Eine Einführung in

Cross Language Information Retrieval

Olaf Poneta

Universität Koblenz, Fachbereich 4
Proseminar Multimediadatenbanken und Retrieval
56068 Koblenz, Deutschland
(Dr. Martin Grzegorzek, Antje Schultz)

Keywords: Cross Language, Information, Retrieval, CLIR, Sprachübergreifend,

Einleitung

Das Internet enthält Informationen und Dokumente in mindestens 51 Sprachen. Cross Language Information Retrieval beschäftigt sich mit Fragenstellungen, die es ermöglichen, aus diesen Informationen nutzbares Wissen zu gewinnen und es sprachübergreifend zugänglich zu machen.

Im ersten Kapitel dieser Arbeit werden grundlegende Konzepte des Information Retrieval vorgestellt und anschließend im zweiten Kapitel auf Fragestellungen des Cross Language Information Retrieval erweitert.

1 Information Retrieval

Information Retrieval (IR) ist ein Fachgebiet der Informatik, welches sich zu großen Teilen in den Bereich der Computerlinguistik einordnen lässt. Diese beschäftigt sich mit der computergestützten Verarbeitung natürlicher Sprache. Wörtlich bedeutet der englische Begriff *information retrieval* Informationswiedergewinnung oder Informationsbeschaffung. Wie aus der Wortbedeutung hervorgeht, werden durch IR vorhandene, aber nicht zugängliche Informationen verfügbar und nutzbar gemacht. Dazu müssen geeignete Repräsentationen und Möglichkeiten zur Strukturierung von komplexen Datenbeständen gefunden und deren effizientes und inhaltsorientiertes Durchsuchen ermöglicht werden. Ein IR-System versucht das Informationsbedürfnis eines Benutzers zu befriedigen, indem es zu einer Anfrage passende Suchergebnisse liefert.
Dieses Kapitel soll einen Einblick in die Grundlagen des Information Retrieval geben. Dazu werden die wesentlichen Vorgänge erläutert und durch viele, zum Teil ausführliche Beispiele veranschaulicht.

1.1 Der Naive Ansatz

Ein Benutzer will anhand bestimmter Schlagworte passende Bücher finden; dabei sollen als Beispiel die Worte *Verschwörung* und *Wissenschaft* in einem Roman vorkommen und das Wort *Illuminati* nicht. Um alle relevanten Bücher zu finden, wäre es möglich, die ganze Bibliothek zu durchsuchen, jedes Buch, in dem das Wort Illuminati vorkommt, auszuschließen und die übrigen bis zum Schluss zu lesen, um am Ende zu wissen, ob die anderen beiden Begriffe vorkamen oder nicht. Im Fall einer überschaubaren Bibliothek würde dieser Ansatz mit der heutigen Hardware wahrscheinlich funktionieren. In vielen anderen Anwendungsbereichen werden jedoch effizientere Vorgehensweisen nötig sein.

1.2 Grundlagen des Information Retrieval

Eine grundsätzliche Strategie von Information Retrieval Systemen ist es, sich auf die eigentliche Suchanfrage, die oft als Query bezeichnet wird, gut vorzubereiten. Im vorangegangenen Beispiel aus 1.1 ist die Suchanfrage „*Verschwörung* AND *Wissenschaft* AND NOT *Illuminati"*. Um nun nicht bei jeder neuen Anfrage die gesamte Datenmenge durchsuchen zu müssen, muss der Datenbestand aufbereitet werden. Dieser Vorgang lässt sich in die drei Phasen Identifikation, datenspezifische Vorverarbeitung und Indexierung unterteilen. Im Folgenden werden die einzelnen Phasen erläutert und im Anschluss an einem Beispiel verdeutlicht.

1.2.1 Identifikation

Abhängig davon, in welchem Kontext und in welcher Form die zu durchsuchenden Daten vorliegen, sind unter Umständen unterschiedliche Vorgehensweisen für die Vorverarbeitung jener Daten nötig. Ein HTML-Dokument wird aufgrund technischer Gegebenheiten anderen Vorverarbeitungsschritten als ein in ASCII vorliegender Text unterzogen. Um zu erkennen welche Vorverarbeitung für ein spezifisches Datum angewandt werden soll, muss dieses also zunächst identifiziert werden. Die Identifikation erfolgt über IR-System spezifische Heuristiken, wie beispielsweise die Analyse von Dateiendungen oder von Dateiheadern.

