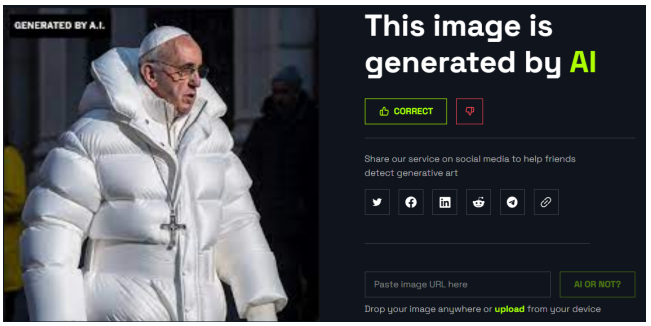
Fool AI Detectors

Mit Text-zu-Bild-Generatoren erzeugte Bilder sehen immer realistischer aus. Große Sprachmodelle schreiben immer komplexere Texte und beantworten Fragen, sodass sie kaum von einem Menschen zu unterscheiden sind. Dadurch entsteht nicht nur unheimlich viel Potenzial, kreative Inhalte zu produzieren und Arbeitsabläufe zu automatisieren, sondern auch die Gefahr von Fake-News, die mit minimalem Aufwand erzeugt werden können. Um dieser Gefahr zu begegnen, hat sich das fiktive Startup "No More Fake News" (NMFK) auf die Fahne geschrieben, ein neues Modell zu entwickeln, das zuverlässig und sicher von KI erstellte Inhalte erkennen und so Fake-News, die durch generative Modelle erstellt worden sind, unterbinden soll.

Das Startup hat Euch beauftragt, zunächst eine technische Marktanalyse durchzuführen. Ihr sollt bereits am Markt bestehende KI-Modelle untersuchen, die für die Text- und Bilderzeugung genutzt werden können. Darüber hinaus sollt ihr evaluieren, welche Modelle für die Detektion von KI generierten Inhalten genutzt werden können und deren Performance beurteilen. Im zweiten Schritt möchte das Startup zeigen, dass der Markt noch nicht saturiert und die Entwicklung eines völlig neuen Tools zur Erkennung von KI-Inhalten unerlässlich ist. Die Aufgabe besteht darin, ein Programm zu entwickeln, das durch möglichst minimale Änderungen von KI-generierten Texten oder Bildern bestehende KI-Detektoren mit wenig Aufwand zuverlässig austrickst. Das NMFK Team erhofft sich, das Programm während des Trainings eines neuen Modells für die Augmentierung ihrer Daten zu verwenden, um ihr eigenes Produkt robust gegen solche Modifikationen zu trainieren. Aus diesem Grund muss die von Euch entwickelte Software sowohl schnell als auch zuverlässig auf möglichst beliebig großen Bildern und beliebig langen Texten funktionieren.



Die diesjährige Aufgabe richtet sich an den Geist der Zeit, bei dem sich alles um KI und generative Modelle dreht. Diese drängen immer mehr in unser Leben ein und werden in Zukunft ein fester Bestandteil vieler Anwendungen im privaten Bereich, aber auch im öffentlichen Leben und der Wirtschaft. Schon heute ist die Verwendung von Assistenzsystemen, wie der Autokorrektur und automatischer Bildbearbeitung, nicht aus dem modernen Alltag wegzudenken. Zukünftig werden

generative Modelle nur noch mächtiger und präsenter in unser Leben treten. Als Teilnehmer am informatiCup müsst Ihr bei dieser Aufgabe beweisen, dass Ihr KI versteht, große Netzwerke analysieren und kreative Lösungen entwickeln könnt, um solche Modelle zu überlisten. Damit verbindet die diesjährige Aufgabe nicht nur Aspekte aus der Wirtschaft, wie eine Marktanalyse und bindet diese in das aktuelle Forschungsthema der generativen Modelle ein, sondern soll zusätzlich anregen, sich mit gesellschaftskritischen Folgen der Forschung in der Informatik, wie der KI-gestützten Erstellung von Falschinformationen, auseinanderzusetzen,.

