



Kandidatutkielma

Tietojenkäsittelytieteen kandiohjelma

# **Tekstipohjaiset vastakkaishyökkäykset NLP-malleja vastaan**

Akira Taguchi

3.8.2022

MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA  
HELSINGIN YLIOPISTO

## Ohjaaja(t)

Prof. Nikolaj Tatti

## Yhteystiedot

PL 68 (Pietari Kalmin katu 5)  
00014 Helsingin yliopisto

Sähköpostiosoite: [info@cs.helsinki.fi](mailto:info@cs.helsinki.fi)

URL: <http://www.cs.helsinki.fi/>

Tiedekunta — Fakultet — Faculty		Koulutusohjelma — Utbildningsprogram — Study programme	
Matemaattis-luonnontieteellinen tiedekunta		Tietojenkäsittelytieteen kandiohjelma	
Tekijä — Författare — Author			
Akira Taguchi			
Työn nimi — Arbetets titel — Title			
Tekstipohjaiset vastakkaishyökkäykset NLP-malleja vastaan			
Ohjaajat — Handledare — Supervisors			
Prof. Nikolaj Tatti			
Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages	
Kandidutkielma	3.8.2022	15 sivua	
Tiivistelmä — Referat — Abstract			
<p>Tässä tutkimuksessa käsitellään tekstipohjaisia vastakkaishyökkäyksiä NLP-malleja vastaan. Tutkielmassa tutustutaan hyökkäystaksonomiaan sekä puolustusmetodeihin yleisimmillä osa-alueilla. Lisäksi tarkoituksena on tarkastella vastakkaishyökkäysten tulevaisuutta teknologian ja yhteiskunnallisten rakenteiden kehittyessä, ja pohtia mahdollisia tulevaisuuden skenaarioita.</p> <p>Tutkielmassa tulee myös tunnistamaan vastakkaishyökkäysten roolin nykYTEknologiassa, sekä syyt merkittävyyteen koneoppimismallien käytössä. Huomaamaan myös hyökkäyspinta-alan kattavan laajemman alueen puolustuspinta-alaan verrattuna vastakkaishyökkäyksissä. Työssä tullaan myös huomaamaan hyökkäyspinta-alan kattavan laajemman alueen puolustuspinta-alaan verrattuna vastakkaishyökkäyksissä. Lopuksi todetaan syyn olevan jo-valjastettu, mutta reunatapauksissa hallitsemaan tekoälyn voima.</p>			
<p><b>ACM Computing Classification System (CCS)</b>  Security and privacy → Human and societal aspects of security and privacy  Computing methodologies → Artificial Intelligence → Natural language processing</p>			
Avainsanat — Nyckelord — Keywords			
Luonnollisen kielen käsittely, vastakkaishyökkäys, koneoppiminen, tekoäly, ladonta, sensuuri			
Säilytyspaikka — Förvaringsställe — Where deposited			
Helsingin yliopiston kirjasto			
Muita tietoja — övriga uppgifter — Additional information			



# Sisältö

<b>1</b>	<b>Johdanto</b>	<b>1</b>
<b>2</b>	<b>NLP-luokittimien käyttö</b>	<b>3</b>
<b>3</b>	<b>Hyökkäystyypit</b>	<b>4</b>
3.1	Roskapostisuodatuksen ohitus . . . . .	4
3.2	Neuroverkkohyökkäykset . . . . .	5
3.3	Sensuurin ohitus . . . . .	5
3.4	Näkymättömät merkit . . . . .	6
3.5	Homoglyfit . . . . .	7
3.6	Uudelleenjärjestelyt . . . . .	7
3.7	Poistatukset . . . . .	8
<b>4</b>	<b>Hyökkäyksiltä suojautuminen</b>	<b>10</b>
4.1	OCR-puolustus . . . . .	10
4.2	Suorituskykykeskeinen puolustus . . . . .	10
<b>5</b>	<b>Yhteenveto</b>	<b>12</b>
	<b>Lähteet</b>	<b>14</b>



# 1 Johdanto

Koneoppimisen käyttötarkoitusten määrä kasvaa vuosi vuodelta suuremmaksi. Tätä teknologiaa voidaan hyödyntää muun muassa ihmisten puhuman kielen käsittely. Luonnollisen kielen käsittely (eng. Natural Language Processing, NLP) on alati kasvavassa kuluttajakäytössä johtuen seuraavista syistä:

- laskentatehon kasvu
- suurien tietomäärien saatavuus
- onnistuneiden koneoppimismenetelmien kehitys
- sekä laajempi ihmiskielen ymmärrys ja sen käyttö eri konteksteissa (Hirschberg ja Manning, 2015).

