



TikToks muskelkraft

NRK Undersøkende Gruppe Sør-Øst
Data-SKUP 2023

Innsendere

Christian Nicolai Bjørke	christian.nicolai.bjorke@nrk.no	93226641
Henrik Bøe	henrik.boe@nrk.no	90256884
Caroline Utti	caroline.utti@nrk.no	40016954

Tittel på prosjektet

TikToks muskelkraft <https://www.nrk.no/1.16183933>

Publisert

Fra desember 2022 til januar 2023.

Reportasjeleder

Tuva Skei Tønset	tuva.skei.tonset@nrk.no	47640456
------------------	--	----------

Redaktør

Anne Vinding	anne.vinding@nrk.no	97593944
--------------	--	----------

Takk til Anne Vinding, Tuva Skei Tønset, Rina Therese Blekkerud, Ingrid Reime, Stian Veum Møllersen, Martin Gundersen, Thomas Nikolai Blekeli, Per Arne Kalbakk og alle i Undersøkende Gruppe i Sør-Øst og de som er på «hjemmekontorene» våre i Fredrikstad, Tønsberg, Drammen og Ål.

INNLEDNING	1
BAKGRUNN: EN TUR INNOM RUSSLAND OG UKRAINA	1
TIKTOK-ROBOT 2.0	2
BARNA I KOMMENTARFELTENE	3
Nedlasting av hundretusenvis av TikTok-kommentarer	4
Finne barna som oppga egen alder	4
Tallet TikTok ikke viste	5
ALGORITME-EKSPERIMENTET, DEL 1	6
FRA DATA TIL HISTORIE	7
ALGORITME-EKSPERIMENTET, DEL 2	7
ANALYSE	8
Ny bruk av videoverktøyet	8
Gjentakelse av eksperimentene	9
Mulige feilkilder	9
ETIKK: DERFOR BRØT VI TIKTOKS RETNINGSLINJER	9
NYE FUNN OG KONSEKVENSER	10
VEDLEGG	11
Vedlegg 1: Oversikt over publisering og oppfølgere	11
Vedlegg 2: Kode-eksempler på GitHub	11

Innledning

TikTok er den plattformen som øker mest blant barn og unge i Norge. Ifølge 2022-tall fra Medietilsynet har en halv million norske barn mellom 9 og 18 år brukerkonto på verdens største app. I løpet av få år har TikTok revolusjonert hvordan vi ser innhold i sosiale medier. Før søkte man etter det man ville se på. På TikTok serverer derimot algoritmene deg det den tror du liker. Samtidig er det kinesisk-eide selskapet som står bak TikTok vage om hvordan algoritmene fungerer¹.

I denne DataSkup-metoderapporten vil vi legge mest vekt på datatekniske metoder og digitale fortellerteknikker. For en mer utfyllende drøfting rundt de etiske vurderingene vi tok i møte med unge kilder, se bidraget vi leverte til Skup tidligere i år². For å gjøre teksten mest mulig lesbar har vi samlet mye av koden og SQL-spørringene i et eget GitHub-repository³.

«TikToks muskelkraft» bygger på det tidligere prosjektet «Én app – to verdener»⁴. Vi vil derfor først gi en kort presentasjon av metodene i det prosjektet slik at det blir lettere å forstå hvordan vi utviklet disse i det neste.

Bakgrunn: En tur innom Russland og Ukraina

«Én app – to verdener» ble satt i gang få dager etter at Russland gikk til angrep på Ukraina i februar 2022. TikTok var ett av få sosiale medier som ikke ble forbudt i Russland, og vi hadde en hypotese om at unge russiske og ukrainske TikTok-brukere fikk servert helt forskjellige nyheter fra krigen.

Vi utviklet følgende metode for å undersøke hypotesen:

- «Residential proxies» for å få TikTok til å tro at vi var til stede i Russland og Ukraina. En proxy er en slags mellomtjener som lar oss velge fysiske maskiner andre steder i verden vi kan sende internett-trafikken vår gjennom. På denne måten kunne vi opprette nye TikTok-brukere i de to landene.
- En robot skrevet i programmeringsspråket Python som kunne gå inn på nettsiden tiktok.com og oppføre seg som en vanlig TikTok-bruker, vise interesse for enkelte videoer og lagre informasjonen.
- Kunstig intelligens fra bildegjenkjenningstjenesten til Microsoft Azure til å identifisere videoer med krigshandlinger. Roboten ville se lenger på disse videoene for å gi TikTok et hint om at den var interessert i slikt innhold.

Etter å ha analysert dataene og gjort flere tester kunne vi avsløre at TikTok ikke viste krigen i Ukraina til russiske brukere.

¹ <https://newsroom.tiktok.com/en-gb/how-tiktok-recommends-videos-foryou-eu>

² Skup 2023: <https://www.skup.no/sites/default/files/2023-03/NRK%20gravegruppa%20S%C3%B8r%20C3%98st%20-%20TikToks%20muskelkraft.pdf>

³ GitHub: <https://github.com/cbjoerke/muskelkraft>

⁴ Data-Skup 2022: <https://www.skup.no/rapporter/data-skup-2022/en-app-verdener>

TikTok-robot 2.0

Russland-Ukraina-prosjektet handlet om hva TikTok *ikke* viste. Nå ville vi se nærmere på hva de faktisk *viste*.

