

Seminar Technische Informatik

Top 10 Algorithms in Data Mining

Stephan Mielke

Zusammenfassung—In diesem Paper werden die 10 von Xindong Wu beschriebenen Data-Mining-Algorithmen eingeordnet sowie die wichtigsten vorgestellt [1]. Zu diesem Zweck wird auf die Kategorie der Cluster-, Klassifizierungs- und Assoziations-Algorithmen eingegangen. Näher vorgestellt werden der k-means und der SVM. Anschließend wird auf Big Data sowie dessen Bedeutung für Data Mining eingegangen.

Index Terms—Assoziation, Big Data, Clustering, Data Mining, k-means, Klassifikation, Support Vector Machines



Abbildung 1. Hubble Ultra Deep Field [2]

1 EINLEITUNG

Das Hubble-Teleskop nahm vom 3. September bis zum 16. Januar 2004 das *Hubble Ultra Deep Field* auf, welches als Ausschnitt in Abb. 1 zu sehen ist. Auf dieser Aufnahme sind über 10.000 stellare Objekte abgebildet. Problematisch ist jedoch die Identifikation eines solchen Objekts. Ist es ein Stern, eine Galaxie, ein Quasar oder einfach eine Störung des Sensorschips?

Das Problem der Unterscheidung, ob es sich bei einem Objekt um einen Stern, eine Galaxie oder etwas Unbekanntes handelt, fällt in den Bereich des Data Mining. Peter J. O’Keefe et al. beschreiben einige Algorithmen zur rechnerunterstützten Lösung des Problems [3]. Konkret werden jedem sichtbarem Objekt 9 Attribute wie Leuchtkraft zugeordnet, die aus Aufnahmen in verschiedenen Frequenzbereichen ex-

Tabelle 1
Erkennungsrate verschiedener Algorithmen zur Klassifizierung stellarer Objekte [3]

Algorithmus	korrekt erkannt Objekte
Random Forest	82.89%
Decision Tree	80.68%
Artificial Neural Network	75.82%
Support Vector Machines	37.82%

trahiert werden und zu einem als *stellary* bezeichneten Wert zwischen 0 und 1 ausgewertet werden. Dabei repräsentiert ein Wert zwischen 0.0 und 0.1 eine Galaxie und zwischen 0.9 und 1.0 ein Stern repräsentiert. Zwischen diesen Intervallen ist das Objekt nicht eindeutig identifizierbar. In Abschnitt 1 sind die Erkennungswahrscheinlichkeiten einiger Algorithmen für die Klassifizierung von stellaren Objekten aufgelistet.

Tabelle 2
Top 10 Algorithmen [1]

Platz	Name	Art
1.	C4.5	Classification
2.	k-means	Clustering
3.	Support Vector Machines	Classification
4.	Apriori	Association
5.	EM Algorithm	Classification
6.	PageRank	Link Mining
7.	AdaBoost	Classification
8.	k-nearest neighbor	Classification
9.	Naive Bayes	Classification
10.	CART	Classification

Während die Abschnitt 1 listet die 10 bedeutendsten Algorithmen in Data Mining im allgemeinen Kontext ausgewählt anhand von Nominierungen¹ durch Preisträger eines ACM KDD Innovation Awards oder eines IEEE ICDM Research Contributions Awards [1].

2 DATA MINING

DATA MINING wird in der Forschung, Vermarktung, Medizin, (Wetter)vorhersagen, Betrugsauflärung usw. eingesetzt. Der Begriff „Data Mining“ wird allerdings im Deutschen fälschlicherweise für den gesamten Prozess der *Knowledge Discovery in Databases* (KDD) verwendet. Wie in Abb. 2 ersichtlich, stellt Data Mining jedoch lediglich einen Schritt im Prozess der KDD dar. Für KDD werden zuerst die Daten gesammelt, aufbereitet und homogenisiert um mittels eines Data-Mining-Algorithmus verarbeitet zu werden, wodurch Muster entstehen, die schließlich zum Ziel das heißt zur Interpretation und dementsprechend der Wissensgewinnung führt. Im Folgenden seien die wichtigsten Kategorien dieser Data-Mining-Algorithmen erläutert.

