Kanditutkielma

Tietojenkäsittelytieteen kandiohjelma

Tekstipohjaiset vastakkaishyökkäykset NLP-malleja vastaan

Akira Taguchi

26.9.2022

Matemaattis-luonnontieteellinen tiedekunta

Helsingin yliopisto

Ohjaaja(t)

Prof. Nikolaj Tatti

Yhteystiedot

PL 68 (Pietari Kalmin katu 5)
00014 Helsingin yliopisto

Sähkopostiosoite: info@cs.helsinki.fi
URL: http://www.cs.helsinki.fi/

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET – UNIVERSITY OF HELSINKI

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
Kanditutkielma	26.9.2022	18 sivua

Tiivistelmä — Referat — Abstract

Tekstin automaattisella luokituksella on tärkeä rooli digiyhteiskunnassa. Tämä luokitus tapahtuu luonnollisen kielen käsittelyyn pohjautuvilla metodeilla. Metodeihin pohjautuvat malli ovat kuitenkin haavoittuvaisia, ja tässä tutkimuksessa käsitelläänkin tekstipohjaisia vastakkaishyökkäyksiä NLP-malleja vastaan. Tutkielman alussa perehdytään automaattisen luokituksen käyttötarkoituksiin. Tämän jälkeen tutustutaan hyökkäystaksonomiaan, eli erilaisiin hyökkäysmetodeihin NLP-malleja vastaan. Lopuksi käsitellään puolustusmetodeita NLP-hyökkäyksiä vastaan.

ACM Computing Classification System (CCS)
Security and privacy → Human and societal aspects of security and privacy
Computing methodologies → Artificial Intelligence → Natural language processing

Avainsanat — Nyckelord — Keywords

Luonnolllisen kielen käsittely, vastakkaishyökkäys, koneoppiminen, tekoäly, ladonta, sensuuri

Säilytyspaikka — Förvaringsställe — Where deposited

Helsingin yliopiston kirjasto

Muita tietoja — övriga uppgifter — Additional information

Sisältö

1 Johdanto

Koneoppimisen käyttötarkoitusten määrä kasvaa vuosi vuodelta suuremmaksi. Tätä teknologiaa voidaan hyödyntää muun muassa ihmisten puhuman kielen käsittelyyn. 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 yksilöllle. Viesti ystävälle viestipalvelussa antaa työstettävän datan NLP-mallille: “Mikä elokuva meidän pitäisi katsoa viikonloppuna? “ NLP-mallin avulla automaattinen mainostaja ymmärtää mainostaa kyseiselle käyttäjälle esimerkiksi suoratoistopalvelua tai sarjalippuja mainostavaa elokuvateatteria. Tämän tiedon löytäminen suuresta määrästä dataa luonnollisen kielen käsittelyllä 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 Tekstin automaattinen luokitus

NLP-luokittimia, eli NLP-malleja käytetään analysoimaan tekstiä, joissa on tehokkaampaa korvata ihmisen manuaalisesti tekemä analysointityö. Ensin käydään läpi neljä yleistä tapausta tekstin automaattisesta luokituksesta. Nämä neljä tapausta ovat roskapostin suodatus sähköposteista, vihapuheen sensurointi sosiaalisesta mediasta, valearvosteluiden tunnistus nettikauppojen arvosteluosioista sekä sentimenttianalyysi. Lopuksi käydään läpi tekstin automaattisen luokituksen edut verrattuna manuaaliseen, ihmisen tekemään luokitustyöhön.

2.1 Roskapostien suodatus

Sähköpostien automaattinen luokitus roskaposteiksi tai kelpoposteiksi onnistuu NLP-mallien avulla. Noin 70% liiketoiminnan sähköposteista on roskapostia. Näiden roskapostien tarkoitus voi muun muassa olla petkutusta, ärsyttämistä tai loukkaamista (Garg ja Girdhar, 2021).