Luonnollisen kielen käsittelyä voidaan hyödyntää kohdennetussa mainonnassa. Analysoimalla NLP-mallin avulla esimerkiksi käyttäjien lähettämiä viestejä toisilleen, voidaan saada selville tuote, jota kannattaa mainostaa mainoskohdennetulle yksilölle. Viesti ystävälle mainoskohdennetussa viestipalvelussa antaa työstettävän datan NLP-mallille: “Mikä elokuva meidän pitäisi katsoa viikonloppuna?” NLP-mallin avulla automaattinen mainostaja ymmärtää mainostaa kyseiselle käyttäjälle miltei välittömästi sarjalippuja mainostavasta elokuvateatterista, suoratoistopalvelua tai mainostavaa aktiviteettikeskusta kyseiselle viikonloppu. Tämän rahanarvoisen tarpeen löytäminen datasta automaation avulla edellyttää kaikkia neljää aikaisemmin mainittua teknologista edistystä kultakin osa-alueelta.

Kaikkien neljän osa-alueen kehittyminen mahdollistaa luonnollisen kielen käsittelyn yleistymisen. Ihmiskielen ymmärtäminen tietokoneen tasolla on kehittynyt huomattavasti, kun ihmisen käyttämää kieltä, virkkeitä ja sanoja on alettu pilkkomaan helpommin ymmärrettäviksi paloiksi (Chowdhury, 2003). Jotta luonnollisen kielen käsittelyn malli olisi rakennettu älykkäästi, tarvitsemme edistyneitä koneoppimismetodeita. Tämä on tullut kehityksen saatossa mahdolliseksi (Jordan ja Mitchell, 2015). Koska datan määrä on kasvanut ja dataa on helpompaa hankkia (Gopalakrishnan, 2018), pystymme kouluttamaan mallin toimimaan mahdollisimman monessa eri tilanteessa. Laskentatehon huomattava kasvu vuosien mittaan (Moore et al., 1965) on alkanut mahdollistaa suurempien datamäärän käsittelyä kuin aikaisemmin.

Tässä tutkielmassa tarkastellaan NLP-hyökkäysten käyttökohteita. Tähän kuuluu hyökkäystaksonomia, puolustusmenetelmät sekä NLP-mallien sekä niihin kohdistuvien hyökkäysten tulevaisuus. Hyökkäystaksonomiassa käymme läpi erilaisia tapoja hyökätä NLP-malleja vastaan, hyökkäysten tarkoituksiin ja onnistumisen todennäköisyyksiin. Puolustusmenetelmät ovat tärkeässä osassa, jotta haavoittuvuuteen kohdistuvat yritykset saavat ohjeita vahingon mitigointiin ja ennaltaehkäisyyn. On tärkeää myös spekuloida mahdollisia kehityksiä koneoppimisessa sekä tästä syntyviä haavoittuvuuksia. Lopuksi käymme läpi mahdollisia luonnollisen kielen käyttökohteita tulevaisuudessa sekä näistä aiheutuvia seurauksia eri osa-alueisiin sekä akateemisella että kaupallisella puolella.



## 2 NLP-luokittimien käyttö

NLP-luokittimia, eli NLP-malleja käytetään tapauksiin, joissa on tehokkaampaa korvata ihmisen manuaalinen tekemä tarkastustyö. Näihin tapauksiin kuuluvat muun muassa roskapostin tunnistus sekä vihapuheen tunnistus sosiaalisesta mediasta. Esimerkiksi Twitterissä käytetään NLP-malleja tunnistamaan sopimatonta sisältä twiiteistä (Kandasamy ja Koroth, 2014).

Tämä kaikki tarkastustyö voitaisiin tehdä manuaalisesti käsin, mutta tarkastettavan sisällön määrän vuoksi tämä ei ole käytännössä mahdollista. Tietoteknistaitoinen ihminen pystyisi tarkastamaan vastaanotetusta sähköpostista, mikäli kyseinen sähköposti olisi esimerkiksi kalasteluroskapostia. Koska roskapostia lähetetään automaattisesti jokaiseen olemassa olevaan sähköpostiosoitteeseen päivittäin, menisi roskapostien tunnistamiseen ihmiseltä liian kauan aikaa päivittäin. Tämän takia useimmissa sähköpostiohjelmissa tulee mukana automaattisesti roskapostia suodattava NLP-malli, joka päästää läpi vain sähköpostit, joista NLP-malli ei ole varma, onko se roskapostia. Perehdytään seuraavassa kappaleessa tarkemmin tämän suodattimen ohitukseen. Tämä haavoittuvaisuus on läsnä myös muissa sovelluksissa, joissa käytetään NLP-mallia.

