Fuzzy Information Retrieval

Seminar presentation
27.07.2005

University Hagen, Department of Computer Science,
Chair of Applied Computer Science VII,
Int. Information and Communication Systems Group of Prof. Helbig,
Seminar Softcomputing, Dr. Glöckner

Vincent Wolowski
vwolowski@gmail.com
# Table of Contents

## Information Retrieval
- Introduction, Definitions, Examples, Components, Conceptual View and Key Words

## Retrieval Models
- Formal Characterization, three main Models

## Indexing
- Indexing and Document Representation, Goal of Indexing

## Boolean Retrieval Model
- Overview and Limitations of Boolean Retrieval Model

## Fuzzy Set Theory and Information Retrieval

## Queries and Weights
- Numeric Query Weights, Retrieval Status Value, Separability Property, Matching Function, Different Types, Problems

## Linguistic Extension of Boolean Query Language

## Aggregation using OWA Operators

## Document Representation and Information Content
- Positional Operator, Compound Fuzzy Document Representation, Challenges of Fuzzy CDR

## Thesaurus and Information Retrieval

## Fuzzy Thesaurus

## Fuzzy Information Retrieval: Advantages and Disadvantages
Introduction: What is Information Retrieval?

Definition: G. Salton (1968)
• Information retrieval is a field concerned with the structure, analysis, organization, storage, searching and retrieval of information.

Definition: G. Klir et al. (1995)
• Information retrieval may be defined, in general, as the problem of the selection of documentary information from storage in response to search questions.

IR (Information Retrieval) is the study of the
• Representation
• Storage
• Organization
• Access of information items (articles, books, web pages, movies, pictures, etc.) for the people who are interested in them.

➔ In the following focus on text-based Information Retrieval.
Information Retrieval System

**Definition: G. Bordogna et al. (1995)**

- Information retrieval systems process user queries for information through a content-based analysis of the information items (e.g. documents) stored in an archive. The document contents must be represented in a form that is automatically processable.

An **IR system**
- is designed to support information needs.
- has the goal to deliver a result list of documents that answer the user information need expressed as a formal query.

A **good IR system** should be able to
- accept a user query,
- understand from the user query what the user requires,
- search a database for relevant documents,
- retrieve the documents to the user, and
- rank the documents according to their relevance.
Examples of IR Systems

- DBMS (*Oracle*, *MySQL*, *MS Access*, etc.)
- Expert systems (*OPS5*, *Clips*, *Jess*, etc.)
- Web search engines (*Google*, *Altavista*, *Lycos*, etc.)
- Digital libraries (*Gutenberg project*, etc.)
- Online Information system (*Medline*, etc.)
- Desktop search tools (*Spotlight*, *MSN Desktop Search*, etc.)
- and others, e.g. airline flight booking system,...
Components of an IR System

Components of IR system (for textual documents):

• User Interface for query and result
  ➔ Allows the user to input a query and view the result set.

• Query interpreter
  ➔ Processes the query in a manner similar to the documents.

• Indexer module
  ➔ Creates the index, which enables faster searching.

• Matching mechanism
  ➔ Determines if a document is relevant or not.

• Documents and document representations
  ➔ The actual pieces of information and their logical view.
IR Process: Conceptual View

1. Indexing
   - Documents

2. Query Formulation
   - Query Representation

3. Searching and Matching
   - Retrieval Function
   - Retrieved Documents

4. Presentation of Result Set
   - Evaluation

5. Evaluation
   - Yes
   - End
   - No
   - Done?

IR System
IR: Key Words

Term
- Individual symbols, words and phrases in a textual document.

Index Terms (= Index, Descriptor)
- A set of content identifiers (either terms or sentences) extracted from the document.

