Formal models for Information Retrieval.
A review and a proposal for keyword assignment

Arturo Montejo Ráez

Scientific Information Service
European Organization for Nuclear Research
CH - 1211 Geneva, Switzerland
arturo.montejo.raez@cern.ch

Abstract. In this paper we review some formal models for information retrieval (IR) systems. The common properties of them are used to define a new formalization which models better in automatic keyword assingment paradigms but also other aspects of current systems due to its simplicity and generalization in the definition of components *document*, *query*, *rank function* and *retrieval set*.

1 Introduction

Formalism and *model* are both terms that should be used with care in Information Retrieval (IR) systems. Contrary to this sugesstion, they are mentioned in almost every paper written inside IR research areas. Hence, we read about several models, starting from classic ones like the vector space propossed by Gerald Salton [Salton et al., 1974], probabilistic and boolean models. In this sense, the word *model* means how relevant items in IR (mainly documents, queries and terms) are related to each other and how they are surrogated by simple structures using vectors, probabilities or logic operators. Also, if we have a look in other approaches for IR, like latent semantic indexing [Bartell et al., 1990], neuronal networks [Belew, 1989] or genetic algorithms [Chen, 1995], we can see that they are often referred to as “models”, whether they should better be called *retrieval strategies* (as stated by Grossman and Frieder in their book [Grossman and Frieder, 1998]).

By tradition, formalism in IR has involved the representation in mathematical notation of these strategies. Therefore, a *formal model* consists in the notation used for detailing a retrieval strategy. We want to give here what we find should be the **definition of formal model for IR**:

> A *formal model for information retrieval* is a mathematical notation able to represent any relevant item in an information retrieval

system, along with any usefull relationship (by functions, maps, binary relations...) that the system uses to perform the retrieval task.

2 Previous models

Differences between existing retrieval strategies have produced a wide variety of formal models. If the model is too general it will be only useful for very high abstracted conceptualisation of the information retrieval task. In the other hand, if we define a model deeply enough to cover all the possible aspects of the system, then we will fall in a complex description which becomes difficult to extend rather than being something practical and hand-able.

We could classify models available in the literature depending on the chosen mathematical basis, being logic and algebraic proposals the most common ones. C. J. van Rijsbergen, for example, uses logic methods to deduce interesting properties of the *Logical Uncertainty Principle* at [van Rijsbergen, 1989] and [van Rijsbergen, 2000]. These works show the expressiveness of formal models when describing particular properties in IR methodologies. Logic acts as a tool to consolidates the search for information using inference. But even if the conclusion is relevant, it has a low impact on the practical side. Anyhow these steps must be accomplished in order to build a real theory for information retrieval.

In the aim for setting a global and as generic as possible IR formal model, many authors lied in proposal which had, as pointed out, neither big nor practical repercussions. Some of them set a model to use it as a reference during the development of a discourse content, instead of being used as a tool to extract and model additional properties. This is the case in many text books.

As an example to illustrate a very general algebraic model, we have chosen the one proposed by Grossman and Frieder [Grossman and Frieder, 1998]. They define an IR system as a tuple (original notations are used)

$$I = (D, Q, \delta) \tag{1}$$

where

D is the set of documents

Q is the set of queries

δ is the retrieval function
$\delta : Q \to 2^D, q \mapsto \delta(q) := \delta_i \in 2^D$ (see note[1])

[1] 2^D is the set of all the possible subsets of D (also called the *power set of* D).

Hence, the retrieval function δ produces a subset of documents δ_i as response to a query $q_i \in Q$. It is simple, elegant and clear. This model can be easily extended to include a thesaurus or to describe distributed IRs. With a thesaurus we have:

$$I = (T, D, Q, \delta) \tag{2}$$

where T is the set of distinct terms (controlled vocabulary) with a relationship:

$\rho \subset T \times T$ such that $\rho(t_1, t_2)$ implies that terms t_1 and t_2 are *synonyms*. This relationship gives us a partition of set T into subsets of synonyms, i.e. all terms in a subset are synonyms. If we set a unique surrogate for every set of synonyms we can identify two kinds of terms:

Descriptors they are unique terms and a descriptor cannot be synonym of another descriptor.