1.2.2 Datenspezifische Vorverarbeitung

Ist eine Datei identifiziert, kann sie in eine vorteilhafte, oft einheitliche Datenstruktur (z.B. einfachen Text) überführt werden. Meistens folgen darauf zwei weitere Vorverarbeitungsschritte: (a) Die Unterteilung des vorliegenden Textes in Tokens und (b) das Suchen und Entfernen von Stoppwörtern.

a. In deutschsprachigen Texten würde die Tokenisierung durch das Trennen eines Textes an seinen Leerzeichen begonnen werden. Der nächste Schritt bestünde darin, Phrasen (wie *juristische Person*) und Eigennamen (*Hong Kong*, *Mercedes Benz, Der König der Löwen*) als solche zu erkennen. Durch eine Kookkurrenzanalyse können Abhängigkeiten und Beziehungen von Wörtern festgestellt werden. Beispielsweise ginge aus einer Kookkurrenzanalyse hervor, dass die Tokens *Hong* und *Kong* im Deutschen

fast immer gemeinsam auftreten. Diese würden deshalb zu einem Token *Hong Kong* zusammengefügt werden. In einem weiteren Schritt der Tokenisierung werden Wörter auf ihre Grundform reduziert. Verschiedene morphologische Varianten eines Wortes werden im IR in der Regel nicht unterschieden. Für diesen als *stemming* bezeichneten Vorgang gibt es verschiedene experimentell begründete Algorithmen, die für die deutsche Sprache gute Ergebnisse liefern. Alle so erhaltenen Tokens kommen für die Indexierung in Frage.

b. Stoppwörter sind im Information Retrieval Wörter, die bei einer Volltextindexierung nicht beachtet werden, da sie in der benutzen Sprache sehr häufig auftreten und gewöhnlich keine Relevanz für die Erfassung des Dokumentinhalts besitzen. Häufig als Stoppwörter bezeichnet werden im Deutschen Artikel, Konjunktionen, Präpositionen oder Satzzeichen. Um die Stoppwörter aus einem Text zu entfernen, gibt es Listen mit entsprechenden Worten, die nur noch abgeglichen werden müssen. Wichtig ist, zu beachten, dass das Entfernen der Stoppwörter erst nach dem Identifizieren der Eigennamen passieren, da diese sonst verfälscht werden könnten (*König Löwen*).

1.2.3 Indexierung

Um für jede neue Suchanfrage den Aufwand einer linearen Suche durch den gesamten Datenbestand zu vermeiden, wird einmalig ein Index erzeugt, worin schnell zu sehen ist, welche Token in welchen Dokumenten vorkommen. Die einfachste Form der Indexierung wird im booleschen IR Modell verwendet. Um bei dem Beispiel der Buchsuche zu bleiben, wird davon ausgegangen, dass zu jedem Buch eine Liste mit allen in der Bibliothek benutzten Wörtern geführt wird (siehe Tabelle 1). Jeweils nur mit der Information, ob ein Wort darin vorkommt (1) oder nicht (0). Aus Tabelle 1 kann nun in Abhängigkeit der Leserichtung entweder ein Vektor pro Token oder ein Vektor pro Dokument ausgelesen werden.

Tabelle 1. Eine Token-Dokument Matrix

Dokument **Token**	**Illuminati**	**Limit**	**Sakrileg**	**Der Schwarm**	**Pattern Recognition**
Verschwörung	1	1	1	1	0
Wissenschaft	1	1	1	1	1
Kunst	1	0	1	0	0
CERN	1	0	0	0	0
Illuminati	1	0	1	0	0
Weltraum	0	1	0	1	0
Flutwelle	0	0	0	1	0
…					

Um die Suchanfrage *Verschwörung* AND *Wissenschaft* AND NOT *Illuminati* auszuwerten, genügt es, die logischen Operationen AND und NOT auf die Vektoren der Tokens korrekt anzuwenden:

11110 AND 11111 AND NOT 10100
= 11110 AND 11111 AND 01011
= 01010

Die Suchergebnisse für die genannte Suchanfrage sind somit *Limit* und Der *Schwarm*.

