokeneista todennäköisin, joka toimii tulosteena. (Vaswani et al., 2017.)

Perusmuodossaan transformer sisältää kaksi pääkomponenttia: enkooderin (*encoder*) ja dekooderin (*decoder*). Komponenttien keskinäinen tehtävänjako voidaan abstrahoida niin, että enkooderi muuntaa kielimallille annetun syötteen abstraktiksi esitykseksi, jonka avulla

dekooderi muodostaa tulosteen. Tuloste voi olla esimerkiksi useista tokeneista koostuva sekvenssi, kuten toiselle kielelle käännetty teksti, tai yksittäinen arvo, kuten ennuste syötteen luokasta. Molempien komponenttien ydin on monipäinen itsehuomiomekanismi (*multi-head attention*), joka laskee painokertoimia ja mallintaa sekä syötteen että tulosteen tokenien keskinäisiä suhteita. Enkooderi mallintaa kaikkien syötetokenien keskinäisiä suhteita, kun taas dekooderi huomioi ainoastaan aiemmin generoitujen tokenien suhteita. (Vaswani et al., 2017; Jurafsky & Martin, 2025.) Perusmuotoisen transformerin arkkitehtuurikuvaus on nähtävissä alla kuvassa 2.2.

Kuva 2.2: Transformer-arkkitehtuuri. Vasemmalla on enkooderi, oikealla on dekooderi. (Vaswani et al. 2017).



Transformer-arkkitehtuurista on sittemmin kehitetty useita muunnelmia.

Tekstilukittelutehtävissä yleisemmin käytetty lienee BERT (*Bidirectional Encoder Representations from Transformers*). BERT-mallit prosessoivat syötteen sekä vasemmalta

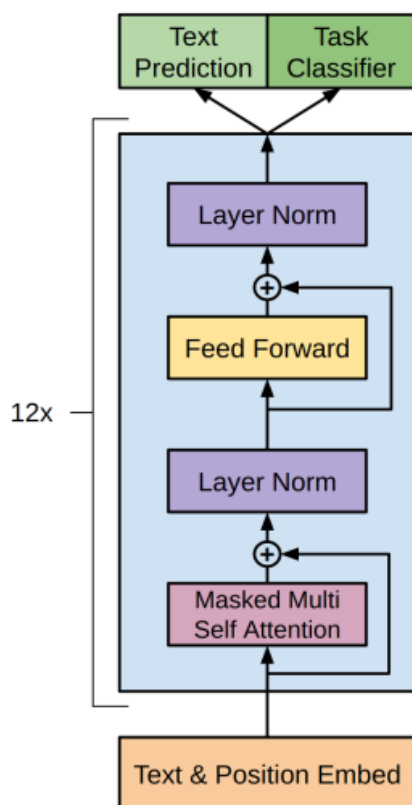
oikealle että oikealta vasemmalle ja muodostavat siitä kontekstuaalisen representaation, jonka perusteella syötteelle voidaan ennustaa luokka (Devlin et al., 2019). Keskeisin ero perusmuotoiseen transformeriin on, että BERT-malleissa käytetään vain enkooderikomponenttia, minkä takia ne soveltuvatkin erinomaisesti tehtäviin, joissa kokonaisten lauseiden ja niiden kontekstin ymmärtäminen on ensiarvoisen tärkeää. Tavallisesti BERT-mallit edellyttävät kuitenkin tehtäväspesifien opetusaineistojen muodostamista ja mallien jatkokouluttamista (Wang et al. 2023).

Tehtäväspesifien aineistojen muodostaminen on BERT-mallien keskeinen heikkous. Esimerkiksi Yang et al. (2023) ovat huomauttaneet, että laajojen opetusaineistojen koostaminen ei arkistoalalla ole useinkaan mahdollista. Haaste korostuu erityisesti arkaluonteisten tietojen tunnistamisen kohdalla. BERT-mallien hyödyntäminen arkaluonteisten tietojen tunnistamisessa on teoriassa mahdollista, mutta kuten luvussa 2.1 todettiin, EU:n tietosuoja-asetus määrittelee henkilötiedon ja erityisten henkilötietoryhmien käsitteet varsin lavasti. Edustavan opetusaineiston muodostaminen olisi siis erittäin työlästä ja paljon henkilöresursseja vaativaa.

Generatiiviset esikoulutetut kielimallit ovat kehittyneet viime vuosina nopeasti. Tätä nykyä ne saattavat koostua jopa sadoista miljardeista parametreista ja kykenevät suorittamaan monenlaisia tehtäviä ilman tehtäväspesifiä koulutusta. Tyypillisesti ne hyödyntävät vain perusmuotoisen transformer-arkkitehtuurin dekodeerikomponenttia (Radford et al., 2018), ja ne voidaan nähdä eräänlaisina BERT-mallien peilikuvina. Ne käsittelevät syötettä ja tulostetta yhtenä jatkumona, minkä takia ne voivat yleistyä puhtaasta tekstigeneroinnista erilaisiin ja vaativuudeltaan eri tasoisiin tehtäviin erinomaisesti (Brown et al., 2020).

Generatiiviset kielimallit esikoulutetaan valtavilla annotoimattomilla tekstikorpuksilla. Tämän jälkeen niitä hienosäädetään pienemmillä, tehtäväkohtaisilla opetusaineistoilla. Usein tavoitteena on opettaa malli tuottamaan tulosteita, jotka vaikuttavat luonnolliselta tai ihmismäiseltä kielenkäytöltä. Lähestymistapa mahdollistaa tekstigenerointiin opetetun mallin soveltamisen moniin eri tehtäviin, ja esimerkiksi Wang et al. (2023) ovat osoittaneet esikoulutettujen generatiivisten kielimallien soveltuvan erinomaisesti myös monenlaisiin tekstiluokittelutehtäviin. Generatiivisen decoder-only transformer -kielimallin perusarkkitehtuuri on nähtävissä kuvassa 2.3.

Kuva 2.3: GPT-arkkitehtuuri (Radford et al., 2018)



Tekstiluoittelutehtävissä generatiiviselle kielimallille annettava syöte muotoillaan luonnollisella kielellä esitettäväksi päättelytehtäväksi. Mallia toisin sanoen pyydetään luokittelemaan saamansa syöte johonkin mahdollisista luokista. Erillistä jatkokouluttamista tai opetusaineiston keräämistä, esikäsittelyä, annotointia ei välttämättä tarvita, vaan luokiteltava aineisto voidaan syöttää kielimallille sellaisenaan ohjeistuksen osana. Lähestymistapa on yksinkertainen, tehokas ja helposti sovellettavissa monenlaisiin luokittelutehtäviin. (Laato et al., 2024; Tarkka et al., 2024; Burnham et al., 2024; Wang et al., 2023.) Vuokaavio generatiivisella kielimallilla tehtävästä tekstiluokittelussa on nähtävissä alla kuvassa 2.4.

Kuva 2.4: Generatiivisella kielimallilla tehtävä tekstiluokittelu (Wang et al. 2023)



Tekstiluoittelutehtävissä käytettävät syötteet eli kehotteet (*prompt*) voidaan jakaa kahteen päätyyppiin. Zero-shot-kehotteissa malli ohjataan suorittamaan tehtävä pelkän sanallisen kuvauksen perusteella. Lähestymistavassa luotetaan siihen, että kielimalli ymmärtää

tehtävänannon ja siltä odotetun tulosteen ja sen muotoilun tai formaatin kerrasta. Sen sijaan few-shot-kehotteissa kielimallille annetaan tehtävänannon lisäksi muutamia esimerkkejä toivotusta lopputuloksesta. Ajatuksena on, että malli suoriutuu tehtävästä paremmin, kun sille näytetään, mitä siltä odotetaan. (Brown et al., 2020; Min et al., 2022.)

Niin zero-shot- kuin few-shot-asetelmissä kehotteen sisältö, rakenne ja muotoilu ovat ratkaisevan tärkeitä (Liu, P. et al., 2023). Niiden rooli korostuu entisestään, mitä vaativammasta tehtävästä on kyse (Wei et al., 2022a). Erilaisista kehotestrategioista onkin käyty vilkasta keskustelua viime vuosina (Schulhoff et al., 2024; Kojima et al., 2022; Khattab et al., 2023). Tämän tutkielman kannalta erityisen mielenkiintoinen kehotestrategia on ketjupäätely (*Chain-of-Thought*), jossa kielimallia ohjeistetaan lähestymään tehtävää askeleittain (*step-by-step*). Lähestymistavan on osoitettu parantavan kielimallien suorituskykyä paitsi matemaattisloogisissa myös monimutkaista kielellistä päättelyä vaativissa tehtävissä (Wei et al., 2022a; Kojima et al., 2022).

Esikoulutettujen generatiivisten kielimallien monipuoliset kyvyt eivät ole jääneet huomiotta arkistoalallakaan (Aangenendt, 2022). Esimerkiksi Zhang et al. (2023) ovat kehittäneet seulontaan, arvonmäärittämiseen ja yleiseen arkistokäyttöön erikoistunutta ArcGPT-kielimallia, kun taas Yang, J. et al. (2023) ovat tutkineet kiinan ja englannin kielillä koulutetun ChatGLM2-6B-mallin käyttämistä henkilötietojen tunnistamisessa kiinankielisestä arkistoaineistosta. Tulokset ovat olleet lupaavia: ChatGLM2-6B tunnisti henkilötietoja kolmesta eri testiaineistosta keskimäärin 90,24 % osuvuudella, 91,64 % tarkkuudella, 88,63 % kattavuudella ja 90,09 % F1-arvolla. Käsittelen arviointimetriikoita tarkemmin luvussa 4.3.

3 Aineisto

Tässä luvussa käsitellään tutkielmassa käytettyä luokitteluaineistoa. Ensimmäisessä alaluvussa esitellään Jahvetin kirjelaatikko -arkistoa. Luvussa kuvaillaan arkistokokonaisuuden syntyä, kokoa ja sisältöä. Toisessa alaluvussa käsitellään asiakirjojen tekstisisältöjen poimimista, esikäsittelyä, annotintia sekä koostamista kahdeksi erilliseksi osakokonaisuudeksi. Lopuksi luvussa esitellään osakokonaisuuksien eli kehitys- ja testiaineistoon sisältyvien dokumenttien pituuksia, henkilötietojen määriä ja niiden jakaumaa.

