Stand: 03.03.2019

# Dokumentation des Fragebogens zur Berufserfassung

Der Aufruf des Fragebogens erfolgt über

<http://ec2-35-157-97-32.eu-central-1.compute.amazonaws.com/shiny/rstudio/beruf/?id=ID&pwd=PWD>

Für ID kann ein beliebiger Text eingegeben werden. Diese ID wird gemeinsam mit den erhobenen Daten gespeichert und dient der späteren Verknüpfung der Daten.

Für PWD muss ein zuvor festgelegtes Passwort eingegeben werden.

## Weitere Parameter

Mit den folgenden optionalen Parametern lässt sich das Aussehen des Tools steuern:

&suggestions.type=[*aux* | *aux.labels* | *kldb*]

Mit *aux* (default) werden Tätigkeiten aus der Hilfsklassifikation angezeigt. Mit *aux.labels* werden Berufsbenennungen angezeigt. Mit *kldb* werden die Bezeichnungen der KldB-3-Steller angezeigt.

*&berufstyp=[aktuell | vergangen]*

Sollen in der Präsens-Form nach dem aktuellen Beruf gefragt werden oder soll in einer Vergangenheitsform nach einem vorherigen Beruf gefragt werden? Default: aktuell

&conversational=*[on / off]*

Wenn *conversational=on* (default) werden zusätzlich zu den Tätigkeiten auch Job Titles und Tätigkeitsbeschreibungen angezeigt.

&numResponses=<int>

Anzahl der maximal vorgeschlagenen Tätigkeiten (Default: 5). An zwei Stellen wird im Programmcode die Anzahl der Vorschläge begrenzt, nämlich in den Funktionen *job\_coding* und *make.suggestions*. *numResponses* bezieht sich bisher bloß auf die Funktion *make.suggestions*, in der mithilfe von Abgrenzungen die finalen Vorschläge ermittelt werden.

*&askFollowUp=anforderungsniveau;aufsicht;spezialisierung;sonstige*

Welche Art von Folgefrage soll im Tool gestellt werden (default: keine)? Mit diesem Parameter lässt sich vermeiden, dass Fragen doppelt gestellt werden falls das Befragungsinstitut einzelne Folgefragen in die eigene Software einbinden möchte. Wenn man beispielsweise nur sonstige Folgefragen verwenden möchte, lautet der Befehl: *&askFollowUp=sonstige.* Wenn man alle vier verfügbaren Arten von Folgefragen verwenden möchte, muss der Befehl wie oben angegeben lauten.

&use.dictionary=*true*

Standardmäßig ausgeschaltet. Falls *use.dictionary=TRUE* wird nach einer gleichlautenden Berufsbezeichnung im Dictionary gesucht und diese wird abgespeichert. Im Erfolgsfall werden keine Vorschläge aus der Hilfsklassifikation/KldB angezeigt. Folgefragen in Abhängigkeit von der Berufsbezeichnung sind bisher nicht implementiert.

&audiorecording=[*on* | *off*]

Ist Audiorecording derzeit ein- oder ausgeschaltet? (Dient als Filter für mögliche weitere Fragen.)

&verbose=TRUE

Zum Debugging können ausführliche Zwischenergebnisse angezeigt werden.

## Datenbankdokumentation Ergebnisse

Die nachfolgenden Informationen gelten hauptsächlich für suggestions.type=aux (default).

Die Ergebnisse werden in den folgenden vier Tabellen abgespeichert. Der angegebene Variablentyp wird so üblicherweise nicht verwendet, da die gespeicherten Daten nach dem AES-Standard verschlüsselt sind.

### Tabelle **resultOverview**

Am Ende eines jeden Aufrufs des Fragebogens wird eine Zeile zur Tabelle hinzugefügt.

`sessionId` ist der eindeutige Tabellenschlüssel.