# 3 Hyökkäystyypit

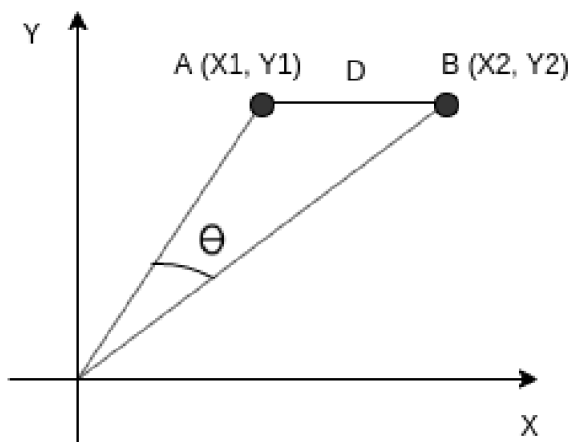
Tässä kappaleessa käydään läpi hyökkästaksonomia, eli hyökkäysrajapinta, NLP-malleja vastaan. Ensin käydään läpi roskapostisuodatuksen roskapostisuodatuksen ohitus, joka on NLP-hyökkäysten keskiössä. Sitten esittelen neuroverkkohyökkäykset, sensuuriohituksen sekä ladontahyökkäykset.

## 3.1 Roskapostisuodatuksen ohitus

Vastakkaishyökkäyksiä voidaan käyttää sähköposteissa roskapostisuodattimien ohitukseen. Roskapostisuodattimet toimivat koulutettujen NLP-mallien mukaan. Nämä mallit merkkavat vastaanotetut sähköpostit joko hyvántahtoisiksi tai pahantahtoisiksi, eli roskaposteiksi. Kuchipudi et al. (2020)

Suodattimia vastaan toimii kolme vastakkaishyökkäystä: (1) Synonyymien korvaus, (2) kelposanan injektointi sekä (3) roskapostisanojen väljennys. Sana ”kelpo” tarkoittaa tässä yhteydessä tekstiä, jonka roskapostisuodatin on merkinnyt hyvántahtoiseksi. Synonyymien korvauksessa tarkoitus on korvata pahantahtoiset sanat hyvántahtoisiksi luokitelluilla synonyymeilla. Sanojen synonyymisyys lasketaan taulukon 2.1 tapauksessa kuvan 2.1 kosini-samankaltaisuudella (taulukko 2.1). Kelposanan injektoinnissa kelposanoja lisätään sähköpostiin niin paljon, kunnes NLP-malli tunnistaa roskapostin olevan kelpopostia. Kelposanoja voidaan injektoida tietokannoista roskaposteihin muuttamatta viestin tarkoitusta rajusti. Roskapostisanojen väljennyksessä roskapostisanoihin sisällytetään välilyöntejä, jotta NLP-malli ei tunnistaisi näitä sanoja roskasanoiksi. Kun väljennystä on harjoitettu tarpeeksi, muuttuu roskaposti NLP-mallin näkökulmasta kelpopostiksi. (Kuchipudi et al., 2020)

Kelposanan injektoinnille ja roskasanojen väljennykselle on olemassa erilaisia implementaatioita. Seuraavissa alaluvuissa tutustutaan ladontapohjaisiin vastakkaishyökkäyksiin. Muun muassa näitä hyökkäysmetodeita voidaan käyttää kahdessa aiemmin mainitussa roskapostisuodattimeen kohdistetussa hyökkäyksessä. Implementaatioita yhdistelemällä ja vaihtelemalla, saattaa NLP-mallin pahantahtoisuuden havaitseminen heikentyä entistään, taaten hyökkääjälle varmemman onnistumisen.



**Kuva 3.1:** Sähköpostien samankaltaisuus voidaan laskea käyttäen kosini-samankaltaisuutta kahden sähköpostivektorin välillä. (Kuchipudi et al., 2020)

## 3.2 Neuroverkkohyökkäykset

Sanatason vastakkaishyökkäykset syvää oppimisverkkoa vastaan paljastavat näiden verkkojen heikkouksia. Vastakkaishyökkääminen näitä verkkoja vastaan on vaikeaa verrattuna kuvatason hyökkäyksiin. Tämä johtuu lauseiden diskreetistä luonteesta. Pienikin sanatai merkkimuutos vaikuttaa lauseen viestiin sekä todennäköisemmin viestin luokitukseen, jonka NLP-malli päättää. (Zang et al., 2020)

## 3.3 Sensuurin ohitus

Koska sensuuria voidaan soveltaa hyödyntäen koneoppimismalleja, voidaan sensuuri myös ohittaa hyödyntäen koneoppimismallin heikkouksia. Vastakkaishyökkäys voisi tunnistaa sensurointia aiheuttavia pikseliyhdistelmiä, ja tässä tutkimuksessa esiteltyjä hyökkäystapoja käyttäen sensuurin laukaiseminen voidaan estää. Tällöin kyseessä ei kuitenkaan enää ole puhdas merkintä (eng. clean label), sillä vastakkaishyökkäyksen todellinen tarkoitus näkyy käyttäjälle silmintarkasteltavana (Gan et al., 2021). Puhtaan merkinnän uupues- sa esimerkiksi tekstipohjaisessa vastakkaishyökkäyksessä mahdollisista myös helpomman puolustautumisen (Pruthi et al., 2019).