3.1 Luokitteluaineisto

Tämän tutkielman luokitteluaineistosta valtaosa kuuluu Jahvetin Kirjelaatikko -nimiseen arkistoon. Jahvetin kirjelaatikko oli Yrjö Kilpeläisen ja hänen toimistonsa tuottama radio-ohjelma. Ensimmäinen lähetys lähetettiin 23.8.1941 ja viimeinen neljä ja puoli vuotta myöhemmin loppiaisena 1945. Ohjelmassa pyrittiin vastaamaan erilaisiin kansalaisia askarruttaneisiin kysymyksiin, ja kuulijakirjeitä saapui noin 100 000. Noin kolmasosaan vastattiin radioaalloilla ja 70 000 lähetettiin postitse. (Kulha 2021; Lehtinen 2006.)⁷ Kuulijakirjeiden lisäksi Jahvetin kirjelaatikko -arkisto sisältää Jahvetin bulletiineiksi kutsuttuja yhteenvetoja saapuneista kirjeistä sekä ulkomaisten medioiden seurannassa syntyneitä muita aineistoja, joita ei sisälly tämän tutkielman luokitteluaineistoon. Jahvetin kirjelaatikko -arkisto koko on yhteensä noin 18 hyllymetriä.⁸

Yksi Jahvetin kirjelaatikko -arkiston perusominaisuuksista on sen syntyajankohta. Yleisimpiä kuulijakirjeissä esiintyviä aiheita ovat hautausavustukset, asepalveluksen korvaukset, rajariidat, asuntokysymykset, salakauppa, säännöstely, työriidat ja maanviljely. Kuulijakirjeissä esiintyy runsaasti länsi- ja itämurteiden eri variantteja ja kielen rekistereitä, ja niiden syntaktinen vaihtelu on runsasta. Vastauskirjeitä puolestaan laativat Lehtisen mukaan (2006) valtionhallinnon ja eri alojen asiantuntijat, ja niissä esiintyykin kielellistä vaihtelua verrattain vähän. Käytännössä katsoen kaikki vastauskirjeet on laadittu huolitellulla yleiskielellä.

⁷ Jahvetin kirjelaatikko -arkistosta enemmän ks. myös <https://portti.kansallisarkisto.fi/fi/aineisto-oppaat/jahvetin-kirjelaatikko>

⁸ Hyllymetri on arkistoinnissa ja asiakirjahallinnassa käytetty suure. Yksi hyllymetri vastaa keskimäärin 10 000 asiakirjasivua.

Koonsa puolesta Jahvetin kirjelaatikko -arkisto on suurehko aineistokokonaisuus. Sisältönsä ja käyttörajoituksensa puolesta se on sen sijaan varsin tyypillinen. Useimmissa sen sisältämissä asiakirjoissa on yksityishenkilöihin yhdistettävissä olevia arkaluonteisia tietoja. Näistä tyypillisimpiä ovat yksityishenkilöiden nimi- ja osoitetiedot sekä talous-, terveys- tai rikos- ja rangaistustiedot. Kysymyskirjeet on lähes poikkeuksetta kirjoitettu käsin ja allekirjoitettu nimellä, kun taas vastauskirjeet ovat konekirjoitettuja ja allekirjoittamattomia.

Luokitteluaineisto sisältää Jahvetin kirjelaatikko -arkiston dokumenttien lisäksi katkelmia Kansallisarkiston antamista seulontapäätöksistä. Ne ovat virallisia hallinnollisia asiakirjoja, joissa määritellään, mitkä julkishallinnon tuottamat tietoaineistot säilytetään pysyvästi ja mitkä hävitetään määräajan kuluttua. Kansallisarkiston antamat seulontapäätökset sisältävät päätöksen antaneiden viranhaltijoiden nimet ja tehtävänimikkeet, mutta koska ne liittyvät virantoimitukseen ja viranomaisen roolissa annettuihin viranomaispäätöksiin, niitä ei tulkita arkaluonteisiksi. Seulontapäätöksissä esiintyy kielellistä vaihtelua vain vähän, ja ne on poikkeuksetta kirjoitettu huolitellulla yleiskielellä. Tässä tutkielmassa käytetyt seulontapäätökset ovat enintään kymmenen vuotta vanhoja.

3.2 Esikäsittely

Digitoitujen Jahvetin kirjelaatikko -arkiston kirjeiden tekstisisällöt poimittiin kahdella eri tekstintunnistusmallilla. Vastauskirjeisiin käytettiin Transkribus Print M1 -mallia, joka on koulutettu erityisesti konekirjoitetun tekstin tunnistamiseen. Transkribus Print M1 tukee yhteensä 16 eurooppalaista kieltä, ja sen merkkivirhetaajuus on 2,2 %.⁹ Mallia pääsee kokeilemaan Transkribus-verkkopalvelussa, jota ylläpitää eurooppalaisten yliopistojen, arkistojen ja instituutioiden muodostama yhteistyöorganisaatio.¹⁰ Kysymyskirjeisiin puolestaan käytettiin Kansallisarkiston Tutkimus ja innovaatiot -toiminnon kehittämää Multicentury-HTR-tekstintunnistusmallia. Multicentury-HTR on koulutettu erityisesti käsin kirjoitetun suomen- ja ruotsinkielisen tekstin tunnistamiseen, ja sen merkkivirhetaajuus on 3,2 %. Malli on jatkuvassa kehityksessä, ja sitä pääsee tämän tutkielman kirjoitusajankohtana kokeilemaan Kansallisarkiston HuggingFace-sivustolla ylläpidetyssä demosovelluksessa.¹¹

⁹ Ks. <https://readcoop.eu/model/transkribus-print-multi-language-dutch-german-english-finnish-french-swedish-etc/>

¹⁰ <https://www.transkribus.org/>

¹¹ <https://huggingface.co/spaces/Kansallisarkisto/Multicentury-HTR-Demo>

Sisältötunnistettujen dokumenttien esikäsittely pidettiin vähäisenä. Transkriptioista poistettiin rivinvaihdot ja ylimääräiset välilyönnit, minkä jälkeen dokumentit annotoitiin kahteen mahdolliseen luokkaan: *julkiseen* tai *arkaluonteisia tietoja* sisältäviin. Annotoinnin suoritti Kansallisarkiston työntekijä Kansallisarkiston sisäisten ohjeiden mukaisesti. Lopullinen tutkielmassa käytetty aineisto sisälsi yhteensä 300 dokumenttia, joista 202 oli peräisin Jahvetin kirjelaatikko -arkistosta ja 98 kappaletta Kansallisarkiston seulontapäätöksistä. Yhteensä 167 (55,67 %) dokumenttia annotoitiin arkaluonteisia tietoja sisältäviksi ja 133 (44,33 %) julkisiksi.

Annotoinnin keskeiset haasteet voidaan jakaa kahteen kategoriaan. Ensinnäkin Jahvetin kirjelaatikko -arkiston dokumenteissa esiintyneiden viranhaltijoiden tai muuten virantoimitukseen liittyvien erisnimien tunnistaminen osoittautui haastavaksi. Erityisesti kysymyskirjeissä saatettiin mainita viranhaltijan tai muussa julkisessa tehtävässä toimivan henkilön nimi ilman tehtävänimikettä tai muuta vastaavaa tunnustetta. Tällöin henkilön rooli tai asema saattoi jäädä tulkinnanvaraiseksi. Toisaalta epäsuorien tunnisteiden arvioiminen osoittautui myös haastavaksi. Jahvetin kirjelaatikko -ohjelmaan lähetettiin kirjeitä kaikkialta Suomesta – niin kaupungeista kuin pienistä maaseudun kylistä –, ja esimerkiksi kylän nimi yhdistettynä tietoon kirjoittajan sukupuolesta tai ammatista saattaa mahdollistaa tunnistamisen. Molemmissa tapauksessa lopulliseksi luokaksi määriteltiin varmuuden vuoksi arkaluonteinen henkilötieto

Annotoinnin jälkeen aineisto jaettiin kahteen erilliseen osioon. Pienempi osio eli kehitysaineisto sisälsi yhteensä 100 dokumenttia, joista 70 oli Jahvetin kirjelaatikko -arkistosta ja 30 seulontapäätöksistä. Kehitysaineistoa käytettiin vain luokittelun tekniseen kehittämiseen, ja sen dokumenteista yhteensä 50 % sisälsi arkaluonteisia tietoja. Vastaavasti suurempi osio eli testiaineisto sisälsi yhteensä 200 dokumenttia, joista 152 oli Jahvetin Kirjelaatikko -arkistosta ja 48 Kansallisarkiston seulontapäätöksistä. Testiaineistoa käytettiin vain lopullisessa suorituskykyarvioinnissa, ja sen dokumenteista 117 kappaletta eli 58,5 % sisälsi arkaluonteisia tietoja. Vastaavasti 83 dokumenttia eli 41,5 % testiaineiston dokumenteista oli määritelty julkisiksi. Luokitteluaineiston tarkemmat tiedot ovat nähtävissä alla olevassa taulukossa 3.1.

Taulukko 3.1: luokitteluaineisto

	Trainset	Testset	Yhteensä
Dokumentit (kpl)	100	200	300
Julkinen	50 (50 %)	83 (41,50 %)	133 (44,33 %)
Arkaluonteinen tieto	50 (50 %)	117 (58,50 %)	167 (55,67 %)
Lyhyin dokumentti (merkkiä)	187	150	150
Pisin dokumentti (merkkiä)	1515	1662	1662
Keskimääräinen dokumentti (merkkiä)	701	681	687

4 Menetelmät

Tässä luvussa käsitellään tämän tutkielman teknistä toteutusta. Ensimmäisessä alaluvussa eritellään luokittelijan teknisiä ratkaisuja, DSPy-ohjelmointikirjastolla suoritettua luokittelijan optimointia sekä luokittelussa käytettyjen kehoitteiden rakennetta. Toisessa luvussa esitellään luokittelussa kokeiltuja kielimalleja sekä niiden teknisiä tietoja. Kolmannessa alaluvussa esitellään luokittelua kuvaavia arviointimittareita, joista yhtä käytettiin luokittelijan optimoinnissa ja muita varsinaisen luokittelun laadullisessa arvioimisessa.

4.1 Luokittelija

Tämän tutkielman tekninen toteutus perustuu DSPy-ohjelmointikirjastoon.¹² DSPy on deklaraatiivinen ohjelmointikehys, jolla voidaan ohjata ja optimoida generatiivisia kielimalleja. Tiivistetysti DSPy toimii siten, että käyttäjä määrittelee halutun lopputuloksen ja parametrit, kun taas DSPy vastaa tehtävän suorittamisesta. (Khattab et al., 2023.)

DSPy-ohjelmissa on kolme keskeistä komponenttia. Allekirjoitus (*signature*) on tehtäväkohtainen määrittely, jossa kuvataan mitä kielimallin tulee tehdä. DSPy:ssä on useita valmiita allekirjoituksia, minkä lisäksi käyttäjä voi itse määritellä, millaisia syötteitä kielimallille annetaan ja mitä tulosteita sen odotetaan tuottavan. Moduuli (*module*) on ohjelmakomponentti, joka suorittaa allekirjoituksen ja käyttäjän määrittelemien parametrien perusteella tehtävää. Toisin sanoen moduuli vastaa siitä, miten kielimalli suorittaa sille annettua tehtävää. Optimoija (*optimizer*) puolestaan vastaa ohjelman automaattisesta optimoinnista: se ohjaa ohjelman suoritusta, tuottaa vaihtoehtoisia kehoitteita ja valitsee niistä parhaimman käyttäjän määrittelemän arviointimetriikan ja kehitysaineistosta lohkaistun validointiaineiston perusteella. (Khattab et al., 2024.)

