KI-SPRACHMODELLE: IT-SICHERHEIT UND EINSATZ IN DER PROGRAMMIERUNG

Michael Sonntag

Assoz.-Prof. Johannes Kepler Universität Linz, Institute of Networks and Security
Altenbergerstr. 69, 4040 Linz, AT
michael.sonntag@ins.jku.at; https://www.ins.jku.at/

Schlagworte: ***LLM, IT-Sicherheit, Phishing, Programmcode-Generierung***

Abstract: *KI bzw große Sprachmodelle (Large Language Models - LLMs; zB ChatGPT) können sehr gute Formulierungen von Texten gewünschten Inhalts – auch in Fremdsprachen – erzeugen, was das Ökosystem von Phishing-Angriffen verändern kann. Dieser Beitrag untersucht dies von technischer wie auch rechtlicher Seite. Eine weitere Anwendung dieser Sprachmodellen ist die Unterstützung bei der Programmierung: Vom verbesserten Auto-Vervollständigen hin zur automatischen Generierung ganze Methoden oder Test-Klassen. Auch hierbei stellen sich Rechtsfragen, insb ob es sich beim Einsatz solcher Techniken um Urheberrechtsverletzungen handelt, was spezifisch von deren Funktionsweise abhängt.*

1. Einleitung

Sprachmodelle und KI können sich in Zukunft negativ auf die IT-Sicherheit auswirken. Insb Phishing, das Bewegen normaler/berechtigter Benutzer zu für sie bzw ihr Unternehmen nachteiligen Handlungen mittels an sie gesandter Mitteilungen (typ E-Mail, aber auch SMS oder Messenger-Dienste) kann sich hierdurch verändern. Dem steht die Möglichkeit gegenüber, Mails uU auch inhaltlich mittels KI zu analysieren (zB in Form einer Zusammenfassung oder als Schlagworte), um ihre Vertrauenswürdigkeit bzw Gefährlichkeit zu beurteilen. Erste Versuche von Angriffen mit KI-Unterstützung können schon beobachtet werden, und es ist mit einer dynamischen Entwicklung in der Zukunft zu rechnen.

Die Sicherheitsbranche scheint darauf derzeit noch wenig eingestellt zu sein, was jedoch teilweise mit der Natur der zusätzlichen/neuen/veränderten Angriffe zusammenhängt: Diese zielen vielfach auf Menschen ab (der Bereich, in dem die neuen KI-Modelle besonders gut sind), bei denen technische Sicherheitsmaßnahmen naturgemäß nur beschränkt einsetzbar sind. Überweist jemand beispielsweise Geld, weil die Geschäftsführerin dies per E-Mail so angeordnet hat (→ CEO Fraud), so können technische Maßnahmen nur wenig dagegen unternehmen: Die Mail enthält keinen Virus, ist „normal“ geschrieben etc. Und gerade der einzige Punkt, an dem Technik hilfreich sein könnte – der Erkennung der E-Mail durch den Empfänger als gefälscht, da der Text „seltsam“ (schlechtes Deutsch, ungewöhnliche Formatierung...) erscheint – wird durch die KI-Unterstützung bei der Formulierung in Zukunft erschwert werden.

2. Technik & Sicherheit: Unterstützung von Phishing durch KI

An Hand des Beispiels in der Einleitung wurde schon eine große Gefahr von LLMs vorgestellt, die täuschend echte Nachahmung von Menschen. Diese muss nicht im Sinne eines Turing-Tests erfolgen: Ein Mensch muss nicht in einem (ihm als solchen bekannten) Versuch entscheiden, ob es sich bei dem Gegenüber um einen anderen Menschen oder eine Maschine handelt, sondern die KI-Modelle müssen nur bei flüchtigster bis hin zu oberflächlicher Betrachtung durch nichtsahnende Menschen – vielfach ohne Rückkanal bzw nur mit Zeitverzögerung, Einschränkung auf schriftliche und asynchrone Kommunikation etc – kurzfristig überzeugend bzw

zu einmalig durchzuführenden Aktivitäten anregend wirken. Das Ergebnis muss nicht perfekt sein, sondern nur gut genug (gewinnversprechende Erfolgsrate[1]).

Was kann KI bzw LLMs nun spezifisch im Hinblick auf Phishing besser als bisher?