Roskapostin vaikutukset käyttäjästä riippuen ovat niin vakavia, että sähköpostipalvelun tarjoajan intresseissä on implementoida roskapostisuodatin. Roskapostit saattavat sisältää viestin avaajaa järkyttävää tai provosoivaa mediaa. Roskaposti saattaa sisältää myös kalasteluyrityksen. Kalasteluhyökkäyksessä tarkoituksena on petkuttaa käyttäjää antamaan erilaisia tunnus-salasana-yhdistelmiä liittämällä roskapostiin esimerkiksi linkin viralliselta näyttävältä sivulle. Sivulla käyttäjää kehotetaan kirjautumaan tunnuksillaan tuttuun palveluun, mutta oikeasti palvelu vain varastaa käyttäjän tunnukset. Roskaposti saattaa myös sisältää haittaohjelmia, joita käyttäjä voi saada koneelleen muun muassa lataamalla ja suorittamalla sähköpostin tiedostoja tai vierailemalla pahantahtoisella sivustolla. Tämä pahantahtoinen sivusto usein sisältää koodia, joka hyväksikäyttää usein jotain selaimen haavoittuvaisuutta esimerkiksi asentaakseen tietokoneelle haittaohjelmia. Myös kiristysviestejä sekä sähköposteja eteenpäinlähettäviä haittaohjelmia kulkee roskapostien mukana, joita sähköpostipalvelun tarjoajat pyrkivät estämään roskapostisuodattimilla.

2.2 Vihapuheen sensurointi

Vihapuheen riittävään sensurointiin tarvitaan luonnollisen kielen käsittelyä. Suodattimen rakentaminen vihapuhetta vastaan pelkkien avainsanojen perusteella ei tuota toivottuja tuloksia, koska katsotun vihapuheen sensuroinnille tarvitaan muun muassa meneillään olevan keskustelun suunta, tarkka ajanhetki, ajankohtaiset maailman tapahtumat, lähettäjän sekä vastaanottajan henkilöllisyys sekä kontekstuaaliset mediat, esimerkiksi kuvat, videot tai ääni (Schmidt ja Wiegand, 2017).

2.3 Valearviointien tunnistus

Ostosten tekemisten mahdollisuuden netissä sekä tuotteiden hyvän saatavuuden vuoksi kuluttajat joutuvat perustelemaan ostopäätöksensä yhä useammin tuotearvosteluihin. Oikeiden arvosteluiden lisäksi tuotesivulla saattaa olla valearvosteluja. Luonnollisen kielen käsittelyyn perustuvalla tekniikalla voidaan kyseiset valoarvostelut tunnistaa ja tuhota (Trupthi et al., 2019).

Valearvostelujen määrä ja kieliasu ovat pääsyyt NLP-luokittimien käyttöön edellä mainitussa käyttökohteessa. Valearvosteluita voidaan tuottaa eri syistä. esimerkiksi tuotteen näennäisen arvon laskeminen kilpailullisen tuotteen näennäisen arvon nostamiseksi. Valearvosteluita voidaan myös tehdä myös pelkästään pahantahtoisella tarkoituksella alentaa tuotteen näennäisarvoa. Luokitin tunnistaa suuresta määrästä arvosteluja valearvostelut, vaikka kieliasu ei olisikaan formaali. Tässäkään tapauksessa pelkkä avainsanafiltteri ei riitä tunnistamaan valearvosteluita aidoista arvosteluista. Valearvostelujen tunnistus luonnollisen kielen avulla on markkinallisista syistä verkkokauppojen intresseissä.

2.4 Sentimenttianalyysi

Tunnesävyn tunnistaminen omasta -tai kilpailijan tuotteesta tuottaa arvokasta tietoa tuotekehitykselle sekä markkinointi -ja asiakassuhteen ylläpidolle. Kuluttajien sentimentaalisuuden analysoiminen automaattisesti luonnollisen kielen käsittelyn tekniikoiden avulla on kustannustehokasta. Manuaalisesti tunnesävyjen selvittäminen jokaisesta olennaisesta netissä olevasta tekstistä vaatisi paljon resursseja eikä välttämättä tuottaisi yhtä paljon tai yhtä laadukkaita tuloksia, kuin automaattinen sentimentaalisuuden analysoiminen (Yi

et al., 2003).