Aufgabenbeschreibung

Die Aufgabe des diesjährigen informatiCup besteht aus zwei Teilen. Im ersten Teil sollt Ihr eine technische Marktanalyse durchführen, bei der Ihr bestehende Modelle untersucht, die für die Text- und Bilderzeugung genutzt werden können, sowie welche Modelle und Algorithmen für die Detektion von generierten Inhalten genutzt werden können. Im zweiten Teil sollt ihr ein Programm entwickeln, dass durch möglichst minimale Änderungen von KI-generierten Texten und Bildern bestehende KI-Detektoren austricksen kann.

Technische Marktanalyse. In der technischen Analyse sollen Generatoren für die *Erzeugung von Inhalten durch KI* und Programme zur *Detektion von KI erstellten Inhalten* recherchiert werden. Die Evaluation der bestehenden Landschaft für generative Modelle soll einen Überblick über eine sich sehr schnell und dynamisch entwickelnde Domäne verschaffen. Es sollte eine möglichst breite Palette an Software betrachtet werden, die heute schon produktiv eingesetzt werden kann, sowie ein Ausblick auf die neueste Forschung in dieser Disziplin gegeben werden. Neben der Analyse der generativen Modelle möchten wir auch, dass Ihr genauso Programme zur Detektion von KI-erstellten Inhalten untersucht und uns genauso einen Überblick über bestehende Produkte sowie die aktuelle Forschung gebt. Zudem möchten wir, dass Ihr die Qualität der erzeugten Texte und Bilder durch die generativen Modelle und die Qualität der Detektion der regressiven Modelle evaluiert. Ihr könnt dabei quantitativ, qualitativ oder auch hybrid vorgehen, um die Performance der Software zu bewerten.

Augmentierung von Daten. Im zweiten Schritt möchten wir, dass ihr eine *Software für die Augmentierung von KI-erstellten Inhalten* entwickelt. Hierbei sollt Ihr zunächst Euren theoretischen Ansatz beschreiben, auf dem die Software basiert. Danach sollt Ihr ein Programm entwickeln, das den Input möglichst minimal ändert, sodass ein KI-Detektor den Output möglichst immer als einen von Menschen erstellten Text oder Bild erkennt. Abschließend möchten wir, dass Ihr auch in diesem Teil die Qualität der Software ausführlich evaluiert.

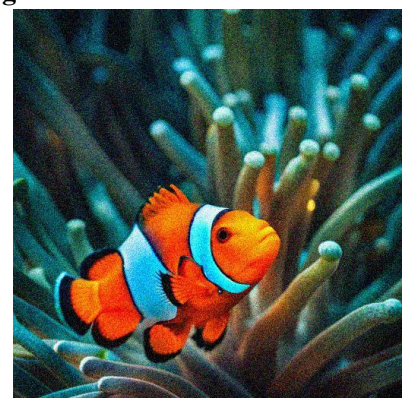
Beispiel

Hier findet Ihr ein Beispiel für eine Augmentierung eines KI-generierten Bildes mit einem einfachen Gaußschen Rauschen, das den Score eines Detektors von 90.7% auf 30.8% reduziert hat.

KI-generiert

Score: 90.7% KI-generiert

Augmentiert mit Gaußschen Rauschen



Score: 30.8% KI-generiert

Technische Anforderungen

Eure Software soll durch einen Kommandozeilenaufruf, gefolgt von zwei Argumenten, gestartet werden. Das erste Argument enthält einen Pfad zu einer Datei, die als Input dient. Das zweite Argument enthält auch einen Pfad mit einem Dateinamen, unter dem der Output geschrieben werden soll. Der Input kann entweder eine Textdatei (.txt) oder eine Bilddatei (.jpg, .png) sein, das Programm soll beide Eingabeformate beherrschen können. Euer Programm soll eigenständig erkennen, ob es sich bei dem Input um einen Text oder ein Bild handelt und den Input entsprechend verarbeiten. Bei Texten soll die Länge des Textes (Anzahl der Wörter und Zeichen) nach Möglichkeit gleich bleiben. Bei Bildern darf sich die Auflösung des Bildes nicht ändern. Texteingaben bestehen immer aus mindestens 3, aber maximal 1024 Wörtern, die Auflösung der Bilder ist minimal 64x64 Pixel und maximal 1920x1080 Pixel. Euer Programm muss eine Ausgabe innerhalb von 5 Sekunden auf einer üblichen Consumer-Hardware (z.B. Intel i5 10th Gen, RTX 2070S, 16GB RAM) generieren. Falls Ihr ein KI-Modell nutzt, darf die gesamte Trainingszeit Eures Modells maximal 24 Stunden betragen.