– Es können automatisch Recherchen erfolgen, um den Inhalt der Nachricht an den/die EmpfängerIn anzupassen. Hier besteht jedoch derzeit noch ein gewisses Problem für die Sprachmodelle: Diese sind mit „dem (ganzen) Internet" trainiert - aber nicht dem „aktuellen". Das bedeutet, sie kennen zB die Wikipedia von vor 2 Jahren. Es ist allerdings damit zu rechnen, dass sich dieser zeitliche Abstand reduziert und in Zukunft auch die Miteinbeziehung aktuellster Daten, zB aus sozialen Medien, möglich sein wird. Dann ist mit tagesaktuellen Informationsquellen als Basis der Generierung zu rechnen. Derzeit müssten diese noch händisch (bzw mit anderen KI-Systemen) gesucht und als Teil der Eingabe an die Sprachmodelle weitergereicht werden – was auch schon einen Fortschritt gegenüber dem reinen Serienbrief bzw komplett manueller Erstellung darstellt.
– Die Nachrichten werden individuell erzeugt und sind daher selbst bei Mehrfachverwendung (selbes Unternehmen, selbe Ebene, unterschiedliche angebliche Absender etc) unterschiedlich. Auch bei mehrfacher Erzeugung für dieselbe Person („Wiederholung wegen Dringlichkeit") werden unterschiedliche Mails erzeugt, selbst in den Teilen, welche nicht für die Zielperson individualisiert sind. Eine Erkennung anhand identischer Passagen (=analog zu Viren-Signaturen) ist daher sehr viel schwieriger. Es bleiben Heuristiken, welche zB verdächtige Kombinationen von Wörtern („Überweisung", „dringend", „geheim"…) erkennen. Bei diesen ist jedoch die Falsch-Positiv-Rate sehr viel höher, da auch korrekte und normale Nachrichten diese Kombinationen von Wörtern enthalten werden[2]. Gleichzeitig können LLMs gezielt Synonyme oder alternative Formulierungen verwenden, sofern sie nachtrainiert werden, zB mit bekannten Signaturen von Erkennungsprogrammen.
– Der Aufwand zur Erstellung individualisierter anstatt generischer Texte, die für viel Personen passen müssen, sinkt stark. Zusätzlich ist davon auszugehen, dass „Übersetzungen" – mit oft schlechtem Ergebnis – nicht mehr nötig sind, da LLMs zumindest in wichtigen/verbreiteten Sprachen Texte direkt erzeugen. Dies ermöglichst es weiters, gezielte Angriffe (mit bisher hohem Aufwand) nicht mehr nur auf hochwertige Ziele (=Erwartung hohen Gewinns) durchzuführen, sondern ebenso auf Klein- bzw Kleinstunternehmen oder Einzelpersonen. Dies ist analog zu Ransomware: Ist der Aufwand für den Angriff nur gering genug, lohnen sich auch Ziele bei denen nur kleine Beträge verlangt werden können. Weiters ist es wichtig, dass die „Vorarbeit", bei welcher die Erfolgsrate am niedrigsten ist, automatisiert wird (zB Nachrichten-Generierung), während erst spätere Phasen nach erfolgreichem Erstkontakt (zB Hilfe bei der Zahlungsabwicklung mit Kryptowährungen) Personaleinsatz erfordern.
– Die KI kann zB auch Bilder einbauen, die aus dem Umfeld der Zielperson (oder des angeblichen Absenders) stammen. Diese können recherchiert worden sein, aber es kann sich auch um Deep Fakes handeln. Passende Bilder erhöhen die Überzeugungskraft einer E-Mail. Deep Fakes dürften hier eher seltener hilfreich sein, da für eine Fälschung ohnehin Bilder der gewünschten Person benötigt und damit gefunden werden müssen (à Social Media). Diese in einen anderen Kontext zu stellen kann aber erleichtern, eine bestimmte Aussage zu transportieren (in Zusammenhang mit durch KI generiertem „Hintergrund" kann

[1] Die komplett manuelle Erstellung von Phishing-Mails ("Spear-Phishing") geht von einem gezielten Angriff auf hochwertige Ziele mit hohem Ressourceneinsatz aus: dies ist für ein Massengeschäft untauglich, besitzt aber eine hohe Erfolgsrate. Bloße Serienmails werden im Gegensatz dazu deshalb so oft empfangen, weil die Erfolgsrate äußerst niedrig ist, benötigen aber dafür kaum einen (und gar keinen individuellen) Aufwand. Eine Kombination von höherer Erfolgsrate (durch LLMs) und automatisierter Erstellung und daher breiter Streuung lässt für Angreifer einen deutlich gestiegenen Gewinn erwarten, und damit auch viel häufigere Angriffe.

[2] Weiters existieren neben dem CEO-Fraud noch weitere Varianten von direkten (PIN/TAN eingeben zur Freischaltung einer Überweisung) oder indirekten (Herauslocken von Passwörtern für spätere Aktivitäten) Angriffen. Diese alle eindeutig zu identifizieren und aktuell zu halten dürfte einigen Aufwand bedeuten – doch auch die Aktualisierung von Virensignaturen ist mindestens gleich aufwändig und vermutlich häufiger erforderlich.

dieser frei bestimmt/angepasst werden, zB bei „CEO ist bekanntermaßen im Ausland am Ort X – siehe Bild von ihr in X – und muss eine Überweisung veranlassen“).

– Bei ausreichender Vorlage und entsprechendem Training können Schreibstil und Wortwahl von Personen nachgeahmt werden – was die Überzeugungskraft wiederum verstärkt. Ein potentielles Problem für Angreifer hierbei ist, dass zwar evtl viel Text aus sozialen Medien vorhanden ist, sich dort der Schreibstil einer Person jedoch von zB geschäftlicher Kommunikation stark unterscheiden kann, und bei letzterer Muster für Dritte schwerer zugänglich sind.

Es existieren jedoch auch (noch?) Probleme, dh Schwachpunkt aus Sicher der Angreifer:

– Spätere Mails können nicht/nur mit Zusatzaufwand auf frühere bzw Antworten eingehen: Die derzeitigen Modelle besitzen noch keine „Erinnerung“; diese müsste daher über den Prompt zusätzlich eingebracht werden. Gleiches gilt für Antworten bei Rückfragen oÄ. In Zukunft ist jedoch mit einem (deutlich längeren) „Gedächtnis“ zu rechnen; bei Erzeugung durch die Angreifer mit einem eigenen (nachtrainierten) Modell spielt auch die erforderliche Speicherung früherer Nachrichten keine Rolle. Diese können jedoch bei Entdeckung als Beweismittel dienen, bzw die Recherche nach weiteren (potentiellen) Opfern erleichtern.
– „Watermarking“ zur Markierung von mit KI erstellten Texten ist möglich (im Text, zB durch Wortwahl etc, also nicht durch geheime Elemente im Format, zB einem Textverarbeitungs-Dokument). Dies müsste jedoch vorgeschrieben werden (oder der Betreiber müsste dies freiwillig implementieren) und kann durch Angreifer einfach umgangen werden: Vortrainierte Sprachmodelle (in geringerer Qualität) existieren auch als Open Source-Tools. Für Angreifer ist es jedenfalls interessanter, diese selbst zu betreiben (und durch zusätzliches Trainieren an den Einsatzzweck anzupassen), als auf einen kommerziellen öffentlichen Dienst – mit jederzeitiger Möglichkeit der Entdeckung als missbräuchlich und dem Erfordernis von Zahlungsvorgängen – angewiesen zu sein. Ob sich auch bei derart vortrainierten Modellen Watermarking noch zuverlässig, dh ein Nachtrainieren überstehend und auch nicht gezielt entfernbar, umsetzen lässt, ist derzeit unbekannt. Das Nachtrainieren eigener Modelle setzt zwar spezielle Kenntnisse voraus, doch bei den heute üblichen arbeitsteiligen Angriffen ist dies kein Problem, insb da dies nur einmalig erforderlich ist, sehr viel weniger Ressourcen als das Grund-Training benötigt, und je nach Anwendungszweck auch automatisiert werden könnte („Sie stellen die Texte der/für die Opfer bereit, wir liefern ein passendes Modell, zusammen mit der Versand-Infrastruktur“ etc). Die komplette Selbsterstellung eines eigenen neuen Modells wäre hingegen selbst für viele staatliche Angreifer unmöglich – zuverlässiges Watermarking besäße daher, sofern es durch nachtrainieren nicht zum Verschwinden gebracht werden kann, einen sehr hohen Nutzwert zur Erkennung automatisiert erstellter Texte.
– Erlangt die Strafverfolgung Zugriff auf das Modell ist es uU möglich, die Trainingsdaten wieder (teilweise) zu extrahieren oder zu verifizieren: Dies kann dann als Nachweis für eine Involvierung beim Training dienen: Mit genau diesen (separat vorgefundenen) Dokumenten wurde trainiert. Dies bedeutet gleichzeitig den Nachweis, dass gezielt für illegale Vorgehensweisen trainiert wurde (=Vorsatz). Damit könnten (sofern auffindbar) nicht nur die unmittelbaren Täter, sondern auch vorgelagerte Akteure belangt werden.
– Es muss ein Text durch KI uU nicht als „Generiert von LLM/KI“ erkannt werden, um Phishing-Mails zu erkennen. Es könnte ausreichen festzustellen, dass hierdurch eine Anweisung für eine Überweisung in Verbindung mit Geheimhaltung erfolgen soll – und dass dies möglicherweise problematisch ist. Dann könnte eine Warnung/Ergänzung um „Es wird empfohlen, nachzufragen“/… eingefügt werden. Vor- bzw Nachteil dieses Ansatzes ist es, dass er auf bestimmte (=bekannte) Vorgehensmodelle aufbaut, wie im Beispiel dem CEO-Fraud. Dies ist eine Einschränkung, könnte aber, da zumindest derzeit nicht so viele verschiedene Grundmodelle existieren, keine stark begrenzende Eigenschaft sein. Nachteilig hierbei ist, dass bei KI-erstellten Texten aufgrund ihrer Einmaligkeit die Analyse für jede Nachricht durchgeführt

werden muss: eine einmalige aufwändige Erkennung/Bearbeitung durch einen Provider und anschließende einfache Wiedererkennung (zB über Hashwerte) ist nicht möglich.[3]

3. Rechtliche Folgen des Einsatzes von KI durch Angreifer

Welche rechtlichen Konsequenzen besitzt der Einsatz von zB Textgeneratoren für Angreifer? Bei der aktuell verfügbaren KI kann nicht die Rede davon sein, dass diese autonom als eigene Person handelt, und daher eine Strafbarkeit nur diese beträfe – auch bliebe immer noch die Bestimmung zur Tat offen.

Auch die Qualifikation von Straftaten als „kriminelle Vereinigung" kann durch den Einsatz von KI nicht begründet werden (§ 278 StGB: „mehr als zwei Personen"). Die Bereitstellung von entsprechenden KI-Werkzeugen kann jedoch als Beteiligung zu werten sein: § 278 Abs 2 „Bereitstellung von Information oder Vermögenswerten oder auf andere Weise in dem Wissen beteiligt, dass er dadurch die Vereinigung oder deren strafbare Handlungen fördert." Dies bedeutet, dass der Missbrauch von öffentlichen Tools für deren Betreiber erst dann zu einer „Mitgliedschaft" im rechtlichen Sinne führt, wenn diese wissen, dass damit eine bestimmte (Argument: „die Vereinigung") Gruppierung bei Straftaten unterstützt wird. Die bloße allgemeine Kenntnis, dass die Tools auch für kriminelle Zwecke eingesetzt werden, reicht hingegen nicht. Diese könnte jedoch immer noch für die konkrete Straftat als Beihilfe gewertet werden: Dass allgemein ein Interesse für kriminellen Nutzungen besteht ist bekannt, und wenn daher gar keine Gegenmaßnahmen getroffen werden, könnte man dies uU als bewusste Beihilfe durch Ignorieren aller Hinweise auf illegale Aktivitäten werten, selbst wenn zB das konkrete Opfer unbekannt bleibt.