Muokattu viesti	Kosini-samankaltaisuus	Ennustus
Ringtone Club: Get the UK singles chart on your mobile each week and choose any top quality ringtone! This message is free of charge.	1	roskapostia
Ringtone Club: <b>acquire</b> the UK single <b>graph</b> on your <b>Mobile_River</b> each <b>hebdomad</b> and <b>take</b> any <b>top_side caliber</b> ringtone! This <b>content</b> is <b>free_people</b> of charge.	0,583	roskapostia
Ringtone Club: <b>become</b> the UK <b>bingle graph</b> on your <b>nomadic</b> each <b>workweek</b> and <b>select</b> any <b>upper_side caliber</b> ringtone! This <b>subject_matter</b> is <b>liberate</b> of charge.	0,583	roskapostia
Ringtone Club: <b>go</b> the UK <b>one graph</b> on your <b>peregrine</b> each <b>calendar_week</b> and <b>pick_out</b> any <b>upside character</b> ringtone! This <b>substance</b> is <b>release</b> of charge.	0,583	kelpopostia

**Taulukko 3.1:** Synonyymien korvaus. Vanhan viestin korvatut osat on lihavoitu. (Kuchipudi et al., 2020)

### 3.4 Näkymättömät merkit

Näkymättömät merkit vaikuttavat tietokoneen NLP-mallin ymmärtämään sisältöön. Kyseinen hyökkäys perustuu Unicode-merkistöstandardiin, joka sisältää yksilöivät koodiarvot kirjoitushetkellä yli 100 000 kirjoitusmerkille, tähän kuuluvat myös aakkoset sekä erikoismerkit.

Esimerkki tällaisesta erikoismerkistä on nollatilavuuden välilyönti -merkki, jonka Unicode merkintä on U+200B. Tällä merkillä voimme esimerkiksi vaikuttaa pelichattiin lähetettävän myrkyllissuodatettavaan merkkijonoon "olet huono"niin, että merkkijono menisi NLP-mallin läpi chätistä. Merkkijono olU+200Bet huU+200Bono saattaisi mennä läpi chatin suodattimesta, mutta vastapuolelle viesti olisi edelleen olet huono.

Kontekstin poistamisen lisäksi näkymättömillä merkeillä voidaan myös tuoda ja syrjäyttää konteksteja toisilla. Esimerkiksi:

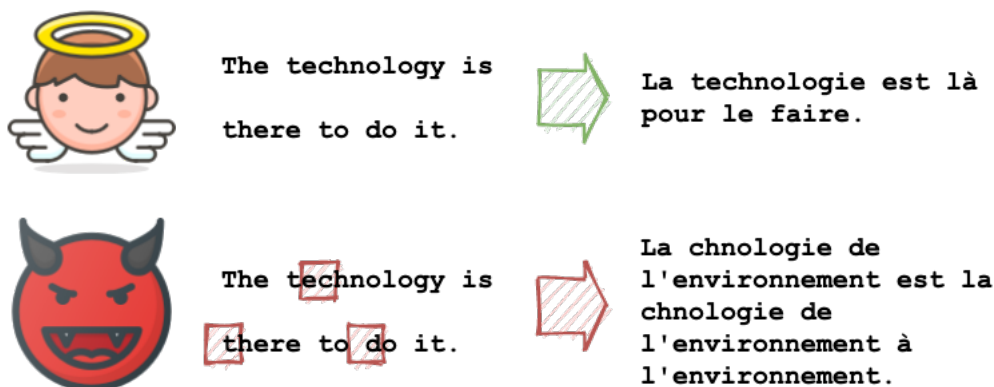
Mikä pyhäinhäväistyksen rakennus!

Miten onnistuit tekemään tämän näin laiskasti? -tekstin negatiivisuus voidaan syrjäyttää positiivisuudella syöttämällä NLP-mallille sen sijaan teksti:

Mikä pyU+200BhäinhävU+200BäistyU+200BksenU+200B rakennus!

Miten onnistuit tekemään tämän U+200BnäU+200Bin laU+200BiskasU+200Bti?.

Kuva 3.1 on esimerkki kontekstin syrjäyttämisestä näkymättömillä merkeillä. Esimerkissä tapahtuu käännös englannin kielestä ranskan kieleen.