Ascriptors they are those terms wich are not descriptors (therefore, they are synonyms of descriptors).

These subsets of synonyms are called *synsets* in WordNet [George A. Miller et al., 1993]. The point here is that we can replace any term by its equivalent descriptor in order to decrease ambiguity. Grossman and Frieder identify another relationship called *generalization*, showing interesting properties of this relation when querying. This generalization is a broader concept for the more accurated *hyponomy* and *meronymy* relationships in WordNet.

The inclusion of the thesaurus in the model is relevant in that it is being used for many purposes related to the retrieval task. Its use for query expansion as proposed by Vassilevskaya [Vassilevskaya, 2002], and for document clustering [Ralf Steinberger et al., 2000], among other applications.

Nevertheless, there is a great difficulty when using this model to describe classic retrieval strategies like vector space, probabilistic and boolean models (no too much in the last one, though). The reason is that current strategies are certainly *ranking strategies* rather than mechanisms to return a finite set of documents as this model states. Hence, it is used when introducing information retrieval concepts but has few repercusion on the rest of their book.

Another example, a little bit more detailed, is the model proposed by Sheridan and Schäuble [Sheridan et al., 1997]. They use the tuple

$$\langle T, \Phi, D; ff, df \rangle \tag{3}$$

where

T is the set of possible terms in a document

Φ is the set of indexing features (lemmatized/stemmed terms)

$\phi : T \to \Phi, \tau \mapsto \phi(\tau) := \phi_i$

D is the set of documents

$d : T \to D, \tau \mapsto d(\tau) := d_j$

$\tau \in T, \phi_i \in \Phi$ (ϕ_i is the lemmatized version of term τ)

ff is the frequency of an indexing feature in a document

$ff(\phi_i, d_j) = |\{\tau \in T | \phi(\tau) = \phi_i \wedge d(\tau) = d_j\}|$

df is the document frequency of an indexing feature

$df(\phi_i) = |\{d_j \in D | \exists \tau \in T : \phi(\tau) = phi_i \wedge d(\tau) = d_j\}|$

Retrieval strategies are applied on this structure in order to effectively complete the IR model. This structure is quite interesting when modeling *dual structures* oriented toward the generation of *similarity thesauri*. But for the purpose of being an strong mathematical framework it resembles more a list of unconnected pieces. There is no component which actually maps queries to documents, and we find that as a major lack, so we cannot fully qualify this structure as a modelization for an IR system.

A good model should be used whenever is possible to allow the "user of the model" understand how the different approaches and strategies fit into the same information retrieval framework. So we find that the model, in its expressiveness, must be well balanced, being a compromising between operability in formal manipulation and depth of modeling. *Depth of modeling* refers to how far the model has gone when representing as a mathematical structure a component used in a particular implementation of a retrieval strategy.

In defining our own model, this balance must be preserved.

Now, we present the model proposed by Baeza-Yates and others [Baeza-Yates and Ribeiro-Neto, 1999]. This model is richer than the ones showed above in that it uses a *ranking* function, so it is actually closer to current retrieval strategies.

For them, an *Information Retrival Model* is a quadruple

$$\langle D, Q, F, R \rangle \qquad (4)$$

where

D is the set of document representations

Q is the set of queries

F is the *framework* for modeling documents, queries and their relationship

R is the ranking function:

$R : Q \times D \to \Re, \langle q_i, d_j \rangle \mapsto R(q_i, d_j) := r_{ij} \in \Re$

The flexibility of the model resides in the *framework* component. This can be the vectorial space with its operators, the set algebra for the boolean model, or any other framework used to model the strategy. This model is complete, in its conception, but too general in practice. So general that the authors do not use it, they just define it for pedagogical purposes.

