

Theory and Applications of Similarity Detection Techniques

Dissertation in fulfillment of the requirements for the academic degree
Doctor of Technical Sciences (Dr. Techn.) in Computer Science

Submitted by
Bilal Zaka

Institute for Information Systems and Computer Media (IICM)
Graz University of Technology
A-8010 Graz, Austria
February, 2009

First reader: Univ.-Prof. Dr. Frank Kappe
Second reader: Univ.-Prof. Dr. Klaus Tochtermann

I hereby certify that the work reported in this dissertation is my own and that work performed by others is appropriately cited.

Ich versichere hiermit, diese Dissertation selbständig verfaßt, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfsmittel bedient zu haben

Bilal Zaka

Abstract:

The measurement of similarity between different objects is the fundamental function of any information retrieval, management, or data mining application. There are a number of ways to compute similarity or dissimilarity among various object representations. In a simplified classification these can be categorized as distance based, geometric, structural, feature and knowledge based techniques. These techniques are used depending on the characteristics of data and scope of application. Examples of information systems that are making collective use of similarity measure of different types are few and far between. A lot remains to be done for the realization of the semantics-aware system. The majority of the current systems in this realm exploit only pattern discovery techniques based on basic similarity measures. This dissertation explores the architecture and design of a framework that supports more elaborate and enhanced systems. A number of case studies have been included in the research to demonstrate the various aspects of this framework.

This dissertation provides an introduction to different types of similarity detection techniques, possible enhancements and their applications in various public and corporate environments. The primary objective of this work is to help improve conventional information management and retrieval tools by adding to them the practical elements of semantic and distributed processing. The initial part of the dissertation describes the broader categories of similarity detection and commonly used techniques along with an introduction to the data processing approach. The following parts of the dissertation cover the case studies to exemplify the extended use and applications of these techniques. These parts shed light on the use of enhanced similarity detection approaches in the plagiarism detection and IPR areas. The work suggests a commonly accessible platform that allows the use and integration of different similarity detection services. The findings of the research are presented in the form of a successful implementation of a collaborative plagiarism detection and prevention network. In the second set of experiments, a successful integration of such services is described to aid personalized content delivery. It illustrates the use of similarity detection in semantic media adaptation and user interest profiling. Finally the work covers the applications of similarity measurement techniques for content organization, re-usability and objects de-duplication in heterogeneous data collections.

Kurzfassung

Die Messung von Ähnlichkeiten zwischen verschiedenen Objekten ist eine elementare Funktion zur Suche, Verwaltung oder Auswertung von Daten. Es gibt verschiedene Möglichkeiten diese Ähnlichkeit zu messen. In einer einfachen Klassifikation wird auf Abstandsmessung, Geometrie und Struktur oder merkmals- und wissensbasierte Verfahren gesetzt, wobei die verwendete Technik von den Datenmerkmalen und dem Anwendungsgebiet abhängt. Es können jedoch nur vereinzelt Beispiele für Informationssysteme gefunden werden, die mehrere dieser unterschiedlichen Verfahren zur Ähnlichkeitsmessung einsetzen. Die Entwicklung von aktuellen Systemen, die auf die Semantik von Objekten setzt, ist noch nicht weit fortgeschritten, da die Ähnlichkeiten von Mustern größtenteils mit einfachen Methoden ausgewertet wird. In dieser Doktorarbeit wird das Grundgerüst einer Architektur und eines Designs vorgestellt, um detaillierte und verbesserte Systeme entwerfen zu können. Die unterschiedlichen Seiten dieses Gerüsts werden bei der Untersuchung von mehreren Fallstudien gezeigt.

In dieser Doktorarbeit werden Techniken zur Ähnlichkeitsbestimmung, mögliche Verbesserungen, und Anwendung in öffentlichen und kommerziellen Umgebung beschrieben. Ziel ist es, bestehende Werkzeuge zur Verwaltung und Suche von Informationen mit Hilfe von praktischen Teilen der Semantik und verteilter Verarbeitung zu unterstützen. Der erste Teil dieser Doktorarbeit beschreibt das umfangreiche Feld der Ähnlichkeitsbestimmung und häufig genutzten Techniken. Die weiteren Teile erläutern den Gebrauch und die Anwendung dieser Techniken. Weiters werden Ansätze für verbesserte Verfahren zur Ähnlichkeitsbestimmung bei der Erkennung von Plagiaten im Bereich des Urheberrechts. In dieser Arbeit wird eine gemeinsame Plattform zur Integration von verschiedenen Diensten zur Bestimmung von Ähnlichkeiten vorgeschlagen. Das Ergebnis der Untersuchung wird anhand der Implementierung eines Netzwerkes zur Bestimmung und erfolgreichen Verhinderung von Plagiaten gezeigt. Im zweiten Teil der Experimente wird ein Dienst zum Finden von personalisierten Inhalten beschrieben. Er basiert auf verschiedenen Verfahren zur Ähnlichkeitsbestimmung und berücksichtigt semantische Informationen und das Profil des Benutzers. Zum Abschluss der Arbeit wird die Anwendung von Techniken zur Ähnlichkeitsbestimmung für die Organisation, Wiederverwendung und die Deduplizierung von Objekten in heterogenen Datenmengen beschrieben.

Acknowledgement

During my stay and work in Austria I was facilitated by many people and I would like to express my gratitude for their support. First of all, I would like to thank my advisor Prof. Dr. Frank Kappe for his continuous support right from the beginning till the end of my doctoral studies. Prof. Kappe has supported me not only academically but also provided the moral support and freedom to explore different ideas. I am thankful to Prof. Dr. Hermann Maurer for accepting and inducting me in IICM. His insight, guidance and ideas really helped me to structure and write this dissertation. For this I am sincerely grateful. I am also indebted to Prof. Dr. Klaus Tochtermann for being part of my dissertation evaluation committee and taking the role of second reader despite his busy schedule.

I am extremely grateful to all the colleagues and friends at IICM who helped me during the course of my work. I would like to say “Thank You!” to: Narayanan Kulathuramaiyer, for helping me in my research activities, reviewing and commenting on my work on such a short notice. Maria-Luise Lampl, for providing a very friendly working environment and an always needed and much appreciated administrative assistance; Christian Safran, for collaborative research work, discussions, and also for providing assistance in numerous non-academic affairs; Mensur Celikovic, for being such a cheerful, helping and encouraging friend. I am also grateful for the cooperation provided by Michael Erwin Steurer, Denis Helic, Christian Guetl, Karl Trummer and Josef Kolbitsch. I also wish to express my gratitude to Salman Khan and Tanvir Afzal for their encouragements.

I am very obliged to the Higher Education Commission of Pakistan for awarding me with the scholarship to study at Graz University of Technology. Similarly I extend my appreciation to the ÖAD for their administrative support. Partial financial support by Styria Media AG, in conjunction with the Endowment Professorship for Innovative Media Technology is also gratefully acknowledged and appreciated. I also acknowledge the opportunity provided by Shell Knowledge Innovation Design initiative, which allowed me to work with data coming from collaborative network of a large corporation.

I owe the biggest acknowledgment to my parents, siblings, wife and daughter for their understanding, encouragements, and support while I am occupied in my research work.

Finally I humbly thank God for providing me with such a great opportunity to learn.

TABLE OF CONTENTS

1. INTRODUCTION	15
1.1 BACKGROUND AND OBJECTIVES	15
1.2 METHODOLOGY AND STRUCTURE	16
1.3 SCIENTIFIC CONTRIBUTIONS	18
2. SIMILARITY MEASURES	19
2.1 SIMILARITY DETECTION IN INFORMATION RETRIEVAL AND DATA MINING	19
2.2 CHARACTERIZATION OF SIMILARITY DETECTION	20
2.2.1 <i>Distance-Based Similarity Measures (Metric Axioms)</i>	20
2.2.1.1 Minkowski Distance.....	20
2.2.1.2 Manhattan/City block distance	21
2.2.1.3 Euclidean distance.....	21
2.2.1.4 Chebyshev distance.....	21
2.2.1.5 Jaccard distance	21
2.2.1.6 Dice's Coefficient.....	21
2.2.1.7 Cosine similarity	22
2.2.1.8 Hamming distance.....	22
2.2.1.6 Levenshtein Distance.....	22
2.2.1.9 Soundex distance	22
2.2.2 <i>Feature-Based Similarity Measures</i>	22
2.2.2.1 Contrast Model.....	22
2.2.3 <i>Probabilistic Similarity Measures</i>	23
2.2.3.1 Maximum likelihood estimation (MLE)	23
2.2.3.2 Maximum a posteriori (MAP) estimation	23
2.2.4 <i>Extended/Additional Measures</i>	24
2.2.4.1 Similarity measures based on fuzzy set theory	24
2.2.4.2 Similarity measures based on graph theory	24
2.3 INFORMATION RETRIEVAL MODELS.....	25
2.3.1 <i>Set-theoretic Models</i>	25
2.3.1.1 Boolean model	25
2.3.1.2 Fuzzy set based model.....	25
2.3.1.3 Extended Boolean model	25
2.3.2 <i>Algebraic Models</i>	25
2.3.2.1 Vector space model.....	26
2.3.2.2 Latent Semantic Analysis based model	26
2.3.2.3 Neural Networks	26
2.3.3 <i>Probabilistic Models</i>	26
2.3.3.1 Inference network.....	27
2.3.3.2 Belief network	27
2.3.4 <i>Knowledge based Models</i>	27
2.3.5 <i>Structure based Models</i>	28
2.4 CONTENT PROCESSING	28
3. PLAGIARISM AND IPR	31
3.1 INTRODUCTION	31
3.1.1 <i>Defining Plagiarism</i>	31
3.1.2 <i>Impact</i>	33
3.2 RESPONSE OF ACADEMIC INSTITUTIONS.....	34
3.3 WHY PLAGIARISM DETECTION IS IMPORTANT.....	37

3.4	DETECTING PLAGIARISM.....	39
3.4.1	<i>Document source comparison</i>	39
3.4.2	<i>Manual search of characteristic phrases</i>	41
3.4.3	<i>Stylometry</i>	41
3.5	AVAILABLE TOOLS	44
3.5.1	<i>Turnitin</i>	44
3.5.2	<i>SafeAssignment</i>	45
3.5.3	<i>Docol@c</i>	46
3.5.4	<i>Urkund</i>	48
3.5.5	<i>Copypatch</i>	49
3.5.6	<i>WCopyfind</i>	49
3.5.7	<i>Eve2 (Essay Verification Engine)</i>	49
3.5.8	<i>GPSP - Glatt Plagiarism Screening Program</i>	49
3.5.9	<i>MOSS - a Measure of Software Similarity</i>	49
3.5.10	<i>JPlag</i>	49
3.6	UNEXPECTED RESULTS.....	50
3.7	ADVANCED TECHNIQUES	57
3.8	PROBLEMS AND VISIONS	62
3.8.1	<i>Access to deep web</i>	62
3.8.2	<i>Plagiarism detection of multimedia contents</i>	62
3.8.3	<i>Semantic plagiarism detection</i>	63
3.8.4	<i>Intrinsic characteristics check</i>	63
3.8.5	<i>Cross language checking</i>	63
3.9	ENHANCEMENTS IN PLAGIARISM DETECTION SYSTEMS	65
3.10	SERVICE ORIENTED COLLABORATIVE PLAGIARISM DETECTION AND PREVENTION	65
3.11	CONCEPTS BEHIND SERVICE ORIENTED ARCHITECTURE	67
3.11.1	<i>Web service model</i>	67
3.11.2	<i>Mashup of search and analysis web services</i>	69
3.11.3	<i>Collaborative authoring, indexing & searching – Access into the deep web</i>	71
3.11.4	<i>Service publication, discovery and access mechanism</i>	72
3.12	CPDNET IMPLEMENTATION	72
3.12.1	<i>Towards a semantic plagiarism detection service</i>	74
3.12.1.1	<i>Fingerprint normalization into generic signatures</i>	74
3.13	RESULTS OF CPDNET PROTOTYPE.....	75
3.13.1	<i>Alerting service</i>	78
3.14	PLAGIARISM IN VIRTUAL WORLDS	79
3.15	MAPPING OF PLAGIARISM DETECTION FROM TEXT TO THE MULTIMEDIA DOMAIN	81
3.15.1	<i>Text Based Plagiarism Detection Systems</i>	81
3.15.2	<i>Finding plagiarism in multimedia contents</i>	81
3.15.2.1	<i>GIFT (the GNU Image-Finding Tool):</i>	82
3.15.2.2	<i>isk-daemon:</i>	83
3.15.2.3	<i>LIRE (Lucene Image REtrieval):</i>	83
3.16	COLLECTION OF TEST CORPUS	83
3.16.1	<i>Object Crawler</i>	84
3.16.2	<i>Test Corpus</i>	85
3.17	SYSTEM FOR FINDING PLAGIARISM IN VISUAL OBJECTS	86
3.18	RESULTS AND SYSTEM ENHANCEMENTS.....	88
3.19	FURTHER WORK	92
3.19.1	<i>Introduction of translation and normalized signature search service</i>	92
3.19.2	<i>Addition of intrinsic characteristic checks</i>	92
3.19.3	<i>Noise reduction in plagiarism detection with domain specific searches and efficient citation checks</i>	93

3.19.4	<i>Scalability and design issues of composite applications</i>	93
3.20	CONCLUSION	94
4.	ADAPTIVE INFORMATION SYSTEMS	97
4.1	USER ADAPTIVE NEWS CONTENT DELIVERY	97
4.2	RELATED WORK	99
4.3	DESIGN OF PERSONALIZED INTERACTIVE NEWS CAST	101
4.3.1	<i>News acquisition and pre-processing</i>	101
4.3.2	<i>Portable User Modeling</i>	103
4.3.3	<i>Aggregation</i>	104
4.3.4	<i>User Interfaces</i>	105
4.3.4.1	WWW Access	106
4.3.4.2	Speech Interface	106
4.3.4.3	E-Ink	107
4.3.4.4	Video	108
4.4	SYSTEM ARCHITECTURE	108
4.5	PINC PROTOTYPE	111
4.6	SUMMARY	113
4.7	INFORMATION SUPPLY FOR KNOWLEDGE WORKERS	114
4.8	SEARCHING THE WEB FOR KNOWLEDGE ACQUISITION	114
4.9	FROM INFORMATION RETRIEVAL TO INFORMATION SUPPLY	116
4.9.1	<i>Term extraction and lexical variations</i>	117
4.9.2	<i>Determine the subject domain with the help of classification systems</i>	117
4.9.3	<i>Query expansion</i>	117
4.9.4	<i>Distributed search services</i>	118
4.9.5	<i>Result analysis with the help of similarity detection</i>	118
4.9.6	<i>Result mapping to knowledge space</i>	119
4.10	SERVICE ORIENTED MODEL	119
4.11	SUMMARY	120
4.12	OUTLOOK	121
4.12.1	<i>Utility computing</i>	122
4.12.2	<i>Workflow in web services</i>	123
4.12.2.1	Orchestration Model	124
4.12.2.2	Choreography Model	124
4.12.3	<i>Distributed orchestration</i>	124
5.	CONTENT ORGANIZATION	127
5.1	INTRODUCTION	127
5.2	CLUSTERING IN PRACTICE	128
5.3	INFORMATION PROCESSING TO LINK SIMILAR DOCUMENTS	129
5.3.1	<i>Corpus Selection</i>	130
5.3.2	<i>Crawling process</i>	130
5.3.3	<i>Indexing and Search</i>	131
5.3.4	<i>Cluster Analysis</i>	131
5.3.4.1	Hierarchical	131
5.3.4.2	Partitioning	131
5.3.4.3	Overlapping	132
5.3.4.4	Ordination	132
5.3.4.5	Model-Based	132
5.3.5	<i>Similarity measures</i>	132
5.3.6	<i>Used approach</i>	133
5.3.7	<i>Result presentation</i>	134

5.4	SYSTEM DESIGN	135
5.5	EXPERIMENTAL RESULTS.....	137
5.6	CONTENTS REUSABILITY	142
5.7	AGGREGATION OF PRESENTATIONS BASED ON TEXT AND TOPIC SIMILARITY	144
5.8	EXPERIMENTS ON THE AGGREGATION OF LEARNING OBJECTS.....	145
5.9	EVALUATION AND USABILITY OF THE SYSTEM	147
5.10	CONCLUSIONS AND WORK IN PROGRESS.....	148
6.	SUMMARY AND OUTLOOK.....	151
6.1	RESULTS AND CONCLUSIONS.....	151
6.2	FUTURE PERSPECTIVES	153

1. Introduction

The scope and domain of work covered in the dissertation

This dissertation examines the role of similarity measures in various information systems. It provides a theoretical overview of similarity detection techniques, discusses possible enhancements and shows the layered use in a number of case studies. The primary objective of this work is to help improve existing and emerging information systems by addition of contextual, structural, semantic and distributed processing in similarity checks.

1.1 Background and Objectives

The information available online has grown to a level where it is very difficult to consume it diligently without the help of various information processing, management, discovery, and filtering tools. The most vital core element of all such systems is no doubt the system's ability to measure similarity among various information segments. Explosive growth of internet usage at the organizational level and increasing means of individual user contributions are the factors behind many emerging internet applications. Surveys, literature reviews, and experimental evaluations of these systems show that the simplistic use of similarity detection in such globally authored linked systems is not enough.

This dissertation is motivated by a desire to evaluate and extend similarity measures in large internet based information systems. It highlights

- The use of context information & domain knowledge in similarity checks
- The importance of introducing semantic similarity checks in internet applications
- The use of un-conventional similarity characteristics to help matching process
- Effects of utility computing on functionalities and scalability of information systems

The following application areas were investigated as part of experimental work to pursue the above mentioned goals:

- Plagiarism detection and Intellectual Property Right protection
- User adaptive information supply environment
- Content organization

The work suggests the usefulness of a commonly accessible platform for enhanced similarity detection services. Experimental results show that successful integration of such services significantly improves existing and emerging information systems. This dissertation points towards a next generation of information systems. It suggests that in new information environments similarity measures will not have a rigid nature. They are of modular nature and allow room for modifications and can be easily molded as per system or user requirements. With an open community driven development, these services conforming to a standard set of specification can be part of any online application. Such a capability is necessary for global scalability of information systems.

1.2 Methodology and Structure

This dissertation covers the discussion of similarity measures and their use in various information systems. It provides the design and architecture of systems that makes use of layered similarity analysis. Experimental work shows the implementation details and results from a number of case studies. As mentioned in previous section a number of application areas were targeted for testing layered similarity checking. The core chapters (3, 4, 5) of this dissertation describes these case studies.

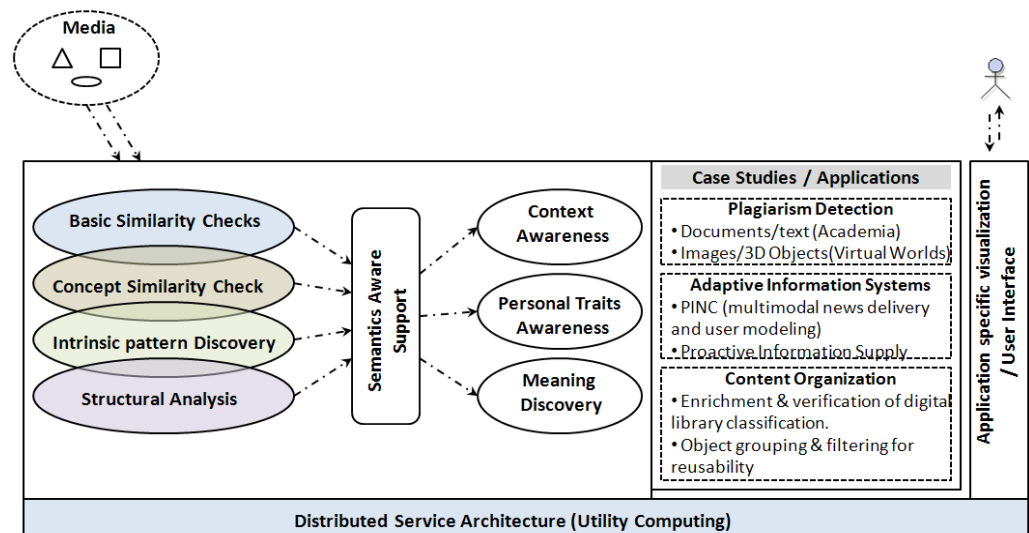


Figure 1. Overview of dissertation

Figure 1 gives an overview of the dissertation. The dissertation is based on a number of papers published in various peer reviewed conferences and journals. They come from three and a half years of research work (2006-2009), exploring and developing different information systems. Brief introduction and selected publications comprising the chapters are described as follows:

After the introductory part the second chapter of this dissertation gives an overview of similarity detection approaches in various information retrieval and management systems. It provides a general understanding of various similarity detection classes, information retrieval models and contents processing techniques.

Applications of similarity detection in IPR domain are discussed in third chapter. The chapter starts with a co-authored survey on plagiarism [Maurer et al., 2006]. It describes the existing system and methods of automated plagiarism detection. Results of survey highlight deficiencies and provide insight for enhancements. The later part of the chapter explains technical and experimental details of enhancements incorporated in plagiarism detection systems for academia and virtual worlds. These case studies are based on the contents of papers showing results of developed prototypes [Maurer and Zaka, 2007] [Zaka, 2009b] [Zaka et al., 2009b].

Adaptive information systems and user modeling are discussed in fourth chapter. Research work shows successful integration of similarity and search services aids personalized content delivery. It provides very effective content and collaborative filtering by means of service oriented computing. It further illustrates the use of similarity detection in semantic media adaptation and user interest profiling. It is mainly based on publications describing Personalized Interactive News Casting system [Zaka et al., 2007], and a book chapter explaining the use of similarity detection for adaptive news content delivery and user profiling [Zaka et al., 2009a]. It also includes the contents of a paper on service oriented information supply [Zaka and Maurer, 2007].

Fifth chapter shows applications of layered similarity checking to organize objects in heterogeneous archives. It starts with details of experiments conducted to verify a practical approach that can be used to group together and filter related documents. [Zaka, 2009a]. It further describes the work on topic-centered aggregation of presentations for learning object repurposing [Zaka et al., 2008].

Chapter 6 provides concluding remarks and a potential future direction of work.

1.3 Scientific contributions

The main contribution of this work is the use of layered similarity detection approach for semantic-aware systems with an architecture to support robust and scalable information systems.

Over the past three years, a number of experimental applications are developed that are available online. These applications include the CPDNet project which is a platform for plagiarism detection and prevention. This is an extension to existing systems in terms of added support for collaborative indexing and search, detection of paraphrased text, and image copy detection. Support for normalized text indexing and search is incorporated in Nutch/Lucene text analyzer. The normalized text (transformed to most significant vocabulary in specific syntactical sense) support is added by use of WordNet ontology. The ability of canonical term representation in similarity detection gives the possibilities of conceptual match detection.

Prototype of a Personalized Interactive News Cast (PINC) system is also developed for multimodal news delivery and user modeling. It provides a number of means for news aggregation and filtering needed for effective delivery on a number of modes for user interaction. This application provides means of building standardized user interest profiles that can be used in a number of other information delivery systems.

The use of enhanced search and indexing APIs are illustrated not only in CPDNet but their use in PINC system, content organization, clustering and filtering prototypes indicates additional application areas. The practical functions to quantify and compare the text structure and writing style characteristics are also an added feature introduced in this work.

2. Similarity Measures

Survey of similarity detection techniques

The concept of similarity is important in almost every scientific field. Detailed discussion covering all similarity measuring concepts and theories is beyond the scope of a single chapter. Thus, this chapter mainly focuses on techniques used to measure similarity in information retrieval and data mining fields.

2.1 Similarity Detection in Information Retrieval and Data Mining

The technologies of information age allowed collection of massive amount of data. The growing collection of digital information to any individual or organization is available in various formats and through various sources. Such huge information base requires a mechanism of finding material that satisfies an information need from within large collections. This process of identification of relevant information (usually documents/text) is generally termed as "Information Retrieval". However the disparate nature of information collections (especially over the web and internet) demands something more than information retrieval. In order to have a better perception of information for management and decision making, another layer of data processing is introduced. The main purpose of this processing layer is to i) extract the implicit, hidden, potentially useful information and ii) discover meaning full patterns from large raw data collections. This activity is generally termed as "Data Mining".

Measuring similarity or distance between two information entities is a core requirement for all information discovery tasks (whether IR or Data mining). Use of appropriate measures not only improves the quality of information selection but it also help reduce the time and processing costs. Several similarity measures are proposed, available and used under different applications and requirements.

2.2 Characterization of Similarity Detection

The concept of similarity is proven to be important not only in every scientific field but it has deep roots in philosophy and psychology. In classical Western philosophy where origins of concepts are discussed, the three principals of association are described as (i) Resemblance (ii) Contiguity in time or place and, (iii) Cause or effect [Hume, 1748]. The strict similarity is defined different from resemblance which requires sharing of strictly identical sensory components. In psychological work (an offshoot of philosophy) substantial work has been carried out. That work address the early understanding of similarity or resemblance which do not consider influence of cognition and knowledge (rather binary relation of identical or not). Wallach's work [wallach, 1958] extends the similarity definition to a level of "potential similarity" (considered as the first modern study on similarity). He incorporated the idea selecting or ignoring features of objects being compared for similarity. Potential similarity considers the effects of context (commonality in environmental conditions) and attention. Extrinsic features (not perceivable directly) were also added to determination of similarity assessment. One can read more about the root of similarity on referenced resources.

This work however deals more with the measure of similarity in computer science domain (information retrieval and data mining to be more specific). Similarity measure in this domain is an algorithm that determines the degree of agreement between entities. In following subsections, mathematical foundations of common techniques used to determine similarity are described.

2.2.1 Distance-Based Similarity Measures (Metric Axioms)

According to a very popular theoretical assumption, the perceived similarity can be inversely associated with the distance in some suitable feature space. In most of the cases it is considered to be a metric space. Many psychological theories assume that closer objects are more similar than the objects that are far apart. Similarity measure (s) can be derived from the distance measure (d) using $s = 1 - d$. The most common form of dissimilarity calculation refers to distance calculation in metric space. In typical information retrieval systems example of this approach include following models/methods.

2.2.1.1 Minkowski Distance

This is generic form of metric distance calculation for multidimensional data. The n norm Minkowski distance measure can be defined as the distance D_{ij} between two parts i and j as,

$$D_{ij} = \left(\sum_{l=1}^d |x_{il} - x_{jl}|^{1/n} \right)^n \quad (i)$$

2.2.1.2 Manhattan/City block distance

The Manhattan is Minkowski distance at norm value of 1. It is the measure of absolute difference between any two points. It is described as,

$$D_{ij} = \sum_{l=1}^d |x_{il} - x_{jl}| \quad (\text{ii})$$

2.2.1.3 Euclidean distance

The Minkowski distance at norm value of 2 is described as Euclidean distance. It is the most commonly used measure to determine distance between two points. It is described as,

$$D_{ij} = \left(\sum_{l=1}^d |x_{il} - x_{jl}|^{1/2} \right)^2 \quad (\text{iii})$$

2.2.1.4 Chebyshev distance

At $n \rightarrow \infty$ Minkowski distance is termed as Chebyshev distance. It represents the greatest distance between two vectors along any coordinate dimension. It is described as,

$$D_{ij} = \max_{1 \leq l \leq d} |x_{il} - x_{jl}| \quad (\text{iv})$$

2.2.1.5 Jaccard distance

The Jaccard distance measures dissimilarity between sample sets. It is complementary to the Jaccard coefficient (size of the intersection divided by the size of the union of the sample sets) and is obtained by subtracting the Jaccard coefficient from 1 [Wikipedia:Jaccard, 2008].

$$J_\delta(A, B) = 1 - J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|} \quad (\text{v})$$

2.2.1.6 Dice's Coefficient

Dice coefficient similarity measure is defined as twice the number of terms common to compared entities/strings (n_t) divided by the total number of terms in both tested strings.

$$s = \frac{2n_t}{n_x + n_y} \quad (\text{vi})$$

2.2.1.7 Cosine similarity

Cosine similarity is a popular vector based similarity measure in text mining and information retrieval. In this approach compared strings are transformed into vector space so that the Euclidean cosine rule can be used to calculate similarity. This approach is often paired with other approaches to limit the dimensionality of the vector space.

$$similarity = \cos(\theta) = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| \|\vec{B}\|} \quad (vii)$$

2.2.1.8 Hamming distance

It is considered to be the most popular measure for binary attributes. It is defined as the number of bits which differ between two binary strings i.e. the number of bits which need to be changed to turn one string into the other. For example the bit strings 1011101 and 1001001 has a hamming distance of 2bits, (two bits are dissimilar). This approach is used for exact length comparisons.

2.2.1.6 Levenshtein Distance

It is also referred to as edit distance and is a generalized form of Hamming distance. The distance between two strings is given simply by the minimum edit operations needed to convert one string into the other. The edit operations are insertion, deletion, or substitution of a single character.

2.2.1.9 Soundex distance

This distance measure is based on phonetic indexing scheme. The terms are encoded into codes according to their pronunciation. This helps in detecting matches even if there are small spelling changes.

In addition, there are a number of other distance measures in use some of which include Smith-Waterman-Gotoh distance, Jaro-Winkler distance, Matching coefficient, Overlap Coefficient, Q-gram distance [Wikipedia:SimMetrics, 2008] etc.

2.2.2 Feature-Based Similarity Measures

An alternate to distance based similarity measures was presented by Tversky (1977). The empirical evidence against the geometrical distance models provided the grounds for suggesting this model.

2.2.2.1 Contrast Model

Tversky suggested the contrast model, where similarity is computed by common features of compared entities. The entities are more similar if they share more common features and dissimilar in case of more distinctive features.

$$s(A, B) = \alpha g(A \cap B) - \beta g(A - B) - \gamma g(B - A) \quad (viii)$$

Above formula can be used to determine similarity between entities A and B. Where $g(A \cap B)$ represents the common features in A and B, $g(A-B)$ represents distinctive features of A and, $g(B-A)$ that of entity B. α, β, γ are used to determine the respective weights of associated values. In text matching area, stylometry can be considered an example of feature based similarity detection.

Stylometric analysis involves identifying patterns in documents using different structural and/or linguistic features. It is commonly used to determine authorship of documents, lately it has been successfully applied music and fine arts as well. Use of this technique makes it possible to detect common attributes between objects produced by same author. Stylometry is an ancient art, but use of sophisticated computers for statistical analysis, artificial intelligence and access to material available via Internet has enhanced its effectiveness enormously [Wikipedia:Stylometry, 2008]

2.2.3 Probabilistic Similarity Measures

In many application areas like image retrieval, face recognition, DNA analysis and, multimedia databases; the complexity of data often makes it difficult to determine exact feature, position metric for similarity relations. In order to calculate relevance among these complex data types, use of probabilistic similarity means are required. Probability density functions are used to indicate the likelihoods of certain feature values. Use of probabilistic similarity approach is considered to perform well in above mentioned special cases, however at the cost of increased computational complexity [Vasconcelos, 2004].

In general the similarity functions take probabilistic density models of compared objects as similarity function argument. Due to the computational overheads of probabilistic similarity measures, sometimes simple similarity measures (e.g. Euclidean distance between Histograms) are used to determine the probabilistic estimation.

2.2.3.1 Maximum likelihood estimation (MLE)

A commonly used method based on R. A. Fisher's approach of fitting mathematical model to given data. MLE approach arranges probability model parameters of experimental data in order to make it more likely [Wikipedia:MLE, 2008].

2.2.3.2 Maximum a posteriori (MAP) estimation

MAP estimation of likelihood is closely related to MLE estimation, however in contrast to MLE approach where only the experimental measures of data are used for estimation, MAP is a Bayesian approach where a prior available distribution is also used for estimation. It is a less commonly used method due to its complexity and often unavailability or reliability issues of a priori information sample [Wikipedia:MAP, 2008].

The underlying probabilistic density models for data representation greatly affect the accuracy of similarity or likelihood calculations. A paper analyzing the

probabilistic similarity measures for image retrievals by N Vasconcelos [Vasconcelos, 2004] shows the relationships and use of Mixture Densities, Gaussian (Normal) model, Vector Quantizer (VQs) and Histogram model.

2.2.4 Extended/Additional Measures

2.2.4.1 Similarity measures based on fuzzy set theory

Several measures of similarity are proposed based on fuzzy set theory. These measures are generally based on union and intersection operations, maximum difference, and on the differences and summation of set membership values [Pappis and Karacapilidis, 1993].

A conventional measure of fuzzy similarity is based on fuzzy numbers. For two trapezoidal fuzzy numbers,

$$\tilde{A} = (a_1, a_2, a_3, a_4), \tilde{B} = (b_1, b_2, b_3, b_4)$$

Then the degree of similarity between the trapezoidal fuzzy numbers is given as [Chen and Chen, 2003],

$$S(\tilde{A}, \tilde{B}) = 1 - \frac{\sum_{i=1}^4 |a_i - b_i|}{4} \quad (\text{ix})$$

where $S(\tilde{A}, \tilde{B}) \in [0, 1]$

2.2.4.2 Similarity measures based on graph theory

A number of similarity measures exist that are based on graph modeling and matching. Graphs provide a powerful mean of data representation, graph matching in turn is very effective in determining the relationship between various parts of data objects. Similarity measures based on graph theory are proved to be very useful in various applications of computer vision, audio content analysis and retrieval [Peng et al, 2006] and document structure analysis [Wan and Peng, 2005].

Core methods of computing graph match include graph, sub-graph isomorphism and maximum common sub-graph. In practical cases strict graph matches are not common; in order to handle noise and provide error-tolerant matching, graph edit operations with appropriate cost functions are used. The edit distance (a quantitative measure of edit operations) provides the similarity measure of graphs. The edit distance for graph g and g' with a maximum common sub-graph g'' the edit distance is given as [Bunke, 2000],

$$d(g, g') = |g| + |g'| - 2|g''| \quad (\text{x})$$

2.3 Information Retrieval Models

An information retrieval (IR) system as whole is responsible for data storage, representation, organization and easy access to desired information. The ultimate goal of the defined similarity techniques is to facilitate an information retrieval process. Similarity measures in IR processes are applied to a set of data objects, select relevant objects, produce a ranked set of relevant objects, and identify best matches. Several IR models exist that make use of various similarity measuring techniques along with appropriate data processing methodologies. Following subsections describe these models in general.

2.3.1 Set-theoretic Models

These information retrieval models are based on set theory. The data object forms sets and set theoretic operations are used to derive similarities.

2.3.1.1 Boolean model

It is the simplest form of an IR model based on set theory framework. Typical operations are AND, OR and NOT. It is referred to as exact match model. Simple Boolean retrieval model is easy to implement and computationally efficient. However it has certain drawbacks such as, complex queries are hard to construct, unavailability of ranked results and no partial matching (rather extreme out put of either logical match or no match).

2.3.1.2 Fuzzy set based model

In order to address the shortcoming of simple Boolean model with strict binary association, fuzzy set based approach is practiced in IR systems for quite some time. The information entities are assigned a degree of membership which allows the notion of marginal association with sets. This makes membership/association a gradual notion rather than binary. Further details of fuzzy IR models can be found in [Kraft et al., 1998] [Kraft et al., 2006].

2.3.1.3 Extended Boolean model

Extended Boolean model adds value to simpler model through the ability of weight assignment, and use of positional information. The term weights added to data objects help generate the ranked output. It is a combination of vector model characteristics and Boolean algebra. This model addresses the strict interpretation of information association in Boolean model and absence of structural characterization in vector based systems. The extended model introduced by [Salton et al., 1982], outperforms both conventional Boolean and vector space based IR models. This approach is used by many modern information retrieval systems.

2.3.2 Algebraic Models

Beside the logical reasoning approach IR models are also based on algebraic calculus. In general information is represented as vectors/matrices and similarity between information has a scalar value.

2.3.2.1 Vector space model

Information is represented as vectors in multidimensional space. Each dimension corresponds to a possible feature of the information (e.g. term in document). A distance function applied to the information vectors provide the match and rank information. It allows improved vector term weighting mechanism.



VSM with Euclidean distance measure

VSM with angular distance (cosine) measure

Figure 2. Vector Space Model

VSM based information retrieval is considered a good mathematical implementation for processing large information sources. It provides possibilities of partial matching and ranked result output. However this approach lacks the control of Boolean model, and has no means to handle semantic or syntactical information.

2.3.2.2 Latent Semantic Analysis based model

In conventional VSM the vector space is formed using literal terms and their frequencies. The exact term based matching does not retrieve information that share same concepts. LSA [Dumais et al., 1988] technique converts the large information matrix (term-document) to a lower dimensional space using singular value decomposition (SVD) technique. The reduced space is believed to be associated with concepts. Such an approach can effectively handle the term relations (synonymy and polysemy). LSA home page at CU Boulder [LSA, 2006] provides a good start to further explore the technique.

2.3.2.3 Neural Networks

In order to enhance information retrieval process, various machine learning approaches are being used to automatically acquire knowledge from given data. Most commonly used approach is the use of Neural Networks [Wilkinson and Hingston, 1991]. This model uses the weighted and interconnected representation of information. Spreading activation (self processing interlink functions) method used in neural networks allows effective learning and adds the ability of intelligent matches.

2.3.3 Probabilistic Models

These IR models are based on probabilistic inferences. Information is retrieved based on the probability of relevance between data. This technique addresses the uncertainty factors present in information need and information at hand. Use of

Bayes theorem that relates the conditional and marginal probabilities of events is a general example [Wikipedia:Bayes, 2008]. Models based on this approach include:

2.3.3.1 Inference network

Probabilistic inference networks approach [Turtle and Croft, 1990] use multiple sources of evidence to compute the conditional probability of match. The inference network consists of a document network, and a query network. The approach intends to capture significant dependencies among query and document network. Previously known document probabilities and conditional probabilities associated with interior nodes are used to calculate the posterier probability (belief), associated with each node of network. Document and query networks are joined by links between calculated representation and query concepts.

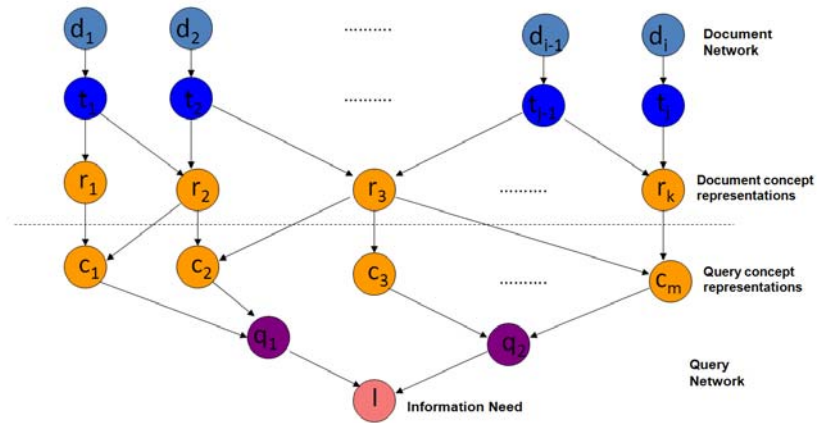


Figure 3. Document Inference Network [Turtle and Croft, 1990]

2.3.3.2 Belief network

Belief network IR model is a generalized form of inference network model, having a clearly defined sample space. The probabilistic considerations over the defined sample space simplify the understanding of model [Ribeiro and Muntz, 1996].

2.3.4 Knowledge based Models

The knowledge based IR models use formalized linguistic information, structural and domain knowledge to discover semantic relevance between objects. The techniques to introduce cognition in retrieval process are extensively used by information science researchers. However their effectiveness suffer because of extra efforts required to acquire and maintain a rich domain knowledge base. Various techniques are being used to represent and construct useful knowledge repositories for information retrieval [Martin and Eklunk, 2000]. Further information about generalized knowledge retrieval model and related theories and technologies is available in work by Yao et al. [Yao et al., 2007].

2.3.5 Structure based Models

The traditional information retrieval techniques in general consider detailed aspects of data contents, while structuring of data is given lesser importance. The structural information retrieval models combine the content and structural characteristics to achieve greater retrieval efficiencies in many applications. Some of these approaches include hybrid model, PAT expression, overlapped lists, List of references, proximal nodes and tree matching [Baeza-Yates and Navarro, 1996].

2.4 Content Processing

Storage and representation of data contents plays an important role in efficient similarity checks and retrieval operations. This type of storage and data representation is generally termed as "index" which is an auxiliary data structure to enhance data access. Two major categories for indexing can be seen as lexicographical indices (where most efficient mean of data storage is inverted files besides PAT trees) and the other is hash based indexing which use signature files for data storage. According to work by Zobel et al. [Zobel et al., 1998] [Zobel and Moffat, 2006] the inverted file based indexing is the most efficient form of data representation for textual contents and hash based indexing is better suited for multimedia contents. The ability of efficiently storing high dimensional feature space (terms in case of text) makes inverted file based storage not only useful in text based systems but also found very useful in CBR (Content Based Retrieval).

The contents go through a number of processes starting from parsing and tokenization. This step involves the extraction of meta data and raw text as character streams from various file formats like Pdf, Html, MSWord, Xml, Ppt etc. Tokenization process identifies meaningful segments as discrete token in parsed data stream. In case of text data steps involved in process are punctuations removal, bad character removal, uniform case folding, application of hyphenation rules and grouped word rules. High frequency words (stop words such as a, the, to, in, for, if, he, she, it...) are also removed in general purpose indexes. Various language processing techniques are also used to get an abstraction of contents. This include morphological and syntactical stemming, stripping of prefixes and suffixes, normalization of singular/plural and past/present. e.g.

computer->*computational*->*computation* generalized to *compute*

Generated token are given local and/or global term weight before indexing. The local token weight is referred to as term frequency *tf* which basically is number of term occurrences in document. This is normalized to prevent bias added by longer documents. The inverse document frequency *idf* is computed as global weight and represent the importance of term in a set of documents. *idf* is calculated by dividing number of all documents in selection by the number of documents that has the term, and then taking the logarithm of that quotient. [Wikipedia:Tf-idf, 2008]

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (xi)$$

$$idf_i = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|} \quad (xii)$$

Advance indexing further include steps to add position and payload information for phrase, proximity searches.

For multimedia content indices feature selection and extraction functions play important role in order to capture comprehensive details of data. In CBR systems, multimedia content features such as color histograms, color layouts, edge histogram, texture layouts constitutes the feature space. Many indexing platforms are adopting MPEG-7 standard [Wikipedia:MPEG-7, 2008] of content descriptors. MPEG-7 offers a comprehensive set of descriptors to define structural, model and content characteristics of audiovisual data. These features are generated by hashing multimedia objects by using various functions e.g. Fourier Transform, Hough Transform, Wavelet Transform, Gabor Transform, edge detection canny operators, and Hidden Markov model presentation of continuous audio video streams. for further understanding of these techniques please refer to [Djeraba, 2002] [Sonka et al., 2007] [Boreczky and Wilcox, 1998].

3. Plagiarism and IPR

Applications of similarity detection techniques in plagiarism detection and IPR

Contents of this chapter are taken from following publications:

"Plagiarism a survey"
[Maurer et al, 2006]

"Plagiarism – a problem and how to fight it" [Maurer and Zaka, 2007]

"Empowering plagiarism detection with a web services enabled collaborative network"
[Zaka, 2009b]

"Framework for Extending Plagiarism Detection in Virtual Worlds" [Zaka et al., 2009b]

This chapter starts with a survey which covers the social, legal and technical aspect of plagiarism. The survey provides a comparative analysis of various tools and techniques used to fight the problem, and insights to the problems and issues with existing approach. Later part of the chapter describes experiments conducted to overcome these problems and results of introduced enhancements. The work in this chapter introduces a platform which is host to a number of information discovery and similarity checking services. The service components of this platform (CPDNet) are later used in other application areas as well.

3.1 Introduction

Plagiarism in the sense of "theft of intellectual property" has been around for as long as humans have produced work of art and research. However, easy access to the Web, large databases, and telecommunication in general, has turned plagiarism into a serious problem for artists, students, publishers, researchers and educational institutions. This chapter describes the complex general setting, then report on some results of plagiarism detection software and draw attention to the improvements required in IPR detection and prevention systems. Practical steps taken towards fulfillment of these requirements are also part of the presented work.

3.1.1 Defining Plagiarism

There are many definitions of what constitutes plagiarism, and we will look at some of them in more detail below. However, according to research resources at plagiarism.org, the things that immediately come to mind as description of plagiarism are:

- turning in someone else's work as your own
- copying words or ideas from someone else without giving credit
- failing to put a quotation in quotation marks

- giving incorrect information about the source of a quotation
- changing words but copying the sentence structure of a source without giving credit
- copying so many words or ideas from a source that it makes up the majority of your work, whether you give credit or not [Plagiarism.org, 2006]

The border-line between plagiarism and research is surprisingly murky. After all, advanced research is only possible by “standing on the shoulders” of others, as it is often said. In some areas (such as e.g. literature or law) a scholarly paper may well consist of a conjecture followed by hundreds of quotes from other sources to verify or falsify the thesis. In such case, any attempt to classify something as plagiarized vs. not-plagiarized just based on a count of lines of words that are taken literally from other sources is bound to fail. In other areas (like in a paper in mathematics) it may be necessary to quote standard literature just to make sure that readers have enough background to understand the important part, the proof of a new result whose length may well be below one third of the paper. In other disciplines like engineering or computer science the real value of a contribution may be in the device or algorithm developed (that may not even be explicitly included in the paper) rather than the description of why the device or algorithm is important that may well be spelled out in a number of text books. In summary, we believe that there is no valid definition of even textual plagiarism that is not somewhat domain dependent, complicating the issue tremendously.