2.5 Manuaalinen luokitus

Edellä käyty tarkastustyö voitaisiin tehdä manuaalisesti, mutta tarkastettavan sisällön määrän vuoksi tämä ei käytännössä ole kannattavaa ja useammissa tapauksissa onkin miltei mahdotonta. Käyttäjä 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. Automaattisella roskapostin lähetyksellä tarkoitetaan tietokoneella ohjelmoitua sähköpostien lähettämistä eri sähköpostiosoitteisiin. Usein nämä sähköpostiosoitteetkin ovat hankittu tietokoneohjelmoinnin avulla, joten roskapostia lähetetään päivittäin paljon. Valearviointien manuaalisessa tunnistuksessa on samanlaisia ongelmia kuin roskapostitunnistuksessa. Vihapuheen sensurointi manuaalisesti vaatii kontekstuaalista ymmärystä keskustelusta. Käytännössä tämä vaatisi yhdeltä tarkastajalta aiheen tutkimista sekä mahdollisiin uusiin vihapuhesanoihin tai vihapuhelauseisiin tutustumista. Kuten edellä mainittu sentimenttianalyysin tekeminen manuaalisesti olisi miltei mahdotonta. Yksittäisen keskutelun tai aiheen tutkiminen ei riittäisi analysoimaan tunnesävyjä, vaan tarkastajan täytyisi käydä läpi mahdollisimman monta netin tekstiä ja analysoida näistä tunnesävyt. Aikaisemmissa alikappaleissa olemme käyneet läpi näiden tekstien automaattisen luokituksen pääpiirteet. Tekstin automaattisella luokituksella on kuitenkin heikkous tietoturvapuolella.

3 Hyökkäystyypit

NLP-malleja vastaan, jotka ovat automaattisen tekstin luokituksen keskiössä, voidaan hyökätä. Tässä kappaleessa käydään läpi hyökkästaksonomia, eli hyökkäysrajapinta, NLP-malleja vastaan. Ensin käydään läpi roskapostisuodatuksen roskapostisuodatuksen ohitus, joka on NLP-hyökkäysten keskiössä. Sitten esitellään sensuuriohitus sekä ladontahyökkäykset. Ladontahyökkäyksistä käydään läpi näkymättömät merkit, homoglyfit, uudelleenjärjestelyt sekä poistatukset.

3.1 Roskapostisuodatuksen ohitus

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

Suodattimia vastaan toimii kolme vastakkaishyökkäystä: (1) Synonyymin korvaus, (2) kelposanan injektointi sekä (3) roskapostisanojen väljennys. Sana ”kelpo” tarkoittaa tässä yhteydessä tekstiä, jonka roskapostisuodatin on merkinnyt hyväntahtoiseksi. Synonyymin korvauksessa tarkoitus on korvata pahantahtoiset sanat hyväntahtoisiksi luokitelluilla synonyymeillä. Lauseiden samankaltaisuuksien vertailua demonstroidaan taulukossa 3.1. Pahantahtoisissa lauseissa pyritään nostattamaan samankaltaisuusastetta vaihtamalla sanoja synonyymeihin, kunnes NLP-malli tunnistaa viestin olevan kelpopostia. 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

Muokattu viesti	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: **acquire** the UK single **graph** on your **Mobile_River** each **hebdomad** and **take** any **top_side caliber** ringtone! This **content** is **free_people** of charge.	0,583	roskapostia
Ringtone Club: **become** the UK **bingle graph** on your **nomadic** each **workweek** and **select** any **upper_side caliber** ringtone! This **subject_matter** is **liberate** of charge.	0,583	roskapostia
Ringtone Club: **go** the UK **one graph** on your **peregrine** each **calendar_week** and **pick_out** any **upside character** ringtone! This **substance** is **release** of charge.	0,583	kelpopostia

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

ja vaihtelemalla, saattaa NLP-mallin pahantahtoisuuden havaitseminen heikentyä entisestään, taaten hyökkääjälle varmemman onnistumisen.