Kuva 3.2: Hyökkäys näkymättömillä merkeillä (Boucher et al., 2021)

## 3.5 Homoglyfit

Homoglyfihyökkäykset NLP-malleja vastaan pohjautuvat siihen, että pahantahtoisten merkkien viralliset esitysmuodot näyttäytyvät hyväntahtoisilta merkkien virallisilta esityksiltä. Joissain kielissä tekstin merkitys muuttuu täysin yhden merkin vaihtuessa. Esimerkkinä homoglyfistä on  $A \rightarrow A$ , missä viimeinen kirjain on todellisuudessa kyrillinen kirjain А. Kuvassa 3.3 homoglyfihyökkäys on muuntanut englanninkielisen tekstin

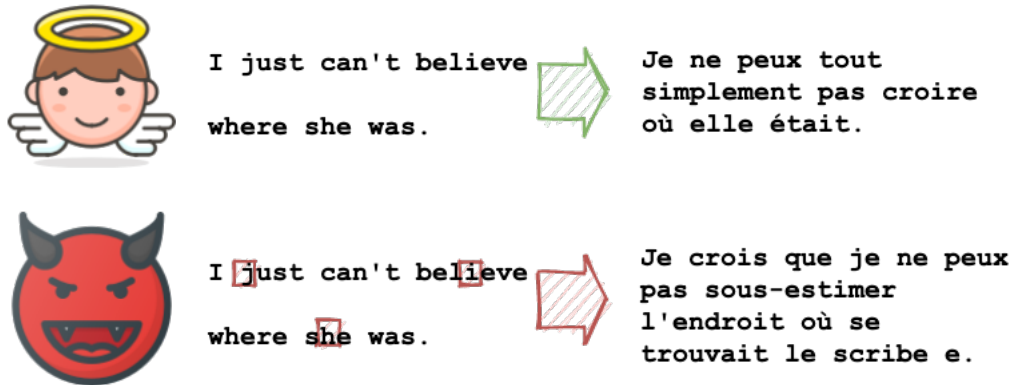
I just can't believe where she was ranskankieliseen käännökseen

I guess I can't underestimate the location of the scribe and.

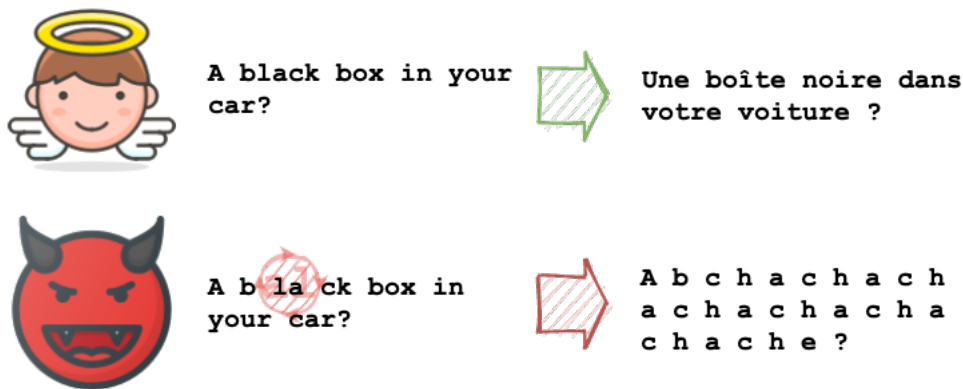
Näkymättömien merkkien lailla homoglyfihyökkäyksen toteutus riippuu ympäristön fontista. (Boucher et al., 2021)

## 3.6 Uudelleenjärjestelyt

Uudelleenjärjestelyhyökkäys pohjautuu näennäisen tekstin uudelleenjärjestämiseen pahantahtoisesti. Pankkitilinumeron 1234567 pystyy esimerkiksi vaihtamaan kaksisuuntaisella algoritmilla tilinumeroksi 7654321 pankin palvelinpuolella maksajan huomaamatta mitään. Unicode-merkintä tälle suunnanvaihdolle on U+200F. Uudelleenjärjestelyjä käyte-



Kuva 3.3: Homoglyfihyökkäys (Boucher et al., 2021)