2.1 Clustering-Algorithmen

ALGORITHMEN DIE DES CLUSTERINGS zugeschrieben werden, ordnen Daten unbekannter Klassen, den so genannten *Clustern* (Abschnitt 2.1.1), zu. Während des Clustering wird eine Funktion gesucht, die Objekte auf Gruppen abbildet. Diese Gruppierung wird mithilfe einer *Distanzfunktion* (Abschnitt 2.1.2) erreicht, die die Ähnlichkeit zwischen Objekten numerisch ermittelt. Alle Clustering-Algorithmen arbeiten mit vereinfachenden Heuristiken, da das Clustering \mathcal{NP} -vollständig ist.

2.1.1 Cluster

DIE GESTALT UND FORM der Cluster ist in den einzelnen Algorithmen sehr unterschiedlich [5]. Beispielsweise können die Cluster untereinander Untermengen bilden (hierarchisches Clustering) oder hierarchisch

unabhängig sein (flaches Clustering). Auch können Objekte zu einem (hartes Clustering) oder zu mehreren voneinander unabhängigen Clustern (weiches Clustering) zugeordnet werden. Bei allen Algorithmen ist die Anzahl der Cluster begrenzt, wobei sich die Schwelle durch Anzahl (fixe Anzahl) oder durch eine minimale Qualität der einzelnen Cluster (dynamische Anzahl) ergibt. Außerdem ist die Clustering-Qualität nie genau definiert, doch sie kann durch Anzahl der einzelnen Objekte die einem Cluster angehören, durch die Größe der maximalen Unähnlichkeit eines Objektes zu seinem Cluster oder das Vermeiden von „Lücken“ im Cluster beschrieben werden [6]. In Abb. 3(b) sind beispielhaft 2 Cluster (rot/blau) und die Clusterzentren (schwarz) des k-means-Algorithmus (Abschnitt 2.1.3) aufgezeigt.

2.1.2 Distanzfunktion

DIE Distanzfunktion bestimmt den Abstandsvektor zwischen zwei Objekten. Statt einer Distanzfunktion wird manchmal auch eine Ähnlichkeits- bzw. Simulationsfunktion benutzt, jedoch sind in diesem Fall die Werte invertiert zu betrachten. Der Gebrauch von Distanzfunktionen hat sich jedoch durchgesetzt, da die zugrundeliegende Algebra numerisch stabiler sind, was auf die Bedingung (Gleichung (2)) zurückzuführen ist. (statt \propto wird $\vec{0}$ angewandt). Jedes Objekt $o_i = (a_1, \dots, a_n)$ besteht aus n Attributen. Jedes Attribut wiederum ist eine numerisches, kategorisches oder anders artige Eigenschaft und besitzt für sich selbst spezielle Distanzfunktionen. Die Bedingungen (Gleichungen (1) bis (3)) müssen für jede Distanzfunktion gelten. Wenn die Bedingung (Gleichung (4)) gilt, handelt es sich um eine Metrik.

$$\text{dist}(o_1, o_2) = d \in \mathbb{R}^{n \geq 0} \quad (1)$$

$$\text{dist}(o_1, o_2) = \vec{0} \Leftrightarrow o_1 = o_2 \quad (2)$$

$$\text{dist}(o_1, o_2) = \text{dist}(o_2, o_1) \text{ (Symmetrie)} \quad (3)$$

$$\text{dist}(o_1, o_3) \leq \text{dist}(o_1, o_2) + \text{dist}(o_2, o_3) \quad (4)$$

Für numerische Attribute sind beispielhaft Distanzfunktionen (Gleichungen (5) bis (8)) und für kategorische Attribute (Gleichung (9)) angegeben. Darüber hinaus existieren noch weitere Attribute wie zum Beispiel textuelle

1. <http://www.cs.uvm.edu/~icdm/algorithms/CandidateList.shtml>

die für ihre gehörigen Distanzfunktion mindestens die Bedingungen Gleichungen (1) bis (3) erfüllen müssen.