Sándor Dominich has an extensive work on formalization of IR models ([Dominich, 2000a] and [Dominich, 2000b]). IR systems are studied from a mathematical point of view and at each on his papers some interesting theorems are stated. He proposes a valid framework for any classical information retrieaval model (*vector space, probabilistic* and *boolean* models). We will have a look into the description of this model, because is the most rigurous we found so far.

First of all, in order to clarify the formalism used later, we introduce the following concepts identified by Dominich:

Identifiers they are any piece of information used to describe a document (index terms, keywords, descriptors...).

Objects It is any piece of information suitable to compound a document (text, images, sound fragments...). Can be, of course, the document itself.

Documents Therefore, a document is, indeed, cluster of objects. In many cases, when collections are made up by just full text documents, a document contains only one object: its text. For this reason several models may collapse these two elements into one.

Criterias They reflect a weighted relationship between two documents (e.g.: similarity, relevance, distance...).

Threshold This component is used when defining the retrieval model. It states a real value used as a cut in criterias values, giving a set of documents satisfying the criteria above that threshold.

Retrieval The retrieval is a mapping from a document to a set of documents.

Let's use the mathematical notation to express all this, let

1. $T = \{t_1, t_2, ..., t_k, ..., t_N\}$ be finite set of *identifiers*, $N \geq 1$,
2. $O = \{o_1, o_2, ..., o_u, ..., o_U\}$ be finite set of *objects*, $U \geq 1$,
3. $(D_j)_{j \in J = \{1,2,...,M\}}$ be a set of *object clusters*, $D_j \in 2^O, M \geq 2$,
4. $D = \{\tilde{o}_j | j \in J\}$ be a set of *documents* where the normalize fuzzy set $\tilde{o}_j = \{(t_k, \mu_{\tilde{o}_j}(t_k)) | t_k \in T, k = 1, 2, ..., N\}, j = 1, 2, ..., M$,

   $\mu_{\tilde{o}_j} : T \to S \subseteq [0, 1] \subset \mathbf{R}$, is a *cluster representative* of object cluster D_j,
5. $A = \{\tilde{a}\}_1, \tilde{a}\}_2, ..., \tilde{a}\}_i, ..., \tilde{a}\}_C\}$ be a finite set of *criteria*, $C \geq 1$, where $\tilde{a}_i = \{((q, \tilde{0}_j), \mu_{\tilde{a}_i}(q, \tilde{o}_k)) | \tilde{o}_j \in D, j = 1, 2, ..., M\}, i = 1, 2, ..., C$, is a normalized fuzzy relation.

   $\mu_{\tilde{a}_i} : D \times D \to [0, 1] \subset \mathbf{R}.q \in D$ arbitrary fixed.

6. $a_{\alpha_i} = \{\tilde{o} \in D | \mu_{\tilde{a}}(q, \tilde{o}) \ > \alpha_i\}, i = 1, 2, .., C$, be a α_i*-cut of criterion* $\tilde{a}_i$. $0 \leq \alpha_i < +\infty$. $q \in D$ arbitrary fixed.
7. $\Re : D \to 2^D$ be a mapping called *retrieval.*

He defines a *Classical Information Retrieval* (CIR) as a system composed by a collection of documents and a retrieval mapping in a 2-tuple:

$$\langle D, \Re \rangle \tag{5}$$

with following properties:

P1. $q = \tilde{o} \Rightarrow \mu_{\tilde{a}_i}(q, \tilde{o}) = 1, \forall q, \tilde{o} \in D, i = 1, 2, ..., C$. This is the so called *reflexivity* property.

P2. $\Re(q) = \{\tilde{o} | \mu_{\tilde{a}_i}(q, \tilde{o}) = max_{k=1,...,C} \mu_{\tilde{a}_k}(q, \tilde{o})\} \bigcap a_{\alpha_i}, i$ arbitrary fixed

The first property only states that, in the case the document is equal to the query, then any criterion must return 1 as value. The second property states that, fixed one criterion arbitrary, the retrieval will be an intersection between two sets: one for those documents with a weight set by the criterion over the given threshold (α_i) and another for those documents that have a weight with the given criterion always higher than the weight returned by any other criteria. A graphic representation of the second property can be seen at figure 1.