Term Frequency
- **absolute**: number of times term $t$ appears in document $d$.
- **relative**: absolute term frequency of term $t$ in document $d$ divided by the highest absolute term frequency.
Inverse Document Frequency (idf)

- measure for the importance of a word.
- logarithm of the ratio of number of documents in a collection to the number of documents containing the given term $t$.

$$\text{idf}(t) = \log \frac{\text{number of documents}}{\text{occurrence of term } t \text{ in documents}}$$

- provides high values for rare terms and low values for common terms.
- often normalized, so longer documents are not unfairly given more weight.
Precision
- The proportion of relevant documents of all documents retrieved.

\[
\text{Precision} = \frac{\text{number of relevant documents retrieved}}{\text{number of retrieved documents}}
\]

Recall
- The proportion of retrieved documents of all relevant documents available.

\[
\text{Recall} = \frac{\text{number of relevant documents retrieved}}{\text{number of relevant documents}}
\]

Ranking
- Ordering of the retrieved documents in regards to their relevance.

Query
- Formal representation of an information need.
Retrieval Models

- For a successful IR, it is necessary to represent the documents in some way.

- A retrieval model specifies the representations used for documents and information needs, and how they are compared (Turtle and Croft, 1992).

- Retrieval models define relevance (explicitly or implicitly).

- A model is an embodiment of the theory in which we define a set of objects about which assertions can be made and restrict the ways in which classes of objects can interact.
An **Information Retrieval model** is a quadruple $<D, Q, F, R>$ where

- $D$ is a set of representations for the documents in the collection.
- $Q$ is a set of representations for the user information needs (queries).
- $F$ is a framework for modeling document representations, queries, and their relationships.
- $R: Q \times D \rightarrow \mathbb{R}$ is a ranking function which associates a real number with a query $q_i \in Q$ and document representation $d_j \in D$.

Three main IR Models

- **Three main groups** of retrieval models

  - **Set-theoretic**
    - Standard Boolean model, e.g. OPACs (Online Public Access Catalogs)
    - Fuzzy Logic model, e.g. Inquiry Assistant at Bielefeld University (www.ub.uni-bielefeld.de/databases/rechercheassistent/)

  - **Algebraic**
    - Vector Space model, e.g. SMART, *Salton et al. 1971*

  - **Probabilistic** (*Van Rijsbergen 1979*)
    - Probability theory-based model, e.g. OKAPI, *Robertson/Sparck Jones 1976*

- There are others IR models, Example: Citation analysis model (e.g. Pagerank Google).
Indexing and Document Representation

• Documents are usually written in a natural language.

• Cannot be compared directly with a query to estimate relevance.

• A suitable representation of the document is needed, which can be manipulated by the computer.

• **Indexing:** the task of finding terms that describe documents well.

• In the **indexing process**
  • the documents are analyzed to provide a formal representation of their content.
  • a surrogate describing the document is stored in an index.
The **goal of indexing** is twofold:

- It identifies the most important concepts described in the document.
- It measures the importance of each concept in the document.

**Ideally**: identification of **concepts** (= semantic, meaningful entities).

Difficult, in practice a semantic model for identifying concepts does not exist.

**Semantic dimension** reduced to terms (= words).

**Full-text Indexing**: Representing documents by the mathematical set of all the terms of the document (excluding terms which do not sufficiently represent document content) providing a wide spectrum of index terms for query formulation.
**Boolean Retrieval Model**

- Basis of the majority of current commercial systems, e.g. often used in search engines on the Internet because it is fast and can therefore be used online.
- Boolean weighting: 1 if term is present, 0 otherwise.
- Index procedure performs simple yes-or-no decision, therefore simplified document interpretation.
- Binary decision: Document is relevant or not (no ranking).
- Presence of term is necessary and sufficient for match.
- Relies on the use of Boolean operators in the query: the terms are combined together with **AND**, **OR** and **NOT**.

\[ F : D \times T \rightarrow \{0,1\} \]

- **F**: Function defining the relation between documents and terms
- **D**: Document
- **T**: Term
Limitations of the Boolean Retrieval Model

- Oversimplified representation of the information items (documents).
- No formal means for qualifying the role and degree of the terms in characterizing document contents.
- Matching mechanism only based on the evaluation of the presence of a given search term in the document representation.
- No way of establishing the degree of usefulness of each single document.
- Problems with Boolean Operators
  - Disjunctive (OR) queries lead to information overload by too many results.
  - Conjunctive (AND) queries lead to reduced, and commonly zero result.
  - Conjunctive queries imply reduction in Recall.
- Query language gives users only a crisp way of specifying their information needs: term is either definitely significant or completely useless.
- No discriminating power.
Fuzzy Set Theory and IR

- Extending the Boolean Retrieval model by applying Fuzzy Set Theory (Zadeh, 1965).
- Fuzzy Information Retrieval: methods of IR that are based upon fuzzy set theory.