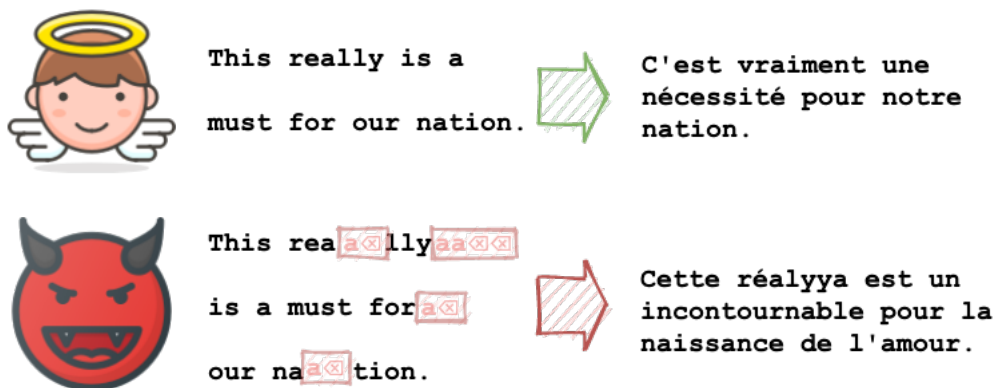


Kuva 3.4: Homoglyfihyökkäys (Boucher et al., 2021)

tään myös NLP-mallin sekoittamiseen, jolloin tulokset NLP-mallista ovat käyttökeltomia. Kuvassa 3.4 uudelleenjärjestelyhyökkäys merkeissä `la` aiheuttaa ranskankielisen käännöksen järjettömyyden. Tämänlaista hyökkäystä voisi käyttää digitaalista sanakirjaa tai kääntäjää vastaan. (Boucher et al., 2021) U+200F ladotaan näkymättömänä näkymättömien merkkien tapaan.

### 3.7 Poistatukset

Viimeisenä käydään läpi poistatushyökkäykset. Poistatushyökkäyksiä on vaikeampi toteuttaa aikaisempiin metodeihin verrattuna. Tämä johtuu useimpien käyttöjärjestelmien estosta kopioida poistatusta sisältävää tekstiä leikepöydälle suoraviivaisilla tavoilla, joilla uhri sen tekisi. Onnistuakseen poistatushyökkäyksessä, hyökkääjän tarvitsee yleisesti injektoida NLP-malliin poistatus itse. Esimerkkejä poistatusmerkeistä ovat askelpalautin (BS, eng. *backspace*), delete (DEL) sekä vaununpalautus (CR, eng. *carriage return*).



Kuva 3.5: Poistatushyökkäys (Boucher et al., 2021)

(Boucher et al., 2021) Kuva 3.5 havainnollistaa poistatushyökkäystä käytännössä.

# 4 Hyökkäyksiltä suojautuminen

Tässä kappaleessa käydään läpi erilaisia puolustusmenetelmiä NLP-hyökkäyksiä vastaan. Ensiksi esitellään OCR-puolustus, sitten kuvataan suorituskeskeinen puolustautuminen.

## 4.1 OCR-puolustus

NLP-hyökkäykset voidaan torjua korkean tason abstraktiolla korkealla yleisrasituksella sekä alemman tason abstraktiolla alemmalla yleisrasituksella. Näytöltäluvun (eng. OCR, On-Screen-Reading) avulla epäselvytydet tekstin aidosta luonteesta voidaan hahmontaa uudelleen tulkitsemalla aineisto uudestaan visuaalisesti. Tämä menetelmä lisää yleisrasi- tusta huomattavasti riippuen käyttötarkoituksesta, mutta poistaa pahantahtoiset merkit ilman NLP-mallin uudelleenkoulutusta. (Boucher et al., 2021)

## 4.2 Suorituskykykeskeinen puolustus

Keskitymme seuraavaksi näkymättömiin merkkeihin, -homoglyfeihin, uudelleenjärjestelyi- hin -ja poistatuksiin perustuvien hyökkäyksien puolustamiseen. Suorituskykykeskeiset puo- lustusmenetelmät ovat kuitenkin laskennallisesti kalliita, eivätkä koneoppimismallin ul- koistaneet yritykset yleensä pysty kustantamaan kyseisiä metodeita (Huang et al., 2019).

Näkymättöimen merkein tapauksessa, tietyt näkymättömät merkit voidaan poistaa suo- raan syötteestä. Mikäli sovelluksessa näitä merkkejä ei voida poistaa, voidaan ne korvata *ei-<unk>* upotuksilla. Korvaus tapahtuu lähdekielisanakirjassa, jonne kuvataan tuntema- ton merkki ”ei-tuntemattomaksi tokeniksi”. Näin tuntemattomat merkit eivät voi häiritä ladontaa merkeillä, joista ladontamoottori ei ole aivan varma. (Boucher et al., 2021)

Homoglyfihyökkäysten torjuminen OCR-menetelmällä on ymmärrettävästi vaikeampaa verrattuina muihin merkkeihin. Paras keino torjua tällaisia hyökkäyksiä olisi kuvata osa homoglyfeistä niiden yleisemmin tunnettuihin vastineisiin. NLP-mallin ylläpitäjä joutuu tekemään tässä siis suurimman työn. (Boucher et al., 2021)

