Hochschule
Ravensburg-Weingarten

Hochschule
Ravensburg-Weingarten
Technik | Wirtschaft | Sozialwesen

Entwicklung einer Anwendung zur automatisierten Beschaffung von personenbezogenen Daten im Internet und deren Integration in Phishing-Mails

Bachelorarbeit

Social Engineering

im Studiengang Angewandte Informatik

an der Hochschule Ravensburg - Weingarten

von

Marco Lang Matr.-Nr.: 27416

Abgabedatum : 17. Februar 2019

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit mit dem Titel

Entwicklung einer Anwendung zur automatisierten Beschaffung von personenbezogenen Daten im Internet und deren Integration in Phishing-Mails

selbstständig angefertigt, nicht anderweitig zu Prüfungszwecken vorgelegt, keine anderen als die angegebenen Hilfsmittel benutzt und wörtliche sowie sinngemäße Zitate als solche gekennzeichnet habe.

Weingarten, 17. Februar 2019

Autor Name

_____

Inhaltsverzeichnis

Kurzfassung

Im Rahmen dieser Abschlussarbeit wird gezeigt, wie eine automatisierte Suche nach personenbezogenen Daten im Internet aussehen kann und wie diese Daten für einen Phishing-Mail-Angriff verwendet werden können.

Abstract

Danksagung

Vorwort

1 Einleitung

1.1 Motivation

Laut dem Bundeskriminalamt hat sich die Zahl der Cyberkriminalität mit einem klaren Trend nach oben entwickelt. [Bun18] Aus diesem Grund werden System immer sicherer und Firewalls immer noch besser. Das hat zu Folge, dass Angreifer oft auf Methoden ausweichen, bei denen der Mensch als Schwachstelle des Systems ausgenutzt wird. Daher ist das E-Mail-Phishing eine häufig verwendetet Technik von Cyberkriminellen.
In den neusten Fällen von Phishing-Mail-Attacken zeigt die Verbraucherzentrale Nordrhein-Westfalen, dass diese meist direkt an eine Person adressiert sind. Das heißt, in dieser Art von E-Mail, werden personenbezogene Daten verwendet. Ein Beispiel dafür, sind die gefälschten DSGVO-E-Mails. Hier wird die Zielperson im Auftrag der Sparkasse, persönlich mit Namen angesprochen. [NW18]
Solch ein Angriff benötigt im Voraus eine ausführliche Recherche über das Opfer. Als Informationsquelle für die Recherche können beliebig viele Quellen verwendet werden. Allerdings ist in der heutigen Zeit das Internet eine der meistgenutzten Informationsquellen und birgt dadurch Gefahren für jeden einzelnen Menschen, von dem personenbezogene Daten im Internet frei zugänglich sind. [All18]

1.2 Zielsetzung

Ziel dieser Arbeit ist es eine Anwendung zu entwickeln, welche automatisiert nach personenbezogenen Daten im Internet sucht und daraus eine Phishing-Mail generiert. Dabei

soll der Fokus auf der automatisierten Informationsbeschaffung liegen.
Es sollen grundsätzlich zwei Arten von Suchfunktionen mit diesem Programm möglich sein.

Ziel 1 *Informationen zu einer ausgewählten Person im Internet suchen.*

Die erste Suchfunktion beinhaltet die Suche nach Informationen einer bestimmten Person. Dadurch können bereits bekannte Daten über die Person angegeben und somit die Suche verfeinert beziehungsweise verbessert werden. Hierbei ist es wichtig zu erkenne wann es sich um eine Information der gesuchten Person handelt.

Ziel 2 *Nach Informationen einer großen Anzahl von unbekannten Personen suchen, indem eine festgesetzte Webseite vollständig durchsucht wird.*

Bei dieser Suchfunktion soll eine bestimmte Webseiten vorgegeben werden, welche durchsucht, analysiert und ausgelesen wird. Dadurch ist es möglich einen weitläufigen "*real-world*" Phishing-Mail-Angriff zu simulieren.

Ziel 3 *E-Mail-Adressen aus den gewonnen Daten generieren.*

Durch die Zusammensetzung von Vorname, Name und Geburtsjahr werden die E-Mail-Adressen generiert. Außerdem kann der Arbeitgeber, falls er bekannt ist, mit in den Generierungsprozess einfließen.

Ziel 4 *Phishing-Mail-Muster erstellt*

Abhängig von den gefundenen Informationen, soll mit Hilfe dieser Muster, eine Phishing-Mail mit glaubhaftem und sinnvollem Inhalt erstellt werden.

Ziel 5 *Phishing-Mail erzeugen.*

Mit der vorhandenen Information, der E-Mail-Adresse und einem passende Muster, soll eine Phishing-Mail erzeugt und versendet werden können.

1.3 Eigene Leistung

In dieser Arbeit wird ein Programm erstellt, welches personenbezogene Daten automatisiert aus dem Internet heraussucht und diese in potentielle Opferprofile ablegt. Die gewonnenen Informationen werden automatisiert in eine personalisierte Phishing-E-Mail eingebaut. Für einen höheren Erfolg werden E-Mail-Muster konzeptioniert und realisiert.
Damit ein kompletter Ablauf eines Phishing-Mail-Angriffs simuliert werden kann, wird ein Algorithmus entwickelt, der aus den gewonnen Informationen eine E-Mail-Adresse generiert.

1.4 Methodische Vorgehensweise

Die Arbeit gliedert sich in einen theoretischen und praktischen Teil auf. Die Theorie beginnt im zweiten Kapitel und beschreibt die Grundlagen 2 im Bereich von personenbezogenen Daten, Social Engineering und der Informationsbeschaffung im Internet. In Kapitel 3 wird das Problem aufgezeigt, auf welches in dieser Arbeit eingegangen wird. Darauf folgt die ethische und rechtliche Betrachtung in Kapitel 4. Die Anforderungsanalyse 5 beschreibt das nächste Kapitel, in welchem die Anforderungen und Prioritäten der Arbeit festgelegt werden. Darauf folgen die Lösungsvorschläge im Kapitel 6 und die Auswahl der Lösung anhand den Anforderungen im Kapitel 7. Anschließend wird bei der Umsetzung auf den Praktischen Teil eingegangen. Dieser unterteilt sich in die Themen Informationsbeschaffung einer ausgewählten Person 8, Informationsbeschaffung einer großen Menge an unbekannten Personen 9 und die Erstellung einer Phishing-Mail 10. Am Ende dieser Arbeit befindet sich die Evaluation der Implementation in Kapitel 11 und die Schlussbemerkung und der Ausblick in Kapitel 12.

2 Grundlagen

2.1 Personenbezogene Daten

Laut dem DSGVO sind *personenbezogene Daten*

"alle Informationen, die sich auf eine identifizierte oder identifizierbare natürliche Person (im Folgenden „betroffene Person") beziehen; als identifizierbar wird eine natürliche Person angesehen, die direkt oder indirekt, insbesondere mittels Zuordnung zu einer Kennung wie einem Namen, zu einer Kennnummer, zu Standortdaten, zu einer Online-Kennung oder zu einem oder mehreren besonderen Merkmalen identifiziert werden kann, die Ausdruck der physischen, physiologischen, genetischen, psychischen, wirtschaftlichen, kulturellen oder sozialen Identität dieser natürlichen Person sind;" [DSG]

2.2 Social Engineering

Die Definition von Social Engineering (SE) ist nicht eindeutig. Es gibt sehr verschiedene Ansichten von der Definition. Die Idee von Social Engineering ist, eine Ziel so zu manipulieren, damit das Ziel eine für den Angreifer bessere Entscheidung trifft. In dem Buch Social Engineering - The Art of Human Hacking, von Christopher Hadnagy, ist Social Engineering definiert als:

"social engineering is the act of manipulating a person to take an action that may or may not be in the "target's" best interest" [Had11]

Wiederum lautet die Definition in dem Buch von Kevin D. Mitnick:

"Social Engineering uses influence and persuasion to deceive people by convincing them that the social engineer is someone he is not, or by manipulation. As a result, the social engineer is able to take advantage of people to obtain information with or without the use of technology" [Mit01]

SE wird Menschen von Geburt an beigebracht und begegnet einem beinahe jeden Tag. Schon ein Baby muss wissen wie es die Eltern manipulieren kann, damit es Dinge wie Essen, Zuneigung, oder ähnliches bekommt. Darüber hinaus ist SE in vielen Berufen ein täglicher Bestandteil. Beispielsweise manipulieren Ärzte viele Patienten mit einer Placebo-Behandlung. Bei dieser Behandlung wird dem Patient ein wirkstoff-freies Medikament verschrieben. Ausschließlich durch die Manipulation des Patienten und den sogenannten Palzebo-Effekt können Erfolge erzielt werden.