So simpel und effektiv das auf den ersten Blick zu sein scheint, so unpraktisch ist es in der Realität (zumindest in diesem Szenario). Leider werden die Grenzen des booleschen IR deutlich, wenn man das Beispiel mit realistischen Zahlen, wie dem aktuellen Bücherangebot von Amazon.com, anreichert: Für eine Datenbank mit 250000 Büchern und einem Gesamtwortschatz von 100000 Wörtern (hier kann davon ausgegangen werden, dass ein Wort einem Token entspricht) müsste, sofern mit einem Bit Speicherplatz pro Eintrag gerechnet wird, eine Tabelle von etwa 2,91GB Größe durchsucht werden. Zunächst erscheint diese Größe (für eine Amazon Datenbank) noch akzeptabel, doch sind durch diesen Vorgang noch lange nicht die eigentlichen Textstellen gefunden, sondern lediglich die Information, in welchen Büchern die gesuchten Wörter vorkommen. Um nun auch schnell die einzelnen Textstellen zu finden, müssten die Dokumente in viel kleinere Dokumente, mit beispielsweise nur zwei Seiten Text, unterteilt werden. Das hätte zur Folge, dass die Größe der Matrix um einen dreistelligen Faktor skaliert würde. Für Bücher mit durchschnittlich 400 Seiten ergäbe das eine Indexierungstabelle mit einer Größe von mehr als 580GB.

Durch die *invertierte Indexierung*, eins der wichtigsten Konzepte des Information Retrieval, kann die Größe der Indexierungstabelle drastisch reduziert werden. Um den Grundgedanken, der hinter dieser Technik steht, nachvollziehen zu können, reicht es folgende Überlegung anzustellen: Wenn auf zwei Seiten Text (in einem Dokument) nur maximal 1000 Wörter stehen, dann sind automatisch in jedem Vektor (eines Wortes) 99000 Elemente 0. Es ist also möglich mindestens 99% des Speicherplatzes einzusparen, wenn nur die Felder mit dem Wert 1 gespeichert werden. Genau das passiert bei der invertierten Indexierung: Für jedes Wort, welches in der Datenbank vorkommt, wird gespeichert, in welchen Dokumenten es enthalten ist. Dazu muss jedes Dokument eindeutig identifizierbar sein und wird deshalb mit einer systeminternen Dokumenten-ID versehen. Die Wörter werden alphabetisch in der invertierten Indextabelle gespeichert, damit sie nicht mehrfach darin vorkommen.

Wie angekündigt werden nun die drei vorgestellten Phasen des Datenaufbereitens beim Information Retrieval an einem großen Beispiel (Tabelle 2) demonstriert. Es sollen schrittweise drei unterschiedliche Dokumente in eine invertierte Indextabelle eingepflegt werden.

Tabelle 2. Information Retrieval Beispiel

Dokument 1	Dokument 2	Dokument 3
<html> <body> Helium Chemisches Element, entdeckt durch den amerikanischen For-scher Bob Mond </body> </html>	Ein Forscher wird in seinem Schweizer Labor ermordet aufgefunden. Er wurde äußerst unsanft zu Boden befördert**.**	Mai 2025: Die Versorgung der Erde scheint gesichert, seit Amerika auf dem Mond das Element Helium-3 fördert.

