SSOAR

Open Access Repository
www.ssoar.info

Computerunterstützte Suche formelhafter Rede

Schindele, Martin

Veröffentlichungsversion / Published Version
Zeitschriftenartikel / journal article

Zur Verfügung gestellt in Kooperation mit / provided in cooperation with:
GESIS - Leibniz-Institut für Sozialwissenschaften

Empfohlene Zitierung / Suggested Citation:
Schindele, M. (1994). Computerunterstützte Suche formelhafter Rede. *Historical Social Research*, *19*(3), 156-163. https://doi.org/10.12759/hsr.19.1994.3.156-163

Nutzungsbedingungen:
Dieser Text wird unter einer CC BY Lizenz (Namensnennung) zur Verfügung gestellt. Nähere Auskünfte zu den CC-Lizenzen finden Sie hier:
https://creativecommons.org/licenses/by/4.0/deed.de

Terms of use:
This document is made available under a CC BY Licence (Attribution). For more Information see:
https://creativecommons.org/licenses/by/4.0

gesis Leibniz-Institut für Sozialwissenschaften

Mitglied der Leibniz-Gemeinschaft

Diese Version ist zitierbar unter / This version is citable under:
https://nbn-resolving.org/urn:nbn:de:0168-ssoar-51182

Historical Social Research, Vol. 19 — 1994 — No. 3, 156-163

HUMANITIES COMPUTING

Computerunterstützte Suche formelhafter Rede

*Martin Schindele (Tübingen)**

0. Motivation

0.1 Arbeitsbegriff. - Unter formelhafter Rede verstehen wir die Verwendung „mehr oder minder starrer" Wortfolgen, denen neben ihrer unmittelbaren Wortbedeutung eine textpragmatische Funktion zukommt.

0.2 Beispiel. - Die Wortfolge „Im Namen des Volkes" kennzeichnet Gerichtsurteile.

0.3 Aufgabenstellung. — Es wird eine Konzeption und ein Programm vorgestellt, mit der zu einem gegebenen Text alle Belegstellen für „gewisse Wortfolgen" bis zu einer gewissen „Streuung" in einem anderen, grösseren Text gefunden werden. Als Belegstelle gilt zunächst eine Stelle, an der die Wortfolge vorkommt, wobei „gewisse Abstandsverhältnisse" gewahrt sind. Diese Aufgabenstellung wird in 2. präzisiert.

0.4 Beispiel für maschinell erzeugte Ergebnisse. — Wortfolgen aus Gen 3,4 und ihre Belegstellen im AT:

(1;7;7;1,1,1,1,1,1,1) Gen 3,4; *wayyo'mär hannåḥåš 'äl hå'iššåh lo' mowt tmutuwn*
Gen 3,4;
(1;6;6;1,1,1,1,1,1) Gen 3,4; *wayyo'mär hannåḥåš 'äl hå'iššåh lo' mowt*
Gen 3,4;
(1;5;5;1,1,1,1,1) Gen 3,4; *wayyo'mär hannåḥåš 'äl hå'iššåh lo'*
Gen 3,4;
(1;5;3;1,0,1,0,1) Gen 3,4; *wayyo'mär * 'äl * lo'*
Gen 3,4; Gen 8,21; Gen 48,18; Ex 11,9; Ex 19,23; Num 22,12; Jos 24,19; Jos 24,21; 1 Sam 15,26; 1 Sam 16,10; 1 Sam 17,33; 1 Sam 17,39; 2 Sam 16,18; 2 Sam 17,7; 2 Sam 19,24; 2 Sam 24,24;

* Protokoll des 57. Kolloquiums über die Anwendung der EDV in den Geisteswissenschaften an der Universität Tübingen am 6. Februar 1993.

Historical Social Research, Vol. 19 — 1994 — No. 3, 156-163

1. Formale Beschreibung des Problems

Wo nichts anderes angegeben, ist W ein Text, V ein Textkorpus mit den Bezeichnungen der Definitionen 1.1 und 1.2.

1.1 Definition. – Sei Σ ein Alphabet, Σ^+ die Menge der Wörter über Σ positiver Länge. Unter einem Text W verstehen wir eine endliche Folge $W = (w_1, ..., w_k)$ von Wörtern aus Σ^+. Die Menge $M_W = \{1, ..., k\}$ ist die Indexmenge von W. $k = |M_W|$ ist die Mächtigkeit von M_W, das heisst die Länge von W.