Im Bereich der Informationssicherheit, wird von Social Engineering gesprochen, wenn Angreifer durch die Manipulierung und Täuschung von Menschen vertrauliche Informationen oder Zugänge zu Systemen bekommen. Die bekanntesten Angriffsmethoden sind Phishing, Pretexting, Baiting und Quad Pro Quo. Bei dieser Arbeit wird hauptsächlich auf das Thema E-Mail-Phishing eingegangen.

Der Aufbau eines SE-Angriffes ist definiert in mehrere Phasen. Das wohl bekannteste Modell für einen Social Engineering-Angriffszyklus ist in dem Buch von Kevin D. Mitnicks [Mit01] definiert. Dieser Zyklus besteht aus den 4 Phasen **Research, Developing rapport and trust, Exploiting trust** und **Utilize information**.
In der **Research-Phase** geht es um die Informationsbeschaffung. Bei dieser Phase will der Angreifer möglichst viele Informationen über das Ziel herausfinden. Die **Developing Rapport and Trust-Phase** beschreibt den Kontaktaufbau zum Ziel, da wenn das Opfer dem Angreifer vertraut, hat dieser ein leichteres Spiel in den kommenden Phasen. Das nun erzeugte Vertrauen wird in der **Exploitung Trust-Phase** ausgenutzt. Hier will der Angreifer die eigentlich Information vom Opfer herausfinden. Dies geschieht einerseits durch bestimmtes Nachfragen oder durch Manipulation. **Utilize Information** ist die letzte Phase. Dort wird die gewonnene Information genutzt um das eigentliche Ziel des Angreifers zu erreichen.
Grundsätzlich werden bei einem Social Engineering Angriff menschliche Wünsche, Ängste

und verbreitete Verhaltensmuster verwendet um ein Opfer zu manipulieren. [uDsiNe15]

2.2.1 Phishing

Das Wort Phishing wird von dem Wort “fishing“ abgeleitet, da die Angreifer nach Informationen fischen. Das “Ph“ kommt von “sophisticated“ und meint damit, dass die Angreifer ausgeklügelte Techniken verwenden um an Informationen heranzukommen. [Jam05]
Die wohl bekannteste Angriffsmethode von Phishing ist das E-Mail-Phishing. Bei diesem Verfahren, versendet ein Angreifer meist eine gefälschte E-Mail, um ein Opfer zu täuschen und dadurch sein Ziel zu erreichen. Die sogenannten Phishing-Mails enthalten meist eine Aufforderung einen Link zu öffnen und sehen täuschend echt aus.
Ein reales Beispiel könnte sein, dass der Angreifer eine gefälschte E-Mail von Amazon an das Opfer versendet und es dabei auffordert, einen Link in der Mail zu öffnen. Nachdem die Zielperson auf den Link geklickt hat, muss Sie sich anmelden. Hier könnte der Angreifer ein täuschend echtes Anmeldeformular erstellt haben, um die Anmeldedaten der Zielperson zu bekommen. Sobald die Anmeldedaten eingegeben wurden, könnte eine Fehlermeldung erscheinen, die einen Authentifizierungsfehler beinhaltet und das Opfer auffordert sich erneut anzumelden. Jedoch wird während diesem Prozess das originale Anmeldeformular geladen und das Opfer kann sich korrekt bei der entsprechenden Webseite anmelden.
Dieser Verfahren ermöglicht Angreifern die Anmeldedaten von einer Zielperson ohne großen Aufwand zu beschaffen. Allerdings benötigt der Angreifer für diese Methode nicht nur Social Engineering sondern auch technische Fähigkeiten. [CH15]

Spear-Phishing

Das Spear-Phishing ist eine erweiterte Methode des herkömmlichen E-Mail-Phishings. Hierbei wird anstatt das Versenden etlicher Phishing-Mails an unbekannte Opfer, eine gezielte Mail an eine ausgewählte Person versendet. [Fir]
Bei dieser Form von E-Mail-Phishing spielt die Opferauswahl und die Informationsbeschaffung eine sehr große Rolle, da diese Information später für personalisierte E-Mails oder vorgetäuschte Identitäten verwendet werden können. Durch diese Art von Täuschung kann

ein Opfer dazu bewegt werden auf einen Link zu klicken und dadurch eine Schadsoftware herunterzuladen. [Fir]
Der Aufwand für die Informationsbeschaffung wird oft in Kauf genommen, da der Erfolg bei dieser Methode vielversprechender ist als beim herkömmlichen E-Mail-Phishing.
91% der Advanced Persistent Threat (APT) Angriffe auf Firmen beginnen mit einer Spear-Phishing-E-Mail. Die Schadsoftware wir meisten als Remote Access Trojans (RATs) in einem Zip-Datei überliefert. [Cal13]

2.3 Informationsbeschaffung im Internet

2.3.1 Web Crawler

Web Crawler sind Computerprogramme, die mit Hilfe der Hypertextstruktur das Internet durchlaufen. Dabei können sie in einen **internen** und **externen Web Crawler** unterschieden werden. Der interne Web Crawler durchsucht ausschließliche interne Seiten einer Webseite und der externe Web Crawler durchsucht unbekannte Webseiten im ganzen Netz. [SG12]

In anderen Worten besteht die Funktionsweise darin, dass in den meisten Fällen ein automatisiertes Programm, Web Crawler, erstellt wird. Dieser lädt Webinhalte herunter und durchsucht den Inhalt nach Hyperlinks. Den gefundenen Links wird gefolgt, um neue Webseiten mit weiteren Links zu laden. So hangelt sich ein Web Crawler von Link zu Link durch das Internet. [Mit15]

2.3.2 Web Scraper

In der Theorie bedeutet *web scraping* die Informationsbeschaffung im Internet mit unterschiedlichsten Mitteln. [Mit15]
Meist wird dies mit einem automatisierten Programm realisiert, welches Daten von einem Webserver anfragt, entgegen nimmt, analysiert und auswertet. In der Praxis gibt es ein

großes Feld von Programmiertechniken und Einsatzmöglichkeiten. Mit Hilfe eines Web Scrapers ist es möglich, große Datenmengen zu erfassen und zu verarbeiten. [Mit15]

Natural Language Processing

Natural Language Processing kurz *NLP* beschreibt eine Technologie, für die Kommunikation zwischen Mensch und Computer. Mit dem Ziel, dass ein Computer die natürliche Sprache verstehen und verarbeiten kann. Dafür werden verschiedenste Methoden aus der Sprach- und Computerwissenschaft sowie aus der künstliche Intelligenz verwendet. Unter anderem hat eine NLP-Anwendung die Aufgabe von **Stemming**. [Lit16]

Stemming ist eine Methode der Wortstandardisierung, bei der verwandte Wörter auf ihrer Stammform reduziert werden. Dabei wird bei dem Rechenvorgang auf den Stamm und die Semantik eines Wortes geachtet. Aus diesem Grund fällt der Name Stammformreduktion öfters in Verbindung von Stemming. [EAD09]
Die Verwendung von Stemming, kann bei der Schlüsselwortgenerierung von Texten sehr hilfreich sein, da die Anzahl der möglichen Schlüsselwörter reduziert werden können.

3 Problemspezifikation

Persönliche Daten sind im Internet oft frei zugänglich. Das heißt, dass unterschiedlichste Webseiten persönliche Information von Menschen öffentlich bereitstellen. Die bekanntesten Webseiten sind die Social Media Seiten wie Twitter, Facebook und Instagram. Allerdings wird auch auf anderen Webseiten personenbezogene Daten in großen Mengen bereitgestellt. Ein Beispiel dafür ist das Fußballportal *"www.fupa.net"*. Diese Art von Webseiten sind perfekte Informationsquellen für Phisher, da im Bereich von Social Engineering, diese Informationen oft genutzt werden um ein Opfer zu täuschen oder zu manipulieren.
Dass hier beschriebene Problem zeigt, dass der Zugang für persönliche Information durch das Internet für die Öffentlichkeit einfacher gemacht wird. Es soll mit einem kritisch Blick darauf gezeigt werden, mit welchem Aufwand, personenbezogene Daten aus dem Internet herausgelesen, analysiert und für einen Phishing-Mail-Angriff verwendet werden können.