Uudelleenjärjestelyhyökkäykset voidaan torjua riisumalla kaksisuuntais-ohjausmerkit syötteestä, varoittamalla käyttäjää kaksisuuntais-ohjausmerkkien ilmestyessä syötteeseen



tai käyttämällä kaksisuuntais-algoritmia halutun syötteen selvittämiseen. Puolustusmenetelmän valinta riippuu kontekstista, sillä esimerkiksi latinaa tai arabiaa kirjoittaessa ohjelma toimisi väärin pakottamalla käyttäjän syötteestä pois kaksisuuntais-ohjausmerkin U+200F. (Boucher et al., 2021)

Poistatukset yleensä havaitaan NLP-mallien ulkopuolella syötteenannon alkuvaiheessa. NLP-mallin tasolla tähän tarvitsee harvemmin puuttua ja käyttäjälle voidaan pahimmassa tapauksessa lähettää varoitus poistatusmerkkien olemassaolosta syötteessä. On silti tärkeää tiedostaa poistatuksien puolustus, mikäli käyttöjärjestelmä unohtaa puuttua kyseiseen hyökkäysrajapintaan. (Boucher et al., 2021)

## 5 Yhteenveto

Kävimme läpi tässä tutkimuksessa tekijöitä luonnollisen kielen käsittelyn kehitykseen, joita ovat laskentateho, tietomäärä, koneoppiminen sekä ihmiskielen ymmärrys. Kävimme läpi hyökkäyspinta-alan ja puolustusmahdollisuudet NLP-malleista johtuvia tietoturvauhkia vastaan. Lopuksi käytiin myös läpi tekstipohjaisten vastakkaishyökkäysten tulevaisuutta NLP-malleja vastaan.

Kuten aikaisemmin mainittiin, neljä mahdollistajaa luonnollisen kielen käsittelyyn kuluttajakäytössä ovat laskentatehon kasvu, suurien tietomäärien saatavuus, onnistuneiden koneoppimismenetelmien kehittäminen sekä laajempi ihmiskielen ymmärrys ja käyttö eri konteksteissa. NLP-mallin mahdollistajien kehittyessä arvaamattomasti, on loogista tutkia myös NLP-hyökkäysten tulevaisuutta.

Vastakkaishyökkäysten motiivit muovautuvat siis ajan myötä ja kasvattavat tahtomattaan näin hyökkäystaksonomiaa.

Hyökkäystaksonomia laajentuu tulevaisuudessa eri formaatteihin. Koneoppimisen kukoistaessa voidaan NLP-malleja soveltaa tiedon ääni -tai videoformaatteihin. Tämä antaa puolestaan mahdollisuuden vastakkaishyökätä kyseiseen koneoppimismallia vastaan. Formaattien sisältäkin löytyy erinäisiä hyökkäysrajapintoja. Esimerkiksi ääniformaateissa käytetään kuhunkin käyttötarkoitukseen sopivaa enkoodausta. Ei siis riitä, että hyökättävää ja puolustettavaa tulee uusien formaattien myötä, sillä formaattien sisälläkin tulee tapahtumaan jatkuvasti huomattavaa kehitystä.

Lisäksi haavoittuvuuksien löytö ruokkii itse itseään. Ensimmäisten vastakkaishyökkäysten kohdistuessa uuteen tietformaattiin, syntyy tarve puolustukseen tätä vastaan. Toetuksesta riippuen puolustusmenetelmän selvittäminen saattaa avata uusia ovia, jotka hyödyttävät hyökkääjiä. Usein haavoittuvuuden tarkastelu vastakkaishyökkäyksissä avaa enemmän mahdollisuuksia uusille hyökkäyksille kuin vanhojen hyökkäysten puolustuksille. Tämä näkyy muun muassa GitHub-repositoriossa *Must-read Papers on Textual Adversarial Attack and Defense (TAAD)*, jossa hyökkäystutkimusten määrä suhteessa puolustustutkimusten määrään on 75 : 23 (kuva 4.2).

Luonnollisen kielen käsittely on muovautunut tärkeäksi osaksi tietokoneteollisuutta. Koneoppimisen avulla kuluttajan käyttämästä ihmiskielestä saadaan käyttöön rahanarvoista

## Must-read Papers on Textual Adversarial Attack and Defense (TAAD)

last commit January PaperNumber 110 PRs Welcome

This list is mainly maintained by [Fanchao Qi](#), [Chenghao Yang](#) and [Yuan Zang](#) from [THUNLP](#).

We thank all the great [contributors](#) very much.