Nylig hadde nemlig TikTok lovet å arbeide for at algoritmene ikke skulle servere mye av én type innhold⁵, og et av eksemplene de nevnte var ekstrem trening. Vi spurte oss selv om det faktisk var slik? Eller om også norske barn kunne få opp så mange treningsvideoer at det ville føre til kroppspress?

Det var ikke mangel på folk som synset om temaet, men vi ønsket å utarbeide en metode som kunne gi dokumentasjon. Vi bestemte oss samtidig for å se det hele fra en gutts perspektiv, siden vi opplevde at mye av journalistikken hadde handlet om hva jenter fikk se i sosiale medier. Først brukte vi noen dager på TikTok for å gjøre oss kjent med muskelmiljøet og systematisere mulige kilder, emneknagger og ord og uttrykk i et Excel-ark. Deretter formulerte vi en hypotese:

En norsk gutt som viser interesse for muskler på TikTok vil i løpet av kort tid få servert store mengder ensidig innhold.

Vi bygget på erfaringene fra det forrige prosjektet, men det var to ting vi måtte gjøre før vi kunne gå i gang:

- Oppgradere roboten vår til å bruke TikTok-appen i stedet for tiktok.com.
- Utvikle en ny metode for å dokumentere at norske barn faktisk får se ekstrem trening på TikTok.

Da vi testet Russland-Ukraina-roboten på tiktok.com, opplevde vi at det tok ganske lang tid før algoritmene tilpasset innholdet til brukeren. Vi hadde derimot inntrykk av at det gikk ganske fort på mobilappen. Kunne det være en forskjell her?

I Russland-Ukraina-saken hadde det ikke vært så viktig hvilken versjon vi brukte. Her var hovedpoenget å vise hvor forskjellig innhold TikTok serverte helt nye brukere, og vi dokumenterte at mesteparten av krigsinholdet var sperret for en russisk bruker. Men siden nesten alle bruker TikTok via en app, ønsket vi å få den nye roboten til å gjøre det samme. Vi forsøkte først å lage en løsning der vi brukte Android-emulatorer (en slags virtuell mobiltelefon) for å kjøre TikToks Android-app, og automatiseringsløsningen Appium til å styre den. Men vi fikk problemer med at appen ikke fungerte. Flere forsøk i ulike emulatorer og søk i programmeringsforum på internett tydet på at TikTok skjønnte at den kjørte i en emulator, og derfor lukket seg. Løsningen ble å bruke en fysisk Android-mobiltelefon som ble styrt fra en datamaskin.

Det er enklere å hente ut all tekst og emneknagger fra tiktok.com enn fra appen. Løsningen med Appium er mer ustabil og noen ganger bruker den lang tid på å hente teksten. Siden hensikten med å hente ut informasjon om hver video var å bruke dette til å avgjøre om roboten skulle sveipe videre eller ei, dukket det derfor opp et dilemma: På den ene siden ville vi hente ut mest mulig info om videoen, men det kunne ta 6-8 sekunder å både hente ut

⁵ <https://newsroom.tiktok.com/en-us/an-update-on-our-work-to-safeguard-and-diversify-recommendations>

tekst og hente inn informasjon via kunstig intelligens. På den andre siden ville vi raskt sveipe videre, slik folk flest gjør, hvis videoen ikke matchet robotens interesser.

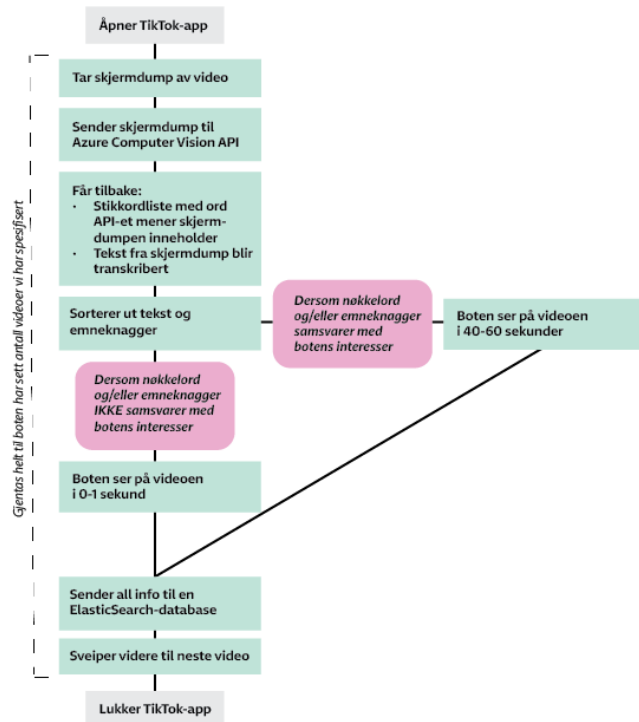
I TikTok-sammenheng er 6-8 sekunder på hver video ganske lenge, og vi fryktet at det ville gi et signal til TikToks algoritmer om at vi var interessert i alle videoene. Derfor droppet vi å hente tekst med Appium og gikk for en løsning basert på skjermdumper⁶. Vi fikk likevel lagret teksten fordi vi sendte skjermdumpen til Microsofts Azure Computer Vision API som leste teksten på bildet ved å kjenne igjen bildet av bokstavene. Metoden kalles Optical Character Recognition (OCR). Da fikk vi også hentet inn tekst som var «brent inn» i videoen, noe som er vanlig på TikTok. Løsningen fungerte som vist på bildet til høyre.