4 Ethische und rechtliche Betrachtung

Das Sammeln von personenbezogenen Daten auf sozialen Netzwerken ist ethisch gesehen ein sehr sensibles Thema. Jedoch werden in dieser Arbeit ausschließlich die Daten verwendet, die öffentlich frei zugänglich sind. Das heißt, unter den Informationen befinden sich keine Passwörter oder Informationen die nicht an die Öffentlichkeit gehören.
Mit diesem realen Experiment, soll die Privatsphäre der Benutzer geschützt werden, indem aufgezeigt wird, wozu veröffentlichte Daten über eine Person im negativen Sinn verwendet werden können. Genau aus diesem Grund ist es wichtig, dass das Experiment in der realen Welt durchgeführt wird.
Des Weitern kann gesagt werden, dass der hier verwendete Crawler nicht stark genug ist, um die Leistung eines sozialen Netzwerkes zu beeinflussen.

5 Anforderungsanalyse und Priorisierung

Die im Kapitel 1.2 definierten Ziele sollen mit den folgenden Anforderungen gewährleistet werden.

5.1 Anforderung an die Informationsbeschaffung

Die Anforderung an die Informationsbeschaffung von personenbezogenen Daten lässt sich in zwei Teile gliedern. Der erste Teil beinhaltet die Informationsbeschaffung von ausgewählten Personen und der zweite Teil die Informationsbeschaffung von einer großen Menge unbekannter Personen.

5.1.1 Informationsbeschaffung von einer ausgewählten Person

Bei dieser Informationsbeschaffung soll eine Suchfunktion entwickelt werden, welche Daten zu einer angegeben Person im Internet sucht. Hierbei sollen so viele Daten wie möglich gefunden und gespeichert werden. Dies soll mit Hilfe eines Web Crawlers und mit einem Web Scraper umgesetzt werden.
Das zu entwickelnde Programm soll für die Suche bekannte Daten wie Vorname, Nachname, Geburtsjahr, Ort und Benutzernamen von Social Media Plattformen einlesen können. Die Eingabe kann mit Hilfe einer Konsole oder einer grafische Oberfläche realisiert werden.
Die Herausforderung besteht darin, zu erkennen, wann und ob es sich um die Information der gesuchten Person handelt. Sowie die Analyse und das Herauslesen dieser Daten.

5.1.2 Informationsbeschaffung von unbekannten Personen

Es soll eine Prototyp-Suchfunktion entwickelt werden, die eine komplette Website nach personenbezogenen Daten durchsucht. Dabei sollen möglichst viele Informationen von möglichst vielen Personen herausgefunden werden. Jedoch sind diese Personen dem Programm-Anwender unbekannt. Die Informationen werden aus Webseiten mit einer großen Anzahl von Mitgliedern herausgelesen. Die festgesetzte Webseite wird vollständig nach personenbezogenen Daten durchsucht.
Dabei soll der zu entwickelnde Web Scraper möglichst performant arbeiten und kann hartkodiert werden. Allerdings müssen E-Mail-Adressen ebenfalls gefunden werden können, obwohl die Position einer E-Mail-Adressen auf einer Webseite variieren kann.

5.2 Anforderung an die Datenverwaltung/-speicherung

Ausgelesene Daten sollen vor dem speichern formatiert und klassifiziert werden, damit die Daten später korrekt in die Phishing-Mails eingesetzt werden können. Die Schwierigkeit besteht darin, zu erkennen, um welche Art von Information es sich handelt. Zusätzlich sollen die Daten in einer gut übersichtlichen Struktur gespeichert werden und müssen beliebig erweiterbar sein.

5.3 Anforderung an die Generierung der E-Mail-Adressen

Da nicht zu jeder Suche eine E-Mail-Adresse im Internet gefunden werden kann, muss die E-Mail-Adresse aus den vorhandenen Informationen generiert werden. Es soll eine größere Anzahl von möglichen E-Mail-Adressen erzeugt werden. Durch den Pool an erzeugten E-Mail-Adressen soll die Wahrscheinlichkeit erhöht werden, dass die richtige E-Mail-Adresse dabei ist. Des Weiteren sollen die Adresse auf Verfügbarkeit und Gültigkeit geprüft werden.

5.4 Anforderung an die E-Mail-Muster

Bei der Erstellung der E-Mail-Muster handelt es sich ausschließlich um das Erstellen potentieller Inhalte einer E-Mail, welche mit den gewonnenen Informationen über eine Person erweitert werden kann. Die Muster sollen erstellt werden und so klassifiziert sein, dass für jedes gefundene Opferprofil ein passendes Muster vorhanden ist. Des Weiteren soll der E-Mail-Text mit den eingesetzten Informationen Sinn ergeben und eine korrekte Grammatik beinhalten. Weiterführend können SE-Fähigkeiten genutzt werden um die Zielperson tatsächlich zu manipulieren und zu täuschen. Hierfür können beispielsweise Gefühle wie Freude und Angst ausgenützt oder gefälschte E-Mails von bekannten Firmen in Betracht gezogen werden.

5.5 Anforderung an die Erstellung der Phishing-Mail

Die Phishing-Mails sollen automatisiert erstellt werden. Die Auswahl des richtigen E-Mail-Musters zu der gewonnenen Opferinformation soll ebenfalls automatisiert ablaufen.

5.6 Weitere Anforderungen

Unter anderem soll die Arbeit Antworten auf die folgenden Fragen finden. Mit welchem Aufwand ist eine Phishing-Mail-Angriff verbunden? Ist es möglich ein Personenprofil zu erstellen, bei dem ausschließlich korrekte Informationen vorhanden sind?

5.7 Priorisierung

Die Tabelle 5.1 zeigt die Priorisierung der Anforderungen. Dabei liegt der eindeutige Fokus auf der Informationsbeschaffung von personenbezogenen Daten und der Erstellung von E-Mail-Mustern.

Tabelle 5.1: Priorisierung der Anforderungen

Anforderung	Priorisierung (A-C)
Informationsbeschaffung von ausgewählten Personen	*A*
Informationsbeschaffung von vielen ubekannten Personen	*A*
E-Mail-Muster erstellen	*A*
Phsishing-Mail erzeugen	*B*
Datenverwaltung/-speicherung	*B*

6 Lösungsideen

In diesem Kapitel werden die Lösungsideen für die Umsetzung der im Kapitel 1.2 definierten Ziele beschreiben.

6.1 Konzept zur Informationsbeschaffung einer ausgewählten Person

6.1.1 Methoden zur Suche nach einer Person im Internet

Für die Suche einer Person im Internet, wird abhängig von den eingegebenen Daten des Programm-Anwenders, die Art der Suche angepasst. Genau genommen heißt das, dass die eingegebenen Daten vor der Suche analysiert werden und dementsprechend die Suche danach angepasst wird.
Die Art der Personensuche lässt sich in zwei Methoden gliedern.

Personensuche mit Hilfe von Suchmaschinen

Hier wird mit Hilfe einer Suchmaschine nach Informationen gesucht. Allerdings muss nicht für jede Suche eine Suchmaschine verwendet werden. Die nachfolgenden Fälle sollen diesen Ansatz verdeutlichen.

Im Fall, dass der Vorname, Nachname und Wohnort der gesuchten Person eingegeben wird, kann mit Hilfe von herkömmlichen Suchmaschinen wie die von Google und Bing nach Information gesucht werden. Die von den Suchmaschinen vorgeschlagenen Seiten werden

anschließend analysiert, ausgelesen und gespeichert. Dadurch können weitere Informationen gewonnen werden. Falls Benutzernamen von anderen Webseiten wie Instagram, Facebook oder ähnliches vorgeschlagen werden, kann somit die Suche mit diesen Daten speziell auf den entsprechenden Seiten erweitert werden.

Ein weiterer Fall beschreibt das Szenario, wenn ein Benutzername der gesuchten Person in das Programm eingegeben wird. Hierbei handelt es sich um einen Benutzernamen von Social-Media-Webseiten wie Facebook, Instagram, et cetera.
Zuallererst, wird hier die entsprechende Webseite nach Informationen zu dem angegebenen Benutzername durchsucht. Dadurch können zusätzliche Daten herausgefunden werden, die bei der weiteren Suche von Vorteil sind.
Sobald die Webseite nach dem Nutzernamen durchsucht und ausgewertet wurde, kann mit herkömmlichen Suchmaschinen die Suche erweitert werden.

Personensuche ohne Suchmaschinen

Unabhängig von den eingegebenen Daten, wird eine festgesetzte Anzahl von Webseiten durchsucht. Diese Art der Personensuche arbeitet allerdings ohne die Verwendung einer Suchmaschine. Vorschläge für die ausgewählten Webseiten sind Facebook, FuPa, Instagram, Xing, LinkedIn und Twitter.