Es werden daher zumindest gewisse Gegenmaßnahmen zu treffen sein[4]: Anweisungen des Musters „Erkläre mir, wie ich folgende Straftat begehen kann" könnten als „Wissen" des Betreibers über eine (geplante) Straftat auszulegen sein, insb da die Anfragen für die Weiterentwicklung der Modelle eingesetzt werden (wenn auch wahrscheinlich nicht mit menschlicher Einzel-Überwachung sondern nur generell[5]). Besonders bei Verfeinerungen, Nachfragen etc ist die Argumentation, dass dies lediglich als Scherz bzw zu Testzwecken gedacht war, wenig überzeugend und es sollte zumindest eine Nachfrage nach dem Zweck (wenn schon keine Blockierung der Antwort) erfolgen. Zumindest als Organisation besteht Kenntnis über die Ein- und Ausgaben, da die Prompts inkl der generierten Antworten zu diesem Zweck auch gespeichert werden[6]. Selbst eine Anonymisierung, dh die Entfernung des Personenbezugs (=wer diese Anfrage stellte und potentiell weitere Daten) ändert daran nichts: auch vorher ist (mangels Identifizierungsmöglichkeit für den Betreiber anhand der IP-Adresse; sofern kein Kunde mit Account) die Person nicht bekannt, wenn auch für Behörden in vielen Fällen identifizierbar. Dies ändert ebenso wenig an der Erkennbarkeit des intendierten kriminellen Einsatzes. Erst wenn die Eingabe so allgemein ist, dass sie zumindest auch für legale Nutzungen (dh: nicht praktisch ausschließlich für Illegales) geeignet ist, entfällt eine Kenntnis.

Bei KI bzw Sprachmodellen könnte es sich auch um besondere Werkzeuge im Sinne von § 126c StGB handeln: Problematisch daran ist, dass der „interessante" Teil von KI heute in vielen Fällen (siehe zB LLMs) kein Computerprogramm sein wird, sondern nur eine Menge an mathematischen Formeln bzw Gewichtungen. Das Programm zu deren Auswertung ist uU sehr einfach und nicht bedeutsam – das relevante ist die Essenz des Gelernten, welche sich gerade nicht im Programmcode niederschlägt. Allerdings sind auch „vergleich-

[3] Als Dienst bei Dritten erscheint dies auf den ersten Blick wenig vielversprechend, da der gesamte Text aller Mails dorthin übermittelt werden müsste. Doch würden dies die großen Mail-Provider (Google, Microsoft etc) anbieten ergäbe sich keine Änderung – diese sehen bereits jetzt den gesamten Inhalt (unverschlüsselter) Mails.

[4] Siehe hierzu auch *Staudegger*, Der Europäische Weg zur Regulierung Künstlicher Intelligenz – wie KI die Rechtswissenschaften fordert, jusIT 2023/2.

[5] ChatGPT setzt zB einen Filter sowohl für Prompts wie auch die generierten Ausgaben ein – dieser kann jedoch immer wieder umgangen werden.

[6] Siehe https://help.openai.com/en/articles/5722486-how-your-data-is-used-to-improve-model-performance (Opt-Out ist möglich).

bare Vorrichtungen“ erfasst. Da diese offensichtlich nicht identisch mit „Computerprogrammen“ sind, ist ein System aus Computerprogramm und Gewichtungs-Daten ein gutes Beispiel hierfür. Weiters erforderlich sind noch bestimmte Ziel-Straftaten aus Abs 1 Z1 bzw Abs 1a: Zumindest Datenbeschädigung (Ransomware) und widerrechtlicher Zugriff auf ein Computersystem (Herauslocken von Passwörtern[7]) kommen in Frage. Dies ist insb wichtig, da schon der Besitz eines solchen Programms strafbar ist – und über spezifisches Nach-Trainieren wird aus einem allgemein nutzbaren und legalen Tool ein verbotenes Werkzeug. Die zweite Alternative (§ 126c Abs 1 Z 2 StGB) könnte ebenfalls passen, da dort auch „vergleichbare Daten“ umfasst sind. Allerdings setzt dies voraus, dass diese den Zugriff auf ein Computersystem ermöglichen. Selbst wenn ein KI-System als Phishing-Tool eingesetzt wird, ist es jedoch nicht das Tool selbst, das den Zugriff ermöglicht. Dies erfolgt erst durch die Preisgabe bzw Eingabe durch den Berechtigten. Diese Alternative wird daher nicht verwirklicht.

Auch datenschutzrechtlich können sich Fragen stellen, ist zB das Trainieren einer KI (etwa für Deep Fakes) mit personenbezogenen Daten legal? Audio- oder Videoaufnahmen ebenso wie Texte einer Person sind personenbezogene Daten, insb wenn es darum geht, diese Person später vorzutäuschen – ihre Identität ist exakt bekannt. Geht man davon aus, dass es sich nicht um besondere Datenkategorien handelt (bei Videodaten uU umstritten), so kommt nur die Generalklausel Art 6 Abs 1 lit f DSGVO in Frage. Eine Verwendung der Daten um später strafbare Handlungen gegen die Betroffenen zu begehen ist keinesfalls ein „berechtigtes Interesse“ – doch dies oder zumindest die Kenntnis darüber müsste erst nachgewiesen werden (trainieren erfolgt – siehe Arbeitsteilung – wahrscheinlich durch andere Personen!). Andererseits ist nicht klar, welche anderweitigen legalen Zwecke ein derart trainiertes Modell erfüllen sollte. Und selbst dann ist mM das Interesse der betroffenen Person, nicht ohne ihre Zustimmung nachgeahmt werden zu können, höher anzusetzen. Daher ist bereits das Trainieren eines Modells für Deep Fakes bzw die Sammlung von Daten hierfür („Erhebung“ benötigt ebenfalls eine Rechtsgrundlage; auch hier dürfte eine legale Erklärung schwerfallen) als Datenschutzverletzung zu qualifizieren. Vorteilhaft ist, dass anders als bei generellen Sprachmodellen es bei diesen Systemen einfach ist nachzuweisen, dass eine bestimmte Person nachgebildet werden soll. Im Hinblick auf Sprachmodelle ist das Trainieren für die Imitation einer bestimmen Person ebenso als Datenschutzverletzung festzustellen: es werden personenbezogene Daten (die Texte) verwendet, welche darüber hinaus (zB bei Kommunikationsinhalten) auch weitere Betroffene enthalten können.