Det var utfordrende å unngå at TikTok oppdaget at appen ble styrt av en datamaskin. Vi erfarte blant annet at appen lukket seg dersom robotens sveip ble for like eller for raske. I utviklerinnstillinger på Android-telefoner kan man velge å se koordinatene for hvor en finger treffer skjermen. Vi brukte dette til å finne posisjoner der det er naturlig at høyre tommel er på starten og slutten av hvert sveip. Koordinatene ble brukt til å styre hvor vår virtuelle finger skulle starte og avslutte hvert sveip. For å unngå robot-oppførsel ble koordinatene endret litt for hvert sveip ved hjelp av det innebygde Random-biblioteket i Python. De ble satt til en tilfeldig verdi innenfor området det er naturlig at fingeren befinner seg. Dermed var 2.0-versjonen av TikTok-roboten nærmest klar.

Imens hadde vi også jobbet med en metode for å dokumentere at norske barn var aktive i kommentarfeltene på TikTok-kontoene til norske kroppsbyggere.

Barna i kommentarfeltene

Den nedre aldersgrensa for å opprette en bruker på TikTok er 13 år. Vi vet at mange yngre barn likevel er på plattformen og utgir seg for å være eldre. Men vi ønsket ikke at TikTok skulle peke på foreldre eller andre som ansvarlige for aktiviteten på deres app. I det nye prosjektet valgte vi derfor å la TikTok-roboten vår være en fiktiv 13-åring med en interesse for å bygge muskler. Målet var å se hvor algoritmene ville lede ham.



⁶ En skjermdump er en funksjon som tar bildet av skjermen, i vårt tilfelle det som ble vist på mobilskjermen.

Men før vi kunne gjøre det, måtte vi sjekke om det faktisk var barn på denne alderen som var engasjert i å bygge muskler på TikTok. Men hvordan skulle vi få til det? TikTok viser ikke alderen til brukerne sine. En spørreundersøkelse ville heller ikke gi oss den dokumentasjonen vi ønsket oss. Men etter å ha brukt mange timer i kommentarfeltene til voksne muskelmenn på TikTok, så vi et mønster. Flere oppga alderen sin når de spurte om tips og råd til hvordan de skulle trene. For eksempel: «Er det bra at jeg benker 70 kg som 15-åring?». Kunne det være mulig å systematisere slike opplysninger?

Nedlasting av hundretusenvís av TikTok-kommentarer

Siden starten av prosjektet hadde vi samlet norske og utenlandske TikTok-kontoer som tilhørte voksne menn som fokuserte på å bygge muskler. Kontoene fant vi ved å følge anbefalinger og tagginger på TikTok. Nettadressen til brukeren ble lagt inn i to Excel-ark underveis, sammen med all informasjonen TikTok oppgir på hver bruker: Navn, brukernavn, antall følgere og antall likerklipp. Etter hvert satt vi med en liste med flere hundre kroppsbyggere. Vi brukte filtreringsfunksjonen i Excel for å sortere kolonnen med antall likerklipp fra størst til minst, siden det gir en indikasjon på hvilke kontoer som er mest populære. Deretter valgte vi de 50 norske kroppsbyggerne med flest likerklipp, i tillegg til de 50 utenlandske kroppsbyggerne med flest likerklipp. Det ga oss et utvalg på hundre populære kontoer. Under videoene deres antok vi at det fantes kommentarer der barn selv oppga hvor gamle de var. Derfor startet vi på en oppgave som ble mer omfattende enn vi hadde trodd på forhånd: Å laste ned alle kommentarene fra utvalget vårt.

Siden TikTok ikke vil at hvem som helst skal kunne hente ut store mengder data, ble det en tidkrevende prosess. Først brukte vi programmeringsspråket Python og biblioteket Selenium til å lage et skript som åpnet én og én bruker, skrollet ned hele siden (TikTok viser bare de 20-30 første videoene med en gang, resten kommer etter hvert som du skroller) og så trykket seg inn på hver enkelt video og hentet ut alle kommentarene. På brukere som hadde flere tusen videoer kunne det ta flere timer. Ofte stoppet også alt opp etter å ha hentet ut informasjon fra én eller to kontoer. Vi antar det er fordi TikToks sikkerhetssystemer oppdaget unormal oppførsel. Dermed måtte vi ofte starte skriptet på nytt til alle døgnets tider for å få inn all dataen vi skulle ha. I en periode satte vi til og med på alarm på telefonen midt på natta.