Euklidische-Distanz:

$$\text{dist}(x, y) = \sqrt{(x_1 - y_1)^2 + \dots + (x_n - y_n)^2} \quad (5)$$

Manhattan-Distanz:

$$\text{dist}(x, y) = |x_1 - y_1| + \dots + |x_n - y_n| \quad (6)$$

Maximum-Metrik:

$$\text{dist}(x, y) = \max(|x_1 - y_1| + \dots + |x_n - y_n|) \quad (7)$$

Alg. L_p -Metrik:

$$\text{dist}(x, y) = \sqrt[p]{\sum_{i=1}^d (x_i - y_i)^p} \quad (8)$$

Summe der Unterschiede:

$$\text{dist}(x, y) = \sum_{i=1}^a \delta(x_i, y_i) \quad (9)$$

$$\delta(x_i, y_i) = \begin{cases} 0 & \text{wenn } (x_i = y_i) \\ 1 & \text{wenn } (x_i \neq y_i) \end{cases} \quad (10)$$

2.1.3 k-means

DER k-means, oder auch Lloyd's Algorithmus genannt, gehört zu den Clustering-Algorithmen und basiert auf einem harten flachen Clustering. Die Objekte o_i werden als ein Vektor aus dem Vektorraum R^n interpretiert. Ein Cluster $A = \{o_1, \dots, o_i\}$ ist eine Menge von Objekten o_i und dessen Zentrum ist wie folgt definiert: $\mu(A) = \frac{1}{m} \sum_{i=1}^m o_i$. Ein Cluster besitzt eine hohe Güte, wenn $\text{RSS}(A) = \sum_{i=1}^m \|d_i - \mu(A)\|^2$ minimal ist. Das gesamte Clustering ist optimal, wenn $\text{RSS}(A_1, \dots, A_k) = \sum_{j=1}^k \text{RSS}(A_j)$ mi-

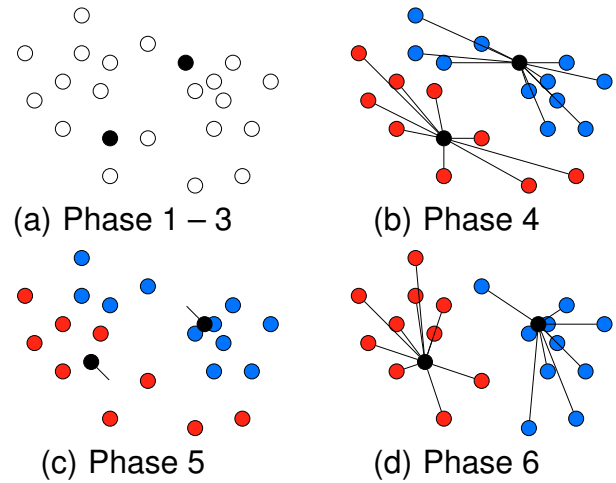


Abbildung 3. k-means

nimal ist.

Die Arbeitsweise des Algorithmus ist wie folgt:

- 1) Selektiere zufällig k Zentren als Startwert
- 2) Erstelle k leere Cluster
- 3) Weise jedem Cluster einen Zentren zu (??)
- 4) Weise jedem Datenvektor den Cluster mit dem nächstem Zentrum zu (??)
- 5) Berechne den Zentren jedes Clusters neu (??)
- 6) Teste, ob die Qualität des Clusterings ausreicht, sonst gehe zu Punkt 2 (??)

2.2 Classification-Algorithmen

BEI der Kategorie der Classification-Algorithmen sind anders als bei den Clustering-Algorithmen die genauen Klassen in die eingruppiert wird bereits bekannt.