3.2 Sensuurin ohitus

Koska sensuuria voidaan soveltaa hyödyntäen koneoppimismalleja, voidaan sensuuri myös ohittaa hyödyntäen koneoppimismallin heikkouksia. Vastakkaishyökkäys voisi tunnistaa sensurointia aiheuttavia tekstiyhdistelmiä, 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 uupuessa esimerkiksi tekstipohjaisessa vastakkaishyökkäyksessä mahdollistaa myös helpomman puolustautumisen (Pruthi et al., 2019). Seuraavissa alikappaleissa käydään läpi ladontatason vastakkaishyökkäysmetodeita. Näitä yhdistelemällä, jopa alan johtava vihapuhefiltteri Google Perspective, on altis sensuurin ohitukselle vastakkaishyökkäyskoulutuksesta huolimatta (Gröndahl et al., 2018). Koulutus vastakkaishyökkäyksiä vastaan tässä kontekstis-

sa tarkoittaa puolustavan NLP-mallin koulutusta syötteellä, joka yrittäisi hyökätä NLP-mallia vastaan. Kyseessä on siis NLP-malli, jonka luokitus lujittuu vastakkaishyökkäyksiä tuottavan NLP-mallin ulostulolla.

3.3 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.

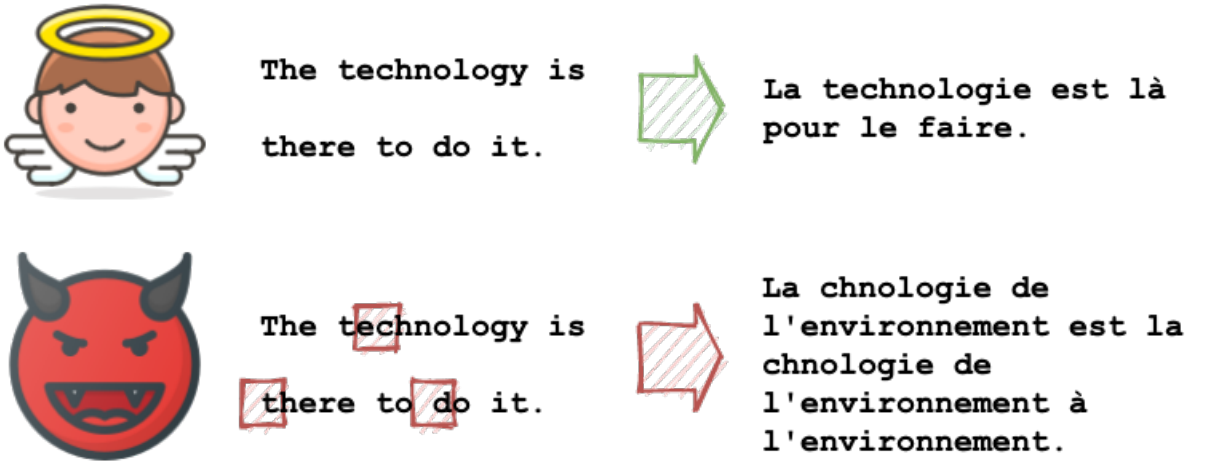
Esimerkki tällaisesta erikoismerkistä on nollatilavuuden välilyönti -merkki, jonka Unicode merkintä on `U+200B`. Tällä merkillä voimme esimerkiksi vaikuttaa pelichattiin lähetettävän myrkyllissuodatettavaan merkkijonoon "olet huono"niin, että merkkijono menisi NLP-mallin läpi chätistä. Merkkijono `olU+200Bet 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.