Why Fuzzy Set Theory?

- Imprecision and subjectivity play an important role at various stages of IR:
  - In the formulation of information requirements.
  - In the estimation of the extent to which each information item from a given source is relevant to a user request, and
  - In deciding which information items must be retrieved with respect to a given request.

- Fuzzy set theory is very suitable to model imprecision in IR:
  - It is a formal tool designed for dealing with imprecision and vagueness.
  - It facilitates the definition of a superstructure of the Boolean model, so that existing Boolean IR systems can be modified without re-designing them completely.

- Main levels of applications of Fuzzy Set Theory to IR:
  - The definition of extensions of the Boolean model, concerning both the representation of documents and the query language.
  - The definition of associative mechanisms, such as fuzzy thesauri and fuzzy clustering.
Traditional Fuzzy Document Representation

**Function** $F$ defined in the following way:

$$F : D \times T \rightarrow [0,1]$$

$F(d,t)$ changes from a crisp set value (either 0 or 1) to a continuous membership value in the range $[0,1]$.

**Index term weight**
- the degree of “aboutness” of a document with respect to a term, expressed by value $F(d,t)$.
- also interpreted as the significance of term in representing the document content.

**How can a useful $F(d,t)$ look like?**
- considering the frequency of the terms in single documents and the frequency of the terms in the whole archive.
- aiming to obtain high precision and good recall.
Salton & McGill (1989): Formula for $F(d, t)$

$$F(d, t) = tf_{dt} \times idf_t$$

$tf_{dt}$: frequency of term $t$ in document $d$:

$$tf_{dt} = \frac{\text{number of occurrences of term } t \text{ in document } d}{\text{number of occurrences of the most frequent term } t \text{ in document } d}$$

$idf_t$: Inverse document frequency of term $t$:

$$idf_t = \log \frac{\text{total number of documents in a collection}}{\text{number of documents containing term } t}$$

$F(d, t)$ increases
- with the number of occurrences within a document.
- with the rarity of the term across the whole collection.
Queries and Weights

Query

- A query is the formal expression of a user information need and consists of a set of selection criteria connected through operators.

Selection Criterion

- A selection criterion is the elementary unit for requesting information.
- Weights can be applied to the selection criterion in order to specify the importance of the search term in the desired documents.

Questions in regards to Query Weights

- How should query weights be interpreted?
- What is the relation between query weights and index term weights?
- How to associate the weights in regards to a query as a whole and to its sub-expressions?
Numeric Query Weights

Numeric Query Weight

(Waller and Kraft: A mathematical model of a weighted Boolean retrieval system, 1979).

Different Interpretation:

- is a formalization of the importance of a search term (*relative importance semantics*).
- allows the user to quantitatively express the importance that terms must have for documents to be selected (*ideal index term weight or threshold semantics*).
- specifies a soft constraint to be satisfied by the fuzzy representation of documents.
- enables ranking in regards to the relevance of the documents retrieved.
Retrieval Status Value and Separability Property

Retrieval Status Value (RSV)
• expresses the degree to which document $d$ matches a query $q$ consisting of a single weighted term.

Separability property (Waller and Kraft, 1979)
• the evaluation of a single query component must not influence the evaluation of the other components and motivates the use of a matching function for each component of the query.
Matching Function for Numeric Query Weights

Matching function $g(F(d,t),w)$

- matches a selection criterion $<t,w>$ (t: term, w: weight) from a query $q$ against a document $d$.
- is defined as follows:

$$g : [0,1] \times [0,1] \rightarrow [0,1]$$

- the value $g(F(d,t),w)$ is the degree of the satisfaction criterion $<t,w>$ by document $d$.
- through satisfaction of the separability property, the query as a whole can be evaluated from bottom up by evaluating each single weighted query term against the document $d$ and then combing the results to achieve the final RSV of the document $d$. 