`userId` VARCHAR(30): Über die URL bereitgestellte ID

`sessionId` VARCHAR(41): Eindeutiger Tabellenschlüssel. Format: `userId`\_TIMESTAMP\_ZUFALLSZAH

`berufTaetigkeitText` VARCHAR(2500): Welche berufliche Tätigkeit üben Sie derzeit hauptsächl. aus?

`dictionaryCodedTitle` VARCHAR(1500): Eintrag aus Dictionary (nur falls *use.dictionary=TRUE)*

`auswahlHilfsklassifikation` INT: ID der ausgewählten Kategorie aus der Hilfsklassifikation

`auswahlHilfsklassifikationText` VARCHAR(2500): zugehörige Bezeichnung

`kldb` CHAR(5): KldB 2010-Kategorie berechnet auf Basis der Hilfskategorie und der Folgefragen

`isco` CHAR(4): ISCO-08-Kategorie berechnet auf Basis der Hilfskategorie und Folgefragen

`followUp1Question` VARCHAR(255): Fragetext der ersten Folgefrage

`followUp1Answer` VARCHAR(250): Ausgewählte Antwort zur ersten Folgefrage

`followUp2Question` VARCHAR(255): Fragetext der zweiten Folgefrage

`followUp2Answer` VARCHAR(250): Ausgewählte Antwort zur zweiten Folgefrage

`auswahlKldbDreistellig` CHAR(3): Ausgewählter Dreisteller (nur falls *suggestions.type=kldb*)

`berufTaetigkeitText2` VARCHAR(3500): Bitte beschreiben Sie mir diese berufliche Tätigkeit genau.

`urlQueryString` VARCHAR(255): Parameter, die als Teil der URL übergeben wurden

### Tabelle **answerSubmitted**

Nach jeder abgeschickten Antwort (=nach jedem Klick auf `Weiter`) wird eine Zeile zur Tabelle hinzugefügt. Wenn auf einer Seite keine Informationen abgefragt wurden, wird nichts gespeichert.

Im Gegensatz zur Tabelle **resultOverview** werden zusätzliche Informationen erfasst, z.B. veraltete Antworten, die später noch einmal korrigiert wurden und Antwortzeiten.

`sessionId` zusammen mit `questId` ist der eindeutige Tabellenschlüssel.

`userId` VARCHAR(30): Über die URL bereitgestellte ID

`sessionId` VARCHAR(41): Eindeutiger Tabellenschlüssel. Format:`userId`\_TIMESTAMP\_ZUFALLSZAH

`questId` VARCHAR(50): ID der jeweiligen Frage

`fragetext` VARCHAR(255): Fragetext der jeweiligen Frage

`antwortId` INT: ID der vom Befragten ausgewählten Antwort

`antwortText` VARCHAR(2500): Text der vom Befragten ausgewählten Antwort

`textNoneSelected` VARCHAR(1500): Zusätzlich eingegebener Text bei halboffenen Fragen

`start` DATETIME: Zeitpunkt, wann die Frage auf dem Bildschirm angezeigt wurde

`end` DATETIME: Zeitpunkt, wann die Frage mittels Klick auf “Weiter” abgeschlossen wurde.

`dictionaryCodedTitle` VARCHAR(1500): Eintrag aus Dictionary (nur falls *use.dictionary=TRUE)*

`kldbDefault` VARCHAR(15): KldB 2010-Kategorie (Default-Wert für die ausgewählte Antwort)

`iscoDefault` VARCHAR(13): ISCO-08-Kategorie (Default-Wert für die ausgewählte Antwort)

`followUp` CHAR(5): FALSE, Ist eine zweite Folgefrage erforderlich? Details in Hilfsklassifikation

### Tabelle **occupationsSuggested**

Die Tabelle enthält alle Antworten aus der Hilfsklassifikation, die dem Befragten zur Auswahl vorgeschlagen wurden. Wenn der Befragte eine Antwort aus der Hilfsklassifikation auswählt (in **answerSubmitted** wird eine neue Zeile mit `questId = auswahl\_hilfsklassifikation` erzeugt) werden zeitgleich mehrere Zeilen mit den vorgeschlagenen Antwortmöglichkeiten in **occupationsSuggested** hinunzugefügt.