Deep Fakes sind ebenso im Hinblick auf Beweise problematisch: Der gefälschte (Bild und Ton) CEO ordnet in einem (interaktiven) Video eine Überweisung an. Selbst wenn es davon eine Aufzeichnung geben sollte (Live-Video-Telefonat!): Ist der/dem MitarbeiterIn dann Fahrlässigkeit vorzuwerfen? Und umgekehrt – falls es tatsächlich der Geschäftsführer war, der die Überweisung allerdings nachträglich bereut und daher behauptet, er wäre gefälscht worden? Gut gemacht Fälschungen sind aktuell sehr schwer erkennbar, sodass Fahrlässigkeit – wenn überhaupt – nur in niedriger Form anzunehmen ist. Typische Gegenmaßnahmen, zB durch Aufforderung zur Vornahme sehr ungewöhnlicher Bewegungen, Verwendung von Dialekt etc (in der Hoffnung, dass die KI diese nicht oder nicht schnell genug auswerten bzw imitieren kann) – sind in diesem Kontext auch kaum zielführend bzw unpassend. Eine unabhängige Überprüfung durch Kontaktaufnahme mit eigene Kommunikationsdaten in die Gegenrichtung (dh nicht durch klicken auf einen Link/Mailen an eine Adresse/… aus dem Video/Mail/…) erscheint hier als einzige Gegenmaßnahme sinnvoll und zielführend.

Der Einsatz von Analyseprogrammen zur Entscheidung, ob ein Artefakt ein Ergebnis eines KI-Tool ist bzw einen Angriffsversuch darstellt, kann uU rechtlich problematisch sein. Im Hinblick auf Programme ist zB das Verbote des Dekompilierens zu beachten: Vorhandener Source-Code kann untersucht werden, ausführbarer Code hingegen nur eingeschränkt: Wird nach Mustern in den Daten (zB enthaltenen Texten) gesucht, so liegt kein Dekompilieren vor. Wird jedoch der Code auch nur teilweise zurückverwandelt, zB welche Kombinatio-

[7] Auf das Problem, ob ein willentlich übergebenes Passwort (in Kenntnis der Sachlage bzw bei Phishing nach Täuschung) dieses Delikt erfüllt, soll hier nicht eingegangen werden. Siehe zB *Reindl-Krauskopf* in *Höpfel/Ratz,* WK² StGB § 118a (Stand 1.3.2022, rdb.at) RZ 27f.

nen von Befehlen kommen wie häufig vor, für welches Idiom wird welcher Code eingesetzt, so wird zumindest in kleinen Teilen der Quellcode wiederhergestellt – wofür jedoch (abseits einer vertraglichen Erlaubnis, zB bei Open-Source) eine Rechtsgrundlage fehlt. Neben der Prüfung selbst ist weiters das nicht immer besonders zuverlässige Ergebnis evtl problematisch, zB wenn jemandem fälschlicherweise vorgeworfen wird, dass ein Produkt (Text, Bild, Musikstück etc) von ihr/ihm angeblich gar keine eigene Schöpfung sei. Dies kann urheberrechtlich ein Problem sein (Nachweis der eigenen Schöpfung: evtl mangels Zwischenergebnissen schwierig), aber auch im Hinblick auf die öffentliche Meinung (Rufschädigung, politische Motive etc). Da entsprechende Werkzeuge zur KI-Erkennung derzeit eher schlechte Ergebnisse liefern (dh oft falsch liegen; ebenso wie Menschen!), ist bei der Formulierung von Aussagen große Vorsicht geboten. Dies gilt umso mehr, wenn eine Intention des Inhalts erkannt und klassifiziert werden soll (vgl das Beispiel oben – Phishing). Siehe etwa § 19 UrhG: Dieser behandelt nicht nur eine Falsch-Zuschreibung sondern auch überhaupt die Bestreitung einer Urheberschaft, passt also auch für (angebliche) KI-Werke. Dies setzt ein Feststellungsinteresse voraus, was jedoch in derartigen Fällen wohl leicht gegeben sein wird. Denn schon die bloße Behauptung, ein eigenes Werk sei in Wahrheit keine eigene Schöpfung, sondern künstlich erzeugt worden, ist für Urheber nachteilig. Auch § 111 StGB (Üble Nachrede) kommt in Frage, und zwar in Form „unehrenhaften Verhaltens“ durch die Ausgabe eines künstlichen Ergebnisses als Eigenes bzw die Behauptung, eine Nachricht wäre ein Betrugsversuch; beides wird auch als Herabsetzung zu werten sein. Eine Rettung durch Beweis guten Glaubens mittels Verweises auf die Ergebnisse eines Analysewerkzeuges dürfte (zumindest derzeit) nicht ausreichen: Bei bekanntermaßen hohen Fehlerraten wird auch diese anzuführen sein, bzw sind weitere, zB manuelle, Nachprüfungen oder Validierungen erforderlich. Dieses Problem ist bereits jetzt sehr ähnlich bei automatischer Spamerkennung anzutreffen und daher nicht neu.