Tässä tutkielmassa käytetty allekirjoitus sisälsi tehtävän kuvauksen sekä kaksi syöte- ja kaksi tulostekenttää. Tehtäväkuvauksessa kielimallia ohjeistettiin luokittelemaan asiakirja joko luokkaan `public_data` tai `sensitive_data`. Syötekentät olivat `text` ja `categories`, joista ensimmäinen sisälsi luokiteltavan asiakirjan tekstin ja jälkimmäinen luokkien kuvaukset. Vastaavasti tulostekenttiä olivat `reasoning` ja `label`, joista ensimmäiseen kielimallin odotettiin tuottavan perustelun ja jälkimmäiseen valitsemansa luokan. Allekirjoitus on esitetty kuvassa 5.1 ja luokkien kuvaus kuvassa 5.2.

¹² <https://dspy.ai/>

Kuva 5.1: Luokittelijan DSPy-allekirjoitus

```
class ClassificationSignature(dspy.Signature):
    """Classify historical documents from the Finnish National Archives as either
    'public_data' or 'sensitive_data'. Base your decision on the GDPR guidelines and
    administrative archival context. Use the classification category definitions to
    justify your decision with step-by-step reasoning. The document will be in
    Finnish. Always return both a justification and a final label."""

    text: str = dspy.InputField(
        prefix="Document",
        desc="The document to classify."
    )
    categories: str = dspy.InputField(
        prefix="Classification Categories",
        desc="Description of classification categories."
    )
    reasoning: str = dspy.OutputField(
        prefix="Reasoning",
        desc="Step-by-step reasoning about whether the document fits"
            "'public_data' or 'sensitive_data'. "
    )
    label: Literal['public_data', 'sensitive_data'] = dspy.OutputField(
        prefix="Label",
        desc="Final classification label."
    )
```

Kuva 5.2: Mahdollisten luokkien kuvaukset****sensitive_data****

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile
- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")
- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data
- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Luokittelijan moduulina käytettiin DSPy-ohjelmointikirjaston valmista ChainOfThought-moduulia. Moduuli ohjaa kielimallin generoimaan valitsemalleen luokalle allekirjoituksessa määritellyn perustelun. Ketjupäätelyn hyödyntämisen kantavana ajatuksena oli, että luokitteluprosessin eksplikointi lisää oikein luokiteltujen dokumenttien määrää, parantaa luokittelun läpinäkyvyyttä ja helpottaa luokittelun jälkeistä virheanalyysiä. Esimerkki kielimallin tuottamasta ketjupäätelystä on nähtävissä kuvassa 5.3.

Kuva 5.3: Esimerkki kielimallin generoimasta päätelystä. Henkilötiedot on poistettu jälkikäteen.

```
<think>
Okay, let's tackle this classification. The document in question is a letter addressed to "Jahvetin Kirjelaatikko" in Helsinki, with a reference number XX/XX/XX. The sender is XXXX XXXX from XXXX, XXXX. The content asks about a daughter who was in a prison, inquiring if she's still there or has been released and where she might have gone.

First, I need to check for personal data. The sender's name is XXXX XXXX, which is a full name. The address is XXXX XXXX. That's a specific address. The letter mentions a daughter, but the daughter's name isn't provided. The prison is referred to generally, without a specific name or location.

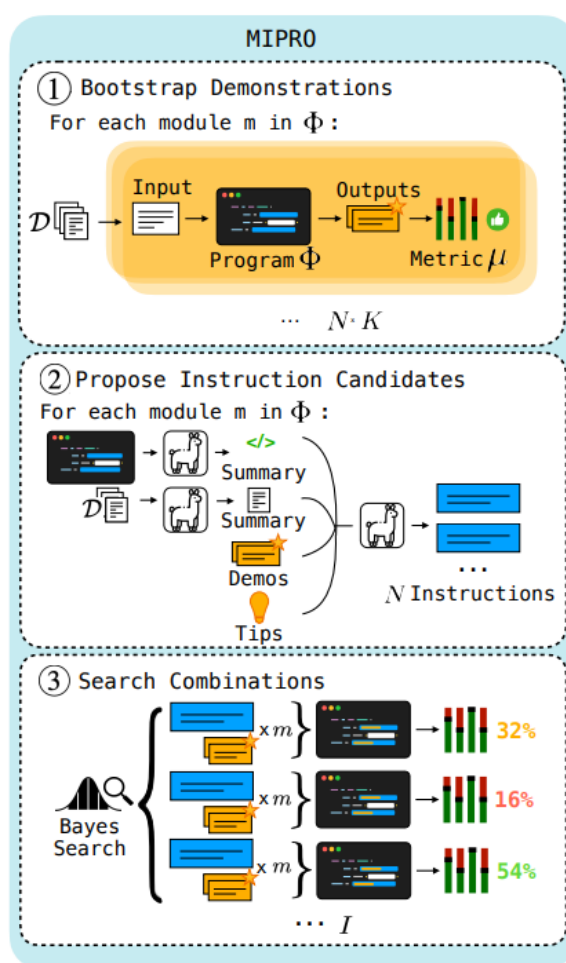
Looking at the categories, sensitive_data includes names, addresses, and data relating to criminal convictions. The sender's name and address are present, which are direct identifiers. The mention of a daughter in prison could relate to criminal data, but without the daughter's name or specific details, it's not clear. However, the sender's personal information is there, which is enough to classify as sensitive.

Public_data examples include anonymized names or general data without identifiers. Here, the sender is identified by name and address, so it's not public_data. The reference to a prison might touch on criminal data, but the main identifiers are the sender's details. Therefore, this should be sensitive_data.
</think>

[[ ## reasoning ## ]]
The document contains explicit personal identifiers: the sender's full name (XXXX XXXX) and address (XXXX XXXX). It also references a family member (a daughter) and a prison, which could imply sensitive information about criminal convictions or incarceration. While the daughter's name and specific prison details are not provided, the sender's identifiable information alone qualifies this as sensitive_data under GDPR criteria, as it directly identifies an individual and involves potentially sensitive topics (prison status).
```

Luokittelijan toimintaa optimoitiin MIPROv2-optimoijalla (*Multiprompt Instruction Proposal Optimizer Version 2*). MIPROv2 muodostaa allekirjoituksessa määritellylle tehtävänannolle useita vaihtoehtoisia muotoiluja ja valitsee joko todellisia tai synteettisiä esimerkkitapauksia eli demoja sen tueksi. Tämän jälkeen se kokeilee eri tehtävänantovarianttien ja esimerkkitapausten yhdistelmä iteratiivisesti ja arvioi niiden tehokkuutta tavoitellun lopputuloksen näkökulmasta. Arviointimittarina käytetään käyttäjän määrittelemää metriikkaa (Opsahl-Ong et al., 2024.), joka tässä tutkielmassa oli osuvuus eli *accuracy*. MIPROv2-optimoijan toimintaa kuvaa alla oleva kuva 5.3.

Kuva 5.4: MIPRO-optimoijan toiminta (Opsahl-Ong et al., 2024)



Tässä tutkielmassa käytettiin MIPROv2-optimoijan automaattista konfigurointia niin sanotussa ”heavy”-tilassa. Tällöin MIPROv2 valitsee sopivimman tehtävänannon ja parhaat esimerkkidokumentit automaattisesti. Sekä zero-shot- että few-shot-asetelmissa vaihtoehtoisten tehtävänantojen ja esimerkkien yhdistelmien määrä oli 50. Zero-shot-asetelmassa tehtävänannon mukana annettavien esimerkkien määrä oli sananmukaisesti nolla,

kun taas few-shot-asetelmassa esimerkkien määrä oli rajoitettu viiteen (n=5). Lopullisen kehotteen rakenteellinen kuvaus on nähtävissä alla kuvassa 5.5.

Kuva 5.5: Rakenteellinen kuvaus DSPy:n muodostamasta kehotteesta. Kehotteessa kielimallille kerrotaan sen saamat syötekentät ja siltä odotetut tulostekentät. Kehote sisältää myös tehtäväohjeistuksen.

```
Your input fields are:
```

1. `text` (str): The document to classify.
2. `categories` (str): Description of classification categories.

```
Your output fields are:
```

1. `reasoning` (str): Step-by-step reasoning about whether the document fits 'public_data' or 'sensitive_data'.
2. `label` (Literal['public_data', 'sensitive_data']): Final classification label.

```
All interactions will be structured in the following way, with the appropriate values filled in.
```

```
[[ ## text ## ]]
```

```
{text}
```

```
[[ ## categories ## ]]
```

```
{categories}
```

```
[[ ## reasoning ## ]]
```

```
{reasoning}
```

```
[[ ## label ## ]]
```

```
{label}          # note: the value you produce must exactly match (no extra characters)  
one of: public_data; sensitive_data
```

```
[[ ## completed ## ]]
```

```
In adhering to this structure, your objective is:
```

```
    Classify the provided Finnish historical document as either 'public_data' or  
'sensitive_data' based on GDPR guidelines and administrative archival context. Justify  
your decision by analyzing identifiers (e.g., names, addresses, personal  
circumstances) and whether the content involves personal data requiring protection.
```

4.2 Kielimallit

Arkaluonteisia tietojen tunnistamista kokeiltiin seitsemällä eri kielimallilla. Malleja ajettiin paikallisesti Kansallisarkiston laitteistolla 4- tai 8-bittiseen painokerrointarkkuuteen kvantisoituina.¹³ Inferenssi toteutettiin Ollama-ohjelmistokehyksen¹⁴ rajapinnan kautta. Mallien valinta perustui Kansallisarkiston laitteiston kyvykkyyteen, mallien vapaaseen saatavuuteen sekä niiden kykyyn generoida suomenkielistä tekstiä.

Llama 3.1 8B on Metan kehittämä generatiivinen kielimalli. Se perustuu decoder-only transformer -arkkitehtuuriin, ja se tukee useita eri kieliä. Llama 3.1 on esikoulutettu 15 biljoonalla tokenilla, ja sen konteksti-ikkuna¹⁵ on 128 000 tokenia. Mallista on olemassa 8 miljardin, 70 miljardin ja 405 miljardin parametrin variantit (Dubey et al. 2024.) Tässä tutkielmassa käytetään 4 bitin painokerrointarkkuuteen kvantisoitua 8 miljardin parametrin versiota.

Llama 3.3 70B on Metan kehittämä generatiivinen kielimalli. Se perustuu decoder-only transformer -arkkitehtuuriin, ja se on esikoulutettu 15 biljoonalla tokenilla. Llama 3.3 on jatkokoulutettu versio edellä esitellystä Llama 3.1 -mallista, ja sen konteksti-ikkuna on niin ikään 128 000 tokenia. Mallista on olemassa vain 70 miljardin parametrin versio. (Dubey et al. 2024.). Tässä tutkielmassa käytettiin 4 bitin painokerrointarkkuuteen kvantisoitua versiota.

