Automatisierte Telefonnummernrecherche per Webservice mit Probabilistic Record Linkage in Megastudien

Dietrich Alte
Institut für Community Medicine, Abt. SHIP-KEF, Universitätsmedizin Greifswald KdöR
Walther-Rathenau-Str. 48
17475 Greifswald
alte@uni-greifswald.de

Jens Puchert
Institut für Community Medicine, Abt. SHIP-KEF, Universitätsmedizin Greifswald KdöR
Walther-Rathenau-Str. 48
17475 Greifswald
puchertj@uni-greifswald.de

Robert Krüger
Institut für Community Medicine, Abt. SHIP-KEF, Universitätsmedizin Greifswald KdöR
Walther-Rathenau-Str. 48
17475 Greifswald
rkrueger@uni-greifswald.de

Anna Beilfuß
Institut für Community Medicine, Abt. SHIP-KEF, Universitätsmedizin Greifswald KdöR
Walther-Rathenau-Str. 48
17475 Greifswald
beilfussa@uni-greifswald.de

Michael Piontek
Institut für Community Medicine, Abt. SHIP-KEF, Universitätsmedizin Greifswald KdöR
Walther-Rathenau-Str. 48
17475 Greifswald
michael.piontek@uni-greifswald.de

Zusammenfassung

Im Probandenmanagement von Megastudien sind aufwändige Telefonnummern-Recherchen nötig. Zur Vollautomatisierung dieses Prozesses wurde ein SAS Makro entwickelt, dass das Zusammenspiel von SAS mit einem Webservice in Form eines http-Servers ermöglicht, der Abfragen von Telefonnummern auf aktueller Datenbasis erlaubt. Die Recherche ist voll in das Datenmanagement der Adressen integriert. Mit technisch überschaubarem Aufwand kann so die Effizienz der Telefonnummernrecherche deutlich erhöht und die Recherche-Kosten etwa halbiert werden.

Schlüsselwörter: Telefonnummernrecherche, Webservice, PROC HTTP, probabilistic record linkage, Megastudien

1 Einführung

1.1 Hintergrund

In sehr großen epidemiologischen Studien (>100.000 Probanden, genannt „Megastudien“), müssen für die Probandenrekrutierung sehr viele Personen schriftlich und telefonisch kontaktiert werden. Die Einwohnermeldeämter oder (soweit in den Bundesländern vorhanden) Behörden, die über zentrale Meldedatenbestände verfügen, stellen für die Stichprobenziehungen die postalischen Adressen und Geburtsdaten, aber in der Regel keine Telefonnummern zur Verfügung, da diese dort nicht geführt werden. Für die telefonische Kontaktaufnahme müssen die Adressen also um Telefonnummern aus speziellen Telefonnummern-Datenbanken angereichert werden.

1.2 Anwendungsbeispiel & Status Quo

Als Beispiel dient hier der Greifswalder Standort der NAKO Gesundheitsstudie [1]. Bei geplanten 20.000 Probanden und einer Teilnahmebereitschaft von ca. 20% müssen insgesamt ca. 100.000 Adressen angeschrieben und - soweit Telefonnummern auffindbar sind - angerufen werden. In wöchentlichen Wellen werden an jeweils ca. 1000 Adressen Einladungen postalisch verschickt.
Anfangs (Frühjahr 2014 bis August 2015) wurden die Telefonnummernrecherchen mit einer handelsüblichen Telefon-CD manuell umgesetzt. Dies erforderte aufgrund software- und lizenztechnischer Beschränkungen einen hohen Personalaufwand. Es konnten immer nur maximal 70 Adressen in einem Durchlauf abgeglichen werden und es wurden jeweils drei Suchszenarien mit unterschiedlichem Übereinstimmungsgrad der Adressbestandteile durchgespielt. Der Aufwand betrug ca. 8 Personenstunden für die Telefonnummernrecherche bei 1000 Adressen. Dies entspricht Kosten von 0,24€ pro Adresse (bei angenommenem Personalkosten von 30€/h).

1.3 Zielstellung

Dieser sehr hohe Rechercheaufwand sollte deutlich reduziert werden. Die dafür zu entwickelnde Lösung sollte folgende Anforderungen erfüllen:

- hohe Flexibilität, inkl. Parametrisierung für die Details des Matchings der Adressen, d.h. welche Adressbestandteile mit welchem Übereinstimmungsgrad berücksichtigt werden sollten,
- hohe Effizienz, vor allem durch Vollautomatisierung,
- mindestens ähnlich hohe Anzahl von Treffern wie bei der manuellen Methode,
- volle Integration in den Prozess der Stichproben- bzw. Wellenziehung,
- hohe Treffer-Genauigkeit durch den Einsatz einer aktuellen und regelmäßig aktualisierten Telefonnummern-Datenbank.