1. Identifikation der Dokumente und Auswahl der Vorverarbeitung

Dokument 1: HTML	Dokument 2: Text	Dokument 3: Text

2. Vorverarbeitung: Dokumente in Einheitliches Format bringen

Helium Chemisches Element, entdeckt durch den amerikanischen For-scher Bob Mond	Ein Forscher wird in seinem Schweizer Labor ermordet aufgefunden. Er wurde äußerst unsanft zu Boden befördert.	Mai 2025: Die Versorgung der Erde scheint gesichert, seit Amerika auf dem Mond das Element Helium-3 fördert.

a) Vorverarbeitung: Tokenisierung und stemming (Beispielhaft)

helium	den	ein	er	mai	amerika
chemie	amerika	forscher	wurde	2025	auf
element	forscher	wird	äußerst	die	dem
entdecken	bob	in	unsanft	versorgung	mond
durch	mond	seinem	zu	der	das
		labor	boden	erde	element
		mord	förden	scheinen	helium3
		finden		sicher	fördern
				seit	

b) Vorverarbeitung: Stoppwörter entfernen

helium	amerika	forscher	äußerst	mai	amerika
chemie	forscher	labor	unsanft	2025	mond
element	bob	mord	boden	versorgung	element
entdecken	mond	finden	förden	erde	helium3
				scheinen	fördern
				sicher	

3. Invertierte Indexierung: Dokument 1

Token	Dokumente in denen das Token enthalten ist
amerika	1
bob	1
chemie	1
element	1

Token	Dokumente in denen das Token enthalten ist
entdecken	1
forscher	1
helium	1
mond	1

Invertierte Indexierung: Dokument 2 und Dokument 3

Token	Dokumente in denen das Token enthalten ist
2025	3
amerika	1, 3
äußerst	2
bob	1
boden	2
chemie	1
element	1, 3
entdecken	1
erde	3
finden	2
forscher	1, 2

Token	Dokumente in denen das Token enthalten ist
fördern	2, 3
helium	1
helium3	3
labor	2
mai	3
mond	1, 3
mord	2
scheinen	3
sicher	3
unsanft	2
versorgung	3

Dieses Kapitel gab eine kurze Einführung ins IR. Es wurden die drei vorbereitenden Phasen Identifikation, Daten-spezifische Vorverarbeitung und Indexierung vorgestellt und an Beispielen erläutert. Außerdem wurde die invertierte Indexierung als wichtiges Konzept des IR vorgestellt und ebenfalls an einem Beispiel gezeigt. Mit diesen Grundlagen soll der Leser ein Gefühl für die Prozesse und Abläufe des Information Retrieval bekommen haben, um auf die Fragestellungen des Cross Language Information Retrieval, um welches es im nächsten Kapitel gehen soll, vorbereitet zu sein.

2 Cross Language Information Retrieval

*"*Cross Language Information Retrieval allows the user to state their query in one language, and retrieve documents in another [1] *"*

Die Schwierigkeit, die beim Cross Language Information Retrieval (CLIR) gegenüber dem klassischen IR hinzukommt, ist, dass die Suchanfrage in einer anderen Sprache sein kann als die Dokumente, die es zu finden gilt. Um dieser Schwierigkeit zu begegnen, gibt es im Wesentlichen drei Ansätze: (1) Die Suchanfrage wird in Echtzeit in alle Sprachen übersetzt. Anschließend wird für jede Sprache eine eigene Suchanfrage gestartet. (2) Alle Dokumente werden bei der Indexierung in alle

[1] [Abusalah, 2005]

möglichen Sprachen, in der die Suchanfragen verfasst sein kann, übersetzt, um dann klassisches IR in der Sprache der Suchanfrage zu betreiben. (3) Alle Dokumente und Suchanfragen werden in eine *Hauptsprache* übersetzt. Das kann eine natürliche Sprache (wie Deutsch, Englisch, Chinesisch) oder ein abstrakter (sprachunabhängiger) Konzeptraum sein. Eine Schwierigkeit, die all diese Ansätze gemeinsam haben, ist das Erkennen der Sprache. Bevor also die unterschiedlichen Probleme der drei Ansätze beleuchtet werden, geht es zunächst in Abschnitt 2.1 um das Identifizieren von Sprachen.

2.1 Identifizieren von Sprachen

Da beim Cross Language Information Retrieval verschiedensprachige Dokumente durchsucht werden, ist es unumgänglich die Sprache eines jeden Dokuments einmal zu identifizieren, um nicht Gefahr zu laufen in deutschen Texten polnische Suchanfragen (oder allgemein in Texten auf Sprache A, Suchanfragen der Sprache B) zu verarbeiten. Dieser Schritt ist beim Indexieren der Dokumente nötig, spätestens jedoch beim Verarbeiten der Suchanfragen oder beim Übersetzen der Texte.