Qwen 3 8B / 32B on Alibaba Cloudin kehittämä generatiivinen kielimalli. Se perustuu decoder-only transformer -arkkitehtuuriin, ja se tukee useita eri kieliä. Qwen 3 -mallit on koulutettu 18 biljoonalla tokenilla, ja niiden konteksti-ikkuna on 128 000 tokenia. Qwen 3 -mallit ovat niin sanottuja reasoning-malleja, jotka hyödyntävät ketjupäätelyä automaattisesti vastatessaan käyttäjän kehoitteeseen. Huomion arvoista Qwen 3 -malleissa on, että ne tukevat virallisesti suomen kieltä. (Yang, A. et al., 2025.) Tässä tutkielmassa käytetään sekä 8 miljardin että 32 miljardin parametrin versioita 8 bitin painokerrointarkkuuteen kvantisoituina.

Deepseek-R1 8b / 70B on Deepseekin kehittämä generatiivinen kielimalli. Se perustuu decoder-only transformer -arkkitehtuuriin, ja se on jatkokoulutettu versio Metan Llama 3 -

¹³ Kvantisoinnissa mallin painokertoimien esitystarkkuutta pienennetään, mikä vähentää tarvittavan muistin määrää ja nopeuttaa inferenssiä. Ks. esim. Li et al., 2023.

¹⁴ Ks. <https://github.com/ollama/ollama>

¹⁵ Konteksti-ikkunalla (*context window*) tarkoitetaan tekstin pituutta, jonka kielimalli pystyy huomioimaan yhdellä kertaa.

sarjan malleista. Deepseek-R1 -mallista on olemassa 7 eri versiota, joista pienin sisältää 1,5 miljardia ja suurin 671 miljardia parametria. Mallit ovat monikielisiä, ja niiden kontekstikkuna on 128 000 tokenia. Myös Deepseek-R1-mallit ovat niin sanottuja reasoning-malleja. (DeepSeek-AI 2025). Tässä tutkielmassa käytetään sekä 8 miljardin että 70 miljardin parametrin versioita 4 bitin parametritarkkuuteen kvantisoituna.

Poro 34B Chat on LumiOpen-tutkimusryhmän kehittämä generatiivinen kielimalli. Se perustuu decoder-only transformer -arkkitehtuuriin, ja se on koulutettu englannin ja suomen kielellä sekä usealla eri ohjelmointikielellä. Se sisältää yhteensä 34 miljardia parametria, ja se on koulutettu noin biljoonalla tokenilla. (Luukkonen et al. 2024.) Tässä tutkielmassa käytetään 4 bitin painokerrointarkkuuteen kvantisoitua versiota Poro 34B Chat -mallista.

4.3 Arviointi

Luokittelutehtävissä *accuracy* eli osuvuus on tavallinen suorituskykyä kuvaava mittari. Se kuvaa, kuinka moni luokitelluista dokumenteista luokiteltiin oikein, ja se lasketaan jakamalla oikein luokiteltujen tapausten määrä kaikkien tapausten määrällä seuraavasti:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Tässä tutkielmassa osuvuus ei riitä kuvaamaan luokittelun onnistumista. Syynä on, että tämän tutkielman luokitteluaineisto sisältää hieman enemmän arkaluonteisia tietoja sisältäviä dokumentteja kuin julkisia. Niinpä osuvuutta käytetään ennen kaikkea MIPROv2-optimojan optimointimetriikkana, ja varsinaisessa luokittelussa kielimallien suorituskykyä arvioidaan osuvuuden lisäksi tarkkuudella, kattavuudella, F1-arvolla ja macro F1 -arvolla.

Precision eli tarkkuus lasketaan jakamalla todellisten positiivisten dokumenttien määrä todellisten ja väärin positiivisten dokumenttien yhteismäärällä (Powers, 2020). Toisin sanoen tarkkuus kuvaa oikein tunnistettujen arkaluonteisia tietoja sisältävien dokumenttien osuutta kaikista mallin arkaluonteisiksi luokittelemista tapauksista. Precision lasketaan seuraavalla tavalla:

$$Precision = \frac{TP}{TP + FP}$$

Recall eli kattavuus puolestaan lasketaan jakamalla todellisten positiivisten dokumenttien määrä todellisten positiivisten ja havaitsematta jääneiden positiivisten dokumenttien määrällä

(Powers, 2020). Kattavuus kuvaa, kuinka suuri osa kaikista arkaluonteisia tietoja sisältävistä asiakirjoista tunnistettiin onnistuneesti. Käytännön tasolla esimerkiksi 50 % kattavuus tarkoittaa, että kaikista arkaluonteisia tietoja sisältävistä dokumenteista tunnistettiin puolet. Kattavuus lasketaan seuraavalla tavalla:

$$Recall = \frac{TP}{TP + FN}$$

F1 score eli F1-arvo on tarkkuuden ja kattavuuden harmoninen keskiarvo, ja se on erityisen hyödyllinen epätasapainoisten aineistojen arvioinnissa (Powers, 2020). Tässä tutkielmassa käytetään kahta F1-arvoa. Toinen lasketaan luokkakohtaisesti, kun taas *macro F1* -arvossa eli F1-keskiarvossa lasketaan keskiarvo molempien luokkien F1-arvoista. Luokkakohtainen F1-arvo sekä *macro F1* -arvo lasketaan seuraavilla tavoilla:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

$$Macro\ F1 = \frac{1}{C} \sum_{i=1}^C 2 \frac{Precision_i \cdot Recall_i}{Precision_i + Recall_i}$$

Tässä tutkielmassa tuloksia havainnollistetaan paitsi tulostaulukoilla myös sekaannusmatriiseilla. Sen solut sisältävät oikeiden positiivisten, oikeiden negatiivisten, väärin positiivisten ja väärin negatiivisten ennusteiden lukumäärät, ja sen avulla on helppo havainnoida, luokitteleeke malli dokumentteja esimerkiksi arkaluonteisia tietoja sisältäviksi liian herkästi. Alla oleva taulukko 3.1 havainnollistaa sekaannusmatriisia ja sen lukemista.

Taulukko 3.1: Sekaannusmatriisi (Jurafsky & Martin, 2025).

	Positive	Negative	
Positive	TP	FP	$Precision = \frac{TP}{TP + FP}$
Negative	FN	TN	
	$Recall = \frac{TP}{TP + FN}$		$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$

5 Analyysi

Tässä luvussa käsitellään tuloksia. Ensimmäisessä alaluvussa esitellään eri kielimallien suoriutumista neljässä erilaisessa koeasetelmassa. Alaluku vastaa tutkielman ensimmäiseen tutkimuskysymykseen. Toisessa alaluvussa puolestaan analysoidaan kielimallien tekemiä virheitä ja pohditaan mahdollisia syitä niille. Toinen alaluku vastaa tämän tutkielman toiseen tutkimuskysymykseen.

5.1 Tulokset

Seuraavissa tulostaulukoissa esitellään eri kielimalleilla saatuja tuloksia eri koetilanteissa. Taulukkojen vasemmanpuoleisin pystyrivi ilmaisee kielimallin ja ylin vaakarivi arviointimetriikan. Arviointimetriikoista accuracy, precision, recall ja F1 score kuvaavat kunkin kielimallin suorituskykyä arkaluonteisten tietojen tunnistamisessa, kun taas Macro F1 eli F1-keskiarvo kuvaa kielimallin kykyä tunnistaa sekä arkaluonteisia tietoja sisältävät dokumentit että julkiseksi määritellyt dokumentit. Yksittäiset solut kuvaavat kunkin mallin suorituskykyä solun sijaintia vastaavassa metriikassa. Suurempi arvo merkitsee parempaa suorituskykyä.

Kaikissa koeasetelmissa kielimallien lämpötila-arvoksi (*temperature*) säädettiin 0,0. Parametri ohjaa mallin tuottamien vastausten satunnaisuutta. Mitä matalampi parametrin arvo on, sitä deterministisempi tuloste on.¹⁶ Rajallisen laskentaresurssiajan takia luokittelu tehtiin kullakin kielimallilla kussakin koeasetelmassa vain kaksi kertaa. Tulokset olivat molemmilla kerroilla identtisiä.

5.1.1 Luokittelun tulokset ilman optimointia

Taulukko 5.1 havainnollistaa eri kielimallien tuloksia kahdessa erilaisessa zero-shot-koeasetelmassa. Ensimmäisessä asetelmassa kielimallia ei pyydetty generoimaan perusteluita päätökselleen. Toisessa asetelmassa kielimalleja päinvastoin pyydettiin generoimaan perustelut päätökselleen. Kummassakaan koeasetelmassa kielimalleille ei annettu esimerkkidokumentteja. Koska Qwen 3 -mallit sekä Deepseek-R1-mallit ovat niin sanottuja reasoning-malleja, jotka generoivat automaattisesti tehtävänantoa analysoivaa tekstiä ennen

¹⁶ Renze et al. 2024 ovat argumentoineet, että yleisestä käsityksestä huolimatta lämpötila-arvo ei tilastollisesti tarkasteltuna vaikuta kielimallien ongelmanratkaisukykyihin. Peeperkorn et al. 2024 ovat myös argumentoineet samansuuntaisesti.

varsinaista vastausta, niiden suorituskkyä arvioitiin vain toisessa asetelmassa. Llama 3 -mallien ja Poro 34B Chat -mallin osalta Zero-Shot-koeasetelma toimi suorituskyyvyn perustasona, kun taas Qwen- ja Deepseek-mallien kohdalla perustasona toimi Zero-Shot w/ CoT -koeasetelma.