2 Methodik

2.1 Prozessbeschreibung

Der Prozess der Vorbereitung der Adressen bis hin zur Übergabe an das Probandenmanagementsystem („Stichprobenpipeline“) besteht grob aus folgenden Schritten:

- Stichprobenziehung im Zentralregister,
- Definition der Adressen der nächsten Einladewelle (SAS),
- Datenaufbereitung, Vorbereitung der XML-Files zum Export in die Treuhandstelle (SAS),
- Registrierung, Dublettenkontrolle und ID-Vergabe (in der Treuhandstelle),
- Import in das Probandenmanagement-System (XML).

Der gesamte Prozess (außerhalb der Treuhandstelle) wurde bereits mit SAS umgesetzt. Daher lag es nahe, auch die Telefonnummernrecherche in diesen Prozess einzubinden (Abb. 1).

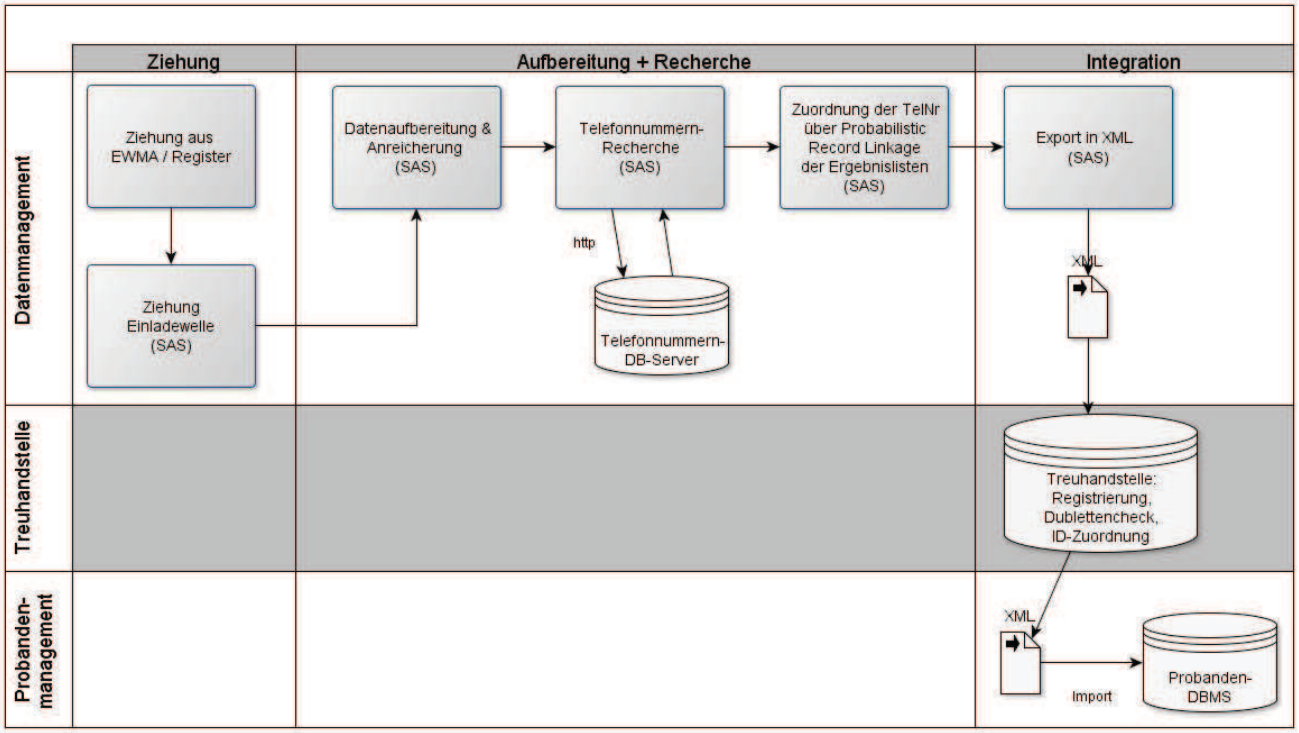

Abbildung 1: Prozessübersicht

2.1 Recherche in der Telefonnummern-Datenbank

Die bislang genutzte, handelsübliche Telefonnummern-CD musste aufgrund der lizenzrechtlichen und softwareseitigen Einschränkungen abgelöst werden. Außerdem war damit keine ausreichende Datenaktualisierung sichergestellt. Als Telefonnummern-Datenbanken entschieden wir uns (auch aus Gründen des Datenschutzes) für eine Web-Serverlösung, bei der die Telefonnummerndatenbank lokal in unserem internen Probandenmanagement-Netz (ohne externen Internetzugriff) installiert werden konnte und