6.1.2 Methoden zum Erkennen einer Person

Bei jeder einzelnen Suchvariante, besteht die Herausforderung darin, zu erkennen, wann es sich um die gesuchte Person handelt. Durch die große Anzahl an verfügbaren Informationen im Internet, besteht eine hohe Wahrscheinlichkeit, dass Personen mit sehr ähnlichen Profilen gefunden werden. Um diesem Problem entgegen zu wirken, kann die Art der Suche anhand den eingegebene Daten angepasst werden. Dies entspricht dem Ansatz im Kapitel 6.1.1. Die Suche kann dadurch verfeinert werden und die Anzahl der fehlerhaften Vorschläge wird geringer. Dadurch wird die Wahrscheinlichkeit höher, dass es sich um die richtige Person handelt.

Darüber hinaus kann die Personensuche mit einer Suchmaschine durch verbesserte Suchbefehle ebenfalls verfeinert werden. In dem Buch “Open Source Intelligence Techniques“ [Baz18], werden Suchbefehle für bekannte Suchmaschinen aufgezeigt, mit denen die Suche verbessert werden kann. Dies bedeutet, bei einer Personensuche ist es mit den richtigen Suchbefehlen möglich, die Anzahl der Vorschläge um einen großen Teil zu verringern. Ein Beispiel in dem Buch von Michael Bazzell zeigt, wie es funktioniert, die Suchergebnisse von 8770 Vorschlägen auf lediglich neun Vorschläge zu reduzieren. [Baz18] Auch bei dieser Lösungsidee wird die Wahrscheinlichkeit erhöht, dass es sich um die gesuchte Person handelt.

Im Fall dass nach diese Maßnahmen dennoch verschiedene Profile angezeigt werden, können die folgenden Erweiterungen in die Suche mit einfließen.

Erweiterte Kriterien

Hierbei handelt es sich um weitere Kriterien, welche die Suche noch mehr eingrenzen sollen. Bekannte Informationen über die Zielperson dienen dazu, die vorgeschlagenen Seiten einer Suchmaschine weiter zu filtern. Ausführlich bedeutet dies, dass das Programm in erster Linie nur die Webseite als Informationsquelle verwendet, die alle Suchbegriffe beinhaltet. Darüber hinaus kann das genaue oder grobe Alter der Zielperson mit in die Suche einfließen. Dadurch kann erkannt werden ob der Zeitrahmen des Artikels oder das Erstellungsdatum einer Webseite mit dem Alter der Person grundsätzlich übereinstimmt.

Kontakte der Suchperson werden in Betracht gezogen

Hier kann die Suche erweitert werden, indem auf soziale und berufliche Verbindungen der Zielperson eingegangen wird. Das heißt, dass bekannte Kontakte der gesuchten Person ebenfalls durchsucht und ausgewertet werden. In diesem Fall könnten Facebook-Freunden, FuPa-Teammitglieder, Instagram-Follower oder LinkedIn/Xing-Kontakte als Kontaktquelle dienen.
Durch dieses Verfahren können weitere Informationen gewonnen werden, die zur Unterscheidung von Profilen nützlich sein könnten.

Identifikationsschlüssel verwenden

Bekannte Information zur Person können als Identifikationsschlüssel verwendetet werden. Allerdings müssen dies einzigartige Daten sein. Als einzigartige Daten zählen beispielsweise die E-Mail-Adresse oder eine Verbindung von mehreren Daten, da der vollständige Name nicht einzigartig ist. Das heißt, häufig verwendete Namen können oft in Verbindung mit unterschiedlichen Personen im Internet vorkommen und sind dadurch nicht als Identifikationsschlüssel verwendbar. Des Weiteren, kann eine Zielperson auf einer Webseite einen erfundenen Benutzernamen und auf der nächsten Seite den vollständigen Namen verwenden.

Im Fall das auch mit diesen Maßnahmen nicht die gesuchte Person identifiziert werden kann, können mehrere Personenprofile erstellt und angezeigt werde. Der Programm-Anwender kann anschließend aus den vorgeschlagenen Profilen eines auswählen.

6.1.3 Methoden zum Erkennen von wichtigen Informationen auf einer Webseite

Für die Suche nach einer ausgewählten Person können verschiedenste Arten von Webseiten gefunden werden. Aus diesem Grund muss das Programm eine gewisse “Intelligenz“ mit sich bringen um die wichtigsten Daten aus einer Seite herauszufiltern. Dabei ist es nicht möglich eine Hartkodierung zu verwenden, um festgelegte Bereiche einer Webseite auszulesen, da jede Webseite eine individuelle Struktur hat.
Die Grundidee zur Lösung diese Problems ist die Analyse des vorliegenden Webseiten-Textes. Eine Methode zur Textanalyse ist die automatisierte Schlüsselwort-Gewinnung. Hierbei wird die HTML-Seite zu einem verwendbaren Text formatiert, wobei die meisten Sonderzeichen herausgefiltert werden. Sonderzeichen wie “.“ und “@“ werden dabei nicht herausgefiltert, da sie für die E-Mail-Erkennung wichtig sind. Anschließend werden Schlüsselwörter aus dem formatierten Webseitentext generiert. Möglichkeiten zur automatisierten Schlüsselwortgenerierung sind die Verfahren RAKE 6.1.3 und die Automatic Keyword Extraction mit NLP 6.1.3, welche im Laufe dieser Arbeit detailliert beschrieben werden.

Nachdem die Schlüsselwörter generiert und in Listen gespeichert wurden, werden Wortsammlung erstellt. Diese Wortsammlungen sind Listen, welche aussagekräftige Schlüsselwörter enthalten und nach Themen kategorisiert werden. Beispiele für den Inhalt der Listen sind alle Hochschulen und Universitäten in Deutschland, Berufsbezeichnungen und Tätigkeiten, Studiengänge, Hobbybezeichnungen und alle Städte und Gemeinden in Deutschland.
Mit diesen Wortsammlungen kann nun die Liste mit den bereits generierten Schlüsselwörtern aus dem Webseitentext verglichen werden. Bei einer Übereinstimmung eines Schlüsselwortes wird das Wort mit der entsprechenden Kategorie vorgemerkt und später in die verwendete Speicherstruktur eingetragen.
Die Wortsammlungen werden mit Hilfe von bekannten Listen im Internet eigenständig befüllt. Als Informationsquelle dafür, dient jegliche Art von Webseite, die nützliche Information enthält.

RAKE

RAKE steht für *Rapid Automatic Keyword Extraction* und stellt eine sehr effiziente Methode zur Schlüsselwortgenerierung dar. Die Funktion von RAKE basiert darin, dass Schlüsselwörter mehrere Wörter mit inhaltlicher Relevanz enthalten, allerdings selten Stoppwörter und Sonderzeichen. [RECC10]
Als Stoppwörter werden Wörter bezeichnet, die sehr oft auftreten und keinen großen Informationsgewinn mit sich bringen. Beispiele dafür sind *und*, *weil*, *der* oder *als*. [Sla]

In einer jungen Wissenschaft wie der Informatik mit ihrer Vielschichtigkeit und ihrer unüberschaubaren Anwendungsvielfalt ist man oftmals noch bestrebt, eine Charakterisierung des Wesens dieser Wissenschaft und Gemeinsamkeiten und Abgrenzungen zu anderen Wissenschaften zu finden. Etablierte Wissenschaften haben es da leichter, sei es, dass sie es aufgegeben haben, sich zu definieren, oder sei es, dass ihre Struktur und ihre Inhalte allgemein bekannt sind.