`sessionId`, `start` und `antwortId` bilden zusammen den eindeutigen Tabellenschlüssel. Eine Verknüpfung mit **occupationsSuggested** ist anhand der beiden Variablen `sessionId` und `start` möglich.

`sessionId` VARCHAR(41): Eindeutiger Tabellenschlüssel. Format:`userId`\_TIMESTAMP\_ZUFALLSZAH

`start` DATETIME: Zeitpunkt, wann die Frage auf dem Bildschirm angezeigt wurde

`antwortId` INT: ID der zur Auswahl vorgeschlagenen Hilfskategorie

`predProb` DOUBLE: berechnete Wahrscheinlichkeit der vorgeschlagenen Antwort

`orderIndicator` INT: Reihenfolge der Antworte: order(orderIndicator, predProb, decreasing = TRUE)

`berufsbezeichnung` VARCHAR(800): Berufsbezeichnung der zur Auswahl vorgeschlagenen Hilfskate

`antwortText` VARCHAR(2500): Taetigkeit der zur Auswahl vorgeschlagenen Hilfskategorie

`taetigkeitsbeschreibung` VARCHAR(2500): Taetigkeitsbeschreibung der zur Auswahl vorgeschlagen

`kldbDefault` VARCHAR(15): KldB 2010-Kategorie (Default-Wert für die vorgeschlagene Antwort)

`iscoDefault` VARCHAR(13): ISCO-08-Kategorie (Default-Wert für die vorgeschlagene Antwort)

### Tabelle **toggleSubmitted**

Nach jedem Klick um für einzelne Antworten Detailinformationen (Tätigkeitsbeschreibungen) anzuzeigen/auszublenden, wird eine Zeile zur Tabelle hinzugefügt.

`sessionId` und `time` bilden zusammen den eindeutigen Tabellenschlüssel. Eine 1:n Verknüpfung mit einer Zeile aus **answerSubmitted** ist möglich bei identischer `sessionId` und zusätzlich answerSubmitted$start < toggleSubmitted$time < answerSubmitted$end

`sessionId` VARCHAR(41): Eindeutiger Tabellenschlüssel. Format:`userId`\_TIMESTAMP\_ZUFALLSZAH

`toggleMessage` CHAR(20): siehe unten

`time` DATETIME: Zeitpunkt 300 Millisekunden nachdem der Nutzer geklickt hat

`toggleMessage` hat folgendes Format:

toggle-pos-4-true besagt, dass der Text, der dem *vierten* Vorschlag zugeordnet ist, *aufgeklappt* wurde.

toggle-pos-2-false besagt, dass der Text, der dem *zweiten* Vorschlag zugeordnet ist, *zugeklappt* wurde.

## Zugriff auf Datenbanken aus R

Der folgende R-Code enthält grundlegende Befehle zum Zugriff auf die Datenbank mit den oben beschriebenen Tabellen.

library(DBI)

library(RMySQL)

encryptKey <- "secretKey"

con <- dbConnect(MySQL(), user = "user\_name", password = "PWD", host = "host", dbname = "dbname")

dbExecute(con, "SET block\_encryption\_mode = 'secret’;")

# Variablennamen

answerSubmittedColnames <- c("userId", "sessionId", "questId", "fragetext", "antwortId", "antwortText", "textNoneSelected", "start", "end", "dictionaryCodedTitle", "kldbDefault", "iscoDefault", "followUp")

resultOverviewColnames <- c("userId", "sessionId", "berufTaetigkeitText", "dictionaryCodedTitle", "auswahlHilfsklassifikation", "auswahlHilfsklassifikationText", "kldb", "isco", "followUp1Question", "followUp1Answer", "followUp2Question", "followUp2Answer", "auswahlKldbDreistellig", "berufTaetigkeitText2", "urlQueryString")

occupationsSuggestedColnames <- c("sessionId", "start", "antwortId", "predProb", "orderIndicator", "berufsbezeichnung", "antwortText", "taetigkeitsbeschreibung", "kldbDefault", "iscoDefault")