bei der eine quartalsweise Aktualisierung der Daten seitens des Anbieters erfolgt. Das eingesetzte Lizenzmodell lässt maximal 10.000 Abfragen pro Tag zu. Webservices mit häufiger aktualisierten Daten und/oder oder höheren Zugriffszahlen können bei entsprechendem Bedarf auch lizenziert werden. Außerdem bietet der Anbieter die Option, einen online-Webservice statt der lokalen Installation zu nutzen.

Der Zugriff auf den Telefonnummern-Datenbank-Server erfolgt über das http-Protokoll. Der Server nimmt immer nur eine einzelne Abfrage entgegen. SAS produziert im Rahmen einer Makro-Schleife deshalb für jede Adresse einzeln eine Abfrage in XML-Form (s. Anlage A), schickt diese http-Abfrage mit PROC HTTP an den http-Server und nimmt die Antwort als Textfile im XML-Format entgegen. Diese wird mittels einer vorab mit dem SAS-XML-Mapper erstellten XML-Map in eine SAS-Datei eingelesen und enthält für jede Abfrage keine, eine, oder mehrere Ergebnisse. Die Ergebnisse aller Abfragen werden untereinander gehängt (PROC APPEND). Nachfolgend werden die Kernzeilen des Makros verkürzt dargestellt:

```
filename request  "&workpath.\request.xml" encoding="utf-8" lrecl=5000;
filename response "&workpath.\response.xml" lrecl=500000;
filename res_utf8 "&workpath.\response_utf8.xml" encoding="utf-8"
                  lrecl=500000;
filename map    "\\Pfad\XML.map";
libname  res_utf8 xmlv2 xmlmap=map access=READONLY;

* request als Textfile schreiben;
     data _null_;
          set _data (firstobs=%sysevalf(&i.+1) obs=%sysevalf(&i.+1));
          file request;
          put request;
          call symput ('sessionid', &id.);
     run;
* request an Datenbank schicken;
     proc http
          in         = request
          out        = response
          url        = "http://192.168.0.1:8090/"
          method     = "post"
          ct         = "text/xml;
          encoding   = utf-8";
     run;
* http-Server response verarbeiten & Transkodierung in utf-8;;
     data _null_;        infile response encoding="latin1";
          file res_utf8 encoding="utf-8";
          input; put _infile_;
     run;
     proc sql noprint;
          select count into :count from res_utf8.results;
     quit;
* XML-Parsing der Ergebnisse für eine Adresse und Ergebnis an Ergebnisliste anhängen;
%if &count. > 0 %then %do;
```

```
     data _attribute;
          set res_utf8.Attribute(where =(not missing(value)));
          format sessionid 10.;
          sessionID = &sessionid.;
          format requestname $15.;
          requestname ="Name"; * Ergebnisse aus Suche in Name;
     run;
     PROC APPEND BASE=_field data=res_utf8.Field; run;[1]
     PROC APPEND BASE=_all DATA=_attribute force; run;
%end;
```

2.3 Probabilistic Record Linkage

Nach Durchlauf der Abfragen aller Adressen an den http-Server werden für jede Person die Abfrage-Ergebnisse über einen probabilistic record linkage Algorithmus abgeglichen und die Telefonnummer des am besten passenden Treffers übernommen. Der Algorithmus ist an den im Open Source Werkzeug E-PIX (Enterprise Patient Identifier Crossreferencing) [2] eingesetzten Algorithmus angelehnt, der auch in der Treuhandstelle der NAKO für die Dublettenkontrolle eingesetzt wird. Es wird hier eine Mischung aus einem regelbasierten Ansatz und dem klassischen Ansatz von Fellegi & Sunter genutzt [3]. Eine Implementierung des Fellegi-Sunter-Algorithmus in SAS ist bei Winter belegt [4]. Der Linkage Prozess nutzt folgende Schritte, die teilweise in Makros ausgelagert sind:

- **Datenbereinigung, Standardisierung, Normalisierung**

```
* Normierung von Texten;
%macro norm(var);
   *Punktierung entfernen (Compress 'p'),
     alles GROß schreiben (upcase),
     Bindestrich durch Leerzeichen ersetzen (tranwrd);
   &var. = upcase(compress(tranwrd(&var.,'-',' '),,'p'));
   *Umlaute angleichen;
   &var. = prxchange('s/Ä/AE/',-1,&var.);
   &var. = prxchange('s/Ö/OE/',-1,&var.);
   &var. = prxchange('s/Ü/UE/',-1,&var.);
   *ß in SS umwandeln;
   &var. = prxchange('s/ß/SS/',-1,&var.);
   *É in E umwandeln;
   &var. = prxchange('s/É/E/',-1,&var.);
   *Á in A umwandeln;
   &var = prxchange('s/Á/A/',-1,&var.);
   *Ó in O umwandeln;
   &var = prxchange('s/Ó/O/',-1,&var.);
%mend;
* Normierung der Strasse;
%macro normstr(var);
   &var. = prxchange('s/STRASSE/STR/',-1,&var.);
```

[1] * alle Fields zusammensuchen, falls neue auftauchen;

```
   &var. = prxchange('s/STRASE/STR/',-1,&var.);
   &var. = prxchange('s/STR./STR/',-1,&var.);
   &var. = compress(&var.); * alles Leerzeichen raus;
%mend;
```

- **Blocking**
  (hier: nur direkter Abgleich der Ergebnis-Adressen aus der Telefonnummernrecherche, mit der Adresse, nach der auch gesucht wurde)
- **Kalkulation von Ähnlichkeitsscores**
  (hier: Text-Distanz-Maß als Levenshtein-Distanz mit der Funktion `complev`)

```
%macro dist(varname, schwellwert, gewicht);
   Lev_&varname.   = complev(&varname._1, &varname._2);
   P_&varname.     = 1 - Lev_&varname. /
       max(length(&varname._1), length(&varname._2));
   if P_&varname. GE &schwellwert. then M_&varname. = 1;
       else M_&varname.=0;
   if M_&varname.=1 then do;
      PM_&varname. = &gewicht. * P_&varname.;
      PU_&varname.=0;
   end;
   else do;
      PM_&varname.=0;
      PU_&varname. = &gewicht. * (1-P_&varname.);
   end;
%mend;
```

- **Regelbasierte Entscheidung**
  Passt die Adresse der Telefonnummer zur gesuchten Adresse oder nicht? Sofern mehrere Adressen aus der Telefonnummernrecherche passen, wird die mit dem höchsten Ähnlichkeitsmaß gewählt und die entsprechende Telefonnummer der Ausgangsadresse zugeordnet.

3 Ergebnisse

3.1 Leistungsfähigkeit & Effizienz

Der bisherige manuelle Recherche-Aufwand für Telefonnummern konnte mit der Implementierung komplett durch eine Vollautomatisierung ersetzt werden. Dadurch wurde die Geschwindigkeit in der Stichprobenpipeline deutlich erhöht und das Personal für den direkten Probandenkontakt, das Telefonieren mit den Probanden freigesetzt.
Bei der Ziehung der Adressen für die nächste Einladewelle werden die Telefonnummern vollautomatisch per Datenbankabfrage hinzugefügt. Die Nummernrecherche für 1000 Adressen dauert ca. 5 min. Wie vorher beim manuellen Vorgehen finden wir für knapp 30% der Adressen passende Telefonnummern. Durch den regelmäßig aktualisierten Datenbestand der Telefonnummern-Datenbank ist eine gleichbleibend gute Trefferquote gewährleistet.

Zum Schluss der Recherche wird eine knappe Statistik über den Erfolg der Recherche als Kontrolle ausgegeben. Diese gibt möglicherweise Hinweise auf Probleme beim Matching oder über regionale Besonderheiten. Z. B. sind durch Gebietsänderungen Umbenennungen von PLZ, Ort und/oder Straßen und Hausnummern möglich, die ggf. zu einer reduzierten Trefferquote in gewissen Gebieten führen können, sofern die Datenbestände nicht bzgl. der Gebietsänderungen denselben Stand haben.