A good survey of further ideas about how to define plagiarism, and famous examples of suspected or perpetrated plagiarisms can be found in the Wikipedia [Wikipedia:Plagiarism, 2006]. Let us now turn, however, to an attempt to classify various types of plagiarism:

Plagiarism is derived from the Latin word “plagiarius” which means kidnapper. It is defined as “the passing off of another person's work as if it were one's own, by claiming credit for something that was actually done by someone else” [Wikipedia:Plagiarism, 2006]. Plagiarism is not always intentional or stealing some things from someone else; it can be unintentional or accidental and may comprise of self stealing. The broader categories of plagiarism include:

- Accidental: due to lack of plagiarism knowledge, and understanding of citation or referencing style being practiced at an institute
- Unintentional: the vastness of available information influences thoughts and the same ideas may come out via spoken or written expressions as one's own
- Intentional: a deliberate act of copying complete or part of someone else's work without giving proper credit to original creator

- Self plagiarism: using self published work in some other form without referring to original one [Wikipedia:Plagiarism, 2006] [Beasley, 2006].

There is a long list of plagiarism methods commonly in practice. Some of these methodologies include

- copy-paste: copying word to word textual contents.
- idea plagiarism: using similar concept or opinion which is not common knowledge.
- Paraphrasing: changing grammar, similar meaning words, re-ordering sentences in original work. Or restating same contents in different words.
- artistic plagiarism: presenting someone else's work using different media, such as text, images, voice or video.
- code plagiarism: using program code, algorithms, classes, or functions without permission or reference.
- forgotten or expired links to resources: addition of quotations or reference marks but failing to provide information or up-to-date links to sources.
- no proper use of quotation marks: failing to identify exact parts of borrowed contents.
- misinformation of references: adding references to incorrect or non existing original sources.
- translated plagiarism: cross language content translation and use without reference to original work.

3.1.2 Impact

A survey (released in June, 2005) conducted as part of Center of Academic Integrity's Assessment project reveals that 40% of students admitted to engaging in plagiarism as compared to 10% reported in 1999 [CAI, 2005]. Another mass survey conducted by a Rutgers University professor in 2003 reports 38% of students involved in online plagiarism [Rutgers, 2003]. These alarming figures show a gradual increase. The new generation is more aware of technology than ever before. Plagiarism now is not confined to mere cut and paste; synonymising and translation technologies are giving a new dimension to plagiarism.

Plagiarism is considered to be a most serious scholastic misconduct; academia everywhere is undertaking efforts to educate the students and teachers, by offering guides and tutorials to explain types of plagiarism and how to avoid it.

This growing awareness is forcing universities and institutes all around to help students and faculty understand the meaning of academic integrity, plagiarism and its consequences. Since plagiarism is often connected with the failure to reference

or quote properly, many institutions suggest following one of the recognized writing styles as proposed by major publishing companies like Springer, or by using well defined citation styles like: Modern Language Association (MLA) style¹, Chicago Manual of style², or American Psychological Association (APA) style³.

3.2 Response of academic institutions

Although plagiarism is reasonably well defined and explained in many forums, the penalty for cases detected varies from case to case and institution to institution. Many universities in the United States have well defined policies to classify and deal with academic misconduct. Rules and information regarding it are made available to students during the enrolment process, via information brochures and the university web sites. Academic dishonesty can be dealt with at teacher-student level or institute-student level. The penalties that can be imposed by teachers include written or verbal warning, failing or lower grades and extra assignments. The institutional case handling involves hearing and investigation by an appropriate committee, with the accused aware and part of whole process. The institutional level punishments may include official censure, academic integrity training exercises, social work, transcript notation, suspension, expulsion, revocation of degree or certificate and possibly even referral of the case to legal authorities. To be specific, we have collected a number of examples:

Stanford University: Stanford University provides its students with a well defined academic misconduct policy (Honor Code, in force since 1921) and a good collection of copyright and fair use resources [Stanford Copyright, 2006]. According to an article in the Stanford daily, the Stanford's office of judicial affairs saw 126 percent increase in honor code violation from 1998 to 2001. This precipitated the increasing usage of anti plagiarism software among instructors at individual levels [Stanford Daily, 2003]. As per the Stanford Honor Code "The standard penalty for a first offence includes a one-quarter suspension from the University and 40 hours of community service. In addition, most faculty members issue a "No Pass" or "No Credit" for the course in which the violation occurred. The standard penalty for multiple violations (e.g. cheating more than once in the same course) is a three-quarter suspension and 40 or more hours of community service" [Stanford Honorcode, 1921]. Some sample cases and sanctions are available at, University's Judicial Affairs website⁴.

Yale University: Yale College Executive Committee Yearly Chair Reports [Yale, 2005] indicate that the committee had to deal with a sizeable number of plagiarism cases every year. They show great concern about increase in web plagiarism. There are discussions about its causes and possible preventive measures mentioned in the

¹ <http://www.mla.org/style>

² <http://www.chicagomanualofstyle.org/>

³ <http://www.apastyle.org/>

⁴ <http://www.stanford.edu/dept/vpsa/judicialaffairs/students/pdf/plagiarism.cases.pdf>

reports. Punishments vary from case to case starting from reprimands, probations and extending to suspension. Despite clear academic misconduct policies there were cases of accidental or mistaken plagiarism, which suggests that there is a need of more effective ways of communicating details to students. Teachers are encouraged to explain plagiarism, citation rules and writing styles to students.

U.C. Berkeley: This University also has clear policies and preventive procedures against academic dishonesty. Instructors are encouraged to resolve the matter personally and issue academic sanctions; in case an accused person does not agree with allegations or sanctions, the matter is handed over to student judicial affairs for further investigations and resolution. Sanctions at U.C. Berkeley for plagiarism are warning/censure, community service, letters of apology, counseling, additional coursework, disciplinary probation, suspension, dismissal, and restitution [Berkeley, 2006].

Massachusetts Institute of Technology: MIT has well defined policies and procedures for handling academic misconduct [MIT policies, 2006]. Teachers are encouraged to educate students about permissible academic conduct. MIT's online writing and communication center [MIT Writing 2006] provides a platform to improve writing abilities and explains various aspects of plagiarism. According to a report available at MIT News Office portal, usually the discipline committee has to handle 12 to 15 cases annually with a tendency of increase in number of cases in recent years [MIT News 2003]. The penalties follow a similar trend as in other universities, starting from reduced grades, warning letters, redo of exam or assignment and in extreme cases with recommendation of the discipline committee, suspension or expulsion.

In Europe, UK is probably ahead of the other countries by taking collective measures against plagiarism. Most of the universities have online guides and tutorials available for students and researchers, helping them to understand academic integrity and improving writing skills. The higher education community in UK took a collective measure by forming a plagiarism advisory service [JISC, 2006] giving all UK institutes access to an online plagiarism detection service.

University of Cambridge: At Cambridge, suspected plagiarism cases involve separate academic and disciplinary elements. Examiners are asked to evaluate and make recommendations about suspected work but they cannot impose any penalty. The proctors, university advocates and courts decide about the sanctions in light of recommendations by examiners and investigations [Cambridge, 2006].

Oxford University: According to the University Gazette March 2005, six plagiarism related cases were dealt with during the previous term. "Three cases were dealt with by the Court of Summary Jurisdiction; in each, the examiners were instructed to disregard the plagiarized work and the candidates were permitted to resubmit (with a marks penalty in one case). The Disciplinary Court dealt with 2 plagiarism cases; in one case the examiners were instructed to disregard the plagiarized work. The

candidate was failed in a previously completed M.St. examination but permitted to retake the examination, and if the examiners are satisfied, permitted to re-enter the degree for M.Phil. In the second case, a candidate had previously been convicted of plagiarism by the Court of Summary Jurisdiction. He/she was permitted to submit new work and some of this was subsequently found to contain plagiarized material. A charge of attempting to cheat or act dishonestly was dismissed, but the candidate was nevertheless failed in the BCL examination. Following a proctorial investigation, and taking into consideration certain mitigating factors, the Examiners were instructed to disregard a candidate's original M.Phil submission. He/she was given permission to submit replacement work to be determined by the Examiners" [Oxford Gazette, 2005].

Elsewhere in Europe, there is also a growing concern and individual efforts have been started by teachers at departmental levels to educate researchers and students about plagiarism. At Graz University of Technology, Austria, a Commission for Scientific Integrity and Ethics defines guiding principles to deal with cases of plagiarism. A catalogue of possible academic, civil and criminal consequences will be ready by end of 2006. Instructors at various institutes of the university started adding information and warnings about plagiarism some time ago, e.g. figure 4, 5 & 6 show responses to plagiarism cases on course websites at various institutes of Technical University Graz.

Plagiate (2003/10/14)

Damit niemand unfair behandelt wird, wird jedes von mir gefundene Plagiat mit 0 Punkten bewertet, da sich daraus keine eigenständige Leistung ablesen lässt.

Figure 4. Taken from course information page, CGV, TU Graz

Unter Plagiarismus versteht man im wesentlichen das unauthorisierte und undokumentierte Verwenden von fremden Materialien (Text, Code, etc.):

Plagiarism is the improper use of another person's writing or ideas. It can be as subtle as the inadvertent omission of quotes or proper references when citing a source or as blatant as knowingly copying an entire paper verbatim and claiming it as original work. (Definition laut Turnitin.com)

Was alles unter den Term "Plagiarismus" fällt können sie hier nachlesen:

http://www.turnitin.com/research_site/e_what_is_plagiarism.html

Plagiarismus wird in den Arbeiten (seien es Texte oder Programme) die sie für den praktischen Teil abliefern streng geahndet. Wenn wir feststellen das sie Textteile (Programmteile) einfach kopiert haben ohne dies entsprechend zu kennzeichnen und die Urheber zu referenzieren bekommen sie 0 Punkte auf den praktischen Teil und können somit die EIS VU nicht mehr positiv beenden.

Um ihre Texte auf Plagiarismus zu überprüfen bedienen wir uns nicht nur einfacher Suchmaschinen, wir verwenden auch kommerzielle Produkte wie zum Beispiel Turnitin.

Figure 5. Taken from teaching information page, IAIK TU Graz

- a seminar paper (related to a topic which is close to your practical work), the *Seminararbeit*, about 6 pages long
- the practical work (source code, demo, benchmark numbers, etc.)
- a paper which documents your practical work, the *Projektarbeit*, about 14 pages long

Plagiarism (copying text from other people without proper references) is **NOT** tolerated. We check your papers with a tool!

Figure 6. Taken from seminar project contents by Elisabeth Oswald, IAIK TU Graz

The problem of academic misconduct and plagiarism also exists in universities of developing countries. The situation there has different dimensions where language problems and lack of guidance create further complications. The concept of plagiarism is generally less known and very little institutional efforts are made to educate students and staff about the plagiarism. However, this is changing rapidly, because of high profile incidents causing an alarming situation and introduction of strict measures to address the problem. The Higher Education Commission of Pakistan issued detailed guidelines and zero tolerance policy against plagiarism to all universities of the country [HEC Press, 2006]. This was initiated due to the discovery of high profile plagiarism cases at Pakistani universities which lead to the resignation of involved faculty members and expulsion of students.

At some places the fight against plagiarism is more about grooming the writers with organized guidelines, tutorials and honor codes; in other cases it is more about detection and punishment. However, a well balanced combination of both is the most effective approach.

3.3 Why plagiarism detection is important

In academia plagiarism detection is most often used to find students that are cheating. It is curious to note that as better and better plagiarism detection software is used, and the use is known to students, students may stop plagiarizing since they know they will be found out; or else, they will try to modify their work to an extent that the plagiarism detection software their university or school is using fails to classify their product as plagiarized. Note that there are already anti-anti plagiarism detection tools available that help students who want to cheat: students can submit a paper and get a changed version in return (typically many words replaced by synonyms), the changed version fooling most plagiarism detection tools.

However, plagiarism is not restricted to students. Staff may publish papers partially plagiarized in their attempt to become famous or at least beat the “publish or perish” rule. It is interesting to note that sometimes persons accused of plagiarism by actually showing to them that they have copied large pieces of text more or less

verbatim sometimes refuse to admit cheating. A tool called Cloze helps in such cases: it erases every fifth word in the document at issue, and the person under suspicion has to fill in the missing words. It has been proven through hundreds of experiments that a person that has written the document will fill in words more than 80% correctly, while persons who have not written the text will not manage more than some 50% correct fill-ins at most.

No plagiarism detection tool actually proves that a document has been copied from some other source(s), but is only giving a hint that some paper contains textual segments also available in other papers. A paper already published in a reputable journal is submitted to a plagiarism detection tool. This tool reported 71% plagiarism. Even after discarding the published journal URI. The explanation was that parts of the paper had been copied by two universities on their servers! This shows two things: first, the tools for plagiarism detection can be used also to find out whether persons have copied illegally from ones own documents and second, it can help to reveal copyright violations as it did in this case: the journal had given no permission to copy the paper.

This raises indeed an important issue: plagiarism detection tools may be used for a somewhat different purpose than intended like the discovery of copyright violation. In examining studies conducted for a government organisation for a considerable amount of money each we found that two of the studies were verbatim copies (with just title, authors and abstract changed) of studies that had been conducted elsewhere. When we reported this to the organisation involved the organisation was NOT worried about the plagiarism aspect (“we got what we wanted, we do not care how this was compiled”) but was concerned when we pointed out that they might be sued for copyright violation.

It is for similar reasons why some journals or conferences are now running a check on papers submitted in a routine fashion: it is not so much that they are worried about plagiarism as such, but (i) about too much self-plagiarism (who wants to publish a paper in a good journal that has appeared with minor modifications already elsewhere?) and (ii) about copyright violation. Observe in passing that copyright statements that are usually required for submissions of papers to prestigious journals ask that the submitter is entitled to submit the paper (has copyright clearance), but they usually do not ask that the paper was actually authored by the person submitting it. This subtle difference means that someone who wants to publish a good paper may actually turn to a paper mill and order one including transfer of copyrights.

Checking for plagiarism becomes particularly complex when the product published is part of some serious teamwork. It is common in some areas (like in medicine) that the list of authors of papers is endlessly long, since all persons that have marginally contributed are quoted. This is handled in different ways depending on the discipline: in computer science it is quite common that when a team of three or more work on a project, one of the researcher, or a subgroup makes use of ideas and

formulations developed by the team without more than a general acknowledgement. This is done since it is often impossible to ascertain which member of the team really came up with a specific idea or formulation first.

Overall, when plagiarism detection software reports that 15% or more of some paper has been found in one or a number of sources it is necessary to manually check whether this kind of usage of material from other sources does indeed constitute plagiarism (or copyright violation) or not. No summary report of whatever tool employed can be used as proof of plagiarism without careful case by case check.

Keeping this in mind we now turn to how plagiarism detection works. In the light of what we have explained “plagiarism warning tools” might be a more exact term for what is now always called “plagiarism detection tools”.

3.4 Detecting plagiarism

Plagiarism detection methods can be broadly categorized into three main categories; the most common approach is by comparing the document against a body of documents, basically on a word by word basis where documents may reside locally or not. The other two approaches are not exploited as much, yet can also be surprisingly successful. One is by taking a characteristic paragraph and just doing a search with a good search engine like Google. And the other is by trying to do style analysis; in this case either just within the document at issue or performing writing style comparison with documents previously written by the same author. This is usually called stylometry.

Let us look at the three approaches in more detail:

3.4.1 Document source comparison

This approach can be further divided into two categories; one that operates locally on the client computer and does analysis on local databases of documents or performs internet searches, the other is server based technology where the user uploads the document and the detection processes take place remotely. The most commonly used techniques in current document source comparison involve word stemming or fingerprinting. This is an approach introduced by Manber [Manber, 1994] where moderately sized strings (Fingerprints) from a document are compared for similarities with preprocessed indexes from other documents. The result gives a similarity approximation among documents being checked. Figure 7 shows a generic structure of document source comparison based plagiarism detection system.

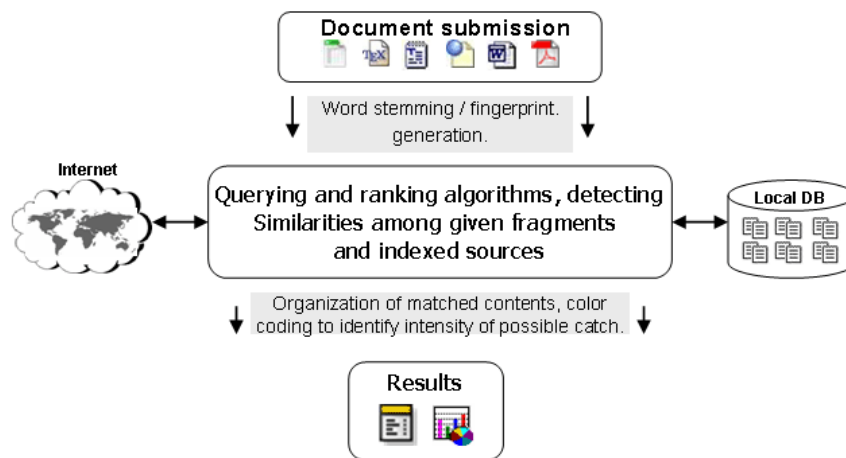


Figure 7. Plagiarism detection with document source comparison

The core finger printing idea has been modified and enhanced by various researchers to improve similarity detection. Many current commercial plagiarism detection service providers claim to have proprietary fingerprinting and comparison mechanisms. The comparison can be local or it can be across the internet. Some services utilize the potentials of available search engines. Many such tools use Google Search API⁵ providing querying capabilities to billions of web resources. Recent steps taken by Google to index the full text of some of the world's leading research libraries [Band, 2006], and its well known searching and ranking algorithm makes it an ideal choice not only for open source and free tools but is also used by many commercial service providers and applications. The more popular commercial and server based approaches claim to use their own search and querying techniques over more extensively indexed internet documents, proprietary databases, password protected document archives and paper mills. (more on paper mills in the next paragraph). The detection services or tools usually represent the similarity findings in a report format, by identifying matches and their sources. The findings are then utilized by users of the service to determine whether the writing under question is actually plagiarized or whether there are other reasons for match detection.

Returning to the issue of paper mills, this term refers to “website where students can download essays, either free or for a service charge. Online paper mills usually contain a large, searchable database of essays. Most paper mills today offer customized writing services, usually charging by the page. Some sites now even offer ready-made college application essays from applicants who have been accepted” [Wikipedia:papermill, 2006].

⁵ <http://www.google.com/apis/>

There are a number of web sites that even list paper mills.⁶

3.4.2 Manual search of characteristic phrases

Using this approach the instructor or examiner selects some phrases or sentences representing core concepts of a paper. These phrases are then searched across the internet using single or multiple search engines. Let us explain this by means of an example.

Suppose the following sentence in a student's essay is found suspicious

"Let us call them eAssistants. They will be not much bigger than a credit card, with a fast processor, gigabytes of internal memory, a combination of mobile-phone, computer, camera"

Since eAssistant is an uncommon term, it makes sense to input the term into a Google query. Indeed if this done the query produces:

"(Maurer H., Oliver R.) The Future of PCs and Implications on Society -

Let us call them eAssistants. They will be not much bigger than a credit card, with a fast processor, gigabytes of internal memory, a combination of ...

www.jukm.org/jucs_9_4/the_future_of_pcs/Maurer_H_2.html - 34k -"

This proves that without further tools the student has used part of a paper published in the Journal J.UCS⁷. It is clear that this approach is labor intensive; hence it is obvious that some automation will make sense, as is done in SNITCH [Niezgoda and Way, 2006]

3.4.3 Stylometry

Stylometric analysis is based on individual and unique writing styles of various persons. The disputed writing can be evaluated using different factors within the same writing. Or it can be cross compared with previous writings by the same author. The detection of plagiarism within the document domain or without any external reference is well described as "intrinsic plagiarism detection" by Eissen and Stein [Eissen and Stein, 2006]. This approach requires well defined quantification of linguistic features which can be used to determine inconsistencies within a document. According to Eissen and Stein "Most stylometric features fall in one of the following five categories: (i) text statistics, which operate at the character level, (ii) syntactic features, which measure writing style at the sentence-level, (iii) part-of-speech features to quantify the use of word classes, (iv) closed-class word sets to count special words, and (v) structural features, which reflect text

⁶ <http://www.coastal.edu/library/presentations/mills2.html>

⁷ <http://www.jucs.org>

organization.” [Eissen and Stein, 2006] The paper quoted, adds a new quantification statistic “the averaged word frequency class” and presents experiments showing its effectiveness. As an example of simple generic intrinsic plagiarism analysis let us take the following paragraph.

*“**Our** goal is to identify files that came from **the same source** or contain parts that came from **the same source**. **We** say that two files are similar if they contain a significant number of common substrings that are not too small. **We** would like to find enough common substrings to rule out chance, without requiring too many so that we can detect similarity even if significant parts of the files are different. However, **my** interest in plagiarism lies within academic institutions, so the document domain will be local research articles. The limited scope of domain will make it easier to determine if it is **same source** or not.”*

A careful reading reveals the following inconsistencies:

- There is a change in pronoun from “our/we” to “my”
- The writer used the article “the” with “same source” in two sentences and missed the article in another.

The bold words show the inconsistency and thus exhibit the possibility of plagiarism, where the writer took text from some source not matching the overall writing style. This approach can be hard to use in case of collaboratively written text where multiple writers are contributing to a single source.

Cross comparisons include a check on change of vocabulary, common spelling mistakes, the use of punctuation and common structural features such as word counts, sentence length distributions etc. (see example of using structural features to detect similarity in “Advanced Techniques” section). In order to further explain stylometry and another approach, we look at a service by Glatt [Glatt, 2006], which uses Wilson Taylor's (1953) cloze procedure. In this approach every fifth word in a suspected document is removed and the writer is asked to fill the missing spaces. The number of correct responses and answering time is used to calculate plagiarism probability. For example the examiner suspects that the following paragraph is plagiarized.

“The proposed framework is a very effective approach to deal with information available to any individual. It provides precise and selected news and information with a very high degree of convenience due to its capabilities of natural interactions with users. The proposed user modeling and information domain ontology offers a very useful tool for browsing the information repository, keeping the private and public aspects of information retrieval separate. Work is underway to develop and integrate seed resource knowledge structures forming basis of news ontology and user models using.....”

The writer is asked to take a test and fill in periodic blank spaces in text to verify the claim of authorship. A sample test based on above paragraph is shown in figure 8.

Your job is to fill in the blanks with the EXACT word you think you used.

Use your cursor to move from one blank to the next blank; DO NOT USE THE TAB KEY.

Do not look at your original paper or the test results will be invalid. Each blank represents ONE word.

Type the word that you think belongs in each blank. Continue until the end of the text.
Remember, you can always go back and make any changes to your answers. When you are satisfied, push the submit button.

Remember, do NOT consult your paper or the test results will be INVALID.

The proposed framework is a very effective approach to deal with information available to any individual. It provides precise and selected news and information with a very high degree of convenience due to its capabilities of natural interactions with the system . The proposed user modelling and information domain ontology offers a very useful tool for

Text:

Figure 8. Stylometric test, Glatt Plagiarism Self-Detection Program

Score

Number of Words Correctly Identified: 7

Number of Words Incorrectly Identified: 4

Total Words Attempted: 11

Percent Correct: 0.64

SCORING FOR SELF-DETECTION TEST

The Glatt Plagiarism Self-Detection Test is based on the theory that each person has a unique style of writing. Furthermore, it is assumed that you know and can remember your own writing better than anyone else.

So how did you do?
Did you get at least 50% correct?

If not, you may want to rewrite the passage and take the Self-Detect Test again.

Figure 9. Stylometric test results

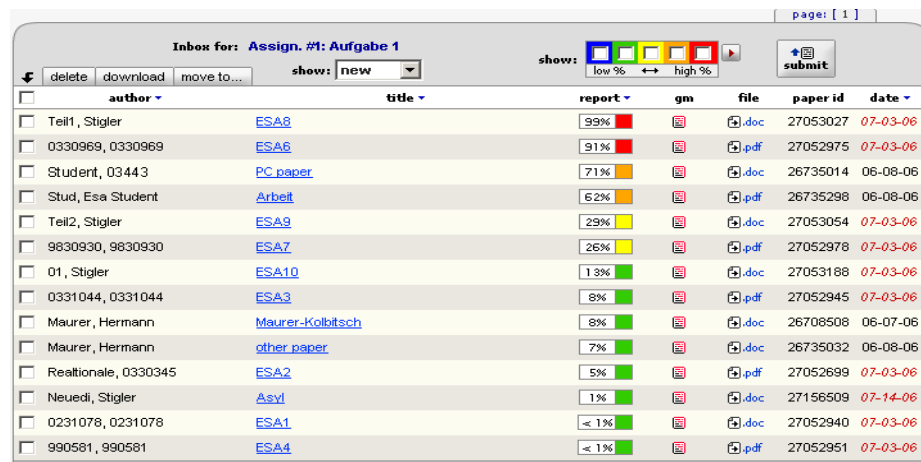
The percentage of correct answers can be used to determine if the writing is from the same person or not. The result of the mentioned test is shown in figure 9. This approach is not always feasible in academic environment where large numbers of documents are needed to be processed, but it provides a very effective secondary layer of detection to confirm and verify the results.

3.5 Available tools

Several applications and services exist to help academia detect intellectual dishonesty. We have selected some of these tools which are currently particularly popular and describe their main features in what follows.

3.5.1 Turnitin

This is a product from iParadigms [iParadigm, 2006]. It is a web based service. Detection and processing is done remotely. The user uploads the suspected document to the system database. The system creates a complete fingerprint of the document and stores it. Proprietary algorithms are used to query the three main sources: one is the current and extensively indexed archive of Internet with approximately 4.5 billion pages, books and journals in the ProQuest™ database; and 10 million documents already submitted to the Turnitin database.



author	title	report	gm	file	paper id	date
Teil1, Stigler	ESA8	99%		.doc	27053027	07-03-06
0330969, 0330969	ESA6	91%		.pdf	27052975	07-03-06
Student, 03443	PC paper	71%		.doc	26735014	06-08-06
Stud, Esa Student	Arbeit	62%		.pdf	26735298	06-08-06
Teil2, Stigler	ESA9	29%		.doc	27053054	07-03-06
9830930, 9830930	ESA7	26%		.pdf	27052978	07-03-06
01, Stigler	ESA10	13%		.doc	27053188	07-03-06
0331044, 0331044	ESA3	8%		.pdf	27052945	07-03-06
Maurer, Hermann	Maurer-Kolbitsch	8%		.doc	26708508	06-07-06
Maurer, Hermann	other paper	7%		.doc	26735032	06-08-06
Reallionale, 0330345	ESA2	5%		.pdf	27052699	07-03-06
Neuedi, Stigler	Asyl	1%		.doc	27156509	07-14-06
0231078, 0231078	ESA1	< 1%		.doc	27052940	07-03-06
990581, 990581	ESA4	< 1%		.pdf	27052951	07-03-06

Figure 10. Turnitin, Instructor view of assignment inbox

Turnitin offers different account types. They include consortium, institute, department and individual instructor. The former account type can create later mentioned accounts and have management capabilities. At instructor account level, teachers can create classes and generate class enrolment passwords. Such passwords are distributed among students when joining the class and for the submission of assignments. Figure 10 and 11 gives an idea of the system's user-interface.

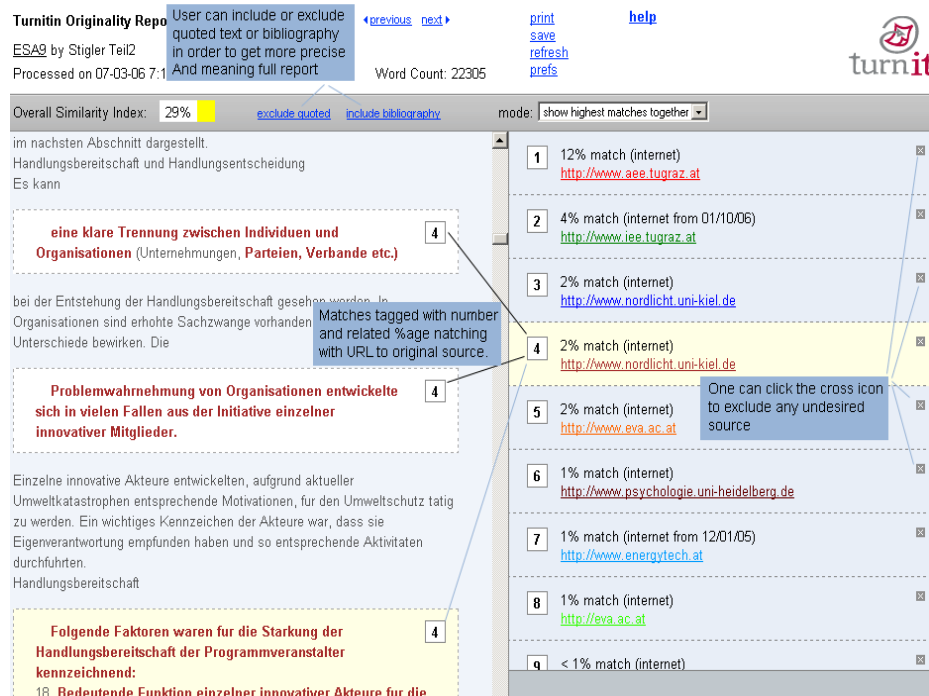


Figure 11. Turnitin, originality report of a submission

The system generates the originality report within some minutes of submission. The report contains all the matches detected and links to original sources with color codes describing the intensity of plagiarism [Turnitin tour, 2006]. It is however not a final statement of plagiarism. A higher percentage of similarities found do not necessarily mean that it actually is a case of plagiarism (for further explanation see Section 3: Unexpected Results). One has to interpret each identified match to deduce whether it is a false alarm or actually needs attention. This service is used by all UK institutes via the Joint Information Systems Committee (JISC) plagiarism Advisory Program [JISC, 2006].

3.5.2 SafeAssignment

This web based service by Mydropbox, claims to search an index of 8 billion internet documents, ProQuest™, FindArticles™ database by LookSmart™ and other major scholastic databases. The system also searches 300,000 documents that are known to be offered by Paper Mills. SafeAssignment also utilizes proprietary archives of institutional partners. Password protected and zipped archives can be indexed on demand. This product keeps fingerprints of the submitted papers in separate databases belonging to the account owner institute in order to avoid any legal or copy right problems. The service uses proprietary searching and ranking algorithms for match detection of fingerprints with its resources. The plagiarism detection result is presented to the user after a couple of minutes of submission, i.e. is similar in this respect with previously mentioned products [Mydropbox, 2006]. Figure 12 displays report of a processed paper.

SafeAssignment Report

Paper Information

Owner: Test Instructor	Folder: tests	Save report to disk: <input type="checkbox"/>
Filename: pc_in_10_jahren.doc	Submitted: 2006-08-24 06:26:41 EST	Print version:
Matching: <div><div></div></div> 78%	Paper ID: 1625508	

Suspected Sources

- ☐ http://www.iicm.edu/Ressourcen/Papers/pc_in_10_jahren.pdf
- ☐ http://www.iicm.edu/Ressourcen/Papers/learnen_ist_wissenstransfer.doc
- ☐ http://www.jucs.org/jucs_9_4/foundations_of_miracle_multimedia
- ☐ <http://www.acm.org/sigs/sigmod/dblp/db/indices/a-tree/>
- ☐ http://www.iicm.edu/iicm_papers
- ☐ http://www.jucs.org/jucs_articles_by_category/H.1

Colored text in report shows found matches and corresponding URL list of original sources. User can exclude any source and reprocess the report

Excluded Sources

- ☒ http://www.iicm.edu/iicm_papers/pc_in_10_jahren.doc

☒ Re-process the paper without these sources

Paper Text

Der PC in zehn Jahren und seine Auswirkungen auf die Gesellschaft

Der PC in zehn Jahren und seine Auswirkungen auf die Gesellschaft

In diesem Artikel wird argumentiert, dass PCs, wie wir sie heute kennen, in zehn Jahren nicht mehr existieren, sondern voll in Handys integriert sein werden. Als ständige Begleiter werden sie das Leben in einem unerhÄrten Ausmass Ändern. Es werden sowohl Technik als auch Auswirkungen kurz erlÄutert.

Einleitung

Laptops werden immer mÄchtiger, GPS kann genauso wie eine einfache Kamera mit denen man Fotos knipsen und

Dieser Trend ist bei weitem nicht a

mit jenen eines Fotoapparates und einer Videokamera, der als vollwertiger Computer verwendbar ist, der A1/4ber ein GPS System und weitere Sensoren verfÄgt (z.B. um die Position des Kopfes des Benutzers zu ermitteln, inklusive Blickrichtung u.a.). Ja dieses kleine UniversalgerÄt a nennen wir es UPC fÄ1/4r "Universal PC": a wird auch als elektronische Identifikation anstelle eines Personalausweises; zum Zahlen und als SchIA1/4ssel fÄ1/4r TA1/4ren oder Safes ausgelegt sein. Es wird A1/4berdies dauernd mit einem Netzwerk in Verbindung stehen: zukA1/4nftige breitbandige drahtlose Netze werden nur volumenorientierte VergebA1/4hrung aufweisen, wobei durch die immer bestehende Netzverbindung keine Kosten anfallen.

URL: http://www.iicm.edu/Ressourcen/Papers/pc_in_10_jahren.pdf Matching: 100 % x

Uploaded Manuscript: Es werden sowohl Technik als auch Auswirkungen kurz erlÄutert

Internet Source: Es werden sowohl Technik als auch Auswirkungen kurz erlÄutert

user can click colored text to get information box which display match percentage and text at both locations.

Figure 12. Mydropbox, paper information report

Mydropbox products integrates with other learning management systems (Blackboard®, WebCT) to extend plagiarism detection capabilities in existing systems running at institutes.

3.5.3 Docol©c

A web based service offered by Institut für Angewandte Lerntechnologien(IFALT)⁸. This service utilizes the searching and ranking capabilities of the Google API. The user of the service uploads the document that needs to be evaluated to a server. The software provides a simple console to set fingerprint (search fragments) size, date constraints, filtering and other report related options. The analysis report is sent to the browser or user's email identifying the matched fragments and internet sources. Figures 13 and 14 show different consoles and detection report by service.

⁸ <http://www.ifalt.com/>

Docol©c

Logged in. [Log out](#)

[Quick Guide](#) [Change Login](#) [Add Paper](#) [View Reports](#) [4]

Local file: [Browse...](#) [Use web-address](#)
[Preferences](#)

☐ demo ☒ professional [Start Plagiarism Search](#)

Send report: ☐ to browser ☒ to my account

☐ by email:

[Contact](#) - [Terms & Prices](#) - [Popollog-Evaluation](#) - [Google & References](#) - [Help](#)

©2006 IfALT - [IBR/ITM](#) research partner - Plagiarism search in more than 8 billion documents
[german](#) english

Docol©c

[Quick Guide](#) [Change Login](#)
[Add Paper](#) [View Reports](#) [4]

Logged in. [Log out](#)

Search options:

Same length of fragments [words](#)

Date constraint ☐ Yes ☒ No

Filter to simplify result ☐ Don't filter ☒ [Apply](#)

Text analysis:

Quality of sentences [words](#) [excellence](#)

Sentence length [sentences](#)

Paragraph length [sentence](#)

Output:

Found documents [found documents](#)

Snippet ☒ [show](#)

URL ☒ [show](#)

[Save preferences](#)

Preferences

Settings in the form left have impact on analysing the documents and the presentation of the review.

[Reset](#) to Docol©c default values.

[IfALT](#) - [Terms & Prices](#) - [Popollog-Evaluation](#) - [Google & References](#) - [Help](#)

©2006 IfALT - [IBR/ITM](#) research partner - Plagiarism search in more than 8 billion documents
[german](#) english

Figure 13. Docoloc, Start page and detection preference settings

Report

Digitally signed

Docolcc

Title: A report on Text to Speech systems

Reviewed document: P_Test_0003.doc

Processing date: Sun, 9.7.2006 11:48:55 CEST

A total of 43 fragments were analysed. As a result 9 fragments (20.9%) were found in other documents. In the document preview below the fragments are marked yellow, and clickable. At most 6 found documents are shown with same text passages.

Cross reference documents

Following list of found documents is grouped by document titles and ordered by found fragments. With a mouseclick on "x fragments" the relevant fragments in the document are colored orange and the window scrolls to the first location. Click on "x fragments" again resets the special marks.

4 fragments were found in a text with the title: "Speech synthesis - Wikipedia, the free encyclopedia", located on:
<http://en.wikipedia.org/wiki/Text-to-speech>
http://en.wikipedia.org/wiki/Voice_synthesis
http://en.wikipedia.org/wiki/Speech_synthesis

4 fragments were found in a text with the title: "Reference.com/Encyclopedia/Speech synthesis", located on:
http://www.reference.com/browse/wiki/Speech_synthesis

4 fragments were found in a text with the title: "Speech Synthesis", located on:
http://en.wikipedia.org/wiki/Speech_synthesis

4 fragments were found in a text with the title: "speech synthesis: Information From Answers.com", located on:
<http://www.answers.com/topic/speech-synthesis>

2 fragments were found in a text with the title: "Speech synthesis - Bvio", located on:
http://bv.o.ngic.re.kr/Bvio/index.php?title=Speech_synthesis
http://bv.o.ngic.re.kr/Bvio/index.php/Speech_synthesis

2 fragments were found in a text with the title: "Speech synthesis - Biocrawler", located on:
http://www.biocrawler.com/encyclopedia/Speech_synthesis

2 fragments were found in a text with the title: "Vergleich von deutschen Sprachsynthese-Systemen 2", located on:
<http://ttsamples.synthetispeech.de/deutsch/index.html>

The next part of TTS takes the symbolic linguistic representation and produces a synthesized voice file. The processes of synthesis generally used can be broadly segmented to two parts. One is called the concatenating synthesis approach and other the formant synthesis approach. In concatenating approach as the name describes several smaller recorded voice segments are joined together to produce the human sound. This results in a better quality sound but requires a well sized voice archives. Most popular techniques for concatenative synthesis are unit selection mechanism in which a large speech database is indexed and sound is produced using special weighted decision tree to assign vocal segment, the concatenated sequence then is processed via certain signal processing techniques to produce smoother sound. This method is believed to produce most natural and quality sounds but requires very large speech DB. Most of commercially available TTS uses this approach. Second approach to unit selection is Diphone synthesis in which only diphones (all available sounds in a language) of particular language is placed in speech archive. The prosody of spoken words are then implemented using DSP methods, this produces lower quality voice then unit selection but smaller size and many free TTS uses this approach. The third approach is domain specific synthesis where speech DB only has phrases words from a specific domain say a time telling system, or traveling schedule guide or weather system. Such systems are fast and efficient but only restricted to very little scope.

Created a graphical representation of this model for better visual understanding

Available systems:

There are many text to speech conversion applications available both open source and free and commercial platforms. These applications are in use at various service levels. Some of the available systems are listed below

Freely available:

1. Festival (<http://www.cstr.ed.ac.uk/projects/festival/>) is a complete diphone concatenation and unit selection TTS system supporting British and American English, Spanish and Welsh.
2. Flite (<http://www.speech.cs.cmu.edu/flite/>) (Festival-lite) is a smaller, faster alternative version of Festival designed for embedded systems and high volume servers.
- FreeTTS written entirely in Java, based on Flite.
3. Festvox (<http://festvox.org/>) another festival variant restricted to several languages.
4. MBROLA (<http://tcts.fpm.s.ac.be/synthesis/mbrola.html>) is a rule-based TTS system for many languages.
5. GnuSpeech (<http://www.gnu.org/software/gnuspcech/>) is an open source TTS system.
6. Epos (<http://epos.ure.cas.cz/>) is a rule-driven TTS system for Czech and Slovak.
7. HTS (<http://hts.ics.nitech.ac.jp/>) is a freely available HMM-based TTS system.

Hizketaren sintesia - Wikipedia (Cache)

Flite (Festival-lite) is a smaller, faster alternative version of Festival designed for embedded systems and high volume servers. ...

speech synthesis: Information From Answers.com (Cache)

Flite (Festival-lite) is a smaller, faster alternative version of Festival designed for embedded systems and high volume servers. ...

Reference.com/Encyclopedia/Speech synthesis (Cache)

Flite (Festival-lite) is a smaller, faster alternative version of Festival designed for embedded systems and high volume servers. ...

Speech synthesis - Biocrawler (Cache)

Flite (Festival-lite) is a smaller, faster alternative version of Festival designed for embedded systems and high volume servers. ...

Speech synthesis - Bvio (Cache)

Flite (Festival-lite) is a smaller, faster alternative version of Festival designed for embedded systems and high volume servers. ...

Figure 14. Docolcc, Sections of test report

This service is totally dependent on the Google API and might become unavailable or change at any point. Service availability is NOT guaranteed by the providers.

3.5.4 Urkund

Another server based plagiarism detection web service which offers an integrated and automated solution for plagiarism detection. It utilizes standard email systems for submission of documents and viewing results. This tool also claims to search through all available online sources giving priority to educational and Scandinavian

48

origin. This system claims to process 300 different types of document submissions [Urkund, 2006].

3.5.5 Copycatch

A client based tool used to compare locally available databases of documents. It offers ‘gold’ and ‘campus versions’ [CopyCatch, 2006], giving comparison capabilities for large number of local resources. It also offers a web version which extends the capabilities of plagiarism detection across the internet using the Goggle API. Users are required to sign up for personal Google API licenses.

3.5.6 WCopyfind

An open source tool for detecting words or phrases of defined length within a local repository of documents [Wcopyfind, 2006]. The product is being modified to extend searching capabilities across the internet using the Google API at ACT labs⁹. The resultant product SNITCH [Niezgoda and way, 2006] is expected to be an application version of Docol©c web service.

3.5.7 Eve2 (Essay Verification Engine)

This tool works at the client side and uses it own internet search mechanism to find out about plagiarized contents in a suspected document [EVE, 2006]. It presents the user with a report identifying matches found in the World Wide Web.

3.5.8 GPSP - Glatt Plagiarism Screening Program

This software works locally and uses an approach to plagiarism detection that differs from previously mentioned services. GPSP detection is based on writing styles and patterns. The author of a suspected submission has to go through a test of filling blank spaces in the writing. The number of correctly filled spaces and the time taken for completion of the test provides the hypothesis of plagiarism guilt or innocence [Glatt, 2006]. This has already been discussed in some detail in Stylometry section.

3.5.9 MOSS - a Measure of Software Similarity

MOSS Internet service [MOSS, 2006] “accepts batches of documents and returns a set of HTML pages showing where significant sections of a pair of documents are very similar” [Schleimer et al., 2003]. The service specializes in detecting plagiarism in C, C++, Java, Pascal, Ada, ML, Lisp, or Scheme programs.

3.5.10 JPlag

Another internet based service [JPlag, 2006] which is used to detect similarities among program source codes. Users upload the files to be compared and the system presents a report identifying matches. JPlag does programming language syntax and structure aware analysis to find results.

⁹ <http://actlab.csc.villanova.edu/>

When using server based applications to evaluate student's work it is advisable to inform students about the online submission of authenticity checks. Such services keep a fingerprint version of student work in their database which is in turn used for further checking processes. This may be considered a violation of student's intellectual property copyrights [IPR overview, 2006]. There are examples of students filing legal cases to prevent their work being submitted to such systems [CNN, 2004] and threatening to sue for negligence when the institution was unable to provide clear policy statements about their prohibitions and treatment of plagiarism [Wikipedia:Kent, 2006]. All this makes it very important for universities to have a well defined policy and guidance system when students enroll at a university that uses such services.

3.6 Unexpected Results

The broad scope of plagiarism makes one wonder about the potential of available services. Some of the test cases worth mentioning are listed in this section.

“Paraphrasing” means using someone else's ideas but rewriting it with different words. This is certainly also plagiarism. Plagiarists who want to avoid even the work of coming up with words of their own can use a thesaurus or some “synonymizer” to do the job for them. A proof of concept of such an obvious cheat is a limited dictionary tool the Anti-Anti Plagiarism System¹⁰. The library of words in such tools can be enhanced to fit individual requirements. A paraphrased portion of writing using this approach was tested with two of the more often used plagiarism detection services.

We chose the following paragraph:

*“According to many **observers**, the **coming decade** will be the **decade** of **speech** technologies. Computer systems, whether **stationary** or mobile, **wired** or **wireless**, will **increasingly** offer users the **opportunity** to **interact** with **information** and **people** through **speech**. This has been made **possible** by the **arrival** of **relatively robust**, **speaker-independent**, **spontaneous** (or **continuous**) **spoken dialogue** systems in the late 1990s as well as through the **constantly falling costs** of computer speed, bandwidth, storage, and component **miniaturisation**. The **presence** of a speech recogniser in most **appliances** **combined** with distributed speech processing technologies will **enable** users to speak their **native tongue** when **interacting** with computer systems for a **very large number of purposes**. ”*

[Bryan Duggan, Mark Deegan, "Considerations in the usage of text to speech (TTS) in the creation of natural sounding voice enabled web systems", ACM International Conference Proceeding Series; Vol. 49, 2003]

¹⁰ <http://sourceforge.net/projects/aaps>

Paraphrasing it, using a simple automatic word replacement tool we obtain;

“Agreeing to many onlookers, the approaching era will be the era of verbal technologies. Computer systems, whether desktop or mobile, with wires or without wires, will progressively offer users the chance to interface with data and persons via speech. This has been made viable by the appearance of comparatively flourishing, speaker-free, impulsive (or continual) verbal conversation systems in the late 1990s as well as through the persistently declining prices of computer speed, network communication capabilities, storage space, and component miniaturization. The existence of a speech recognizer in most devices united with distributed speech processing technologies will allow users to speak their local language when working with computer systems for a great number of reasons.”

Note in passing that such simple automatic paraphrasing results in fairly poor English. To really use such an anti-anti/plagiarism tool more sophisticated linguistic techniques are essential.

The originality reports from two service providers in figure 15 and 16 show failure of detection.

The screenshot shows a web-based originality report interface. At the top, there's a blue header with a document icon and 'Help' and 'Close' links. Below this is a 'Paper Information' section with a yellow background. It contains a table with fields: Owner (Test User), Folder (Default), Save report to disk (checkbox), Filename (P_Test_0003.doc), Submitted (2006-08-16 10:22:01 EST), and Print version (checkbox). A 'Matching' bar shows 24% completion. Below this is a 'Suspected Sources' section with a blue background, listing four URLs with checkboxes and icons. The last section is 'Paper Text' with a blue background, containing the text of the report. A green circular icon with a 'P' is next to the title 'Text To Speech systems:'. The text of the report is a paraphrased version of the text in Figure 14, enclosed in large curly braces. At the bottom, there's a small paragraph about Text-to-speech (TTS) technology.