Finne barna som oppga egen alder

Alt av data ble lagret i en database. Vi brukte open source-programmet DB Browser for SQLite til å jobbe med databasen. Etter å ha lastet ned 85.000 norske kommentarer, gjorde vi en test-spørring for å se om vi var på rett spor. Vi valgte å søke etter alderen 9-14 år, for å både fange opp de yngste «lovlige» brukerne, men også de «ulovlige». Etterpå gikk vi manuelt gjennom lista for å fjerne kommentarer som handlet om noe annet enn alder (for eksempel «Jeg har trent i 10 år») eller åpenbare tullekommentarer («Jeg er 9 år og tar 4500 kilo i benk. Er det bra?»).

Etter den manuelle gjennomgangen satt vi igjen med 185 brukerkontoer som oppga å være i alderen 9-14 år. Ved hjelp av SQL-metoden «inner join» kunne vi i tillegg finne alle kommentarene de hadde lagt igjen i datasettet vårt. «Inner join» fungerer slik at den kombinerer rader fra to datasett ved hjelp av en matchende verdi. Vi skjønnte at vi var på rett spor, og fortsatte arbeidet med å hente inn kommentarer. For å ha oversikt over utviklingen,

lagde vi et slags dashboard i Jupyter Notebook som presenterte oppdaterte funn fra databasen.

Da vi avsluttet datainnhentingene fra norske kroppsbyggere, satt vi igjen med 230.000 kommentarer skrevet av i overkant av 67.000 brukere. Etter en siste manuell gjennomgang, kunne vi dokumentere følgende:

- 519 kontoer skrev på norsk og hadde oppgitt å være mellom 9 og 14 år.
- Til sammen hadde disse kontoene lagt igjen 4.500 kommentarer.

Dermed følte vi oss trygge på at norske barn fikk se ekstrem trening på TikTok og at de engasjerte seg i innholdet. Neste steg var å forsøke å ta kontakt med dem for å høre mer om deres opplevelser. Men da måtte vi først ta en grundig etikkrunde der vi tok hensyn til barnas sårbarhet, alder og aktivitet på plattformen⁷. Deretter opprettet vi en tydelig merket journalist-konto og fulgte dem på TikTok. Litt over 60 av dem valgte å følge oss tilbake, noe som åpnet for at vi kunne sende dem en melding. Til slutt fikk vi verdifulle erfaringer fra 10 norske 13-14-åringere som vi brukte i anonymisert form i den ferdige saken.

Å sitte igjen med 519 av 67.000 brukere kan ved første øyekast virke lite, men dette er kun de som i en kommentar har oppgitt sin egen alder til 14 år og yngre. De fleste ungdommer blir lært opp til at man skal være forsiktig med å dele personlig informasjon i sosiale medier. Vi søkte også etter de aller yngste brukerne, noen av dem har til og med ikke lov til å være på plattformen, så det kan også ha spilt inn på antallet vi fant. I tillegg forsøkte vi å finne de 519 kontoene igjen i det internasjonale utvalget som telte rundt 500.000 kommentarer, for å se om de norske brukerne også var aktive der. Men vi fant bare en liten håndfull. Siden vi var mest interessert i norske forhold, gikk vi ikke videre med dette sporet.

Tallet TikTok ikke viste

Det er også verdt å nevne at vi gjorde en innsats for å finne ut hvor mange visninger de norske muskelmennene hadde totalt på videoene sine. TikTok oppgir nemlig kun hvor mange følgere og likerklikk en konto har. Selv om dette sier litt om hvor populær en konto er, er antall visninger en bedre målestokk for å vise hvor mye en video har blitt spredt. Heldigvis viser TikTok antall visninger på enkeltvideoer.

Vi gjenbrakte dermed deler av koden som hentet inn kommentarer, og justerte den slik at den hentet inn visningstallet på de 26.493 videoene som de 50 norske kroppsbyggerne på lista vår hadde lagt ut. Det ble en liten opprydningsjobb før vi kunne summere tallene, siden TikTok ikke opererte med hele tall når det var mer enn 9999 visninger. For eksempel ble 17300 visninger presentert som «17.3K» og 3500000 ble til «3.5M». Men ved hjelp av «update»-funksjonen i SQL kunne vi søke etter bokstavverdiene («K» og «M») og erstatte disse med tall. Til slutt kunne vi bruke «sum»-funksjonen til å summere opp tallene og slå fast at kroppsbyggerne hadde fått nærmere 900 millioner visninger. Det er et tall som viser hvilken enorm spredning denne typen ekstrem trening har på TikTok.

⁷ Se vår metoderapport fra Skup 2022 for en mer inngående drøfting av etikken rundt å kontakte barn.