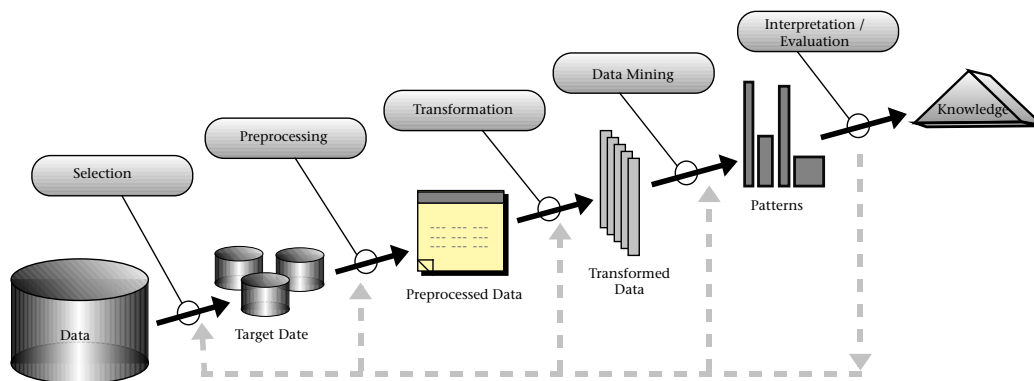


Abbildung 2. KDD nach Fayyad [4]

Tabelle 3
Beispiel Daten Versicherung

Alter	Autotyp	Risikoklasse
23	Familie	Hoch
17	Sport	Hoch
43	Sport	Hoch
68	Familie	Niedrig
32	LKW	Niedrig

Zusätzlich werden jedoch Trainingsdaten bei er Klassifikation verwendet (Abschnitt 2.2.1). Die verwendeten Distanzfunktionen sind mit denen des Clusterings (Abschnitt 2.1.2) vergleichbar.

2.2.1 Training

TRAININGSDATEN sind eine Menge von Objekten $O = \{o_1, \dots, o_n\}$ bei denen die Klassen $C = \{c_1, \dots, c_m\}$ bereits bekannt sind oder manuell ermittelt wurden. Für die Objekte gelten die gleichen Eigenschaften wie beim Clustering (Abschnitt 2.1.2). Um das Training zu verdeutlichen wird ein Beispiel aus dem Buch [5] verwendet. Tabelle 3 illustriert die Objekte mit entsprechenden Attributen und entsprechend zugeordneter Klasse. Aus diesen Trainingsdaten wird der, in Listing 1 als Bedingung formulierte, Entscheidungsbaum erzeugt. Mit welchem unklassifizierte Objekte klassifiziert werden können.

```

if Alter > 50 then Risikoklasse = Niedrig
if Alter <= 50 and Autotyp = LKW
  then Risikoklasse = Niedrig
  else Risikoklasse = Hoch

```

Listing 1. Entscheidungsbaum

2.2.2 Support Vector Machines

BEI den Support Vector Machines (SVM) handelt es sich um eine Untergruppe des *Statistical Learning* und darüber hinaus um eine Unterkategorie der Klassifikation. Der SVM kann spaltet eine Menge von Objekten in zwei disjunkte Teilmengen. Zwischen den beiden Teilmengen der Trainingsdaten wird eine Hyperplane im n -dimensionalen Vektorraum erstellt. Der Abstand zwischen der Hyperplane und den Begrenzungsobjekten der Teilmengen, die so genannten Supportvektoren, ist maximal. Dieser Abstand wird mittels