Die verschiedenen Algorithmen zur Sprachidentifikation in elektronischen Dokumenten basieren letztlich auf ein und demselben Prinzip: Zeichenfolgen im zu identifizierenden Text werden mit Zeichenfolgen aus einem vorher trainierten System verglichen. Dieses enthält Informationen über die Häufigkeitsverteilung bestimmter Zeichenfolgen aller zu erkennenden Sprachen. Naheliegend ist, dass die im trainierten System enthaltene Sprache mit der größten Ähnlichkeit zum vorliegenden Text auch die Sprache des Textes ist. Die Unterschiede der verschiedenen Spracherkennungs-algorithmen liegen also hauptsächlich in dem Trainieren des Systems und in den Bewertungskriterien für Ähnlichkeit von Zeichenfolgen.

Innerhalb des Europäischen Sprachraums werden heute vorwiegend wortbasierte Ansätze zur Sprachidentifikation benutzt, dabei kennt[2] das trainierte System häufig vorkommende Worte und Wortformen aller Sprachen und deren Frequenz (durchschnittliche Häufigkeit des Auftretens innerhalb eines regulären Textes). Das Training eines solchen Systems ist relativ aufwändig, da es halbautomatisch erfolgt. Besonders bei Sprachen mit starker Flexion müssen die Texte, mit denen das System trainiert wird, sehr lang sein und die automatisch als Worte klassifizierten Zeichenketten manuell auf Richtigkeit überprüft werden. Dennoch wird dieser wortbasierte Ansatz dem weniger umständlichen, auf dem Abgleichen von Bytefolgen basierten Ansatz vorgezogen, weil (1) für die meisten Sprachen bereits verfügbare, trainierte Systeme vorliegen und (2) so die Erkennung von sehr ähnlichen Sprachen verbessert wird.

Um auch Sprachen zu identifizieren, für die kein trainiertes System verfügbar ist oder für die wortbasierte Algorithmen nicht anwendbar sind (z.B. Asiatische Sprachen), kann das System auch mit Bytefolgen statt mit Worten trainiert werden, was oft als N-Gramm-Technik bezeichnet wird. Für die meisten Anwendungsfälle ist dieser Ansatz ohnehin ausreichend. Nur bei Sprachen mit sehr ähnlichen *Byte-N-*

[2] Für die meisten Sprachen gibt es Stopwortlisten mit den häufigsten Worten. Diese werden meist für die Identifikation einer Sprache eingesetzt.

Grammen, was durch gleiche Wortstämme innerhalb ähnlicher Sprachen zu Stande kommt, kann es zu Fehlern kommen.

Keine Schwierigkeiten für die Sprachenidentifizierung bereiten Standarddokumente einer Länge von mehr als 20 Wörtern, die regulären Text enthalten - d.h. sie enthalten zumindest einige gängige Funktionswörter oder sonstige hochfrequente Wortformen. Hier liegen die Erkennungsraten aller bekannten Algorithmen über 99%, wenn von der Unterscheidung extrem eng verwandter Sprachen abgesehen wird.[3,4]

2.2 Probleme und Methoden beim CLIR

Beim Cross Language Information Retrieval gibt es, wie schon erwähnt, drei grundsätzlich verschiedene Herangehensweisen. In den folgenden Abschnitten wird auf die einzelnen Ansätze eingegangen. Außerdem werden Probleme, die sie jeweils mit sich bringen, diskutiert.

2.2.1 Übersetzen der Suchanfrage

In den meisten CLIR Systemen wird die Suchanfrage in die Sprachen der zu durchsuchenden Dokumente übersetzt. Anschließend wird klassisches Information Retrieval durchgeführt. Mit Hilfe von Wörterbüchern können Wörter direkt von einer Sprache in eine andere übersetzt werden. In der Literatur ist in Zusammenhang mit der Übersetzung bei CLIR oft von der Verwendung multilingualer Thesauri die Rede. Ein Thesaurus ist ein Wortnetz, dessen Begriffe durch Relationen miteinander verbunden sind. Multilinguale Thesauri enthalten Äquivalenzrelationen zwischen Begriffen in unterschiedlicher Sprache.[5] Mit Hilfe dieser Informationen können verschiedene Wörter, die semantisch jedoch äquivalent sind, zusammengefasst werden. Das hat zur Folge, dass ohne Qualitätseinbußen mehr relevante Suchergebnisse gefunden werden können.

