An Extended Fuzzy Boolean Model of Information Retrieval Revisited

Sławomir Zadrożny
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01–447 Warsaw, Poland
zadrozny@ibspan.waw.pl

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01–447 Warsaw, Poland
kacprzyk@ibspan.waw.pl

***Abstract*— An extension to the classical Boolean model of information retrieval is discussed. The approach is based on recent advances in the area of fuzzy logic in a narrow sense. A strictly formal logical interpretation is provided for all elements of the model including the representation of both documents and queries and the evaluation of queries.**

I. Introduction

Research on information retrieval has a long history. Traditionally three models of retrieval are assumed in the literature: Boolean, vector space and probabilistic. Many variations and extensions to these models have been proposed. In practice the first one seems to be the most widely employed in commercial information retrieval systems. In its classical form it offers a strong querying language based logical concepts and at the same time a rather inflexible document representation. The vector space model appears to be complementary in this respect. Hence, the combination of them should be beneficial. The re-interpretation of the Boolean model should provide a sound theoretical background for such a combination.

The idea of the fuzzy logic based information retrieval system has been studied and developed by many authors, cf., e.g., [1], [2], [3], [4], [5], [6], [7], [8], [9]. However, new advancements in the area of fuzzy logic in a narrow sense exemplified by the work of Hajek [10] or Novak and Perfilieva [11], [12], [13] open new possibilities for a consistent logical treatment of all the elements of the information retrieval model.

Basically, a model of information retrieval should address at least three aspects:

- representation of documents
- representation of queries
- methods used to evaluate the matching of a query against a document.

In this paper[1] we look for the interpretation of all this aspects in terms of fuzzy propositional logic. We combine concepts and solutions proposed previously in the literature with some original contributions. Our goal is to provide for a unified interpretation of the retrieval process in terms of fuzzy logic rather than to offer specific solutions regarding particular aspects of this process, e.g., new weighting scheme.

[1]Research supported by the KBN Grant 3 T11C 052 27

The paper is composed as follows. Section II presents the classical Boolean model of information retrieval. We try to formalize all elements of the model in the spirit of the classical propositional calculus. We also briefly discuss some other logical approaches that may be relevant for our purposes. In Section III fuzzy propositional logics are discussed, mainly from the perspective of [12], [13], [10]. The first reference is in a sense (for our purposes) a "naive" one, directly emerging from fuzzy sets theory. We are less rigorous in its presentation. The second reference is a more sophisticated logic which provides a theoretical background for a more comprehensive extension of the classical Boolean model. Finally, in Section IV we demonstrate how elements of extended Boolean model may be expressed in terms of fuzzy propositional logic discussed in the previous section.

II. The classical Boolean model of Information Retrieval

We will briefly characterize the classical Boolean model in terms of the document and query representation, as well as of the query evaluation. This will be a starting point for the extension of the model towards its fuzzy counterpart. In the paper we assume the following notation:

$$D = \{d_i\}_{i=1,N} \quad - \quad \text{a set of text documents}$$
$$T = \{t_j\}_{j=1,M} \quad - \quad \text{a set of index terms}$$

In what follows we will denote by d both a document and its representation and the proper interpretation of d should be clear from the context. In the Boolean model a document is represented by a set of index terms assigned to it. It may be specified formally in several ways. In a most straightforward way, a document may be identified with a set of index terms, i.e.,:

$$d_i = \{t_k\}_{k=1,K} \qquad d_i \in D, t_k \in T$$

Second, such a set may be equivalently specified by its characteristic function or, more practically, by a binary vector with coordinates corresponding to the particular index terms. This representation may be further generalized, towards the formalism typical for the vector space model, introducing a function F such that:

$$F : D \times T \longrightarrow \{0, 1\} \tag{1}$$

Still another propositional logic based formalisation of the Boolean representation of documents is possible, and it is more appropriate for the logical character of the model. Namely, we associate with each index term t_j a propositional variable s_j. Then, a document is represented by a propositional formula being a conjunction of propositional variables corresponding to the index terms assigned to this document, and the negations of propositional variables corresponding to the rest of index terms. For example, let a set of index terms consist of "computer", "internet", "network", "journal" and "magazine". Then, a document d to which the first three of them are assigned is be represented as:

$$s_1 \wedge s_2 \wedge s_3 \wedge \neg s_4 \wedge \neg s_5 \tag{2}$$

assuming that the propositional variables $s_1, \ldots s_5$ are associated with the subsequent index terms. Such an approach opens new vistas for a more sophisticated representations of documents in which not just a conjunction of index terms, but more complex formulae are also applicable. This is in the spirit of the logical approach to the modelling of information retrieval initiated by van Rijsbergen [14], [15]. We will dwell on it just after discussing the evaluation mechanism of the Boolean model. Moreover, we will then offer still another view on the representation of a document in the framework of the Boolean model.