The nature of the constraint imposed by the weighted selection criterion depends on the semantics associated with the weight.

**Relative importance semantics**
- Defines query weights as measures of the “relative importance” of each term with respect to the others in the query (*Radecki, 1979; Bookstein, 1980*).

**Threshold semantics**
- Proposing a “threshold” semantics for query weights: a weight w associated with a query term requires that documents be evaluated by checking their degrees of significance $F(d,t)s$ against the threshold $w$ (*Buell and Kraft, 1981*).

**Ideal Index Term weight semantics**
- Applying the weight at two distinct levels: the query term weight and the query weight, using different semantics (*Cater and Kraft, 1989; Bordogna, Carrara and Pasi, 1991*). The query is interpreted as the specification of a “perfect collection” of documents or as a set of “ideal” documents satisfying user needs. By selecting a query term weight, users specify the desired index-term weight of the term in an ideal matching document.
Problems with Numeric Query Weights

Problems

- the operators aggregating the selection criteria are limited to **AND** and **OR**.
- incapable of modeling the imprecision and vagueness of users in formulating a request for information.
- the user is forced to quantify the qualitative and vague concept of importance into a numeric value, which is counter-intuitive.
- several numeric query weights can express the same constraint of importance.

⇒ selection requirements could be more naturally expressed in a linguistic form, e.g. a given term \( t \) be “fairly important” in the desired documents.
Linguistic extension of the Boolean query language introduces fuzziness in queries at two distinct levels:

- to fuzzify selection requirements.
- to define new aggregation operators besides AND and OR.

Goals

- simplify query formulation.
- improve expressiveness.
- allow a compensation between the constraints satisfaction values.

- Linguistic descriptors play the role of fuzzy weights associated with the query terms.
Linguistic Descriptors with Ideal Semantics

Linguistic descriptors with ideal semantics

- Bordogna and Pasi have defined a fuzzy retrieval model in which the linguistic descriptors are formalised within the framework of fuzzy set theory through linguistic variables (Bordogna and Pasi, *Linguistic aggregation operators of selection criteria in fuzzy information retrieval*, 1995).

  - generalization of numeric query term weights based on ideal semantics.

  - linguistic selection requirements can be defined by replacing the numeric weights with linguistic descriptors, which fuzzify the numeric values.

  - This linguistic model adopts the traditional fuzzy document representation.

Linguistic variables (= linguistic term)

- Variables used in Fuzzy Logics to express qualities such as height, which can take linguistic values such as “tall” or “short”.

Hedges

- common set of operations on linguistic variables.

  - is a qualifier on a linguistic variable.

  - modifies the membership function to reflect a variation on its semantics.

  - examples: very, quite, mostly, few...
A \(<t, l>\) pair identifies a qualitative selection criterion

- **t**: term
- **l**: value belonging to the term set of the linguistic variable (e.g. “Importance”) with range of \([0,1]\).

Such a query language can be employed by any IRS with a weighted document representation.

To compute the degree of satisfaction of a pair \(<t, l>\) by a given document \(d\), the compatibility of the index term weight \(F(d, t)\) is evaluated with respect to the constraint imposed by the linguistic query weight \(l\).

The term set of the linguistic variable can be formally generated by means of a context-free grammar.

Example of a term set:

\[ T(Importance) = \{\text{important, very-important, fairly-important, not-important,}...\} \]
Meaning of a linguistic variable \( l \)

- Defined by a function \( \mu_l \) which assesses the compatibility of the representation of documents, i.e. the \( F(d,t) \)s, with the linguistic variable \( l \).

\[
\mu_l(F(d,t)) = \max_{w \in [i,j]} g(F(d,t), w)
\]

- Function \( g \) depends on the semantics adopted for query weights.
- \( i \) and \( j \), with \( i < j \), define the range of numeric values satisfying the linguistic variable \( l \).