1.2 Definition. – Unter einem Textkorpus $V = (V_1, ..., V_m)$ verstehen wir eine endliche Folge von Texten $V_i = (V_{i1}, ..., V_{ik_i})$. $M_{V_i} = \{1, ..., k_i\}$ ist die Indexmenge von V_i.

1.3 Definition. – Sei $U = (u_1, ..., u_l)$ eine Wortfolge, $W = (w_1, ..., w_k)$ ein Text mit den Indexmengen M_U und M_W. $f : M_U \to M_W$ sei eine streng monoton wachsende Funktion, derart, dass für alle $i \in M_U$ gilt, dass $u_i = w_{f(i)}$. Dann sagen wir, dass U in W vorkommt, und die Menge $F_W = \{f(i) \mid i \in M_U\}$ heisst eine Fundstelle von U in W. f heisst die Verweisfunktion von U nach W, $\sigma_{F_W} = f(l) - f(1) + 1$ heisst die Streuung von F_W.

1.4 Bemerkung. – Während es zu jeder Wortfolge U keine, genau eine oder mehrere Fundstellen in W geben kann, gibt es umgekehrt zu jeder Teilmenge F_W der Indexmenge M_W genau eine Wortfolge U, von der F_W Fundstelle ist.

1.5 Satz. – Sei W ein Text mit Indexmenge M_W. Dann sind die Teilmengen von M_W bezüglich der Relation $\subseteq$ partiell geordnet.

1.6 Satz. – Sei W ein Text mit Indexmenge M_W. $P(M_W)$ sei die Menge der Teilmengen von M_W. $<_*$ sei eine totale (lineare) Ordnung auf $P(M_W)$ derart, dass für alle Teilmengen $F_{W,i}$ und $F_{W,j}$ von M_W gilt, dass aus $|F_{W,i}| < |F_{W,j}|$ folgt dass $F_{W,i} <_* F_{W,j}$ ist. Dann ist $<_*$ eine totale (lineare) Ordnung auf $P(M_W)$ die die partielle Ordnung aus 1.5 respektiert.

1.7 Definition. – Als Standardordnung auf einer Fundstellenmenge bezeichnen wir im folgenden die totale Ordnung, die die Fundstellen als aufsteigend geordnete Mengen erstens der Mächtigkeit nach absteigend und zweitens lexikalisch aufsteigend anordnet.

1.8 Definition. – Unter einem Verweis Q von einem Text W in ein Textkorpus V verstehen wir ein geordnetes Tripel (F_W, U, F_{V_i}) bestehend aus einer Wortfolge U und Fundstellen F_W, F_{V_i} von U in W und in einem V_i in V.

1.9 Definition. – Als Standardordnung auf einer Verweismenge bezeichnen wir im folgenden die totale Ordnung, die die Verweise erstens aufsteigend ordnet

bzgl. der Standardordnung auf den W-Fundstellen, zweitens aufsteigend anordnet bzgl. der Textnummer der V-Fundstellen und drittens aufsteigend anordnet bzgl. den Standardordnungen auf den Fundstellen der Texte von V.

1.10 Definition. – Unter einem einpunktigen Verweis verstehen wir einen Verweis $Q = (F_W = \{j_W\}, U, F_{V_i} = \{j_{V_i}\})$, also mit $|F_W| = 1$. Q ist durch j_{V_i}, i und $Q_d := j_{V_i} - j_W$ eindeutig bestimmt.

1.11 Definition. – Als Standardordnung auf einer Menge einpunktiger Verweise bezeichnen wir im folgenden die totale Ordnung, die die Verweise erstens aufsteigend anordnet bzgl. der Textnummer der V-Fundstellen, zweitens aufsteigend anordnet bzgl. Q_d und drittens aufsteigend anordnet bzgl. den Standardordnungen auf den Fundstellen der Texte von V.

1.12 Definition. – Sei $Q = (F_W, U = (u_1, ..., u_k), F_{V_i})$ ein Verweis. f_W, f_{V_i} seien die zugehörigen Verweisfunktionen von U nach W bzw. V_i. Q heisst homogen, wenn für alle m,n in M_U (der Indexmenge von U) gilt $f_{V_i}(m) - f_W(m) = f_{V_i}(n) - f_W(n)$, das heisst, wenn für die einpunktigen Verweise $Q_1 = (\{w_{f_W(1)}\}, u_1, \{v_{f_{V_i}(1)}\}), ..., Q_k = (\{w_{f_W(k)}\}, u_k, \{v_{f_{V_i}(k)}\})$ gilt $Q_{1,d} = ... = Q_{k,d}$.

