

# AUTOMATISCHE KONTAKTVERVOLLSTÄNDIGUNG

## MOTIVATION

Die Idee für dieses Projekt entstand bei meinem Arbeitgeber POLYPOINT AG, nach dem wir letztes Jahr das Ziel hatten personalisierte Weihnachtskarten an die Entscheidungsträger bei unseren Kunden zu schicken. Praktisch bedeutete dies, dass der CEO eine andere Nachricht als zum Beispiel der CFO oder IT Verantwortliche bekommen hat. Dabei war die grösste Herausforderung, dass wir festgestellt hatten, dass unsere Stammdaten sehr inkomplett und zum Teil auch sehr veraltet sind. So hatten wir einen Rücklauf von fast 10% Prozent, da die Personen nicht mehr in dem Betrieb aktiv waren. Zudem haben wir vor dem Versand einen sehr hohen Aufwand investiert um die Karten überhaupt adressieren zu können.

## ZIELE

Ziel dieser Arbeit soll es sein zu evaluieren, ob eine automatische Stammdatenpflege von Kontaktdaten überhaupt möglich ist. Dabei sollen die geschäftswichtigen Attribute definiert werden und mögliche Quellen aus denen diese bezogen werden können. Zum Test soll einmal evaluiert werden ob Vor- wie auch Nachname und Geschlecht aus einem beliebigen Inputstring (Konkret vermutlich einer E-Mailadresse) korrekt gelabelt werden können. Da der Vor- und Nachname sowie Arbeitgeber im täglichen Business als Identifizierung ausreichen, ist die Vermutung, dass diese als eindeutige Identifizierung einer Person ausreichen. Da es vermutlich eine endliche Anzahl von Namen gibt, ist die Idee ein Lexikon mit Vor- und Nachnamen aufzubauen, mithilfe welchem der vollständige Namen und das Geschlecht gelabelt werden kann.

## DATENSTRUKTUR UND MÖGLICHE QUELLEN

Attribute	Beispiel	Quelle
Firmenname:	Muster AG	div. Firmenregister, Google Maps,...
Kategorie:	Softwareentwicklung	div. Firmenregister, Google Maps,...
Rechtsform:	Akzentgesellschaft	Handelsregister, Aus Name entnehmen
Strasse:	Teststrasse	Google Maps, Webseite Kontaktangaben
PLZ:	1234	Google Maps, Webseite Kontaktangaben
Ort:	Testhausen	Google Maps, Webseite Kontaktangaben
Land:	Schweiz	Google Maps, Webseite Kontaktangaben
Kontaktsprache:	Deutsch	Sprache der Webseite
Domain:	test.com	Aus E-Mailadresse, Suchmaschine
Gründung (wichtig für Liquidität):	1.1.1970	Handelsregister
Eintrag im Handelsregister:	Ja	Handelsregister
Anzahl Mitarbeitende:	100	Webseite über uns
Umsatz:	1 Mio / Jahr	Pressemitteilungen
—	—	—
Geschlecht:	M	Aus Vorname
Titel:	Dr.	Aus Firmenwebseite
Vorname:	Peter	Aus E-Mailadresse, aus Firmenwebseite
Nachname:	Müller	Aus E-Mailadresse, aus Firmenwebseite
Position:	Geschäftsführer	aus Firmenwebseite, Social Network (Xing, LinkedIn)Handelsregister
Unterschriftsberechtigt:	Ja	Handelsregister
Interessen für (B2C):	Sport, Unterhaltung	Social Media

## AUFBAU NAMENSLEXIKON

Da ein Name keinen logischen Aufbau hat, wird es wohl am sinnvollsten sein, diesen mittels eines Lexikons abzugleichen. Es gibt zwar gewisse Muster, wie z.B. dass im deutschen Sprachraum weibliche Namen oft mit A oder E enden. Dies ist aber keine fixe Regel und trifft nicht nur auf Namen zu. Darum muss eine Quelle gesucht werden, welche möglichst viele Namen enthält. Hier würde sich zum Beispiel Wikipedia eignen, um sich die Namen der dort dokumentierten Personen zu verwenden.

### WIKIDATA

Wikidata ist eine Graph-basierte Datenbank, welche die Daten von Wikipedia maschinenlesbar abrufbar machen soll. Queries können direkt unter <https://query.wikidata.org/> ausprobiert werden, mit folgenden Queries wurden die Daten extrahiert und als CSV heruntergeladen.

#### Familienname

```
SELECT ?NAME WHERE {  
  ?N WDT:P31 WD:Q101352.  
  ?N WDT:P1705 ?NAME.  
  ?N WDT:P282 WD:Q8229  
}
```

#### Männliche Namen

```
SELECT ?NAME WHERE {  
  ?N WDT:P31 WD:Q12308941;  
  WDT:P1705 ?NAME;  
  WDT:P282 WD:Q8229.  
}
```