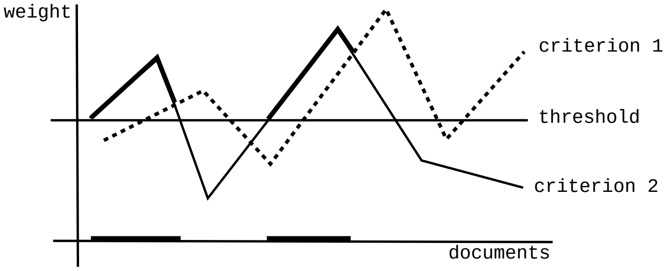

Figure 1: Relevance in a Classical Information Retrieval system

Here, for a given query, using the criterion 2, documents with the widest line as shown would be selected. Using this formalism, it is easy to define vector space and probabilistic models. We will not go deeper here, but we will refer to another work of Dominich which modelizes how the user is related to an information retrieval system. In this work [Dominich, 2001] formal grammar and languages are applied to define another information retrieval model. We will just show here how information need of the user is modelized:

$$IR = m[\Re(O, (Q, \langle I, \vdash \rangle))] \tag{6}$$

where

O is the set of objects to be retrieved (documents)

Q is the set of queries

I is the information we now about the user

$\vdash$ is the information derivable (deductible) from user information I

$\Re$ is a relationship between objects and *information need.* The information need is, therefore:

$IN = (Q, \langle I, \vdash \rangle)$

i.e. queries, user information and deducted information from user information by applying certain rules.

m represents that the relation $\Re$ is stablished with some uncertainty.

We can see that the model formalizes the so called *user profiles*, since we store personalized information about the user in order to infer additional information when specifing his/her *information need.*

A grammar with a language is used as means to represent documents and queries in *disjuntive normal form.* Hence, both can be represented as a boolean expression composed by terms and logical operators ($\vee, \wedge$ and $\neg$).

Unfourtunaly this model is not related directly with the previous one. Also certain concepts like the thesaurus are not modelized in these two proposals.

In the work of Tague and others [Tague et al., 1991] an extensive use of grammars and hypergraphs is proposed to deal with high structured texts. This model proposes a first step of segmentation of documents into parts with properties and operations related to each type of part (keyword, paragraph, reference...). This segmentation follows a grammar described in extended BNF notation (called *constraint schema*).

Once the document is segmented we have a parse tree following the former grammar. This grammar is converted into other grammars for indexing purposes. Relations between parts are defined by edges which act as *specialization, generalization* and *aggregation* relationships in the resulting parse tree (which becomes a hypergraph). The nodes are the parts of the document. After indexing we can perform complex queries like *give me the paragraph where keyword 'cat' appears*, *give me the reference which is of first series*, or the even more complex *give me style of cause where defendant contains 'cola'.*

This is a very detailed model, since each part has a distinctive relevance (properties and operations), therefore the richness of the query language used is higher than for other models. But there is a main problem with it: its level of specialization. A constraint of the model is that we must be able to parse documents in a deterministic way in order to produce the needed hypergraph. When dealing with unestructured text implementing a segmentation is not trivial. Nevertheless, this model states the advantages when this option is feasible. It is also a good reference work to state which are those minimal components which make up the retrieval system. They identify:

1. Original documents: as a compendium of fixed attributes (metadata) and natural language text.
2. Surrogates of original documents: abstracts and keyword lists.
3. References: internal (to other parts in the text) or external (to other documents)
4. Thesauri
5. Queries
6. Operations: selection, ranking and browsing

There are some operations missing in the model like *filtering* and *routing* which are mentioned by Grossman and Frieder, and other authors.