# Zusätzliche Variablennamen speziell für die Befragung mit Infas (erfordert andere Zugangsdaten)

resultOverviewColnames <- c(resultOverviewColnames, c("berufTaetigkeitKontrolltext", "interviewerSchwierigkeiten1", "interviewerSchwierigkeiten1a", "interviewerSchwierigkeiten2Id", "interviewerSchwierigkeiten2Text", "befragterSchwierigkeiten1", "befragterUebereinstimmung", "warumEtwasAnderesId", "warumEtwasAnderesText"))

# Datenbankabfragen

answerSubmitted <- dbGetQuery(con, paste0("SELECT ", paste0("AES\_DECRYPT(`", answerSubmittedColnames, "`, '", encryptKey, "') AS ", answerSubmittedColnames, collapse = ", "), " FROM answerSubmitted"))

resultOverview <- dbGetQuery(con, paste0("SELECT ", paste0("AES\_DECRYPT(`", resultOverviewColnames, "`, '", encryptKey, "') AS ", resultOverviewColnames, collapse = ", "), " FROM resultOverview"))

occupationsSuggested <- dbGetQuery(con, paste0("SELECT ", paste0("AES\_DECRYPT(`", occupationsSuggestedColnames, "`, '", encryptKey, "') AS ", occupationsSuggestedColnames, collapse = ", "), " FROM occupationsSuggested"))

toggleSubmitted <- dbGetQuery(con, sqlInterpolate(con, "SELECT \* FROM toggleSubmitted"))

# Datenbankverbindung schliessen

dbDisconnect(con)

# Falls Umlaute falsch dargestellt werden, lässt sich dies jeweils korrigieren mithilfe

Encoding(answerSubmitted$fragetext) <- "UTF-8"

#############################

# Nur wenn Daten nicht verschlüsselt abgespeichert würden, würden auch folgende Befehle funktionieren.

#############################

# Datenbankabfragen

# resultOverview <- dbGetQuery(con, sqlInterpolate(con, "SELECT \* FROM resultOverview"))

# mittels WHERE userId = 35 können wir auf eine ausgewählte userId zugreifen

# resultOverview <- dbGetQuery(con, sqlInterpolate(con, "SELECT \* FROM resultOverview WHERE userId = ‘35’"))

# ORDER BY erlaubt Sortierung

# answerSubmitted <- dbGetQuery(con, sqlInterpolate(con, "SELECT \* FROM answerSubmitted ORDER BY start"))

# occupationsSuggested <- dbGetQuery(con, sqlInterpolate(con, "SELECT \* FROM occupationsSuggested ORDER BY start"))

# toggleSubmitted <- dbGetQuery(con, sqlInterpolate(con, "SELECT \* FROM toggleSubmitted"))

## Hinweise zur Berechnung der Wahrscheinlichkeiten und Reihenfolge der Anzeige

(nur gültig bei &suggestions.type=“aux“), Stand vom 24.01.2019

Wir wissen, dass unsere Wahrscheinlichkeiten schlecht kalibriert sind, also möglicherweise nicht uneingeschränkt als Wahrscheinlichkeiten interpretiert werden können.

Startpunkt: wir haben eine Tabelle mit KldB-IDs und zugeordneten Wahrscheinlichkeiten, die mittels similarity-based Reasoning (wordwise string similarity) berechnet wurden.