Paper Information		
Owner: Test User	Folder: Default	Save report to disk: <input type="checkbox"/>
Filename: P_Test_0003.doc	Submitted: 2006-08-16 10:22:01 EST	Print version: <input type="checkbox"/>
Matching: <div><div></div></div> 24%	Paper ID: 1522350	

Suspected Sources

- ☐ <http://www.eloq.com/>
- ☐ http://omnipelagos.com/entry?n=speech_synthesis
- ☐ http://eu.wikipedia.org/wiki/Hizketaren_sintesia
- ☐ <http://ttsamples.syntheticspeech.de/>

☒ Re-process the paper without these sources

Paper Text

A report on Text to Speech systems

A report on Text to Speech systems

xyz

Abstract. This document is about the analysis of text to speech engine quality and performance. I will try to review various free and commercial TTS for different performance parameters. Main emphasis will be to sort out a TTS best capable of translating text contents into German as well as English language.

Text To Speech systems:

Agreeing to many onlookers, the approaching era will be the era of verbal technologies. Computer systems, whether desktop or mobile, with wires or without wires, will progressively offer users the chance to interface with data and persons via speech. This has been made viable by the appearance of comparatively flourishing, speaker-free, impulsive (or continual) verbal conversation systems in the late 1990s as well as through the persistently declining prices of computer speed, network communication capabilities, storage space, and component miniaturization. The existence of a speech recognizer in most devices united with distributed speech processing technologies will allow users to speak their local language when working with computer systems for a great number of reasons.

Text-to-speech (TTS) is a speech processing application that is used to create spoken sound from textual contents. This speech synthesis processes enable visually disabled or simply a lazy person to listen to contents of a web page a text message or may be a book.

Figure 15. Originality report by first service

Originality Report [previous](#) [next](#) [print](#) [save](#) [refresh](#) [help](#)

Ptest 03 by Test 03
 Processed on 07-06-06 11:07 AM CEST ID: 27079086 Word Count: 1054

Overall Similarity Index: 30% [exclude quoted](#) [exclude bibliography](#) mode: show highest matches together

Abstract. This document is about the analysis of text to speech engine quality and performance. I will try to review various free and commercial TTS for different performance parameters. Main emphasis will be to sort out a TTS best capable of translating text contents into German as well as English language.

About To Speech systems:
 Agreeing to many onlookers, the approaching era will be the era of verbal technologies. Computer systems, whether desktop or mobile, with wires or without wires, will progressively offer users the chance to interface with data and persons via speech. This has been made viable by the appearance of comparatively flourishing, speaker-free, impulsive (or continual) verbal conversation systems in the late 1990s as well as through the persistently declining prices of computer speed, network communication capabilities, storage space, and component miniaturization. The existence of a speech recognizer in most devices united with distributed speech processing technologies will allow users to speak their local language when working with computer systems for a great number of reasons.

Text-to-speech (TTS) is a speech processing application that is used to create spoken sound from textual contents. This speech synthesis processes enable visually disabled or simply a lazy person to listen to contents of a web page a text message or may be a book.

A typical architecture of text-to-speech synthesis application can be mainly distributed into two parts. First part takes the raw input in form of text, and after performing language processing which includes text normalization which

- 11% match (internet from 03/20/06) <http://www.objectsspace.com>
- 10% match (internet) <http://ttsamples.syntheticspeech.de>
- 3% match (internet) <http://en.wikipedia.org>
- 2% match (internet) <http://www.aculab.com>
- 1% match (archived internet from 06/21/03) <http://liceu.uab.es>
- 1% match (internet) <http://www.voicesearchonline.com>
- 1% match (internet) <http://www.dolphin.no>
- 1% match (internet from 03/20/06) <http://www.objectsspace.com>

Figure 16. Originality report by second service

The above example shows the weakness of word by word comparison or using fingerprints just involving the exact words occurring in a text. We will come back to this issue later in section 6 where we will discuss possible solutions for this problem.

At times, various systems show a very high percentage of matches; this does not necessarily mean that the document is plagiarized. Rather, it can be due to the fact that we are checking some paper that has already been put on some server, hence the match is made with exactly the same contribution by the same author. In such a case, one can use the facility to exclude the high percentage matching original source and regenerate the report showing other matches detected by the system. Figures (17 - 19) show such a case and two versions of originality report.

Inbox for: Assign. #1: Aufgabe 1

show: [new](#) [low %](#) [high %](#) [submit](#)

<input type="checkbox"/>	author	title	report	gm	file	paper id	date
<input type="checkbox"/>	0330969, 0330969	ESA6 A high percentage match case	91%			27052975	07-03-06
<input type="checkbox"/>	Student, 03443	PC paper	71%			26735014	06-08-06
<input type="checkbox"/>	Stud, Esa Student	Arbeit	62%			26735298	06-08-06
<input type="checkbox"/>	Teil2, Stigler	ESA9	29%			27053054	07-03-06

Figure 17. System showing 91% match for a particular paper

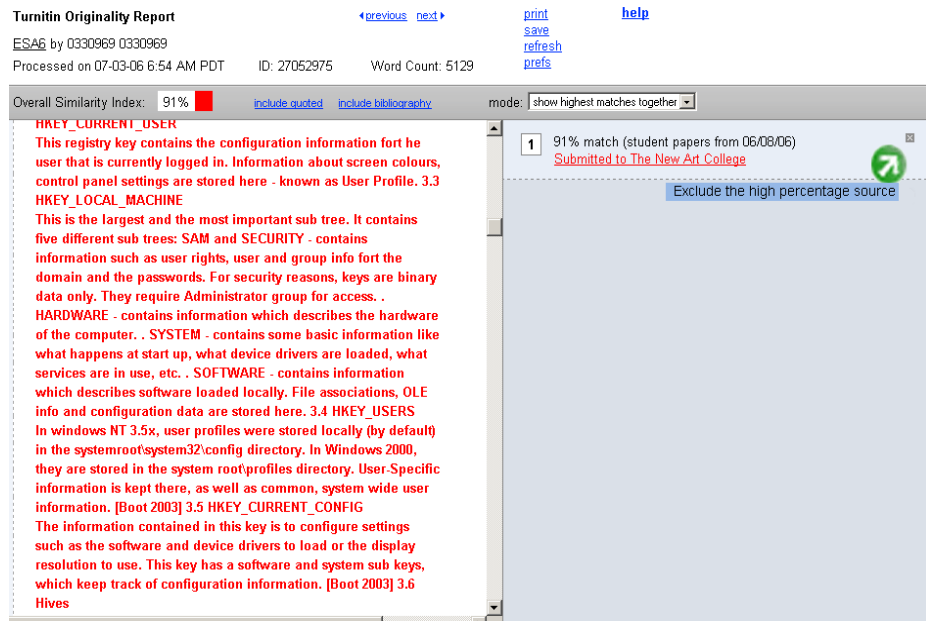


Figure 18. Report showing high percentage of match from a single source

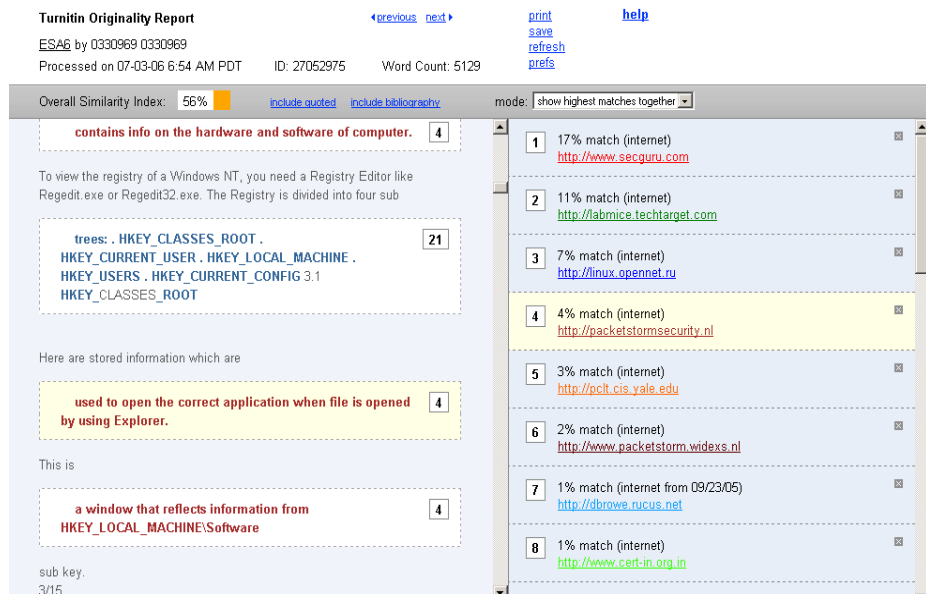


Figure 19. More meaning full report after excluding the high percentage source

Hence if a system finds a very high percentage match it can mean that the uploading was done in the wrong order.

Testing with tabular information and text in languages with special characters (German, Swedish, French etc.) showed that some of available systems are unable to correctly process data in table cells. Figure 20 shows few portions of test

documents submitted to different systems. The collected text in test comes from internet available documents and websites.

Some thing taken	Another part from some other location
Distance education is an eminently suitable mode of study for adult learners. If distance education can build on its existing strengths and respond to the concerns and support needs of adult learners, then there is a potential opportunity of overcoming inhibitions and anxieties which act as a barrier to large scale participation by adult learners.	<ul style="list-style-type: none">• Providing increased access and flexibility for study to students who work and/or have family obligations that prevent full-time or traditional enrollment.• Providing increased access to those who are geographically isolated from higher education.• Providing an opportunity to take classes that are transferable in order to fulfill a degree requirement.• Providing training that enhances employment options including
[Test Submission Number:09]	
Förståelse för mänsklig perception, kognition och beslutsfattande är centralt. Mycket av de metoder vi arbetar med bygger på kunskaper från beteendevetenskaper, särskilt psykologin. Kunskaper från datavetenskap är en annan viktig grundsten. Metoder utvecklas för analys, design och konstruktion av användargränssnitt. För att skapa förutsättningar för anpassning av datorstödd utvecklas metoder för användarcentrerad utveckling och för utvärdering av användbarhet. Kunskap om arbetsorganisation och arbetsmiljö är viktiga.	
Some thing in German	
Das System bietet außerdem wichtigen Vorteil, daß es sich Bereitschaft des Lehrers Zeit darin zu investieren auf die Akzeptanz neuer Lehr auf Seiten der Studenten einstellt. Es kann e nur unterstützend zu einer in traditioneller Weise gehaltenen Ausbildung verwendet werden (z.B. als definierter Parameter in Datensammlungen und Diskussionsforen, als elektronisches Skriptum), zur Nachbetreuung (Frage/Anwort	Die Regelanwendung kann auch iterativ mit schwächer werdenden Kriterien erfolgen. Hier werden jeweils nach Anwendung eines Kriteriums alle in der Ergebnismenge konfliktfreien Zuordnungen bestätigt, dann die in der Gesamtmenge der möglichen Zuordnungen mit diesen in Konflikt stehenden Zuordnungen verworfen, und auf die übrige Menge der möglichen Zuordnungen das gleiche Kriterium mit abgeschwächten Parametern erneut angewandt. Diese Iteration kann dann bis zu einem festgelegten
	polynômes et la nature des contraintes initiales. Ainsi, notre implantation de leur algorithme est valable pour un nombre de

Figure 20. Original tabular data with text containing special characters

Processing of testing documents through different detection services showed that in some cases the sentences are broken irregularly making a wrong fingerprint which might lead to false or no match detection. Some systems are also unable to properly process special characters; this might be the cause of no or lesser percentage of match detection in few test cases. Figures (21, 22) show few portions of resulting reports.

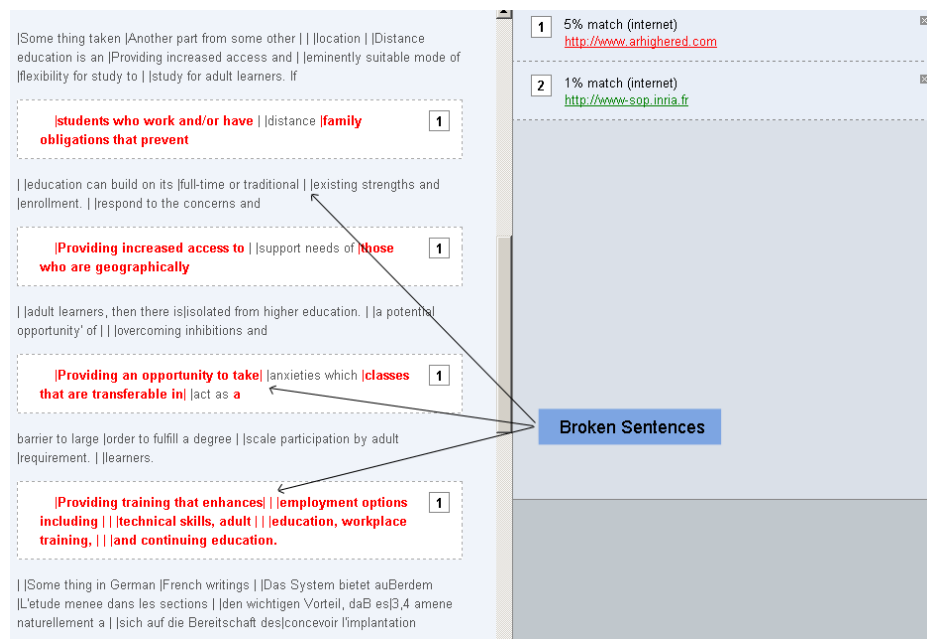


Figure 21. Report with broken table cell text

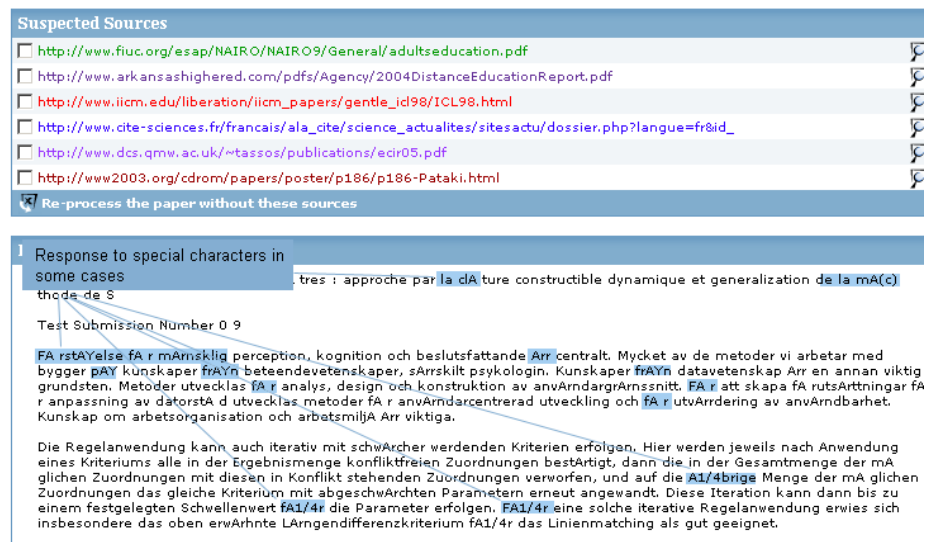


Figure 22. Document report with special characters

One interesting fact about the use of plagiarism detection services is that they can be also employed to discover illegal copies of our own writing as well. One such example is shown below: A paper produced by reputable writer showed a 71% match using one of the plagiarism detection services. A more detailed analysis of the report revealed the fact that various portions of the paper were used illegally at different places. Figures (23,24) show the relevant reports.

page: [1]

Inbox for: **Assign. #1: Aufgabe 1**

show: **new** | low % | high % | submit

<input type="checkbox"/>	author	title	report	gm	file	paper id	date
<input type="checkbox"/>	Maurer, Hermann	PC paper	71%		.doc	26735014	06-08-06
<input type="checkbox"/>	Stud, Esa Student	Arbeit	62%		.pdf	26735298	06-08-06
<input type="checkbox"/>	0330969, 0330969	ESA6	56%		.pdf	27052975	07-03-06
<input type="checkbox"/>	Teil2, Stigler	ESA9	29%		.doc	27053054	07-03-06
<input type="checkbox"/>	9830930, 9830930	ESA7	26%		.pdf	27052978	07-03-06

Figure 23. Use of plagiarism detection tools to discover copies of own writings

Turnitin Originality Report | [previous](#) | [next](#) | [print](#) | [help](#)
[save](#) | [refresh](#) | [prefs](#)

PC paper by Hermann Maurer
 Processed on 06-08-06 6:47 AM PDT | ID: 26735014 | Word Count: 4055

Overall Similarity Index: 71% | [exclude quoted](#) | [include bibliography](#) | mode: [show highest matches together](#)

Der PC in zehn Jahren
H.

Maurer, Graz University of Technology
hmaurer@iicm.edu

www.iicmedu/maurer

Kurzfassung

In diesem Artikel argumentiere ich, dass

PCs, wie wir sie heute kennen, in zehn Jahren nicht mehr existieren werden, sondern ihre Funktionen voll in weiterentwickelte Handys integriert sein werden. Als ständige Begleiter werden diese das Leben der Menschen in einem unerhörten Ausmaß verändern. Ich erläutere zunächst oberflächlich technische Aspekte (wobei einige kaum überraschen werden), gehe dann aber ausführlicher auf die zum Teil durchaus überraschenden Auswirkungen dieser dann jederzeit verfügbaren technologischen Wunder ein. Ich möchte vorweg ausdrücklich betonen, dass ich durchaus nicht alle möglichen und wahrscheinlichen Anwendungen positiv sehe, sondern dass auch große Gefahren damit verbunden sind.

Erstens, indem solche PC14 die Menschen sehr von sich abhängig machen können (was in [10] sehr deutlich beschrieben wird), zweitens, dass damit das Ausmaß der Überwachung noch weit über den Orwell'schen großen

1 47% match (internet)
<http://www.twi.uni-hannover.de>

2 21% match (internet)
<http://vl.rkmhessen.de>

3 < 1% match (internet from 10/10/05)
<http://www.jucs.org>

4 < 1% match (internet from 12/17/05)
<http://www.jucs.org>

5 < 1% match (internet from 12/01/05)
<http://www.jkraemer.net>

6 < 1% match (internet)
<http://www.education-quality.de>

7 < 1% match (internet)
<http://is.tn.tue.nl>

8 < 1% match (archived internet from 09/29/03)
<http://www.iicm.edu>

9 < 1% match (internet from 09/25/05)

Two major sources of found matches

Figure 24. Report with links showing copied portion of text

The highlighted/plagiarised portions in the report are linked to a specific URL pointing to the source. Visiting these sources confirms that the text was illegally copied from the author's paper that had appeared in a journal previously.

	Turnitin	Mydropbox	Docol©c
Technology	Web based, server side processing, support internet and other external scholastic databases	Web based, server side processing, support internet and other external scholastic databases	Web based, server side processing, support internet searches via Google API
Supported file types	MS Word, WordPerfect, PostScript, PDF, HTML, RTF, and plain text	ZIP, DOC, TXT, PDF, RTF, HTML and Direct text paste in text box at site	PDF, DOC (Word®), RTF, HTML, PPT, (Power Point®), XLS (Excel®), and TXT
Verbatim/Cut-Paste check	Yes	Yes	Yes
Paraphrase check	No	No	No
Tabular information processing	Showed problem in some cases	Yes	Yes
Translation check	No	No	No
Image/multi-media checks	No	No	No
Reference validity check	No	No	No
Exclusion/selection of sources	Yes	Yes	No

Table 1. Comparison of plagiarism detection capabilities

We tested three commonly used commercial services (Turnitin, Mydropbox and docoloc) with a selected set of submissions. The experiments showed generally similar results. Table 1 shows the feature matrix of these services.

3.7 Advanced techniques

Most services and tools described in earlier sections address verbatim plagiarism and utilize the document source comparison approach for detection. Thus, similarities that are not detectable by just comparison of word-based fingerprints usually escape those tools. However, more sophisticated similarity detection which is the core of source comparison is used to some extent already in many other areas such as data mining, indexing, knowledge management and automated essay grading.

Although we are not aware of concept-oriented or semantic similarity detection in existing plagiarism detection services we do find experimental research projects and

other commercial products which utilize innovative similarity detection methodologies, often for simpler tasks e.g. just checking whether a question asked is similar to one in the list of available FAQs.

A research paper in this direction describing so-called Active Documents explains that the most satisfying approach for checking whether a similarity exists in the meaning of different pieces of text is of course to determine their semantic equivalence. “To actually prove that two pieces of text are semantically equivalent one would require a complete understanding of natural language, something still quite elusive. However, we can consider a compromise: rather than allowing a full natural language we restrict our attention to a simplified grammar and to a particular domain for which an ontology (semantic network) is developed. Clearly, sufficiently restricting syntactic possibilities and terms to be used will allow one to actually prove the equivalence of pieces of text.” [Heinrich and Maurer, 2000]

Before we further look at various experiments that use semantic information and find aspects that may limit their use in similarity analysis we first describe one mathematical approach generally used in similarity detection.

A popular approach to similarity detection or pattern recognition is the use of a vector space model to determine cosine (i.e. angular) similarity among vectors of keywords/function-words extracted from the text under inspection.

To elaborate more let us take an example of two sentences

Text A: “*A rainy day with a cold wind*”

Text B: “*A sunny day with blue sky*”

Each text is represented in a word frequency table as follows:

Text A:	Text B:	Complete vocabulary:
a: 2	a: 1	a
rainy: 1	blue: 1	blue
day: 1	day: 1	cold
with: 1	sunny: 1	day
cold: 1	sky: 1	rainy
wind: 1	with: 1	sky
		sunny
		wind
		with

Table 2. Word-frequency in text, and complete vocabulary

The representation of the two pieces of text as vectors based against the vocabulary is: Text A= {2,0,1,1,1,0,0,1,1} and Text B= {1,1,0,1,0,1,1,0,1}.

Now let us take some text for similarity detection e.g. C: “A cold day”. The vector representation is $C = \{1,0,1,1,0,0,0,0\}$.

The cosine similarity measure between text A and C is calculated using formula

$$\frac{\text{Vector-A} \bullet \text{Vector-C}}{|\text{Vector-A}| |\text{Vector-C}|}$$

Calculations give us similarity measure of 0.769 between document A and C and 0.471 between B and C. Thus one can make assumption of similarity even if the two pieces of text are not completely identical. In real applications word vectors are made by the removal of stop words (frequently occurring words that can be ignored in a query, e.g. the, is, of, be, a etc.) and keyword vectors generally are made using tf-idf weights. These are very common methods and their functionality and limitations are well known. One can imagine that using a semantic matrix of words and concepts for a large corpus of text and complete language information, the vector space can be easily too large and noisy for practical computation. Thus, we need ideas and methodologies to improve this analysis. Examples are limiting the domain (i.e. to the ontology of subject in question) as described earlier in this section or other techniques which we will discuss a bit later.

The plagiarists today are becoming aware of limitations of existing systems and avoid detection by using linguistic tools as demonstrated in one example above. They can replace functional words after small intervals by using synonyms, retaining the idea or concept behind the sentences, yet remain undetected.

However, semantic or syntactic elements of any language can be used to enhance similarity detection mechanism and anti plagiarism software as well. One such approach to empower document similarity detection using semantic analysis is discussed by Iyer and Singh. Their system extracts keywords (nouns, verbs, adjectives in this case, ignoring adverbs, pronouns, prepositions, conjunctions and interjections) representing structural characteristics of documents. Synonym clusters for keywords are looked up from WordNet¹¹ and each cluster is represented with a numeric value. All keywords that are present in the structural characteristic tree of the document also carry the numeric value of the synonym cluster they belong too. Thus, when comparing sources, the binary comparison of synonym cluster numbers tells whether two words are synonyms. The software runs the comparison algorithms initially on the structural characteristic tree of the complete document. If similarities are above a certain threshold, only then is sentence level comparison initiated. This makes the system capable of detecting similarity even with minor semantic modifications at sentence level [Iyer and Singh, 2005].

Another approach of “Using Syntactic Information to Identify Plagiarism” shows the effectiveness of linguistic information to detect similarities among different words to express the same material. This experimental study goes beyond just using

¹¹ <http://wordnet.princeton.edu/>

synonyms, it “presents a set of low-level syntactic structures that capture creative aspects of writing and show that information about linguistic similarities of works improves recognition of plagiarism” [Uzuner et al., 2005]. This research experiment identifies classes for different syntactic expressions for the same content, called “syntactic elements of expression”. These elements of expression include: different variations of initial and final phrases of a sentence, argument structures of verb phrases and syntactic classes of verb phrases. All possible variations are considered to combat initial and final phrase structure alterations.

For example, a sentence may have following class of three different expressive alterations:

- (a) Martha can finally put some money in the bank.
- (b) Martha can put some money in the bank, finally.
- (c) Finally, Martha can put some money in the bank.” [Uzuner et al., 2005]

This research experiment also enriches its syntactic elements of expressions by employing Levin’s classes [Levin, 1993] of verbs. In Levin’s classes verbs are classified using various syntactic alterations a verb is subject to, and the classes of verbs with similar meanings. These features are combined to create further elements of expression for testing data (including English translations of literary work by different translators). This data is then used for recognition of paraphrased writings with similar contents. Although this is a computationally expensive approach compared to conventional content recognition approaches such as comparing tf-idf weighted keywords, function words, distribution of word lengths and sentence lengths, the results presented show a significantly better average of similarity detection over baseline/conventional approaches [Uzuner et al., 2005].

There are services available that evaluate the text contents on a conceptual level for automated essay grading. They compare semantic similarities among contents (written essay and domain knowledge) to calculate grades. A method used in such systems is “Latent Semantic Analysis” (LSA). This is a statistical technique for extracting and representing the similarity of meaning of words and passages by the analysis of large bodies of text” [LSA, 2006]. A matrix of words and related segments is used to build a word to concept semantic domain space. The text needed to be checked for similarity with this domain space is also represented in document vector form. If the document vector is similar to the model answer vector (again the measure of angle between vectors defines closeness to each other) in this domain the document will have higher similarity grade. This kind of system which detects semantic similarities to grade some writing can also be used effectively for paraphrased plagiarism detection. But even with a singular value decomposition approach in LSA to reduce word and context matrix, the matrix dimensions are still large and the vector space analysis is computationally demanding.

As mentioned before, in the case of plagiarism detection we are usually dealing with a very large corpus of textual information making such analysis not as yet practical. This necessitates methodologies to enhance processing and making the methods mentioned feasible for practical environments.

Another approach utilizing the power of Normalized Word Vectors (NWV) is to further reduce the word-concept vector space by normalizing all words to a thesaurus root word. The convergence to a singular concept word reduces the domain space and document vectors significantly. The cosine similarity measure can then be used to find semantic relevance among answers [Williams, 2006]. This in turn leads to a reduced computational load and can perhaps make such methodology practical for plagiarism detection.

A more generic technology of query formulation is being investigated which use NWV technology and dynamic ontological filtering to help extend the semantic similarity detection mechanism in various applications [Dreher and Williams, 2006].

It is interesting to note that some times less computationally demanding simple text structure analysis techniques such as average word lengths, sentence counts, words per sentence etc. can be very useful in cases of suspected plagiarism in different documents. A simple example in Figure 25 show the use of sentence and word counts to determine style similarity between two paragraphs in the test case used before, where simple synonym replacement made similarity undetectable using conventional plagiarism detection services. We developed a simple program to calculate the standard deviation of the difference vector of the sentence lengths (calculated on the basis of number of words) of two suspected paragraphs. This can be a good indicator of text structure similarity, and can be used to identify potentially similar documents.

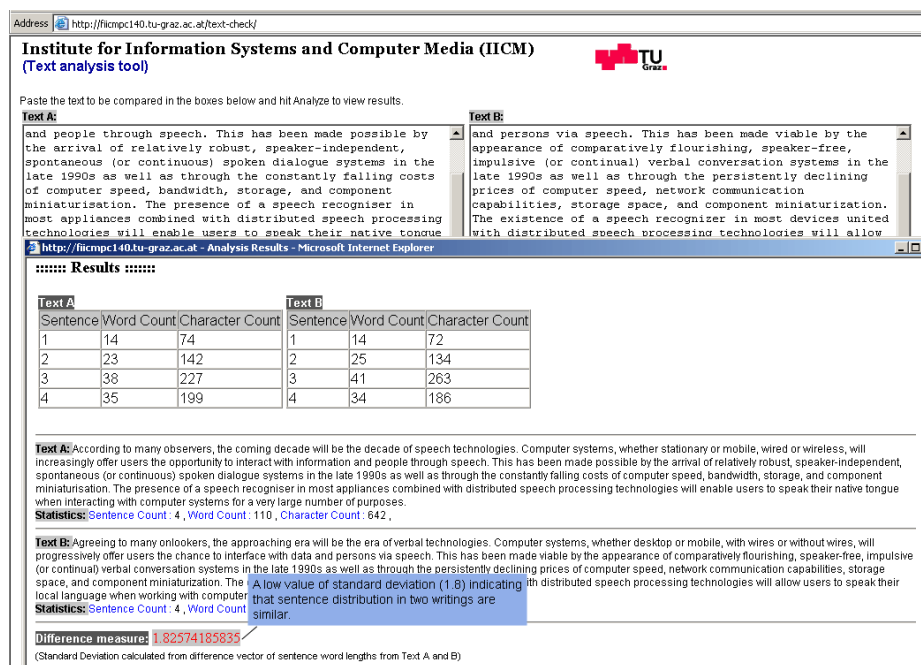


Figure 25. Statistical text structure analysis

Statistical analysis may determine a preliminary similarity measure. Suspected parts can then be put to further more advanced semantic or syntactic testing algorithms to confirm the detection.

3.8 Problems and Visions

Looking at the extent of the problem, it is quite obvious that academia requires tools and services to automate and enhance plagiarism detection. Our analysis of these tools revealed a number of areas which need attention.

3.8.1 Access to deep web

The invisible contents of World Wide Web not indexed and available through common search tools are considered to be 500 times larger than commonly linked and searchable web. It is very important for a plagiarism detection tool to have strong and wide spread access and search capabilities. Generally available tools claim to have self maintained indexes, and use partner web search service. No single index or search service can cover the immense deep web archives.

3.8.2 Plagiarism detection of multimedia contents

Almost all plagiarism detection tools and services are used by academia. Images have always been an important part of all types of scholastic documents. However none of the available systems provide mechanism to detect copied image contents. Musicians, painters, videographers use preventive measures like watermarking however they don't have access to tools for detecting copyright violations.

3.8.3 Semantic plagiarism detection

A plagiarism detection tool must have some mechanism of detecting similarity beyond exact words match. This is necessary otherwise a simple thesaurus tool can be used to beat the detection system (as already demonstrated). The addition of semantic capability to similarity detection may add noise to match process. It is necessary to control the level of abstraction with good word sense disambiguation.

3.8.4 Intrinsic characteristics check

Commonly used services and tools perform cross document comparisons for copy detection. They do not utilize intrinsic document characteristics to determine plagiarism checks. Quantification of these characteristics can help detect inconsistent text and possible cases of copy in document without any external reference.

3.8.5 Cross language checking

Scholastic articles are available in number of languages. The availability of very good translation services gives the possibilities of cross language duplication of work. There is no automated tool or service available to check for content similarities in different languages.

Almost all tools and services produce results that cannot be used as a final report without human interpretation. The problems pointed out by the system have to be analyzed by domain experts for verification and further investigation. This limitation suggests more work is required to adapt systems to provide an analysis layer that triggers further investigative matches and produces a more conclusive result. A viable solution will probably have to be interactive, with feedback from the examiner to confirm system assumptions before proceeding with additional analyses.

The results of research studies and experiments described in the previous section shows an increasing awareness of problem and availability of tool to fight it. However, to date, we found no evidence of any released tool or service which uses language information, syntactic, intrinsic, multimedia and semantic aspects of writings to detect plagiarism. Current detection tools are lagging behind without having broad and generic ontology of linguistic or writing parameters which convert the search patterns to a certain level of abstraction.

Increased ease of access to global and multilingual contents makes detection of translated plagiarism a vital requirement for detection systems. The detection services can use translation tools to convert foreign language contents into a basic English form, apply normalization techniques to generate a generic index of document sources and apply semantic similarity checks for detection. To illustrate what we mean consider the following example

Synonym classes in German:

{Cabriolet, Cabrio, Zweisitzer, Automobil, Personenauto, PKW, Auto, ...} --> Auto

{tiefblau, azurblau, türkisblau, blau, ...} --> blau

{Klatsch, Plumps,...} --> Lärm

{fallen, sinken, herunterfallen, hinunterfallen} -->fallen

{laut, heftig, stark, groß,...} --> groß

{Bach, Fluss, Teich, See, Wasser, ...} --> Wasser

Synonym classes English:

{cabriolet, car, limousine, automobile, ...} --> car

{deep blue, azul, azure, sky-blue, dark blue, ...} --> blue

{splash, splish, ...} --> noise

{fall, drop, ...} --> fall

{loud, strong, great, big, ...} --> big

{creek, brook, stream, river, pond, pool, lake, ...} --> water

Let us now see, how the two sentences: "Das azurblaue Cabriolet fiel mit lautem Klatschen in den Bach" (German) and "The deep-blue limousine dropped with a big splash into the river" (English) can be determined to be similar:

The sentence:

"Das azurblaue Cabriolet fiel mit lautem Klatschen in den Bach" is converted using grammatical rules (such as stemming, conjugation, etc.) and employing German synonym classes to:

“blau Auto fallen gross Lärm Wasser”

A machine translation of this will provide: "blue car fall big noise water".

The English sentence “The deep-blue limousine dropped with a big splash into the river” is converted using grammatical rules (like reducing to singular, nominative, infinitive, etc.) and synonym classes to: “blue car fall big noise water”

Using such an approach, the two sentences have been proven similar.

Another functionality lacking in existing systems is the ability to process textual images for similarity checks. Sometimes one has to deal with textual information in scanned format. Most of such images contain text in typed form which can be very accurately converted to text with the use of Optical Character Recognition (OCR) engines.

The missing components in existing systems also include better tabular information processing, proper support for foreign language characters, reference validity and relevance checks. It is likely that next generation high quality services for plagiarism detection will have these missing links.

3.9 Enhancements in plagiarism detection systems

In order to address shortcomings identified in previous section, we worked on the development of number of experimental systems. These systems aim to extend discovery and search capabilities and strengthen the relevance detection through combined use of similarity measures, searching services and Natural Language Processing. Following sections describes the common platform (CPDNet) used to implement these experiments. The two prototypes along with experimental results described in this chapter address the issues of deep web access, and show our efforts to extend plagiarism detection for non text contents.

3.10 Service Oriented Collaborative Plagiarism Detection and Prevention

The service oriented IPR framework explains how collaborative efforts in terms of technology and content, can help improve plagiarism detection and prevention. It presents a web service oriented architecture, which utilizes the collective strength of various search engines, context matching algorithms and indexing contributed by users. The proposed framework is an open source tool, yet it is extremely efficient and effective in identifying plagiarism instances. By creatively using distributed processing capabilities of web services, this tool offers a comprehensive mechanism to identify pirated contents. With an aim to extend current plagiarism detection facilities, the proposed framework not only tries to reduce known deficiencies but also aims to provide plagiarism protection mechanism. The distributed indexing approach adapted in the system provides scalability to examine deep web resources. Network nodes with more focused indexing can help build domain specific information archives, providing means of context aware search for semantic analysis.

As mentioned before, the ease with which digitized contents can be accessed accounts for the rise in plagiarism. Without a doubt, the ease of content availability is an attribution of internet usage. Naturally the most popular tools used to detect plagiarism are also built on the idea of efficiently checking for document source

availability over the internet. The commercial services claim to use personalized crawlers and up-to-date internet indexes for a comprehensive check. Over the years these programs and services have indeed proven their effectiveness in educational and industrial environments. However, there is still room for considerable improvements. A recent survey on plagiarism [Maurer et al., 2006] is a good starting point for a better understanding of various plagiarism detection strategies and strengths/weaknesses of available tools. Experimental results in the referenced survey suggest that in order to have a more precise plagiarism detection tool, the inspection system requires broader and an up-to-date content index, added semantic elements for similarity check, cross language content similarity detection and finally a mechanism to verify the findings. Existing tools following either desktop applications or software as a service approach lack these capabilities. Albert Einstein once said "The secret to creativity is knowing how to hide your sources", and yes, plagiarists today are more creative. Copied contents are often not publicly available or modified in a way which is hard to detect using existing applications and approach. Further experiments to benchmark capabilities of popular plagiarism detection services revealed that intelligent use of good search engines can greatly add value to plagiarism detection applications [Maurer and Zaka, 2007].

As an attempt to fulfill the needed requirements in plagiarism detection systems, collaborative service oriented architecture for plagiarism detection is presented. The proposed service oriented collaborative network openly available to educational community aims at extending the existing similarity check methods in the following ways:

- i. It offers a seamless, combined use of multiple search services. This technique provides a broader and more context aware internet search, which proves to be more revealing than any single searching and querying approach.
- ii. Collaborative authoring and indexing of document sources at each node enhances the search capabilities with addition of documents not available publicly. This also provides users an option to add intellectual contents for protection against copyright infringements. Participating institutes allow access to deep web, hidden from normal search engines.
- iii. The system provides multiple services for search result analysis. More services can be added to the system due to its modular nature. The user has an option to use mere text matching to deduce similarity or can apply writing structure analysis, semantic or cross language analysis.
- iv. The system offers the possibility of synonym normalization and translation in collaborative search service and peer node indexes. This adds semantic matching capabilities not possible in conventional internet searches.

This system makes use of off-the-shelf tools (web services) and user contributed contents to extend plagiarism detection. Its pluggable services constitute composite web applications offering flexibility and variety in use.

Having described the basic idea behind the service oriented collaborative plagiarism detection network, the following section describes the conceptual design of the system. Section 3.12 describes a practical realization of the architecture and section 3.13 compares results of the prototype with other services.

3.11 Concepts behind service oriented architecture

Service Oriented Architecture (SOA) can be described as a heterogeneous environment of applications with self describing and open components which allow inter application communication. SOA offers distributed and collaborative computing infrastructure over the network or internet. A research study for the future of flexible software [Bennet et al., 2000] provides a vision of personalized, self adapting and distributed software environment. The software is structured in small simple units which co-operate through rich communication structures. The collaborative units work in a transparent way to provide a single abstract computing environment. The study shows interdisciplinary approach would be critical to developing a future vision of software. A significant proportion of software and associated data does not exist in isolation but in a political, social, economic and legal context. In order to have applications with high level of productivity and quality, it is essential that they don't have rigid boundaries but offer rich interaction and interlinking mechanisms with users as well as other applications.

The service oriented approach has been in use for almost a decade and adds the aforementioned functionalities in software systems. These integration technologies exist in the form of Component Object Model (COM), Distributed Component Object Model (DCOM), Enterprise JavaBeans (EJB) and Common Object Request Broker Architecture (CORBA). However what really boosted the concept recently is the emergence of the next generation of SOA based on "web services". Web services are built using standardized and platform independent protocols based on XML. The service components enable us to build a user-tailored, distributed and collaborative web application environment. A framework built on top of web services will offer the extension and flexibility to plagiarism detection as described in the introductory part.

3.11.1 Web service model

"A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-process able format. Other systems interact with the Web service in a manner prescribed by its description using SOAP¹² messages, typically conveyed using

¹² Simple Object Access Protocol, <http://www.w3.org/TR/soap>

HTTP with an XML serialization in conjunction with other Web-related standards” [W3C, 2004]. A typical web service can be described using three components

1. Description: (XML based service description, specifically WSDL¹³)
2. Publishing and Discovering (Registry, index or peer-to-peer approach of locating services, e.g. UDDI¹⁴)
3. Messaging (XML based message exchange over the network, specifically SOAP or REST [Fielding and Taylor, 2002])

The proposed collaborative plagiarism detection framework consists of composite web applications to search the internet and shared document sources. These network distributed applications use a set of web services for searching and sharing documents.

Web service interaction can be either synchronous or asynchronous. Commonly available and popular internet search web service APIs use synchronous request/response communications. This approach works well in limited use environments where the web service can process a request in quickly. However, in plagiarism detection, search normally requires exploring the internet for a large number of queries (moderate size finger prints of a document) or browsing through document signatures from a number of distributed nodes. In this scenario using asynchronous service interaction for the user is the better solution.

The proposed framework consists of a service proxy that enables asynchronous use of synchronous internet search APIs. The time independent interaction model (asynchronous) is implemented using multiple synchronous request/response web services. The first service initiates processing from the end user by sending the information parameters. The service sets an identifier of the submitted job and responds to the end user with same. The end user can then use the second service and the identifier as a parameter to check if the submitted job is complete, pending or failed. [Hogg et al., 2004] The first request in asynchronous communication mode validates and acts as a buffer between the end user and the synchronous internet search service. The submitted job is processed using search and analysis services at the respective network node. The similarity detection results are stored and the job identifier status is updated for later reference of the end user. Figure 26 shows an overview of CPDNet’s service linked architecture. Further details of web services workflow is discussed in Chapter 4 (section 4.12.2).

¹³ Web Services Description Language, <http://www.w3.org/TR/wsdl>

¹⁴ Universal Description Discovery and Integration, <http://www.uddi.org/>

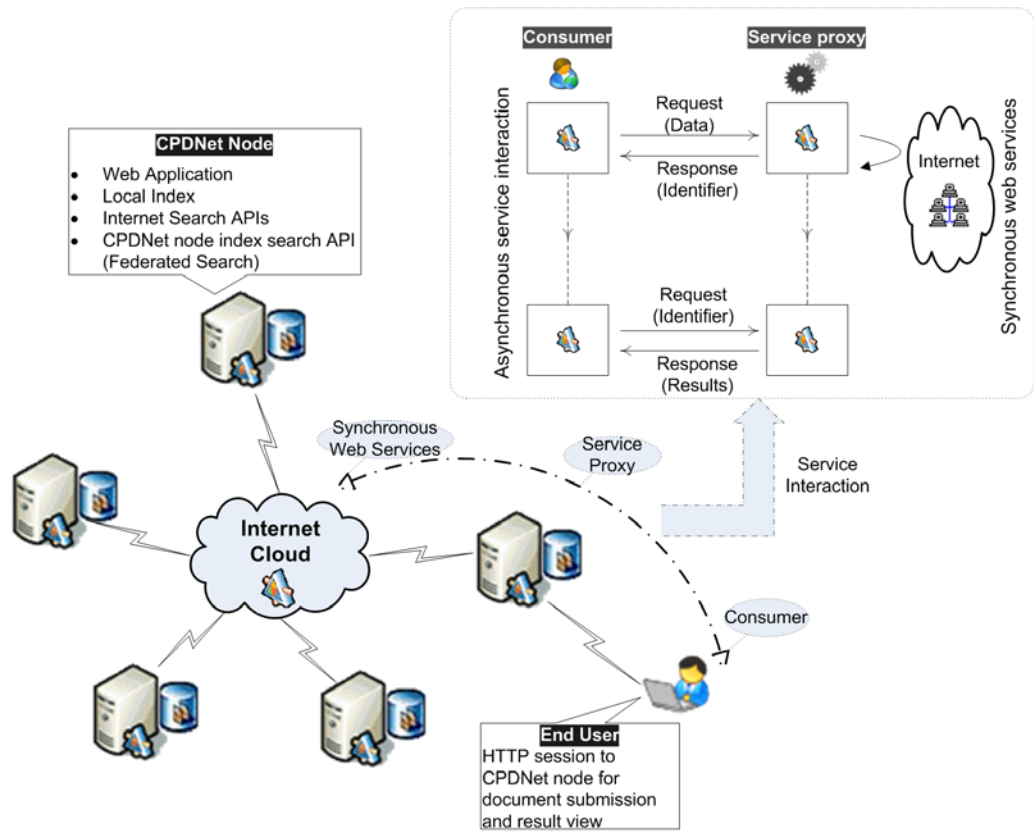


Figure 26. Collaborative Plagiarism Detection Network Overview

3.11.2 Mashup of search and analysis web services

One of the major strengths of the system is the next generation search capabilities termed “Search 2.0” by Ezzy [Search2.0, 2006]. It is defined as a search "designed to combine the scalability of existing internet search engines with new and improved relevancy models; they bring into the equation user preferences, collaboration, collective intelligence, a rich user experience, and many other specialized capabilities that make information more productive" [Search2.0, 2006]. In the concept described here, users are given the option to select a number of system compatible internet & collaborative search services. The search results are processed and passed through further analysis algorithms in order to detect content and context similarities. Combining the strengths and scalability of existing internet search engines broadens the web search scope compared to searching via a single source. Further mashup with collaborative search API built using full text query mechanism on user contributed finger print data and local node resources greatly add to value. The collective search is not conventional meta-search where the user might have to weed through irrelevant matches. The initial result set lists the possible matches by each search service. Analysis services applied to search results produce precise and productive output for the final report.

The system has been tested using a few popular search services. The results of our experiments presented in a later section indicate that using the search services systematically can detect cut paste plagiarism more effectively than any other commercial plagiarism detection service. This is mainly because of recent open access and indexing initiatives by publishers. More and more options are becoming available to do full text search on digital libraries via a particular search engine or a library's own search mechanism. One significant example of such an initiative is Crossref search pilot. A group of 45 leading journal publishers including ACM, IEEE, Blackwell, Springer, Oxford University press and John Wiley & Sons, are providing full text search options using Google via Crossref gateway [CrossRef, 2007]. A plagiarism detection system with up-to-date search capabilities can outperform similar tools of its class. The proposed service oriented approach gives its user an option to integrate any existing search service and any upcoming more powerful search service.