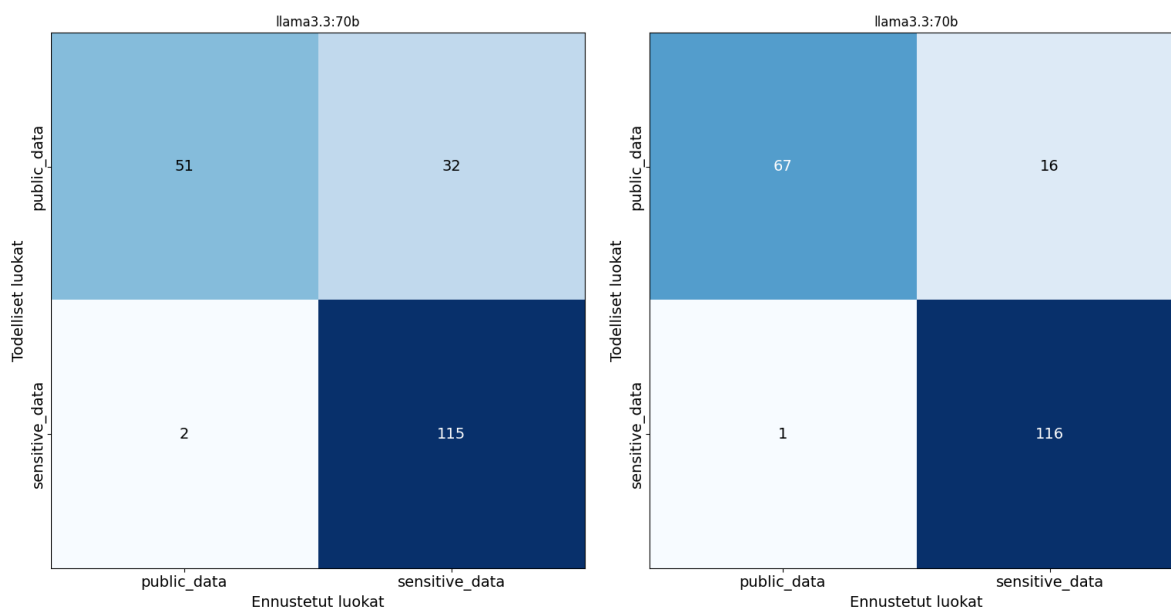
Taulukko 5.1: Eri kielimallien luotettavuus zero-shot-asetelmissa

Model	Accuracy	Precision	Recall	F1	Macro F1
<i>Zero-Shot</i>					
Llama 3.1 8B	59,50	59,18	99,15	74,12	40,51
Llama 3.3 70B	83,00	78,23	98,29	87,12	81,06
Qwen 3 8B	-	-	-	-	-
Qwen 3 32B	-	-	-	-	-
Deepseek-R1 8B	-	-	-	-	-
Deepseek-R1 70B	-	-	-	-	-
Poro 34B Chat	57,50	58,08	98,29	73,02	36,51
<i>Zero-Shot w/ CoT</i>					
Llama 3.1 8B	66,50	70,16	74,36	72,20	65,03
Llama 3.3 70B	91,50	87,88	99,15	93,17	90,96
Qwen 3 8B	81,00	89,11	76,92	82,57	80,84
Qwen 3 32B	86,50	88,79	88,03	88,41	86,12
Deepseek-R1 8B	78,00	77,86	87,18	82,26	76,66
Deepseek-R1 70B	89,00	85,71	97,44	91,20	88,27
Poro 34B Chat	57,50	58,16	97,44	72,84	37,57

Zero-shot-koeasetelmassa Llama 3.3 70B suoriutui odotetusti paremmin kuin parametrimäärältään pienemmät Llama 3.1 8B ja Poro 34B Chat. Kaikista arkaluonteisiksi annotoiduista dokumenteista malli tunnisti lähes kaikki, minkä lisäksi se kykeni erottamaan arkaluonteiset dokumentit julkisista kohtalaisen luotettavasti. Sen sijaan Llama 3.1 8B ja Poro 34B eivät kyenneet erottamaan dokumentteja luokittain, vaan molemmat ennustivat lähes jokaisen saamansa dokumentin arkaluonteiseksi sen todellisesta sisällöstä huolimatta. Parhaiten suoriutuneen Llama 3.3 70B -mallin sekaannusmatriisi on nähtävissä seuraavalla sivulla kuvassa 5.2.

Llama 3.3 70B suoriutui niin ikään parhaiten zero-shot w/ CoT -koeasetelmassa. Se tunnisti kaikista arkaluonteisiksi annotoiduista dokumenteista 99,15 prosenttia, minkä lisäksi se kykeni erottelmaan arkaluonteiset dokumentit julkisista kiitettävästi. Huomion arvoista on se, että ketjupäätely paransi erityisesti Llama-mallien suorituskkyä, ja esimerkiksi Llama 3.1 8B -mallin suorituskky nousi F1-keskiarvolla mitattuna lähes 25 prosenttiyksikköä. Parhaiten suoriutuneen Llama 3.3 70B -mallin sekaannusmatriisi on nähtävissä alla kuvassa 5.2.

Kuva 5.2: Optimoimattomissa koeasetelmissa parhaiten suoriutuneen Llama 3.3:70B -mallin sekaannusmatriisit. Vasemmalla Zero-Shot-koeasetelma ja oikealla Zero-Shot w/ CoT -koeasetelma.



5.1.2 Luokittelun tulokset optimoiduilla kehoiteilla

Taulukko 5.3 havainnollistaa eri kielimallien tuloksia kahdessa erilaisessa MIPROv2-optimoidussa koeasetelmassa. Ensimmäinen koeasetelma oli zero-shot-asetelma, jossa MIPROv2-optimoijalla optimoitiin vain kielimallille annettua tehtävänantoa. Toinen koeasetelma oli few-shot-asetelma, jossa MIPROv2-optimoijalla optimoitiin paitsi tehtävänantoa myös kielimallille annettavia esimerkkejä.

Yleisellä tasolla MIPROv2-optimointi paransi lähes jokaisen kielimallin suorituskkyä molemmissa koeasetelmissa. Poikkeuksen tekevät Llama 3.3 70B, jonka suorituskky heikkeni zero-shot-asetelmassa, sekä Qwen 3 8B, jonka suorituskky heikkeni few-shot-asetelmassa. Erityisesti jälkimmäistä saattaa selittää se, että pienempien kielimallien suorituskky voi tietyissä tilanteissa kärsiä kehoitteiden pituuden kasvaessa (Liu, N. et al., 2023; Li, T. et al., 2024; Levy et al. 2024).

Taulukko 5.3: Eri kielimallien suorituskyvyt MIPROv2-optimoiduissa koeasetelmissa

Model	Accuracy	Precision	Recall	F1 Score	Macro F1
<i>MIPROv2 Optimized Zero-Shot w/ CoT</i>					
Llama 3.1 8B	74,00	70,97	94,02	80,88	70,13
Llama 3.3 70B	90,00	86,47	98,29	92,00	89,33
Qwen 3 8B	88,50	89,17	91,45	90,30	88,09
Qwen 3 32B	89,00	85,71	97,44	91,20	88,27
Deepseek-R1 8B	74,00	69,70	98,29	81,56	68,75
Deepseek-R1 70B	88,00	84,44	97,44	90,48	87,13
Poro 34B Chat	61,50	61,76	89,74	73,17	52,51
<i>MIPROv2 Optimized Few-Shot w/ CoT</i>					
Llama 3.1 8B	85,00	84,25	91,45	87,70	84,24
Llama 3.3 70B	94,00	94,12	95,73	94,92	93,80
Qwen 3 8B	81,00	86,92	79,49	83,04	80,72
Qwen 3 32B	94,00	94,87	94,87	94,87	93,82
Deepseek-R1 8B	81,00	85,59	81,20	83,33	80,62
Deepseek-R1 70B	92,00	89,15	98,29	93,50	91,55
Poro 34B Chat	59,50	59,09	100,00	74,29	39,50

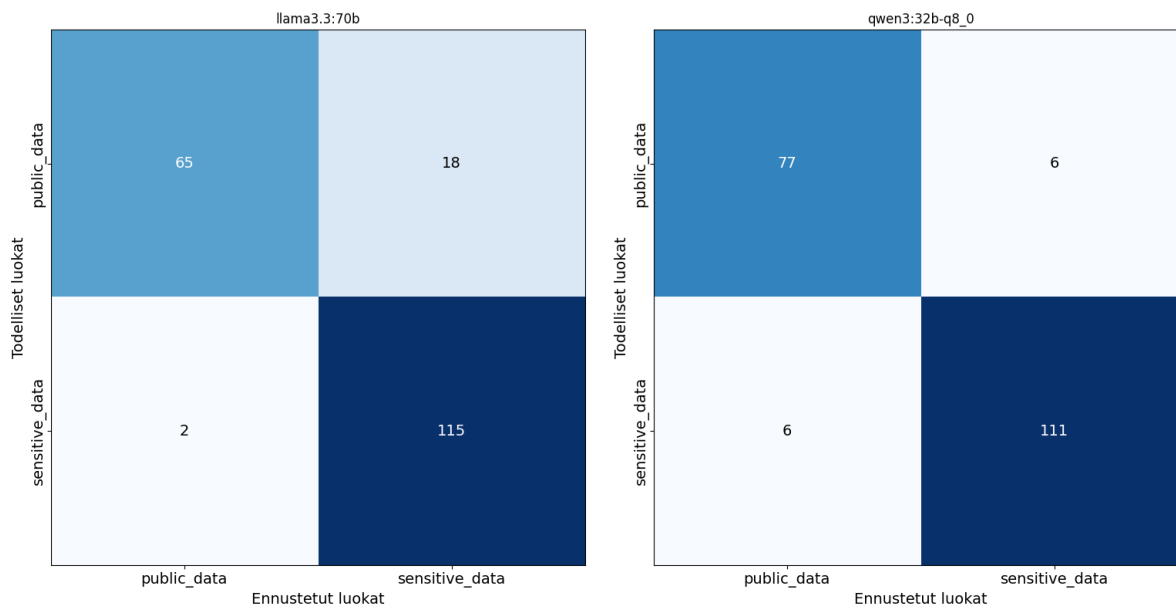
MIPROv2-optimoidussa zero-shot-koeasetelmassa parhaiten suoriutui Llama 3.3 70B. Se tunnisti kahta lukuun ottamatta jokaisen arkaluonteisen dokumentin. Seuraavaksi parhaiten suoriutuneet Qwen 3 32B ja Deepseek-R1 70B suoriutuivat vain marginaalisesti Llama 3.3 70B -mallia heikommin. MIPROv2-optimoidussa zero-shot-asetelmassa parhaiten pärjänneen mallin sekaannusmatriisi on nähtävissä kuvassa 5.4.

MIPROv2-optimoidussa few-shot-asetelmassa parhaiten suoriutui Qwen 3 32B. Yhteensä 117 arkaluonteiseksi annotoidusta dokumentista se tunnisti 111. Lisäksi se luokitteli vain kuusi arkaluonteista dokumenttia julkisiksi ja kuusi julkista dokumenttia arkaluonteiseksi.

Seuraavaksi parhaiten suoriutunut Llama 3.3 70B tunnisti kahta lukuun ottamassa kaikki arkaluonteiset dokumentit, mutta sen kokonaissuorituskykyä laskee 18 arkaluonteiseksi luokiteltua julkista dokumenttia. Huomion arvioista MIPROv2-optimoidussa few-shot-asetelmassa on se, että myös pienimmät mallit kuten Llama 3.1 8B Deepseek-R1 8B suoriutuivat varsin hyvin, jos kohta niiden suorituskyky jäi useamman prosenttiyksikön

suuremmista verrokeistaan. MIPROv2-optimoidussa few-shot-asetelmassa parhaiten pärjänneen mallin sekaannusmatriisi on nähtävissä alla kuvassa 5.4.

Kuva 5.4: Parhaiten suoriutuneiden MIPROv2-optimoitujen mallien sekaannusmatriisit. Vasemmalla Zero-Shot w/ CoT -asetelmassa parhaiten suoriutunut Llama 3.3 70B ja oikealla few-shot w/ CoT -asetelmassa parhaiten suoriutunut Qwen 3 32B.