Algoritme-eksperimentet, del 1

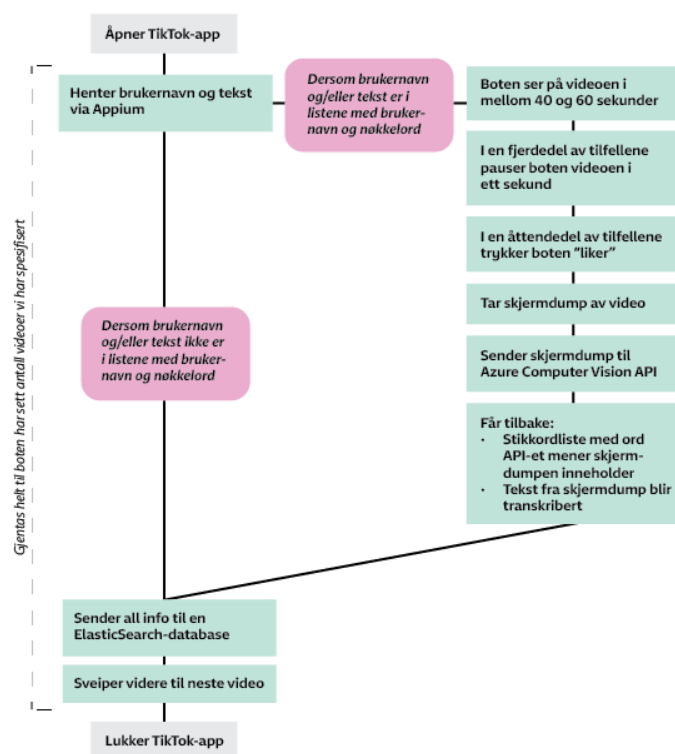
På dette tidspunktet hadde vi grunnlag for å si at muskelvideoer på TikTok blir sett av mange, og at svært unge gutter er blant dem som engasjerer seg. Dermed kunne vi starte algoritme-eksperimentet med TikTok-roboten vår.

For å få til et mest mulig naturlig brukermønster når roboten skulle skrolle, laget vi en tidsplan som skulle passe omtrent til en 13-åring. Det vil si at vi skulle starte roboten morgen, ettermiddag og kveld – ikke i skoletiden eller etter klokken 22. I tillegg til at vi sendte metadata fra hver video til en database, tok vi videoopptak av at roboten sveipet gjennom TikTok. Da brukte vi et kamera på stativ som filmet telefonen. Vi kunne ha løst dette ved å bruke telefonens innebygde skjermopptaksfunksjon. Årsaken til at vi ikke gjorde det, var at vi i størst mulig grad ville unngå å gi TikTok signaler om at vi var en «uvanlig» bruker.

En mandag morgen satte vi den første boten i gang. Den så på videoer i fire forskjellige tidsrom i løpet av ettermiddagen og kvelden, til sammen 498 videoer den første dagen. Boten fortsatte å se på videoer de to neste dagene, men løsningen vår fungerte ikke så godt som forventet. Problemet var at det tok for lang tid for boten å bestemme seg for om den var interessert i en video eller ikke. Dermed så den for lenge på videoer den ikke skulle være interessert i, og ga feil signaler til TikToks algoritmer. For å kunne utføre eksperimentene vi ønsket, måtte vi rett og slett «slanke» roboten.

Det som tok lengst tid, var å ta skjermdump, sende denne til bildegjenkjenningstjenesten og behandle dataene vi fikk tilbake. Samtidig var det nettopp denne løsningen som gjorde roboten vår god til å skjønne hva en video inneholdt selv om det ikke stod noe om det i teksten eller i en emneknagg. Vi så oss likevel nødt til å tenke annerledes. Vi strippet derfor løsningen ned slik at boten utelukkende brukte tekst, emneknagger og brukernavn til å avgjøre om den skulle se lengre på videoen. Den nye versjonen av boten var vesentlig raskere enn den gamle versjonen. Den brukte stort sett fra under ett sekund til to sekunder før den sveipet videre fra videoer den ikke var interessert i, noe som bedre samsvarer med en menneskelig adferd. Se illustrasjon til høyre.

Ett problem gjenstod likevel: Som nevnt tidligere tok det noen ganger veldig lang tid å hente ut brukernavn og/eller tekst via Appium. I de aller fleste tilfellene tok dette bare noen millisekunder, men noen ganger kunne det ta opp mot et halvt minutt. Vi løste dette ved å legge inn en «timeout»-funksjon. Dersom det



tok mer enn 4 sekunder å hente ut brukernavn eller tekst, ville boten vår sveipe til neste video.

Egentlig var det et lykketreff at roboten ikke fungerte med en gang. For mens vi jobbet med å justere den, kikket vi også over Excel-arkene våre med mulige kilder. Et av dem inneholdt norske ungdommer i alderen 15-18 år som hadde TikTok-kontoer som i hovedsak handlet om å bygge muskler. Vi tok kontakt med noen for å lære mer om å være ung på muskel-TikTok. En av dem var 17 år gamle Samuel Vedeler fra Nøtterøy. Han skulle snu opp ned på historiefortellingen vår.