3.2 Flexibilität

Die Abfragen der Telefonnummerndatenbank und auch der probabilistic record linkage Algorithmus, d.h. welche Adressbestandteile mit welchem Übereinstimmungsgrad berücksichtigt werden, können für lokale Anforderungen mit diversen Parametern angepasst und optimiert werden. Damit kann die Spezifität und Sensitivität der Nummern-Suche flexibel gesteuert werden.

3.3 Integration

Die Telefonnummernrecherche ist komplett als SAS Makro umgesetzt und ist mit einem einfachen Makroaufruf in den Prozess der Stichproben- bzw. Wellenziehung voll integriert:

```
%telrecherche (data=Adressen, out=Adressen_mit_TelNr, ID=id);
```

3.4 Aufwand, Kosten

Die Entwicklung der SAS-Skripte verursachte einen Programmieraufwand von ca. 45h. Hinzu kommen die einmaligen organisatorischen Maßnahmen für die Lizenzierung und Installation der Telefonnummern-Datenbank. Insgesamt kann von einem Aufwand von ca. 60h ausgegangen werden. Die Kosten für die Lizenz des Telefonnummern-Datenbankservers inkl. der Updates belaufen sich auf ca. 2.400€ pro Jahr. Bei einer kompletten Umlage der Entwicklungskosten (60h x 50€/h = 3.000€) auf 100.000 Adressen und einer angenommenen Studien-Laufzeit von 4 Jahren (4 Jahre x 2400€ = 9.600€) entstehen Gesamtkosten von ca. 12.600€ bzw. 0,126€ pro Adresse, was nur ca. 50% der Kosten der bisherigen manuellen Recherche beträgt.

4 Schlussfolgerungen

Das Zusammenspiel von SAS mit einem (hier lokal installierten) Webservice in Form eines http-Servers, der Abfragen von Telefonnummern auf aktueller Datenbasis erlaubt, kann technisch mit überschaubarem Aufwand umgesetzt werden und kann die Effizienz im Probandenmanagement in Megastudien deutlich erhöhen.

Literatur

[1] German National Cohort (GNC) Consortium (2014): The German National Cohort: aims, study design and organization. Eur J Epidemiol. 2014 May;29(5):371-82. doi: 10.1007/s10654-014-9890-7.

[2] ID-Management mittels E-PIX (2015), https://mosaic-greifswald.de/werkzeuge-und-vorlagen/id-management-e-pix.html, Zugriff 18.09.2015.

[3] Ivan P. Fellegi & Alan B. Sunter, 'A Theory for Record Linkage', Journal of the American Statistical Association 64 (1969): 1183–1210. http://www.jstor.org/stable/2286061, Zugriff 04.02.2016.

[4] Glenn Wright (2011) Probabilistic Record Linkage in SAS, Western Users of SAS Conference, http://www.wuss.org/proceedings11/Papers_Wright_G_76128.pdf, Zugriff 04.02.2016.

Anlage A: http request im XML-Format

```
data _data; set &data. (keep = id Name Vorname StrasseNr Strasse
Hausnummer_num Zusatz Postleitzahl Wohnort);
length request request2 $ 5000; format _numeric_;
* XML-request in Text-Variable schreiben;
request=catx(
"", '<?xml version="1.0" encoding="utf-8"?>
<Envelope><Header><SessionID>',
&id. ,'</SessionID></Header><Body><GetSelect DB="Addresses">
<Select><SelFields>',
     '<SelField ID="Name">', Name, '</SelField>',
     '<SelField ID="ZIP_Code">', Postleitzahl, '</SelField>',
'</SelFields>',
'<Params>',
   '<Param ID="Name">fuzzy</Param>''</Params>',
'</Select><Results><ResFields>','
     <ResField ID="Address" />     <ResField ID="First_Name" />
     <ResField ID="Last_Name" />   <ResField ID="Add_Name" />
     <ResField ID="Title" />       <ResField ID="Prefix_Name" />
     <ResField ID="Trailer_Name" /><ResField ID="Street" />
     <ResField ID="Street_No" />   <ResField ID="Street_No_Add" />
     <ResField ID="ZIP_Code" />    <ResField ID="City" />
     <ResField ID="City_Plain" /> <ResField ID="Phone_Area_Code" />
     <ResField ID="Phone_No" />    <ResField ID="Phone_No_Add" />
     <ResField ID="TNA" />         <ResField ID="Email" />
</ResFields>',
'<Start>0</Start>
<End>10000</End>
<Sort>Last_Name</Sort>
<Max>10000</Max>
</Results></GetSelect>
</Body></Envelope>');
run;
```