5.2 Virheanalyysi

Kuten mainittu, tässä tutkielmassa paras tulos saavutettiin Qwen 3 32B -mallilla. Malli luokitteli 200 dokumentista yhteensä 188 kappaletta oikeisiin luokkiin ja 12 kappaletta väärin luokkiin. Väärin luokkiin luokitelluista dokumenteista kuusi kappaletta oli väärä negatiivisia ja kuusi väärä positiivisia. Alla on eritelty väärin luokiteltuja tapauksia ja niiden syitä.

Kuudesta väärästä negatiivisesta viisi kappaletta oli Jahvetin kirjelaatikon vastauskirjeitä. Kirjeet sisälsivät suoria tunnisteita kuten kokonaisia erisnimiä ja asuinpaikkoja. Näistä jokaisessa tapauksista Qwen 3 32B tunnisti erisnimet muusta tekstistä mutta luokitteli asiakirjan asiasisällön perusteella julkiseksi. Perusteluina oli, että kirjeet ovat joko viranomaistoiminnassa syntyneitä asiakirjoja tai niissä käsitellään ”yleisiä linjauksia”. Kuudes väärä negatiivinen puolestaan oli kuulijakirje, joka sisälsi kirjeen kirjoittaneen henkilön etunimen, sotilasarvon sekä komppanian ja joukko-osaston, joiden perusteella henkilöllisyys on varsin helposti selvitettävissä. Tästä huolimatta Qwen 3 32B luokitteli dokumentin julkiseksi ja perusteli päätöstään sillä, että kirjeeseen sisältynyt nimi on yleinen, eikä se mahdollista tunnistamista edes muihin tietoihin yhdistettynä.

Vääristä positiivista kaksi kappaletta sisälsi viranhaltijoiden nimiä. Ensimmäinen näistä oli seulonnapäätös, joka sisältää Kansallisarkiston pääjohtajan ja tutkimuspäällikön nimet. Toinen oli Jahvetin kirjelaatikon vastauskirje, jossa ohjeistetaan ottamaan yhteyttä kulkulaitosten ja yleisten töiden ministeriön tiedotussihteeriin. Tiedotussihteeri mainittiin nimeltä. Toiset kaksi kappaletta vääristä positiivisista sisälsi mainintoja sairauksista ja potilasasiakirjojen säilyttämisestä ilman henkilöihin yhdistettäviä tunnisteita. Yhdessä vääressä positiivisessa tapauksessa kielimalli luokitteli dokumentin arkaluonteiseksi kadunnimessä esiintyneen erisnimen perusteella, ja toisessa yksittäistapauksessa arkaluonteisen luokittelun perusteena oli nimettömässä kirjeessä esiintynyt mainita kirjoittajan pojasta ja hänen etunimestänsä ilman muita tunnistamista mahdollistavia tietoja.

5.3 Johtopäätökset

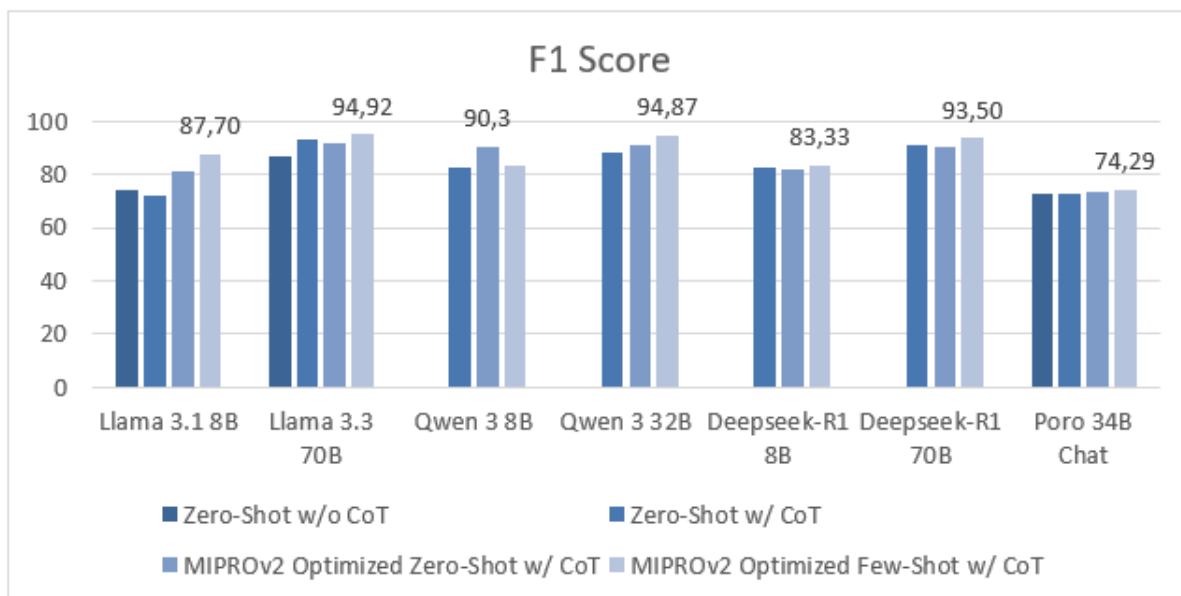
Tämän tutkielman ensimmäisessä tutkimuskysymyksessä kysyttiin, kuinka luotettavasti esikoulutetut generatiiviset kielimallit tunnistavat arkaluonteisia tietoja suomenkielisestä arkistoaineistosta. Tunnistamista kokeiltiin kuudella eri kielimallilla, joiden suoriutumista arvioitiin neljällä eri koeasetelmalla ja viidellä eri arviointimetriikalla.

Tulosten perusteella tässä tutkielmassa käytettyjen esikoulutettujen generatiivisten kielimallien suorituskkyky on lähtökohtaisesti korkealla. Parhaimmillaan ne kykenivät tunnistamaan arkaluonteisia dokumentteja pelkän tehtävänannon ja mahdollisten luokkien kuvausten perusteella erinomaisesti. Toisaalta tulokset myös osoittavat, että kielimallien suorituskkykyä on mahdollista parantaa entisestään ketjupäättelyllä, kehoteoptimoinnilla ja esimerkkidokumenteilla. Johtopäätöksenä on, että generatiiviset kielimallit voivat tunnistaa arkaluonteisia tietoja suomenkielisestä arkistoaineistosta tehokkaasti ja luotettavasti. Kuten luvussa 2.2.3 todettiin, generatiivisten kielimallien käyttö tekstiluokittelutehtävissä on paitsi mahdollista myös yksinkertaista, tehokasta ja helposti sovellettavissa monenlaisiin luokittelutehtäviin, kuten arkaluonteisten tietojen tunnistamiseen.

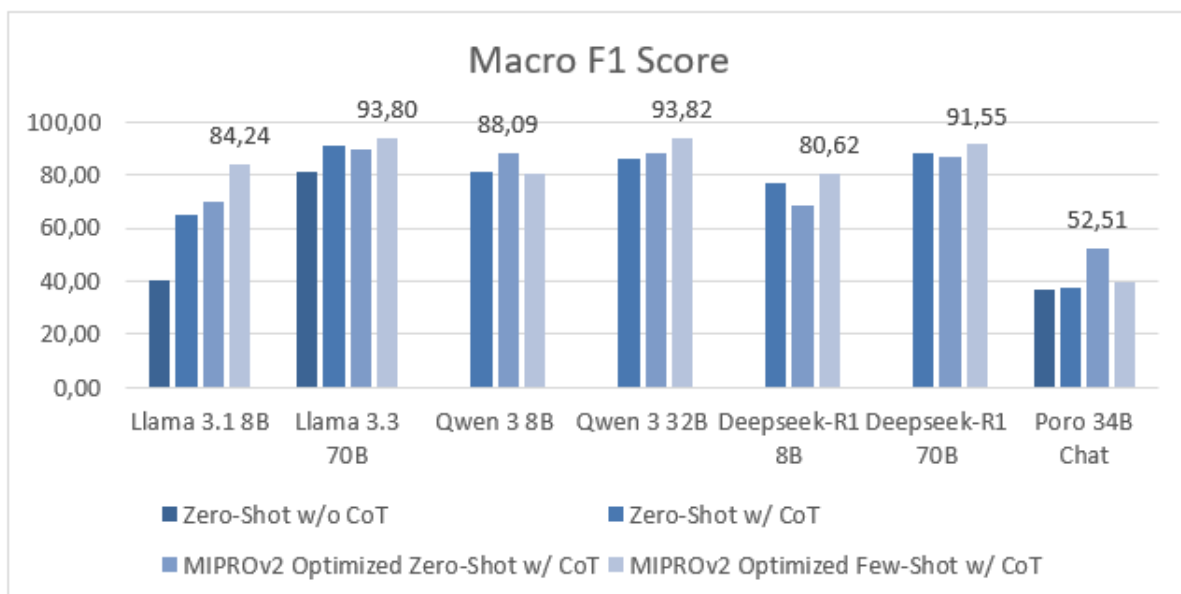
Toisessa tutkimuskysymyksessä puolestaan kysyttiin, onko eri kielimallien luotettavuudessa merkittäviä eroja ja mitkä kielelliset, kontekstuaaliset tai tekniset tekijät selittävät niitä. Tulosten perusteella kielimallien suorituskkyvyissä voi olla huomattaviakin eroja. Suurimpia suorituskkykyyn vaikuttavia tekijöitä ovat tämän tutkielman perusteella mallien sisäisten parametrien ja koulutusaineiston määrät. Mitä enemmän kielimallilla on parametreja ja mitä suuremmalla määrällä tokeneita se on koulutettu, sen todennäköisemmin se suoriutuu arkaluonteisten tietojen tunnistamisesta hyvin. Lisäksi kielimallille annetun kehotteen muoto

ja sisältö näyttävät vaikuttavan suorituskykyyn voimakkaasti. Kuvat 5.5 ja 5.6, joissa on eriteltyä kunkin kielimallin suoriutuminen kussakin koeasetelmassa F1-arvolla ja F1-keskiarvolla mitattuna, havainnollistavat parametrimäärien ja kehostrategioiden vaikutusta luokitteluun.

Kuva 5.5: Kaikkien kielimallien suorituskyky F1-arvolla mitattuna arkaluonteinen tieto -luokassa



Kuva 5.6: Kaikkien kielimallien suorituskyky F1-keskiarvolla mitattuna molemmissa luokissa



6 Lopuksi

Tämä tutkielma on syntynyt tarpeesta. Sen tavoitteena oli tuottaa uutta tietoa generatiivisten kielimallien käyttömahdollisuuksista suomenkielisten digitaalisten arkistoaineistojen luokittelussa, tukea käytännön arkistotyötä tehostavien työkalujen kehittämistä sekä tutkia yhdellä grafiikkasuorittimella ajettavien kielimallien kykyjä ymmärtää suomenkielisiä tekstejä ja niiden kontekstuaalisia piirteitä. Sen täsmällisenä tutkimustehtävänä oli selvittää, voiko generatiivisilla kielimalleilla tunnistaa henkilötietoja tai erityisen henkilötietoryhmän tietoja suomenkielisestä digitaalisesta arkistoaineistosta.