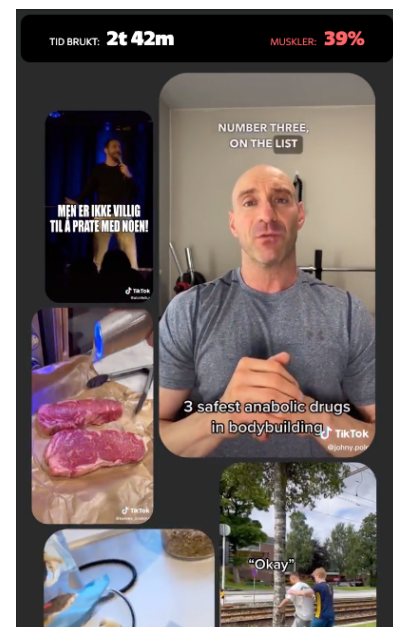
Fra data til historie

En ting er å samle mye data. Noe helt annet er å bruke det til å fortelle en historie som engasjerer og berører. For å få til det siste, hadde vi snakket om historiefortellingen helt fra start. Vi hadde tidlig sett for oss at utviklingen til TikTok-roboten skulle være den bærende historien. Planen var å sveip for sveip følge den 13 år gamle brukeren og dokumentere hva han fikk servert. For å menneskeliggjøre ham hadde vi tenkt å bruke kunstig intelligens til å lage et virkelighetsnært profilbilde, slik vi gjorde med robotene i prosjektet om Russland og Ukraina.

Men så kom Samuel inn i bildet. I en av videoene han hadde delt på sin åpne TikTok-profil fortalte han om hvordan han som 13-åring hadde vært misfornøyd med kroppen sin og søkt etter «sixpack» i sosiale medier for å bli mer muskuløs. Det hadde ledet ham inn i en stadig sykere tilværelse der alt handlet om mat, trening og kontroll på egen kropp. Vi fikk en idé om å døpe roboten «Samuel» og gi den det samme utgangspunktet som virkelighetens Samuel hadde som 13-åring. På den måten kunne vi kombinere menneske og robot i en kronologisk fortelling. Vi presenterte ideen for Samuel, som heldigvis synes det var en kul måte å gjøre det på.

I tillegg valgte vi å innføre en dogme-regel, slik som danske filmregissører på 1990-tallet: «Alle kildene vi bruker i saken skal dukke opp i feeden til TikTok-roboten eller i databasen over 9-13-åringer». Vi opplevde at denne selvpålagte begrensningen hjalp oss å konsentrere oss om kjernen i fortellingen og ikke falle for fristelsen til å hente inn ekspertkommentarer fra øst og vest.

For å gi leseren en følelse av å være med på reisen, la vi inn en «sticky bar» i den ferdige artikkelen – et fast element i toppen av skjermen som viste hvor lenge roboten «Samuel» hadde sveipet og hvor stor prosent av feeden hans som var muskler. Vi la også inn tre videobolker for å vise denne utviklingen.



Algoritme-eksperimentet, del 2

Før vi lot den siste versjonen av roboten vår sveipe gjennom TikTok, nullstilte vi telefonen vi skulle bruke. Dette gjorde vi for å unngå at tidligere aktivitet på telefonen skulle påvirke hva TikTok serverte oss. Den nye brukerens alder ble satt til 13 år. Først sveipet vi gjennom noen

få videoer manuelt, siden det dukker opp en del spørsmål man må svare på rett etter at man har opprettet en ny TikTok-konto. På spørsmål om hvilke språk brukeren vår forsto, svarte vi norsk og engelsk. Deretter søkte vi på ordet «sixpack», slik virkelighetens Samuel hadde gjort. Vi sveipet gjennom 13 videoer som dukket opp etter søket og trykket «liker» på fire av dem. Så lukket vi appen, og lot roboten ta over.

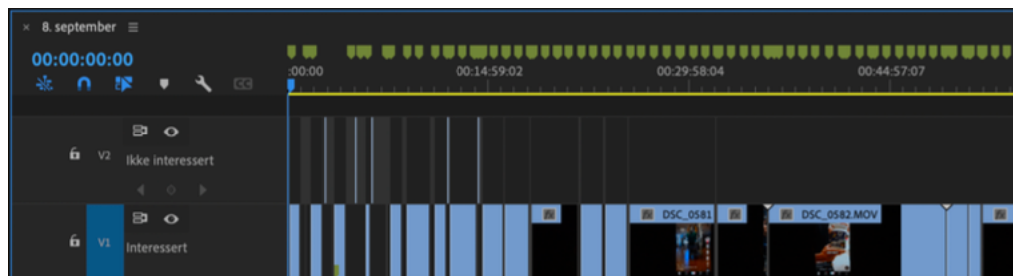
Den neste dagen tok det ikke lang tid før robotens «For You»-feed begynte å fylles med muskelvideoer. Totalt sveipet roboten gjennom 406 videoer denne dagen. Den var interessert i 40,8 prosent av videoene. Fra og med dag tre var et stort flertall av videoer i robotens feed muskelvideoer. Vi hadde bestemt oss for at TikTok-roboten ikke skulle holde på i mer enn ti dager. På dag 9 oppdaget vi at en av videoene til den nå 17 år gamle Samuel hadde dukket opp i feeden. Da skjønnte vi at vi hadde både begynnelse og slutt på historien.