3 Considerations

Some considerations can be deducted so far:

1. We have to dissect the components deeply enough to reflect all the possible objects involved along with their interactions (relations). In this way we will be in conditions to model any strategy onto our scheme. I.e., model 1 is so generic that we have to supply more mathematical machinery to define any strategy in a formal way.
2. We must avoid to go too far. As opposite to the former point, we cannot increase the granularity of relationships. In equation 3 we have already specify the attributes "frequency of a term in a document" and "document frequency". Most of the models need more than just these two attributes.
3. Later on, we will formalize these components into a mathematical model which must be able to abstract, at least, classical information retrieval systems.
4. Hence, one of the first steps is to state which are the fundamental components of an information retrieval system. Some components that seem to be mandatory in the model are:
   - Documents
   - Terms
   - Queries
   - Relation between queries and terms (the *retrieval* component).
5. We will modelize also the thesaurus, due to the relevance of this item in some actual systems. We must be able to modelize effectively some of the applications in the use of thesauri within an information retrieval system.

4 A new model

We propose here a new formal representation for information retrieval systems. These model is similar to former ones but enfasizes the use of a ranking function as nexus between documents and queries, along with a more flexible definition of *document* and *query* components, placing them under the common concept of *text.* We state as a premise that there is not underlaying structure within a document, that is, documents are sets of terms (so this model omits segmentation).

An **Information Retrieval System** can be modelized as the tuple

$$\langle D, Q, T, r \rangle \tag{7}$$

where

D is the set of documents in the collection, $D \subseteq 2^T$, that is, a document is a set of terms.

Q is the set of queries, $Q \subseteq 2^T$, hence, also a query is a set of terms.

T is the set of terms whichfrom documents and queries are composed.

r is the ranking function. $r : 2^T \times 2^T \rightarrow \Re$. As we can see, we define the domain of this function as any pair of sets of terms, therefore, we can assume a rank value between two documents, two queries or a document and a query.

$r(x, y) = z; x, y \in 2^T \wedge z \in \Re$

With the following properties:

1. $r(x, x) = 1, \forall x \in 2^T$ (*reflexivity*)
2. $r(x, y) = r(y, x), \forall x, y \in 2^T$ (*symmetric*)

This simple model generalizes those models where a query retrieves a set of n documents, and the fact that also documents can be used as queries. Since the mapping of the retrieval is a *function* and the image any pair *(text, text)* (being *text* any document or query) we find that the set of documents turns into a *full ordered set* as follows:

$$\langle D, \leq_{r_x} \rangle \tag{8}$$

where $\leq_{r_x}$ relation is defined as

$$a, b \in 2^T, a \leq_{r_x} b \Leftrightarrow r(a, x) \leq r(b, x) \tag{9}$$

Now we can define a new set called **retrieval set** which will give us the set of those n documents with the higher rank value for a given document or query:

Given

- $n \in \mathbf{N}$ a fixed natural value
- $x \in 2^T$ a given text used as query
- $I = \{I_1, I_2, ..., I_m\}$ a *partition* of D

we define the *retrieval set* R_x^n as

$$R_x^n = I_k \in I \text{ such as } (|I_k| = n) \wedge (d_i >_{r_x} d_j), \forall d_i \in I_k, \forall d_j \in \langle D - I_k, \leq_{r_x} \rangle \quad (10)$$

We can now define a **keyword assigner** (also know as *keyword enhancer or descriptor indexer*) as the tuple

$$\langle W, D, \rho \rangle \quad (11)$$

where

W is the set of descriptors (keywords)

D is the set of labeled documents (documents with keywords assigned)

ρ is the assignment function. This function is a mapping

$\rho : D \to 2^W; d_i \mapsto \rho(d_i) = \rho_i \in 2^W$

That is, ρ takes a document as argument and produces a set of keywords belonging to the controlled vocabulary (thesaurus) W.