Bild 6.1: Beispieltext

Zu Beginn wird der zu analysierende Text, hier der Beispieltext in Bild 6.1, durch einen Worttrenner in ein Array, bestehen aus möglichen Schlüsselwörtern, aufgeteilt. Das erzeug-

te Array wird anschließend in Sequenzen von zusammenhängenden Wörtern unterteilt. Dabei erhalten die Wörter in einer Sequenz die gleiche Position und Reihenfolge wie im Ursprungstext und dienen gemeinsam als Kandidatenschlüsselwort. [RECC10]
Nachdem die möglichen Schlüsselwörter identifiziert sind, wird für jeden einzelnen Kandidaten ein Score ausgerechnet. Dieser besteht aus dem Quotient des Grades $deg(w)$ und der Häufigkeit des Vorkommens eines Wortes innerhalb der Kandidaten $freq(w)$. Daraus ergibt sich die Formel:

$$deg(w)/freq(w)$$

Dabei beschreibt der Grad eines Wortes, dass gemeinsame Auftreten mit sich selbst und anderen Schlüsselwörtern. In der Tabelle 6.1.3 ist der Grad für jedes Wort ablesbar, indem die Einträge in der entsprechenden Reihe summiert werden. Beispielsweise Beträgt der Grad des Wortes *"Wissenschaft"* den Wert *3*. Dies ergibt sich aus der Rechnung:

$$2 + 1 = 3$$

Das Wort *"Wissenschaft"* kommt hier selbst zweimal in dem Kandidaten-Array vor und davon einmal in Verbindung mit dem Worten "jungen".
Die Häufigkeit des Vorkommens eines Wortes lässt sich ebenfalls in der Tabelle 6.1.3 ablesen. Allerdings muss hier in der Reihe und Spalte des jeweiligen Wortes nachgeschaut werden. Für das Wort *"Wissenschaft"* beträgt die Häufigkeit des Vorkommens den Wert *3*.
Zusammenfassend kann gesagt werden, dass *deg(w)* die Kandidaten bevorzugt, welche oft und in langen Schlüsselwörtern, die mehrere Wörter enthalten, vorkommen. Dies bedeutet, dass beispielsweise *deg(etabliert)* eine höhere Bewertung als *deg(informatik)* bekommt, obwohl beide Wörter gleich oft im Text vorkommen. Dagegen wird bei *freq(w)*, ausschließlich die Häufigkeit des Vorkommens bewertet. Bei der Formel *deg(w)/freq(w)* werden die Wörter bevorzugt, welche überwiegend in langen Kandidatenwörtern vorkommen. Diese Formel bietet dadurch einen guten Mittelweg zur Schlüsselwortgewinnung. Ein Beispiel dafür sind die Wörter *"Wissenschaften* und *"allgemein"*. Hier ist der Quotient von *deg(allgemein)/freq(allgemein)* höher als von *deg(Wissenschaften)/freq(Wissenschaften)*, obwohl die Häufigkeit des Wortes *"Wissenschaften"* höher und der Grad gleich hoch ist. [RECC10]

Durch das genannte Verfahren und der Formel *deg(w)/freq(w)* für die Bewertung, ergeben sich die im Bild 6.2 befindenden Kandidaten mit den dazugehörigem Endbewertungen. [RECC10]

	wissenschaften	wissenschaft	sei	etablierte	informatik	aufgegeben	gemeinsamkeiten	oftmals	charakterisierung	jungen	inhalte	allgemein	bekannt	struktur	wesens	bestrebt	unüberschaubaren	anwendungsvielfalt	definieren	abgrenzungen	leichter	finden	vielschichtigkeit
wissenschaften	2			1																			
wissenschaft		2								1													
sei			1																				
etablierte	1			1																			
informatik					1																		
aufgegeben						1																	
gemeinsamkeiten							1																
oftmals								1															
charakterisierung									1														
jungen		1								1													
inhalte											1	1	1										
allgemein											1	1	1										
bekannt											1	1	1										
struktur														1									
wesens															1								
bestrebt																1							
unüberschaubaren																	1	1					
anwendungsvielfalt																	1	1					
definieren																			1				
abgrenzungen																				1			
leichter																					1		
finden																						1	
vielschichtigkeit																							1

Tabelle 6.1: Co-occurance

inhalte allgemein bekannt (9.0), unüberschaubaren anwendungsvielfalt (4.0), jungen wissenschaft(3.5), etablierte wissenschaften (3.5), wissenschaften (1.5), wissenschaft (1.5), wesens (1.0), vielschichtigkeit (1.0), struktur (1.0), sei (1.0), oftmals (1.0), leichter (1.0), informatik (1.0), gemeinsamkeiten (1.0), finden (1.0), definieren (1.0), dass (1.0), charakterisierung (1.0), bestrebt (1.0), aufgegeben (1.0), abgrenzungen (1.0)

Bild 6.2: Schlüsselwörter mit zugehörigem Score

Automatic Keyword Extraction mit NLP

Bei dieser Methode wird der vorliegende Text in die einzelnen Wörter unterteilt. Dabei wird eine Liste mit potentiellen Schlüsselwörtern erstellt, in der *Stoppwörter* und Sonderzeichen herausgefiltert werden. Bei den Schlüsselwörtern handelt es sich nicht ausschließlich um ein Wort sondern auch um Wortsequenzen.
Mit Hilfe von Stemming kann nun die Anzahl der Wörter in der Liste weiter reduziert werden, wodurch eine bessere Schlüsselwortgenerierung möglich ist.
Die Liste mit den möglichen Schlüsselwörtern, kann nach der Häufigkeit des Vorkommens eines Wortes im Text sortiert werden. Das hat den Vorteil, dass die Schlüsselwörter, welche am Häufigsten im Text vorkommen, in den darauf folgenden Schritten zuerst verwendet werden und dadurch eine Laufzeitverbesserung der Anwendung entsteht.
Ergänzende Regeln wie, eine Mindestanzahl von Buchstaben in einem Wort, können die Schlüsselwörter weiter begrenzen.

6.2 Konzept zur Informationsbeschaffung von einer großen Menge unbekannter Personen

Für die *real-world* Simulation eines Phishing-Mail-Angriffs eine Webseiten mit einer großen Menge von personenbezogenen Daten benötigt. Hierfür wird manuell nach einer Webseite gesucht, die eine große Menge an personenbezogenen Daten enthält. Diese wird anschließen als Informationsquelle festgelegt. Möglichkeiten, ausgenommen von den bekannten Social Media Seiten, sind die Webseiten FuPa, Xing und LinkedIn.

6.2.1 Methode zur Suche nach Information

In diesem Konzept gibt es keine automatisierte Suche nach Informationen, jedoch eine automatisierte Suche nach internen Links. Diese interne Suche kann mit einem Web Crawler realisiert werden. In Vorbereitung darauf wird der Aufbau der Seite analysiert.

6.2.2 Methode zum Auslesen der Information

Zum Auslesen einer großen Menge an Daten wird ein Web Scraper erstellt. Dieser könnte für die ausgewählte Webseite hartkodiert werden. Eine Alternative dazu, wäre die Analyse des Webseitentextes, was dem Ansatz 6.1.3 von der Suchfunktion einer ausgewählten Person entsprechen würde.

6.3 Konzept zur Erstellung einer Phishing-Mail

Die Generierung einer Phishing-Mail läuft voll automatisch ab. Das bedeutet, dass das Programm eigenständig die E-Mail-Adressen generiert und selbst passende E-Mail-Muster auswählt.

6.3.1 Methoden zur Generierung von E-Mail-Adressen

Eine Möglichkeit zur Generierung der E-Mail-Adressen kann das Open Source-Tool von Michael Bazzell [Baz] sein, welches mit Hilfe eines automatisierten Webbrowsers verwendet werden kann. Bei diesem Tool werden zuerst über ein Formular, Daten für die E-Mail-Generierung eingetragen. Unter anderem sind das Vorname, Nachname und der E-Mail-Provider. Daraufhin werden die vorgeschlagenen E-Mail-Adressen angezeigt,kopiert und in ein Suchfeld eingefügt. Anschließend kann bei Google, Bing, und Facebook nach Einträgen gesucht und falls ein Eintrag gefunden wurde auch angezeigt werden.

Eine Weitere Möglichkeit wäre ein Algorithmus zu entwickeln, der alle möglichen E-Mail-Adressen aus den Kombinationen von Vorname, Nachname, Geburtsjahr, Benutzernamen und den Domains von den bekanntesten E-Mail-Providern generiert. Dazu gehören *GMX*, *WEB.DE*, *Gmail*, *T-Online*, *Freenet* und *1&1*. [Anb19]
Für den Fall, dass der Arbeitgeber der Zielperson bekannt ist, kann auf der Firmenwebseite nach E-Mail-Adressen gesucht werden. Dadurch ist es möglich die Domain einer Firmen-Mailadresse zu bestimmen und eine Anzahl möglicher Firmenadressen für die Zielperson zu generieren.
Schon bei der Suche von personenbezogenen Daten wird ebenfalls nach E-Mail-Adressen

gesucht. Dadurch kann bereits eine bis jetzt unbekannte Anzahl von Adressen gefunden werden.

6.3.2 Methode zur Erstellung von E-Mail-Mustern

Für die Erstellung der E-Mail-Muster kann eine eigene Klasse erstellt werden, welche für die Erzeugung des Textes zuständig ist. In dieser Klasse werden Strings gespeichert die einem Lückentext ähneln. Abhängig von den gefundenen Daten wird ein Lückentext ausgewählt, welcher anschließend mit den Daten an den passenden Lücken ergänzt wird. Mit dieser Methode muss jedoch für jede Kombination aus gewonnenen Daten ein Lückentext vorhanden sein.
Die Lückentexte werden so kategorisiert, dass für jede gefundene Information ein passender Lückentext vorhanden ist. Eine denkbare Unterteilung wäre in die Kategorien Privat und Geschäftlich.