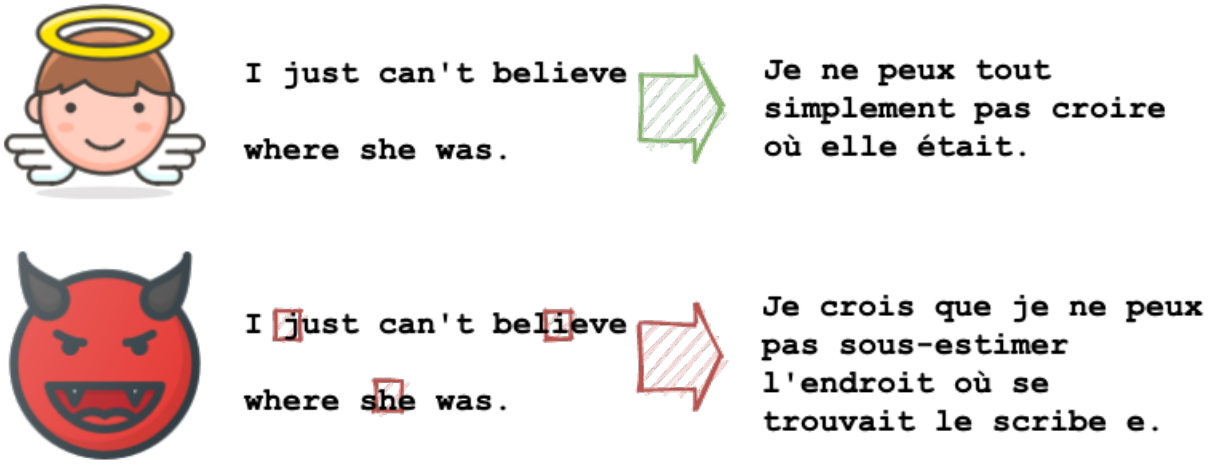
3.4 Homoglyfit

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 A. Kuvassa 3.2 homoglyfihyökkäys on muuntanut englanninkielisen tekstin `I just can’t belive where she was` ranskankieliseen käännökseen



Kuva 3.1: Hyökkäys näkymättömillä merkeillä (Boucher et al., 2021)

`I guess I can’t underestimate the location of the scribe and.`

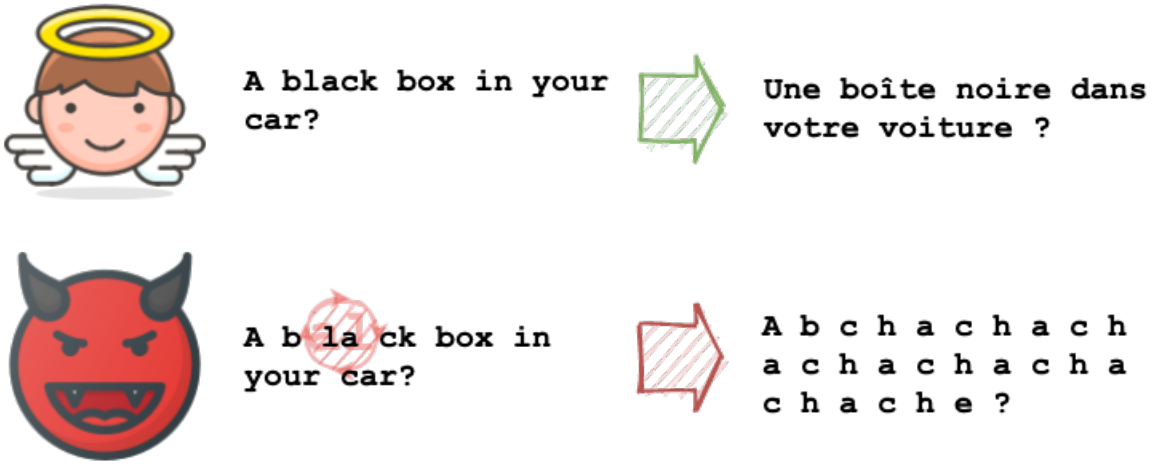


Kuva 3.2: Homoglyfihyökkäys (Boucher et al., 2021)

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

3.5 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 käytetään myös NLP-mallin sekoittamiseen, jolloin tulokset NLP-mallista ovat käyttökelvot-



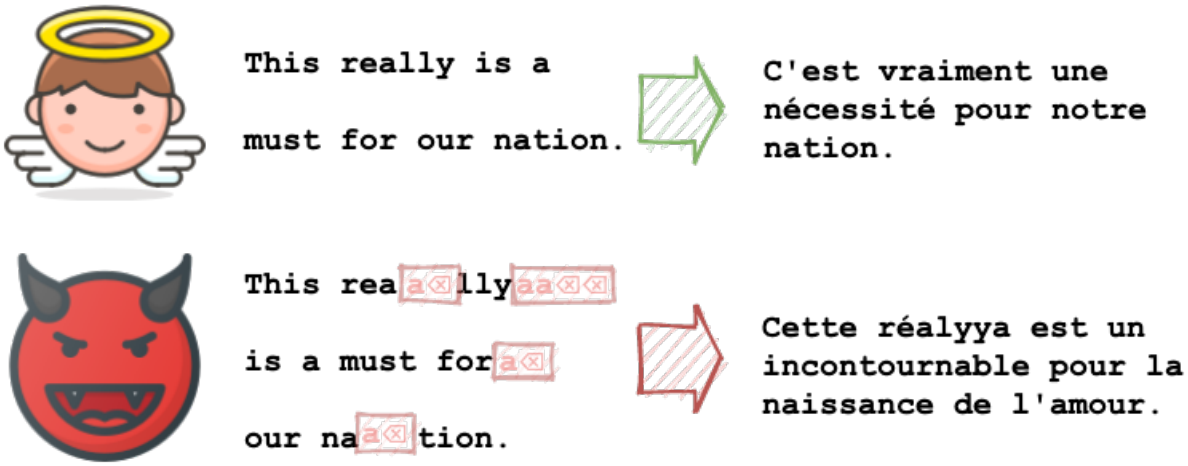
Kuva 3.3: Homoglyfihyökkäys (Boucher et al., 2021)