4. KI als Programmier-Hilfsmittel

Künstliche Intelligenz wird inzwischen verstärkt in der Programmierung eingesetzt: man beschreibt ein Problem und lässt sich Programmcode hierfür automatisch generieren, bzw wird dieser während der Eingabe vorgeschlagen. Aus sicherheitstechnischer Sicht ist dies gefährlich, denn in vielen Fällen wird auf eine genaue menschliche Nachprüfung (was macht dieses Codestück überhaupt, funktioniert es auch in ungewöhnlichen Fällen, enthält es Sicherheitslücken etc) verzichtet werden. Hintergrund ist, dass die Überprüfung fremden Codes (= auch von KI generiert) oft aufwendiger ist, als diesen selbst zu schreiben[8]. Ist die KI bzw ihr Modell selbst kompromittiert, könnten so gezielt Sicherheitslücken in verschiedensten Programmen erzeugt werden, wo immer das Modell als Generator eingesetzt wird – und der Nachweis, dass dies nicht einfach nur eine zufällig fehlerhafte Ausgabe ist, kann ohne Trainingsdaten und genaue Protokolle des Trainingsvorgangs wohl nicht zu führen sein. Ein KI-Modell zu „hacken“ erscheint derzeit allerdings ebenso kaum möglich zu sein: Es wüsste entweder durch ein vom Angreifer erstelltes Modell ersetzt werden (=trivial erkennbar), oder es müsste die Manipulation schon während des Trainingsvorgangs erfolgen, zB durch Manipulation der Eingabedaten. Nachträgliche gezielte Anpassungen der Werte sind derzeit anscheinend unmöglich.

[8] Beispiel: Die automatisch generierte Berechnung der Zeit des Sonnenaufgangs an einem bestimmten geographischen Ort (https://codepal.ai/) liefert sehr überzeugenden Code, wobei jede Formel sprechende Namen für die Variablen hat (zB solarNoon, solarMeanAnomaly) und auch einen Kommentar enthält (zB “// Calculate the solar noon time”, “// Calculate the solar mean anomaly”). Daraus wird gleichzeitig sichtbar, dass diese Kommentare nicht unbedingt besonders hilfreich sind. Das Hautproblem ist jedoch, dass das Programm Fehler enthält, sodass immer auf ganze Stunden gerundet wird (Minuten sind immer 0), und die Formeln darüber hinaus fehlerhaft sein müssen: Die Sonne geht in Linz im Oktober nicht um 24 Uhr auf (Ergebnis der Berechnung vor falscher Zuweisung: -12,76 Stunden), und auch im Mai ist ein Sonnenaufgang um 23 Uhr (-13,0 Stunden) unwahrscheinlich. Ein anderer Generator (https://zzzcode.ai/) liefert zB ein Programm mit Syntaxfehler und ebenso (aber anders) falschem Ergebnis. In beiden Fällen wäre es sehr aufwändig herauszufinden, wo in den Formeln der Fehler steckt. Selbst die Identifikation des Minuten-Problems bzw die Behebung des Syntaxfehlers benötigt eine gewisse Zeit.

Aus rechtlicher Sicht sind hier zwei Aspekte besonders interessant: Handelt es sich bei umfangreichem KI-Einsatz noch um ein urheberrechtlich geschütztes Werk, bzw wie „autonom" ist der KI-generierte Code entstanden und handelt es sich bei diesem evtl um die (partielle) Kopie fremden Codes? Der erste Punkt ist wiederum ein Nachweisproblem: KI-generierter Code ist meistens eher kurz, und aufgrund seiner Eigenschaft als Programmcode funktional. Dies bedeutet, dass es im Vergleich zu Texten oder Bildern noch viel schwerer ist, diesen Code als „künstlich generiert" zu erkennen. Auch eine Markierung mit Wasserzeichen dürfte (außer evtl in miterzeugten Kommentaren) äußerst schwer bis unmöglich sein[9]. Dies bedeutet zweierlei: Einerseits ist es schwer bis unmöglich, den Einsatz von KI-Hilfsmitteln alleine aus dem Code heraus nachzuweisen (dh ohne Dokumentation des Einsatzes, Aussagen von ProgrammiererInnen, Vorgehensanweisungen des Unternehmens, Logs des KI-Anbieters etc). Auch eine stark reduzierte menschliche Kreativität wegen umfangreichem KI-Einsatz kann dann noch (insgesamt) als schöpferische Leistung gewertet werden (wobei das Niveau aufgrund der „Kleinen Münze" dennoch ausreichen sollte). Andererseits kann auch komplett menschlich geschriebener Code als KI-generiert behauptet werden, was zB im Hinblick auf Haftung (wäre das menschlich programmiert worden, wäre der Fehler wahrscheinlich nicht aufgetreten) bzw im Hinblick auf Preisminderungen (der Preis ist überhöht, da der menschliche Aufwand viel geringer war) oder Beanstandungen (es war menschliche Programmierung vereinbart, nicht automatische Generierung) relevant sein kann.

Der zweite Aspekt ist, ob es sich bei den KI-generierten Codestücken nicht um fremden (im Folgenden als urheberrechtlich geschützt angenommenen) Code handelt, sodass eine Übernahme eine Urheberrechtsverletzung darstellt. Hierbei ist insb auf die Sampling-Entscheidungen (BGH mehrfach, sowie EuGH) hinzuweisen (EuGH 29.07.2019, C-476/17): Auch sehr kurze Fragmente stellen eine Vervielfältigung dar (RZ 29). Ist der Ausschnitt jedoch geändert und nicht mehr wiedererkennbar, so liegt keine Vervielfältigung mehr vor (RZ 31). Es kommt daher in Analogie zu Musik-Samples darauf an, ob KI erzeugte kurzen Codestücke tatsächlich:

– verändert wurden: Dies ist bei Computerprogrammen wenig relevant, denn eine Veränderung analog zu Musik (zB Verzerrung) ist bei Programmcode kaum möglich. Es könnten jedoch zB Variablen umbenannt werden. Ansonsten ist bei Unterschieden fast immer davon auszugehen, dass Inhalte aus mehreren Quellen kombiniert werden, was jedoch dann den zweiten Punkt betrifft und keine „Veränderung" eines Teiles eines einzelnen Werks ist.
– wiedererkennbar sind: Dies ist theoretisch leicht zu beurteilen, indem das Ergebnis mit der angeblichen Quelle verglichen wird. Das Stück muss jedoch groß genug sein, dass es nicht aus beliebigen anderen Werken übernommen worden sein kann (Musik: eine Note; Programm: ein Befehl/einfache Zeile), also in einer Vielzahl an Werken vorkommt: Ob die Anweisung „for" einer Schleife aus Programm A oder Programm B stammt, ist anhand des Ergebnisses nicht entscheidbar. Es muss sich (Verletzung des Vervielfältigungsrechts) auch um ein einziges Werk handeln – Werke verschiedener oder auch desselben Urhebers fallen heraus. Selbst wenn es daher als Ganzes in dem behaupteten Werk vorkommt ist zusätzlich zu prüfen, ob es nicht identisch auch in anderen Werken vorkommt – und daher ebenso aus diesen stammen könnte (nicht wiedererkennbar für Teile die keinen Werk-Character erreichen; der Beweis, dass das Sprachmodell den Teil aus genau diesem Werk übernahm und nicht aus einem anderen ist selbst dem Hersteller unmöglich) oder es sich um eine übliche und landläufige Lösung handelt (und daher selbst kein Werk ist).

Zur rechtlichen Beurteilung ist es weiters erforderlich, die Arbeitsweise von Sprachmodell-basierter Codegenerierung zu untersuchen: Die meisten Systeme beruhen (teilweise mit Erweiterungen, zB um Erklärungen

[9] Es bestehen bei kurzen Sequenzen so wenige Freiheitsgrade, dass schon die Begründung der Kreativität Schwierigkeiten bereitet. Darin noch weitere Daten manipulationssicher zu verstecken dürfte einfach aus Platzgründen unmöglich sein (Watermarks bei Bildern sind zB wenige Bytes in sehr großen Dateien). Weiters sind Kommentare im ausführbaren Code nicht mehr enthalten.

oder Nachweise liefern zu können, bekanntermaßen fehlerhaften oder unsicheren Code auszufiltern[10] etc) auf großen Sprachmodellen. Diese arbeiten derart, dass für eine begrenzte Anzahl an vorgegeben Worten (Prompt, bzw dieser sowie die vorher generierten Worte; bei Programmierung unmittelbar davor – uU auch danach bzw an sonst relevanten Stellen – befindlichen Code) diejenige Worte/Befehle/Zeilen/Programmkonstrukte/… zu finden, die am wahrscheinlichsten danach kommen werden: Es handelt sich um eine „automatische Satz-Vervollständigung" basierend auf Wahrscheinlichkeiten, welche durch die Analyse einer großen Menge anderer Texte (hoffentlich menschlich generiert – ein Problem bei zukünftigem Trainieren von Modellen!) gelernt wurden (=Training). Das Problem aus urheberrechtlicher Sicht ist daher, dass jeweils auf mehrere Worte nur ein einziges (teilweise zufälliges) anderes Element folgt, was vielfach wiederholt wird. Es wird daher nicht danach gesucht, welches einer großen Anzahl an Computerprogrammen am ähnlichsten ist, und von dort eine Zeile oder eine ganze Methode entnommen: Dies wäre eindeutig ein Vervielfältigungsvorgang, denn aus einem bestimmten Einzelwerk wird ein unveränderter und wiedererkennbarer (wenn lang oder komplex genug) Teil übernommen. Werden hingegen wie hier nur einzelne Worte hintereinander zusammengesetzt, so stammen diese aus einer Wahrscheinlichkeitsfunktion, dh sind unabhängig von einzelnen Werken. Diese Werke trugen im Training dazu bei, die entsprechenden Wahrscheinlichkeiten zu erzeugen, werden aber nicht in ihren Wort-Kombinationen übernommen. Im Falle eines Verfahrens käme es daher mM nach darauf an, wie groß die Teile sind, die bei Programmier-Unterstützung generiert werden, bzw mit denen trainiert wurde. Sind es tatsächlich nur einzelne Worte/Befehle, so liegt eine Urheberrechtsverletzung durch die Verwendung des Modells (anders als beim Trainieren, wo das Werk jedenfalls als Ganzes verwendet wird, selbst wenn in kleinere Teile aufgesplittet[11]) nicht vor. Mittels speziellen Trainings, zB für eine ganz bestimmte Programmiersprache (dh ein Spezial-Modell nur für Java oder Python oder…), könnte syntaktisch korrekter Programmcode also selbst nach der Methode „Wort für Wort" bzw Befehlstoken/Variable/… entstehen. Wie exakt zB ChatGPT trainiert wurde, wird jedoch nicht preisgegeben.

Werden hingegen größere Teile (=mehrere Zeilen) als Ganzes generiert, dh in einem einzigen Schritt und daher auch als ganzes übernommen, so könnte eine Urheberrechtsverletzung vorliegen. Aufgrund der Einschränkungen, dass Programmcode einer strikten Syntax folgen muss, ist es möglich, dass Programmier-Unterstützungs-Systeme größere Teile (zB Zeilen) einsetzen, bzw Zwischenergebnisse mit größeren Einheiten abgleichen. Gegen eine Verletzung an einem einzigen Werk spricht weiters, dass Sprachmodelle nur dann Dinge lernen, wenn diese oft zumindest vergleichbar (wenn auch nicht identisch) im Trainingsmaterial vorkommen. Der eingefügte (in diesem Fall also größere) Teil ist daher in identischer oder zumindest sehr ähnlicher Form in mehreren anderen Programmen ebenso enthalten, ansonsten wäre er nicht gelernt und dann vorgeschlagen worden. Diese mehrfachen Vorkommen können natürlich (legale) Vervielfältigungen einer einzigen Ur-Quelle sein, sodass große Bereiche aus einem einzigen Werk, vielfach kopiert, auch als Ganzes gelernt und später reproduziert werden.