Note that the thesaurus has been simplified omitting relations like generalization, synonymy, etc. Here we consider the thesaurus nothing but a list of controlled terms. However, it can be extended to exhibits those relationships, but with the given description we can already modelize classical models as described briefly in next section.

4.1 Classical models

It is not difficult to find that when representing classical models we only have to specify an appropiate ranking function to each of them. For the **vector space model** this ranking function is given by the *cosine similarity* between the vectors of a query and the documents in the collection. We will have to define the additional functions of *term frequency* ($tf(t, d)$ and *inverse document frequency* ($idf(t)$). For example, if we set to 1 the weight of a term in the query vector we can define the ranking function as follows:

$$r(d, q) = \frac{\sum_{t \in d \cap q} tf(t, d) \cdot idf(t)}{\sqrt{\sum_{t \in d}(tf(t, d) \cdot idf(t))^2}} \tag{12}$$

The **probabilistic model** details the ranking function based on the *number documents related to each term* over the *total number of documents* in the collection. Depending on the specific formula used additional values may be defined.

Finally, the **boolean model** can be easily modelized using the *normal disjuntive form* as defined in [Baeza-Yates and Ribeiro-Neto, 1999, page 26]. We would get 1 for documents satisfying the query or 0 when the query is not satisfied by the document. The retrieval set should be, therefore, redefined as follows:

$$R_q = \{d \in D | r(d, q) = 1\} \tag{13}$$

Every valid document is returned and r is not a real ranking function, but a discrimination function.

We will not go deeper here, since our main goal is to show how some applications like *crosslingual queries* and *automatic assingment* of descriptors can also be modelized within the model.

4.2 Crosslingual queries

The work of Steinberger and others [Ralf Steinberger et al., 2002] in the use of a multilingual thesaurus for crosslingual queries is a prominent application in the use of keyworded collections. It states basically that if we have two different collections of documents D_1 and D_2 in different languages but there is a mapping to a common set of terms W, then we can stablish a *crosslingual ranking function* using the thesaurus as nexus. The computation of the rank value is more complex than the approach we propose here, which we have choosen just to show how the model can reflects the described paradigm:

Given two retrieval systems IR_1 and IR_2 and two mappings M_1 and M_2 such as

$IR_1 = \langle D_1, Q_1, T_1, r_1 \rangle$; $IR_2 = \langle D_2, Q_2, T_2, r_2 \rangle$

$M_1 = \langle W, D_1, \rho_1 \rangle$; $M_2 = \langle W, D_2, \rho_2 \rangle$

we can stablish a new **crosslingual rank** r' as follows

$$r'(q_1, d_2) = \sum_{\forall d_i \in D_1} (\sum_{\rho_1(d_i) \cap \rho_2(d_2)} r(q_1, d_i)) \tag{14}$$

As we can see, thanks to a common set of descriptors W we can break the barrier of language.

4.3 Automatic assignment

Another advantage of this notation is that it enables us to modelized without effort a simple automatic indexing approach like the one proposed by [Ezhela et al., 2001]. Given

$\langle D, Q, T, r \rangle$ and $\langle W, D', \rho \rangle$ where $D' \subset D$, we can define a new assigner over the whole collection $\forall d_i \in D$ as

$$\rho'(d_i) = \begin{cases} \rho(d_i) & \text{if } d_i \in D' \\ \bigcap_{d_j \in R^n_{d_i}} \rho(d_j) & \text{otherwise} \end{cases} \tag{15}$$

Therefore, for those documents in the domain of ρ (D'), we just return the set of descriptor defined by the mapping. For those where the mapping is not defined, we intersect the keywords of the n documents which would be retrieved using the document to be indexed as a query.

5 Conclusions

Some formal methods have been reviewed, emphasizing their richness when used as general models for IR. This has allowed us to identify common components and relevant relations.