#### Weibliche Namen

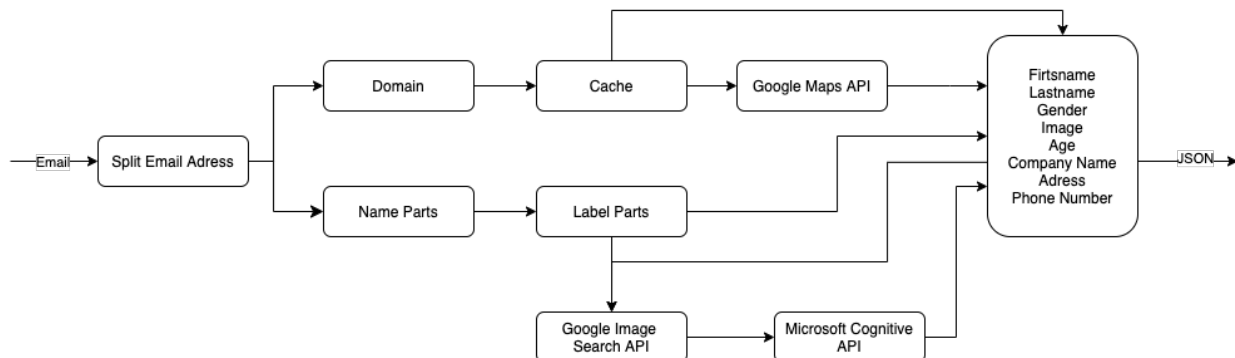
```
SELECT ?NAME WHERE {  
  ?N WDT:P31 WD:Q11879590;  
  WDT:P1705 ?NAME;  
  WDT:P282 WD:Q8229.  
}
```

# PROTOTYP

## ARCHITEKTUR

Um schnell mit dem Prototypen starten zu können und die Hostingkosten tief zu halten wurden vor allem auf mir bereits bekannt Technologien gesetzt. Die API wurde in PHP umgesetzt, das Lexikon in einer mySQL Datenbank gespeichert und das Frontend wurde in JS umgesetzt. Der Prototyp ist unter folgender URL zu finden: <https://leadfinder.ch/email.php>

## ABLAUF




Im ersten Schritt wir geschaut ob es sich beim Inputstring um eine E-Mailadresse oder um ein Name handelt. Enthält der Inputstring ein @ Zeichen wird alles hinter dem @ als Firmendomain gewertet, alles vor dem @ als Name. Im Fall ohne @ wird der ganze String als Name angesehen. Mithilfe der Google Maps API wird geschaut ob es einen Google Maps eintrag gibt zu dieser Domain. Dieser wird als Adresse als JSON encodiert ausgegeben. Der vordere Teil der E-Mailadresse wird mit dem Lexikon verglichen und Vor- und Nachnamen entsprechend klassifiziert. Danach wir nach dem Name in Kombination mit dem Firmennamen nach einem Bild gesucht und der erste Treffer der Microsoft Azure Cognitive API übergeben. Diese analysiert das Gesicht und gibt Geschlecht so wie Alter zurück. All diese Daten werden dann JSON decodiert zurückgegeben. Da es sich bei der Google Maps API um eine Kostenpflichtige API handelt, wurde noch ein Cache vor diese API gebaut, welcher die Firmenadresse in einer Tabelle ablegt um die Anzahl Anfragen zu reduzieren.

Type in an E-Mail in the format (name.name@company.domain)

Input:

Google Image Search:



Microsoft Cognitive API:

Geschlecht:	male	Alter:	~44
Vorname:	Jürgen	M:	undefined
Nachname:	Vogel		German
	<a href="#">more...</a>		

own classification based Lexikon:

Google Maps API:

Firma:	HKB Hochschule der Künste Bern, Berner Fachhochschule BFH		
Segment:	Hochschule in Bern		
Phone:	031 848 38 38		
Adresse:	Fellerstrasse 11, 3027 Bern		
Website:	<a href="http://bfh.ch">bfh.ch</a>		

## ERGEBNISSE

Dieser Prototyp wurde nun mit echten Namen aus unserem CRM getestet. Das Testset enthielt 10'198 Kontaktdaten. Dabei entstanden folgende Werte.

Test	Anzahl erkannt	%
Vorname erkannt:	7133	69.94%
Geschlecht erkannt:	7115	69.76%
Nachname erkannt:	6846	67.13%
Email is available:	9701	95.1%
Format firstname.lastname@company.domain:	7444	72.99%
info@ or support@ adreses:	817	8.01%

## HERAUSFORDERUNGEN

Mit folgenden Inputs gab es noch Probleme oder war die Klassifizierung gar nicht möglich.

- Doppelnamen wie Keller-Sutter
- Namen mit Sonderzeichen. (für ä,ö,ü wurde bereits ein Workaround eingebaut, dass auch nach den jeweiligen Äquivaläten ae, oe, ue gesucht wird)
- Namen mit Leerzeichen wie von Allmen.
- Namen, die sowohl Vor- wie auch Nachnamen sein können (Robert Franz)
- E-Mailadressen, die nicht das Format vorname.nachname aufweisen.
- private E-Mailadressen bei dem die Domain nicht dem Arbeitgeber entspricht

## WEITERES VORGEHEN

- Das Lexikon soll ausgebaut werden. Hierfür wären mögliche Quellen Namen mit Babyseiten oder andere Namensregister.
- Lösung für Doppelnamen
- E-Mail Pattern speichern
- Weitere E-Mail Formate unterstützen
- Domain crawlen für weitere Informationen (Jobtitel)
- LinkedIn crawlen für weitere Informationen (Jobtitel, ist die E-Mailadresse noch aktuell)
- E-Mail Signature parsen
- Massenabgleich via CSV
- API zur Integration ins Endsystem