7 Bewertung der Lösungsideen anhand der Anforderung

Um möglichst viele Informationen über eine Person im Internet zu finden, bietet die Personensuche, welche sich abhängig von den eingegebenen Daten variieren kann, die Lösung mit den meisten Vorteilen. Unter anderem kann die Arbeit des web crawlings ausgelagert werden, da nur noch die Suchergebnisse analysiert werden müssen. Allerdings muss beachtet werden, dass Benutzern bei verschiedensten Social-Media-Seiten auswählen können, ob das Benutzerprofil von einer Suchmaschine gefunden werden kann oder nicht. Aus diesem Grund, werden bei dieser Suchart die Ergebnisse kontrolliert ob sich die geforderten Seiten darin befinden. Wenn das nicht der Fall ist, wird separat auf diesen Seiten nach Information gesucht. Zu den geforderten Seiten zählen beispielsweise *XING* und *LinkedIn*.
Für die Bewertung der Lösungsideen zur Frage, wann es sich um die gesuchte Person handelt in Kapitel 6.1.2, gilt, dass alle Ideen eine Verbesserungen des Ergebnisses mit sich bringen. Allerdings gibt es Unterschied in der Wirksamkeit und in der Laufzeit des Programms. Die Erweiterung der Kriterien 6.1.2 bringt keine große Laufzeitänderung mit sich und stellt eine sehr gute Eigenschaft zur Optimierung der Informationsfindung dar, da die Zeit ebenfalls mit einbezogen wird.

Wenn die Kontakte der Suchperson in Betracht gezogen werden, kann erkannt werden wann es sich um die gesuchte Person handelt. Darüber hin Für die optimal Informationsbeschaffung einer ausgewählten Person eignet sich die Methode der Automatic Keyword Extraction um die Information wird bei der Informationsbeschaffung einer ausgewählten Person der Ansa !!XING kann man angeben ob man durch google gefunden wird!!!

Die Suchfunktion für eine große Anzahl von Personen kann *hartkodiert* werden und benötigt dadurch keine Textanalyse, da der Aufbau der Webseite im voraus bekannt ist. Das bedeutet, dass das Programm genau weiß wo welche Information auf einer Webseite steht. Auf der Seite "*www.fupa.net*" befindet sich beispielsweise der Name einer Person immer an der gleichen Position einer Tabelle. Das bringt den Vorteil mit sich, dass der Text nicht analysiert werden muss und das Programm genau weiß, was mit diesen Daten gemacht werden muss. Zusätzlich entsteht eine sehr performante Methode zur Auslesung von personenbezogenen Daten.

Für die E-Mail-Adressgenerierung wird ein eigener Algorithmus entwickelt. Im Gegensatz zu dem Open Souce-Tool [Baz18] besteht bei diesem Algorithmus eine höhere Wahrscheinlichkeit, dass die richtige E-Mail-Adresse enthalten ist, da das Geburtsjahr, falls es bekannt ist, mit einbezogen wird. Für eine bessere Laufzeit des Programms, wird ein Skript zur Überprüfung der Adressen auf Verfügbarkeit und Gültigkeit, verwendet.

8 Informationsbeschaffung einer ausgewählten Person

8.1 Programmiersprache

Damit das Programm anhand den Lösungsideen umgesetzt werden kann, ist der erste Schritt die Auswahl der Programmiersprache.

Anforderung an das Programm bzw. an die Programmiersprache

Es soll eine möglichst übersichtliche und performante Skriptsprache verwendet werden, mit der eine automatisierte Informationsbeschaffung gut möglich ist. Eine Eingabe über die Konsole oder über eine graphische Benutzeroberfläche soll ebenfalls möglich sein. Aus diesem muss die Programmiersprache keine GUI-Programmierung mit sich bringen.

Lösungsideen für Programmiersprache

Für die Auswahl der Programmiersprache gibt es viele Auswahlmöglichkeiten. Allerdings bringt die Programmiersprache Pyhton, alle Nötigen Eigenschaften mit sich.

Für die Eingabe von Suchdaten, besteht für beide Informationsbeschaffungen die Möglichkeit eine Grafische-Bedienoberfläche oder Konsolen-Eingabe zu verwenden.

Bewertung Programmiersprache

Mit der Programmiersprache Python lässt sich das Programm entsprechend den Anforderungen entwickeln und es kann sowohl eine Konsolenanwendung als auch eine Oberflächenanwendung programmiert werden. Es bringt alle Module mit sich um das Projekt mit dem vorgegebenen Zielen umzusetzen. Außerdem eignet sich Python sehr gut für die Bearbeitung von linguistischen Daten. [BKL09]

8.2 Personensuche mit Hilfe der Google-Suchmaschine im Internet

Für die Personensuche im Internet wird die Google-Suchmaschine verwendet. Gesucht wird nach den eingegebenen Daten. Dafür werden die Daten über eine Konsole eingelesen.

8.2.1 Eingabe der bekannten Daten

Es besteht die Möglichkeit den **Vorname, Nachname, Wohnort, Arbeitgeber, Instagram Benutzername, Facebook Benutzername, Twitter Benutzername**, und das genaue beziehungsweise geschätzte **Geburtsjahr** der gesuchten Person über eine Konsole einzugeben. Falls der genaue Jahrgang der Zielperson nicht bekannt ist, kann ein geschätztes Geburtsjahr eingetragen werden. Dadurch ist es möglich, den groben Zeitraum der Webseite und der Zielperson zu vergleichen.
Zu Beginn werden alle Personen-Variablen mit einem leeren String initialisiert. Das bedeutet dass all die Variablen, zu denen keine Information eingegeben wurde, einen leeren String enthalten.

Verarbeitung der Daten

Zu Beginn der Anwendung werden Abfragen gemacht, um zu erkennen in welchen Variablen sich Information befindet. Anschließend werden mit diesen Variable Kombinationen für die spätere URL-Generierung erstellt. Mögliche Kombinationen für erfolgreiche Suchergebnisse sind:
Vorname, Nachname;
Vorname, Nachname, Wohnort;
Vorname, Nachname, Geburtsjahr;
Vorname, Nachname, Arbeitgeber;
Vorname, Nachname, Benutzername einer Social-Media-Seite;
Vorname, Nachname, Wohnort, Geburtsjahr;
Die Kombination aus vielen oder allen Daten ist ebenfalls eine mögliche Option, allerdings

wird dadurch oft kein Ergebnis gefunden, da nicht zur jeder Information ein Eintrag im Internet ist.
Sobald die Kombinationen aus den Daten bekannt sind, werden die Such-URLs für die Google-Suchmaschine generiert.

8.2.2 Aufbau Google Such-URL

Für den Aufbau eines Google-URLs gilt, dass der URL-Teil "https://www.google.com/search?", bis auf die Protokolle HTTP und HTTPS, gleich bleibt. Des Weitern repräsentieren "%20" ein Leerzeichen und "%22" ein Anführungszeichen.
Hier befindet sich ein Beispiel link:
https://www.google.com/search?q=marco+lang

Such-URL optimieren

Um die Suchergebnisse zu verbessern, können die Suchbegriffe in Anführungszeichen gesetzt werden. Das bedeutet, dass ausschließlich nach diesen Begriffen gesucht wird und nicht nach einer Abwandlung. Ein Beispiel hierfür ist die Suche nach "Mike Bazzell". Wenn diese Suche ohne Anführungszeichen durchgeführt wird, werden Webseiten vorgeschlagen die den Namen Mike Bazzell anstatt Micheal Bazzell beinhalten. Diese erweiterte Suche kann dazu führen, dass unzählige Webseiten vorgeschlagen werden, die nicht unbedingt was mit dem Thema der Suchbegriffe zu tun hat. Um dem vorzubeugen können Anführungzeichen verwendet werden, welche die Anzahl der Suchergebnisse um einen sehr großen Teil verringern wird. [Baz18]
Für die Suche nach **Marco Lang** werden ungefähr **96.400.000** Ergebnisse mit Hilfe der Google-Suchmaschine gefunden. Wird die Suche mit den Anführungszeichen verfeinert, werden für **"Marco" "Lang"** etwa **55.600.000** Ergebnisse gefunden. Allerdings werden hier Webseiten vorgeschlagen, welche die Wörter MMarcoünd "Lang" beinhalten, jedoch müssen diese nicht direkt nebeneinander und auch nicht in der Reihenfolge vorkommen. Es wäre Möglich, dass bei dieser Suche Webseite mit Referenzen auf die Namen "Marco Mustermann" und "Max Lang" beinhaltet. Aus diesem Grund kann nach **"Marco Lang"** gegoogelt werden. Dadurch wird die Anzahl der Suchergebnisse auf **45.500** Ergebnisse

reduziert, da nun die . Die Ergebnisse werden so stark verringert, da Wird nun der Wohnort hinzugefügt, wie in dem Beispiel **“Marco Lang“ Tettnang**, werden lediglich **113** Ergebnisse vorgeschlagen. Der zu dieser Sucheingabe gehörender URL lautet:

https://www.google.com/search?q=%22Marco+Lang%22+Tettnang

Nicht nur die Reduzierung der Suchergebnisse, sondern auch das herausfiltern von unerwünschten Webseiten hat einen positiven Effekt auf diese Arbeit. Die vorgeschlagenen Seiten müssen nämlich in den folgenden Schritten analysiert werden. Das bedeutet, dass jede unerwünschte Seite die allein durch die Suche herausgefiltert werden kann, einen großen Laufzeitvorteil mit sich bringt.

