Alternative Solutions to Deepfake Detection

I. Faking IDs for KYC and Dating App Verification (example)

Alternative Solution:

- Many online ID verification tools only check for valid watermarks, correct name, and correct address, meaning a bad actor can impersonate someone with only basic identifying information.
- By **collecting a "digital fingerprint" of a user during ID verification** [see https://amiunique.org], we could prevent one device spoofing as many users.[1]

UX:

- "Always on" logging for spoofing behavior patterns, with alerts including the spoofer, a list of accounts they've created, and the frequency of account creation—the latter determining the alert's urgency (push notification vs. appearing in daily report)
- Provide a simple explanation of digital fingerprinting inside the alert (eg. expandable dropdown)

II. Faking Verification With AI-Generated Images/Video (example)

Alternative Solution:

- AI-generated images often exhibit greater variance image-to-image in terms of *features* than a human does (changing proportions, inexact eye color, etc).
- Dating and Social Platforms both involve multiple images, so you can **plug all images into existing facial recognition algorithms to discern image similarity**: are they the same face?
- Another approach would be iterating on current Distilling and Dispelling Autoencoders—which can align "features" detected by a model to real human features—to see if these features are consistent across all a user's images.[2]

UX:

- There is little/no revenue incentive for social media companies like Reddit & Snapchat to spend on deepfake detection, but Dating Apps might be more interested

---

[1] One parameter that's invaluable for this is battery level - many similar parameters with the exact same battery level signal that a user is operating from multiple tabs: many accounts, one battery level. If a bad actor is using dozens of devices, they all likely have the same configuration, another giveaway.

[2] Both approaches could work better than, for example, Tinder's ID + Photo Verification because they'd be added to your existing "Model of Models" to further improve performance.

- For Dating Apps, giving a user a “hint” that a profile might be a deepfake can be enough to prevent harm
- False positives (marking a real user as AI) can be incredibly harmful. Given that we won’t have control over the way a company utilizes an API endpoint, the endpoint could respond with a likelihood and give the likelihood context (eg. color depending on risk).

III. Impersonating Public Figures (example)

Alternative Solution:

- **Build a public figure identification dataset** + model [or use existing open source models]
- Companies using this could save on cost by only triggering when a public figure is mentioned in comments, or only when vetting ads

UX:

- Built an API endpoint as an “extra-step” for an org’s automated content review system
- Return a likelihood score: high scores would be automatically removed and moderately-high scores would be moved to human review.

(My resume is on the next page!)

Benjamin Guzovsky

781-330-6849 ● guzovsky@princeton.edu ● github.com/benguz ● benguzovsky.com

Education

Princeton University *May 2024*

AB, History, Computer Science minor. 3.8 GPA.

Relevant Coursework: Algorithms and Data Structures, Programming Systems, Machine Learning, Computational Biology.

Projects

- **Open Source Prompt Engineering Splitscreen** (Node.js) with 80 daily active users running side-by-side prompt engineering across any number of models and prompts, parallelizing requests for efficiency (December 2023).
- **Feature-Rich Website Builder** (Flask, PostgreSQL, Google Cloud, JS) enabling users to deploy websites straight from their Google Docs, auto-syncing changes in real time (April – May 2023).
- **Analyzed an emerging market opportunity for KVH**, a global leader in mobile connectivity and inertial navigation, sharing slides with the company's 500 employees (February 2022).
- **AI Research at Columbia University** (Python, JS, AWS Lambda) culminated in building a context-dependent thesaurus for creative writers, giving ML suggestions that improve cadence and creativity (May – August 2021).

Experience

Doorstop Education

Founder *May 2022 – Present*

- Independently conducted ethnographic fieldwork at high schools in 15 states, listened to the unmet needs of over 100 students one-on-one, spending 250+ hours observing classes to discover opportunities for self-advocacy.
- Incorporating an education nonprofit with a mission to inspire student agency nationwide, designing and collaboratively developing two highly technical user-facing web apps in the process:
- **Doorstops** (Python, JS), 100+ interactive digital experiences that match high school students with personalized self-advocacy strategies with an optimized recommender system running locally on a user's device, keeping costs negligible and protecting data privacy.
- Perfected and ADHD-sensitive touch/swipe interactions across all devices through 90 student beta tests.
- **OpenSchool** (Django, JS, Beautiful Soup, MySQL), a school search platform that empowers NYC's 80,000 8th graders to make informed choices between 700 high schools, synthesizing previously opaque, discrete datasets into actionable, transparent pipelines that are easily digestible for students.
- Evaluating school climate by identifying key indicators (teacher retention, student/counselor ratio) and parsing relevant datasets for proxy metrics (e.g. gauging teacher retention by comparing NYC teacher pay scales to school finance data).
- Organically grew social media on $0 in ad spend to 500k+ views across platforms.

Perrfy

Product Manager *May – August 2021*

- Contracted for an early-stage startup in the web development performance optimization market, roadmapping a pivot from B2C to B2B based on data-driven insights and UX interviews that boosted conversion rates by 40%.
- Mastered complex web development concepts in order to: rewrite 100 pages of Perrfy website content from technical to user-friendly, create 50 new website pages, and write 40 pages of blog content.
- Managed 12 independent contractors, synthesizing programming, design, content, and marketing needs to prioritize task assignments through close collaboration across stakeholder groups.

Skills

Programming: Java, C, Python (TensorFlow, Beautiful Soup), SQL, React, JavaScript, Node.js, CSS, HTML, AWS Lambda.
Technical Skills: Figma, Illustrator, & Photoshop; Scrum & Agile Development; Office Suite & Excel.
Languages: Fluent in Russian with experience in translation & intermediate in Mandarin.