We have defined a model which allows the use of both queries and documents as queries. This model generalizes the domain of the ranking function and stablishes how the retrieval set can be built given a natural number and a given query, which simplifies many proposals published so far. It is not too specific, since the ranking function is just defined in its range and domain, leaving the calculation of the value to be defined by each specific model, as shown for classical models.

The model has been used here to modelize an approach for crosslingual searching and for keyword assignment, showing that we can formalize some of the current approaches in automatic assignment of keywords, stating a valid framework to formalize these methods (which lack nowadays of a consistent notation for their modelization).

References

[Baeza-Yates and Ribeiro-Neto, 1999] Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval.* Acm Press Series.

[Bartell et al., 1990] Bartell, B. T., Cottrell, G. W., and Belew., R. K. (1990). Latent semantic indexing is an optimal special case of multidimensional scaling.

[Belew, 1989] Belew, R. K. (1989). Adaptive information retrieval: Using a connectionist representation to retrieve and learn about documents. In *Proceedings of the Twelfth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Artificial Intelligence/Connectionism, pages 11–20.

[Chen, 1995] Chen, H. (1995). Machine learning for information retrieval: neural networks, symbolic learning and genetic algorithms. *J. Am. Soc. Inf. Sci.*, 46(3):124–216.

[Dominich, 2000a] Dominich, S. (2000a). Formal foundation of information retrieval. In *Proceedings of the ACM SIGIR MF/IR*, pages 8–15.

[Dominich, 2000b] Dominich, S. (2000b). A unified mathematical definition of classical information retrieval. *Journal of the American Society for Information Science*, 51(7):614–625.

[Dominich, 2001] Dominich, S. (2001). On applying formal grammar and languages, and deduction to information retrieval modelling. In *Proceedings of the ACM SIGIR MF/IR*, pages 37–41.

[Ezhela et al., 2001] Ezhela, V. et al. (2001). Citations as a mean for discovery and automatic indexing of the scientific texts with new knowledge for a given subject.

[George A. Miller et al., 1993] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller (1993). Introduction to wordnet: An on-line lexical database.

[Grossman and Frieder, 1998] Grossman, D. A. and Frieder, O. (1998). *Information Retrieval, Algorithms and Heuristics.* Kluwer Academic Publishers.

[Ralf Steinberger et al., 2002] Ralf Steinberger, Bruno Pouliquen, and Johan Hagman (2002). Cross-lingual Document Similarity Calculation Using the Multilingual Thesaurus EUROVOC. *Third International Conference on Intelligent Text Processing and Computational Linguistics.*

[Ralf Steinberger et al., 2000] Ralf Steinberger, Johan Hagman, and Stefan Scheer (2000). Using thesauri for automatic indexing and for the visualisation of multilingual document collections. pages 130–141.

[Salton et al., 1974] Salton, G., Wong, A., and Yang, C. S. (1974). A vector space model for automatic indexing. Technical Report TR74-218, Cornell University, Computer Science Department.

[Sheridan et al., 1997] Sheridan, P., Braschler, M., and Schäuble, P. (1997). Cross-language information retrieval in a multi-lingual legal domain. In Peters, C. and Thanos, C., editors, *Proceedings of ECDL-97, 1st European Conference on Research and Advanced Technology for Digital Libraries*, pages 253–268, Pisa, Italy.

[Tague et al., 1991] Tague, J., Salminen, A., and McClellan, C. (1991). Complete formal model for information retrieval systems. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 14–20. ACM Press.

[van Rijsbergen, 1989] van Rijsbergen, C. J. (1989). Towards an Information Logic. In Belkin, N. J. and van Rijsbergen, C. J., editors, *Proceedings of the 12th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 77–86, Cambridge, Massachusetts. ACM Press.

[van Rijsbergen, 2000] van Rijsbergen, C. J. (2000). Another look at the logical uncertainty principle. *Information Retrieval*, 2(1):17–26.

[Vassilevskaya, 2002] Vassilevskaya, L. A. (2002). An approach to automatic indexing of scientific publications in high energy physics for database spires-hep. Master Thesis.