Erweiterungen

Der informatiCup möchte seine Teilnehmer in der Kreativität ihrer Lösungen nicht einschränken und auch für Engagement über die geforderte Aufgabe hinaus belohnen. Sinnvolle Erweiterungen der Software werden positiv in die Bewertung mit einfließen. Hierbei kann es sich um ein schönes Benutzerinterface, ein Browser-Plugin oder sogar einen eigenen KI-Detektor handeln. Insbesondere bietet es sich an, Eure Software über eine API-Schnittstelle mit öffentlichen KI-Detektoren zu verbinden und so eine automatisierte Evaluation der Augmentierung durchzuführen. Die Jury behält sich vor, die Einreichungen als Ganzes zu bewerten.

Anmeldung und Einreichung

Die Anmeldung zum Wettbewerb findet über Teammates[1] statt. Auf diesem Online-Portal könnt Ihr Euch als Teilnehmer registrieren, mit anderen Teilnehmern ein Team bilden und dieses Team schließlich zum Wettbewerb anmelden. Ihr könnt in Teammates auch Eure Einreichung vornehmen. Die FAQs zum laufenden Wettbewerb, Beispieleingaben sowie eine kurze Beispielimplementierung findet Ihr in dem GitHub-Repository[2] des diesjährigen Wettbewerbs.

Bewertung

Die Bewertung der **technischen Marktanalyse** erfolgt durch das Jury-Komitee anhand der Ausarbeitung unabhängig von der eingereichten Software für die Augmentierung der Daten. Da die Bereiche der generativen Modelle und Detektoren starke Wechselwirkungen aufweisen, erwarten wir nicht nur eine unabhängige Betrachtung, sondern auch eine Analyse der Korrelation beider Domänen. Zusätzlich entwickelte Software, die für die Marktanalyse notwendig war, kann in die Bewertung mit einfließen.

Die Auswertung der Software zur **Augmentierung der Daten** ergibt sich aus

$$S_{Final} = \Delta_{Gen} \sum_{i=0}^{n} \alpha_{i,\,Gen} S_{i,\,Gen} + (1 - \Delta_{Hum}) \sum_{i=0}^{n} \beta_{i,\,Hum} S_{i,\,Hum} .$$

Wir werden die Auswertung auf einer vorher unbestimmten Anzahl n von KI-Detektoren durchführen. Somit sind die Gewichte α und β den Teilnehmern auch unbekannt. Der Score für jeden Detektor und jeden Datensatz ergibt sich aus

$$S_{i,\,Gen} = \frac{1}{|X|} \sum_{x \in X_{Gen}} D_i(x) - D_i(A(x)) \text{ und } S_{i,\,Hum} = \frac{1}{|X|} \sum_{x \in X_{Hum}} min(0,\, D_i(x) - D_i(A(x))) ,$$

wobei $S_{i,\,Gen}$ die Qualität der Einreichung auf dem KI-generierten Datensatz X_{Gen} und $S_{i,\,Hum}$ auf einen von Menschen erstellten Datensatz X_{Hum} darstellt. Der KI-Detektor und das Programm zur Augmentation der Daten sind jeweils durch D und A gekennzeichnet. Wobei $D(x) = 1$ bedeutet, dass die Eingabe x maximal KI-generiert ist. Zudem wird durch eine unbekannte Ähnlichkeitsmetrik $Sim \rightarrow [0, 1]$ das Original mit der augmentierten Versionen verglichen und die semantische, syntaktische, optische und strukturelle Ähnlichkeit Δ der Daten bewertet.

$$\Delta = \frac{1}{|X|} \sum_{x \in X} Sim(x,\, A(x)) .$$

[1] https://teams.informaticup.de/
[2] https://github.com/informatiCup/informatiCup2024

Diese Metrik besteht zum Teil aus etablierten Algorithmen, aber auch zum Teil aus Eindrücken der Jury, die durch Betrachtung repräsentativer Beispieleingaben zusammengesetzt werden.

Die Bewertung der Software erfolgt neben dem reinen Score auch anhand einer **wissenschaftlichen Betrachtung der Methodik**. Bitte achtet darauf, dass dieser Aspekt in Eurer Ausarbeitung thematisiert wird. Die **Softwarearchitektur und -qualität** sind auch maßgebend.