Analyse

Nå var det på tide å sjekke om vår opprinnelige hypotese stemte: Hadde «Samuel» på kort tid fått servert store mengder ensidig innhold?

Ny bruk av videoverktøyet

For å legge til rette for analysen, importerte vi videoene til redigeringsverktøyet Adobe Premiere og lagde én sekvens for hver dag. Deretter brukte vi markører for å vise starten på hver video. Videoer som ikke handlet om muskler eller kosthold ble klippet ut og flyttet over på et annet videospor. På skjermdumpen på neste side kan man se de gule markørene som representerer hvert videoklipp øverst. Videoene som ikke handler om muskler, er lagt over i et annet videospor. Utviklingen er tydelig:



Vi summerte opp antall muskelvideoer og antall andre videoer for hver dag. Så regnet vi ut den prosentvise fordelingen av muskelvideoer i forhold til videoer totalt per dag. Etter å ha gjort det, kunne vi se at vi allerede på dag 3 var oppe i mer enn 90 prosent muskelvideoer.

En svakhet ved metoden vår, var imidlertid at vi ikke sikret at vi fikk videoopptak av alt. Noen dager hadde vi bare delvis videoopptak fra. Årsaken var at vi opprinnelige kun hadde tenkt å bruke det til stikkprøver. Sett i ettertid burde videoopptak vært høyere prioritert, siden «slanking» av den siste versjonen av roboten vår gjorde at den hentet inn mindre informasjon om videoen. For å finne prosentandel for tidspunkter der vi ikke hadde video, gikk vi derfor gjennom databasen manuelt. Vi så gjennom alle dataene fra videoer som roboten vår ikke så lenge på, og talte hvor mange av dem som var muskelvideoer. For mange av videoene var det enkelt å tolke at det dreide seg om muskelvideoer. Andre søkte vi opp på TikTok for å sjekke. For alle dagene bortsett fra én kunne vi se at minst 80 prosent av videoene var muskelvideoer

Gjentakelse av eksperimentene

Etter at vi hadde gjennomført den første testen, ønsket vi å sjekke at det ikke var et engangstilfelle. I den første testen søkte vi på «sixpack». Etter søket likte vi noen av videoene som dukket opp. Ville vi ende opp på samme sted dersom vi droppet dette søket?

Vi gjenopprettet telefonen til fabrikkinnstillinger, og lagde en ny TikTok-konto. Deretter satte vi roboten i gang med de samme interessene som den første roboten, men uten å søke etter «sixpack». Etter 245 videoer fant den en video den var interessert i. Som i det forrige forsøket var det først nøkkelordet «protein» som slo inn. Deretter dukket det opp mer og mer treningsorienterte matvideoer. Mot slutten av dagen ble det stadig flere muskelvideoer i feeden. Dagen etter var et flertall av videoene muskelvideoer. Vi avsluttet forsøket med denne roboten da vi så at resultatet i stor grad var likt det vi endte opp med i det første forsøket. Til slutt gjorde vi samme prosedyre en gang der vi også kuttet ut nøkkelordet «protein». Resultatet var i stor grad det samme.

Mulige feilkilder

- Ved å bruke en robot til å sveipe gjennom TikTok, bruker vi appen på en måte den ikke er ment for. Det kan ha betydning for utfallet.
- Til tross for at vi har gjort roboten så menneskelig som mulig, vil en robot aldri fullt ut ha en menneskelig oppførsel. Vi må derfor holde muligheten åpen for at roboten kan ende opp med en annen feed enn et menneske.
- Det er vanskelig å vite nøyaktig hvilke faktorer algoritmene legger vekt på. Det kan være faktorer som vi ikke kjenner til som er viktige for hvilken feed du får.
- Vi kan ha feilbedømt i hvor stor grad våre tester ble farget av faktorer som IP-adresser, fysisk plassering, mobilenheter og nettverk som vi brukte. Vi brukte et 4G-modem som ikke hadde blitt brukt til annen nettverkstrafikk, siden vi ville unngå å bruke et nettverk som nylig hadde blitt brukt av journalistene som jobbet med saken. Men vi brukte det samme 4G-modemet til de påfølgende testene. Vi har også brukt den samme telefonen. Samtidig er metodene beskrevet for TikTok, og de har ikke gitt motforestillinger til vårt arbeid.