Tutkielmassa kokeiltiin yhteensä seitsemää eri kielimallia. Malleja ajettiin paikallisesti Kansallisarkiston laitteistolla, eikä niitä hienosäädetty tai jatkokoulutettu arkaluonteisten tietojen tunnistamiseen. Mallit olivat kooltaan kahdeksasta miljardista parametrasta seitsemäänkymmeneen miljardiin. Tutkielman aineistona toimi aidoista historiallisista dokumenteista koostettu aineistokokonaisuus, joka sisälsi todellisten henkilöiden henkilötietoja ja kertoi heidän kohtaloistaan talvi- ja jatkosodassa. Tunnistamisen onnistumista mitattiin osuvuudella, tarkkuudella, kattavuudella, F1-arvolla, F1-keskiarvolla ja sekaannusmatriiseilla.

Tutkielman keskeinen havainto on, että generatiivisilla kielimalleilla voidaan tunnistaa henkilötietoja tai erityisen henkilötietoryhmän tietoja suomenkielisestä digitaalisesta arkistoaineistosta. Tässä tutkielmassa käytetyt kielimallit suoriutuivat tehtävästä parhaimmillaan erittäin hyvin, ja niiden käyttö oli yksinkertaista ja tehokasta. Parhaat tulokset saavutettiin few-shot-asetelmassa käyttäen DSPy-optimoitua tehtävänantoa ja viittä esimerkkidokumenttia.

Tutkielmani keskeisin rajoite liittyy sen tulosten yleistyvyyteen. Kuten mainitsin analyysiluvussa, rajallisen laskentaresurssiajan takia kussakin koeasetelmassa kullakin kielimallilla voitiin ajaa vain kaksi luokitteluajoa. Näissä tulokset olivat identtisiä, jos kohta sitä selittänee suppeahko testiaineisto ja käytetty lämpötila-arvo. Yhtä kaikki tämän tutkielman tulokset ovat enemmän alustavia kuin lopullisia, ja suuremmalla testiaineistolla luokittelun deterministisyys olisi saattanut rakoilla ja eri luokitteluajojen välille syntyä eroja.

Tutkielmassa kehitetyllä luokittelumenetelmällä on yksi keskeinen tekninen rajoite. Tässä tutkielmassa käytettiin kielimalleja, jotka kykenevät käsittelemään vain tekstisyötteitä. Näin ollen kaikkien luokittelijalle annettujen dokumenttien tekstisisältö täytyi tunnistaa

sisällöntunnistumalleilla ennen varsinaista luokittelua. Tämä osaltaan lisäsi arkaluonteisten tietojen tunnistamisen kompleksisuutta ja siten kasvatti virheellisten ennustusten mahdollisuutta. Suunnitelmissa onkin kokeilla luokittelua niin sanotuilla multimodaalisilla kielimalleilla, jotka voivat prosessoida bittikarttamuotoisia aineistoja ilman erillistä sisällöntunnistamista.

Tutkielmani käytännöllinen hyöty kytkeytyy Kansallisarkiston digitaalisten tietoaaineistojen käytön edistämiseen. Edelleen kehitettynä ja tuotantokäyttöön sovellettuna se mahdollistane käyttörajoitettujen aineistokokonaisuuksien, sarjojen, alasarjojen ja arkistoyksiköiden osittaisen avaamisen. Tällä hetkellä käyttörajoitus kirjataan vähintään arkistoyksikkötasolle, vaikka arkistoyksikkö saattaa sisältää sekä julkisia että käyttörajoitettuja asiakirjoja. Syy tälle on se, että salattujen tai käyttörajoitettujen tietojen manuaalinen tunnistaminen yksittäisistä dokumenteista ei ole henkilöresurssien puolesta mahdollista.

Miellän tutkielmani merkityksen muodostuvan kolmella tavalla. Yhtäältä se on osin virkатыnä tehtyä teknistä kehitystyötä, joka lisää vaatimattomalla panoksellaan ymmärrystä suurten kielimallien kyvyistä sekä tekoälypohjaisten järjestelmien käyttömahdollisuuksista arkistotoimessa. Toisaalta se on kirjallinen opinnäyte, jonka tekemisen aikana saavutin omia tavoitteitani ja onnistuin vastaamaan asettamiini tutkimuskysymyksiin. Ennen kaikkea se on kuitenkin perustutkimusta, jolla on nähdäkseni tärkeä itseisarvo sen sovellusalasta tai käyttötarkoituksesta riippumatta.

Tutkielmani avaa vähintään neljä jatkotutkimusaihetta. Ensinnäkin julkisella sektorilla olisi nähdäkseni tarve suomenkieliselle, henkilötietojen ja erityisen henkilötietoryhmän tietojen suojaamiseen erikoistuneille kielimalleille. Mallien kouluttamiseksi voisi muodostaa koulutusaineiston, jonka sisältämät tiedot olisivat keinotekoisia tai synteettisiä ja joka olisi avoimesti jaettavissa. Toisaalta tässä tutkielmassa käytettyjä menetelmiä ja havaintoja olisi myös suhteellisen helppo yleistää julkishallinnon dokumenttien laajempaan sisällönanalyysiin ja metatiedottamiseen. Sen sijaan tieteellisessä kontekstissa olisi hedelmällistä tutkia kontekstioppimisen rajoja: kuinka suuri osuus konteksti-ikkunan maksimipituudesta voidaan käyttää ennen kuin kielimallien suorituskyky alkaa heikentyä suomenkielisen aineiston käsittelyssä. Samalla olisi hedelmällistä tutkia kielimallien itsetietoisuutta ja kehittää menetelmä, jolla voitaisiin mitata tai kvantifioida kielimallin itse ilmaisemaa luokittelun varmuutta tai epävarmuutta.

Lähteet

- Aangenendt, G. (2022): *Archives in the Digital Age: The use of AI and machine learning in the Swedish archival sector*. <https://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-478426>
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M, Gray, S, Chess, B, Clark, J, Berner, C, McCandlish, S., Radford, A., Sutskever, I., Amodei, D. (2020): Language models are few-shot learners. Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS 20). 1877–1901. *arXiv preprint arXiv:2005.14165*
- Burnham, M., Kahn, K., Wang, R. Y., & Peng, R. X. (2024). Political debate: Efficient zero-shot and few-shot classifiers for political text. *arXiv preprint arXiv:2409.02078*.
- DeepSeek-AI. (2025). DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv preprint arXiv:2501.12948*.
- Dubey, A. et al. (2024): The Llama 3 Herd of Models. <https://doi.org/10.48550/arXiv.2407.21783>
- Han, J., Kamber, M., Pei, J. (2012). Classification. <https://doi.org/10.1016/B978-0-12-381479-1.00008-3>
- Hutchinson, T. (2018). Protecting Privacy in the Archives: Supervised Machine Learning and Born-Digital Records. IEEE International Conference on Big Data (Big Data), 2696–2701. <https://doi.org/10.1109/BigData.2018.8621929>
- Jurafsky, D. & Martin, J. H. (2025). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd edition. <https://web.stanford.edu/~jurafsky/slp3>. [haettu 9.5.2025].
- Kojima, T., Gu, S., Reid, M., Matsuo, Y., Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35, 22199–22213.
- Laato, J., Kanerva, J., Loehr, J., Lummaa, V., & Ginter, F. (2025). Extracting Social Connections from Finnish Karelian Refugee Interviews Using LLMs. *arXiv preprint arXiv:2502.13566*.
- La Mela, M., Vats, E. (2023). Automatic classification of historical texts using a BERT model: News about wild berries, 1860-1910. Book of Abstracts, DH Benelux 2023,

May 31-June 2, Brussels, Belgium, 1–4.

<https://doi.org/zenodo.org/doi/10.5281/zenodo.7990441>

- Laurer, M., van Atteveldt, W., Casas, A., Welbers, K. (2023). Building Efficient Universal Classifiers with Natural Language Inference. *arXiv preprint arXiv:2312.17543*.
- Levy, M., Jacoby, A., & Goldberg, Y. (2024). Same task, more tokens: the impact of input length on the reasoning performance of large language models. *arXiv preprint arXiv:2402.14848*.
- Li, M., Lv, T., Cui, L., Lu, Y., Florêncio, D.A., Zhang, C., Li, Z., & Wei, F. (2021). TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models. AAAI Conference on Artificial Intelligence. *arXiv:2109.10282*
- Li, T., Zhang, G., Do, Q. D., Yue, X., & Chen, W. (2024). Long-context LLMs struggle with long in-context learning. *arXiv preprint arXiv:2404.02060*.
- Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P.S., & He, L. (2020). A Survey on Text Classification: From Traditional to Deep Learning. *arXiv preprint arXiv:2008.00364*.
- Litzen, V. (1994): Johdanto. Suomen historian asiakirjalähteet. Toim. Orrman, E., Pispala, E. Kansallisarkisto & WSOY.
- Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, 55(9), 1-35.
- Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2023). Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*.
- Luukkonen, R., Burdge, J., Zosa, E., Talman, A., Komulainen, V., Hatanpää, V., Sarlin, P., & Pyysalo, S. (2024). Poro 34B and the Blessing of Multilinguality. *arXiv preprint arXiv:2404.01856*.
- Lybeck, J., & Kansallisarkisto. (2006). *Arkistot yhteiskunnan toimiva muisti: Asiakirjahallinnon ja arkistotoimen oppikirja*. Arkistolaitos.
- Lönnblad, R. (2025). *EU:n yleisen tietosuoja-asetuksen vaikutukset arkistoinnin edellytyksiin*. Kansallisarkisto.
- Marciano, R., Underwood, W., Hanaee, M., Mullane, C., Singh, A., Tethong, Z. (2018). Automating the Detection of Personally Identifiable Information (PII) in Japanese-American WWII Incarceration Camp Records. IEEE International Conference on Big Data (Big Data), 2725–2732. <https://doi.org/10.1109/BigData.2018.8622634>

- Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., & Zettlemoyer, L. (2022). Rethinking the role of demonstrations: What makes in-context learning work?. *arXiv preprint arXiv:2202.12837*.
- Opsahl-Ong, K., Ryan, M. J., Purtell, J., Broman, D., Potts, C., Zaharia, M., Khattab, O. (2024). Optimizing instructions and demonstrations for multi-stage language model programs. *arXiv preprint arXiv:2406.11695*.
- Peeperkorn, M., Kouwenhoven, T., Brown, D., & Jordanous, A. (2024). Is temperature the creativity parameter of large language models? *arXiv preprint arXiv:2405.00492*.
- Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*.
- Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training. OpenAI. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf [haettu 11.4.2025].
- Rasimus, Tomi. (2019). *Julkisen hallinnon tietoaaineistojen säilyttäminen ja arkistointi*. Pro gradu -tutkielma, Itä-Suomen yliopisto. <http://urn.fi/urn:nbn:fi:uef-20200201>
- Renze, M., & Guven, E. (2024). The Effect of Sampling Temperature on Problem Solving in Large Language Models. *arXiv:2402.05201*.
- Repo, L., Skantsi, V., Rönqvist, S., Hellström, S., Oinonen, M., Salmela, A., Biber, D., Egbert, J., Pyysalo, S., & Laippala, V. (2021). Beyond the English Web: Zero-Shot Cross-Lingual and Lightweight Monolingual Classification of Registers. *Conference of the European Chapter of the Association for Computational Linguistics*.
- Schulhoff, S., Ilie, M., Balepur, N., Kahadze, K., Liu, A., Si, C., Li, Y., Gupta, A., Han, H., Schulhoff, S., Dulepet, P.S., Vidyadhara, S., Ki, D., Agrawal, S., Pham, C., Kroiz, G.C., Li, F., Tao, H., Srivastava, A., Costa, H.D., Gupta, S., Rogers, M.L., Goncarenco, I., Sarli, G., Galynker, I., Peskoff, D., Carpuat, M., White, J., Anadkat, S., Hoyle, A., & Resnik, P. (2024). The Prompt Report: A Systematic Survey of Prompt Engineering Techniques. *arXiv preprint arXiv:2406.06608*.
- Tarkka O., Koljonen J., Korhonen M., Laine J., Martiskainen K., Elo K., Laippala V. (2024). Automated Emotion Annotation of Finnish Parliamentary Speeches Using GPT-4. In *Proceedings of the IV Workshop on Creating, Analysing, and Increasing Accessibility of Parliamentary Corpora (ParlaCLARIN) @ LREC-COLING 2024*, pages 70–76, Torino, Italia. ELRA and ICCL.

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30. *arXiv:1706.03762*
- Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., Pyysalo, S. (2019). Multilingual is not enough: BERT for Finnish. *arXiv preprint arXiv:1912.07076*.
- Wang, Z., Pang, Y., & Lin, Y. (2023). Large Language Models Are Zero-Shot Text Classifiers. *arXiv preprint arXiv:2312.01044*.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E.H., Xia, F., Le, Q., & Zhou, D. (2022a). Chain of Thought Prompting Elicits Reasoning in Large Language Models. *Advances in neural information processing systems*, 35, 24824–24837.
- Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E.H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., & Fedus, W. (2022b). Emergent Abilities of Large Language Models. *arXiv preprint arXiv:2206.07682*.
- Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu, D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin, H., Tang, J., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Zhou, J., Lin, J., Dang, K., Bao, K., Yang, K., Yu, L., Deng, L., Li, M., Xue, M., Li, M., Zhang, P., Wang, P., Zhu, Q., Men, R., Gao, R., Liu, S., Luo, S., Li, T., Tang, T., Yin, W., Ren, X., Wang, X., Zhang, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Zhang, Y., Wan, Y., Liu, Y., Wang, Z., Cui, Z., Zhang, Z., Zhou, Z., & Qiu, Z. (2025). Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Yang, J., Zhang, X., Liang, K., Liu, Y. (2023). Exploring the Application of Large Language Models in Detecting and Protecting Personally Identifiable Information in Archival Data: A Comprehensive Study. *IEEE International Conference on Big Data (BigData)*, 2116-2123, <https://doi.org/10.1109/BigData59044.2023.10386949>
- Zhang, S., Hou, J., Peng, S., Li, Z., Hu, Q., Wang, P. (2023). ArcGPT: A Large Language Model Tailored for Real-world Archival Applications. *arXiv preprint arXiv:2307.14852*.

Liitteet

Liite 1 Esimerkki kehoitteesta ja vastauksesta

[2025-05-21T17:30:39.848812]

System message:

Your input fields are:

1. `text` (str): The document to classify.
2. `categories` (str): Description of classification categories.

Your output fields are:

1. `reasoning` (str): Step-by-step reasoning about whether the document fits 'public_data' or 'sensitive_data'.
2. `label` (Literal['public_data', 'sensitive_data']): Final classification label.

All interactions will be structured in the following way, with the appropriate values filled in.

```
[[ ## text ## ]]
{text}
```

```
[[ ## categories ## ]]
{categories}
```

```
[[ ## reasoning ## ]]
{reasoning}
```

```
[[ ## label ## ]]
{label}      # note: the value you produce must exactly match (no extra characters) one
of: public_data; sensitive_data
```

```
[[ ## completed ## ]]
```

In adhering to this structure, your objective is:

Classify the provided Finnish historical document as either 'public_data' or 'sensitive_data' based on GDPR guidelines and administrative archival context. Justify your decision by analyzing identifiers (e.g., names, addresses, personal circumstances) and whether the content involves personal data requiring protection.

User message:

This is an example of the task, though some input or output fields are not supplied.

[[## text ##]]

(Esimerkki Jahvetin kirjelaatikko -arkiston dokumentista. Poistettu jälkikäteen.)

[[## categories ##]]

****sensitive_data****

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile
- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")
- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data

- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Respond with the corresponding output fields, starting with the field `[[## reasoning ##]]`, then `[[## label ##]]` (must be formatted as a valid Python Literal['public_data', 'sensitive_data']), and then ending with the marker for `[[## completed ##]]`.

Assistant message:

[[## reasoning ##]]

Not supplied for this particular example.

[[## label ##]]

sensitive_data

[[## completed ##]]

User message:

This is an example of the task, though some input or output fields are not supplied.

[[## text ##]]

(Esimerkki Jahvetin kirjelaatikko -arkiston dokumentista. Poistettu jälkikäteen.)

[[## categories ##]]

****sensitive_data****

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile

- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")
- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data
- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Respond with the corresponding output fields, starting with the field `[[## reasoning ##]]`, then `[[## label ##]]` (must be formatted as a valid Python Literal['public_data', 'sensitive_data']), and then ending with the marker for `[[## completed ##]]`.

Assistant message:

[[## reasoning ##]]

Not supplied for this particular example.

[[## label ##]]

sensitive_data

[[## completed ##]]

User message:

This is an example of the task, though some input or output fields are not supplied.

[[## text ##]]

(Esimerkki Jahvetin kirjelaatikko -arkiston dokumentista. Poistettu jälkikäteen.)

[[## categories ##]]

****sensitive_data****

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile
- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")
- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data
- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Respond with the corresponding output fields, starting with the field `[[## reasoning ##]]`, then `[[## label ##]]` (must be formatted as a valid Python Literal['public_data', 'sensitive_data']), and then ending with the marker for `[[## completed ##]]`.

Assistant message:

```
[ [ ## reasoning ## ] ]
```

Not supplied for this particular example.

```
[ [ ## label ## ] ]
```

```
sensitive_data
```

```
[ [ ## completed ## ] ]
```

User message:

This is an example of the task, though some input or output fields are not supplied.

```
[ [ ## text ## ] ]
```

(Esimerkki Jahvetin kirjelaatikko -arkiston dokumentista. Poistettu jälkikäteen.)

```
[ [ ## categories ## ] ]
```

```
**sensitive_data**
```

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be

protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile
- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")
- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data
- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Respond with the corresponding output fields, starting with the field `[[## reasoning ##]]`, then `[[## label ##]]` (must be formatted as a valid Python Literal['public_data', 'sensitive_data']), and then ending with the marker for `[[## completed ##]]`.

Assistant message:

[[## reasoning ##]]

Not supplied for this particular example.

[[## label ##]]

sensitive_data

[[## completed ##]]

User message:

[[## text ##]]

(Esimerkki Jahvetin kirjelaatikko -arkiston dokumentista. Poistettu jälkikäteen.)

[[## categories ##]]

****sensitive_data****

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile
- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")
- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data
- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Respond with the corresponding output fields, starting with the field `[[## reasoning ##]]`, then `[[## label ##]]` (must be formatted as a valid Python Literal['public_data', 'sensitive_data']), and then ending with the marker for `[[## completed ##]]`.

Assistant message:

[[## reasoning ##]]

The document contains personal details such as the individual's birth year (1925), military service dates (1943), and hometown (Viitasaari, Sisä-Suomen). However, there is no explicit name, address, ID number, or other direct identifiers. The text focuses on a military service request and general location references. While birth year and location could theoretically contribute to indirect identification, the absence of a name and the generic nature of the information (e.g., "born in 1925," "serving in the army") reduce the risk of re-identification. Military service records are often considered public data in Finland, and the content aligns with public_data examples (demographic/service data without direct identifiers). The request for reassignment and technical details about communications infrastructure do not contain sensitive personal information.

[[## label ##]]

public_data

[[## completed ##]]

User message:

[[## text ##]]

(Esimerkki Jahvetin kirjelaatikko -arkiston dokumentista. Poistettu jälkikäteen.)

[[## categories ##]]

****sensitive_data****

Contains personal data or special category data. Based on this information, a natural person or 'data subject' can be identified directly or indirectly. This data must be protected in accordance with the General Data Protection Regulation (GDPR).

Examples of sensitive personal data:

- Name
- Address
- ID card or passport number
- Income information
- IP address
- Cultural profile
- Race or ethnic origin
- Sexual orientation or behavior
- Political opinions
- Religious or philosophical beliefs
- Trade union membership
- Genetic data
- Biometric data
- Health data
- Data relating to criminal convictions and offenses
- Other data considered sensitive

****public_data****

Does not contain personal data or special category data. The information does not enable the identification of a natural person either directly or indirectly. This data does not need to be protected under the General Data Protection Regulation (GDPR).

Examples of public data:

- Names of officials who have made public authority decisions in public documents
- Anonymized or pseudonymized names (e.g., "Person T", "Mr. X")

- General demographic data without identifying information
- Contact information of companies or entrepreneurs
- Public registers and openly available information
- General preferences or interests without personal data
- Reviews, opinions, and feedback without personal data
- General statistics and research results
- Company name or registration number
- General email addresses (e.g., info@company.com)
- Other anonymized data

Respond with the corresponding output fields, starting with the field `[[## reasoning ##]]`, then `[[## label ##]]` (must be formatted as a valid Python Literal['public_data', 'sensitive_data']), and then ending with the marker for `[[## completed ##]]`.

Response:

[[## reasoning ##]]

The document contains explicit personal data: the sender's full name (XXXX) and address (XXXX). It also references a family member (a daughter) and a prison, which could imply sensitive information about criminal convictions or incarceration. While the daughter is not directly named, the context of a prison inquiry and the sender's identifiable details create a risk of indirect identification (e.g., combining the sender's identity with public records). Under GDPR, data relating to criminal convictions (even indirectly) falls under "special categories" of sensitive data. The presence of direct identifiers (name and address) and the sensitive context (prison inquiry) justify classification as ****sensitive_data****.

[[## label ##]]

sensitive_data

[[## completed ##]]