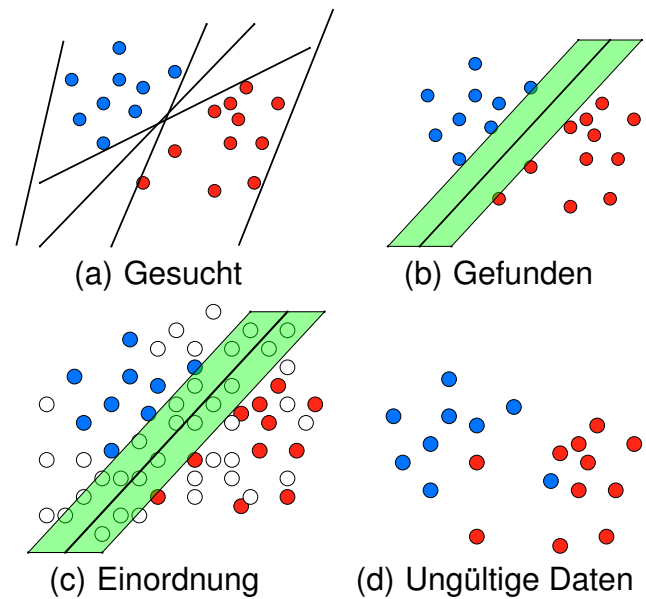


Abbildung 4. SVM

Differenzfunktionen wie beim Clustering (Abschnitt 2.1.2) ermittelt.

In der Abb. 4(a) sind die Trainingsdaten den beiden Klassen (rot / blau) zugeordnet. Die beiden äußeren Hyperplanekandidaten sind ungültig, da diese die Menge der Objekten nicht in zwei disjunkte Teilmengen schneidet. Von den inneren Hyperplanekandidaten fallen die beiden Äußerer weg, da diese einen geringeren Abstand zu den Supportvektoren als die Mittlere haben. Der mittlere Hyperplanekandidat ist dementsprechend die richtige Hyperplane (Abb. 4(b)). Der grün markierte Bereich kennzeichnet den Bereich des maximalen Abstands zu den Supportvektoren. Nach dem Training werden die unklassifizierte Objekte eingefügt und werden je nach dem auf welcher Seite der Hyperplane sie sich befinden der passenden Klasse zugeordnet (Abb. 4(c)). Bei ungültigen Trainingsdaten kann keine gültige Hyperplane gefunden werden, wodurch Objekte zwangsweise Objekte falsch klassifiziert werden (Abb. 4(d)).

2.3 Assoziation-Algorithmen

DIE meisten online Käufer ist der Satz „Kunden, die diesen Artikel gekauft haben, kauften auch...“ wohl bekannt. Die angezeigten Ergebnis werden mit Hilfe von Data Mining Algorithmen der Assoziation-Kategorie ermittelt. Die Grundidee dieser Algorithmen

ist es, Regeln der Form $A \Rightarrow B$ (Definition ??) zu finden, die einen *Support* (Definition ??) und eine *Konfidenz* (Definition ??) besitzen, welche einen festgelegten Schwellenwert übersteigt. Um die Regeln zu finden, benötigt man eine Transaktionsdatenbank, wie zum Beispiel eine Einkaufshistorie. Die Transaktionsdatenbank $D = \{T_1, \dots, T_n\}$ ist eine Menge von Transaktionen, wobei jede Transaktion $T_i \subseteq I$ eine Teilmenge aller Items (Waren) ist. Die Itemmenge $I = \{i_1, \dots, i_m\}$ stellt die verkaufbaren Waren dar, während ein Itemset $X \subseteq I$ eine Teilmenge aller Waren ist. Der Unterschied zwischen T_i und X ist, dass T_i eine reale Transaktion und X somit ein Ausschnitt einer Transaktion ist.

Assoziationsregel:

$$R_i = X \Rightarrow Y \text{ es gilt: } X, Y \subseteq I \wedge X \cap Y = \emptyset \quad (11)$$

Support der Regel:

$$\delta(R_i, D) = \delta(X \cup Y, D) \quad (12)$$

Support der Menge:

$$\delta(X, D) = \text{Anteil (\%)} \text{ aller } T_i \text{ für die gilt } X \subseteq T_i \quad (13)$$

Konfidenz der Regel:

$$\phi(R_i, D) = \delta(Y, \{T_i \mid \forall T_i \in D \wedge X \subseteq T_i\}) \quad (14)$$