The initial prototype includes an analysis services based on the vector space model [Wikipedia:VSM, 2007] approach to measure cosine similarity. The queried text and search engine's returned matching snippet are converted to word vectors, based upon the vocabulary of both. The angular measure (dot product) of vectors is used as a score to determine similarity between the queried text and any searched result. The combination of the highest similarity scores of the queried text segments represents the percentage of plagiarism in a document. There is a number of other possibilities for similarity analysis within a document or with the search service's detected contents. One such analysis approach tested for the proposed framework involves a structural comparison of suspected documents. This statistical analysis service uses a measure of standard deviation in the document structures (lines, words) to determine a possible match.

Another analysis planned to be part of the framework is stylometric analysis based on Jill Farrington's CUSUM (cumulative sum) technique [Farrington, 1996]. The CUSUM technique is based on the assumption that every person has some quantifiable writing or speaking habits. The measure of consistency of these habits can be used to determine single or multiple authorships. The numerous factors which determine authorship include checking of sentence length consistencies, checking the use of function words, nouns and other common language practise throughout the document. This technique is used by courts in England, Ireland and Australia to determine authenticity of writings in different cases such as witness statements, suicide notes, ransom notes and copy right disputes. Although this technique may not be considered very effective, especially in the case of multiple authors, it can be beneficial in pointing out any suspicious portion in the text coming from a single author. The suspected parts can then be checked by other more extensive search services. Future research which could be conducted on the system also includes the development of semantic analysis service that uses language ontology. The idea is further described in section 3.12.1.

3.11.3 Collaborative authoring, indexing & searching – Access into the deep web

The ability of collaborative authoring and indexing at participating institute nodes of network is an important feature in extending plagiarism checks. The motive behind collaborative indexing and search approach is the fact that conventional search engines only index the shallow internet contents and do not cover deep web contents. Shallow contents are generally static web pages linked with each other and openly available to search engine spiders. However the deep web consists of unlinked or dynamically generated pages, databases, protected sites, intranets and contents behind firewalls. These contents are invisible to the index of general internet search engines. A study by BrightPlanet in 2001 estimated that the deep web information is 500 times larger than the commonly defined World Wide Web [Bergman, 2001]. It is very important to have access to this massive information base for thorough plagiarism checks. Educational institutes usually have a very large collection of un-linked and non-indexed local contents. Institutes and research groups within an institute also have privileged access to, and better knowledge of specific deep web resources. This access and knowledge enables them to gather resources not commonly available. Collaborative networking provides the means of creating a local searchable index of these resources. Any network node run by an institute can setup a search gateway service providing access to its invisible contents and can access to protected digital libraries. A collaborative search API consumes the local search services according to the access policy of each peer node. The collaborative search access produces limited results usable for similarity analysis services. The search results may only contain specific matching portion and associated meta information. A local index can be restricted to a specific domain e.g. an institute specializing in computer science articles. Collaborative searches can be made context aware by selecting domain specific peer indexes of the deep web. This means that in addition to general internet search services; the proposed system also use collaborative search service which harnesses the deep web contents of participating nodes.

The collaborative search channel is also important in terms of reducing the dependency of certain search engines. Researchers have shown concern in recent studies that the search engine monopoly gives them the role of gatekeeper in the information flow. A search engine can determine what is findable and what is kept outside the view of the common user [Kulathuramaiyer and Balke, 2006]. The view restriction or any other implication a search engine may apply or is applying can be seen in the form of web search API switching from Google. Shifting from an XML standard and generic SOAP based access to a more restraining AJAX API is not seen as a welcome move by many research and development forums. It is thus imperative to have an alternate and more open channel of searching the intellectual content base.

System users can contribute documents to the associated network node for indexing. User contributed content authoring is done either by conventional

indexing and making complete documents openly available. Or by generating moderately sized searchable plain text snippets of submitted document (called fingerprints or signatures in more abstract form). In the case of a search match, only a content snippet and meta information are sent from the index, not the complete document. Any matches found in such snippets point to the original source for further verification. Authoring resources can be tempting for a user or node administrator, because of following reasons

1. Contributing resources can expose the contents to all network search APIs in a protective manner. This approach helps where users cannot index complete contents in a finished formatting for the public internet.
2. User contributed authoring acts as a “personal copyright tool” which protects against any possible piracy of articles, personal blogs, assignments, presentations etc. Submitted documents can be indexed with the author’s meta information. Users or node administrators may choose to receive alerts produced by any similarity matches from other sources in the future. This can help authors keep track of legal or illegal use of their contents.
3. The envisioned framework in its mature form is based on P2P incentive based resource access scheme. Users and nodes with a higher index of shared resources will receive better access to local resources of peer nodes.

3.11.4 Service publication, discovery and access mechanism

Web services for end users are available as selectable index of compatible searching APIs. No technical details or WSDL is required at the end user level. User can choose any service by simply selecting or providing personal usage credentials e.g. API code or key. Master nodes keep a well descriptive index of available services to share and search. The system administrator of each node can incorporate the available services on a specific node and make them available to the end user. The local document source (collaboratively authored) sharing services at each node uses either an open access policy or implements restrictions on access. Peer nodes may contribute more search services and sample service consuming codes to master service index. Initial implementation uses a plain index approach and open access policy at selected test nodes. Later stages of the project include a more controlled central registry to maintain service descriptions and access mechanisms.

3.12 CPDNet Implementation

Based on the abstract architecture which is described in the previous section, a partial implementation is developed as a proof of concept. The prototype serves as a valuable tool to benchmark and test the search engine capabilities, the match detection algorithms and the document source generation. Initial experiments show

very promising results closely comparable (better in some cases) to already existing commercial services which detect plagiarism. Prototype named CPDNet¹⁵ (Collaborative Plagiarism Detection Network) is available for test purposes, although it is an early stage of development. Users may register for an account with their personal Search API code to test drive the system. CPDNet currently supports Google SOAP search API¹⁶, and Microsoft Live Search API. The server and client for web services are created using PHP SOAP and AJAX technologies. Running nodes can choose any available indexing server to link local contents with collaborative search. Existing CPDNet nodes use Lucene, an open source Java based indexing and search technology. Result sets are generated in OpenSearch¹⁶ standard. The collaborative search client uses a SOAP interface to discover matches from all available service nodes.

The process of detecting plagiarism includes the following steps

1. The submitted document is broken down into moderately sized text chunks also called fingerprints. This document source can also be marked for indexing in the local database, accessible via a collaborative search API.
2. The plagiarism check is initiated by querying the internet using the fingerprint data. The selected search APIs searches the web. Locally indexing document sources (signature in more abstract form) and that of peer nodes can also be queried if collaborative search service is enabled. The normalized signature generation and indexing process is described
3. Most relevant matches obtained via the search services are passed to similarity analysis service. The existing active service uses word vector based similarity detection as described earlier.
4. Fingerprint similarity scores of a document are calculated to determine the plagiarism percentage. The document text linked with the similarity scores, matching contents and source links, is presented to the user within a final report.

The described process steps are visualized in figure 27. Here you can see the various services used in the system.

¹⁵ Collaborative Plagiarism Detection Network: <http://www.cpdnet.org>

¹⁶ OpenSearch: <http://opensearch.a9.com/>

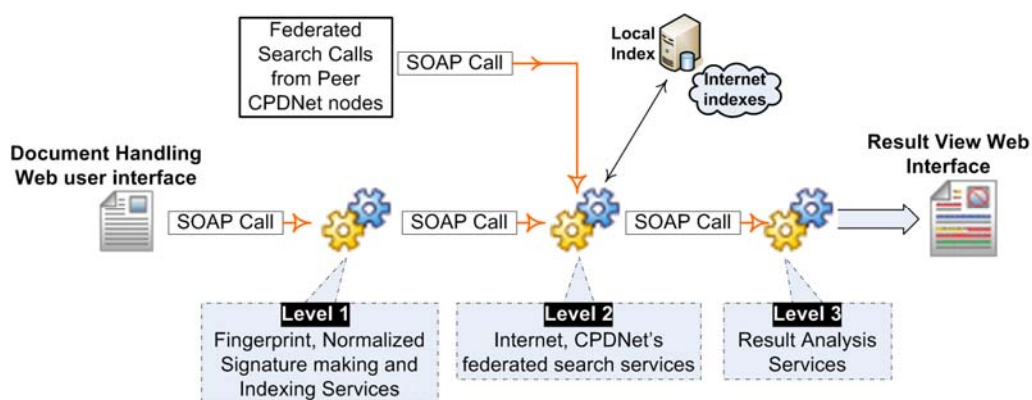


Figure 27. Web Service flow in CPDNet

The architecture is flexible to accommodate numerous services at each level. The running services in the current prototype can be further explored at the project portal.

3.12.1 Towards a semantic plagiarism detection service

To trace paraphrased and cross language plagiarism, algorithms are required to discover similarities on the semantic level. This kind of analysis requires detecting similar word replacement (synonymizing), word deletion, word addition and translation etc. The application of these checks on a large scale with conventional internet indexes and current search APIs seems far-fetched and computationally very expensive. However, the proposed framework provides a mean of indexing submitted contents in a normalized form. The normalized contents which are shared at each node can be queried using a collaborative search API of peer nodes. The queried finger print is also normalized in order to determine its conceptual equivalence. The semantic level similarity check can certainly help its users in more than just plagiarism detection. The findings can also be used by knowledge workers to discover relevant contents already available on internet. In the near future, the focus of this project's research will include following:

3.12.1.1 Fingerprint normalization into generic signatures

A system component for generation and indexing of signatures (semantic fingerprints) is being developed and tested. This component will normalize the submitted contents to a root level with the help of a POS (Part of Speech) tagger and language thesaurus. Initial development includes modification in crawling and indexing process of open source index server that constitutes collaborative search service nodes of network. The crawled contents are passed through a POS tagger, this process provides the exact sense of a word which then is normalized to a basic form with the help of WordNet SynSet thesaurus. The content in this basic language form is then processed for indexing. The query for such an index is again treated for normalization to develop a semantic match.

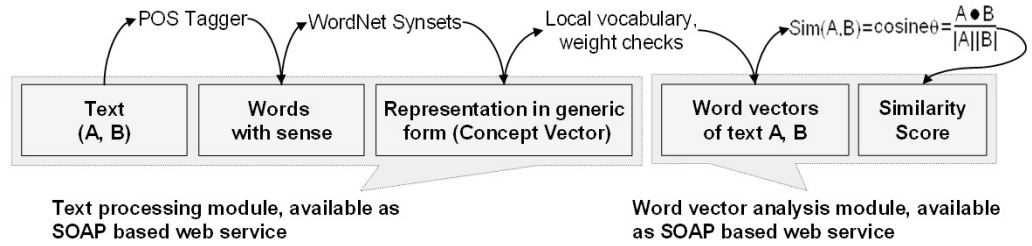


Figure 28. Process of normalization of text

The process of normalization of text adds conceptual plagiarism check capability in system, such conceptual check is available for documents that are indexed in local CPDNet nodes. Further details and example of process is available in indexing section of project portal¹⁷.

3.13 Results of CPDNet prototype

In order to benchmark the system, a document corpus is generated with various proportions of plagiarized contents from both deep and shallow internet. The test document set consist of undergraduate student assignments, and manually tailored documents that contains plagiarized contents from access restricted intellectual archives such as IEEE Xplore, ACM and SpringerLink Digital Library. Search APIs of Google, Microsoft Live, Yahoo were used for coverage of internet public index (standard check). For testing standard and conceptual plagiarism checks a local index of 1174 documents is created. The documents come from 164 issues of an online digital journal¹⁸. Table 3 shows statistics of standard and normalized local index maintained using Lucene based CPDNet node.

Type of index	# of documents	# of terms	Size (KB)
Standard	1174	388467	23949
Normalized	1174	366013	23391

Table 3. CPDNet Index statistics

Test results from the selected corpus show significantly better similarity detection capabilities of the system compared to other services. The graphs in figure 29 give an overview of the plagiarism detection capabilities of CPDNet. Better plagiarism detection by the system developed to date, is due to the enhanced SOA based searching capabilities. Compared to other systems that claim to use their own index and search mechanism this system makes use of broader more up to date discovery approach with multiple search services.

¹⁷ <http://www.cpdnet.org/indexer>

¹⁸ JUCs: <http://www.jucs.org>

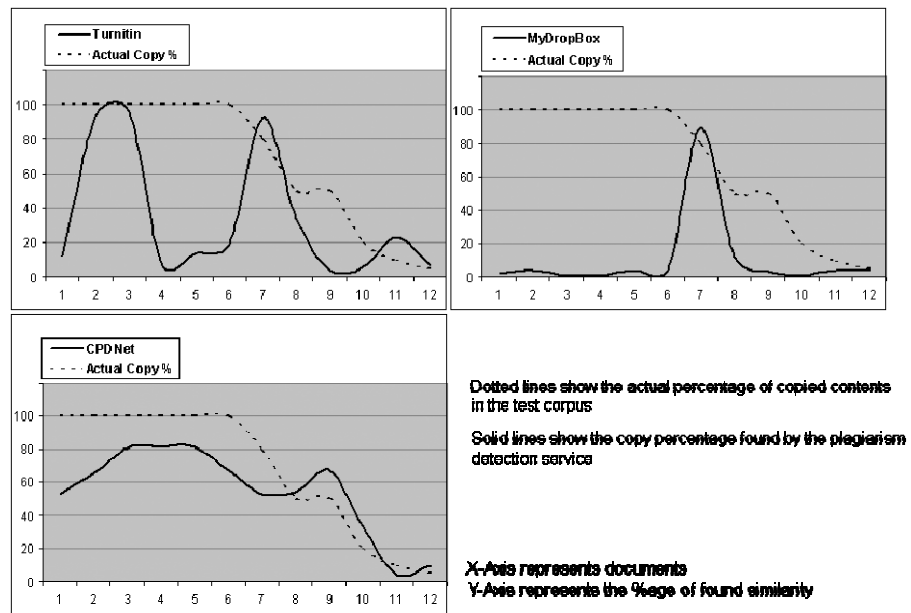


Figure 29. Comparison of search and similarity detection capabilities

Better plagiarism detection by the system developed to date, is due to the enhanced searching capabilities added to the system.

We have tested Turnitin®, Mydropbox® and CPDNet and other tools with various sets of documents. However, to keep things simple we will just report on the findings for Turnitin®, Mydropbox® and CPDNet using two very dissimilar sets of documents.

The first set of documents consisted of 90 term papers in the (2005) undergraduate year at our university. The results for the first 40 of those paper is shown in Figure 30, the result for the other 50 papers is very similar.

The total percentages of overlap of each student essay with documents on the Web are shown in the figure 30, the bars showing the result of Mydropbox®, Turnitin® and CPDNet, respectively. Note that papers 13, 19, 21, 22, 30, 31, 36 and 38 show around 20% or more for each of the tools. It seems surprising that our home-baked solution CPDNet (with Google API) is doing so well, is actually also identifying 8, 9, 27, 28, 29, 39 and 40 close to or above the 20% threshold.

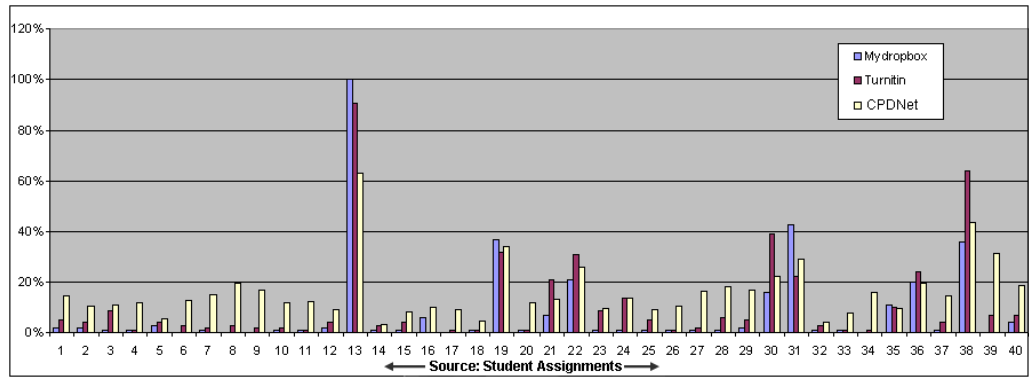


Figure 30. Comparison with student papers

We will explain this surprising result after discussing Figure 31. It shows the analogous comparison for 40 documents, this time taken from documents in journals that are not available free of charge, and in none of the databases searched by Turnitin® and Mydropbox®.

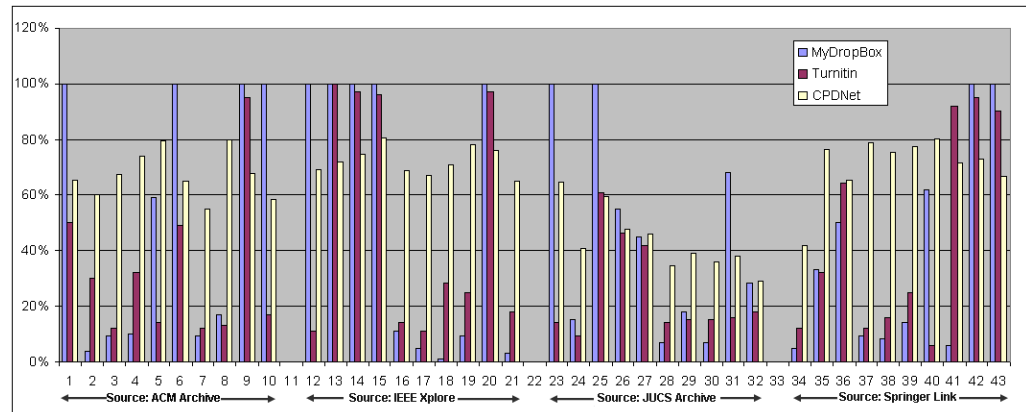


Figure 31. Comparison of papers not accessible on the Web without charge

Since those documents are actually available verbatim on the web, all tools should show 100% plagiarism. However, as can be seen from the diagram both Turnitin® and Mydropbox® do not recognize more than 15 of the papers as plagiarized. However, CPDNet shows all documents high above the threshold. Thus, CPDNet is the most successful tool. As designers of CPDNet we might be happy with the result. Yet we are not. We are not doing anything better than Turnitin® or Mydropbox®, but we are using multiple services (Google, Live, and Yahoo BOSS APIs) in addition to a home-grown search engine. And all these search engines are evidently indexing many more Web sites than the other search tools are using, including small sites where authors who have published their paper in a journal keep their own copy on their own server: free, and hence detectable by search service mashup.

3.13.1 Alerting service

The developed system provides an alerting service to its users. Authors can use this similarity notification service for submitted documents. It automatically notifies authors about matching contents over the internet or in local repository. Author can set the frequency of running internet search and email reception. In case no new matches are found during periodic scanning process, user will not receive any notification email. This adds a protective measure against possible IPR violations in future. Figures 32 show the use of alerting service for CPDNet node of JUCs archive available at <http://jucs.cpdnet.org>

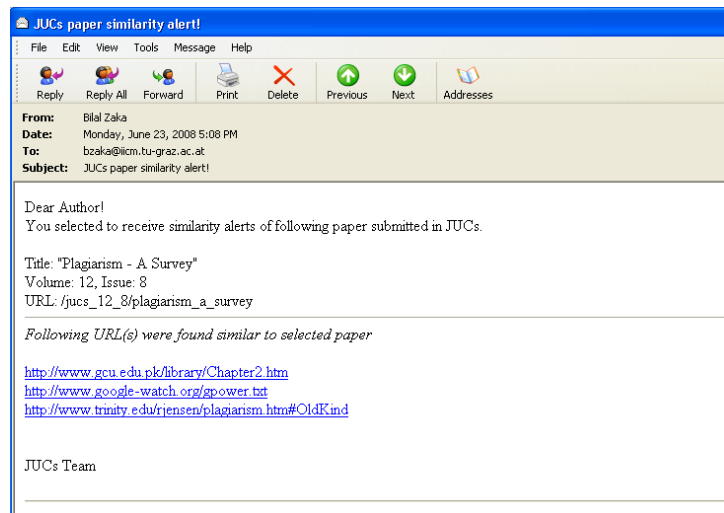
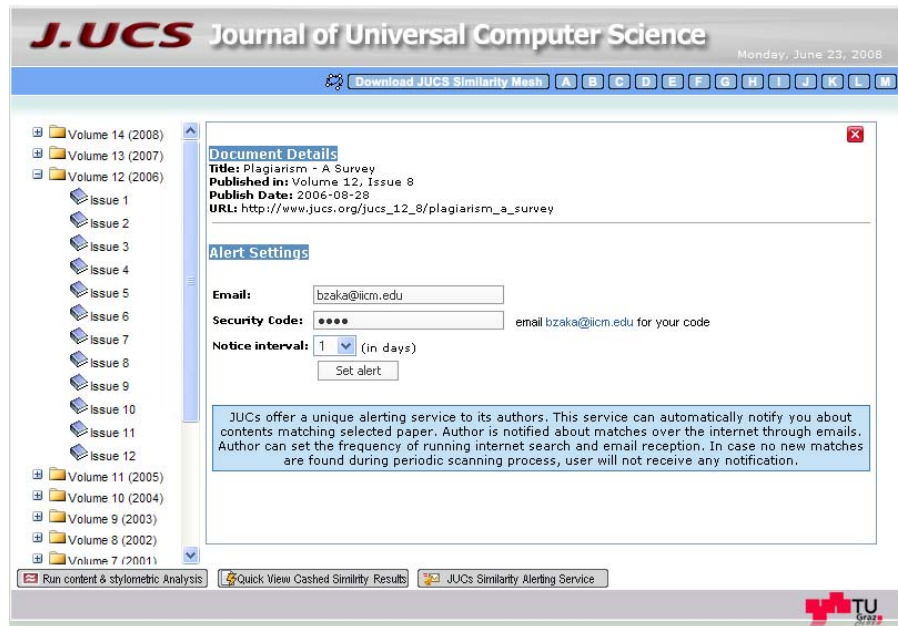


Figure 32. Alerting Service interface added to JUCs CPDNet node

3.14 Plagiarism in Virtual Worlds

Plagiarism is not confined only to text or educational world. The wide spread digitization makes audio visual contents equally vulnerable to the plagiarism problem. The absence of effective automated systems for discovery and detection of plagiarism in non text domain is already pointed out in problems and vision section of this chapter. Following work looks at contents theft in different domain (Virtual Worlds) and describes a mechanism for detecting plagiarism in non-text contents.

Virtual world communities are expanding at a rapid pace; one prominent example is Second Life with a current registered user base of 16 million and approximately 1 million active users [SecondLife, 2008]. The virtual world residents retains the intellectual property rights of originally created objects *e.g.* clothing items, images, textures, scripts, 3D objects (sculpties) etc. The virtual economies of such virtual worlds rely on the transactions of these objects. Like the IPR violations in real world digital world also suffers from the same issue of theft and unauthorized use of virtual goods. The problem is alarming because of the ease of theft in such environments. Starting from CopyBot [Wikipedia:CopyBot, 2008], currently there are a number of tools available that can capture objects along with associated Universally Unique Identifier (*UUID*). A little digging into the Second Life related blogs and users discussions will show the seriousness of content theft issues [Baily, 2008].

In January 2007, Linden Labs released the entire source code of the Second Life client along with the client server communication protocol. This allowed developers to improve and modify the entire communication between the Second Life network and development of customized clients. Besides Second Life, the open source project openSimulator¹⁹ introduced an open source 3D application server based on the Second Life communication protocol. A 3D application server is part of a virtual world and is responsible for one specific region. It provides the necessary computational power for the region, *e.g.* the login process for the clients or the connection to other application servers. Further, it is connected to a asset server that stores the entire inventory objects of avatars. All items on the asset server have a unique identifier which is a 128 bits Universally Unique Identifier; we will further refer to it as *UUID*. Due to the effort of the openSimulator project there are several grids beside Second Life that provide the infrastructure for virtual worlds. This infrastructure includes asset servers, a login server, or a list of all connected application servers. As in Second Life, users can create a grid specific avatar and log into this grid with the client software. The communication protocol is the same as in Second Life so one can use the official Linden viewer. The user only has to change the IP address of the login server to the specific login server of the grid.

¹⁹ www.opensimulator.com

Besides commercial grids like the *openlifeGrid*²⁰ there are also free grids that allow individuals to connect their servers to a network without charges. One example is the *OSGrid*²¹ with about 1000 application servers²². In contrast to Second Life there is no economy in *OSGrid* which implies that users can upload images without any charges. As in the Second Life virtual world, users can upload their images in various formats but on the server side these images are all stored in Jpeg2000 format. The upload process is the same as in Second Life. The user can choose the images and the client uploads them to the asset server. After that the images appear as items in the inventory folder of the avatar. The interoperability recently announced by IBM and Linden Lab [Linden, 2008] and possibilities of future linkage between asset servers of OpenSIM and Second Life will raise more issues of object security.

In response to such copyright allegations, Linden Lab is adopting the procedure formulated by DMCA (Digital Millennium Copyright Act) to give notifications of copyright infringements to service providers and concerned parties [DMCA, 2008]. The DMCA procedure in Second Life may result in taking down the stolen contents from the asset server, issuance of warnings, counter notification, account termination, and formal lawsuit.

However, there is still no platform for virtual world content creators that identify the object theft in OpenSIM or Second Life. The availability of such a platform will allow users to register their object to the system. This in turn will help identify matching textures with *UUIDs* different than the original. Additional meta information available in such an index (*e.g.* registration date) will help in DMCA procedure. There is also no means of easily identifying copied contents spread across a large user base and simulators. Interoperability between OpenSIM and Second Life grid [Linden, 2008] and future exchange of assets will create further problems in this regard.

Looking at the problem we suggest the use of a content theft detection approach generally adapted by educational community for textual contents. The proposed system makes use of CBR (Content Based Retrieval) technology [Yoshitaka and Ichikawa, 1999] to maintain a feature index of virtual world objects. It runs similarity analysis on any reported or newly added objects (in linked asset servers) to detect similar contents. Associated meta data, visual similarity scores of matched objects, along with defined originality criteria will help determine to the case of IPR violations.

²⁰ <http://openlifegrid.com>

²¹ <http://osgrid.org/>

²² http://opensimulator.org/wiki/Grid_List

3.15 Mapping of plagiarism detection from text to the multimedia domain

3.15.1 Text Based Plagiarism Detection Systems

Theft of intellectual work is and always was a major issue in multiple disciplines including academia, arts, music, literature, computer code, graphics etc. However major work in order to find automated ways of plagiarism detection is done only in text domain. Most of the research and tools available as desktop software and web based services only target the text contents [Maurer et al., 2006][PlagiarismToday, 2008][Chester, 2001]. In order to have a better understanding of how these systems work, we will describe the general workflow of these systems. A Textual plagiarism detection approach involves creation of a feature index of published articles. The documents selected for indexing can have local scope where only a local archive is used for similarity detection, or the index can have global scope (along with local) spanning to documents available on public internet and remote protected archives. Articles selected for plagiarism test or newly created work is forwarded to the plagiarism detection service. The system processes it for feature extraction and generates the feature vectors. The generated feature vectors are compared with the feature database (index), in case of no match (or within certain threshold limit) the article is considered as original work. The features extraction from documents includes plain text extraction, tokenization, stop word removal, syntactical stemming, use of ontologies for concept stemming etc. In text based plagiarism detection systems, the index of features is maintained in a format that can be efficiently compared for similarity. Usually inverted list based index files are used for standard text [Zobel and Moffat, 2006]. A search based discovery process is initiated that traverses through multiple indexes to detect similar contents [Maurer et al., 2006]. A report identifying possible matches is presented to the system's user. This report facilitates a quick review of possible cases of copyright infringements. It provides grounds to determine whether it is plagiarism or not.

Size and scope of document feature index, the level of abstraction added to feature space, and similarity or distance measure plays an important role in system's performance. Almost all these services ignore images in documents during the feature extraction process.

3.15.2 Finding plagiarism in multimedia contents

There are options to find similar images using conventional image search engines but that only relies on the associated annotations or the file name. There are very limited options available for artists, photographers, musicians, and video makers to discover plagiarism. Techniques such as digital watermarking [Wikipedia:DW, 2008] are used to add protection in multimedia contents, but there is no good platform available for detection at large scale. We suggest a similar approach of plagiarism detection for multimedia contents (starting with visual contents) as adapted in academia. In case of virtual worlds the commodities subject to plagiarism and theft have dominant visual characteristics compared to dominant

textual characteristics of academic documents. These visual characteristics can be used to build a feature space required of similarity comparisons. The text indexing and search platform integrated in text plagiarism systems is replaced (or in our case complemented) with content-based image retrieval (CBIR) [Wikipedia:CBIR, 2008] for storage of visual characteristics of objects. Before we further describe the developed system let us first discuss the available CBIR systems that can be used for the task at hand.

A CBIR system provides

- i. Representation of digital media: A uniform representation suited for search, comparison and storage. This is usually done by hashing the image using Fourier Transform, Hough Transform, Wavelet Transform, Gabor Transform, Canny edge detection algorithm and Hadamard transform etc. [Sonka et al., 2007], [Dunn and Higgins, 1995], [Canny, 1986], [Pratt et al., 1969]. Intrinsic characteristics are extracted through use of a number of these functions.
- ii. Storage space for extracted feature: The storage systems should be capable of handling multiple dimensional data representing a large number of features extracted from huge and heterogeneous object collections. Various systems (described later on) use conventional data base systems, signature files, or inverted file based storage.
- iii. Distance measure: Or measure of similarity, which establishes a relation among images in the index. Distance measure like Euclidean distance, Dice similarity, Jaccard distance etc. are applied to query feature vectors and index feature space to determine closeness of objects.
- iv. Sorting and filtration process according to similarity/distance measure. Filtered images that pass a certain similarity threshold value are presented.

There are a number of CBIR platform available [GIFT, 2008][imgSeek, 2008][LIRE, 2008], which can be tweaked for indexing virtual world object (Jpeg2000 images, Sculpties) files along with additional meta information. For our experimental setup we evaluated the following leading open source tools.

3.15.2.1 GIFT (the GNU Image-Finding Tool):

GIFT [GIFT, 2008] formerly known as VIPER, uses multiple features for image indexing. They include color histograms and color layout by using a palette of 166 colors, derived by quantizing HSV space into 18 hues, 3 saturations, 3 values and 4 grey levels. In case of black and white images the color absence reduces the feature space, in such cases level of grey can be increased for higher dimensional feature space. Color layouts are computed by recursively partitioning image into four segments, at four scales. Global and local block level texture features are generated using Gabor filters. Gift offers a wide range of possible features (~85,000), where on average an image is described using 1,000 or 2000 of these features. The feature space (image index) is stored using an inverted file approach inspired by text indexing [Squire et al., 1999][Müller et al., 2004]. GIFT supports a number of feature weighting algorithms *e.g.* classical IDF, separate normalization etc. Other

positive aspects of the system include the relevance feedback mechanism and provision of standardized MRML [MRML, 2008] interface.

3.15.2.2 isk-daemon:

isk-daemon [imgSeek, 2008] is an open source Python based image indexing library. It generates image signatures using multi resolution wavelet decomposition. It allows effective similarity comparison even among images with different resolutions. This approach is fast and computationally less expensive; it gives a very good image approximation with few coefficients. The latest release added the support for indexing video scenes. isk-daemon is capable of running as clustered application allowing greater scalability and performance.

3.15.2.3 LIRE (Lucene Image REtrieval):

It also uses inverted file approach to store image features. The feature space in LIRE [LIRE, 2008] is based on the MPEG-7 [Wikipedia:MPEG-7, 2008] standard. The available features include color histograms in RGB and HSV color space, scalable color, color layout, edge histogram, texture features [Lux and Chatzichristofis, 2008]. The Java based LIRE library allows indexing based on fast, default and extensive functions with a possibility of feature space selection. The searcher function allows the specification of the size of result set and adjustment weights for color histogram, color distribution and texture. Being an extension of Lucene [Lucene, 2008], we get access to rich development resources for extension in feature space, index management and maintenance tools.

Because of the flexible architecture and more useful feature space we decided to use LIRE in our experiment setup.

3.16 Collection of test corpus

Due to the design of Second Life and various security issues it is not possible to directly access images that are stored on the asset server. Users can not even download their uploaded images outside the viewer if they are the creators of the files. Only simulators can connect to the asset server, fetch all necessary images, and forward them to the viewer where they are displayed. The connection between the simulator and the viewer relies on an open source protocol which can be easily monitored to analyze the transmitted content. To do so we can employ libopenmetaverse developed by the Open Metaverse Foundation²³. Basically, it is a Second Life client library that provides basic methods for the interaction with the Second Life grid.

One possible application of the openmetaverse library is a simple command line client to connect to any grid that is based on the Second Life communication protocol. Compared to the official Second Life viewer we do not have any graphical representation of the virtual world. Although, we just have a textual representation

²³ <http://openmetaversefoundation.com/>

of the virtual world we can do basic interaction with the server. Users can upload images to their inventory or list the inventory and it's associated identifiers on the asset server.

The image uploading process is quite simple. One can just log into a simulator grid, execute the `imageupload` command, and specify an image file on the hard disk. The `openmetaverse` library will upload the image to the server, respectively to the asset server. To verify the upload process one can execute `listinventory` and check the added item.

Although the official client software does not support the download of images to the hard disk, it is possible to fetch the images by the help of *libopenmetaverse*. We have already mentioned that the server sends the data, *i.e.* the images, to the client via public known protocol. Hence, we can counterfeit the official graphic client and send an image request to the server. The request consists of user information, a session identifier, and the *UUID* of the requested image. According to this request the server replies with the image data. Due to the open transfer protocol we monitor this data, assemble the packets and save the received image to the hard disk. Regardless of any permission of the requested textures we are able to download them just by using their unique identifiers *UUIDs*.

3.16.1 Object Crawler

Second Life and other virtual worlds suffer from the problem of content protection. We have already discussed that it is not very difficult to download images and textures from the Second Life asset server. One can employ so called copybots that move around in the virtual world and just download any textures they find. In the following we describe the operational details of a copybot we deployed. It provides the proof of concept for content plagiarism in virtual worlds and benchmarking of an automated image crawler which is integral part of described plagiarism detection system.

We have already described how to download images from the simulator just by a unique identifier. To get these *UUIDs* we can again employ the `openmetaverse` library that provides limited access to the avatars in the current simulator. Besides the name and the current position of the avatar, we are able to fetch information about the clothing they wear. This information basically consists of the *UUIDs* for the textures used for this clothing. We can use this information to send requests to the server and save the clothing textures of the near avatars on the hard disk. To improve the performance of the crawler we log into a popular region within the Second Life virtual world and iterate over all avatars to fetch their clothes. After that, one can move to the next region and get the avatars textures again.

Due to the periodically switching of the region and the corresponding traffic we have a large overhead in the download process. So we are only able to get about 40 MB of pictures per hour. The pictures are stored in Jpeg2000 format with an

average size of about 100kB. This yields in an average number of about 400 received pictures per hour and copybot.

Besides the image data crawler we also store meta information about the avatar and the object in a database. For each processed avatar we determine the current location within the current simulator, the current time, and the texture *UUID*. For privacy reasons we make all the received data anonymous. This prevents from the tracking of specific avatars and a link between an avatar's name and the resort to a place. Further, we are not able to determine if a specific item is illegally copied or stolen. Even the avatar itself can only determine the creator of the piece of clothing but does not have any information about the used textures. Linden Labs only provides the creator information if the actual texture is in the avatars inventory.

For optimized data collection we tested two different scenarios. In the first scenario we download textures and simultaneously add the context information to the database. If we detect an already downloaded texture we just add the given context information to the database. This implies that we add more items to the database than we actually download. The difference between the downloaded image and the added database entries is the number of texture duplicates, *i.e.* two or more avatars wear the same textures. The performance of this task mainly depends on the image download process. Due to the 400 downloaded pictures per hour we add about 450 items per hour to the database.

In the other scenario we do not download the detected images but just add the context information to the database. We do not suffer from the texture download bottleneck anymore and can theoretically add about 1500 texture context items per hour. On average we download approximately 4.5 items from each avatar. Therefore, we would need 330 different avatars per hour to get all the 1500 texture information items. If one considers only regions with a high avatar density of more than 20 avatars we have to switch the region 17 times per hour. In this scenario the performance mainly suffers from the overhead in the numerous region switch events.

3.16.2 Test Corpus

The information in the created database consists of about 11950 texture entries collected from about 1800 avatars. From these 11950 textures we found nearly 10% that are shared by at least two avatars. Figure 1 shows the distribution of these items based on *UUID* information. The main peak in the diagram indicates that most of the items are shared between two avatars. These 420 items are equivalent to over 5% of all registered items. For the rest of the diagram we have about 360 items that are shared between three and 14 avatars. The remaining 60 items are shared between up to 120 avatars. These items are basically the predefined items from Linden Labs or other public available textures under a common license.

Distribution of Shared Textures

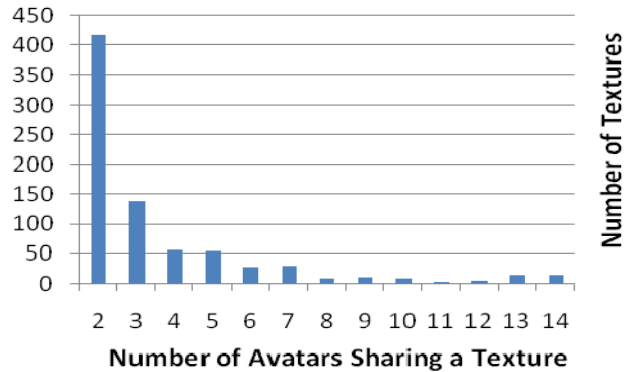


Figure 33. Texture distribution in test corpus

The result shown in Figure 33 is unusual for us; we expected a more balanced distribution. Basically, there are two reasons for this significant amount of items shared between two avatars. Avatars can easily tick a box in the properties of cloths that marks them as disposable. They can specify a price and sell the item to any interested avatar. These offers are not promoted by a shop and so the group of buyers is very limited. The other reason is theft of the texture and therefore an unauthorized distribution of copied items with different *UUID*. This can be done again by intercepting the communication between the server and the client. Then it is possible to create clothes and other items with a specific texture just by the knowledge of the texture's *UUID*. Section 3.18 shows similar object distributions having different *UUIDs* in test corpus, detected through the proposed framework.

3.17 System for finding plagiarism in visual objects

The system to detect similarity among virtual world objects and determine originality is composed of following modules

1. A Crawler that collects the objects and meta information (with direct access to asset server, or in our test case the information is collected from virtual world itself)
2. The feature indexing and comparison engine; that transforms and stores the characteristics of objects.
3. An originality evaluation module that applies the predefined originality assessment rules to rank similar objects.
4. A user interface to browse the object index. It allows users to input queries for object originality checks and presents the assessment reports.

The operational and technical details of crawler are already discussed in previous section. However we mainly described its function in development phase where we do not have the direct access to an asset repository. In principal a legitimate crawler will have access to objects repositories of connected simulators. It will collect the desired information directly from asset servers. The second module is a CBIR engine with capabilities of various image digest generation functions, index read/write/append and linear search capability. Table 4 shows few statistics of Lucene image index generated through LIRE API.

Generated using LIRE 0.7 API

<i>LIRE Index Type</i>	Extensive
<i>No. of Objects</i>	11950
<i>Available fields</i>	Image Identifier ColorLayout EdgeHistogram ScalableColor
<i>Total Index Size</i>	9161 KB

Table 4. Object feature DB

This platform provides an index of registered items which are periodically checked with items available in asset servers. Owners of registered objects are notified about matching contents above a certain similarity threshold, with content originality rating. Probably after a deeper manual investigation owners can choose to file a DMCA notification for removal of found plagiarized item. Figure 34 below gives an overview of system architecture and interaction among different system modules.

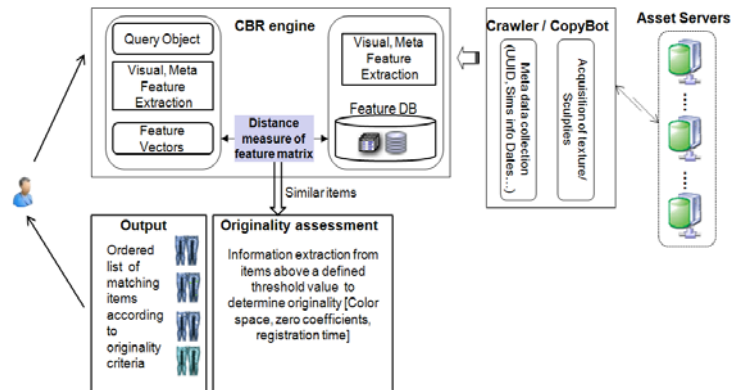


Figure 34. Architecture of plagiarism detection system

Along with an inverted file based image feature DB, we are also maintaining a meta information DB in a MySQL database. The CBR engine provides the list of visually similar document to originality assessment module, this module also fetches the meta information from MySQL database, associated with given similar images.

The general purpose rule for originality assessment is perhaps the use of the creation date. However, in case of non availability of existence information, tempered creation data or to further strengthen the originality assessment we need secondary measures of originality assessment. Since we can not base our assessment on some reference object (all equal while comparing for originality) we have to rely on blind image quality assessment. In this particular environment the textures are stored on asset servers using Jpeg2000 compression.

The typical theft involves the download of these compressed images. After certain modification (optional) upload the image for usage. The open source client Hippo does not allow upload of same Jpeg2000 format file so there is a requirement of image type conversion. During the upload and download there are certain Jpeg2000 compression losses we observed at client and server side. Repeated Jpeg2000 compression adds the blurring and ringing effects to images. These effects can be determined by a number of quantifiable image quality parameters described by [Sheikh et al., 2002][Marziliano et al., 2002][Barland and Saadane, 2005]. They are used to deduce the originality of similar images. However there are options to upload the exact same image into one's personal inventory in Jpeg2000 format without any loss of quality. In such cases we will get very high similarity results. In such case the meta information and visual inspection along with similarity scores are used to determine the theft process.

The last component of the systems is web based user interface that allows uploading a sample image to feature database for inspection, system user can also browse through the available image inventory for similarity comparisons. The output to user comes in form of an originality report with ranked listing of similar images, according to originality criteria *i.e.* oldest created items, blurring artifact measure (attenuation of high spatial frequency coefficient, quantization of DCT/Wavelet coefficient values at finer scales) [Sheikh et al., 2002][Marziliano et al., 2002][Barland and Saadane, 2005] .

3.18 Results and system enhancements

The shared texture distribution discussed in section III is for the objects sharing a same *UUID*. In such cases probably the common objects is legitimately sold or distributed. However the cases where items are visually similar and have different *UUID* value, the changes of theft are high. In order to determine such instances we ran a similarity analysis on objects in our index. Items having similarity of 40% or more are recorded. The results from this analysis are given in Figure 35.

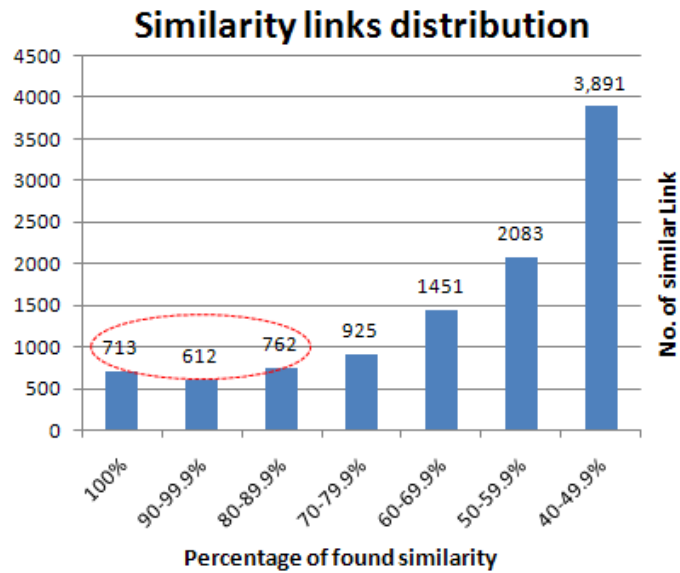


Figure 35. Distribution of similar images with different UUIDs

This analysis shows that there are a great number of objects that have very high visual similarities with different *UUID* values (circled area in picture). “17.5%” of suspected plagiarism cases in a randomly collected test corpus are alarmingly high.

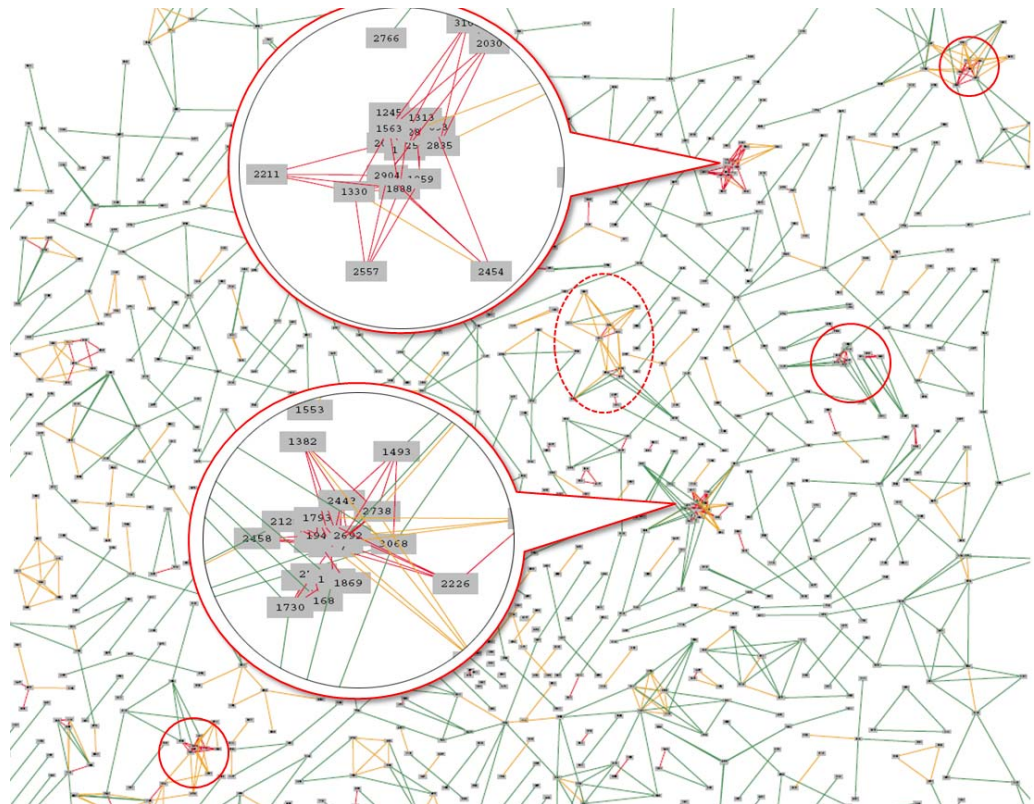


Figure 36. Identification of possible theft cases in a segment of objects

For having a bird eye view of possible meshes of similar object, we plotted the cached similarity links. Node links in graph of figure 36 are inversely related to visual match. We were able to easily identify the possible cases of mass copies among the distribution.