Das größte Problem beim Übersetzen einer Suchanfrage rührt aus ihrer Länge. Eine durchschnittliche Suchanfrage hat eine Länge von einem bis maximal fünf Wörtern.[6] Doch bereits bei einer Länge von unter 20 Wörtern, ist eine zuverlässige Identifikation, wie aus Abschnitt 2.1 hervorgeht, problematisch. Viel problematischer als das reine Identifizieren kurzer Suchanfragen ist jedoch noch das tatsächliche Übersetzen.

Die meisten Wörter haben multiple, teilweise weit auseinandergehende Bedeutungen. Die meisten Bedeutungen können wiederum durch unterschiedliche Wörter der Zielsprache ausgedrückt werden. Diese Eigenschaft nennt man Ambiguität (Mehrdeutigkeit). Wegen ihr ist es ohne Kontextinformationen besonders schwer, aus der Fülle an Übersetzungsmöglichkeiten die richtige Übersetzung zu wählen.

Es gibt verschiedene Verfahren zur Disambiguierung. Bereits in Kapitel 1 wurde die Kookkurrenzanalyse, bei der es darum geht zu ermitteln wie häufig Begriffe innerhalb

---

[3] [Langer]
[4] ausführliche Behandlung der Sprachidentifizierung in [Manning, 2009]
[5] [Schöller, 2008]
[6] [Tatham, 2009]

eines kontextuellen Rahmens zusammen auftreten, erwähnt. Viele Verfahren versuchen mit der Kookkurrenzanalyse, aus der Suchanfrage auf einen Kontext zu schließen, um die am wenigsten in diesen Kontext zu passenden Übersetzungen auszuschließen. Welche Übersetzung wann ausgeschlossen wird hängt von den Wahrscheinlichkeiten ab, die einem System für das gemeinsame Auftreten der gesuchten Wörter vorliegen. Problematisch wird dieses Verfahren dann, wenn der Benutzer eben genau diese ausgeschlossenen Übersetzungen gebraucht hätte. Das kann schnell passieren, wenn das Informationsbedürfnis des Benutzers sehr speziell ist.

Eine weitere Möglichkeit der Disambiguierung besteht darin, wegen des grammatikalischen Kontexts einer Suchanfrage bestimmte Bedeutungen auszuschließen. Das setzt voraus, dass die Suchanfrage mindestens Teilsätze beinhaltet und das ist in der deutschen Sprache nur in sehr wenigen Fällen möglich. In der englischen Sprache macht dieser Ansatz jedoch durchaus Sinn, da die Bedeutung gleich geschriebener Worte oft durch ihren grammatikalischen Kontext eingeschränkt werden kann.

(the) fear (of sth.)	-	die Furcht (vor etw.)
to fear (sth. / that…)	-	(etw.) befürchten

Steht, wie im vorrangehendem Beispiel das Wort *the* vor dem Wort *fear*, kann bereits eine große Anzahl der Bedeutungen ausgeschlossen werden.

Verbreitet ist auch die Anwendung von sogenanntem Query-Structuring, bei dem alle Übersetzungsvarianten eines Terms als Synonyme aufgefasst und in der Anfrage durch geeignete Operatoren miteinander verknüpft werden.[7]

Zum Abschluss dieses Abschnitts soll noch auf zwei Dinge hingewiesen werden. Es gibt viele unterschiedliche Wörterbücher und Thesauri. Die Qualität des Wörterbuchs ist entscheidend für eine gute Übersetzung einer Suchanfrage. Mit der Verwendung von so genannten Phrasen-Wörterbüchern können - ohne großen Aufwand - ähnliche Ergebnisse wie durch die vorgestellten Lösungsansätze erzielt werden.