Zur Verdeutlichung ist eine Transaktionsdatenbank mit sechs Einträgen gegeben Tabelle 4. In drei der sechs Einträgen ist ein Itemset von $x = \dots$ vorhanden und besitzt somit einen Support (Gleichung (13)) von 50% ($\delta(X, D) = 50\%$). Als nächstes wird ein Itemset gesucht, bei dem ein weiteres Item existiert und von dem X eine Teilmenge ist. Hierfür wird das Itemset $Z = \{\text{Kaffee, Milch, Kuchen}\}$ gewählt, aus welchem die Regel $R = \{\text{Kaffee, Milch}\} \Rightarrow \{\text{Kuchen}\}$ gebildet wird. Die Regel R (Gleichung (11)) besitzt eine Konfidenz (Gleichung (14)) von 66% ($\phi(R, D) = 66\%$) und einen Support (Gleichung (12)) von 33% ($\delta(R, D) = 33\%$). Somit existiert eine 33%-ige Wahrscheinlichkeit, dass ein Kunde zugleich alle drei Produkte kauft und mit einer Wahrscheinlichkeit von 66% kauft ein Kunde ebenfalls Kuchen sofern er schon Kaffee und Milch gekauft hat.

3 BIG DATA

NACH Schätzungen speicherte die gesamte Menschheit im Jahre 2007 ganze 300 Exabyte an digitalen Daten. Bis zum Jahr 2013

Tabelle 4
Transaktionsdatenbank

T_i	Itemset (X_i)
1	Brot, Kaffee, Milch, Kuchen
2	Kaffee, Milch, Kuchen
3	Brot, Butter, Kaffee, Milch
4	Milch, Kuchen
5	Brot, Kuchen
6	Brot

vervierfachte sich diese Datenmenge auf 1200 Exabyte, dies würde auf CDs gespeichert fünf Stapel von der Erde bis zum Mond bilden [7]. Diese Menge an Daten wird steigt weiter dramatisch an, was auf ... zurück zuführen ist. Noch nie in der Geschichte der Menschheit, hat der Mensch so viele Daten produziert und aufgezeichnet wie zum heutigen Tage, wobei damit nicht nur die Sammelwut unserer Geheimdienste gemeint [?].

Big Data verschiebt die Schwierigkeit von der Datenbeschaffung zur Datenauswertung. Bis ins letzte Jahrzehnt war die Gewinnung der Daten, sei es in Forschung, Medizin, Marketing oder auch „Spionage“ usw., im Gegensatz zur Analyse und Auswertung das eigentliche Problem der Datenverarbeitung. Damit verschiebt sich die Erkenntnis durch Big Data vom WARUM ist etwas so wie es ist, zum WAS ist wie es ist. Der Grundgedanke von Big Data ist die Verarbeitung riesiger Datenmengen zur Gewinnung von Wahrscheinlichkeiten, welche zu einer genaueren Vorhersagen genutzt werden können, wie zum Beispiel: gehört eine E-Mail zu Spam; ist bei der Autokorrektur das eingegebene „dei“ eher ein „die“; welcher Spieler der Bundesliga passt am Besten in das Team; wie ist die zukünftige Entwicklung an der Börse und so weiter. Jedoch ist bei Big Data zu bedenken, dass alle Aussagen sich auf Wahrscheinlichkeiten beziehen und somit bedingt durch ... eine falsche Schlussfolgerung zulassen. Zitat: „Was wir an Genauigkeit auf der Mikroebene verlieren, gewinnen wir an Erkenntnis auf der Makroebene.“ [7]

3.1 HACE Theorem

DAS HACE Theorem behandelt die dauerhafte Generierung unterschiedlichster und im Extremfall ungenauer bis falscher

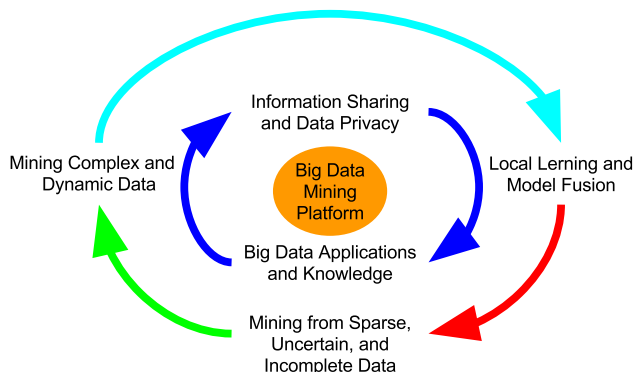


Abbildung 5. Big Data Herausforderungen [8]