Etikk: Derfor brøt vi TikToks retningslinjer

Vi brøt TikToks retningslinjer da vi opprettet falske brukere og laget programmer som automatisk skrollet gjennom TikTok. Vær Varsom-plakaten paragraf 3.10 presiserer at forutsetningen for å bruke skjult identitet må være at dette er eneste mulighet til å avdekke forhold av vesentlig samfunnsmessig betydning. Vi vurderte at det å bruke en slik falsk robot for å skrolle igjennom TikTok tilsvarer det å gå inn i andre miljøer med skjult identitet. Det at ting foregår på digitale flater og i digitale rom løsriver ikke pressen fra våre etiske retningslinjer, og vi var svært bevisste på at dette måtte stå seg på lik linje som om vi fysisk hadde gått inn i et fysisk rom med skjult identitet.

I stedet for å bruke en robot kunne vi ha bedt mennesker om å sveipe gjennom TikTok mens de var ekstra interessert i et bestemt tema. Ulempen med en slik løsning er at det er vanskelig å hente inn store mengder data. Det er også vanskelig å kontrollere nøyaktig hvordan testen gjennomføres. Ved bruk av en robot kan vi sveipe så lenge vi vil, vi kan gjøre små justeringer og vite nøyaktig hva vi har gjort. Dette var avgjørende for å kunne vise fram det systematiske i TikToks algoritmer.

Det er også et moment i vår vurdering at vår bruk av falsk identitet ikke påfører noen større belastning enn om vi hadde laget nye brukerkontoer for mennesker som skulle utføre lignende tester.

Nye funn og konsekvenser

Med artikkelen «TikToks muskelkraft» kunne NRK:

- For første gang vise hvordan og hvor raskt en norsk 13 år gammel gutt blir dratt inn i en verden av ekstrem trening på TikTok.
- Dokumentere at flere hundre norske barn i alderen 9-14 år er aktive i kommentarfeltene til voksne kroppsbyggere på TikTok.
- Dokumentere at TikTok ikke hadde etterfulgt sine egne løfter om å unngå ensidig innhold, slik de tidligere hadde lovet å gjøre.

Dette førte til at:

- Kulturminister Anette Trettebergstuen tok funnene våre opp med TikTok rett etter publisering. TikTok lovet å opprette et kontaktpunkt i Norge for bedre dialog med norske myndigheter.
- Kulturministeren varslet et fellesnordisk regelverk for å regulere TikTok og andre store teknologiselskaper for å unngå det hun kalte «skadelig innhold».
- Trettebergstuen lovet også at Norge aktivt skal jobbe med EU for å begrense teknologigigantens påvirkning av norske barn, og tok til orde for et regelverk som krever fysisk tilstedeværelse fra TikTok i land hvor appen er i stor bruk.
- TikTok lovet å «se nærmere på utfordringene NRK peker på». Etter vår publisering sendte TikTok ut en pressemelding om at de innfører en funksjon som viser grunnen til at algoritmene anbefaler en video til deg. Senere har selskapet også lagt til en mulighet for europeiske brukere der du kan få en algoritme-fri For You-feed⁸ og kuttet ut målrettet reklame mot barn.
- Fagutvalget for influencemarkedsføring (FIM) sier de skal endre kjønnsbalansen i utvalget etter NRK-sak, og vurderer å ta opp saker på eget initiativ for å fange opp flere mannlige påvirkere som promoterer ekstreme treningsprodukter.

⁸ <https://newsroom.tiktok.com/en-eu/fulfilling-commitments-dsa-update>

Vedlegg

Vedlegg 1: Oversikt over publisering og oppfølgere

03.12.2022: «TikToks muskelkraft» (nrk.no) <https://www.nrk.no/1.16183933>

05.12.2022: «Kulturministeren rystet av NRK-sak: – Dette kan ødelegge liv» (nrk.no) <https://www.nrk.no/1.16206762>

05.12.2022: «Kroppspress på TikTok» (NRK Dagsrevyen) <https://tv.nrk.no/se?v=NNFA19120522&t=96s>

07.12.2022: «Ekspertene svarer på barns spørsmål om trening og helse» (nrk.no) <https://www.nrk.no/1.16173989>

09.12.2022: «TikTok om NRK-avsløring: – Det er et delt ansvar» (nrk.no) <https://www.nrk.no/1.16201787>

09.12.2022: «Unge får muskelreklame: – Jeg prøver å scrolle det bort, men får det opp ukentlig» (nrk.no og NRK Nyhetsmorgen) <https://www.nrk.no/1.16210588>
<https://radio.nrk.no/serie/nyhetsmorgen/sesong/202212/NPUB32024522#t=53m48s>

13.12.2023: «TikTok – gøy eller farlig?» (NRK Radio) <https://radio.nrk.no/serie/arena/MKAK27017422>

05.01.2023: «Reagerer på forskjellsbehandling av kvinnelige og mannlige influensere» (nrk.no) <https://www.nrk.no/1.16222997>

10.01.2023: «Kroppspress på TikTok» (NRK Super) <https://tv.nrk.no/se?v=MSUM02011023&t=116s>

Vedlegg 2: Kode-eksempler på GitHub

Vi har lagt ut et utvalg koder, spørringer og annen dokumentasjon av metode her: <https://github.com/cbjoerke/muskelkraft>