### Contents

- [0. Toolkits](#)
- [1. Survey Papers](#)
- [2. Attack Papers](#) (classified according to perturbation level)
  - [2.1 Sentence-level Attack](#)
  - [2.2 Word-level Attack](#)
  - [2.3 Char-level Attack](#)
  - [2.4 Multi-level Attack](#)
- [3. Defense Papers](#)

**Kuva 5.1:** Hyökkäystutkimusten määrä verrattuna puolustustutkimuksiin. Käyty sivulla 14.5.2022 11:16.

mainontatietoa, jota yritys pystyy käyttämään joko itse tai myymään sen eniten tarjoavalle taholle. Tarve ihmiskielen koneelliseen ymmärrykseen on tuonut mukanaan kiinnostuksen lisäksi tietoturvatietoisuutta aiheesta.

Koneoppiminen on todennäköisesti vasta kehitysvaiheen alkupuolella. Jo nyt näkemämme vastakkaishyökkäykset osoittavat useampia haavoittuvaisuuksia, kuin mitä vastaan pysymme puolustautumaan. Selvästi suurin syy tälle on tekoälyn valjastettu voima, jota emme vielä pysty hallitsemaan kaikissa reunatapauksissa.

# Lähteet

- Boucher, N., Shumailov, I., Anderson, R. ja Papernot, N. (2021). *Bad Characters: Imperceptible NLP Attacks*. arXiv: [2106.09898](https://arxiv.org/abs/2106.09898) [cs.CL].
- Chowdhury, G. G. (2003). "Natural language processing". *Annual Review of Information Science and Technology* 37.1, s. 51–89. DOI: <https://doi.org/10.1002/aris.1440370103>. eprint: <https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/aris.1440370103>. URL: <https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/aris.1440370103>.
- Gan, L., Li, J., Zhang, T., Li, X., Meng, Y., Wu, F., Guo, S. ja Fan, C. (2021). "Triggerless Backdoor Attack for NLP Tasks with Clean Labels". *CoRR* abs/2111.07970. arXiv: [2111.07970](https://arxiv.org/abs/2111.07970). URL: <https://arxiv.org/abs/2111.07970>.
- Gopalakrishnan, K. (2018). "Deep learning in data-driven pavement image analysis and automated distress detection: A review". *Data* 3.3, s. 28.
- Hirschberg, J. ja Manning, C. D. (2015). "Advances in natural language processing". *Science* 349.6245, s. 261–266. DOI: [10.1126/science.aaa8685](https://doi.org/10.1126/science.aaa8685). eprint: <https://www.science.org/doi/pdf/10.1126/science.aaa8685>. URL: <https://www.science.org/doi/abs/10.1126/science.aaa8685>.
- Huang, X., Alzantot, M. ja Srivastava, M. (2019). *NeuronInspect: Detecting Backdoors in Neural Networks via Output Explanations*. DOI: [10.48550/ARXIV.1911.07399](https://arxiv.org/abs/1911.07399). URL: <https://arxiv.org/abs/1911.07399>.
- Jordan, M. I. ja Mitchell, T. M. (2015). "Machine learning: Trends, perspectives, and prospects". *Science* 349.6245, s. 255–260.
- Kandasamy, K. ja Korothe, P. (2014). "An integrated approach to spam classification on Twitter using URL analysis, natural language processing and machine learning techniques". Teoksessa: *2014 IEEE Students' Conference on Electrical, Electronics and Computer Science*, s. 1–5. DOI: [10.1109/SCECS.2014.6804508](https://doi.org/10.1109/SCECS.2014.6804508).
- Kuchipudi, B., Nannapaneni, R. T. ja Liao, Q. (2020). "Adversarial Machine Learning for Spam Filters". Teoksessa: *Proceedings of the 15th International Conference on Availability, Reliability and Security*. ARES '20. Virtual Event, Ireland: Association for Computing Machinery. ISBN: 9781450388337. DOI: [10.1145/3407023.3407079](https://doi.org/10.1145/3407023.3407079). URL: <https://doi.org/10.1145/3407023.3407079>.
- Moore, G. E. et al. (1965). *Cramming more components onto integrated circuits*.

- Pruthi, D., Dhingra, B. ja Lipton, Z. C. (heinäkuu 2019). "Combating Adversarial Misspellings with Robust Word Recognition". Teoksessa: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, s. 5582–5591. DOI: [10.18653/v1/P19-1561](https://doi.org/10.18653/v1/P19-1561). URL: <https://aclanthology.org/P19-1561>.
- Zang, Y., Qi, F., Yang, C., Liu, Z., Zhang, M., Liu, Q. ja Sun, M. (2020). "Word-level Textual Adversarial Attacking as Combinatorial Optimization". Teoksessa: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics. DOI: [10.18653/v1/2020.acl-main.540](https://doi.org/10.18653/v1/2020.acl-main.540). URL: <https://doi.org/10.18653/v1/2020.acl-main.540>.