2.2.2 Übersetzen der Dokumente

Manchmal ist es nötig, dass alle Dokumente in allen Sprachen, in denen die Suchanfrage gestellt werden kann, verfügbar sind. Hierbei muss man allerdings einige Einschränkungen in Kauf nehmen. Entweder ist die Anzahl der durchsuchbaren Dokumente und der benutzten Sprachen soweit eingeschränkt, dass eine manuelle Übersetzung der Dokumente erfolgen kann und Sinn macht oder es wird auf maschinelle Übersetzung zurückgegriffen mit zum Teil starken Einbußen bei der Qualität der übersetzten Texte. Für beide Möglichkeiten gibt es reale Szenarien. Meistens werden diese Übersetzungen jedoch nicht mit dem Hintergrund des Verbesserns des Information Retrieval angefertigt, sondern aus anderen Gründen benötigt. Genau betrachtet handelt es sich bei diesem Ansatz nicht um CLIR, da beim eigentlichen IR keine oder nur sehr wenig Cross-Language-Funktionalität zum Einsatz kommt. Diese wird in vorherigen Arbeitsschritten, wie etwa dem Indexieren, benötigt. Für den Großteil aller Sprachpaare und somit für die praktische Nutzung gilt, dass die vermeintlichen Vorteile, die sich für das IR ergeben, durch die noch

[7] [Schöller, 2008]

nicht ausreichend ausgereifte Übersetzungstechniken wett gemacht werden und somit ignoriert werden können.

Das maschinelle Übersetzen von Dokumenten ist eine eigene, sehr große Disziplin deren Entwicklung maßgebende Auswirkungen auf das CLIR hat.

2.2.3 Suchanfragen und Dokumente in einheitliche Sprache übersetzen

a) Handelt es sich bei der einheitlichen Sprache um eine gebräuchliche Sprache, dann kann dieser Ansatz als Mischform der beiden vorherigen Ansätze unter 2.2.1 und 2.2.2 betrachtet werden. Die Vorteile dieser Technik sind vielversprechend. Würde z.B. Englisch als einheitliche Sprache eines Systems gewählt, so wäre es möglich, alle IR Werkzeuge der englischen Sprache zu nutzen. Darüber hinaus wäre die Indexierungstabelle um ein Vielfaches kleiner. Auch bei schwer zu verarbeitenden Sprachen wie Finnisch, Deutsch oder asiatischen Sprachen wäre es einfacher, die Suchanfrage dem richtigen Ergebnis zuzuordnen. Es bleibt jedoch die Problematik, dass beim automatischen Übersetzen Fehler passieren, die, ähnlich wie unter 2.2.1 und 2.2.2, so schwerwiegend sind, dass die Vorteile kaum Gewicht haben. Ein Lichtblick ist jedoch, dass die maschinelle Übersetzung in bestimmte Sprachen (1) rasante Fortschritte macht und (2) einfacher ist als die Übersetzung in andere Sprachen. Beispielsweise ist es aufgrund der grammatikalischen Strukturen einfacher, einen Text vom Deutschen ins Englische zu übersetzen als umgekehrt. Werden diese Eigenschaften ausgenutzt und als einheitliche Sprache eine besonders günstige gewählt, so könnte dieser Ansatz sich in Zukunft durchsetzen.

b) Eine ganz anderes Verfahren versucht *sowohl Dokumente als auch Anfragen in einen sprachunabhängigen Konzeptraum zu transferieren und ohne aufwendige Übersetzungsverfahren auszukommen.*[8] Man nennt dieses Verfahren Latent Semantic Indexing (LSI). Beim LSI wird in den zu durchsuchenden Dokumenten nach sogenannten Konzepten[9] gesucht. Ohne einzelnen Begriffen Bedeutungen zuzuordnen, kann in einem solchen Konzeptraum ein Zusammenhang der Wörter Auto, Wagen, Karre, car, voiture und samochud festgestellt und ausgenutzt werden. Das Verfahren beruht auf der Theorie, dass durch eine Singulärwertzerlegung der Daten die Termwertfrequenz angenähert und als maßgebende Information bei der Sprachrepräsentation verwendet werden kann. Das hat zur Folge, dass die Vorteile des monolingualen IR ausgenutzt werden, ohne auf Mehrsprachigkeit zu verzichten. Dieses Verfahren scheitert in der Praxis jedoch daran, dass ein solcher Konzeptraum bislang nicht zuverlässig und nur sehr aufwändig berechnet werden kann. Für die Berechnung ist ein *multilingualer Sprachkorpus*[10] notwendig. Außerdem beträgt der mathematische Aufwand der Singulärwertzerlegung $O(n^2 * k^3)$.