8.2.3 Erstellen des eigenen Such-URLs

In diesem Absatz wird beschrieben wie Google-URLs zur Suche, mit dem Wissen aus Kapitel 8.2.2, erstellt werden.
Für jede genannte Kombination aus den eingegebenen Daten werden Link-Muster erzeugt, die einem Lückentext entsprechen. Sobald die entsprechenden Muster ausgewählt wurden, werden die Lücken mit den Daten befüllt. Dadurch wird eine variierende Anzahl von Suchlinks erstellt.
Wenn allerdings der Benutzername einer Social-Media-Seite bekannt ist, wird ein anderer Aufbau des Suchlinks verwendet, da speziell nach Einträgen auf der entsprechenden Webseite gesucht wird.

URL für beliebige Webseiten

https://www.google.com/search?q=%22Marco+Lang%22+Tettnang

URL für Social-Media-Seiten

https://www.google.com/search?q=site%3Ainstagram.com+%22Lamarcong%22
Probleme mit Facebook

8.2.4 Mit welcher Bibliothek werden Serveranfragen umgesetzt?

Damit eine Person im Internet gesucht werden kann, muss das Programm dazu in der Lage sein, Anfragen an einen Server zu schicken.
Um Anfragen an einen Server zu versenden, kann die Python "request" Bibliothek verwendet werden. Eine Alternative dazu wäre ein automatisierten Web-Browser, welcher mit Hilfe der Selenium Webdriver Bibliothek erstellt werden kann.
Für einfach Anfragen an einen Server eignet sich die request Bibliothek von Python sehr gut. Des Weiteren hat die Bibliothek einen großen Laufzeit-Vorteil gegenüber dem automatisierten Webbrowser. Allerdings lässt sich mit der request Bibliothek keine Javascript-Seite anfordern.
Da Webseiten wie Facebook und Xing Javascript verwenden und diese Seiten elementar für diese Arbeit sind, wird ein automatisierter Webbrowser für die Suche nach einer Person verwendet.

8.2.5 Web Crawler erstellen

Nachdem der automatisierte Browser und die Personensuche implementiert wurde, wird ein Web Crawler benötigt um den, von den Suchmaschinen, vorgeschlagenen Seiten, zu folgen. Dazu muss die Google-Seite mit den Vorschlägen analysiert werden, damit erkannt werden kann wo sich die vorgeschlagenen Links auf der Seite befinden. Diesen Links kann anschließend gefolgt werden.

Googel-Suchseite analysieren

Wie werden Links herausgesucht? Wie werden korrekte links erkannt?

8.3 Die gesuchte Person erkennen

8.3.1 Zeitrahmen wird mit Beachtet

Wie kann Alter der Webseite herausgefunden werden

8.3.2 Kontakte in Betracht ziehen

Auf welcher Seite können mögliche Kontakte gefunden werden

Wie werden Kontakte ausgelesen?

Identifikationsschlüssel erstellen

Was dient als Identifikationsschlüssel

8.4 Herausfiltern von wichtigen Informationen auf einer Webseite

8.4.1 Automatic Keyword Extraction

Schlüsselwortgenerierung mit Python NTLK

Mit dem *Natural Language Toolkit* ist es möglich, den vorhandenen Webseitentext zu analysieren. Zu Beginn können sogenannte “stopwords“ aus dem vorgegebenen Text herausgefiltert werden. Stopwords sind Wörter die sehr oft auftreten und keinen großen Informationsgewinn mit sich bringen. Beispiele dafür sind ist, ein, einer, usw. Dadurch verringert sich die Anzahl der gesamten Wörter im Text um einen sehr großen Teil. Anschließend können Funktionen wie das Zählen des Vorkommens einzelner Wörter angewendet werden, um einen Überblick von dem Text zu bekommen. Des Weiteren kann der Text in Fragmente zerlegt werden um weitere Informationen über den Inhalt zu erlangen.

Abschließend kann eine Liste der analysierten Wörter bzw. Fragmente erstellt werden.
Für die Erkennung wichtiger Schlüsselwörter Es wäre denkbar, Datenbanken bzw. Wortsammlungen zu erstellen, welche die zu suchenden Schlüsselwörter beinhalten. Mit diesen Datenbanken kann nun die Liste mit den bereits verarbeiteten Wörter verglichen werden. Die Datenbanken können mit Hilfe von bekannter Listen im Internet befüllt werden. Beispiele hierfür sind eine aktuelle Liste aller Hochschulen in Deutschland, Berufsbezeichnungen, Studiengänge, Hobbys, Städte und Gemeinden, etc..

8.4.2 Wortsammlungen erstellen

Wie werden Wortsammlungen befüllt?

Wie werden sie am effektivsten verglichen?

8.5 Speicherung der gewonnenen Daten

Die gewonnenen Daten können in einem beliebig erweiterbaren Personen-Objekt gespeichert werden. Darüber hinaus lässt sich das Objekt mit bekannten Kontakten der zu suchenden Person erweitern.
Eine andere Möglichkeit wäre die Daten in eine Datei auszulagern. Hierfür wäre eine Datei mit dem Format *CSV* oder *TXT* möglich.

9 Informationsbeschaffung einer großen Anzahl von Person

9.1 Festlegung einer ausgewählten Webseite

Warum Fupa?

9.2 Aufbau einer Webseite analysieren

9.3 Erstellung eines internen Web Crawlers

Damit die Webseite *www.fupa.net* komplett nach Spielerdaten durchsucht werden kann, wird ein interner Web Crawler benötigt. Dieser wird sich anhand den internen Links, über die ganze Seite hinweg, durchhangeln.
Für die Erstellung eines hartkodierten Web Crawlers muss zuerst einmal der komplette Aufbau einer Webseite bekannt sein. Dies lässt sich einfach mit Hilfe der Entwicklertools in einem Browser durchführen.

9.3.1 Funktionsweise des Web Crawlers

Links mit Spielerinformationen speichern. Die Funktionsweise des Web Crawlers besteht darin, dass das Programm auf der Startseite von Fupa.net beginnt nach links zu suchen und diesen folgt.

9.3.2 Probleme bei der Erstellung

1. Python hat einen verkürzten und erkennbaren Standard http-Header. Dieser wird von vielen Administratoren geblockt und mit der Fehlermeldung 451 erkennbar gemacht. 451 for legal reason
2. Honeypots gewollt oder ungewollt, hier Kalender darstellung mit links zu neuen Jahren die eine sehr hohe bis überhaupt keine Begrenzung haben.
3. Rekursion erreicht schnell die Maximale tiefe von 1500.
4. Zu langsamer Algorithmus

9.3.3 Lösungen

1. http-Header selber konfigurieren
2. Links mit möglichen Honeypots nicht beachten
3. Stack Klasse schreiben damit keine Rekursion benötigt wird
4. Algorithmus anpassen auf fupa-Webseite

9.4 Auslesen der Webseite durch Hartkodierung

9.5 Datenverwaltung und Speicherung

Für die Speicherung der gewonnen Daten kann eine SQL-Datenbank erstellt werden. Als Alternative kann eine Datei angelegt werden, bei der alle Daten zu allen Personen gut strukturiert gespeichert werden können. Eine Möglichkeit dafür wäre das Dateiformat *CSV* oder *TXT*.