1.13 Definition. – Sei $Q = (F_W, U, F_{V_i})$ ein (homogener) Verweis. Q heisst maximal bzgl. seiner Streuung oder σ-vollständig, wenn für jeden (homogenen) Verweis $Q' = (G_W, U', G_{V_i})$ mit $F_W \subset G_W$ und $F_{V_i} \subset G_{V_i}$ gilt $\sigma_{G_W} > \sigma_{F_W}$.

1.14 Bemerkung. – Ist Q ein σ-vollständiger homogener Verweis, so ist Q nicht notwendig ein σ-vollständiger Verweis.

2. Maschinelle Lösung des Problems: Die Spezifikation

Eine Lösung des Problems wurde in der Abteilung für Methodik computerunterstützter Textinterpretation der Fakultät für Informatik in TUSTEP implementiert, und liefert zufriedenstellende Ergebnisse für alttestamentliche (hebräische) Textkorpora. Sie ist derzeit folgendermassen spezifiziert:

2.1 Eingabedaten. – Ein Text W, ein Textkorpus V; diese müssen „gewissen" formalen Kriterien genügen. Ferner ein $n \in \mathbb{N}$ mit $2 \leq n \leq 99$. W muss in V enthalten sein, d.h. es gibt einen Text V_i in V mit einer Fundstelle von W der Streuung k.

2.2 Ausgabedaten. –
$\{Q = (F_W, U, F_{V_i}) \mid Q$ homogen $\wedge$ (Q $\sigma-$vollständiger homogener Verweis $\vee \exists_{Q'=(F_W,U,F'_{V_j})}$ Q' $\sigma-$vollständiger homogener Verweis) $\wedge 1 < \sigma_{F_W} \leq n\}$ als Liste mit „gewissem" Format.

3. Maschinelle Lösung des Problems: Die Programmierumgebung

Im Bezug auf die traditionellen Hindernisse für den TUSTEP-Programmierer, die komplizierte Unterprogrammtechnik und die Entfernung der TUSTEP-Syntax zu den heute gängigen programmierlogischen Konzeptionen konnten Erfolge verbucht werden, von denen zwei hier kurz vorgestellt werden. Im Übrigen ist das hier vorgestellte Programm als Makropaket in eine kleine Bibliothek von TUSTEP-Makros eingebunden, die auch anderweitig Verwendung finden. Das Programm kann in Form einzelner Makros aufgerufen werden, die vom Benutzer zu einem Hauptprogramm zusammengebunden werden können.

3.1 Beispiel. – Verarbeitung dynamischer Listen mit KOPIERE.
Listen werden mittels eindeutiger Kennungen als Textdatensätze implementiert.
Beispiel: @sv@|27@17675@PS150:005|27@17676@PS150:006
|28@17677@IJOB001:001@t@
repräsentiert eine (alttestamentliche) Belegstellenliste. "@sv@" markiert den Listenanfang, "|" den Anfang der einzelnen Elemente. Die ersten beiden Zahlen jedes Elements bedeuten Buchnummer und fortlaufende Versnummer. Mit Austauschoperationen wird ein Zeiger "@i@" vor das erste Element bzw. nach vorwärts bewegt. Mit Austauschoperationen werden Listenelemente gelöscht, mit dem Ersetzungstext neue eingesetzt. Mit Hilfe dieser Konzeption ist z.B. ein Programm erarbeitet worden, das oben angeführte Liste in zitierfähiges "Ps 150,5f.Job 1,1" verwandelt.

3.2 Beispiel. – Eine *while*-Schleife als TUSTEP-Makro.
Mit dem folgendermassen spezifizierten TUSTEP-Makro MWHILE kann ein anderes TUSTEP-Makro ausgeführt werden, solange gewisse Bedingungen gegeben sind. Die Art der Spezifikationsübergabe an das von MWHILE gerufene Makro ist von diesem abhängig und kann auch über globale Variable erfolgen.