The crawling not only highlighted the technical details of content theft it also gave us a reasonable test corpus (11950 images) to evaluate our proposed system. Initial similarity analysis by the proposed system helped us get a rough idea about suspected plagiarism cases in the corpus.

In order to explain the work flow of defined system we take a typical example, where system user is uploading a new item to server. The asset server passes the new item to the CBR engine through crawler. The visual and meta information of added item is stored and compared for similarities with existing DB. After computing the originality parameters a report is presented to user. As a test case we copied a clothing item and uploaded with different *UUID*, produced originality assessment is presented in Figure 37.

Similar Objects and Originality Assessment






Query Item		Originality Information:
Identifier = fe443877-3ef3-a402-e8dd-6c9480b2d01c Sim. Score = 1.0 Creation Date = Not available	<input checked="" type="checkbox"/> 	Item "d58072b8-7edd-e69f-1824-7e7792a1a1e" has better quality then item "fe443877-3ef3-a402-e8dd-6c9480b2d01c" Select other items for calculation and press originality comparison button
Identifier = d58072b8-7edd-e69f-1824-7e7792a1a1e Sim. Score = 0.912733 Creation Date = Not available	<input checked="" type="checkbox"/> 	
Identifier = fa2fbb2b-0ef9-bcc9-baf6-4d01b071d883 Sim. Score = 0.6779046 Creation Date = Not available	<input type="checkbox"/> 	
Identifier = a2b44236-5796-8d7f-fccf-345807577d1b9 Sim. Score = 0.20690435 Creation Date = Not available	<input type="checkbox"/> 	
Identifier = 3c84ed02-7f37-e42e-8de1-90874013762f Sim. Score = 0.19703233 Creation Date = Not available	<input type="checkbox"/> 	

Figure 37. Similarity and originality computation

The system user is provided with visually similar items with available meta information. User may select a number of suspected items for further computation of originality. The originality assessment module in the developed prototype relies on the changes in DWT (Discrete Wavelet Transform) coefficients of the image.

Figure 38 shows the coefficient probability of the Discrete Wavelet Transformation (DWT) of an original and a copied image. Both images have most of the coefficients near zero which implies that lower frequencies are dominating. Figure 39 depicts the detailed probability for coefficients between -50 and 50. The lossy Jpeg2000 compression algorithm suppresses higher coefficients and amplifies lower coefficients. Due to these additional lower frequencies the copied image is more blurred than the original one.

This is rather a simplistic approach adapted (considering blur effects only) in the prototype and will be replaced by more complex quality assessment models already referenced.

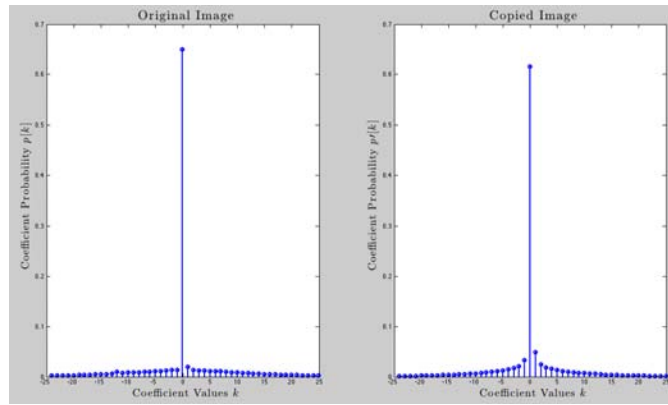


Figure 38. DWT coefficient probability distribution of original and copied image

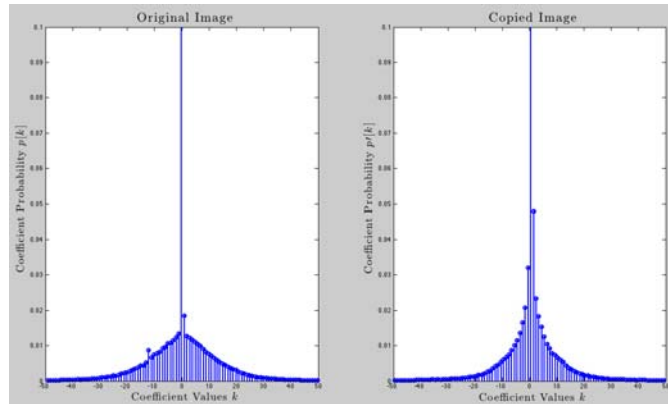


Figure 39. Detailed probability distribution of wavelet coefficients

Our future road map also includes enhancements in CBR engine's feature space. In addition to extraction and indexing of simple visual contents the enhanced feature

indexer will include descriptors for sculptures and scripts. This in turn will result in a more domain specific feature space covering most of the intrinsic qualities of all virtual world objects.

Virtual worlds are mainly made up of user contributed digital contents. There are a number of users who are trying to earn through selling their work in virtual worlds. Our work shows that digital, open, and somewhat unprotected nature of underlying system makes it is very easy to steal other people's work. Existing copyright enforcement systems and policies are not enough to provide protection against these increasing IPR violations. There is a great need for a system that provides more convenient discovery, detection, and most of all some level of deterrence against the theft of contents. The system described here aims to achieve this goal, its implementation at grid management level or even community level may provide an extended level of protection against plagiarism in virtual worlds.

3.19 Further work

The experimental work described in this chapter addresses the initial three problems identified in earlier part of chapter (section 3.8). Based on our further learning through development of CPDNet platform, following issues need to be addressed.

3.19.1 Introduction of translation and normalized signature search service

This issue is already identified in section 3.8 after reviewing the current services. In order to check plagiarism across language barrier, another service at a different abstraction layer is required. This must translate the normalized indexes and queries into standard English. Each node can provide translation into and from a specific language depending on the local resources. This service will complement the normalization on a more global and conceptual level. Such abstraction may produce undesired and false results at some levels. However it is worth experimenting with the cross language similarity checks, because of the large availability of intellectual contents in non-English languages.

3.19.2 Addition of intrinsic characteristic checks

This deficiency is also identified earlier. The statistical text structure comparison, and authorship test already discussed shows the usability of stylometric analysis. Generation and use of a comprehensive and generic text feature matrix can help introduce blind plagiarism detection capabilities to system. Chapter 5 contains related work on generation of write prints and their use in content organization and filtering process.

3.19.3 Noise reduction in plagiarism detection with domain specific searches and efficient citation checks

Similarity detection on an abstract level may introduce unnecessary noise in generated matches. It would be helpful to restrict the semantic level search and analysis to a domain specific index. Subject specific searching capability will be introduced by means of ...

1. Setting up specialized indexes of certain domains. The participating institute's local index can be categorized based on various departments or research groups.
2. Using topic maps to categorize subject domains and grammar to link contextual queries.
3. Introducing a service before performing search that determines the context of the document being analyzed. One such example is the use of Yahoo Term Extraction service. This service provides its users the context aware relevance technology behind Y!Q [Yahoo:TermExtraction, 07]

Another level of service is required to decrease false positives while detecting plagiarism. Some plagiarism detection services give their users the option of ignoring texts found within quotation. This approach however is not sufficient in determining proper citations. There is a need to automatically compare the referenced articles and remove any plagiarism score coming from these sources. Such automation can be achieved by scanning through the referenced articles and creating an index of referenced resources in a document. The user can then choose to ignore the similarity matches which are available in the reference index. The automated detection of proper attribution or citation in a document will save the examiner both time and effort. Developing such a reference index may require a persistent and universally accessible resource identifier associated with the citation. The increasing web publication trend and the emergence of common linking and resource identification standards like DOI [Warren, 2005] are encouraging factors which will lead to further investigations in this area.

3.19.4 Scalability and design issues of composite applications

The core architecture suggested and used for enhancements in plagiarism detection applications is based on the mashup of web search services and similarity analysis services. The application areas considered in dissertation are also based on service oriented architecture. Developing large scale web service platform and integration of these services requires efficient workflow management. Web service workflow system complements the service discovery, description and messaging capabilities. For testing and prototype development we relied on a manual choreographic approach to integrate services. However more generic composite application model may require more controlling service orchestration approach. The scalability issues

and service composition in distributed web applications is further discussed in chapter 4.

3.20 Conclusion

It is fair to say, that current plagiarism detection tools work reasonably well only on textual information that is available on the internet or in other electronic sources. They do break down:

1. When systematic attempts are made to combat plagiarism tools by e.g. using extensive paraphrasing with the help of synonymising tools, syntactic variations or different expressions for same contents. (NOTE: most of the better systems are stable against the order in which paragraphs are arranged: fingerprinting is usually not done on a sequence but on a set of data, hence order does not matter)
2. When plagiarism is based on documents that are not available electronically (Since they only are available in printed form, or in archives that are not accessible for the tool used)
3. When plagiarism crosses language boundaries.
4. When plagiarism is done using non text contents (images).

Of the four points mentioned above there is good hope concerning item (2): more and more material is being digitized, and some tools have managed to get access to hidden material in paper mills and such. Based on CPDNet experimental results, it can be safely stated that the platform presented addresses the second issue effectively. This is due to the additional support of internet searching API mashup and the collaborative indexing approach. Item (1) (3) will be a challenge for some time to come. However, the availability of various analysis services, such as vector space similarity, structural or stylistic evaluation (more details in chapter 5) of suspicious documents and fingerprint normalization in the CPDNet system is a promising attempt to handle issue 1 and 3. The presented approach to handle issue 4 is tested in non academic domain; it successfully demonstrated the use of CBIR technology to detect theft of images. Document parsers that extract images can complement document indexing with image feature hash. This adds the capability of finding similar images in documents and determining originality based on quality factors.

Collaborative web service oriented architecture substantially extends current plagiarism detection systems. With flexible and extendable services, rich web user interface, standardized XML based inter application communication and collaborative authoring, it brings us a step closer towards Web 2.0 applications. The technology industry has a rapidly growing interest in web services. Many

companies and service providers already have web service components available with their applications. Almost every software and internet organization focuses on web services as a core element in future strategies. This tendency suggests that the proposed web services enabled platform is best suited to carry out multiphase plagiarism detection. It will offer the flexibility to incorporate any new processing, discovery or indexing components that may become available to its users. The user-centered collaborative nature of this system makes it an ideal choice to build specialized indexes which are capable of handling semantic considerations in the similarity detection process. Text normalization adds a semantic level of detection capabilities in plagiarism applications. Later part of our work in chapter 5 (our experiments on stylometric analysis with JUCs documents) shows the importance of intrinsic characteristic checks.

In closing we want to mention two further important points:

First, plagiarism is not confined to academia. It is rampant and still not much recognized in schools, particularly in high schools where many assignments are of the general essay type, exactly the kind of stuff easily found on the internet. It also appears in a different form when government agencies or other organizations commission some ‘study’ or report to be compiled: in a number of cases they get what they want, pay quite some money for it, but what they get is just obtained by simply copying and pasting and minor changes or additions of existing material. In those cases it is not so much a question to detect plagiarism after the fact, but rather have some specialists spend a few hours searching on the net if the material requested is not available anyway before commissioning a report.

Second, plagiarism is getting lots of attention in academia right now. The reaction has been that many universities purchase tools for plagiarism detection. It is our belief that to detect plagiarism at a university you need more than a software tool: you need a set of them, specialists who know how to work with those tools, domain experts and also language experts if we ever want to go beyond the boundary of one language. This implies that a substantial group is necessary to do good work, and this cannot be achieved by any one university. It requires a joint effort i.e. a center for plagiarism detection that is run on a national or even supra-national (e.g. European) level.

4. Adaptive Information Systems

Applications of similarity detection in personalized content delivery and user profiling

Contents of this chapter are taken from

“Personalized Interactive Newscast (PINC): Towards a Multimodal Interface for Personalized News” [Zaka et al., 2007]

“Use of similarity detection techniques for adaptive news content delivery and user profiling” [Zaka et al., 2009a]

“Service Oriented information Supply Model for Knowledge workers” [Zaka and Maurer, 2007]

Staying informed is one of the key factors to success in business and technology. Accessing concurrent information is a key to interpret current events as well as to build up knowledge about long-term developments. These days it is no longer a problem to access information, but to identify important information in the vast amount of available contents. Finding and filtering relevant information according to personal preferences is a time-consuming task in the daily effort to stay well-informed. This chapter describes two experiments undertaken to study extensions in information supply environments. The first part presents a system that uses linguistically enhanced similarity detection technique to tailor the syndicated information to the individual’s need. It helps create standardized user interest profiles that can be reused in number of information retrieval applications. The second part introduces information provision environment for knowledge workers. While they work on a problem system in the background is continuously checking to determine if similar or helpful material has not been published before, elsewhere. The technique described aims to reduce effort and time required to search relevant data on the World Wide Web by moving from a “pull” paradigm, where the user has to become active, to a “push” paradigm, where the user is notified if something relevant is found. The approach facilitates work by providing context aware passive web search, result analysis, extraction and organization of information according to the tasks at hand.

4.1 User adaptive news content delivery

The increased diffusion of communication technologies and their applications made our lives very information intensive. Exploring, organizing and preserving this information space are complex tasks and varies with type of information and its medium of delivery. A huge volume of information is available to individuals in form of daily news. The sources for this type of information range from conventional print media such as newspapers, radio, television to more recently developed ways of getting personal and general news via web portals, emails,

content syndication, digital media streams, pod-casts and many more. With this variety of sources at hand it is becoming difficult and time consuming to get the desired information, based on the reader's interest and preferences. The user has to spend a reasonable amount of time and effort to filter the desired information from all these sources, especially since different source are preferable for different types of content. User profiles and preferences that form the basis of adaptive information systems are generally system specific. Profiling techniques used in common information retrieval system give very less or no consideration to user ownership, portability and reuse of user interest profiles. This is frustrating for users when they have to duplicate filtration effort at various sources. A research study demonstrates significant negative relationships between information overload and stress, decision making, job fulfillment [Klausegger et al., 2007]. Such an abundance of information affects the natural cognitive capabilities of individuals. According to a research firm Basex1 who predict information overload as the biggest problem of the year 2008, information overload has serious effects on productivity of individuals and can cause loss of billions of dollars for large organizations. These factors make adaptive reception of information very critical in order to fight information overload.

With varying environmental and physical conditions it is not always desirable or possible to efficiently interact with a number of information systems individually. In this situation it is preferable to access one central system that provides aggregated access to various sources. In order to provide an effective and suitable way of accessing the system, the interaction has to be adapted in modality and media to contextual requirements. Providing multimodal interaction [Oviatt et al., 2000] is necessary, as the application of the personal computer-based paradigms is not always possible in the conditions described above. In many situations telephone or PDA are more readily available than a PC or laptop computer.

Another hurdle in successfully and conveniently navigating through the diverse information base is the constraints posed by interface modality. Spread of wireless data networks and emerging handheld devices offer a number of new ways to access information systems. Many information systems already provide specialized layouts and communication interface for unconventional devices. However in most cases such interfaces are more of a hindrance than a convenience. The design of these unconventional device interfaces compared to conventional desktop devices is still relatively unexplored. The development of revolutionary technologies such as smart phones, digital media players, digital interactive TV and E-ink devices marks the evolution from the current desktop computing era to ubiquitous computing. This results in the change of concepts for device interaction and urges researchers to increase the work on new, multimodal systems [Larson et al., 2003]. Such systems in turn will extend the information paradigm of the computer-based information systems and Internet to these more common platforms. In conventional user interfaces, interaction with system for precise information retrieval is a lot closer to machine perception of user requirements; input via keyboards/GUIs is interpreted with a higher level of certainty than in multimodal systems where the system's

interpretations are probabilistic [Oviatt and Cohen, 2000]. Even then, in case of conventional interfaces there are many users who have limited knowledge of all the available information retrieval and filtering techniques (e.g. limitations of vocabulary, awareness of advance search operators). Precision in information retrieval gets more challenging in case of un-conventional modes of interactions. Thus, it is very important to provide information filtering and retrieval means which are based upon user's spontaneous interaction context and a defined history of interest. Furthermore it is beneficial to add semantic meanings in multimodal interactions in order to reduce uncertainty and increase efficiency of communication.

One approach for such a system, with focus on the individualized delivery of news items and multiple user interface modes, is presented here. The suggested framework uses conceptual similarity detection techniques for personalized news delivery. It offers user controlled, standardized and portable user interest profiling system. The ongoing user profiling, based on implicit and explicit feedback as well as group preferences, is used to create personal information filters. With a standardized profiling system it is possible to use personal interest data in a number of existing and upcoming information retrieval applications. PINC system offers a context-aware news item relevance system. It uses term extraction and synonym set services to link content items and user filters. This approach, based on conceptual semantics, lexical relation and service-oriented architecture allows increased efficiency of the information filtering system. Proposed system also offers an enhance recommender system. Conventional recommender systems use contents based matching, collaborative filtering or knowledge based techniques. A survey and experiments on recommender systems show that more successful systems are those using a combination of these techniques [Burke, 2002]. Our system takes advantage of semantic knowledgebase and collaborative filtering for its hybrid recommendation capability. The system is also capable of preparing filtered news items as a seamless information source that supports cross-media publication. Multi-channel distribution ensures the availability of news contents in different mediums with varying physical and environmental conditions.

4.2 Related Work

The proposed framework addresses news harvesting, metadata extraction, context determination, and filtration for the creation of a personalized newscast. It also deals with cross-media publishing and multimodal interaction for its access. All the mentioned areas have attracted interest lately and considerable research has been published on these individual topics.

Focusing on personalization and filtering functions first, several systems addressing these topics deserve mentioning. Such systems include “SELECT” [Scheidl et al., 1999], one of the early efforts to reduce information overload. It introduces the information environment tailoring to meet individual needs with the help of

information filters. These filters provide recommendations derived from an individual's past choices and behavior of other users with similar interest. SELECT emphasis on social and collaborative filters and importance of a strong rating and feedback mechanism to support filtering of mentioned types. This project also explores the use of implicit as well as explicit feedback techniques to enhance the rating database.

A more recent, ontology-driven user profiling approach is the “Quickstep and Foxtrot” system [Middleton et al., 2004] which has introduced hybrid content-based and collaborative recommendation techniques with effectiveness of presenting user profiles in ontological terms. Another project, “News” [Fernández et al., 2005], also utilizes semantic technologies to extend personalized delivery capabilities of online news contents. This system provides an RDF based news ontology for news item categorization. It also provides annotation components to automatically produce metadata for news items. Social networking sites, blog aggregators that use folksonomies (user tagging of information they generate or consume) in addition to taxonomies are becoming popular. Most of us have seen the effectiveness of user collaborative recommender systems while browsing Amazon portal², where a recommender system presents items under the labels: Customers who bought this item also bought, Customers interested in this title may also be interested in, what do customers ultimately buy after viewing items like this? In general we see that there is a tremendous increase in availability of syndicated contents and in turn aggregation tools for personalized view. A survey conducted to compare existing news aggregation services in terms of their features and usability, reveals that the most desirable features by users are the advance search functionalities, user friendly interface, quality of sources, browsing and personalization functionalities [Chowdhury and Landoni, 2006].

There are number of experiments and studies that highlight improvements in personalized information access through effective user modeling [Billsus and Pazzani, 2007] [Teevan et al., 2005] [Kan et al., 2006]. These approaches include profiling based on user provided explicit data or implicitly gathered information through analysis of interest and activities. Research suggests that automatic capture of user preferences is necessary especially in case of heterogeneous contents and changing interest of the user. Systems offering personalized contents are an appealing alternative to “one size fits all” approach. This personalization approach is perhaps the major factors in success of online e-commerce company Amazon.com. This portal is well known for its personalized service which starts offering custom store views even after few mouse clicks and covers a detailed user view and purchase history.

The second focus of the proposed news delivery system is on multimodal interfaces. Although multimodal interfaces are designed with a focus on flexibility and extending usability, only few of them are capable of adapting to different user preferences, tasks, or contexts [Xiao et al., 2003]. The same applies to content adaptation in a multimodal approach.

The main problem of the existing solutions is the coverage of only a part of the requirements of the modern user of news systems. Personalization and filtering approaches lack the possibility of being ubiquitously accessible. In Personalization knowledge about the individual user is used and contents are adapted according to the user's needs. The collection of this knowledge is an on-going process that depends on how well user actions are interpreted from various modalities. The effective interpretation of these actions and conversion into a knowledge base that forms the user models remains a challenging task in multimodal systems. Moreover many of these approaches do not take into account the particular context of the news domain. This problem can be effectively addressed by using semantic relationships between input from interaction devices and the collection of entities in a system.

An effective system must be able to aggregate semantically equivalent news contents from different sources and present these collectively, arranged and filtered by user and group preferences. Multimodal and cross-media publishing systems can be used to access news content, but generally they lack the support for association by semantic or collaborative equivalence as described above. The key to adaptive content reception and recommender systems remains automated discovery of personal interest, preferences, environmental and social characteristics. Adaptive systems tend to gather as much information as they can and store it for personalized interaction with user. Normally a typical user is not aware of what and how much personal information is stored in an adaptive system. This raises a lot of privacy concerns [Riedl, 2001]. One way of addressing the issue of privacy is providing the user more control over how the information is stored and processed in a standardized way.

4.3 Design of Personalized Interactive News Cast

PINC aims to enhance the end-user's access to news in a way the previously presented approaches cannot. It provides a solid solution for news harvesting, personalization and presentation.

4.3.1 News acquisition and pre-processing

The aggregated news contents of the newscast include news articles acquired from various syndication services and web mining. The news contents are collected, processed and indexed on the server side. The intervals for this acquisition process are set by a system user. Information fetching agents responsible for the collection of the news contents are easily modifiable and extensible. The plug-in based crawling agents traverse through the selected sources periodically for collection of updated information. Fetched news contents are relayed to the information pre-processing unit, where extraction of metadata and categorization is done. This component stores the news entities in the main information repository and builds the information resource knowledge base by extracting meta-information from the

fetches content. This extracted meta-information normally includes: source, publishing date/time, type of media, author, keywords and description.

Fetches contents and meta-information are normalized to a generalized language form before the creation of an index. This process of normalization is conducted by using natural language processing techniques of POS tagging, term extraction, and finding most common form of each word/term. Part of Speech tagging is used to determine the correct syntactical sense of words (verb, noun, adjective etc.). This syntactical sense is later used to determine the respective group of synonyms. The synonym groups are selected using WordNet lexical data. The most common word in a selected synonym set based on its frequency reference in language ontology (tag_count parameter of WordNet) is picked as normalized representation of a particular word/term. Information normalized in described way, when compared for similarity, provides a greater depth of concept matching. Figure 40 depicts the process of normalization. System architecture section (4.4) further describes the process through example.

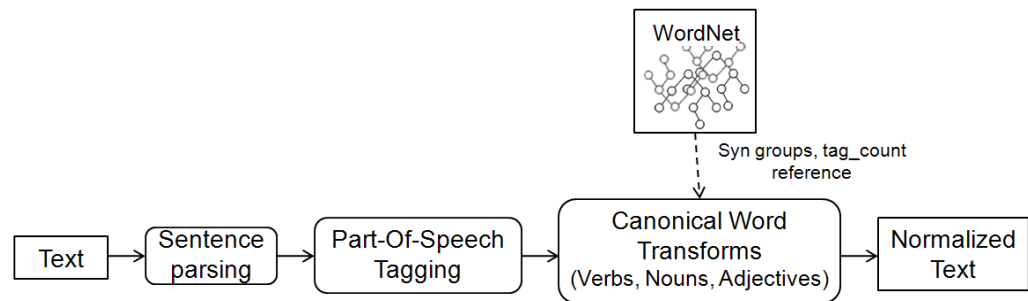


Figure 40. Normalization of text to find conceptual similarities

Further meta-descriptors are generated by applying term extraction on fetched contents. A term designates certain meaning/concept to any information. Different linguistic and statistical techniques for term extraction are in use. They determine importance of words by consistency, frequency, structural location, linguistic morphology. Already available news category information and generated meta descriptors of fetched news entities are compared for similarity with the system's news category taxonomy. This allows the system to automatically categorize the news entities in a given taxonomy even when there is little or no classification information is available. In addition the described approach provides an automated way of using data mining techniques to convert a basic news taxonomy into a rich news ontology. Use of various news sources captures the view of many domain experts, thereby making our news ontology more effective [Parekh et al., 2004]. It works as a rule-based categorization agent, linking news entities and metadata to individual elements of the seed news taxonomy. Similarity detection is used to determine the news item category. Angular measure based on vector space model determines the relevancy between the news item's meta-information and the news category keywords. This enables the system to go through an iterative process of evaluation, enrichment and refinement of the news category descriptors. The system

maintains the inverted file index of the normalized contents. Such storage outperforms conventional database systems in terms of faster search, and lesser storage requirements. A combination of Boolean and vectors space based retrieval models are used to determine relevance between filter queries (based on user models) and indexed news data. Figure 41 gives an overview of information pre-processing and indexing.

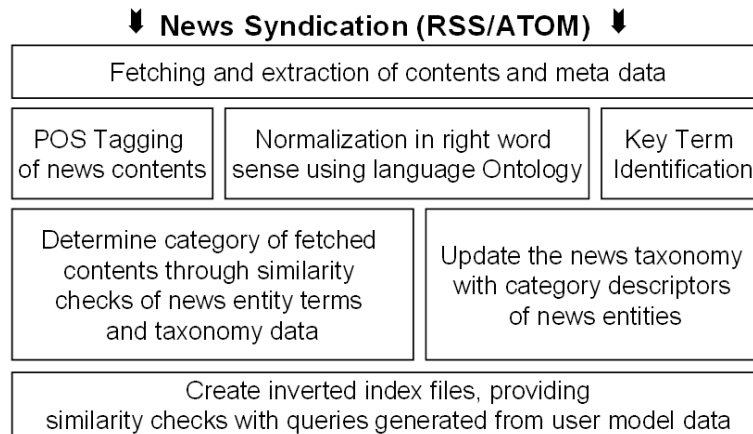


Figure 41. Information pre-processing

4.3.2 Portable User Modeling

PINC uses the idea of wrapping heterogeneous data sources into a uniform knowledge representation, with semantic annotation. This offers integrated and personalized view of data [Abel et al., 2005]. News contents can be categorized and characterized using the additional semantic information. The process of annotation is done by using term extraction techniques and enrichment of terms (concept defining words) with lexical variations. The process of adding greater depth of associated terms to news entities and creating concept vectors is exemplified in section 4.4. A well defined user model structure is the key to the creation of personal views of news entities. A user model is initiated by integrating explicitly stated user preferences in profile. These preferences may include demographic data, user knowledge skills, capabilities, interests, selection of predefined categories. References and links among user models will be used to share knowledge about mutual interests in order to form groups and enhance the recommendations by collaborative filtering techniques.

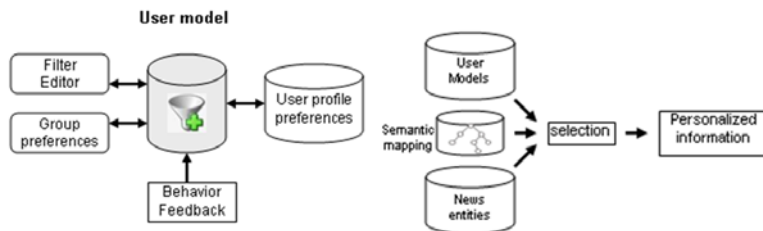


Figure 42. User Model and Personalization

The user model filters and group/social links are automatically updated based on usage data; this includes explicit tagging, user ratings and implicit behavior such as selective actions, use frequencies, hardware environment, location data etc. Figure 42 shows the visual representation of user model and process of personalization. A personalized view is created by finding conceptual equivalence between filters available in user model and normalized news entities processed in information pre-processing.

A growing concern in the context of personalization is privacy. In order to meet the requirement of users for control about their personal data we decided to integrate Attention Profiling Mark-up Language (APML). This is an XML based markup language for the description of the user's interests, designed to be shareable and controlled by the user himself. It is intended to improve the ability of information system to provide information fitting the user's need, reducing the information overload. As such APML is dedicated to four fundamental rights for the user. Firstly the profile is a property of the user, his attention is owned and controlled by him. Secondly the user has the right to move his attention wherever he wants whenever he wants. Thirdly the user's attention has worth. He can pay attention to whomever he wishes and receive value. Finally the user has a right for transparency, being able to see exactly how the attention is being used and based upon this decide who to trust. An APML file contains implicit attention, which is derived automatically from the behavior of the user, as well as explicit attention which is added by the user. For both categories concepts and sources can be specified, the latter being information sources like an URL, or an RSS feed. Each of these elements is assigned a value between 1 and -1, where high positive values indicate a lot of attention and negative value explicit dislike.

APML is already used in a number of services, most prominently Digg and Bloglines. Due to the fact that APML is designed to provide benefits for both advertisers as well as users it can be assumed that further services are likely to follow. PINC provides a tool to generate initial profiles from the users browsing habits. To that end the browser history is scanned, the visited pages are retrieved and analyzed. Subsequently terms are extracted. These terms are assigned with attention values between 0 and 1, based upon the term frequency. Negative values are ignored in this context. The resulting APML-file is provided to the user for editing and can finally be uploaded and incorporated into the personalization process of PINC, presenting an initial interest model.

4.3.3 Aggregation

The system acts as a universal news aggregator. It fetches the news contents; parses the contents for metadata enrichment and stores in a local repository. In the final aggregation to a newscast, the filtered and arranged news items are retrieved from the repository. The corresponding articles are dynamically fetched from the sources and, appropriate to the content type, either embedded or linked in a NewsML news envelope. NewsML is a standard by the International Press Telecommunication

Council to present news contents in text, images, audio or video using XML. The use of XML at various levels allows ease of data interchange and multimodal publishing.

NewsML is envisioned as a way of standardizing news aggregation for multimedia, multidiscipline and multimodal delivery. It provides an XML envelope to manage and represent news through its lifecycle. This lifecycle starts with definition of news story along with comprehensive representation of meta data such as domain, media, origin and history. The standard organization also facilitates ease of transformation for enhanced/multimodal user consumption (via xslt or by any other means). NewsML is being used by leading newspaper organizations and publishers.

The information aggregation component of PINC represents an imperative concept of web 2.0 applications called mashup. The term mashup is initially introduced by modern music community and used when vocals and music from different songs are mixed to produce something new. In technology, mashup refers to applications that combine contents from different sources and present them to users in a seamless manner. Mashups are rapidly spreading their roots and popular types include map mashups available through Google MAP API, Microsoft Virtual Earth API, Yahoo Maps API, shopping mashups like Geizhals, Pricegrabber, and photo mashup like Flickr. News sources such as Reuters, Associated Press, BBC, CNN, AFP, APA are using RSS feed to distribute contents for quite some time and various news mashup applications exist that use all these feeds to present users with a combined or selective view of contents. PINC's aggregation component forms a personalized and context-independent information dataset using content and collaborative filters. This filtration is based on semantic relations among user models and the meta-information (see Figure 31). News items are aggregated into a standardized NewsML structure which provides wealth of data interchange for multimodal publishing.

4.3.4 User Interfaces

Current personalized news information systems mainly focus on the presentation of the content via the personal computing paradigm. Technology trends show that in the coming years ubiquitous computing will replace the current personal computing era and change the ways of users' interaction. Conventional input/output devices will play a very small role, making ways for Perceptual User Interfaces (PUI) [Turk and Robertson, 2000], maximizing the use of natural human communication with digital devices and systems. PUIs demands capability of automatically extracting user's need by translating human interaction with system. In general the user input is perceived through the sophisticated analysis of body gestures, voice and navigation patterns.

In order to follow this direction and provide access to a personal newscast in almost all situations, PINC framework provides the end user with a choice of selecting the most appropriate mode for delivery of personalized news. A dynamic user model containing attention data provides the perception of user's information need in

multiple modes of interaction. The aggregated data in NewsML form is converted to a specific publishing format using an appropriate XSL transformation. The proposed initial interfaces include:

4.3.4.1 WWW Access

The PINC publishing module provides news and information contents for desktop or mobile device browsing via XHTML transformation. The transformation fitting to client specification is achieved through a combination of user-agent sensing and transparent content negotiation mechanism [Holtman and Mutz, 1998]. HTTP delivery module contains formatting scripts capable of sensing the user agent environment variable for browser, OS types and general display capabilities. Plain user agent based adaptive method relies on up-to-date knowledge base of all the available browsers and there capabilities, it fails to function in case of non availability of data about new clients. This problem is minimized by adding capability of mime based content negotiations between client and server (where supported). The “Accept” header information sent by client is used to determine appropriate content format for delivery. The properties of content negotiations sent in “Accept” header from client are Media Type (with quality parameter), Language, Encoding and Character set. The added client information help customize HTML news presentation for different browsers.

4.3.4.2 Speech Interface

The PINC framework includes a VoiceXML 2.1 browser, supported by compatible Text-to-speech (TTS) and Automated Speech Recognition (ASR) engines. VoiceXML (VXML), basically, is a way of defining voice dialogs which take input from the user in form of Dual Tone Multi Frequency (DTMF) signal or speech phrases and responds with pre-recorded voice or synthesized voice via TTS. This standard is considered to be the most accepted solution for voice web applications. Figure 43 shows the user interaction via HTTP and voice interface.

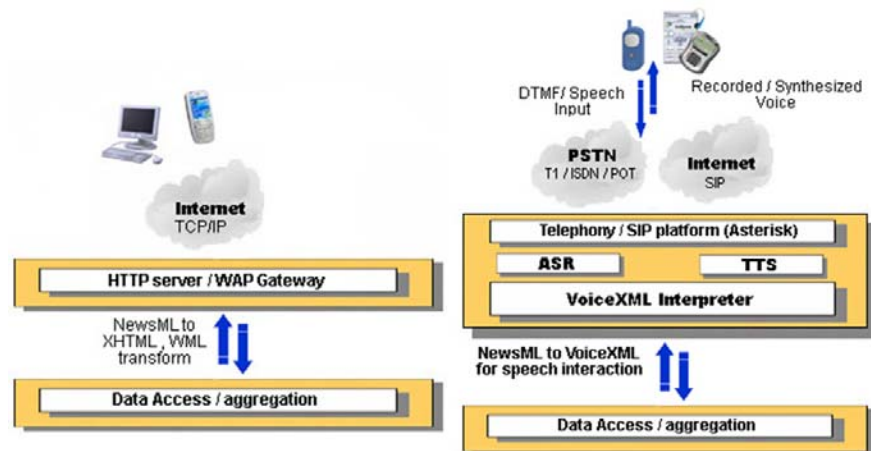


Figure 43. HTTP and speech access

VXML is an extension of XML and designed specifically to provide aural interfaces for web applications. NewsML is converted to VXML using appropriate XSLT transformations, and presented to the end user for voice browsing. The filtered and sorted news items are pushed to the user in form of interactive voice dialogues. The VXML feed is reorganized based on user browsing interest coming from simple voice commands and keystrokes (DTMF). Figure 44 shows a sample VXML news cast snippet.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
- <vxml version="2.1">
  <property name="timeout" value="15" />
  <property name="voicename" value="cepstral callie" />
+ <form id="form1">
- <form id="form2">
  - <field name="act">
    - <grammar mode="voice" xml:lang="en-US" version="1.0" root="command">
      - <rule id="command" scope="public">
        + <one-of>
          </rule>
        </grammar>
        <prompt bargein="true" bargeintype="hotword" timeout="15">Can Microsoft make
          Silverlight shine?.The would-be Flash killer works on Windows and Mac OS
          and is headed to Linux, but Web developers want to see it on lots and lots of
          machines before they'll commit.</prompt>
        <option dtmf="1" value="more">more</option>
        <option dtmf="6" value="next">next</option>
        <option dtmf="4" value="back">back</option>
        <option dtmf="5" value="store">store</option>
        <option dtmf="7" value="similar">similar</option>
      - <catch event="noinput nomatch">
        <goto next="#form3" />
      </catch>
    - <filled>
      - <if cond="(act == 'next' || act ==6)">
        <goto next="#form3" />
        <elseif cond="(act == 'back' || act ==4)" />
        <goto next="#form1" />
        <else />
        <submit method="post" namelist="act" next="vxml_interact.php?
          action=showarticle&aid=547" />
      </if>
    </filled>
  </field>
</form>
```

Figure 44. VXML news snippet

VoiceXML based news feed is served to a number of user agents which include either a standard telephone/ mobile phone or Session Initiation Protocol (SIP) based soft phones. The telephony and SIP interface to the VoiceXML browser is implemented by means of Asterisk IP PBX.

4.3.4.3 E-Ink

One mode of publishing supported by PINC is output optimized for E-Ink. The technology of this electrophoretic imaging film is based on a new method of converting an electrical signal into a viewable image. Unlike liquid crystal displays (LCD's), E-ink display contains electrically charged pigment particles that reflect

and absorb light. These particles interact with light in the same way as ink with paper. It results in a bright, high-contrast reflective image that is clearly legible from almost any viewing angle. Films come in very thin flexible paper format as well. When the electric field is removed, the particles remain in position, leaving behind a stable image that is readable for days, weeks, even months.

Publishing, media and content industry has shown a lot of interest in these thin flexible displays. Media hype about a recent product Amazon Kindle can be seen as signs of this interest. The E-Ink device interface in PINC envisions delivery of personal newspaper on these books like devices. Figure 45 depicts e-ink and video client interface with system.

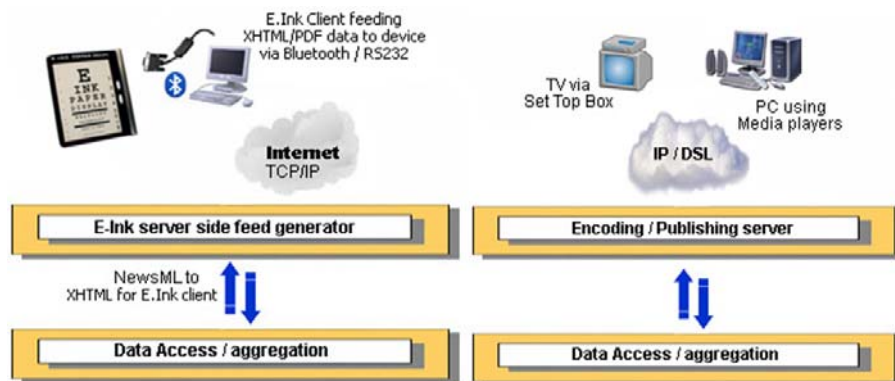


Figure 45. E-Ink and video access

4.3.4.4 Video

PINC provides on demand customized video news via video media server. In general, IP-TV service is considered as simple TV broadcast over the internet, however, there is more to it than simple streaming. IP-TV is a more controlled platform capable of user interaction and delivery of personalized and targeted contents. Recent IPTV platforms integrate multiple ways to trace user choices, preferences and selections over time. This in turn helps build user attention data for a more personalized video feed. The IPTV interface takes a selection of video contents from the news repository, encode them into formats suitable for unicast or multicast streaming and relay to the client. PINC IP-TV services can be accessed on TV via Set Top boxes or media player clients running on various desktop and mobile devices.

4.4 System Architecture

The framework is composed of distributed web components. The modules responsible for building information need and performing filtration use application syndication in form of web services. Such distributed computing gives access to

linguistic resources and extensible analysis methods that are necessary for semantic filtering. The personalization module makes use of content filtering with the application of conceptual similarity detection techniques. Collaborative filtering helps to correlate news items for users of similar interest. This approach is effective when the news items such as movies or voices have very little metadata to build content-based relevance.

The news acquisition and processing component is based on individual internet crawling agents. They are responsible for the harvesting of general news entities and personal news from numerous syndication sources. The processing unit extracts metadata data from news contents and does lexical normalization for conceptual relevancy. The text normalization process provides the semantic mapping between user interest and news items. This process helps generate and store concept term vectors of news data and filter queries based on user interest model.

Example of concept vector generation in PINC

News entity X_1 :

Runway safety in tough atmospheric conditions is poor, federal report says:
Providing pilots with more accurate information about icy or snowy runways is vital to reducing accidents, said a congressional report Wednesday that blamed safety problems at U.S. airports on sluggish government action.

Response from POS tagger service:

```
Array ( [0] => Runway() [1] => safety(NN) [2] => in(IN) [3] => tough(JJ) [4] => atmospheric(JJ) [5] =>
conditions(NNS) [6] => is(VBZ) [7] => poor(JJ) [8] => federal(JJ) [9] => report(NN) [10] => says
Providing() [11] => pilots(NNS) [12] => with(IN) [13] => more(JJR) [14] => accurate(JJ) [15] =>
information(NN) [16] => about(IN) [17] => icy(JJ) [18] => or(CC) [19] => snowy(JJ) [20] =>
runways(NNS) [21] => is(VBZ) [22] => vital(JJ) [23] => to(TO) [24] => reducing(VBG) [25] =>
accidents(NNS) [26] => said(VBD) [27] => a(DT) [28] => congressional(JJ) [29] => report(NN) [30] =>
Wednesday(NNP) [31] => that(IN) [32] => blamed(VBD) [33] => safety(NN) [34] => problems(NNS)
[35] => at(IN) [36] => U.S.(NNP) [37] => airports(NNS) [38] => on(IN) [39] => sluggish(JJ) [40] =>
government(NN) [41] => action(NN) )
```

Response from term extraction Service:

```
Array ( [0] => risky atmospheric conditions [1] => runway safety [2] => congressional report [3] => safety
problems [4] => government action [5] => runways [6] => accidents [7] => airports [8] => federal report )
```

Response from normalization service:

```
*tough atmospheric conditions -> bad (bad, badness, tough, risky) weather (weather,
weather_condition, atmospheric_condition)
*runway safety -> runway (runway, track) guard (guard, safety)
*congressional report -> congress (congress, United States Congress, U.S. Congress, US Congress)
account (account, study, written report, news report, story, paper, write up)
*safety problems -> guard (guard, safety) problem (problem, trouble)
*government action -> government (government, authorities, regime, politics, governing, governance,
government_activity) action (action, activity)
*runways -> runway (runway, track)
*accidents -> accident (accident, stroke, fortuity, chance event)
*airports -> airport (airport, airdrome, aerodrome)
*Federal report -> federal (federal) account (account, study, written report, news report, story, paper,
write up)
```

Concept vector of news entity X_1 :