tomia. Kuvassa 3.3 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.

3.6 Poistatukset

Viimeisenä käydään läpi poistatushyökkäykset. Poistatushyökkäyksen tarkoituksena on poistaa käyttäjälle näkyvästä tekstistä ladontavaiheessa haluttu määrä tekstiä pois. Uhri esimerkiksi voisi olla myymässä pois asuntoaan, jolloin tämä kopioi ja liittää sähköiseen sopimukseen hyökkääjän ehdottaman summan. Latomisvaiheessa käyttäjä kuitenkin unohtaa tarkistaa sopimuksen, jolloin poistatusmerkit saattavat poistaa myyntihinnasta esimerkiksi muutaman nollan.

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). (Boucher et al., 2021) Kuva 3.4 havainnollistaa poistatushyökkäystä käytännössä. Esimerkissä poistatusmerkkejä on pistetty sanojen väliin, muuttaen näin ladottujen lauseiden merkityksen.



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

4 Hyökkäyksiltä suojautuminen

Tässä kappaleessa käydään läpi erilaisia puolustusmenetelmiä NLP-hyökkäyksiä vastaan. NLP-hyökkäykset voidaan torjua korkean tason abstraktiolla, suurella yleisrasituksella, sekä alemman tason abstraktiolla, pienemmällä yleisrasituksella. Ensiksi esitellään korkean tason abstraktion OCR-puolustus, sitten kuvaillaan alemman tason abstraktion suorituskeskeinen puolustautuminen.

4.1 OCR-puolustus

OCR-metodia (eng. OCR, Optical character recognition), eli tekstintunnistusta on käytetty esimerkiksi printatun, skannatun tai käsinkirjoitetun tekstin muuntamiseen muokattavaksi tekstiksi. Joskus tekstint tunnistaminen on vaikeata johtuen esimerkiksi tekstin eri koista, tyyleistä, suuntauksesta tai monimutkaisesta tekstin taustasta. Esimerkiksi nollamerkki ”0” ja aakkonen ”o” voivat olla jollekkin OCR-työkalulle vaikeita erottaa toisistaan. Kyseistä virhelukua havainnollistetaan kuvassa 4.1 ja 4.2. Eri OCR-työkalut saavuttavat tekstintunnistuksen eri tavoilla, mutta esimerkiksi avoimen lähdekoodin OCR-työkalu Tesseractilla on seuraavat vaiheet tekstintunnistukseen kuvasta: (1) kuvan muuntaminen binäärikuviksi, (2) merkkien ääriviivojen tunnistamiset, (3) merkkien ääriviivojen tunnistaminen sanoiksi, (4) sanat tunnistetaan kahteen kertaan, jonka jälkeen teksti on tunnistettu kuvasta. Kuva 4.3 havainnollistaa edelläkuvattua Tesseractin tekstintunnistusprosessia (Patel et al., 2012).

OCR-puolustuksen läpikäynnin jälkeen voidaan huomata metodin eliminoivan huomattavan monta NLP-hyökkäystä. Näkymättömät merkit, poistatukset sekä uudelleenjärjestelymerkit poistuvat kokonaan tekstintunnistuksen jälkeen tuotetusta tekstistä, koska kyseiset merkit eivät ihmissilmällekkään näkyisi. Roskapostisuodatuksen -ja sensuurin ohituksen lisäksi OCR-puolustus alisuoriutuu eräällä toisellakin osa-alueella.