3.2.1 Spezifikation. –

MWHILE führt ein TUSTEP-Makro solange aus, wie keine benutzereigene TUSTEP-Datei namens FAMMI zum Schreiben angemeldet ist, die leer ist und WAHLSCHALTER 7 gesetzt ist. WAHLSCHALTER 7 ist vom Benutzer vorher zu setzen. Der übergibt zur Spezifikation MDATEI den Namen der Datei, in der sich das aufzurufende Makro befindet, zur Spezifikation MAKRO den Namen des aufzurufenden Makros. Diese Spezifikationen werden nachgefordert, bei leerer Eingabe oder Eingabe von "-" wird die Ausführung von MWHILE beendet. Es können wahlweise sieben weitere Spezifikationen S1,...,S7 angegeben werden, die dann an das aufzurufende Makro als erste bis siebte Spezifikation weitergereicht werden.
Spez.folge: MDATEI MAKRO S1 S2 S3 S4 S5 S6 S7

4. Maschinelle Lösung des Problems: Das Verfahren

Mit den Bezeichnungen von 1. und 2. geht man folgendermassen vor:

$M, M', M'' := \emptyset$;
for $l := 1$ *to* k *do*
{ kopiere einpunktige W-V-Verweise mit $F_W = \{l\}$ nach M;
sortiere Verweise in M nach 1.11;
fasse Verweise in M mit gleicher Textnummer und gleichem Q_d zu σ-vollständigen homogenen Verweisen, deren W-Fundstelle $l - n + 1$ und mindestens ein weiteres Element enthält zusammen und kopiere diese nach M';
entferne einpunktige W-V-Verweise mit $F_W = \{l - n + 1\}$ aus M;
}
for $l := k - n + 2$ *to* k *do*
{ fasse Verweise in M mit gleicher Textnummer und gleichem Q_d zu σ-vollständigen homogenen Verweisen, deren W-Fundstelle l und mindestens ein weiteres Element enthält zusammen und kopiere diese nach M';
entferne einpunktige W-V-Verweise mit $F_W = \{l\}$ aus M;
}
sortiere M' absteigend nach 1.9;
while M' enthält Verweise $Q = (F_W, U, F_{V_i})$ mit $|F_W| < \sigma_{F_W}$ *do*
{ *for* $l := k - 1$ *downto* 1 *do*
{ $F :=$ W-Fundstelle F_W des ersten Verweises
$Q = (F_W, U, F_{V_i})$ mit $min(F_W) = l$ und $|F_W| < \sigma_{F_W}$;
entferne die Verweise mit W-Fundstellenmenge F aus M' und kopiere sie nach M'';
berechne aus den folgenden Verweisen $Q' = (H_W, U', H_{V_i})$ mit $min\ H_W = l$ und $F \subset H_W$ die fehlenden homogenen Verweise $Q'' = (F, U, G_{V_i})$, $(G_{V_i} \subset H_{V_i})$, und kopiere diese nach M'';
}
}
kopiere M' nach M'';
sortiere M'' nach 1.9;
entferne Duplikate aus M'';
bereite M'' zum Formatieren auf.

5. Maschinelle Lösung des Problems: Die Eingabedaten

Maschinenlesbare Texte können zu Eingabedateien W und V aufbereitet werden, wenn sie gewissen Ansprüchen genügen. Diese lauten im einzelnen:

5.1.1 Jeder TUSTEP-Satz beginnt mit der Anfangskennung "@" und endet mit "@@ @?", wenn es sich um den letzten Satz eines Textes aus einem Textkorpus handelt, darf dieser nicht mit "@@ @?" sondern nur mit "@?" enden.

5.1.2 Nach der Anfangskennung steht eine Belegstelle (BUCH KAPITELNUMMER "," VERS ";"; BUCH darf keine Ziffern enthalten, VERS ist eine beliebige Zeichenfolge).

5.1.3 Der Text zwischen dem ersten ";" und der Schlusskennung enthält nur druckbare Zeichen des 7-bit-ASCII Codes ausser "<", ">", "@", "|", "*", "#" und "\", ausser dem Apostroph, den Akzenten, dem Komma und der Tilde. BLANK wird als Worttrenner interpretiert, Gross- und Kleinschreibung werden nicht unterschieden.

5.2 Beispiel. – Gen 3,4f kann folgendermassen vorliegen:

@GEN003,004; WAYYO)MER HANNFXF$)EL HF)I$$FH LO) MOWT TMUTUWN @@ @?
@GEN003,005; KIY YOD"( )LOHIYM KIY BYOWM)KFLKEM MIMMENNUW WNIPQXUW ("YN"YKEM WIHYIYTEM K")LOHIYM YOD("Y +OWB WFRF(@?