Problematisch hierbei ist, dass selbst der Hersteller des Modells oft nicht angeben kann, aus welchem Werk ein bestimmter Vorschlag stammt oder warum genau dieser (im Sinne: stammend aus welchem bzw welchen Trainings-Dokumenten) Vorschlag erfolgte. Es kann daher nur das Ergebnis verglichen werden, aber die „Entstehungsgeschichte" bleibt selbst dem Hersteller verborgen. Für eine Urheberrechtsverletzung durch Vervielfältigung ist Vorsatz oder Verschulden allerdings nicht erforderlich. Bei längeren Übereinstimmungen (=Angebliche Übernahme aus einem bestimmten Werk; wiedererkennbar) bleibt folglich nur mehr die Möglichkeit, plausible Alternativquellen zu suchen: Könnte es auch aus anderen Werken stammen, welche zum Training verwendet wurden (den Trainings-Korpus kennt zB nur der Hersteller der Modells!) und sind diese keine Kopien der angeblichen Quelle? Ist dies der Fall, so ist nach dem Funktionsmodus anzunehmen, dass es

[10] https://github.blog/2023-02-14-github-copilot-now-has-a-better-ai-model-and-new-capabilities/.

[11] Siehe dazu *Oehri*, Chat GPT & Co. – Was sagt das Urheberrecht? https://hub.hslu.ch/management-and-law/2023/04/05/chat-gpt-co-was-sagt-das-urheberrecht/.

tatsächlich aus einer Kombination bzw vielen Werken erzeugt wurde. Erst dann müsste eine Urheberin nachweisen, dass gerade von ihr kopiert wurde (und nicht den anderen Quellen), was sie aber genauso wenig wie der Hersteller des Modells können wird.

Künstliche Intelligenz kann weiters beim Testen von Programmen eingesetzt werden: eine typische Art des Testens ist es, automatisiert Unmengen an Varianten zufälliger Eingaben an das Programm zu schicken und die Ergebnisse (bzw Abstürze) zu prüfen. Dies ist in der beschrieben rein-zufälligen Form nicht sehr hilfreich oder effizient, weil fast alle Eingaben als „unzulässig" abgewiesen werden und daher kein tiefergehender Test erfolgt. In der Vergangenheit wurden daher bereits verschiedenste Strategien entwickelt, derartige Tests gezielter durchzuführen (zB menschliche Bedienungsvorgänge als Ausgangspunkt, welche dann automatisiert variiert werden). Mittels künstlicher Intelligenz könnte dies verbessert werden, indem weniger Vorbereitung erforderlich ist, und die Prüfung genauer erfolgt. Dies scheint rechtlich keine Auswirkungen zu besitzen: es werden keine (neuen/anderen) personenbezogenen Daten eingesetzt und es handelt sich lediglich um eine etwas anderer Art des Testens (welche selten vertraglich exakt vereinbart wird), wobei verwandte Verfahren schon jetzt eingesetzt werden. Als zusätzliche Methode ist dies daher auch nicht als „Mangel" anzusehen. Lediglich wenn ausschließlich dies eingesetzt wird, kann man nicht von einem Testen entsprechend dem Stand der Technik sprechen. Abgesehen davon scheint nach Berichten im Internet Code-Generierung für Testmethoden besonders gut zu funktionieren (sehr viele Teile hierbei sind repetitiv und immer wieder sehr ähnlich) und daher häufig eingesetzt zu werden. Selbst wenn personenbezogene Daten (rechtmäßig) zum Nachtrainieren eines speziellen Modells für eine bestimmte Anwendung eingesetzt werden, sind diese im Ergebnis nicht mehr als solche enthalten (sondern nur mehr als unzusammenhängende Einzelwerte), sodass die Nutzung des Modells für andere Programme damit kein Rechtsproblem darstellt. Auch hier kommt es jedoch auf die Größe der Trainingselemente an: Werden zB Vor- und Nachname nicht getrennt sondern als Ganzes betrachtet und damit gelernt, bleibt dieser Name als Ganzes im Modell enthalten – woraus die Beinhaltung personenbezogener Daten folgt (zumindest, dass diese Person als Testdatenquelle diente).

5. Zusammenfassung

Künstliche Intelligenz stellt auch die IT-Sicherheit vor Herausforderungen: Einerseits sind evtl Verbesserungen bei der Erkennung von Angriffen möglich, andererseits kann sie auch gerade für individualisierte Angriffe eingesetzt werden. Zumindest derzeit ist nicht absehbar, welcher Aspekt stärker ausgeprägt sein wird und ob es daher insgesamt gefährlicher oder sicherer wird. Die wahrscheinlichste Variante dürfte jedoch sein, dass das Ergebnis sich je nach Sektor unterscheiden wird.

Im Hinblick auf den Einsatz bei der Programmierung ist derzeit davon auszugehen, dass Urheberrechtsverletzungen schwer nachzuweisen sind. Es ist oft unklar, wie die Modelle und mit welchen Daten trainiert wurden (siehe dazu den AI-Act der EU). Auch die „Nachbearbeitung" wird nicht offengelegt, doch genau diese ist bei dem Grundkonzept von LLMs, der statistischen Aneinanderreihung von Wörtern, bei Computerprogrammen besonders wichtig. Dieses Problem ist jedoch nicht ganz neu: auch bisher durfte zB für den Nachweis von Urheberrechtsverletzungen ein Computerprogramm nicht decompiliert werden.