Tekstintunnituksen avulla epäselvyydet tekstin aidosta luonteesta voidaan hahmontaa uudelleen tulkitsemalla aineisto uudestaan visuaalisesti. Tämä menetelmä lisää yleisrasitusta huomattavasti riippuen käyttötarkoituksesta, mutta poistaa pahantahtoiset merkit ilman NLP-mallin uudelleenkoulutusta. (Boucher et al., 2021)

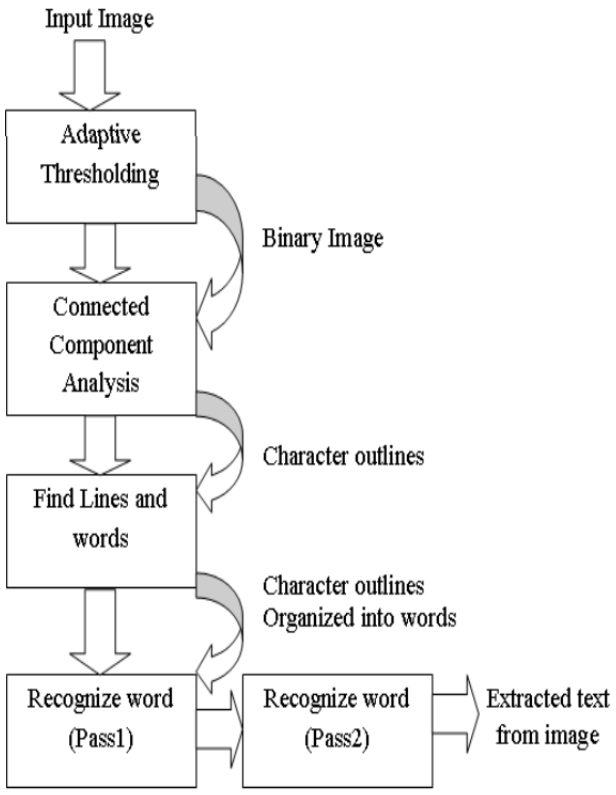
Kuva 4.1: Tekstitunnistettava kuva (Patel et al., 2012)

Kuva 4.2: Tesseract epäonnistuu tunnistamaan ylätekstin ”Effect” kuvasta (Patel et al., 2012)

4.2 Suorituskykykeskeinen puolustus

Keskitymme seuraavaksi näkymättömiin merkkeihin, -homoglyfeihin, uudelleenjärjestelyihin -ja poistatuksiin perustuvien hyökkyksien puolustamiseen. Osat suorituskykykeskeisistä puolustusmenetelmistä ovat kuitenkin laskennallisesti kalliita, eivätkä koneoppimismallin ulkoistaneet yritykset yleensä pysty kustantamaan kyseisiä metodeita (Huang et al., 2019).

Näkymättöimen merkkein tapauksessa, tietyt näkymättömät merkit voidaan poistaa suoraan syötteestä. Mikäli sovelluksessa näitä merkkejä ei voida poistaa, voidaan ne korvata *ei*-`<unk>` upotuksilla. Korvaus tapahtuu lähdekielisanakirjassa, jonne kuvataan tuntematon merkki ”ei-tuntemattomaksi tokeniksi”. Näin tuntemattomat merkit eivät voi häi-



Kuva 4.3: Tesseractin tekstintunnistuksen vaiheet (Patel et al., 2012)

ritä ladontaa merkeillä, joista ladontamoottori ei ole aivan varma. 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. 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`. Poistatukset yleensä havaitaan NLP-mallien ulkopuolella syötteen annon 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)

Roskapostisuodattimeen -ja sensuurin ohitukseen kohdistuvat hyökkäykset voidaan yrit-

tää torjua muun muassa DNE-metodilla (eng. Dirichlet Neighborhood Ensemble). Metodissa korvataan virtuaalisten lauseiden sanoja näiden synonyymeillä. Tämän jälkeen puolustava NLP-malli koulutetaan kyseisiä lauseita vastaan. Metodin tarkoituksena on siis puolustautua synonyymin korvausta vastaan, joka esiteltiin alikappaleessa 2.1, Roskapostisuodatuksen ohitus (Zhou et al., 2020).