6. Maschinelle Lösung des Problems: Einzelne Teilprobleme

6.1 Organisation der ersten beiden *for*-Schleifen in 4.
Die ersten beiden *for*-Schleifen werden dadurch realisiert, dass W in eine TUSTEP-Programmdatei folgenden Inhalts verwandelt wird:

```
#tue,iimsh01*sintc3.mak,k3,
@wi@1'@ws@1@1@GEN003:004|'WAYYO)MER'2''?1''?2
#tue,iimsh01*sintc3.mak,k3,
@wi@2'@ws@1@1@GEN003:004|'HANNFXF$'2''?1''?2
                .
                .
                .
#tue,iimsh01*sintc3.mak,k3,
@wi@21'@ws@1@2@GEN003:005|'WFRF('2''?1''?2
#tue,iimsh01*sintc3.mak,k4,@@'2''?1''?2
```

Der jeweilige Schleifenrumpf ist also das Unterprogramm K3 bzw. K4, es gilt hier $n = 2$. Die Programmdatei kann dann mit Hilfe eines anderen Makros portionsweise ausgeführt werden.

6.2 Codierung einpunktiger Verweise
Ein einpunktiger Verweis sieht typischerweise folgendermassen aus:

6.2.1 Beispiel. -

@**3tr**@5@wi@l@ws@l@l@GEN003:004|@wort@WAYYO)MER
@pi@l@d@2719@l@l@vt@l@vi@2720@sv@l@205@GEN008:021
@t@|WAYYO)MER@?

@**3tr**@5@wi@3@ws@l@l@GEN003:004|@wort@)EL
@pi@l@d@2719@l@l@vt@l@vi@2722@sv@l@205@GEN008:021
@t@|)EL@?

@str@5@wi@5@wsCHCH@GEN003:004|@wort@LO)
@pi@l@d@2719@l@l@vt@l@vi@2724@sv@l@205@GEN008:021
@t@|LO)@?

6.3 Codierung homogener Verweise
Ein homogener Verweis sieht typisch er weise folgender inassen aus:

6.3.1 Beispiel. -

@str@7@wi@000001
@ws@|l@l@GEN003:004|*|l@l@GEN003:004|*|l@l@GEN003:004
@wort@|WAYYO)MER|*|)EL|*|LO)
@l@05@vt@0001@vi@00002720
@sv@|l@205@GEN008:021|*|l@205@GEN008:021
|*|1@205@GEN008:021
@t@|WAYYO)MER|*|)EL|*|LO)@typ@03@l@0@l@0@l@?

6.4 Zusammenfassung eiripunktiger Verweise zu ^-vollständigen homogenen Verweisen
Aus den Verweisen von 6.2.1 wird unter anderem der Verweis in 6.3.1 berechnet.

6.5 Berechnung fehlender nicht *er*-vollständiger aber homogener Verweise zu vorhandenen ^-vollständigen homogenen Verweisen
Für Beispiel 0.5 werden zunächst nur

(11;4;4; 1,1,1,1) Gen 3,5; ***kiy byowm 'kälkäm rnimmännuw***
Gen 3,5;
(11;4;3;1,1,0,1) Gen 3,5; ***kiy byowrn * rnimmännuw***
Gen 2,17;
(11;4;2;1,0,0,1) Gen 3,5; ***kiy * * rnimmännuw***
Num 13,31;

als er-vollständige homogene Verweise berechnet. Um die restlichen beiden Verweise zu berechnen, muss der Rumpf der iu/»i7e-Schleife in 4. zweimal durchlaufen werden: Das erste Mal werden aus den angeführten Verweisen die Verweise

(11;4;2;1,0,0,1) Gen 3,5; ***kiy * * rnimmännuw***
Gen 2,17;
(11;4;2;1,0,0,1) Gen 3,5; ***kiy * * rnimmännuw***
Gen 3,5;

berechnet, das zweite **Mal** wird aus

(11;4;4;1,1,1,1) Gen 3,5; ***kiy byowm kälkäm mimmärmuw***
Gen 3,5;
(11;4;3;1,1,0,1) Gen 3,5; ***kiy byowm * mimmännuw***
Gen 2,17;

der Verweis

(11;4;3;1,1,0,1) Gen 3,5; ***kiy byowm. * mimmännuw***
Gen 3,5;

berechnet.