Die **abschließende Bewertung** ergibt sich aus den Einzelwertungen der technischen Marktanalyse, des theoretischen Ansatzes, sowie der Qualität der Software, die die Jury für jede Abgabe vergeben wird. Die Analyse anderer verwandter Aspekte kann die Bewertung weiter positiv beeinflussen, genauso wie diverse Erweiterungen der Software. Wenn Ihr Eure Einreichung im Finale präsentieren dürft, fließt zudem die Präsentation in die finale Bewertung mit ein.

Checkliste der Bewertungskriterien

Bitte nutzt diese Checkliste, um sicherzugehen, dass Eure Einreichung vollständig ist.

Ausarbeitung

Die technische Marktanalyse, sowie der theoretische Ansatz müssen in einer Ausarbeitung (eine zusammenhängende pdf-Datei) dargestellt werden, die zusammen mit der Implementierung eingereicht wird. Bewertet werden sowohl der Inhalt als auch die Formalien.

Die Ausarbeitung soll die Ergebnisse der Analyse, der Evaluierung, sowie die Theorie hinter der verwendeten Methodik beschreiben.

- ❏ **Analyse:** Wie wurde die bestehende Produktlandschaft betrachtet?
- ❏ **Hintergrund:** Der theoretische Hintergrund. Welche theoretischen Ansätze wurden verwendet? Warum wurden diese verwendet?
- ❏ **Auswertung:** Wie ist die Qualität? Nach welchen (wissenschaftlichen) Kriterien wurde bewertet? Warum wurden diese Kriterien verwendet?
- ❏ **Diskussion:** Was für Probleme sind offen? Wie lässt sich die Software praktisch einsetzen?
- ❏ **Quellen:** Wurden (wissenschaftliche) Quellen richtig und angemessen verwendet?

Eine gute Form ist entscheidend für die Lesbarkeit einer Ausarbeitung. Beachtet deshalb neben dem reinen Inhalt der Ausarbeitung auch einige Formalien:

- ❏ **Rechtschreibung:** Rechtschreibung und Grammatik sind korrekt.
- ❏ **Lesefluss:** Es gibt einen Lesefluss in der Ausarbeitung.

- **Layout:** Das Dokument hat ein einheitliches Layout. Dieses kann frei gewählt werden, darf aber nicht den Lesefluss stören.
- **Quellenangaben:** Quellen sind richtig und einheitlich angegeben.

Softwaretechnik

Da eine etablierte Softwarearchitektur nur mit hohem Aufwand zu ändern ist, sollte sie durchdacht und begründet werden. Eine solide Projektorganisation unterstützt eine effiziente Softwareentwicklung und sollte deshalb auch nicht vernachlässigt werden. Gerne dürfen hier weitere Aspekte kurz und präzise beleuchtet werden.

- **Organisation:** Team- und Projektorganisation
- **Architektur:** Beschreibung der Hauptkomponenten und ihrer Beziehungen
- **Testing und Conventions:** Begründetes Konzept, Umsetzung
- **Wartbarkeit:** Mit welchem Aufwand kann das System weiterentwickelt werden?

Präsentation

Im informatiCup-Finale werden die besten Einreichungen vor einer Fachjury präsentiert. Die Bewertung erfolgt anhand dieser Kriterien:

- **Verständlichkeit:** Ist der Vortrag verständlich? Wird der Inhalt in einem angemessenen Tempo erklärt? Wird nötiges Vorwissen geschaffen?
- **Struktur:** Ist der Vortrag logisch strukturiert? Ist ein roter Faden erkennbar?
- **Vortragsstil:** Weckt der Vortragsstil Interesse an der Präsentation? Kann man dem Vortrag leicht und intuitiv folgen? Ist der verwendete Sprachstil adäquat?
- **Foliendesign:** Unterstützen die Folien den Inhalt der Präsentation oder lenken Sie ab? Sind Sie ansprechend gestaltet?
- **Diskussion:** Können Nachfragen beantwortet werden?

Optional

Die Bearbeitung dieser Aspekte ist nicht verpflichtend.

- **CLI und GUI:** Werden eine Kommandozeilenschnittstelle mit erweiterten Konfigurationsmöglichkeiten und/oder eine grafische Benutzeroberfläche eingereicht, ist eine kurze Bau- und Bedienungsanleitung hinzuzufügen (z.B. in Form einer Readme-Datei oder eines gesonderten pdf-Dokuments).