Daten und deren immer komplexer sowie verzweigter werdenden Beziehungen. Wu Xindon definiert es wie folgt: „Big Data starts with large-volume, Heterogeneous, Autonomous sources with distributed and decentralized control, and seeks to explore Complex and Evolving relationships among data.“ [8]

3.2 Herausforderungen für Data Mining

DURCH das HACE Theorem entstehen für das Data Mining viele Schwierigkeiten. In der Abschnitt 3.2 sind die drei Ebenen des Data Mining für Big Data und deren Aufgaben, von innen nach außen, beschrieben. In der ersten Ebene, existieren die Schwierigkeiten des Datenzugriffs und der Datenanalyse, Da sich die Datenquellen räumlich getrennt von einander befinden, müssen diese über das Internet an den Verarbeitungsknoten transportiert werden. Des Weiteren sind die meisten Data Mining Algorithmen für kleine Datenmengen ausgelegt, weil alle Daten lokal im RAM gespeichert sein müssen. Auf der zweiten Ebene sind ethische und moralische Bedenken ein Problem. Das Verknüpfen von Informationen über Patienten, wie Ernährung, Aktivitäten, Krankheitsverläufe usw., in der medizinischen Forschung ist für die Gesellschaft weniger ein Problem als das gleiche für die Risikobewertung in einer Krankenversicherung. Aus diesen Gründen beleuchtet die zweite Ebene die Gefahr des gläsernen Menschen. Die dritte Ebene befasst sich mit dem Algorithmen Design für die Anforderungen an Big Data mit dem HACE Theo-

rem. Diese Algorithmen müssen die heterogenen und teilweise ungenauen bis falschen oder auch doppelte Daten filtern und homogenisieren. Das gefundene Wissen bzw. die Muster oder Modelle müssen nach der Analyse zusammengefasst und in ihrer Gesamtheit betrachtet werden.

4 ZUSAMMENFASSUNG

MITTELS Data Mining werden Zusammenhänge und „Wissen“ aus relativ kleinen Datenbeständen erzeugt. Im Alltag kommt jeder mit dessen Ergebnissen in Kontakt, sei es bei der Google Suche oder einem Einkauf bei Amazon. Big Data verändert unser Verständnis von der Wissensgewinnung und verdrängt das „Bauchgefühl“ durch auf Fakten und Statistiken fundierte Entscheidungen. Die Gefahr des gläsernen Mensch ist bei Big Data omnipräsent.

LITERATUR

- [1] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, „Top 10 algorithms in data mining,“ *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, 2008.
- [2] S. B. S. NASA, ESA and the HUDF Team. (2004) Hubble ultra deep field. [Online]. Available: <http://imgsrc.hubblesite.org/hu/db/images/hs-2004-07-a-pdf.pdf>
- [3] P. J. O’Keefe, M. G. Gowanlock, S. M. McConnell, and D. R. Patton, „Star-galaxy classification using data mining techniques with considerations for unbalanced datasets,“ in *Astronomical Data Analysis Software and Systems XVIII*, vol. 411, 2009, p. 318.
- [4] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, „From data mining to knowledge discovery in databases,“ *AI magazine*, vol. 17, no. 3, p. 37, 1996.
- [5] M. Ester and J. Sander, *Knowledge discovery in databases: Techniken und Anwendungen*. Springer Heidelberg, 2000, vol. 2, no. 4.
- [6] W.-T. Balke, „Data warehousing and data mining techniques,“ University Lecture, 2014.
- [7] V. Mayer-Schönberger and K. Cukier, *Big Data*. Computer Press, 2014.
- [8] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, „Data mining with big data,“ *Knowledge and Data Engineering, IEEE Transactions on*, vol. 26, no. 1, pp. 97–107, 2014.