1. Merge KldB-ID mit IDs aus der Hilfsklassifikation.
2. Berechne Wahrscheinlichkeiten für Kategorien aus der Hilfsklassifikation:
   1. Wenn eine KldB-ID mehreren Hilfskategorien zugeordnet ist, teile die verfügbare Wahrscheinlichkeitsmasse der KldB auf die Hilfskategorien gleichmäßig auf.
   2. Wenn mehrere KldB-IDs zur gleichen Hilfskategorie führen, addiere die Wahrscheinlichkeiten zusammen.
3. Behalte die 5 wahrscheinlichsten Hilfskategorien (und nur wenn ihre Wahrscheinlichkeit > 0.005 ist)
4. Prüfe ob Vorschläge angezeigt werden oder ob alle Vorschläge zu schlecht sind.
   1. Falls wordwise string similarity verwendet wurde: Wenn die Summe der 5 wahrscheinlichsten Hilfskategorien eine Wahrscheinlichkeit > 0.535 hat, fahre mit Schritt 5 fort, andernfalls wiederhole die Schritte 1-4 unter Verwendung von Wahrscheinlichkeiten, die mittels substring similarity berechnet wurden.
   2. Falls substring similarity verwendet wurde: Wenn die Summe der 5 wahrscheinlichsten Hilfskategorien eine Wahrscheinlichkeit > 0.618 hat, fahre mit Schritt 5 fort, andernfalls werden keine Vorschläge erzeugt.
5. Verwende die Abgrenzungen aus der Hilfsklassifikation um weitere ähnliche Hilfskategorien vorzuschlagen
   1. Für Hilfskategorien, deren Wahrscheinlichkeit > 0.2 ist: Füge diejenigen Hilfskategorien hinzu, die abgrenzung = „hoch“ haben. Setze die Wahrscheinlichkeit der neu hinzugefügten Kategorien auf 0.05
   2. Für Hilfskategorien, deren Wahrscheinlichkeit > 0.8 ist: Füge diejenigen Hilfskategorien hinzu, die abgrenzung = „mittel“ haben. Setze die Wahrscheinlichkeit der neu hinzugefügten Kategorien auf 0.005
   3. Aufgrund der hinzugefügten Hilfskategorien kann eine Hilfskategorie mehrfach vorkommen. Jede Hilfskategorie wird nur ein einziges Mal beibehalten, wobei die Wahrscheinlichkeiten aufaddiert werden. (Wenn also eine Hilfskategorie 10 Mal aufgrund Abgrenzung = „hoch“ hinzugefügt wird, ist die neue Wskt. = alte Wskt. Aus Schritt 2 + 10 \* 0.05.)
6. Sortiere Hilfskategorien in absteigender Reihenfolge nach (order.indicator und Wahrscheinlichkeit).
   1. Hilfskategorien mit Wahrscheinlichkeit > 0.5 (Wskt. Aus Schritt 2 ohne die Neuberechnung in Schritt 5) haben order.indicator = 2 und stehen damit immer vorne. In diesem Fall wird der order.indicator auf 1 gesetzt für alle anderen Hilfskategorien ausgenommen derer, die aufgrund von abgrenzung = „mittel“ hinzugefügt wurden. Dies führt dazu, dass hohe Abgrenzungen der äußerst wahrscheinlichen Kategorie vor den übrigen Vorschlägen vom Algorithmus stehen können wenn diese eine Wahrscheinlichkeit < 0.05 haben.
   2. Wenn keine Hilfskategorie eine Wahrscheinlichkeit > 0.5 ist der order.indicator bei allen Kategorien = 0 und wird daher ignoriert.
   3. Neu hinzugefügte Abgrenzungen vom Typ = „mittel“ haben eine Wahrscheinlichkeit von 0.005 (kleinstmöglich, siehe Punkt 3) und stehen damit immer am Ende.
7. Schlage die 5 Hilfskategorien vor, die nach der Sortierung am weitesten oben stehen.
   1. (In einer zukünftigen Studie könnte man sich überlegen, dieses deterministische Verfahren zu ersetzen durch irgendeine Art von Zufallsziehung).