```
( bad[1] weather[1] runway[2] guard[2] congress[1] account[2] problem[1] government[1]
action[1] accident[1] federal[1] )
```

Figure 46. Process of normalization and concept vector generation

Processing of news contents in Figure 46 shows different stages of concept vector generation. If a filter query containing “risky weather” is used the system will normalize it in similar manner (i.e. converting it into “bad weather”) and show higher match with news entity X_1 although query and news entity do not contain exact terms. After the processing of news entities the information is stored in system’s data repository. The information repository consists of inverted index file structures and a relational database. The inverted list based index holds the

normalized contents of news entities. The database maintains user models, news category taxonomy and links table of news entities in index with news taxonomy. News classification and user models are defined in XML. APML compliant user models are made up of explicit and implicit concept keys. These concept keys are basically terms or keywords that are used to form information filters. User can define explicit filters that include selection of feed sources, predefined category selection, and specification of terms of interest. Implicit filters are made up from user browsing or read history. News category taxonomy evolves over time; it is enriched by flow of fetched news items and metadata.

The Personalization component makes use of content and collaborative filters to generate user adaptive news contents in standard NewsML format. User interest concept vectors are generated from the user model, and, if collaborative filtering is enabled, interest vectors from matching user profiles are added to personal news selection filters. These news selection filters are then compared for similarity with concept vectors of news entities. Matching news items above user specified threshold value are passed on for user presentation.

Modules responsible for information processing and personalization make use of external resources for natural language processing. Access to these linguistic resources (WordNet database, POS tagging rules, term extraction) and similarity checking algorithms is provided via SOAP based service calls. Such internet scale computing (a.k.a. cloud computing) provides the system capability of efficiently handling complex computational tasks. This is achieved by distributing different components of system over commodity hardware based servers across internet. At presentation and data access layer contents are transformed into appropriate format for delivery through a particular user interface. Interaction module use browsing and tagging feedback to update user profile. The interaction component is responsible to cache an active newscast until a user-set timeout or a manual reload occurs. Individual news items or overviews are extracted from the newscast and handed on to the publishing component. It moreover relays request for reload to the information aggregation component and updates the user profile and model by explicit and implicit feedback as well as the information about news items already read. Finally it holds the position in an active newscast. Information flow through various components of the system is presented in Figure 47.

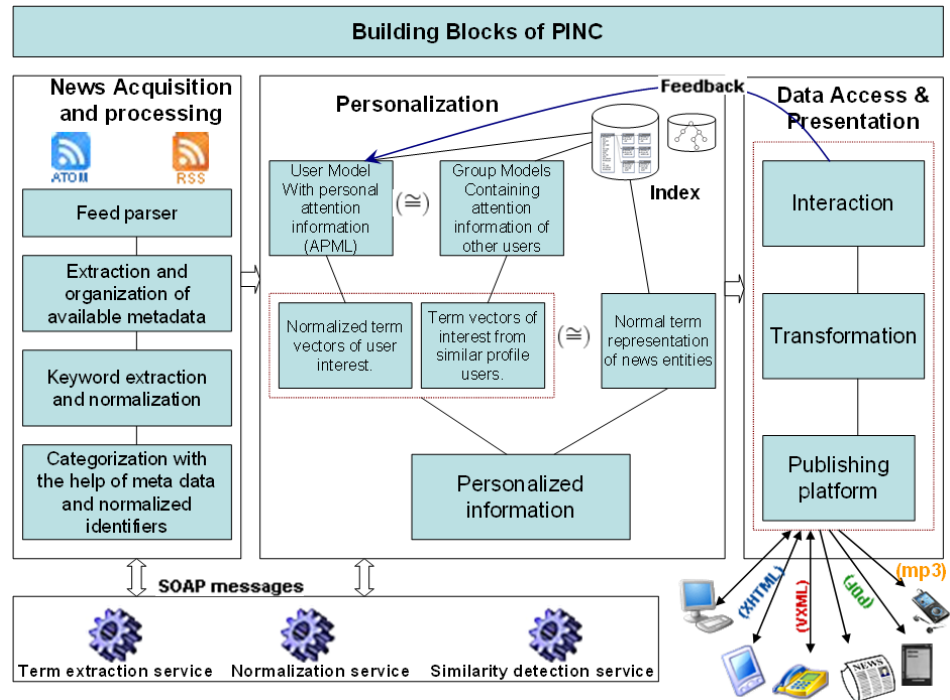


Figure 47. PINC Architecture

The publishing component of the system is responsible for the transformation and delivery of the aggregated content to a specific user interface. The NewsML structure is transformed according to contextual requirements of the interaction modality and restraints of the interface device. It also embeds feedback mechanisms to (i) give implicit (behavior based) feedback and (ii) give explicit relevance feedback to update user model. It also provides control mechanisms to navigate through a newscast. Actions by the user are relayed to the interaction component.

The system management component (not shown in main information flow diagram) is available through web based portal. It offers facilities to system users for editing the base news classification, manage the news repository, manage harvesting agents and edit user and group models. Moreover, it offers the possibility for all general users to view and edit personal preferences and content filters.

4.5 PINC Prototype

Based on the proposed architecture, a partial implementation of an interactive newscast with two modes of user interaction has been developed. The system consists of an information retrieval system to fetch news and information contents from affiliated news sites. The news fetching and processing agent is based on a Nutch crawler. The modified parse-rss, and parse-html plugins are used as fetching agents. These agents traverse through specified feed sources to gather content descriptions and meta data. Fetched contents are normalized using specially

designed web services²⁴ integrated with the Nutch crawler. We tested our prototype with external term extraction services from Yahoo and Topicalizer. There are also possibilities of using simpler and faster local service that use removal of stop words and statistical measures to determine keywords. Normalization service use a Wordnet lexical database ported into local repository.

Lucene, an open source Java based API for indexing and search, is used to create a normalized index of fetched news entities. The prototype is developed keeping in view the requirements of handling heterogenous and large collection and news entities. The open source and plugin based architecture of Lucene and Nutch allows ease of modification and handling of multiple content types. The process of detecting similarity is performed in a dynamic manner on incremental index. The search processing is far more efficient than any conventional database or file based system. The similarity detection service is based on the vector space model. It creates weighted vectors of contents being compared for similarity and user attention filters. These vectors are mapped against the combined local vocabulary of compared contents. The angular measure (dot product) of these vectors is used as a score to determine similarity.

The system has a web-based management console for user registration, scheduling for content retrieval agents, profile and interest parameter insertion. The management console furthermore has the capability to add or modify the information retrieval agents, news categories, and interest groups. Based on the user profile the selected news and information is aggregated as an XML source which in turn is fed to XSL transformation routines for generating appropriate contents for the user's view. The system uses an Apache web server with "mod_negotiation" and PHP Content Negotiation library for client specific automated formatting of XHTML contents.

Currently the system is providing access to users with standard desktop browsing support through an application web server and a dialog based interactive speech browsing through a VXML 2.1 compliant browser. Limited port phone/SIP connectivity is also available for voice access tests. We tested our system with Loquendo's Voxnauta and Voxeo's prophecy VXML publishing platforms, the latter being freely available with port restriction. Both platforms are capable of VoIP access via Session Initiation Protocol (SIP) clients. Telephony support is added via Integrated Services Digital Network (ISDN) Channel on Asterisk linking to the Voice browser via SIP. Common-ISDN-API interface module in asterisk is used for communication through basic rate interfaces (BRI) card linking 2 phone channels to PINE's VXML server.

For vocal presentation the textual contents and dialogs are generated at runtime via the integrated TTS. The archived audio files are converted and transcoded to the

²⁴ <http://fiicmpc140.tu-graz.ac.at/webservices/>

proper format which is suitable for relaying on telephone and internet channels. The user activities and system access is logged and stored in a behavior database.

4.6 Summary

This Part presented a framework that provides ability of adaptive news content selection from heterogeneous sources and allow access at any time, any place. The first goal is achieved by using similarity detection on enhanced metadata to aggregate semantically equivalent news. Moreover collaborative filtering is applied to integrate further news items based on the selection of users with similar interests. Use of adaptive information agents and recommender systems to help users handle the increasing amount of information has increased considerably during last few years. These adaptive systems use content based, knowledge based, social or hybrid filtering mechanisms. A survey [Adomavicius and Tuzhilin, 2005] about state of art and possible extension in recommender systems suggest that despite all advances in filtering mechanism (content/knowledge/social) there is still room for further improvement. Possible improvements include less intrusive and improved user modeling, more meaning full and contextual annotation of items, and support for multi criteria ratings. Our research effort tries to fill this gap by application of conceptual hybrid filtering and a standardized user modeling approach. This work describes a user modeling approach that uses both explicit knowledge and implicit behavior based interest data, it stores this information in a reusable format, owned and controlled by individuals not the system.

The second goal of PINC requirements is met by applying cross-media publishing technologies and integrating multimodal interaction with the system. Thus a wide range of interfaces can be used to access PINC. An analysis of US based internet newspapers found out that 86 percent of these news companies had cross media publishing support. These publishing modes include print, online, television and radio [duPlessis and Li, 2004]. Addition of cross media publication and multimodal interactions helps overcome inherent weaknesses of any single delivery media. It increases the system audience with alternative access possibilities to meet impulsive user needs. There are efforts to complement news delivery with addition of one or more media channels [Ma et al., 2004]. These systems show a need for complementary information infrastructure to filter, link and present information that satisfy delivery context. In all aspects PINC is designed for modifiability and extensibility, in order to support most of the commonly used information delivery channels. It provides a standardized platform which adds this complementary information infrastructure.

Future work includes user feedback or rating analysis to find effectiveness of semantic mapping between information need and news items. We are also exploring the use of a user modeling component as user interest profiler. Such a system can be used to automatically create a rich user interest knowledgebase.

Standardized user attention model provide possibilities of reuse in a number of supporting information retrieval environments.

A complete deployment of the system aims to revolutionize the way a person deals with daily information sources. PINC will give convenience of selecting a single most appropriate way of interaction with a vast, personalized body of news, depending on physical and environmental conditions.

4.7 Information supply for knowledge workers

Information search- and retrieval- processes play a vital role in the productivity of a knowledge worker. Every knowledge worker has to do extensive searches at some point in time to find information that may help, or show that certain aspects have already been covered before. Search engines provide the basic means of interaction with the massive knowledge base available on the World Wide Web.

Conventional search technology uses a pull model: i.e. search engines require an input from the user in form of a query consisting of keywords. This active search paradigm has a number of downsides: knowledge workers normally are not trained for really comprehensive searching. They may not know all the tricks required to locate the right sources of information. They may not know how to formulate a query describing all that they want to find.

The formulation of search queries is often difficult due to special terminology, or just the difference of terminology used by authors in various sources. Another constraining factor of typical search engines is the fact that they only cover shallow web contents, ignoring the almost 500 times larger, “deep” or invisible web information base [Bergman, 2001]. There are special search interfaces for domain specific searches but not all are commonly known to general users. Thus, any organization or individual pays a high price due to ineffective information discovery. According to an IDC white paper by Susan Feldman, knowledge workers spend 15% – 35% of their time searching for information. The success rate of searches is estimated at 50% or less. The study further states that knowledge workers spend more time recreating information that already exist, simply because it was not found when needed. These factors contribute to a waste of time, costing individuals and organizations a substantial amount of money [Feldman, 2006].

4.8 Searching the web for knowledge acquisition

The primary step in knowledge work is the acquisition of information. Access to first hand information comes by means of reading literature, by meeting subject specialists, by learning technologies and finally but most importantly making use of the immense information space of the internet.

Most knowledge workers consider online technologies to be the most efficient way to access information. Web search engines no doubt are the gateway to this information base. A research study [Hölscher and Strube, 2000] describing behavior of web search by both experts and newcomers shows that when presented with information seeking tasks, the majority resorts to search engines instead of browsing direct sources such as collections of journals made available by some publishing company. Further experiments show that searchers quite frequently switch back and forth between browsing and querying. The switching involves reformulation, reformatting of queries and change of search engines, based on previous result sets. The study also states that web search experts make more use of advanced search features such as specialized search engines, Boolean operators, search modifiers, phrase search, and proximity search options. Domain experts tend to show more creativity as far as terminology is concerned and form better and longer queries. The overall information seeking process as described by the experiments in the study shows the difficulty and the reduction of productivity in the majority of cases. A knowledge seeker is required to have good searching capabilities as well as good domain knowledge with rich vocabulary to successfully accomplish the goal. However, this is not always the case: there was and still is a big need for the enhancement of search environments for knowledge seeker.

There are attempts to facilitate searches in the form of

1. Meta search services covering multiple search databases e.g. Jux2, Dogpile, Clusty²⁵ etc.
2. Web based and desktop tools based on search APIs to facilitate advance searching e.g. Ultraseek, WebFerret²⁶ and many web search API mashups available at ProgrammableWeb portal²⁷
3. Desktop tools to explore local resources e.g. Google desktop, Yahoo desktop, Copernic²⁸ etc.
4. Specialized search engines, typically content specific like Google Scholar, Live Academic²⁹, CiteSeer, Scirus, IEEEExplore³⁰ etc. or media specific like image search, video search etc.
5. Semantic search e.g. Swoogle, Hakia³¹ etc.

Meta search approaches provide access to multiple search sources but the process of advanced query generation is more complex due to different query formulation options of the search engines. Sometimes variations in results by meta search requires greater examining time and effort. Tools to facilitate searches based on APIs provide another meta search option with some query optimization facilities (spelling suggestions, synonyms support, easy use of advance search options, etc.).

²⁵ <http://www.jux2.com/>, <http://www.dogpile.com/>, <http://clusty.com/>

²⁶ <http://www.ultraseek.com/>, <http://www.ferretsoft.com/>

²⁷ Mashup & web 2.0 API portal: <http://www.programmableweb.com>

²⁸ Copernic search engine for desktop: <http://www.copernic.com/>

²⁹ <http://scholar.google.com>, <http://academic.live.com>

³⁰ <http://citeseer.ist.psu.edu/>, <http://www.scirus.com/>, <http://ieeexplore.ieee.org>

³¹ <http://swoogle.umbc.edu/>, <http://www.hakia.com/>

However, they all suffer from another drawback: the limit of allowed queries in terms of quantity and quality. A quantitative analysis [McCown and Nelson, 2007] of results produced using conventional WUI (Web User Interface) and APIs, shows significant discrepancies. Results produced by search engine's own interface and APIs are rarely identical. This seems to suggest that API access probably is restricted to a smaller index. Specialized search engines provide platforms to look for information in a broader context. Locating relevant search sources and seeking information in limited indexes individually is again a time consuming task. More recent attempts to add semantic element to search suffers from the limited scope of available ontologies. The semantic web vision based on the Resource Description Framework (RDF) and Web Ontology Language (OWL) is yet to gain popularity among common content providers and developers. Information systems are still a long way from reasonable data annotation in standardized formats. Finally, all above automation attempts fall under the same query based active search model (referred to as pull model in introductory part) and only provide surface web search. All regular searchers and particularly knowledge workers feel a strong need to go beyond these restrictions.

4.9 From Information retrieval to information supply

In an interview, Yahoo's Vice-President for research and technology describes the next generation search to be a "search without a box". This formulation indicates a move from information retrieval towards information supply, where information comes from multiple sources, in a given context, and without actively searching [Broder, 2006]. For an effective information supply, understanding the context is very important. An ideal approach for context aware search would be the use of semantic inferences. However as we have mentioned earlier even after almost eight years --- this is how long the concept of semantic web has already been around---, there is no sign of mass implementation. Billions of web pages and documents still contain no or very few annotations and pieces of meta information. Thus, a mechanism is required to effectively finding the context of information and bridge the gap between conventional and semantic web. Context can be defined as effectively interpreting the meaning of a language unit at issue. An analysis of a large web query logs [Beitzel et al., 2004] shows that average query length is 1.7 terms for popular queries and 2.2 terms averaged over all queries. This seems to indicate that input information is not enough to determine the context of a search.

We propose an information supply model for knowledge workers based on similarity detection. The proposed information supply method is utilized at the level where information seekers have an initial draft of their work available in written form. This input document is used to define the information need. It could be either the abstract, the initial draft of the task at hand or some document similar to the area of work currently carried out. The information supply engine performs the following processes to facilitate the search process:

4.9.1 Term extraction and lexical variations

Terms can be seen as elements in a language to describe particular thoughts. They are the designators of concepts in any document. That is why automated processing of term recognition and extraction has been a critical aspect of natural language processing research. Term extraction approaches can be mainly categorized as statistical and linguistic. Statistical techniques identify terms and important phrases using factors such as consistency and structural location. Other approach makes use of linguistic morphology and syntactical analysis to identify terms of high importance. Substantial research in both techniques has transformed cutting edge research into usable applications and products. We can find examples (Yahoo Term Extraction³², Topicalizer and TermExtractor³³) that successfully apply these methods to identify important terms highlighting concepts behind just textual information. Linguistic resources are used to find lexical variations in terms. Lexical variations (Synsets in WordNet) are also used for canonical representation of information need. This normalized representation will be used for search result analysis at later stages. Terms can be extracted from knowledge work space on word count basis (fixed text chunks), paragraphs or complete documents. The extracted term batches will form queries to be processed by search services.

4.9.2 Determine the subject domain with the help of classification systems

In order to enhance the quality of search, additional meta data association can be very useful. There are several standardized classification initiatives in the form of taxonomies and topic maps. A subject domain can not only help to determine additional meta information to enrich search queries, but can also help in the selection of appropriate search services and results. Domain specific selections by adding this semantic element to information supply to search sources and results will reduce the “noise” (i.e. the undesirable information) generated.

4.9.3 Query expansion

Another approach is to use lexical enhancements. Cognitive synonyms and domain significant co-occurrences found with the help of lexical resources are used to expand queries. The idea behind lexical enhancements is to identify related information even if user defined terms and information terms do not match. This expansion provides an improvement in classical search where matching is based on simple content match, vector space, link analysis ranking etc. A still further step is to move to advanced query formation. Cognitive terms and phrases are organized to form complex Boolean search queries. A query routing agent determines the use of AND, OR, phrase, include, exclude, and proximity operators for a specific search interface.

³² <http://developer.yahoo.com/search/content/V1/termExtraction.html>

³³ <http://www.topicalizer.com/>, <http://lcl2.di.uniroma1.it/termextractor/>

4.9.4 Distributed search services

Our proposed information supply engine maintains an up-to-date index of general search APIs and deep web resources. Knowledge workers may configure the information supply based on general web services or include domain specific deep web search. Almost every search engine provides web service interfaces and access to its index through the XML standard. Search services combine results of common web search engines, along with deep web search engines. The processed term batches are sent as queries to search interfaces available in distributed framework of system.

A recent book by Milosevic highlights the importance of distributed information retrieval systems. The suggested framework makes use of intelligent agents to establish coordination and apply filtering strategies [Milosevic, 2007]. In our information supply architecture we propose the use of distributed indexing and sharing to address the restriction of view issue of web search companies and access to deep web. On the basis of success of peer to peer file sharing applications a similar indexing and searching network is envisioned. The distributed nodes working at institutional or personal level provide open search access to deep web resources. Such distributed indexing approach can have numerous other applications; one example is illustrated in Collaborative Plagiarism Detection Network architecture [Zaka, 2009b][CPDNet, 2008]. The Web presents a huge and continuously growing information base, but “search engines that crawl the surface of the web are picking up only a small fraction of the great content that is out there. Moreover, some of the richest and most interesting content cannot even be crawled and indexed by one search engine or navigated by one relevancy algorithm alone” [OpenSearch, 2007]. Open and distributed search initiatives provide common means of search result syndication from hundreds of shallow and deep web search engines.

4.9.5 Result analysis with the help of similarity detection

In conventional active search models the iterative process of obtaining and quickly examining the results provided by search engine consumes a lot of time. Information supply for knowledge worker provides an automated analysis of result using similarity detection techniques. The resulting URIs are fetched by an analysis module. The fetching module uses PHP’s cURL library to extract textual contents and removes the formatting information (HTML tags, scripts etc.). Term extraction and lexical data services are used again to obtain the gist of contents. Similarity is calculated between information need and processed result contents with the help of vector space mathematics. The similarity detection service is fed with term batch of knowledge workspace and the terms of search result URIs. The terms/words are mapped as vectors against a compound term vocabulary. The dot product of two vectors (cosine similarity) determines the relevance. The semantic space for result analysis is built with the help of two additional services, namely POS (Part of Speech) tagger and Synonym [CPDNet, 2008]. POS tagger returns the sense of each word (i.e. Verb, Noun etc.) and Synonym service based on WordNet 3.0

returns the related synonyms and Synset IDs. The normalisation or canonical lexical representation of terms introduces a semantic relevance element in similarity detection process. A paper on power of normalized word vectors [Williams, 2006] presents the described concept in detail. Such analysis capability can replace typical result browsing and filtering in information seeking process.

4.9.6 Result mapping to knowledge space

Key terms and phrases of the knowledge work space (input document) are linked with matching information sources. Result mapping with a rich web user interface provides a clear picture of relevant documents. The links include subject domain, key terms, persistent phrases, and summary of matching source. This function provides far better first hand knowledge about information source than the simple hit-highlighting as is done by a normal search engine. Users will have a higher degree of knowledge whether to further investigate matches suggested or not.

4.10 Service oriented model

The model shown in Figure 48 is a step towards the realization of a comprehensive information supply system. In order to provide the functions described in the previous section, we emphasize the use of off-the shelf tools in form of web services. Combination of external and internal services provides the depth in search and automation in information supply and analysis. The first level in the information supply environment is to determine the information need with the help of term extraction. The knowledge work space is submitted to term extraction services. A term extraction process in information supply model consists of converting formatted contents into plain text. The plain text is passed to a Part of Speech (POS) tagger in order to determine word sense and to eliminate function words. Key terms (verbs and nouns with higher occurrence frequencies) are further enriched using linguistic resources: WordNet and Wortschatz³⁴ lexical databases are used to get synonym terms and synonym group IDs with similar sense. This data is used for query expansion and canonical lexical representation. The initial level also attempts to relate information requirement to a subject domain. Initially the subject domain is determined with the help of predefined standard taxonomies. One example is the use of ACM Computing Classification System³⁵: the keyword index of each category can be checked for similarity with extracted terms. A web service developed to calculate angular and distance measure among two word/term vectors mapped in compound term vocabulary is used for similarity check.

³⁴ Wortschatz: <http://wortschatz.uni-leipzig.de/>

³⁵ ACM Classification: <http://www.acm.org/class/>

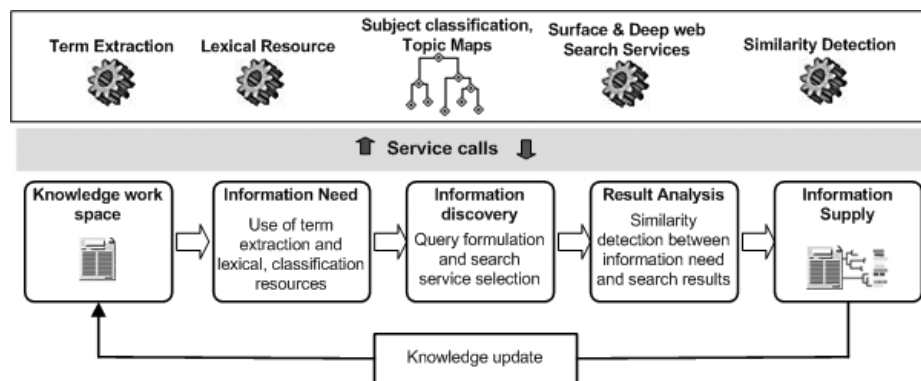


Figure 48. Information supply model for knowledge workers

After formation of the consolidated information need, the system selects available search services. The mashup of search services provides a broader coverage of distributed indexes of the shallow and the deep web. A distributed search test bed is used in the proposed information supply model³⁶. The distributed search framework acts as a proxy for search services. It not only includes popular search APIs like Google and Microsoft Live but also support OpenSearch and peer search. The search results are filtered at the next level with the use of already mentioned similarity detection techniques. Cosine similarity measure is determined from term vector of knowledge workspace and term vector of returned results from search services. The term vectors can be additionally presented in normalized form (canonical lexical representation) in order to develop semantic relevance. Filtered results with high angular similarity measure are linked with the knowledge workspace. Knowledge workers can see first hand similar data available on the internet. With an update of the knowledge space by incorporating new relevant documents found users can initiate the same process for an updated information supply.

4.11 Summary

In the second part of this chapter, a composite use of web services for an information supply model is suggested. The model presented is a step forward from classical IR to proactive search systems. We introduce the use of lexical services and similarity detection to (i) find the context of a search and to map the information need to a specific subject domain, and (ii) provide an automated result scanning service, similar to human judgment. Both elements play an essential role in efficient information supply for knowledge workers. The research indicates that additional meta information found via context of search and lexical resources can prove to be very useful in the creation of automatic search queries. The use of mathematical techniques to determine information relevancy can also eliminate a time consuming and iterative process of manually scanning search results.

³⁶ <http://www.cpdnet.org/isdemo>

Our experiments of search service mashup indicated very good possibilities of search access across multiple jurisdictions. The research on information retrieval in the internet and role of search engines also pointed out issues of restrictions in general web search APIs. The success of the next generation web or web 2.0 depends not only on the collaborative efforts from users but also on open and honest syndication and standard API provision by enterprises. The discontinuation of application consumable search service by Google, no availability of search syndication by specialized search engines like Google Scholar and Live Academia are examples of undesirable restrictions against which the community should protest before it is too late. There is a strong requirement for a scalable and open search and indexing platform. In order to develop such a platform, use of peer to peer search with user or institute level indexing is worth serious consideration.

4.12 Outlook

The two experiments discussed in this chapter show a shift of information systems towards scalable information services. Scalability defines the behaviors in growing or more demanding environment. For web applications scalability specify the response or ability to perform expected operation on larger scale. By larger scale we mean expanding environments in terms of users, data, services, operation domain and hardware. The following part of chapter includes a discussion about scalability and composite use of web services based on our experimental experiences.

The two approaches of making system scalable are: number one, support for vertical scalability where response is controlled with the expansion of system within a single logical unit. Number two is the support of horizontal scalability which refers to expansion as multiple logical units working as a single entity.

Because of current nature of web based information retrieval environments (shown in investigated application areas of this dissertation) we are more interested in horizontal scalability. The factors effecting horizontal scalability include ability to leverage open industry standards, use of flexible technology to include different systems and data structures. It also requires a generic interface to clients for broader coverage. But most importantly it needs a good controlling mechanism for integration of resources. Although addition of horizontal scalability in system does not require addition of expensive hardware for larger operations; it also saves the effort for building and organization and storage of combined knowledge space. However in order to achieve successful horizontal scaling, applications must be built using a specific architecture. This architecture must have support for network files system (for scalability of data layer), distributed computing and load balancing at application layer

4.12.1 Utility computing

In order to make information systems horizontally more scalable several distributed computing models exist. Most notable current ones include cluster computing paradigm where a group of interconnected stand alone computers cooperatively work together as a single and integrated computing resource. Another distributed computing models is the Grid computing approach, according to IBM Terminology [IBM:Term, 2008] “A Grid is a type of parallel and distributed system that enables the sharing, selection, and aggregation of geographically distributed ‘autonomous’ resources dynamically at runtime depending on their availability, capability, performance, cost, and users' quality-of-service requirements.” In general Grid based systems provide the users with high level services for accessing information and application on the grid, all embedded into a consistent security framework [Jones, 2008].

A more recent model of distributed internet scale computing is termed as Cloud Computing. It is seen as the evolution of the Software as Service. According to the definition at Wikipedia "Cloud computing is a general concept that incorporates software as a service (SaaS), Web 2.0 and other recent, well-known technology trends, in which the common theme is reliance on the Internet for satisfying the computing needs of the users. For example, Google Apps provides common business applications online that are accessed from a web browser, while the software and data are stored on the servers." The later mentioned two models visualize the idea of utility computing.

Web search trend by Google (figure 49) shows the interest of people in these models over the last few years. One can see that interest in cloud computing is on the rise.

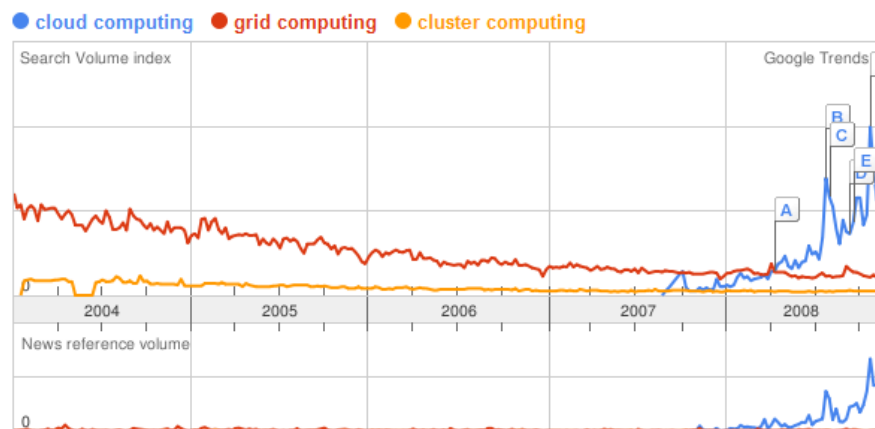


Figure 49. Google trend for Cluster, Grid and Cloud Computing

Although the Grid and Cloud models are considered new, however the idea behind them is not new. Even before the inception of modern internet and World Wide Web the idea of disseminating information and computing resources as services

was present. John McCarthy (MIT Centennial) states in 1961 “If computers of the kind I have advocated become the computers of the future, then computing may someday be organized as a public utility just as the telephone system is a public utility... The computer utility could become the basis of a new and important industry”. A press release from UCLA [UCLA, 1969] informing about the launch of ARPA Network (birth of internet) states “As of now, computer networks are still in their infancy, says Dr. Kleinrock. But as they grow up and become sophisticated, we will probably see the spread of ‘computer utilities’ which, like present electric and telephone utilities, will service individual homes and offices across the country.” This vision of utility computing involves organization and provision of wide range of hardware and software resources spread across shared infrastructure. The resources are made available to individuals based on traditional utility service model. In such model a services is accessed and used as per need, without caring about the hosting and delivery issues.

During last few years World Wide Web has transformed into a distributed and global application platform. Web sites and applications are turning more and more into services that allow information exchange with other systems in addition to information provision to user. Major factor that transformed web into a platform for distributed computing is the use of XML. XML describes structured data in standard plain text rather than any application specific representations. This enabled web applications to share and store data globally; this provided the ground bases for developing loosely coupled applications. What really boosted utility service based computing environments is the use of universal connectivity in form of SOAP and REST based web services [Coyle, 2002]. Web services provide interoperable machine-to-machine interaction over WWW. Based on standardized and platform independent protocols they help in building distributed and collaborative web applications. The vast industry adoption of web services has made SaaS (Software as a Service) a generally available approach. SaaS is considered as a subset of cloud computing model. Cloud computing is based on actual idea of utility computing proposed earlier. This approach is basically a large scale and more organized form of web service oriented architecture. Major companies scaling up their service to the level of cloud computing include Google, Amazon, IBM. Yahoo also joined the race with an open and supportive initiative of cloud computing research. The support of Hadoop³⁷ and rolling out open coordination tools gives research community a chance to better organize their web services and be part of cloud computing development.

4.12.2 Workflow in web services

The reasons described in previous section and as per trend shown in figure 49, many organizations are now interested in developing large scale web service platform and integration of these services in various applications. Such scalability requires efficiency, reliability, and security at a greater level. For application built using these composite services, efficient workflow management is of great

³⁷ Hadoop: <http://hadoop.apache.org/>

importance. Web service workflow system complements the service discovery, description and messaging capabilities. The web service architecture described in Chapter 3 (section 3.9.1) outlines service operation in terms of i) publishing and discovery, ii) service description, and iii) message exchange. Workflow is added as fourth element for composite use of services. It adds capabilities of seamless operations, coordination and monitoring. In general two approaches for developing workflow systems are based on:

4.12.2.1 Orchestration Model

Orchestration based composition of services adds workflow management from a single party prospective (centralized approach). All communication is routed through a workflow engine. It monitors and controls message interactions among services that describe business logic and task execution order. [Peltz, 2003]. The WS-BPEL (Web Service Business Process Execution Language) [WSBPEL, 2007] provides the XML based grammar for composition of service flows in orchestration engines.

4.12.2.2 Choreography Model

Choreography is more collaborative in nature, without any centralize control unit for service coordination. Individual collaborating services are more active and aware of workflow in this model. Coordination is achieved through the public message exchange among web services. WSCI (Web Service Choreography Interface) defines the message exchange operation e.g. service correlation, sequencing rules, exception handling, transactions etc. [Peltz, 2003]. Figure 50 shows an overview of workflow methodology in two models.

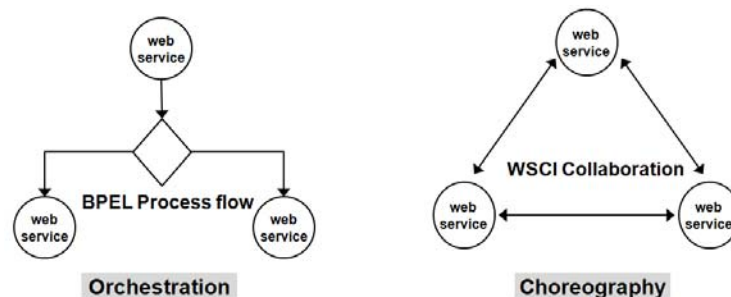


Figure 50. Orchestration Vs. choreography

4.12.3 Distributed orchestration

Choreography based composition is based on observable interaction among peers, unlike orchestration there is no central controlling body. Each service should be aware of when, how and with whom to interact. This makes workflow management based on choreography a rather complex task. Such systems have lesser fault tolerance and require extra effort for service integration. On the other hand orchestration is more flexible where web services are coordinated for computational tasks without their being aware of participation in a larger business process. However the centralized coordination approach also suffers from of issues such as

single point of failure, QoS, secure communication etc. The developed prototypes use the collaborative approach where individual service components are responsible for sequencing the flow of operations. To reduce additional development load on applications using the services and better management the orchestration based workflow model is a better candidate. Future work includes development of distributed orchestration model, which overcomes the inherent deficiencies of conventional single point of failure orchestration approach. The use of distributed coordination tools³⁸ for orchestration of composite web services is currently being investigated.

³⁸ ZooKeeper: <http://hadoop.apache.org/zookeeper/>

5. Content Organization

Applications of similarity checking for context aware object classification and increased content reusability.

Contents of this chapter are taken from

"A practical approach to enrich classification of digital libraries" [Zaka, 2009a]

"Topic-Centered Aggregation of Presentations for Learning Object Repurposing" [Zaka et al., 2008]

In today's digital age, there is an abundance of electronic information. Although this makes information more accessible, it becomes a challenge for individuals and organizations to store, retrieve, and reuse information effectively. There are number of ways to organize information, organization can be either word based matches (which is commonly used), can be structural, time specific, spatial, topic specific, task specific or audience specific (rating based). These organization schemes rely on various similarity measures to determine relationships among contents. The work reported in this chapter uses techniques that exploit combination of various similarity models, structural and linguistic resources to discover and classify matching contents. First application shows a practical approach for finding relations and groups among documents within a local repository or archives spanning across an intranet or Internet. The strengths of this technique are discussed through its use to enrich the classification of a digital library, and provide means to retrieve similar documents. Second experimental investigation shows the use of layer similarity checking for topic centered content aggregation and repurposing of learning contents.

5.1 Introduction

Document archives represent the accumulated knowledge of any knowledge worker or organization. This knowledge base grows quickly due to the massive information overload mainly attributed to current digital information technologies. Effective use of this information is vital for success in current information society. The size of collection, diversity in contents, different means of access and type of documents are the few factors that make information discovery a challenging task [Beil et al., 2002]. The growing information repositories like user generated contents, documents at web or file servers and downloaded files are often unclassified or poorly classified. In general the manual categorization is made based on documents types or broader domain of documents. The automated approach tries to categorize the documents based on their contents. The automated categorization is either

supervised, where documents are categorized using some external reference (predefined categories, human input); or the categorization is unsupervised, usually referred to as clustering. In unsupervised approach, documents are categorized without any external or predefined reference.

Common methods used to achieve clustering information include partitioned based clustering, hierarchical clustering, overlapping clustering and model based clustering. Some description of these techniques is available in section 5.3.4. A comprehensive review of basic algorithms and techniques is available at following reference [Jain et al., 1999]. Modern NLP techniques can also aid in improving clustering. Few of these are stemming, term selection and weighing, latent semantic analysis which analyze the relation between a document and term, by arranging them in a matrix. Although an old problem, clustering still poses challenges in terms of efficiency, quality and scalability of clustering algorithms. The strategies to update clustering information in case of growing repositories and its application in various fields are also few points of significant interest. This work illustrates a practical approach that makes use of enhanced indexing, information processing, search mechanism and content structure analysis to identify matching group of documents. The clustering is normally used to facilitate information retrieval system; in this particular approach information retrieval techniques are used to facilitate clustering process. The usability of developed prototype is shown through the enrichment of a digital library classification system. This describes the basic idea behind the approach to detect similar documents in document collections, following section illustrate use of clustering. Section 5.3 describes various steps of information processing to support clustering process. Section 5.4 presents the architectural design of system. Section 5.5 shows experimental results of system and future direction of work.