10 Erstellung einer Phishing-Mail

10.1 Generierung der E-Mail-Adressen

10.1.1 Funktion des eigenen Algorithmus

10.2 Validität der generierten Mail-Adressen prüfen

10.2.1 Methoden zum Prüfen der Validität

Die erzeugten Adressen werden anschließend auf Validität geprüft. Hierfür gab es früher eine *VRFY* Anfrage von SMTP. Mit dieser Anfrage konnte eine angegebene E-Mail-Adresse überprüft werden. Allerdings wurde der Dienst von Spammern ausgenutzt und wird dadurch von den meisten SMTP-Servern nicht mehr zu Verfügung gestellt. [BPH+10] Demnach muss die Validität auf einem anderen Weg geprüft werden. Eine Möglichkeit zur Prüfung ist die Verwendung bereitgestellter Webseiten, bei der die zu prüfenden E-Mail-Adresse angegeben werden kann. Eine anschließende Rückmeldung verrät dann, ob die Adresse verwendet wird oder nicht. Eine Webseite dafür wäre "*https://centralops.net/co/*". Als Alternative dazu, ist die Entwicklung eines Skriptes, welches die Validität der Adresse prüft.

Im Fall, dass mehrere Adressen von diesem Adresspool gültig sind, kann nach mit Hilfe dieser Mail-Adressen nach Einträgen im Internet gesucht werden. Wenn es eine Übereinstimmung mit der Zielperson gibt, wird diese E-Mail ausgewählt. Andernfalls wird an jede gültige Adresse eine Phishing-Mail gesendet.

10.3 E-Mail-Muster erstellen

10.3.1 Kategorien erstellen

Grundsätzlich können die Muster in zwei große Kategorien unterteilt werden. Es gibt einen privaten und geschäftlichen Teil. Der private Teil hat weiter Unterteilungen wie beispielsweise Familie, Hobby und Interessen. Der Text kann hier in einer Alltagssprache erstellt werden. Für ein geschäftliches Muster sollte eine gehobene Sprache verwendet werden und Daten wie der Firmenname muss bekannt sein.

10.3.2 Lückentexte erstellen

11 Evaluation der Implementation

12 Schlussbemerkungen und Ausblick

12.1 Wie kann eine Person weiter identifiziert werden?

Durch die Google Bildersuche ist es möglich, anstatt einem Suchbegriff ein Bild zu verwenden und nach diesem zu suchen. Dabei kann ein zu suchendes Bild selbst hochgeladen oder ein URL angegeben werden. Bei dem Ergebnis kann es sich um ein ähnliches Bild oder eine Webseite, die das Bild enthält, handeln.
Als Alternative zur Google-Bildersuche kann eine Bilderkennungssoftware verwendet werden um Personen zu identifizieren bzw. zu unterscheiden.

12.2 Keyword Extraction mit Hilfe von Machine Learning

In der Theorie ist es möglich, ein Neuronales Netz mit den Begriffen zu trainieren und eine Kategorisierung durchzuführen. Dabei entsteht ein Netz, welches selbst entscheiden würde, in welche Kategorie ein Wort fällt. Das Wort “Fußball“ müsste dadurch in die Kategorie Hobby eingeordnet werden.

A Ein Kapitel des Anhangs

Glossar

Active Directory

Active Directory ist in einem Windows Server 2000, Windows Server 2003, oder Windows Server 2008-Netzwerk der Verzeichnisdienst, der die zentrale Organisation und Verwaltung aller Netzwerkressourcen erlaubt. Es ermöglicht den Benutzern über eine einzige zentrale Anmeldung den Zugriff auf alle Ressourcen und den Administratoren die zentral organisierte Verwaltung, transparent von der Netzwerktopologie und den eingesetzten Netzwerkprotokollen. Das dafür benötigte Betriebssystem ist entweder Windows Server 2000, Windows Server 2003, oder Windows Server 2008, welches auf dem zentralen Domänencontroller installiert wird. Dieser hält alle Daten des Active Directory vor, wie z.B. Benutzernamen und Kennwörter. 3

Glossareintrag

Erweiterte Informationen zum einem Wort oder einer Abkürzung, ähnlich einem Eintrag im Duden. 3

AbkÃ¼rzungsverzeichnis

AD Active Directory 3

Symbolverzeichnis

π	Die Kreiszahl. 3

Literatur

[All18] Allensbach, IfD: *Meistgenutzte Informationsquellen der Bevoelkerung in Deutschland im Jahr 2018.* https://de.statista.com/statistik/daten/studie/171257/umfrage/normalerweise-genutzte-quelle-fuer-informationen/, 2018. Abrufdatum: 18.01.2019.

[Anb19] *Bei welchem Anbieter haben Sie Ihr Haupt-E-Mail-Postfach?* https://de.statista.com/statistik/daten/studie/170371/umfrage/nutzung-von-e-mail-domains/, 2019. Abrufdatum: 04.02.2019.

[Baz] Bazzell, Michael: *Email Assumptions.* https://inteltechniques.com/osint/email.html. Abrufdatum: 01.02.2019.

[Baz18] Bazzell, Michael: *Open Source Intelligence Techniques: Resources for Searching and Analyzing Online Information.* CreateSpace Independent Publishing Platform, USA, 6th , 2018.

[BKL09] Bird, Steven, Ewan Klein Edward Loper: *Natural language processing with Python: analyzing text with the natural language toolkit.* Ö'Reilly Media, Inc.", 2009.

[BPH+10] Balduzzi, Marco, Christian Platzer, Thorsten Holz, Engin Kirda, Davide Balzarotti Christopher Kruegel: *Abusing social networks for automated user profiling. International Workshop on Recent Advances in Intrusion Detection*, 422–441. Springer, 2010.

[Bun18] Bundeskriminalamt: *Polizeilich erfasste Fälle von Cyberkriminalität im engeren Sinne* in Deutschland von 2004 bis 2017.* https://de.statista.com/statistik/daten/studie/295265/umfrage/polizeilich-erfasste-faelle-von-cyberkriminalität-im-engeren-sinne-in-deuts 2018. Abrufdatum: 29.10.2018.

[Cal13] Caldwell, Tracey: *Spear-phishing: how to spot and mitigate the menace.* Computer Fraud & Security, 2013(1):11–16, 2013.

[CH15] Christopher Hadnagy, Michele Fincher: *Phishing Dark Waters: The Offensive and Defensive Sides of Malicious E-mails*. 2015.

[DSG] DSGVO: *Art. 4 DSGVO Begriffsbestimmungen*. `https://dsgvo-gesetz.de/art-4-dsgvo/`. Abrufdatum: 09.01.2019.

[EAD09] Eldesouki, Mohamed I, W Arafa K Darwish: *Stemming techniques of Arabic language: Comparative study from the information retrieval perspective*. The Egyptian Computer Journal, 36(1):30–49, 2009.

[Fir] FireEye, Inc: *Spear-Phishing-Angriffe ? Warum sie erfolgreich sind und wie sie gestoppt werden knnen*.

[Had11] Hadnagy, Christopher: *Social Engineering: The Art of Human Hacking*. 2011.

[Jam05] James, Lance: *Phshing Exposed: Uncover Secrets from the Dark Side*. 2005.

[Lit16] Litzel, Nico: *Was ist Natural Language Processing?* `https://www.bigdata-insider.de/was-ist-natural-language-processing-a-590102/`, 2016. Abrufdatum: 10.02.2019.

[Mit01] Mitnick, Kevin D.: *The art of deception:controlling the human elemnet of security*. 2001.

[Mit15] Mitchell, Ryan: *Web Scraping with Python: Collecting Data from the Modern Web*. 2015.

[NW18] Nordrhein-Westfalen, Verbraucherzentrale: *Phishing-Radar: Aktuelle Warnungen*. `https://www.verbraucherzentrale.nrw/wissen/digitale-welt/phishingradar/phishingradar-aktuelle-warnungen-6059`, 2018. Abrufdatum: 29.10.2018.

[RECC10] Rose, Stuart, Dave Engel, Nick Cramer Wendy Cowley: *Automatic keyword extraction from individual documents*. Text Mining: Applications and Theory, 1–20, 2010.

[SG12] Sharma, Arvind Kumar PC Gupta: *Study & Analysis of Web Content Mining Tools to Improve Techniques of Web Data Mining*. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 1(8):pp–287, 2012.

[Sla] Slavin, Tim: *Stop Words*. `https://www.kidscodecs.com/stop-words/`. Abrufdatum: 29.01.2019.

[uDsiNe15] Netz e.V., DATEV und Deutschland sicher im: *Verhaltensregeln zum Thema "Social Engineering"*. 2015.