2.3 Fazit

Die Schwierigkeit der Identifikation von Sprachen kann als weitestgehend gelöstes Problem angesehen werden. Für alle geläufigen Sprachen gibt es hinreichend gut funktionierende Heuristiken, wie das nutzen von Stopwortlisten oder die n-gram-

[8] [Schöller, 2008]
[9] Konzepte kann man sich abstrakt als wiederkehrende Muster in Daten vorstellen.
[10] Äquivalente, richtig zugeordnete Daten in allen Sprachen

Verfahren. Da sich nur eine der drei vorgestellten CLIR Herangehensweisen, nämlich das Übersetzen der Suchanfrage in alle anderen Systemsprachen, zum heutigen Zeitpunkt logistisch und technisch für ein großes Einsatzgebiet realisieren lässt, ist dieses auch das zurzeit beste Verfahren. Die Ergebnisse, die dieses Verfahren liefert, sind brauchbar, haben aber mit der sprachlichen Ambiguität und der nicht ganz ausgereiften Technik der maschinellen Sprachübersetzung zu kämpfen. Auch ist die Qualität der Ergebnisse von Sprachpaar zu Sprachpaar unterschiedlich. Die anderen beiden Verfahren haben jeweils Ihre Vor- und Nachteile. Das Übersetzen aller Daten in alle möglichen Sprachen lässt sich manuell nur für ein sehr stark eingeschränktes Einsatzgebiet realisieren und hat darin Ihren Hauptnachteil. Aber auch mit voranschreitender Technik bezüglich der automatischen Übersetzung wäre die große Kapazität, welche dieses Verfahren in Anspruch nimmt, ein Nachteil. Ein Vorteil dieser Technik ist hingegen die zweifelsfrei beste Qualität der Suchergebnisse. Liegt ein multilingualer Korpus vor, so kann mit derselben Qualität von Suchergebnissen gerechnet werden wie bei klassischem monolingualen IR. Das letze vorgestellte Verfahren hat viel Potential und ist bezogen auf die Ressourcen das beste Verfahren. Es bleibt abzuwarten, ob die Qualität der Ergebnisse mit der Qualität der anderen Verfahren vergleichbar sein wird und ob effizientere Algorithmen zur Transformation in einen sprachunabhängigen Konzeptraum gefunden werden können.

3 Literaturverzeichnis

Langer, S.: Grenzen der Sprachenidentifizierung. CIS. Universität München.

Manning, C. D., Raghavan, P., & Schütze, H. (2009): An Introduction to Information Retrieval, *Cambridge University Press*

Oard, D. W. (2009): Multilingual Information Access. University of Maryland, College Park, MD 20742 , USA.

Oard, D. W. (2006): Transcending the Tower of Babel: Supporting Access to Multilingual Information with Cross-Language Information Retrieval. University of Maryland, College Park, MD 20742 , USA.

Schöller, K. (2008). Diplomarbeit: Sprachübergreifendes Retrieval von ähnlichen Dokumenten aus großen Textkollektionen. Bauhaus-Universität Weimar.

Tatham, M. (2009). *www.hitwise.com.* Abgerufen am 27. 01 2010: Google Received 72 Percent of U.S. Searches in January 2009: http://image.exct.net/lib/fefc1774726706/d/1/SearchEngines_Jan09.pdf

Abusalah, M., Tait, J., & Oakes, M. (2005). Literature Review of Cross Language. *World Academy of Science, Engineering and Technology 4* .

Stock, W. G. (2007). *Information Retrieval.* München: Oldenbourg Wissenschaftsverlag GmbH.