5.2 Clustering in practice

Linking similar objects or content clustering has quite a few promising applications. Particularly in search and information retrieval environments clustering is used for

- Assisted information discovery: Many search engines give user options like “Related Pages”, “More like this”, “Find Similar”
- Result grouping: To help user easily navigate through returned result set organized in document clusters based on contents e.g. Clusty (clusty.com)
- Search directory formation: Clustering facilitates organization of documents for systematic discovery. One prominent example would be Yahoo which is the oldest searching directory.
- Recommenders: Clustering based on similarity of various factors helps build efficient recommenders. Some example could be Internet Movie Database (www.imdb.org), Amazon (www.amazon.com).

Clustering proved its importance in many general data mining applications. With huge repositories of unstructured information, division of data entities into related subsets is always desirable. Clustering helps extract meaning information out of such vast archives in number of ways. Well known cases are

- News topic categorization: done by Google News (news.google.com)
- SNA: Clustering is widely used in Social Network Analysis (SNA) and market research to identify communities of common interest and segment to be targeted for a particular campaign.
- Content classification: Web/intranet based enterprise information management system use clustering to maximize search and access functionalities.
- Libraries: used for ordering digital books

Biology and medical sciences also use clustering in a number of applications such as grouping genetic information, grouping of plant and animal species on various levels. A more recent use of clustering is the email spam filtering. The filtering system makes use of clustering algorithms to automatically classify an email as spam or legitimate email. Clustering techniques are exercised in data de-duplication applications where identical elements from various types of object collection are removed. This process helps maintain a cleaner and space efficient data collection. Clustering approaches are also very useful in plagiarism detection systems. With clustering it is more practical to find a text/document similar to any given document. System can opt to use more deeper, semantic and computationally expensive plagiarism checking algorithms on a roughly matching cluster. Such detailed analysis may not be practical on complete set of data.

5.3 Information processing to link similar documents

The unsupervised organization of document collections involves a number of information processing and management tasks; they are visually highlighted in figure 51.

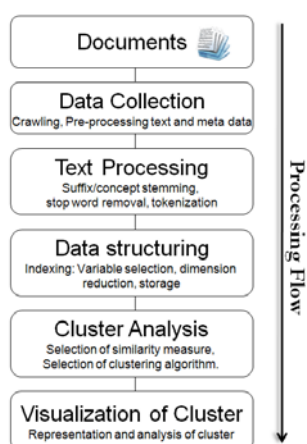


Figure 51. Processing steps in document clustering

The approach to find similar document presented here can be described in following subsections. The processes described in subsections carry out the various steps shown in figure 51.

5.3.1 Corpus Selection

The empirical dataset selected for testing comes from an online repository of research articles manually categorized in a standard way. Further details of document corpus are available in Section 5.5. The document collection contains files in various formats and contains text on different subjects.

5.3.2 Crawling process

Crawling can be described as a process of collecting information in a uniform format. A crawling application automatically traverses through the document collection for fetching of contents of different file types. It implements policies to gather updated contents and pass that on to indexing system for storage. Crawling is not restricted to web or internet domain; focused or targeted crawlers can be used to gather contents from local file system or used over networks via a variety of access protocols. In online clustering applications data collection and index updating strategies play an important role. There is a lot more to crawling than mere fetching of the data. Tasks of crawlers also include transformation or extraction of data from various file formats. A good crawling application must be capable of effectively stripping undesirable data such as formatting information, executable codes and extracting meta information. While crawling for the information over the internet one must take into account the legal and ethical aspects. Some sites do not want their contents analyzed automatically. One way of restricting crawlers from doing so is the use of robot.txt file which specifies the rule for crawling a particular domain/site. A good crawler must follow the rules given in robot.txt file. Multithreaded crawl jobs tend to generate a lot of internal and external network traffic; parallel fetching should be restricted from a single remote host.

5.3.3 Indexing and Search

Once all the data to be clustered is acquired in a uniform format, it is processed for variable selection and storage. Unnecessary information is stripped off and data is stored in an optimized format for application of similarity checks and clustering algorithms. In presented system, internet search and indexing platform is used for efficient data storage and processing. Indexing is a process of converting data into suitable format for search and analysis. There are a number of indexing techniques available such as inverted file, signature files, Suffix array and Suffix trees. However the most effective index structure for large document collections and text processing is considered to be inverted list based [Zobel and Moffat, 2006]. Advance indexing applications maintain enhanced inverted lists with application of various filters, tokenization methods, stop word removal, word stemming and term weighting. Advance index structures may also contain additional information such as word position data to support phrase queries, proximity search etc. In order to find similar documents, system is required to efficiently store huge amounts of structured/unstructured data and perform exhaustive search operations.

In given scenario, use of large scale search engine platform is preferred over storage in either conventional database or file system. Search engine platforms can be seen as specialized databases that maintain indexes and store records to ensure that indexes are loaded quickly. Indexing platform adopted, creates data structures that are well organized for efficient search with possibilities of local or distributed file storage. The query processor available in search platform is tailored for providing results for similarity/popularity conditions instead of just finding match for fixed logical conditions.

5.3.4 Cluster Analysis

There are a number of clustering techniques available to group similar objects. They can be mainly categorized as

5.3.4.1 Hierarchical

This type of clustering is known to produce better quality cluster, however effectiveness of this approach is limited by the time complexity. The sequential agglomerative hierarchical method is the most popular of this kind. It is a bottom up approach where each entity of object is considered as separate cluster. Processing to merge most similar is continued until only one cluster is left. This generates a nested sequence of partitions (dendogram) with single cluster of all objects at top and individual objects as clusters in bottom. Another type is divisive method which follows the reverse pattern of agglomerative technique.

5.3.4.2 Partitioning

This type of clustering algorithms generally produces a specific number of distinct, non overlapping clusters at once. The k-Means algorithm is most widely used technique of partitioning clustering. This technique starts with a selection of specific number of clusters (K). K random points are generated as cluster centers, and each

individual object of collection is assigned to nearest cluster center. In multipass based algorithms new clusters centers are recomputed and same process of object assignment to nearest center is repeated until convergence of assignment is observed.

5.3.4.3 Overlapping

In such algorithms a single object may belong to more than one cluster with a certain degree of closeness. The algorithm used to produce overlapping clusters (Fuzzy C-Means) uses fuzzy sets to cluster data. In fuzzy C-Means Clusters, every point has an extent of association to clusters, as in fuzzy logic, rather than distinct association with a single cluster. The objects on the edge of some cluster are considered to have lesser degree of association than the objects in the center of cluster.

5.3.4.4 Ordination

Techniques based on ordination uses the low dimensional projection (usually two dimensions) of multi dimensional objects. Similar objects are placed close together, and dissimilar objects are placed far apart revealing any intrinsic pattern of similarity. A number of techniques are being used in this class. They include Principal Component Analysis (PCA), Reciprocal Averaging (RA) - Correspondence Analysis, Detrended Correspondence Analysis (DCA) and Nonmetric Multidimensional Scaling (NMS).

5.3.4.5 Model-Based

Model-based clustering techniques use certain mathematical/statistical models for cluster generation; by optimizing a fit between the data objects and model. Commonly used algorithms include Gaussian Mixture Model, Neural Network based (self organizing map SOM, learning vector quantization LVQ etc.), Bayesian model.

Following references can be used for further details of these techniques [Milligan and Cooper, 1987] [Xu and Wunsch, 2005].

5.3.5 Similarity measures

Measure of closeness among data objects is the most important component of all clustering algorithms. There are a large number of similarity/dissimilarity (or distance) measuring means. Selection and use of the measure of closeness depends on the type of data and used algorithm. Some of the commonly used measures include

Jaccard Similarity: A simple and efficient means of similarity calculation.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

Minkowski Distance: Generalized metric distance for high/multidimensional data.

$$D_{ij} = \left(\sum_{l=1}^d |x_{il} - x_{jl}|^{1/n} \right)^n \quad (2)$$

At $n=1$ it becomes Manhattan/city block distance, at $n=2$ becomes Euclidean distance which is commonly used in K-means. At $n \rightarrow \infty$ it becomes Chebyshev distance.

Cosine Similarity: Vector based similarity measure and the commonly used for finding document similarity [Steinbach et al., 2000].

$$\text{similarity} = \cos(\theta) = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| \|\vec{B}\|} \quad (3)$$

Two major advantages of cosine similarity measure are the computational efficiency for highly multidimensional vectors, and document length normalization effect added by division of dot product by Euclidean distance. Due to the suitability of cosine similarity measure for text contents, proposed system also utilize this similarity measure for sequential search queries. Simpler Jaccard similarity measure is used to analyze the returned result snippets. A detail of common similarity measures is already available in chapter 2.

Conceptual Similarities: The presented approach uses a special form of text normalization service to create concept word vectors for similarity analysis. Usually document clustering approaches use the similarity measures on identical terms to find matches, however they do not work very well in case of difference in terminology with similar meaning. In order to minimize this deficiency WordNet thesaurus is used to find most commonly used form of words in a particular sense. An explanation of text normalization is already available in Chapter 3 and 4 (sections: 3.12.1, 4.3.1). Calculation of cosine similarity between concept vectors of source, and result set are made at two phases of analysis; initially at immediate query and matching snippet level and later at complete document level.

In addition to geometric approach a combination of feature and structural based similarity measure is also used organize and filter contents. More discussion on write print analysis is added in later part of chapter.

5.3.6 Used approach

The used analysis algorithm to find matching document cluster is broken into two phases. The first phase uses the contents from input document as search input to the indexing system. The best matches from the sequential search queries generated from document chunks are grouped together. Repeatedly matching documents with high similarity scores are taken as candidate cluster documents for further analysis. The searching mechanism uses a number of similarity detection models and ranking algorithms to retrieve possible match to input query. The most common similarity models for search queries in use include boolean, vector space, probabilistic. In recent information retrieval systems a combination of these techniques are used to facilitate users with best possible ranked output. In the developed prototype

(explained in section 5.4), the analysis module uses a combination of the vector space model and the boolean model to determine how relevant a document in index is to a given query. System also offers possibilities of internet scale similar document detection by means of service calls to general web search engines. The mashup of syndicated search services provide a broader coverage of distributed indexes of deep and shallow web. Once the candidate similar document set is estimated by means of local/web search, the normalized key terms from candidate documents are used for creation of concept vectors. In second analysis phase similarity relations among concept vectors of estimated similar collection are calculated. Documents above a certain user defined threshold value of similarity score among them form a match group. Similarity score is the dot product of concept term vectors of candidate document set being analyzed. Two dimensional ordination is used to visualize the found clusters of document objects.

Effective presentation of perceived results provides the ability to easily identify relevant documents. There are a number of visual and textual formats that can be used to specify links to similar document e.g. document ranked lists, link/association trees, sammon maps, dendro maps [Carey et al., 2003]. Color coded hyper linking is also used to identify possible document matches. Main objective of result visualization paradigms is to give the user an immediate overview of the similarity result. Systematic placement of images on a graphical plan can be less space consuming and more meaning full then hundreds of document titles and matching relationships. User can get an understanding of similarity relations at a glance. In different test scenarios various result representation methods were used. Figure 52 shows few visualization approaches to present similar document found in a repository and its links.

Figure 52. Visualization of similar documents in collection.

The presentation of found similarity relations in document corpus exhibits an ordination that can be used to analyze the clustering approach. The presentational graphs of system are further discussed in section 5.5.

5.4 System Design

The prototype system to detect similar documents in a large heterogeneous collection consists of following two major parts. First one is “Crawling and indexing” and second part is “Retrieval, analysis & presentation”. Figure 53 shows a graphical representation of the system.

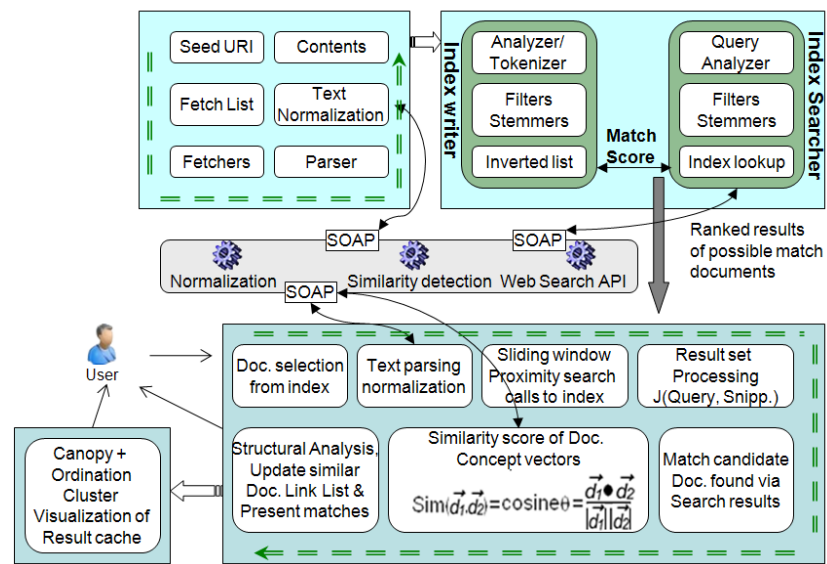


Figure 53. System architecture

Crawling and indexing Subsystem is developed using Apache Lucene API and Nutch [10]. According to user contributed benchmarking and test cases available at selected API's website, used framework supports creation of index up to a couple of million of documents running on a single server. It supports scalability to billions pages in distributed environment. The process of data collection in developed prototype is initiated by pointing to File System, HTTP or FTP seed URI. According to user specified link traversal settings, the fetcher starts gathering data from files. Appropriate parser plug-in is called to extract text and meta information of various document types. The modified content parser plug-ins performs text normalization for the creation of standard and concept index. The contents of documents are normalized to generic language form through text normalization web service. Links to used SOAP services are available at project home page³⁹. The processed contents are passed to indexing writer for addition of document in index.

³⁹ Project Home, Prototype with JUCs dataset. <http://jucs.cpdnet.org/>

The Second subsystem responsible for finding similar documents provides an interface to general user of the system. It allows user to select a document from index and obtain its matching document cluster. This interface also allows administrative users to select web search APIs and input new crawl jobs for index update. The process of finding similar documents for an individual document is started by processing its text in normalized form. A sliding window of fixed word length is imposed over text to get text chunks from document serving as search queries. The generated intermediate candidate result set is scanned for suitable match between query and matching snippet via jaccard coefficient. This eliminates ranking noise added by internal or external indexing engine (PageRank, URL boosting etc.). The top 3 results from each query are aggregated as possible matches for a particular document. The aggregated result set is then scanned for duplicate entries or results with higher match scores. The duplicate possible matches in individual query results of a document are selected as possible candidates for similar document group. In the final conceptual document analysis phase system creates concept term vectors. These term vectors come from document being compared and possible matching document from search results. Vectors are mapped in common vocabulary space of both documents. The DOT product of these vectors gives a cosine similarity score. This score is used as a matching link if it exceeds a certain user defined threshold value. The match scores of compared documents are stored in a document match link list and matching document cluster along with match scores are presented to user. An additional layer of structural similarity analysis is performed to further improve the relevance ranking of documents obtained via content matching. The structural feature analysis includes the word, sentence length distribution comparison, and WritePrint generation. WritePrint constitute a writing style feature matrix based on the use of punctuations, articles, pronouns, conjunctions (a total of 44 characteristics).

The locally indexed documents in experimental setup are batch processed for similarity scores. Simple Canopy Clustering [McCallum et al., 2000] approach is used for group formation, and visualized based on ordination.

The prototype is developed keeping in view the requirements of handling heterogenous and large document collections. The open source and plugin based architecture of Lucene and Nutch allows ease of modification and handling of multiple file types. Initial testing with system includes similarity checks of XML, HTML, MS word, PDF, and MS PowerPoint documents. The corresponding parser plugins were modified to support conversion of fetched text into normalized form. By default nutch (latest nutch 0.9 stable release) performs rather strict boolean AND based multi-term/phrase search, basic query filter is modified to add support of boolean OR based phrase search. The process of detecting similarity is performed in a dynamic manner on incremental index. The indexing and search technology adopted help maintain data for online clustering. The search processing is far more efficient than any conventional database or file based system. Service oriented

computing architecture allows integration of common web search APIs from Google and Microsoft Live.

5.5 Experimental results

In order to test the functionality of system described in previous section, number of experiments were conducted. In the first testing setup, an index consisting of scientific papers from an open access online digital journal⁴⁰ was created. The online archive was used because of the availability of manual categorization information of documents. The documents in library are organized using an extended version of ACM Computing Classification schema. The similarity information of documents obtained through described system is compared to available document grouping under ACM categories. This helps in validating found similarity relations and also help enrich the existing categorization information. Contents of 1174 documents from 164 issues were used in experiments.

<i>Type of Index</i>	<i>No. of Documents</i>	<i>No. of Terms</i>	<i>Size (KB)</i>
Standard	1174	388467	23949
Normalized	1174	366013	23391

Table 5. JUCs index statistics

Table 5 shows some details of generated index. After building the index, each document available in archive is processed for finding similarity scores with documents of index using the technique explained in section 5.4. The similarity scores above a threshold value of 15% were used to plot a linkage mesh of documents. The linkage mesh is a visualization of documents as nodes connected through varying length links which are inversely proportional to the similarity score. Hence nodes plotted closer together are more similar then the nodes plotted farther away. The nodes of similarity mesh are color coded to represent different ACM categories associated with documents and hyperlinks to nodes are created for further study and verification of content similarities. This visualization is generated using Graphviz [Graphviz, 2008] toolkit.

⁴⁰ Journal of Universal Computer Science: <http://www.jucs.org/>

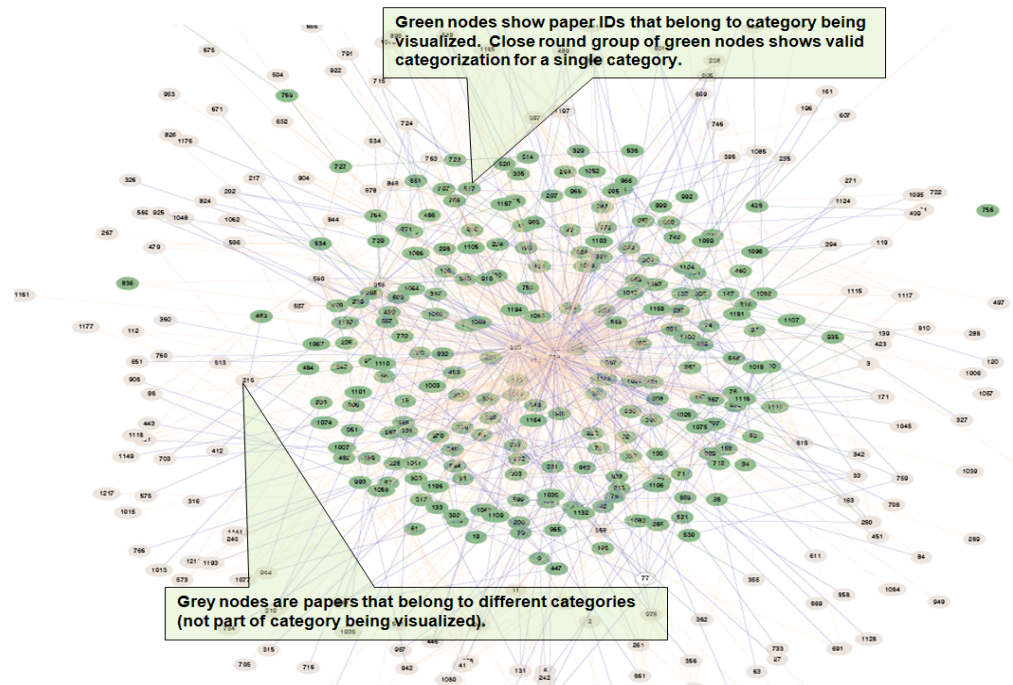


Figure 54. Document similarity links in ACM category D

The graph in figure 54 shows similarity links of papers in ACM category D. The dark colored nodes (green) show papers that are manually classified under same category. The lighter color nodes are documents that are not classified under the same category but some amount of similarity was found through system. The overall picture shows that colored nodes are more tightly grouped together confirming the accordance of manual and automated approach.

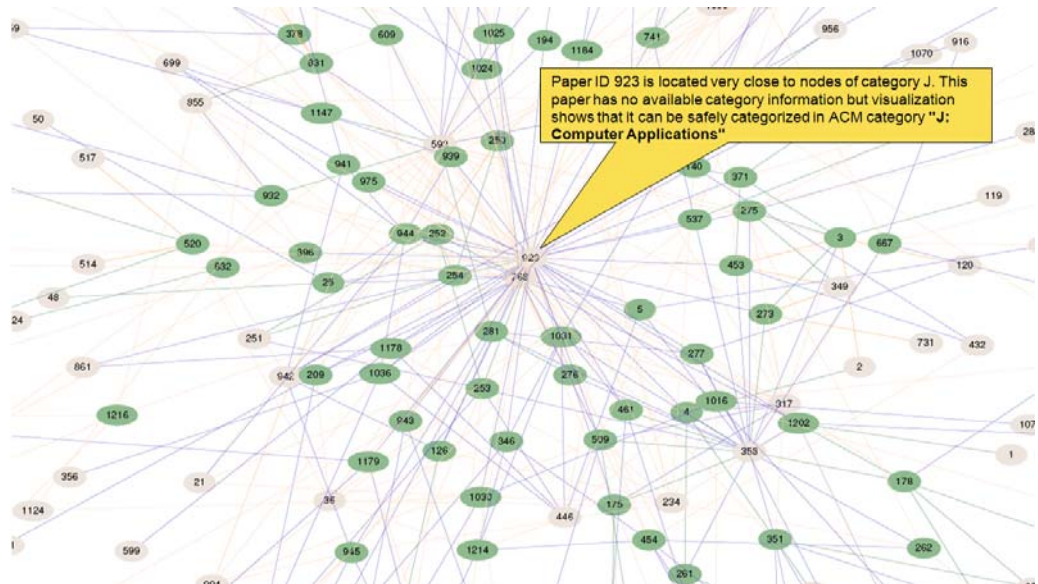


Figure 55. Document similarity links in ACM category J

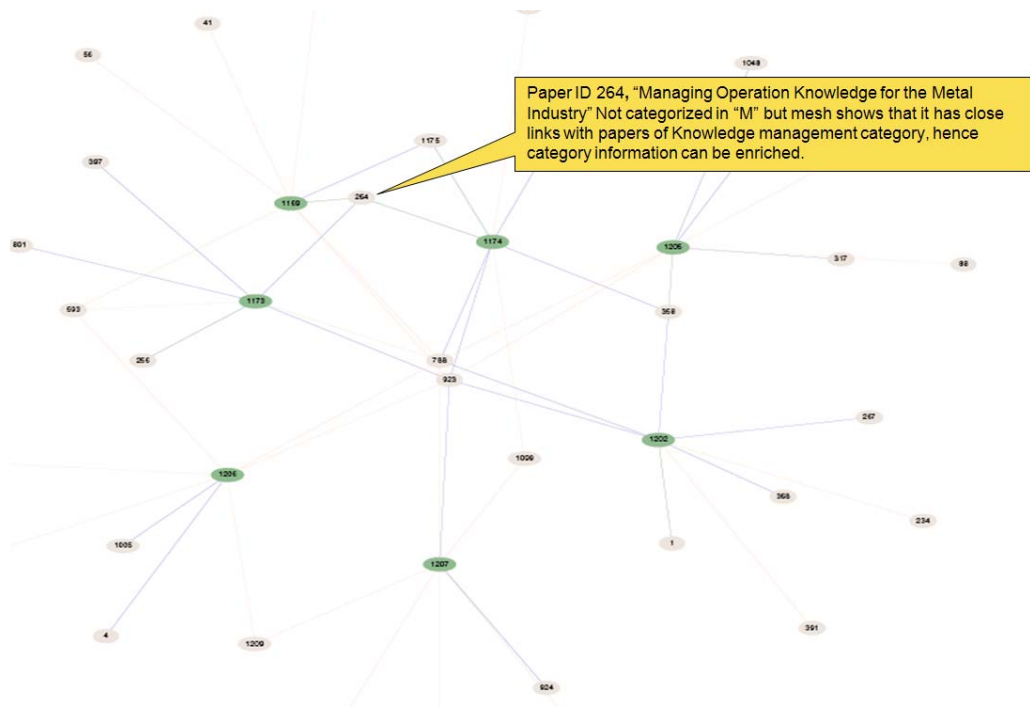


Figure 56. Document similarity links in ACM category M

Figure 55, 56 shows visualization of system's similarity analysis of document categories J and M. It shows an example of enriching categorization information associated with nodes. A node (ID 923) not part of documents category being plotted has very close links with documents of this category; hence make it an ideal candidate for inclusion in the plotted category. Careful analysis of graphs can also help identify anomalies in categorization. An example could be identifying papers that are part of some specific category; however show farther links with related category and have closer links with other categories. High resolution visualizations of various categories are available on project home page (<http://jucs.cpdnet.org>).

Following screen shots of application show the content and structural analysis results produced by the prototype system.

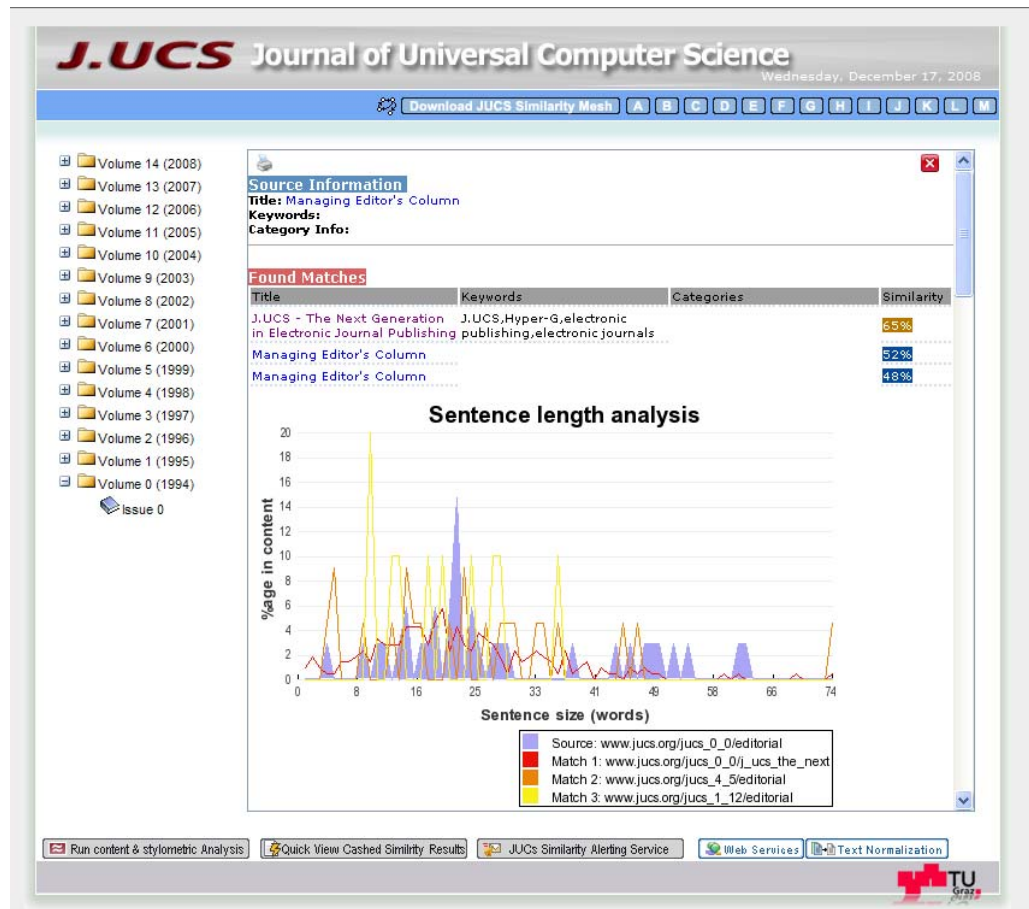


Figure 57. Structural similarity of matching documents

Figure 57 shows the matching documents of a selected article. The structural similarity standard deviation of sentence distribution) of document having highest content match (65%) is lesser then that of other two documents. The higher structural resemblance of other two documents makes them better candidate for relevancy in combined similarity scope.

Figure 58 show the writeprint analysis of another matching set of documents. The comparison of writing features can be used for writer profiling, write style detection for authors from different disciplines or countries.

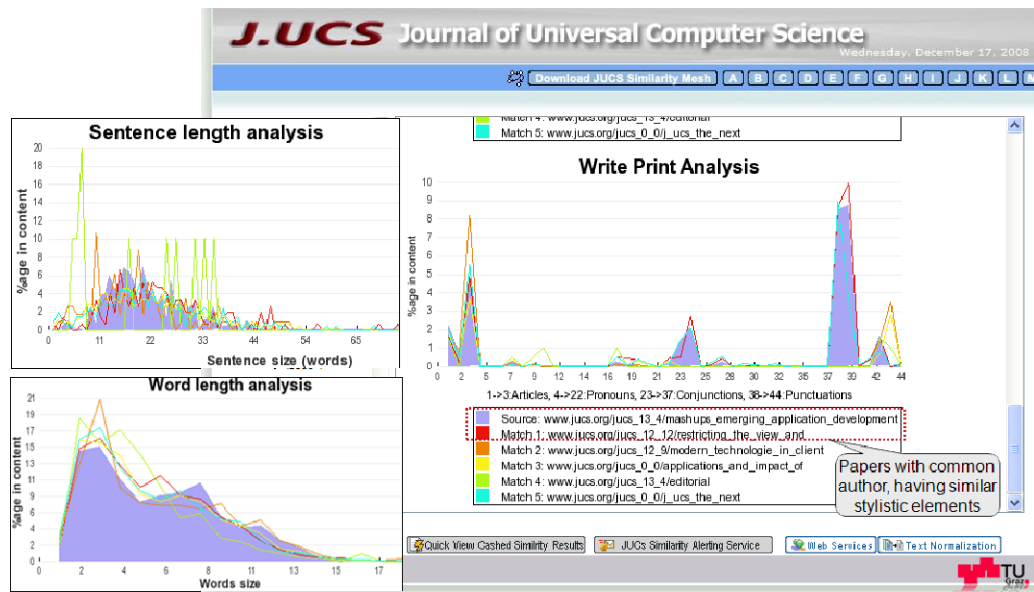


Figure 58. WritePrints taken from <http://jucs.cpdnet.org>

In a broader test environment the system is used for plagiarism detection and prevention (Alerting Service). Multiple deployments of system create a distributed indexing and search environment where searching services are available at every deployed system for local and external indexes. The system performs document similarity checks by using locally stored index (of each deployment), index of peer systems and external internet indexes to produce originality reports. The searching process was extended to internet with the help of SOAP based search API's of Google, Yahoo BOSS and Microsoft Live. User is provided with an interface to upload document for sequential search based similarity comparison, during the process the document becomes part of growing index of system. This application of described method produced results reasonably comparable and in some cases even better than commercially available applications. The ability of maintaining normalized index of large collections and federated search, helps detect documents overlaps even if paraphrasing or word replacements are used. The idea of finding document similarity relations through normalized indexing and search is tested with repositories of scientific document collections. One can observe by looking at index statistics that normalization did not reduce the number of terms or size of index to a greater extent. This is mainly because the used language ontology does not contain the domain specific (computer science mainly) term groups. And the term transformation is restricted to a specific syntactical sense in order to avoid noise in abstraction level. The processes of testing system's response is being extended to larger, more heterogeneous collections covering a number of web servers containing contents of various types, topics and lengths. Such large scale normalized indexing and analysis may require computational power and resources beyond a single computing source.

The future work includes testing of search, normalization, indexing and clustering processes distributed to a number of nodes. This distributed computing will be supported by an internet scale parallel computing model [Weiss, 2007]. The presented system supports an open source platform Hadoop [Hadoop, 2008] for cloud computing. The document similarity detection system can prove to be a good test bed to evaluate cloud computing model, publicized as future of parallel computing running on commodity hardware.

5.6 Contents reusability

Content generation occupies a large chunk of the working time in both the academic and the business area. The content generated is usually either collected in reports or (more often) put together in some kind of presentation for immediate communication of salient topics to students, colleagues, or managers. Especially the last kind of communication is already since long supported by a variety of applications like Microsoft PowerPoint or Open Office Impress that allow for an easy creation of presentations. The content generated generally addresses a specific audience and thus is custom-made: even if the overall content is similar, different audiences may need different information blocks to create or refresh the understanding of a topic in the desired way. The reusability of such information blocks is however not supported in traditional applications.

This is particularly true in the area of learning or training, where a topic can be understood in a number of ways or seen from different angles and the usefulness of reusing blocks for creating new or adjusted slide sets is obvious. Therefore capturing these basic information blocks in so-called ‘learning objects’ (LOs) has spawned a tremendous body of research work discussing how to correctly model courses, learning units, etc. in an abstract way in a variety of levels of granularity ranging from topic level to media level and for different fields and institutions. An especially interesting part of this work deals with the question of reusability (or repurposing) of learning objects. The basic question here is how to annotate learning objects with suitable meta-data for later sensible reuse.

To standardize the meta-data several standards have been presented like the IEEE Learning Object Model [LOM, 2002], the Sharable Content Object Reference Model [SCORM, 2008], or National Education Training Group Learning Object Model (NETg). And already large repositories for learning objects ready for reuse have been built. For instance in the ARIADNE knowledge portal⁴¹ [Duval et al., 2001] tools for annotating and indexing learning objects using IEEE LOM, as well as a federated search over several repositories is offered. But still, all these repositories have to rely on a (mostly manual) annotation of learning objects. However, it has also been described [Cardinaels et al., 2005] that most creators of presentations in fact do not annotate their content properly and already some

⁴¹ <http://www.ariadne-eu.org/>

approaches striving for automatic annotation have been presented like for instance [Cardinaels et al., 2005], where the meta-data of Microsoft Office files is extracted as automatic annotations. Nevertheless, especially for smaller granularities of content units the problem is hard to solve.

Here we do not address the problem of actual meta-data generation, but investigate how to provide a tool that allows users to compare and efficiently repurpose presentations collected in some institution's or company's content repository. The basic idea is to integrate similar presentation in such a way that topically similar parts are interleaved in the target presentation and then can be easily edited by the author. The repurposing of content is thus basically broken down to a few simple steps:

- choosing one or more topics for a presentation (in our system this is done by providing a sample presentation or providing a list of topics containing the relevant topics)
- then automatically integrating all available similar presentations from some repository
- and finally selecting or deselecting the content parts relevant for the intended audience and filling in suitable transitions and additional topics

As a practical use case let us present two typical scenarios for which technical support for such an aggregation of presentations is necessary:

- A university department has a collection of courses on similar topics in a suitable learning repository. There can be similar lectures in different application fields, several versions of a course from previous years, etc. How can the overlap of lectures be assessed and how can new lectures or updated versions efficiently be created reusing available content?
- A business organization has a large repository of slide sets created for different target groups (e.g., product presentations, sales figures, etc.). How can slide-sets aiming at new target audiences be derived in a time-effective manner?

In the following we showcase our approach based on both document- and topic-similarity for the case of Microsoft PowerPoint slide sets and discuss the effectiveness and usability issues of our approach. Our preliminary experiments show that it is indeed possible to extend the reusability of LOs beyond its originally intended usage. We propose a context specific repurposing of content found in institutional archives. The idea is to enhance the reusability of these legacy contents by providing support for the guided discovery of matching material from a content repository. As opposed to [Najjar et al., 2005], which explores the repurposing of LOs based on a domain specific ontology, our approach dynamically extracts

similar LOs based on the implicit similarity measure in usage and structure of the proposed content. We employ a two-phased similarity checking scheme to suggest a set of similar documents and topics, created in the past.

5.7 Aggregation of presentations based on text and topic similarity

In this section we describe the layered approach employed in the extraction of similar content. The user submits a PowerPoint presentation to seek system input on related content that could be repurposed for a particular task. The system then performs document-level similarity checking to present to the user with those documents closely matching some presentation or a planned course plan.

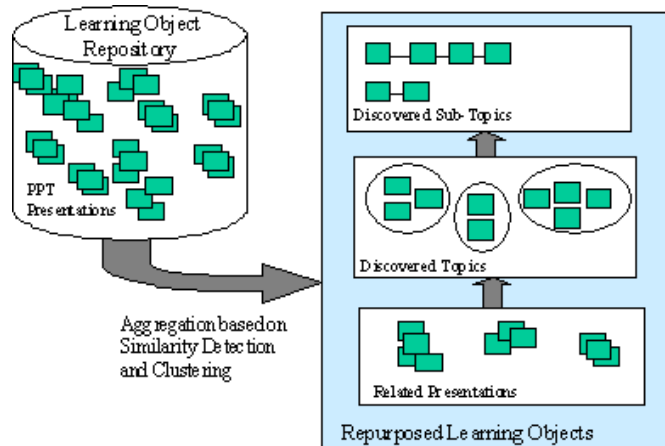


Figure 59. Layered Approach for Repurposing of Learning Objects

This process will result in a set of presentations that best match a source PowerPoint document in a query-by-example fashion. If as result of the similarity check there are no documents currently available with the needed grade of match. The user may either decrease the similarity threshold or make changes in the description of the LOs in the query document.

The system subsequently identifies groups of slides with the highest, content fragment-level similarity within the chosen documents at document level matching. Here we consider each slide as a minimum size content fragment. The order of slides is also taken into consideration in determining structure-level similarity within the documents discovered. As a result we present the user similar content at three levels, document level similarity, similarity at a topic level comprised of overlapping group of slides and sub-topic level similarity based on individual slide similarity. (Fig. 59) illustrates the layered similarity check approach described.

We developed a prototype for performing the actual integration of topically similar presentations. We will now first describe the process of transforming the

PowerPoint presentations in a repository of documents into an internal normalized form. The text from a collection of presentations is first extracted. This includes all text presented on slides together with user notes attached to slides and text from hidden objects maintained by PowerPoint file format. The Apache POI⁴² based parser allows text segmentation based on slide boundaries. At each slide level key terms are identified, the term identification process (contrary to external term extraction used in PINC experiment of chapter 4) is performed locally through removal of common stop, common words. The term importance/weight is determined through occurrence behavior (frequency, and location), and number of times the specific term was tagged in the semantic concordances of WordNet corpus [Miller, 1995]. A word is considered a term with higher weight if it has dominant present in content with less frequent occurrence in common language ontology. The term word vectors can further be reduced to normalized canonical representation using already described normalization service. An inverted index is then built based on term frequencies of normalized root forms. The vector space of the normalized root senses both reduces dimensionality (as document fingerprints) and facilitates the retrieval of concept-level (synonymous) terms. For performing the similarity check, the resolution of the fingerprints used for matching of target documents can be varied to either perform coarse or fine-grained similarity checking. For a document level similarity checking a document is used as fingerprint, whereas a slide is used as fingerprint at the topic level.

5.8 Experiments on the Aggregation of Learning Objects

We have conducted preliminary experiments to illustrate the workings of our prototype system for the repurposing of LOs. A collection of about 350 PowerPoint presentations with an average slide-count of 25 was used throughout our experiments. In our experimental scenario we explored the ability of the system to support the generation of content from repurposed objects. A sample presentation with some topic was presented to the system. For evaluating the capability of the system in repurposing of learning objects, we also carried out the same experiment by using text queries. In the showcase below we used the query terms ‘future of computing’, ‘ubiquitous computing’ and ‘reading and writing’. We will first highlight the results of the document match experiment and will discuss the comparison with text-based queries in the evaluation section. We have also explored a comparison in performance when carrying out a presentation level matching of documents as opposed to a snippet-based matching at the slide level. In our system, the difference is given by the choice of fingerprint resolution. (Fig. 60) shows both the document and slide level similarities between the source and the retrieved target documents found in the repository. The results show that the system has been able to perform an aggregation of contents at both the document and topic-specific levels. The threshold values shown provide an indication of the degree of similarity to facilitate further exploration. (Fig. 61) then demonstrates the mapping

⁴² <http://poi.apache.org/>

between the presentations and the learning units discovered. The tag cloud style representation was used to depict the weights of terms. In our current implementation, the weights are merely based on term frequency. As demonstrated here, the environment is seen to enable a deeper analysis of available contents.

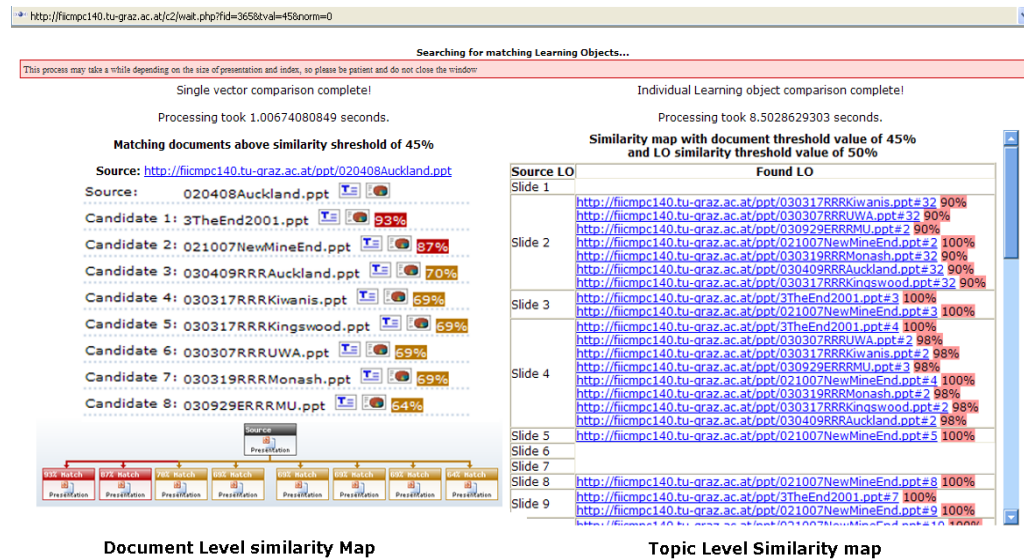


Figure 60. Results of layered similarity checking



Figure 61. Environment for the Discovery of Learning Objects

5.9 Evaluation and usability of the system

Apart from providing a presentation as query document, we also experimented with text string inputs as a query. In this experiment, we also explored various combinations of term sets. As expected, the use of just these terms produced a larger number of text matches as compared to the matching of whole documents. Exact term matching produced no results or few results in most cases.

A non-phrase specific matching was able to produce a large number of results from slides talking about the same area. The concept-level indexing was found to be useful, as the system was able to identify related slides despite the differences in words used in a query (e.g. ‘mobile’ as being synonymous to ‘ubiquitous access’). In this situation however, the user will still have to manually go through each relevant slide and decide individually how and where it can be applied. The proposed approach of allowing the use of PowerPoint documents as query object has thus proven to have immense value in that it provides a great deal of information about the context of work. (Fig. 62) demonstrates the three levels of discovery enabled by the proposed system.

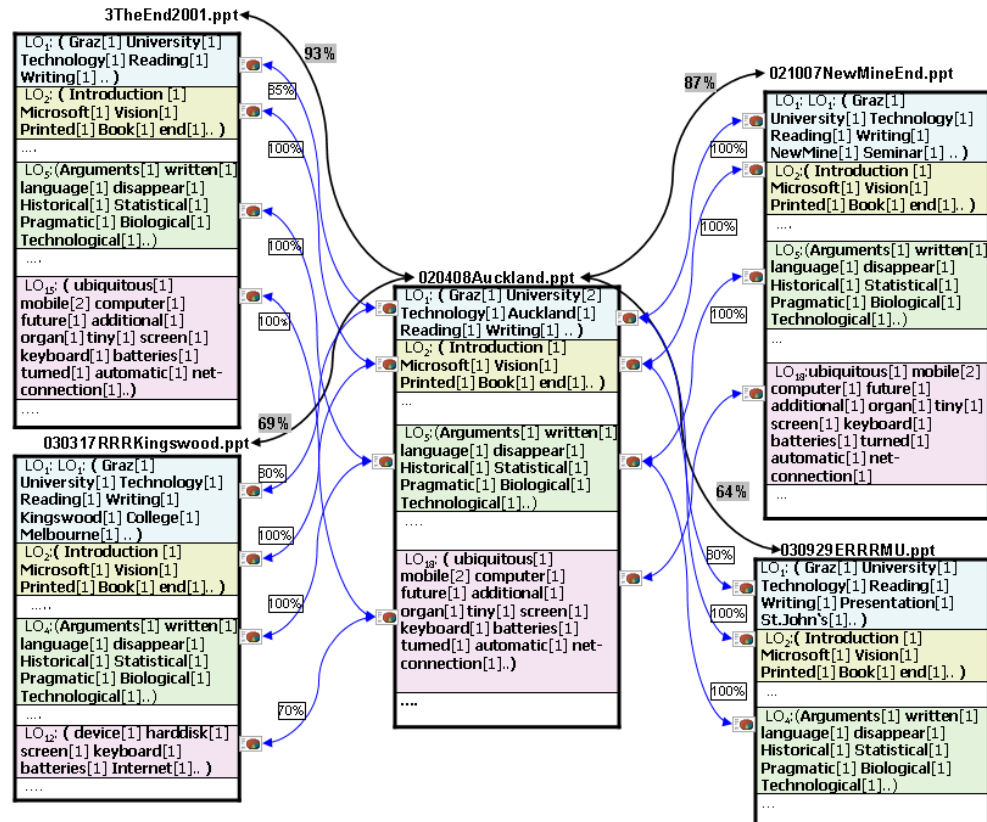


Figure 62. Illustration of the three levels of contents aggregation

There is great benefit that can be gained from a systems-enabled repackaging of content to serve the needs of a user's context of work. In helping an instructor in preparing a lesson plan, as described for scenario 1, it is important to present related learning content initially at a coarse-grained level (based on document-level similarity) with a gradual refinement of similarity results fine-tuned to take into consideration user feedback particular to the current task. The layered approach in presenting relevant content at multiple level of granularity is seen to be promising in assisting the user when generating content.

Our experiments also revealed that a document-level similarity check (based on a larger vector space) required much less time. Our results in figure 60 shows that document-level similarity took only 1 second, while the slide-level checking of the entire collection took 8.6 seconds. This was found to be particularly reflective of the sparse vector space of typical PowerPoint presentations. The abstract level could thus be use (even in a real-time environment) to serve as a first coarse filtering step or to provide a quick overview. Another interesting finding was that there were a number of hidden words associated with presentations stored as internal object, together with slides by PowerPoint, which produced surprising results. There seem to be an internal representation of presentation objects that was still maintained internally, even after an object in a presentation was deleted.

5.10 Conclusions and work in progress

This chapter has presented a layered approach for the aggregation of content and the effective repurposing of learning objects. The use of distance based, and structural feature based similarity measures at different application levels is demonstrated. They include

- Cosine measure among weighted word vectors for index search.
- Jaccard coefficient measure among fingerprint and matching snippet for intermediate result analysis.
- Cosine / dot product measure among normalized document term vectors for document analysis.
- Structural deviation measures for documents and presentations (organization of LO, sentence length, word length, and write print analysis)

The initially described experimental environment shows the applications of compound similarity measures. The test environment shows the use of found similarity relations to enhance existing classification data. The additional layer of similarity analysis is introduced through comparing the stylistic elements. The generation of writeprints, importance of different characteristics (organizations, tone, and word choice etc.) and its use in document relevance detection requires deeper investigation. The proposed system provides a convenient platform to test and evaluate these characteristics.

Preliminary results of second experiments have revealed the benefits and significance of the proposed approach. Dividing the similarity check into 3 layers of similarity, namely document-layer, topic-layer and slide-layer, nicely reflects the granularity of learning objects. By first doing a similarity check on presentation examples, already a large number of unrelated presentations can be ruled out. For the rest we perform a topic similarity check and interleave slides sets from different presentations based on the outcome. Finally, our slide similarity allows finding specific slides that are needed for customizing a presentation with respect to a certain target group. The research presented here also will lead to the provision of support mechanisms for the on-the-job elicitation of metadata. By first acquiring an overview of similar presentations, the user is able to gain quick insights into the relevance of the contents and also regarding the extent of re-usability. This also allows metadata annotation to be performed at both the presentation level as well as the slide level in this case. In other words both context-specific information as well as task-specific information can be acquired from the user and applied as annotations to previously created content.

The discovery of related contents at multiple level of granularity holds the key to the discovery of deeper insights into large document archives. The incorporation of implicit information of content abstraction and structure will further enhance the value of content analysis.

6. Summary and Outlook

Problems, future line of work and concluding remarks

This brief chapter presents an overview of work completed, important findings and lessons learned. It also includes some discussions on limitations and open issues of proposed techniques, and presents the future direction of work.

6.1 Results and Conclusions

This dissertation explores the supplementations in similarity detection processes to support emerging information retrieval and management systems. The discussed work shows a mix of distance, feature and knowledge supported similarity measures in existing and new application domains. The output of this exploratory work can be described by looking at results coming from experiments performed in three application areas.

In first application domain, use of distance based similarity measures at various levels of processing along with structure and language feature comparison is tested to extended currently available plagiarism detection systems. The standard indexing and search systems are supplemented with concept stemming capability through use of common language ontology. The use of normalized terms (generic form of text) in similarity comparison overcomes the problem of paraphrasing and synonymizing to some extent. The additional layer of simplified similarity check (distance based) at intermediate searching stage eliminates the noise generated by common search APIs. This allows the use of general purpose web search APIs for plagiarism specific matching tasks. The ability and flexibility to use a broad range of searching APIs is perhaps the only way to cover maximum portion of the exponentially growing digitized data. Common approaches to detect plagiarism rely on distance based similarity measures alone. In case of unavailability of match of terms, it is not possible to determine the relevance. During the progressive research process use of styling and structural comparison is introduced in plagiarism checking model as an additional layer for match detection. Similarity measures based on the structural deviations in contents and checks for common use of specific language elements are