The meanings of non-primary linguistic variables \( l \) in a term set \( T \) are obtained by

- defining the compatibility function associated with the primary term \( \mu_l \), and
- modifying \( \mu_l \) according to the semantics of the hedges.
One solution for the aggregation was realized by R. Yager through Ordered Weighted Averaging (OWA).

**Definition of OWA Operator**

- An OWA operator of dimension $n$ is a mapping $F : R^n \rightarrow R$ having an associated $n$ vector $W = [w_1, w_2, \ldots, w_n]$ with $w_j \in [0, 1]$ and $\sum_j w_j = 1$

  where

  $F(a_1, \ldots, a_n) = \sum_{j=1}^{n} w_j b_j$

  with $b_j$ is the $j^{th}$ largest of the $a_i$

**Re-ordering step**

- Using OWA operators allows the ordering of the values prior to the combination.
- An aggregate $a_i$ is not associated with a particular weight $w_i$, but rather a weight is associated with a particular ordered position of an aggregate.
Properties and Types of OWA Operators

Through weighting in relation to ordering, OWA operators enable soft aggregation of constraints.

Properties of OWA operators:
- commutative
- monotone
- bounded
- and also idempotent.

Different OWA operators are distinguished by their weighting function (through appropriate selection of the weights in $W$).

Three important cases:
- $w_i = 1$ and $w_j = 0$ for all $j \neq 1$ (Max-Function)
- $w_n = 1$ and $w_j = 0$ for all $j \neq n$ (Min-Function)
- $w_j = \frac{1}{n}$ for all $j$ (Average-Function)
Dispersion and Orness of OWA Operators

Two important measures for OWA operators defined by R. Yager:

**Dispersion (or Entropy):**

\[ Disp(W) = -\sum_{i=1}^{n} w_i \ln(w_i) \]

•  \( Disp(W) \) measures the degree to which all the aggregates are used equally.

**Orness:**

\[ Orness(W) = \frac{1}{n-1} \sum_{i=1}^{n} (n-i)w_i \]

•  \( Orness(w) \) allows to classify OWA operators in regards to their location between \( AND \) and \( OR \).
Benefits and Limitations of Traditional Fuzzy Document Representation

Benefits

• The retrieved documents can be ranked in decreasing order of their significance with respect to the user query.
• Matching mechanism is softened to a partial matching: computes the degree of relevance of each document to the user query, on the basis of membership values of the query term in document representations.

Limitations

• Extension is limited to document representation: therefore, the representation of queries and of documents is no longer adequate.
• The evaluation of $F$ does not depend on the position of the terms within the documents.

→ A different evaluation of $F$ on the basis of the document structure: 

*Compound fuzzy document representation*
Document Representation and Information Content

• How are documents represented in regards to the information they contain?
• Many IR models take documents into account as carrying **homogenously distributed information**.
• Often, documents have **subparts** and are structured into **semantic units** (called **classes**).
• Example for the structure of a document (e.g. CiteSeer research paper): Title, Authors, Keywords, Abstract, Introduction, Main Body, Conclusion, References etc.

• The **information role** of each term differs according to the class to which the term belongs.

• The **query evaluation mechanism** should be able to make use of the information role.
Positional Operator

Positional Operator

• Many IR systems use a positional operator to allow the user to request documents in which a term is contained in a specific subpart.

Problems of a Boolean model based Positional Operator

• The significance of a term in representing a document content is again a binary value.

• Hence, no distinction between documents containing a term in all classes and documents containing a term in only a few classes.

• No evaluation of the location of the term in the document, for example: appearance in title or keyword section of a document indicates that the term represents a main topic/theme of the content.

Improvement

• To have the significance of a term in a document depend on the classes in which it is contained and on the nature of the information contained in them.
In traditional fuzzy document representation:

- The evaluation of the index term weight function $F(d,t)$ is based on parameters which depend on characteristics of the whole document.
- Define function $F$ so that documents can be managed as structured entities.

The traditional Document Representation is replaced by **Compound Document Representation (CDR):**

- Each term-document pair is associated with a value by $F(c,d,t)$ in which $c$ denotes a class of document $d$.