6.6 **Druckausgabc**
Die Druckausgabe unterstützt wahlweise TUSTEP (FORMATIERE) oder TßX. **VT**-Textpassagen werden für die Ausgabe geeignet **transkribiert,** Stellenlisten in zitierbare Belegstellen umgewandelt. Für die **Transkriptionsroutinen** sind Traiiskriptionsroutiiicn für das hebräische AT voreingestellt, die Namen dieser Unterprogramme können über globale Variable verändert werden.

7. Geplante Erweiterungen

7.1 Nichthebräische Texte
Die Bereitstellung ergänzender Makros (Traiiskriptoren u.a.) zur Bearbeitung von anderen Texten als solchen des hebräischen ATs erfolgt im Bedarfsfall. Aus Kapazitätsgründen muss die Bereitstellung eines LXX-Textkorpus leider noch eine Weile warten.

7.2 **Erhebun**g aller homogener Verweise
Geplant ist, die *while*-Schleife des Verfahrens aus 4. wahlweise durch eine Prozedur ersetzbar zu machen, mit der zu gegebenen ^-vollständigen homogenen Verweisen für jeden Verweis alle **streuungsgleichen** homogenen Verweise berechnet werden, deren PV-Fimdstellenmengc in ihrer **W^-Fundstellenmenge** enthalten ist.

7.3 Nicht**-ho**mogene Verweise
Geplant ist, ein Verfahren zu implementieren, das auch nicht-homogene Verweise erfasst. Dieses ist allerdings wesentlich aufwendiger zu realisieren und wesentlich **rechenaufwendiger,** so dass das hier beschriebene Programm auch dann noch gebraucht wird.

7.4 Rechtschreib- und sonstige Varianten von *W*-**Wörtern**
Bis jetzt werden nur *W*-Wortfolgen in *V* gefunden, die ohne jede Abweichung in *V* gleich wie in *W* vorkommen. Verfahren zum Auffinden „ähnlicher" Belege für Einzelwörter werden zur Zeit am Institut diskutiert.

Historical Social Research, Vol. 19 — 1994 — No. 3, 156-163

FORTHCOMING EVENTS

ZHSF - WORKSHOP 1994

Wie aus Zahlen Bilder werden: Die Anwendung graphischer Darstellungstechniken in der Historischen Sozialforschung

9. - 10. Dezember 1994

Angebot: In diesem Workshop werden Methoden der Visualisierung von Zahlen dargestellt. Sie lassen sich in zwei Gruppen aufteilen:

Präsentationsgraphik: Die Graphik ist vorwiegend dafür gedacht, Ergebnisse statistischer Untersuchungen einem Leser- oder Zuhörerkreis zu präsentieren. Dabei geht es um die Erstellung des Endproduktes der statistischen Analyse. Beispiele sind die Herstellung von Säulen- und Balkendiagrammen, Kuchendiagrammen, aber auch die Darstellung von Zeitreihen. Die entscheidenden Ziele der Präsentationsgraphik bestehen in der adäquaten Wiedergabe der Ergebnisse und in der gut lesbaren und ästhetisch ansprechenden Darstellung.

Graphische Datenanalyse: Unter graphischer Datenanalyse versteht man im Gegensatz zur Präsentation von statistischen Ergebnissen, den Einsatz visueller Mittel zur Datendarstellung, um

- weitere Einblicke über die Struktur der Daten zu erhalten und um
- zu prüfen, welche statistischen Verfahren zur weiteren Auswertung der Daten eingesetzt werden können.

Die Visualisierung dient somit dem Zweck, die Eignung der Daten für weiterführende Analysen zu überprüfen.

Inhalte: Rolle der graphischen Darstellung in der Statistik; Geschichte und Entwicklung der statistischen Graphik; Aufgaben der Graphik in den einzelnen Teilbereichen der Statistik; Vor- und Nachteile graphischer Darstellungen; Anforderungen an graphische Darstellungen und abgeleitete Standards für graphische Darstellungen; Klassifikation der statistischen Graphik; Präsentationsgraphiken (das passende Schaubild finden: Klassifikation und Einsatzmöglichkeiten); Beurteilung von Verzerrungen bei der Wahl der Darstellung; Datenanalysegraphiken (grundlegenden