also found useful for plagiarism detection. A major deficiency found in all plagiarism detection services was inability to handle image plagiarism. An approach using the CBIR is introduced to discover similar images. The blind quality assessment is further used to determine the originality.

In order to test above mentioned enhancements, a platform of loosely coupled web applications was built. This platform allows its users to glue together desired methods of content processing and filtering that use various similarity measures.

In second set of experiments the developed platform allowing user to index and compare contents, was tested for adaptive information delivery services. Two new application areas were evaluated. These are: a multimodal news filtering and delivery system with reusable user interest profiling, and a proactive information supply environment to support knowledge workers. The advantages of conceptual similarity checks in content and collaborative news filtering systems were also demonstrated. The concepts extracted from syndicated contents not only provide a better search and filtering mechanism needed for multimodal interfaces, but also help build reusable user profiles. The mashup of search services formed for plagiarism detection was applied for assisted information discovery. The layered similarity checks were put to use for developing a pull based information supply environment. Such an application assists knowledge workers with context aware automated information discovery.

In third application area distance based similarity measures are combined with structural, language feature and geometric characteristics for content organization, reusability and filtering. In conducted experiments the document similarity analysis is put to use for enriching the classification data of digital libraries. Use and importance of structure and stylistic elements is exemplified by adding it as an additional similarity filter. Besides the use of language knowledge in distance based similarity checks, the use of language information is shown to construct the document feature space (write prints). This information can be used for writer profiling and intrinsic similarity checks.

The experiments conducted during this work also show how an evolved service oriented computing architecture minimizes the added burdens of proposed layered similarity operations and language processing. Although prototype applications were not deployed on a greater number of nodes to determine the true potentials of cloud computing in our case studies, however we did some experiments to verify the positive effects of this approach. Figure 63 shows some statistics from CPDNet's 2 node cloud. It shows the indexing operations done on different data collections, the graph shows a comparison of standard indexing and a language normalized indexing done using a single machine (Master node with 3 GHz PIV processor and 1 GB RAM) and with an added slave machine (a PIII 866MHz system with 786MB of RAM). The results show how a computationally expensive natural language processing effects the performance in terms of computation time and how an addition of distributed computing element minimize this effect

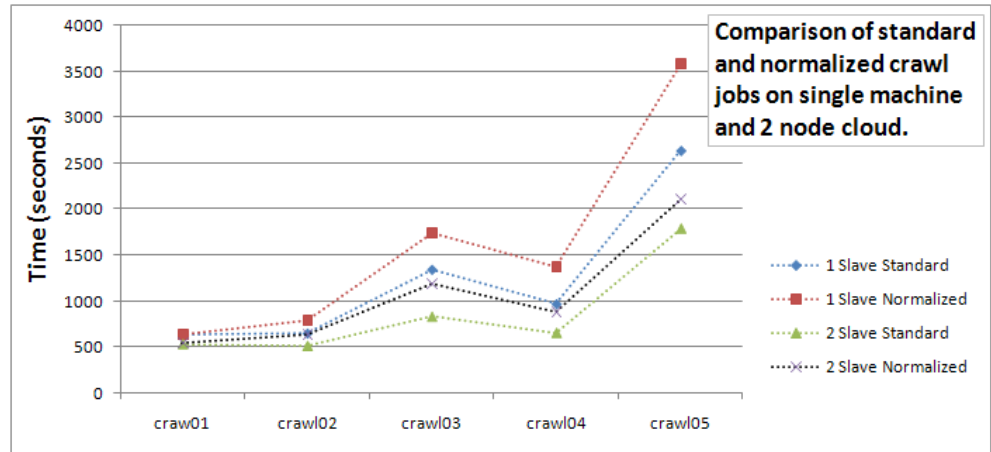


Figure 63. Effects of cloud computing in prototype system

The red line shows the increased amount of time required to complete language normalized indexing jobs. The black line shows the amount of time taken to complete the same job in cloud. The performance is more significant in cases of crawls containing more data.

6.2 Future Perspectives

The existing CPDNet standard and generic text index contains documents of JUCs digital library, ed-media conference paper archive, random ripping from open libraries containing research articles (approx. 15 K documents). They are indexed on a single commodity hardware based server. However there is a need to test the efficiency of generic index generation at larger scale. The possibility of distributed computing support available in parent indexing API can be exploited to increase computational expensive generic indexing and search tasks. The absence of cross language similarity checks needs to be addressed. Commonly available machine translation of text produces an output which is, in most cases, not comparable to humanly generated text for a specific language. However the translation of normalized form of text may provide better possibilities for comparison. The use of different language ontologies (EuroNet?) for text normalization and use of translation services for cross language comparison needs further investigation.

The experimental data for plagiarism detection in visual contents comes from copied textures of virtual world. However same approach can be used to complement academic plagiarism detection applications. There is a need to revamp document crawlers/parsers of existing plagiarism systems, allowing image extraction from common document formats. The visual descriptors extracted from images can then be stored along with the conventional term index of respective documents. In reported work we used the Lucene based CBIR library, that allows convenient possibility of extending our text indexes with associated image feature

space. Further work is underway to build CPDNet's custom PDF parser for image extraction and indexing.

Existing intrinsic document style checks or structural checks may provide a trivial estimation of authorship proof or document match however deeper and more extensive test are required to build a comprehensive writeprint feature space.

I am determined to put my work to practical use, and intend to launch CPDNet as an open service. Higher Education Commission and individual universities in Pakistan are paying a sizeable amount of money for such tools and services. The common students, researchers and even teachers have very less or little knowledge about the plagiarism detection processes. No easy and common access to such systems exist thus resulting in duplication of work that may cause embarrassment to students, teachers and universities at later stage. I believe common availability of such systems will give an alternative to paid services. It will improve knowledge about intellectual property rights in fresh students and researchers, and aid them tremendously in their research activities. The use of information supply feature during the article write-up phase will maximize the use of digital libraries as relevant documents will be made available to users without rigorous search.

LIST OF FIGURES

FIGURE 1.	OVERVIEW OF DISSERTATION	16
FIGURE 2.	VECTOR SPACE MODEL.....	26
FIGURE 3.	DOCUMENT INFERENCE NETWORK [TURTLE AND CROFT, 1990].....	27
FIGURE 4.	TAKEN FROM COURSE INFORMATION PAGE, CGV, TU GRAZ	36
FIGURE 5.	TAKEN FROM TEACHING INFORMATION PAGE, IAIK TU GRAZ	36
FIGURE 6.	TAKEN FROM SEMINAR PROJECT CONTENTS BY ELISABETH OSWALD, IAIK TU GRAZ	37
FIGURE 7.	PLAGIARISM DETECTION WITH DOCUMENT SOURCE COMPARISON	40
FIGURE 8.	STYLOMETRIC TEST, GLATT PLAGIARISM SELF-DETECTION PROGRAM.....	43
FIGURE 9.	STYLOMETRIC TEST RESULTS.....	43
FIGURE 10.	TURNITIN, INSTRUCTOR VIEW OF ASSIGNMENT INBOX	44
FIGURE 11.	TURNITIN, ORIGINALITY REPORT OF A SUBMISSION	45
FIGURE 12.	MYDROPBOX, PAPER INFORMATION REPORT	46
FIGURE 13.	DOCOLOC, START PAGE AND DETECTION PREFERENCE SETTINGS	47
FIGURE 14.	DOCOLOC, SECTIONS OF TEST REPORT	48
FIGURE 15.	ORIGINALITY REPORT BY FIRST SERVICE.....	51
FIGURE 16.	ORIGINALITY REPORT BY SECOND SERVICE	52
FIGURE 17.	SYSTEM SHOWING 91% MATCH FOR A PARTICULAR PAPER.....	52
FIGURE 18.	REPORT SHOWING HIGH PERCENTAGE OF MATCH FROM A SINGLE SOURCE.....	53
FIGURE 19.	MORE MEANING FULL REPORT AFTER EXCLUDING THE HIGH PERCENTAGE SOURCE	53
FIGURE 20.	ORIGINAL TABULAR DATA WITH TEXT CONTAINING SPECIAL CHARACTERS.....	54
FIGURE 21.	REPORT WITH BROKEN TABLE CELL TEXT	55
FIGURE 22.	DOCUMENT REPORT WITH SPECIAL CHARACTERS	55
FIGURE 23.	USE OF PLAGIARISM DETECTION TOOLS TO DISCOVER COPIES OF OWN WRITINGS	56
FIGURE 24.	REPORT WITH LINKS SHOWING COPIED PORTION OF TEXT	56
FIGURE 25.	STATISTICAL TEXT STRUCTURE ANALYSIS	62
FIGURE 26.	COLLABORATIVE PLAGIARISM DETECTION NETWORK OVERVIEW.....	69
FIGURE 27.	WEB SERVICE FLOW IN CPDNET	74
FIGURE 28.	PROCESS OF NORMALIZATION OF TEXT	75
FIGURE 29.	COMPARISON OF SEARCH AND SIMILARITY DETECTION CAPABILITIES.....	76
FIGURE 30.	COMPARISON WITH STUDENT PAPERS	77
FIGURE 31.	COMPARISON OF PAPERS NOT ACCESSIBLE ON THE WEB WITHOUT CHARGE	77
FIGURE 32.	ALERTING SERVICE INTERFACE ADDED TO JUCs CPDNET NODE	78
FIGURE 33.	TEXTURE DISTRIBUTION IN TEST CORPUS.....	86
FIGURE 34.	ARCHITECTURE OF PLAGIARISM DETECTION SYSTEM	87
FIGURE 35.	DISTRIBUTION OF SIMILAR IMAGES WITH DIFFERENT UUIDS.....	89
FIGURE 36.	IDENTIFICATION OF POSSIBLE THEFT CASES IN A SEGMENT OF OBJECTS	89
FIGURE 37.	SIMILARITY AND ORIGINALITY COMPUTATION	90
FIGURE 38.	DWT COEFFICIENT PROBABILITY DISTRIBUTION OF ORIGINAL AND COPIED IMAGE	91
FIGURE 39.	DETAILED PROBABILITY DISTRIBUTION OF WAVELET COEFFICIENTS.....	91
FIGURE 40.	NORMALIZATION OF TEXT TO FIND CONCEPTUAL SIMILARITIES	102
FIGURE 41.	INFORMATION PRE-PROCESSING	103
FIGURE 42.	USER MODEL AND PERSONALIZATION	103
FIGURE 43.	HTTP AND SPEECH ACCESS.....	106
FIGURE 44.	VXML NEWS SNIPPET	107
FIGURE 45.	E-INK AND VIDEO ACCESS	108
FIGURE 46.	PROCESS OF NORMALIZATION AND CONCEPT VECTOR GENERATION	109
FIGURE 47.	PINC ARCHITECTURE	111
FIGURE 48.	INFORMATION SUPPLY MODEL FOR KNOWLEDGE WORKERS	120

FIGURE 49.	GOOGLE TREND FOR CLUSTER, GRID AND CLOUD COMPUTING	122
FIGURE 50.	ORCHESTRATION VS. CHOREOGRAPHY	124
FIGURE 51.	PROCESSING STEPS IN DOCUMENT CLUSTERING.....	130
FIGURE 52.	VISUALIZATION OF SIMILAR DOCUMENTS IN COLLECTION.	134
FIGURE 53.	SYSTEM ARCHITECTURE.....	135
FIGURE 54.	DOCUMENT SIMILARITY LINKS IN ACM CATEGORY D	138
FIGURE 55.	DOCUMENT SIMILARITY LINKS IN ACM CATEGORY J.....	138
FIGURE 56.	DOCUMENT SIMILARITY LINKS IN ACM CATEGORY M	139
FIGURE 57.	STRUCTURAL SIMILARITY OF MATCHING DOCUMENTS.....	140
FIGURE 58.	WRITEPRINTS TAKEN FROM HTTP://JUCS.CPDNET.ORG	141
FIGURE 59.	LAYERED APPROACH FOR REPURPOSING OF LEARNING OBJECTS	144
FIGURE 60.	RESULTS OF LAYERED SIMILARITY CHECKING	146
FIGURE 61.	ENVIRONMENT FOR THE DISCOVERY OF LEARNING OBJECTS.....	146
FIGURE 62.	ILLUSTRATION OF THE THREE LEVELS OF CONTENTS AGGREGATION	147
FIGURE 63.	EFFECTS OF CLOUD COMPUTING IN PROTOTYPE SYSTEM.....	153

BIBLIOGRAPHY

- [Abel et al., 2005] Abel, F., Baumgartner, R., Brooks, A., Enzi, C., Gottlob, G., Henze, N., Herzog, M., Kriesell, M., Nejd, W., and Tomaschewski, K. "The Personal Publication Reader" Semantic Web Challenge, 4th International Semantic Web Conference, Galway Ireland Nov. 2005.
- [Adomavicius and Tuzhilin, 2005] Adomavicius, G., and Tuzhilin, A. "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions". IEEE Trans. on Knowl. and Data Eng. Volume 17, Issue 6, Jun. 2005, Pages 734-749. DOI= <http://dx.doi.org/10.1109/TKDE.2005.99>
- [Baeza-Yates and Navarro, 1996] Baeza-Yates, R., and Navarro, G. "Integrating contents and structure in text retrieval". SIGMOD Rec. 25, 1 (Mar. 1996), 67-79. DOI= <http://doi.acm.org/10.1145/381854.381890>
- [Baily, 2008] Bailey, J. "Content Theft and Second Life", blog Mar 12th, 2008, Retrieved November 20, 2008, from <http://www.plagiarismtoday.com/2008/03/12/content-theft-and-second-life/>
- [Band, 2006] Band, J., "The Google Library Project: Both Sides of the Story", Plagiarism: Cross-Disciplinary Studies in Plagiarism, Fabrication, and Falsification, 1 (2): 1-17. 2006
- [Barland and Saadane, 2005] Barland, R., and Saadane, A. "Reference free quality metric for JPEG-2000 compressed images" In Proceedings of ISSPA, 2005, Sydney, Australia
- [Beasley 2006] Beasley, J. D. "The Impact of Technology on Plagiarism Prevention and Detection" Plagiarism: Prevention, Practice and Policies 2004 Conference.
- [Beil et al., 2002] Beil, F., Ester, M., and Xu, X. „Frequent term-based text clustering" In Proceedings of the Eighth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, July 23 - 26, 2002. KDD '02. ACM, New York, 436-442. DOI= <http://doi.acm.org/10.1145/775047.775110>
- [Beitzel et al. 2004] Beitzel, S., M., Jensen, E., C., Chowdhury, A., and Grossman, D. "Hourly Analysis of a Very Large Topically Categorized Web Query Log", In Proceedings of 2004 ACM Conf. on Research and Development in Information Retrieval (SIGIR-2004), Sheffield, UK, July 2004.
- [Bennet et al. 2000] Bennett, K., Layzell, P., Budgen, D., Brereton, P., Macaulay, L., and Munro, M. "Service-based software: the future for flexible software" In Proceedings of the Seventh Asia-Pacific Software Engineering Conference (December 05 - 08, 2000). APSEC. IEEE Computer Society, Washington, DC, 214.
- [Bergman, 2001] Bergman, M., K. "The deep web: Surfacing hidden value" The Journal of Electronic Publishing. Michigan University Press. July 2001; Online at <http://www.press.umich.edu/jep/07-01/bergman.html> (Accessed April 07, 2007)

[Berkeley 2006] Berkeley University of California, Student Conduct, Sanctions, <http://students.berkeley.edu/osl/sja.asp?id=1004> visited: 22 July 2006

[Billsus and Pazzani, 2007] Billsus, B., and Pazzani, M., J. "Adaptive News Access" *The Adaptive Web* ISSN 0302-9743 (Print) 1611-3349 (Online) Volume 4321/2007 DOI 10.1007/978-3-540-72079-9 Pages 550-570

[Boreczky and Wilcox, 1998] Boreczky, J., S., and Wilcox, L., D. "A hidden Markov model framework for video segmentation using audio and image features" In *Proceedings of Int. Conf. Acoustics, Speech, and Signal Proc.*, 6, Seattle, 1998, pp. 3741-3744.

[Broder, 2006] Broder, A. "Search without a Box" (Interview). Yahoo! Search Blog. March 09, 2006; <http://www.ysearchblog.com/archives/000262.html> (Accessed April 2007)

[Bunke, 2000] Bunke, H. "Graph matching : Theoretical foundations, algorithms, and applications" In *Proceedings of Vision Interface 2000*, Montreal, pages 82–88, 2000.

[Burke, 2002] Burke, R. "Hybrid Recommender Systems: Survey and Experiments" *User Modeling and User-Adapted Interaction* Vol. 12, Issue 4, Nov. 2002, pages 331-370. DOI= <http://dx.doi.org/10.1023/A:1021240730564>

[CAI, 2005] The Center for Academic Integrity's Assessment Project Research survey by McCabe, D. http://www.academicintegrity.org/cai_research.asp visited: 22 July 2006

[Cambridge, 2006] University of Cambridge, Procedure for dealing with cases of suspected plagiarism, visited: 22 July 2006, <http://www.admin.cam.ac.uk/offices/gradstud/committees/plagiarism/procedure.html>

[Canny, 1986] Canny, J. "A computational approach to edge detection" *IEEE transactions on Pattern Analysis and Machine Intelligence*, archive Volume 8, Issue 6, November 1986 ISSN:0162-8828

[Cardinaels et al., 2005] Cardinaels, K., Meire, M., and Duval, E. "Automating Metadata Generation: the Simple Indexing Interface" In *proceedings of International World Wide Web Conference*, 2005, Chiba, Japan 548 - 556

[Carey et al., 2003] Carey, M., Heesch, D., and Rüger, S. "Info Navigator: A visualization tool for document searching and browsing" In *proceedings of Int'l Conf on Distributed Multimedia Systems, DMS*, Florida, Sept 2003, pp 23-28

[Chen and Chen, 2003] Chen, S., J., and Chen, S., M. "Fuzzy risk analysis based on similarity measures of generalized fuzzy numbers" *IEEE Trans. on Fuzzy Systems* 11(1), 45–56 (2003)

[Chester, 2001] Chester, G., "Final Report on the JISC Electronic Plagiarism Detection Project" Joint Information Systems Committee, August 2001, Retrieved November 20, 2008, from http://www.jisc.ac.uk/uploaded_documents/plagiarism_final.pdf

[Chowdhury and Landoni, 2006] Chowdhury, S., and Landoni, M. "News aggregator services: user expectations and experience" Online Information Review, Volume: 30, Issue: 2, 2006, pages 100-115.

[CNN, 2004] CNN COURTTV; Student wins battle against plagiarism detection requirement, January 2004, <http://www.cnn.com/2004/LAW/01/21/ctv.plagiarism/> visited: 22 July 2006

[CopyCatch, 2006] CopyCatch product website, <http://www.copycatchgold.com/> visited: 22 July 2006

[Coyle, 2002] Coyle, F., P. "XML, web services and the changing face of distributed computing". Ubiquity Volume 3, Issue 10 (Apr. 2002), 2. DOI= <http://doi.acm.org/10.1145/763747.763749>

[CPDNet, 2008] Collaborative Plagiarism Detection Network (2008), Retrieved March 10, 2008, from <http://www.cpdnet.org/>

[CrossRef, 2007] Crossref Search: Online at <http://www.crossref.org/crossrefsearch.html> (Accessed April 04, 2007)

[DMCA, 2008] Digital Millennium Copyright Act in Second Life, Retrieved November 20, 2008, from <http://secondlife.com/corporate/dmca.php>

[Djeraba, 2002] Djeraba, C. "Guest Editor's Introduction: Content-Based Multimedia Indexing and Retrieval," IEEE MultiMedia, vol. 9, no. 2, pp. 18-22, Apr-Jun, 2002

[Dreher and Williams 2006] Heinz, D., and Williams, R. "Assisted Query Formulation Using Normalised Word Vector and Dynamic Ontological Filtering" Flexible Query Answering Systems: 7th International Conference, FQAS 2006, Milan, Italy, June 7-10, 2006 pp. 282 – 294

[Dumais et al., 1988] Dumais, S., T., Furnas, G., W., Landauer, T., K., Deerwester, S., and Harshman, R. "Using latent semantic analysis to improve access to textual information" In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Washington, D.C., United States, May 15 - 19, 1988). J. J. O'Hare, Ed. CHI '88. ACM, New York, NY, 281-285. DOI= <http://doi.acm.org/10.1145/57167.57214>

[Dunn and Higgins, 1995] Dunn, D., and Higgins, W., E. "Optimal Gabor filters for texture segmentation", Dept. of Comput. Sci. & Eng., Pennsylvania State Univ., University Park, PA; Image Processing, IEEE Transactions on Publication, Jul 1995

[duPlessis and Li, 2004] duPlessis, R., and Li, X. "Cross-Media Ownership and Its Effect on Technological Convergence of Online News Content---A Content Analysis

of 100 Internet Newspapers" Paper presented at the annual meeting of the International Communication Association, New Orleans Sheraton, New Orleans, LA, May 2004, http://www.allacademic.com/meta/p113386_index.html

[Duval et al., 2001] Duval, E., Forte, E., Cardinaels, K., Verhoeven, B., Durm, V., R., Hendriks, K., Wentland-Forte, M., Ebel, N., Macowicz, M., Warkentyne, K., and Haenni, F. "The ARIADNE Knowledge Pool System" *Communications of the ACM* 44 (5), 73-78.

[Eissen and Stein 2006] Eissen, S., M. zu, and Stein, B. "Intrinsic plagiarism detection" In M. Lalmas, A. MacFarlane, S. M. Rüger, A. Tombros, T. Tsirikla, and A. Yavlinsky, editors, *Proceedings of the European Conference on Information Retrieval (ECIR 2006)*, volume 3936 of *Lecture Notes in Computer Science*, pages 565-569. Springer, 2006.

[EVE, 2006] EVE Plagiarism Detection System website, <http://www.canexus.com/eve/> visited: 22 July 2006

[Farrington, 1996] Farrington J., M., with contributions by Morton, A., Q., Farrington, M., G., and Baker, M., D. "Analysing for Authorship: A Guide to the Cusum Technique" University of Wales Press, Cardiff, 1996. ISBN 0-7083-1324-8

[Feldman, 2006] Feldman, S. "The Hidden Costs of Information Work" (IDC #201334, April 2006) <http://www.idc.com/getdoc.jsp?containerId=201334> (Accessed April 2007).

[Fernández et al., 2005] Fernández, L., S., García, N., F., Bernardi, A., Zapf, L., Peñas, A., and Fuentes, M. "An experience with Semantic Web technologies in the news domain", *Workshop on Semantic Web Case Studies and Best Practices for eBusiness*, Ireland, Nov. 2005.

[Fielding and Taylor, 2002] Fielding, R., T., and Taylor, R., N. "Principled Design of the Modern Web Architecture" *ACM Transactions on Internet Technology (TOIT)* Pp. 115–150, DOI:10.1145/514183.514185, ISSN 1533-5399

[GIFT, 2008] GIFT (the GNU Image-Finding Tool) Retrieved November 20, 2008, from <http://www.gnu.org/software/gift/>

[Glatt, 2006] Glatt Plagiarism Services website, <http://www.plagiarism.com>, visited: 22 July 2006

[Graphviz, 2008] Graphviz, Graph Visualization Software, Accessed July 5, 2008 <http://www.graphviz.org/>

[Hadoop, 2008] Hadoop Platform, Retrieved March 10, 2008, from <http://hadoop.apache.org/core/>

[HEC Press, 2006] Higher Education Commission Pakistan Press release (7 Feb. 2006), http://www.hec.gov.pk/htmls/press_release/2006/Feb/feb_6.htm and (10 May

2006), http://www.hec.gov.pk/htmls/press_release/May_06/May-10.htm visited: 22 July 2006

[Heinrich and Maurer, 2000] Heinrich, E., Maurer, H. "Active Documents: Concept, Implementation and Applications" J.UCS 6, 12 (2000), 1197-1202

[Hogg et al., 2004] Hogg, K., Chilcott, P., Nolan, M., and Srinivasan, B. "An evaluation of Web services in the design of a B2B application" In Proceedings of the 27th Australasian Conference on Computer Science - Volume 26 (Dunedin, New Zealand). Estivill-Castro, Ed. ACM International Conference Proceeding Series, vol. 56. Australian Computer Society, Darlinghurst, Australia, 331-340.

[Hölscher and Strube, 2000] Hölscher, C., and Strube, G. "Web search behavior of internet experts and newbies" Computer Networks, Vol. 33, Issues 1-6, June 2000, Pages 337-346.

[Holtman and Mutz, 1998] Holtman, K., and Mutz, A. "RFC 2295, Transparent Content Negotiation in HTTP" IETF Draft March 1998; <http://tools.ietf.org/html/rfc2295>

[Hume, 1748] Hume, D. "An enquiry concerning human understanding", Harvard Classics Volume 37 Copyright 1910 P.F. Collier & Son originally published in 1748.

[IBM:Term, 2008] IBM Terminology, Retrieved Dec 11, 2008, from <http://www-01.ibm.com/software/globalization/terminology/gh.jsp>

[imgSeek, 2008] imgSeek Content Based Search, Retrieved November 20, 2008, from <http://www.imgseek.net/>

[iParadigm, 2006] iParadigms anti plagiarism product website, <http://www.plagiarism.org/> visited: 22 July 2006

[IPR overview, 2006] Intellectual Property Rights: Overview, March 2006, <http://www.jisclegal.ac.uk/pdfs/IPROverview.pdf> visited: 22 July 2006

[Iyer & Singh 2005] Iyer, P., and Singh, A. "Document Similarity Analysis for a Plagiarism Detection Systems" 2nd Indian International Conference on Artificial Intelligence (IICAI –2005), pp. 2534-2544

[Jain et al., 1999] Jain, A., K., Murty, M., N., and Flynn, P. J. "Data clustering: a review" ACM Computing Surveys 31, 3 September 1999, 264-323. DOI= <http://doi.acm.org/10.1145/331499.331504>

[JISC, 2006] Joint Information Systems Committee (JISC) plagiarism Advisory Program website, <http://www.jiscpas.ac.uk>, visited: 22 July 2006

[Jones, 2008] Jones B. "An EGEE Comparative Study: Grids and Clouds Evolution or revolution", Report comparing EGEE grid to Amazon Web Services, Dated 11/06/2008, EDMS Id 925013, Ret. from <https://edms.cern.ch/document/925013/>

[JPlag, 2006] JPlag website, <https://www.ipd.uni-karlsruhe.de/jplag/> visited: 22 July 2006

[Kan et al., 2006] Kan, L., K., Peng, X., and King, I. "A user profile-based approach for personal information access: shaping your information portfolio" In Proceedings of the 15th international Conference on World Wide Web (Edinburgh, Scotland, May 23 - 26, 2006). WWW '06. ACM, New York, NY, 921-922. DOI= <http://doi.acm.org/10.1145/1135777.1135945>

[Kraft et al., 1998] Kraft, D., H., Bordogna, G., and Pasi, G. "Information Retrieval Systems: Where is the Fuzz? " Presentation at the IEEE International Conference on Fuzzy Systems, Anchorage, Alaska

[Kraft et al., 2006] Kraft, D., H., Pasi, G., and Bordogna, G. "Vagueness and uncertainty in information retrieval: how can fuzzy sets help?" In Proceedings of the International Workshop on Research Issues in Digital Libraries (Kolkata, India, December 12 - 15, 2006). P. Majumder, M. Mitra, and S. K. Parui, Eds. IWRIDL '06. ACM, New York, NY, 1-10. DOI= <http://doi.acm.org/10.1145/1364742.1364746>

[Klausegger et al., 2007] Klausegger, C., Sinkovics, R., R., and Zou H., J. "Information overload: a cross-national investigation of influence factors and effects" Marketing Intelligence & Planning ISSN: 0263-4503 Year: 2007 Volume: 25, Issue: 7 Page: 691 - 718 DOI: 10.1108/02634500710834179

[Kulathuramaiyer and Balke, 06] Kulathuramaiyer, N., and Balke, W., T. "Restricting the View and Connecting the Dots – Dangers of a Web Search Engine Monopoly" Journal of Universal Computer Science, vol. 12, no. 12 (2006), 1731-1740

[Larson et al., 2003] Larson, J., A., Raman, T., V., and Raggett, D. "W3C Multimodal Interaction Framework", W3C NOTE, 06 May 2003, <http://www.w3.org/TR/2003/NOTE-mmi-framework-20030506/>

[Levin, 1993] Levin. "English Verb Classes and Alternations. A Preliminary Investigation" University of Chicago Press 1993.

[Linden, 2008] Linden, H. "IBM and Linden Lab Interoperability Announcement", blog July 8th, 2008, Retrieved November 20, 2008, from <http://blog.secondlife.com/2008/07/08/ibm-linden-lab-interoperability-announcement/>

[LIRE, 2008] LIRE (Lucene Image REtrieval) Retrieved November 20, 2008, from <http://www.semanticmetadata.net/lire/>

[LOM, 2002] IEEE: Standard for learning object metadata (2002) Sponsored by the Learning Technology Standards Committee of the IEEE, Retrieved November 28, 2008, from <http://ieeeltsc.org/>

[LSA, 2006] Latent Semantic Analysis, web site at University of Colorado, <http://lsa.colorado.edu/> visited: 22 July 2006

[Lucene, 2008] Lucene, The Apache Software Foundation, (2008), Retrieved March 10, 2008, from <http://lucene.apache.org>

[Lux and Chatzichristofis, 2008] Lux, M., and Chatzichristofis, S., A. "Lire: lucene image retrieval: an extensible java CBIR library", In Proceeding of the 16th ACM international Conference on Multimedia (Vancouver, British Columbia, Canada, October 26 - 31, 2008). MM '08. ACM, New York, NY, 1085-1088. DOI=<http://doi.acm.org/10.1145/1459359.1459577>

[Ma et al., 2004] Ma, Q., Nadamoto, A., and Tanaka, K. "Complementary information retrieval for cross-media news content" In Proceedings of the 2nd ACM international Workshop on Multimedia Databases, Washington, DC, USA, November, 2004. MMDDB '04. ACM, New York, NY, Pages 45-54. DOI=<http://doi.acm.org/10.1145/1032604.1032613>

[Manber, 1994] Manber, U. "Finding similar files in a large file system" Winter USENIX Technical Conference 1994, San Francisco, CA, USA

[Martin and Eklund, 2000] Martin, P., and Eklund, P., W. "Knowledge Retrieval and the World Wide Web" IEEE Intelligent Systems 15, 3 (May. 2000), 18-25. DOI=<http://dx.doi.org/10.1109/5254.846281>

[Marziliano et al., 2002] Marziliano, P., Dufaux, F., Winkler, S., and Ebrahimi, T. "A no-reference perceptual blur metric" In Proceedings of International Conference on Image Processing, 3, pp. 57-60, (Rochester, NY), Sept. 22-25, 2002

[Maurer and Tochtermann, 2002] Maurer, H., and Tochtermann, K. "On a New Powerful Model for Knowledge Management and its Applications" J.UCS vol.8., No.1, 85-96.

[Maurer et al., 2006] Maurer, H., Krottmaier, H., and Dreher, H. "Important Aspects of Digital Libraries": International Conference of Digital Libraries, New Delhi, Dec.5-8, 2006

[Maurer and Zaka, 2007] Maurer, H., and Zaka, B., "Plagiarism – a problem and how to fight it" In proceedings of ED-MEDIA 2007, World Conference on Educational Multimedia, Hypermedia & telecommunication, June 25-28 Vancouver, Canada, Pp. 4451-4458

[Maurer et al., 2006] Maurer, H., Kappe, F., and Zaka, B. "Plagiarism- a Survey". Journal of Universal Computer Science 12, 8, 1050-1084.

[McCallum et al., 2000] McCallum, A., Nigam, K., A., and Ungar, L., H. "Efficient clustering of high-dimensional data sets with application to reference matching" In Proceedings of the Sixth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (Boston, Massachusetts, United States, August 20 - 23, 2000). KDD '00. ACM, New York, NY, 169-178. DOI=<http://doi.acm.org/10.1145/347090.347123>

[McCown and Nelson, 2007] McCown, F., and Nelson, M., L. "Agreeing to Disagree: Search Engines and their Public Interfaces" In Proceedings of Joint Conference on Digital Libraries (JCDL) 2007.

[Middleton et al., 2004] Middleton, S., E., Shadbolt, N., R., and De Roure, D., C. "Ontological User Profiling in Recommender Systems", ACM Transactions on Information Systems (TOIS), Volume 22, Issue 1, ACM Press, Jan. 2004, NY, USA, pages 54–88.

[Miller, 1995] Miller, G. A. "WordNet: a lexical database for English" Communications of the ACM 38 (11), 39 - 41.
<http://www.acm.org/pubs/articles/journals/cacm/1995-38-11/p39-miller/p39-miller.pdf>

[Milligan and Cooper, 1987] Milligan, G., W., and Cooper, M., C. "Methodology Review: Clustering", Applied Psychological Measurement, Vol. 11, No. 4, 329-354 (1987) DOI: 10.1177/014662168701100401

[Milosevic, 2007] Milosevic, D. "Beyond Centralised Search Engines" An Agent-Based Filtering Framework, VDM Verlag Dr. Müller 2007 ISBN: 978-3-8364-1222-3

[MIT News, 2003] MIT, (March 25, 2003), "Budget projections, student discipline report presented to faculty", <http://web.mit.edu/newsoffice/2003/facmeet.html> visited: 22 July 2006

[MIT policies, 2006] Massachusetts Institute of Technology Policies and Procedures, <http://web.mit.edu/policies/10.0.html> visited: 22 July 2006

[MIT Writing, 2006] MIT Online Writing and Communication Center, <http://web.mit.edu/writing/> visited: 22 July 2006

[MOSS, 2006] MOSS, A System for Detecting Software Plagiarism website, <http://www.cs.berkeley.edu/~aiken/moss.html> visited: 22 July 2006

[MRML, 2008] MRML (Multimedia Retrieval Markup Language) Retrieved November 20, 2008, from <http://www.mrml.net>

[Müller et al., 2004] Müller, H., Pun, T., and Squire, D., M. "Learning from User Behavior in Image Retrieval: Application of Market Basket Analysis", International Journal of Computer Vision, Volume 56, Numbers 1-2 / January, 2004, PP 65-77, DOI: 10.1023/B:VISI.0000004832.02269.45

[Mydropbox, 2006] Mydropbox, SafeAssignment Product Brochure, http://www.mydropbox.com/info/SafeAssignment_Standalone.pdf visited: 22 July 2006

[Najjar et al., 2005] Najjar, N., Klerkx, J., Vuorikari, R., and Duval, E. "Finding Appropriate Learning Objects: An Empirical Evaluation" Research and Advanced Technology for Digital Libraries, Springer Berlin / Heidelberg, 3652, 323-335.

[Niezgoda and Way, 2006] Niezgoda, S., and Way, T., P. "SNITCH: a Software Tool for Detecting Cut and Paste Plagiarism." SIGCSE Technical Symposium (SIGCSE 2006), March 2006.

[OpenSearch, 2007] Open Search: Documentation, Online at <http://www.opensearch.org/> (Accessed April 26, 2007)

[Oviatt and Cohen, 2000] Oviatt, S. and Cohen, P., "Multimodal Interfaces That Process What Comes Naturally", Communications of the ACM, Vol. 43, No. 3, March 2000, pages 45-53.

[Oviatt et al., 2000] Oviatt, S., Cohen, P., R., Wu, L., Vergo, J., Duncan, L., Suhm, B., Bers, J., Holzman, T., Winograd, T., Landay, J., Larson, J., and Ferr, D. "Designing the user interface for multimodal speech and gesture applications: State-of-the-art systems and research directions", Human Computer Interaction, 2000, 15(4), pages 263-322.

[Oxford Gazette, 2005] Oxford University Gazette, (23 March 2005), http://www.ox.ac.uk/gazette/2004-5/supps/1_4728.htm visited: 22 July 2006

[Pappis and Karacapilidis, 1993] Pappis, C., P., and Karacapilidis, N., I., "A comparative assessment of measures of similarity of fuzzy values" Fuzzy Sets and Systems 56 (1993) 171-174.

[Parekh et al., 2004] Parekh, V., Gwo, J., and Finin, T., W. "Mining Domain Specific Texts and Glossaries to Evaluate and Enrich Domain Ontologies" International Conference on Information and Knowledge Engineering, Las Vegas USA, June 2004, pages 533-540.

[Peltz, 2003] Peltz, C. "Web Services Orchestration and Choreography" Computer, vol. 36, no. 10, pp. 46-52, Oct., 2003

[Peng et al., 2006] Peng, Y., Ngo, C., Fang, C., Chen, X., and Xiao, J. "Audio similarity measure by graph modeling and matching" In Proceedings of the 14th Annual ACM international Conference on Multimedia (Santa Barbara, CA, USA, October 23 - 27, 2006). MULTIMEDIA '06. ACM, New York, NY, 603-606. DOI=<http://doi.acm.org/10.1145/1180639.1180763>

[Plagiarism.org, 2006] Research resources at [plagiarism.org, http://www.plagiarism.org/research_site/e_what_is_plagiarism.html](http://www.plagiarism.org/research_site/e_what_is_plagiarism.html) , visited: 22 July 2006

[Plagiarism, 2007] Plagiarism.org: Statistics. Online at http://www.plagiarism.org/plagiarism_stats.html (Accessed March 15, 2007)

[PlagiarismToday, 2008] Plagiarism Today, Retrieved November 20, 2008, from <http://www.plagiarismtoday.com>

[PPT Demo, 2008] Demo application (2008), Retrieved March 10, 2008, from <http://cluster.cpdnet.org>

[Pratt et al., 1969] Pratt, W., K., Kane, J., and Andrews, H., C. "Hadamard transform image coding" In proceedings of the IEEE Publication Jan. 1969

[Ribeiro and Muntz, 1996] Ribeiro, B., A., and Muntz, R. "A belief network model for IR" In Proceedings of the 19th Annual international ACM SIGIR Conference on Research and Development in information Retrieval (Zurich, Switzerland, August 18 - 22, 1996). SIGIR '96. ACM, New York, NY, 253-260. DOI= <http://doi.acm.org/10.1145/243199.243272>

[Riedl, 2001] Riedl, J. "Guest Editor's Introduction: Personalization and Privacy" IEEE Internet Computing Volume 5, Issue 6, Nov. 2001, Pages 29-31. DOI= <http://dx.doi.org/10.1109/4236.968828>

[Rutgers, 2003] Study at Rutgers Confirms Internet Plagiarism Is Prevalent, <http://ur.rutgers.edu/medrel/viewArticle.html?ArticleID=3408> visited: 22 July 2006

[Salton et al., 1982] Salton, G., Fox, E. A., and Wu, H. "Extended Boolean Information Retrieval" Technical Report. UMI Order Number: TR82-511., Cornell University.

[Search2.0, 2006] Search 2.0 vs. Traditional Search: Written by Ezzy, E., and edited by MacManus, R. July 20, 2006 Online at http://www.readwriteweb.com/archives/search_20_vs_tr.php (Accessed March 14 2007)

[SecondLife, 2008] Second Life Economic Statistics, Retrieved November 20, 2008, from http://secondlife.com/whatis/economy_stats.php

[Scheidl et al., 1999] Schiedl A., R., Ekhal J., van Gelovan O., Kovacs L., Micsik A., Lueg C., Messnarz R., D.M. Nichols, Palme J., Tholerus T., Mason D., Procter R., Stupazzini E., Vassali M., and Wheeler M. "SELECT: Social and Collaborative Filtering of Web Documents and News" 5th ERCIM Workshop on User Interfaces for All: User-Tailored Information Environments, Dagstuhl, Germany, Nov. 28th - Dec. 1st 1999, pages 23-37.

[Schleimer et al., 2003] Schleimer, S., Wilkerson, D., S., and Aiken, A. "Winnowing: local algorithms for document fingerprinting" In SIGMOD: Proceedings of the 2003

[SCORM, 2008] Sharable Content Object Reference Model by Advanced Distributed Learning, Accessed Dec. 10, 2008, <http://www.adlnet.org>

[Sheikh et al., 2002] Sheikh, H., R., Wang, Z., Cormack, L., and Bovik, A., C. "Blind quality assessment for JPEG2000 compressed images" In proceedings of Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002.

[Sonka et al., 2007] Sonka, M., Hlavac, V., and Boyle, R. Image Processing, Analysis, and Machine Vision, 3rd edition ISBN: 049508252X , 9780495082521 2007

[Squire et al., 1999] Squire, D., M., Müller, W., Müller, H., and Pun, T. "Content based query of image databases: inspirations from text retrieval", Pattern Recognition Letters (Selected Papers from The 11th Scandinavian Conference on Image Analysis SCIA '99), 21(13-14)

[Stanford Copyright, 2006] Copyright and Fair use portal at Stanford University, <http://fairuse.stanford.edu/> visited: 22 July 2006

[Stanford Daily, 2003] The Stanford Daily, Feb. 12, 2003 By Ali Alemozafar, [http://daily.stanford.edu/article/2003/2/12/online Software Battles Plagiarism At Stanford](http://daily.stanford.edu/article/2003/2/12/online_Software_Battles_Plagiarism_At_Stanford) visited: 22 July 2006

[Stanford Honorcode, 1921] Stanford Honor Code, <http://www.stanford.edu/dept/vpsa/judicialaffairs/guiding/pdf/honorcode.pdf>, visited: 22 July 2006

[Steinbach et al., 2000] Steinbach, M., Karypis, G., and Kumar, V., "A Comparison of Document Clustering Techniques", Technical Report 00-034, May 2000, University of Minnesota, Minneapolis USA

[Teevan et al., 2005] Teevan, J., Dumais, S. T., and Horvitz, E. "Personalizing search via automated analysis of interests and activities" In Proceedings of the 28th Annual international ACM SIGIR Conference on Research and Development in information Retrieval (Salvador, Brazil, August 15 - 19, 2005). SIGIR '05. ACM, New York, NY, 449-456. DOI= <http://doi.acm.org/10.1145/1076034.1076111>

[Turk and Robertson, 2000] Turk, M., and Robertson, G. "Perceptual user interfaces" Communications of the ACM, Volume 43, Issue 3 ACM Press, NY USA, March 2000, pages: 32 – 34.

[Turnitin tour, 2006] Plagiarism Tour at [turnitin.com](http://www.turnitin.com), <http://www.turnitin.com/static/flash/tii.html>, visited: 22 July 2006

[Turtle and Croft, 1990] Turtle, H. and Croft, W. B. "Inference networks for document retrieval" In Proceedings of the 13th Annual international ACM SIGIR Conference on Research and Development in information Retrieval (Brussels, Belgium, September 05 - 07, 1990). J. Vidick, Ed. SIGIR '90. ACM, New York, NY, 1-24. DOI= <http://doi.acm.org/10.1145/96749.98006>

[UCLA, 1969] UCLA Press Release, 3 July, 1969. <http://www.lk.cs.ucla.edu/LK/Bib/REPORT/press.html>.

[Urkund, 2006] Urkund website, <http://www.urkund.com/> visited: 22 July 2006

[Uzuner et al. 2005] Uzuner, Ö., Katz, B., and Nahnsen, T. "Using Syntactic Information to Identify Plagiarism" Proceedings of the 2nd Workshop on Building Educational Applications Using NLP, pages 37–44, Ann Arbor, June 2005. Association for Computational Linguistics.

[Vasconcelos, 2004] Vasconcelos, N. "On the efficient evaluation of probabilistic similarity functions for image retrieval" (2004). IEEE Transactions on Information Theory. 50 (7), pp. 1482-1496. Post print available free at: <http://repositories.cdlib.org/postprints/694>

[W3C, 2004] W3C: Web Services Architecture, W3C Working Group Note 11 February 2004. Online at <http://www.w3.org/TR/ws-arch/> (Accessed March 13, 2007)

[Wallach, 1958] Wallach, M., A. "On psychological similarity" Psychological Review, 65, 103–116

[Wan and Peng, 2005] Wan, X., and Peng, Y. "A Measure Based on Optimal Matching in Graph Theory for Document Similarity", Information Retrieval Technology, Volume 3411/2005, Springer Berlin / Heidelberg ISSN 0302-9743 (Print) 1611-3349 (Online), Pages 227-238, DOI: 10.1007/b106653

[Warren, 2005] Warren S. A., "DOIs and Deeplinked E-Reserves: Innovative Links for the Future" Technical Services Quarterly, Vol. 22, number 4, 2005. DOI: 10.1300/J124v22n04_01

[Wcopyfind, 2006] WCopyfind website. <http://plagiarism.phys.virginia.edu/Wsoftware.html> visited: 22 July 2006

[Weiss, 2007] Weiss, A. "Computing in the clouds" netWorker 11, 4, December 2007, 16-25. DOI= <http://doi.acm.org/10.1145/1327512.1327513>

[Wikipedia:Bayes 2008] Bayes' theorem. In Wikipedia, The Free Encyclopedia. Retrieved 13:56, November 6, 2008, from http://en.wikipedia.org/w/index.php?title=Bayes%27_theorem&oldid=250007340

[Wikipedia:CBIR, 2008] Content-based image retrieval. In Wikipedia, Retrieved, October 30, 2008, from http://en.wikipedia.org/w/index.php?title=Content-based_image_retrieval&oldid=241031072

[Wikipedia:CopyBot, 2008] CopyBot, In Wikipedia, Retrieved November 19, 2008, from <http://en.wikipedia.org/w/index.php?title=CopyBot&oldid=251130559>

[Wikipedia:DW, 2008] Digital watermarking. In Wikipedia, Ret. Nov. 26, 2008, from http://en.wikipedia.org/w/index.php?title=Digital_watermarking&oldid=250325938

[Wikipedia:Jaccard, 2008] Jaccard index. In Wikipedia, The Free Encyclopedia. Retrieved 13:41, October 29, 2008, from http://en.wikipedia.org/w/index.php?title=Jaccard_index&oldid=246539705

[Wikipedia:MAP, 2008] Maximum a posteriori. In Wikipedia, The Free Encyclopedia. Retrieved 12:22, October 31, 2008, from http://en.wikipedia.org/w/index.php?title=Maximum_a_posteriori&oldid=242935560

[Wikipedia:MLE, 2008] Maximum likelihood. In Wikipedia, The Free Encyclopedia. Retrieved 12:09, October 31, 2008, from http://en.wikipedia.org/w/index.php?title=Maximum_likelihood&oldid=243890441

[Wikipedia:MPEG-7, 2008] MPEG-7. In Wikipedia, Ret. Nov. 14, 2008, <http://en.wikipedia.org/w/index.php?title=MPEG-7&oldid=251064118>

[Wikipedia:papermill, 2006] Paper mill (essays). In Wikipedia, The Free Encyclopedia. Retrieved 09:13, July 25, 2006, from http://en.wikipedia.org/w/index.php?title=Paper_mill_%28essays%29&oldid=64074352

[Wikipedia:Plagiarism, 2006] Plagiarism. In Wikipedia, The Free Encyclopedia. Retrieved 09:11, 22 July 2006, from <http://en.wikipedia.org/w/index.php?title=Plagiarism&oldid=65284248>

[Wikipedia:Kent, 2006] University of Kent. In Wikipedia, http://en.wikipedia.org/w/index.php?title=University_of_Kent&oldid=64849655. visited: July 25, 2006

[Wikipedia:SimMetrics, 2008] SimMetrics. In Wikipedia, The Free Encyclopedia. Retrieved 15:26, October 29, 2008, from <http://en.wikipedia.org/w/index.php?title=SimMetrics&oldid=246093040>

[Wikipedia:Stylometry, 2008] Stylometry. In Wikipedia, The Free Encyclopedia. Retrieved 16:27, October 29, 2008, from <http://en.wikipedia.org/w/index.php?title=Stylometry&oldid=234689876>

[Wikipedia:Tf-idf, 2008] Tf-idf. In Wikipedia, The Free Encyclopedia. Retrieved 14:35, December 4, 2008, from <http://en.wikipedia.org/w/index.php?title=Tf%E2%80%93idf&oldid=248453488>

[Wikipedia:VSM, 2007] Vector Space Model, In Wikipedia, The Free Encyclopedia. Online at http://en.wikipedia.org/w/index.php?title=Vector_space_model&oldid=113611338 (Accessed March 15, 2007).

[Williams, 2006] Williams, R. "The Power of Normalised Word Vectors for Automatically Grading Essays" The Journal of Issues in Informing Science and Information Technology Volume 3, 2006 pp. 721-730

[Wilkinson and Hingston, 1991] Wilkinson, R. and Hingston, P. "Using the cosine measure in a neural network for document retrieval" In Proceedings of the 14th Annual international ACM SIGIR Conference on Research and Development in information Retrieval (Chicago, Illinois, United States, October 13 - 16, 1991). SIGIR '91. ACM, New York, NY, 202-210. DOI= <http://doi.acm.org/10.1145/122860.122880>

[WordNet, 2008] WordNet Lexical Database (2008) Retrieved March 10, 2008, from <http://wordnet.princeton.edu/>

[WSBPEL, 2007] WS-BPEL 2.0 Primer. OASIS Web Services Business Process Execution Language. version 2.0, Primer, 9 May 2007. <http://docs.oasis-open.org/wsbpel/2.0/Primer/wsbpel-v2.0-Primer.pdf>

[Xiao et al., 2003] Xiao, B., Lunsford, R., Coulston, R., Wesson, M., and Oviatt, S. "Modeling multimodal integration patterns and performance in seniors: toward adaptive processing of individual differences" 5th international conference on Multimodal interfaces 2003, ACM Press, pages 265–272.

[Xu and Wunsch, 2005] Xu, R. and Wunsch II D., "Survey of clustering algorithms", IEEE Transactions on Neural Networks, Vol. 16, No. 3, May 2005, 645-678, DOI: 10.1109/TNN.2005.845141

[Yale, 2005] Yale College Executive Committee Yearly Chair Reports, <http://www.yale.edu/yalecol/publications/executive/index.html> visited: 22 July 2006

[Yahoo:TermExtraction, 2007] Yahoo Content Analysis Web Services: Term Extraction, Online at <http://developer.yahoo.com/search/content/V1/termExtraction.html> (Accessed April 10, 2007)

[Yao et al., 2007] Yao, Y., Zeng, Y., Zhong, N., and Huang, X. "Knowledge Retrieval (KR)" In Proceedings of the IEEE/WIC/ACM international Conference on Web intelligence (November 02 - 05, 2007). Web Intelligence. IEEE Computer Society, Washington, DC, 729-735. DOI= <http://dx.doi.org/10.1109/WI.2007.139>

[Yoshitaka and Ichikawa, 1999] Yoshitaka, A. and Ichikawa, T. "A survey on content-based retrieval for multimedia databases," Knowledge and Data Engineering, IEEE Transactions on , vol.11, no.1, pp.81-93, Jan/Feb 1999

[Zaka, 2009a] Zaka, B. "A practical approach to enrich classification of digital libraries" Accepted to appear in proceedings of Third International Conference on Research Challenges in Information Science, RCIS, Morocco, April 2009

[Zaka, 2009b] Zaka, B. "Empowering plagiarism detection with a web services enabled collaborative network" Accepted to appear in Journal of Information Science and Engineering, ISSN: 1016-2364

[Zaka and Safran, 2008] Zaka, B., and Safran, C. "Emerging Web Based Learning Systems and Scalability Issues" In proceedings of International Conference on

Information Technology in Education, CSSE Dec. 2008 Wuhan China, Pp.889-892, DOI: 10.1109/CSSE.2008.187

[Zaka and Maurer, 2007] Zaka B., and Maurer H., “Service Oriented Information Supply Model for Knowledge Workers” In proceedings of 7th International Conference on Knowledge Management i-KNOW Sep. 2007 Graz Austria.

[Zaka et al., 2007] Zaka, B., Safran, C., and Kappe, F. “Personalized Interactive Newscast (PINC): Towards a Multimodal Interface for Personalized News” In proceedings of Second International Workshop on Semantic Media Adaptation and Personalization 2007 London UK, On page(s): 56-61, ISBN: 0-7695-3040-0, DOI: <http://doi.ieeecomputersociety.org/10.1109/SMAP.2007.4414387>

[Zaka et al., 2008] Zaka, B., Kulathuramaiyer, N., Balke, T., and Maurer, H., “Topic-Centered Aggregation of Presentations for Learning Object Repurposing” In proceedings of E-Learn 2008 Las Vegas, Nevada Nov. 17-21, 2008

[Zaka et al., 2009a] Zaka, B., Safran, C., and Kappe, F. “Use of similarity detection techniques for adaptive news content delivery and user profiling” Book Chapter, Advances in Semantic Media Adaptation and Personalization Volume 2, CRC press, ISBN: 978-1-4200-7664-6, to appear in March 2009

[Zaka et al., 2009b] Zaka, B., Steurer M. E., and Kappe F. “Framework for Extending Plagiarism Detection in Virtual Worlds” Accepted to appear in proceedings of Third International Conference on Research Challenges in Information Science, RCIS, Morocco, April 2009

[Zobel and Moffat, 2006] Zobel, J, and Moffat, A. “Inverted files for text search engines” ACM Computing Surveys 38, 2, Jul. 2006, DOI= <http://doi.acm.org/10.1145/1132956.1132959>

[Zobel et al., 1998] Zobel, J., Moffat, A., and Ramamohanarao, K. “Inverted files versus signature files for text indexing” ACM Trans. Database Syst. 23, 4 (Dec. 1998), 453-490. DOI= <http://doi.acm.org/10.1145/296854.277632>