**Zadeh (1979):** Definition of a **ternary fuzzy relation $FC$**

$$FC : C \times D \times T \rightarrow [0,1]$$

$C$ is the set of classes of documents in the archive of all documents stored.
Document $d$ can then be represented as a fuzzy binary relation $R(d)$:

$$R(d) = \left\{ <(t, c), \mu_d(t, c)> \mid (t, c) \in T \times C \right\}$$

$$\mu_d(t, c) = FC(c, d, t)$$

- expresses the *significance* of term $t$ in class $c$ of document $d$.
- $\mu_d(t, c)$ depends primarily on the characteristics and semantics of class $c$. 
Fuzzy CDR: Challenges

1) How can classes be differentiated in regards to their semantic structure?
   - G. Bordogna et al. (1992) identified two main types of classes:
     - **Structured classes**: containing short texts organized in a well-defined structure.
     - **Unstructured classes**: containing unstructured textual descriptions of variable length.

2) How to evaluate the significance of a term in regards to its occurrence within a class?

   \[ \mu_c(d,t) = FC(c,d,t) = \begin{cases} 
   1 & \text{if } tf_{d_{tc}} \cdot idf_t \geq 1 \\
   tf_{d_{tc}} \cdot idf_t & \text{otherwise} 
\end{cases} \]

3) How to evaluate the term significance regarding all classes of the document?
   - The degrees of significance of a term in the \( n \) classes of a document \( d \)

   \[(\mu_{c_1}(d,t), ..., \mu_{c_n}(d,t))\]

   must be aggregated in order to obtain the overall significance of the term in the whole document.
In general, a thesaurus is a set of items (phrases or words) plus a set of relations between these items.

In regards to IR, a thesaurus

• is an associative index that associates entry terms with related terms expanding queries.
• provides a precise and controlled dictionary which can be applied to coordinate document indexing and retrieval.

→ A thesaurus can also be used to write more precise queries.
Issues related to Thesauri in IR

There are three basic issues related to thesauri in IR as follows:

- **Construction**
  There are two types of thesauri, manually and automatically constructed.

- **Access**
  Given a particular query the thesaurus must be accessed and used in some way to improve or expand the query.

- **Evaluation**
  - After a thesaurus is built, it is important to know how good it is
  - Manual thesauri are evaluated in terms of the soundness, coverage of classification and thesaurus item selection. The evaluation of automatic thesauri is generally done via query expansion to see if retrieval performance is improved.
Fuzzy Thesaurus (Miyamoto 1990)

- Approach for finding words which are semantically related.
- In a fuzzy thesaurus, concepts (= words/semantic entities) are usually organized into a hierarchy being connected via reflexive fuzzy relations.

- **Fuzzy relations**: degrees of association between two categories, represented by membership grades in a fuzzy (binary) relation \([0,1]-\)weighted; to represent the associations between terms.

- A Fuzzy Thesaurus consists of semantic entities and associations among pairs of these semantic entities.

- By using the Fuzzy Thesaurus, the user query can be expanded to contain all the associative semantic entities.

- The expanded query is expected to retrieve more relevant documents.
Fuzzy IR: Advantages

Advantages

• Extends, unlike other IR models, seamlessly the Boolean retrieval model.
• Adds more flexibility to IR.
• Can deal with imprecise data.
• Finer granularity by weighting of query and document terms.
• Can model nonlinear functions of arbitrary complexity.
• Allows partial matching and thereby the retrieval of documents that approximate the query.
• The retrieval result can be ranked and ordered.
• Supports natural language queries through the use of linguistic terms.
Fuzzy IR: Disadvantages

Disadvantages

• based on the assumption that the index term weights are mutually independent.

• problematic how to estimate the membership functions.

• is in itself static and does not adapt through learning and feedback (*can be improved by adding learning capability, e.g. neural network*).

• lacks a semantic model for taking the meaning of the terms into account (*can be improved through the use of thesaurus*).

• requires higher computing costs for membership and aggregation function.

• rapid growth of complexity when number of input variables increases.

• less popular than vector space or probabilistic IR model (*this might change!*?).
Thank you for your attention!