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 tietoformaattiin, syntyy tarve puolustukseen tätä vastaan. Toteutuksesta riippuen puolustusmenetelmän selvittäminen saattaa avata uusia ovia, jotka hyödyttävät hyökkääjiä. Usein haavoittuvuuden tarkastelu vastakkaishyökkäyksissä avaa enemmän mahdollisuuksia uusille hyökkäyksille kuin vanhojen hyökkäysten puolustuksille.

Luonnolisen kielen käsittely on muovautunut tärkeäksi osaksi tietokoneteollisuutta. Koneoppimisen avulla kuluttajan käyttämästä ihmiskielestä saadaan käyttöön rahanarvoista mainontatietoa, jota yritys pystyy käyttämään joko itse tai myymään sen eniten tarjoavalle taholle. Rahanarvoisen hyödyn lisäksi luonnollisen kielen käsittely tarjoaa myös yleishyödyllisiä ratkaisuja, kuten vihapuheen esto ja roskapostisuodattimet. Tarve ihmiskielen koneelliseen ymmärrykseen ja haluun valjastaa kestävästi sen hyödyt ovat tuoneet mukanaan kiinnostuksen luonnollisen kielen käsittelyn tietoturvaan.

Lähteet

Boucher, N., Shumailov, I., Anderson, R. ja Papernot, N. (2021). *Bad Characters: Imperceptible NLP Attacks*. arXiv: 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. URL: https://arxiv.org/abs/2111.07970.

Garg, P. ja Girdhar, N. (2021). "A Systematic Review on Spam Filtering Techniques based on Natural Language Processing Framework". Teoksessa: *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, s. 30–35. DOI: 10.1109/Confluence51648.2021.9377042.

Gopalakrishnan, K. (2018). "Deep learning in data-driven pavement image analysis and automated distress detection: A review". *Data* 3.3, s. 28.

Gröndahl, T., Pajola, L., Juuti, M., Conti, M. ja Asokan, N. (2018). "All You Need is "Love": Evading Hate Speech Detection". Teoksessa: *Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security*. AISec '18. Toronto, Canada: Association for Computing Machinery, s. 2–12. ISBN: 9781450360043. DOI: 10.1145/3270101.3270103. URL: https://doi.org/10.1145/3270101.3270103.

Hirschberg, J. ja Manning, C. D. (2015). "Advances in natural language processing". *Science* 349.6245, s. 261–266. DOI: 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. 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.

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. URL: https://doi.org/10.1145/3407023.3407079.

Moore, G. E. et al. (1965). *Cramming more components onto integrated circuits*.

Patel, C., Patel, A. ja Patel, D. (2012). "Optical character recognition by open source OCR tool tesseract: A case study". *International Journal of Computer Applications* 55.10, s. 50–56.

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. URL: https://aclanthology.org/P19-1561.

Schmidt, A. ja Wiegand, M. (huhtikuu 2017). "A Survey on Hate Speech Detection using Natural Language Processing". Teoksessa: *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*. Valencia, Spain: Association for Computational Linguistics, s. 1–10. DOI: 10.18653/v1/W17-1101. URL: https://aclanthology.org/W17-1101.

Trupthi, M., Pabboju, S. ja Gugulotu, N. (2019). "Deep Sentiments Extraction for Consumer Products Using NLP-Based Technique". Teoksessa: *Soft Computing and Signal Processing*. Toim. J. Wang, G. R. M. Reddy, V. K. Prasad ja V. S. Reddy. Singapore: Springer Singapore, s. 191–201. ISBN: 978-981-13-3393-4.

Yi, J., Nasukawa, T., Bunescu, R. ja Niblack, W. (2003). "Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques". Teoksessa: *Third IEEE International Conference on Data Mining*, s. 427–434. DOI: 10.1109/ICDM.2003.1250949.

Zhou, Y., Zheng, X., Hsieh, C.-J., Chang, K.-w. ja Huang, X. (2020). *Defense against Adversarial Attacks in NLP via Dirichlet Neighborhood Ensemble*. DOI: 10.48550/ARXIV.2006.11627. URL: https://arxiv.org/abs/2006.11627.