In the classical Boolean model the query is a formula in the sense of propositional logic. Again, the propositional variables in such a formula represent the particular index terms, and may be combined using the logical connectives, notably the negation, conjunction and disjunction. For example, a query: "*Find documents on computer or network*" is represented, assuming the same correspondence between propositional variables and index terms as in (2), as:

$$s_1 \vee s_3 \tag{3}$$

Finally, the third element of the Boolean model, i.e., the query evaluation mechanism refers to the truth or validity of the formula representing the query. Informally, for a given document d, a propositional variable s_j of a query q is assigned a truth value "true" if the corresponding index term appears in the representation of the document d, and is assigned the value "false" otherwise. Then, the truth degree for the whole formula/query is computed as usually in the framework of the propositional logic semantics, i.e., exploiting truth functionality of the logical connectives. The document is deemed to be relevant to the query if and only if the formula evaluation yields "true".

More formally, in the spirit of van Rijsbergen's [14], [15] approach, the issue of matching query q against document d is considered as related to the truth of (or the probability of the truth of) formula $d \Rightarrow q$. Sebastiani [16] discussed several possible, and studied in the literature, interpretations of $d \Rightarrow q$ as an indicator of the relevance of document d for query q. Namely, [16], in order to declare the matching one may require that:

1) $d \Rightarrow q$ is true under some particular valuation of propositional variables s_j,
2) q is semantically implied by d ($d \models q$),
3) $d \Rightarrow q$ is a tautology ($\models d \Rightarrow q$),
4) q is provable from d ($d \vdash q$),
5) $d \Rightarrow q$ is a theorem ($\vdash d \Rightarrow q$)

(4)

In the specific context of the classical propositional logic, the interpretations 2) – 5) coincide, while the interpretation 1) does not seem to make much sense.

For our purposes, we assume an equivalent for of the interpretations 2) —5) but apparently more natural understanding of the relevance based on a slightly different representation of documents. Namely, a document might be understood not as a formula of propositional logic, but as a valuation in the sense of semantics of this logic (i.e., as an assignment of truth values to propositional variables). More specifically, such a *valuation* v_d (subindex d will be used to indicate that this valuation is implied by document d) directly corresponds, informally speaking, to the above mechanism for the Boolean query evaluation, i.e.:

$$v_d(s_j) = \begin{cases} \text{true} & \text{if } \ t_j \in d \\ \text{false} & \text{otherwise} \end{cases} \tag{5}$$

Then, query q matches document d if it is true under valuation v_d.

The classical Boolean model is criticized for several reasons. The most obvious problem lies in the binary nature of document representation. It is widely advocated that the index terms might have varying importance for the representation of a document, while in the classical Boolean model an index term may be only either completely relevant or completely irrelevant. In terms of our interpretation of this representation (cf. (5)), this boils down to the limitations of the classical, two-valued truth structure given by the set {true,false} (or, {0,1}). Thus, a remedy may be to redefine the classical Boolean model in terms of a multivalued logic. Before doing that in Section IV, we first describe such a logic in Section III.

III. Fuzzy propositional logics

In order to extend the classical Boolean model we need to adopt a version of the *fuzzy propositional logic*. Actually, depending on the scope of this extension (discussed in the next section) we need a more or less elaborated version of such a logic. Let us start with a simple, "naive" version, traditionally related to fuzzy sets theory. Moreover, we will confine ourselves mainly to the semantic aspects of such a logic.

The language of this simple version of fuzzy logic is identical with the classical one. Semantics is based on the structure:

$$\mathcal{L} = ([0, 1], \max, \min, \neg, \rightarrow) \tag{6}$$

where particular operators correspond to the disjunction, conjunction, negation and implication, respectively. The negation and implication operators are ussually assumed to be $\neg x = 1 - x$ and $x \rightarrow y = \max(1 - x, y)$, respectively. Other forms of the operators are also possible. Now, a valuation υ assigns to each propositional variable a number from the interval $[0, 1]$ and compound formulae are valuated employing the indicated correspondence between the logical connectives and the operators listed in the structure $\mathcal{L}$.

Such a fuzzy propositional logic will be sufficient for a simple, straightforward extension of the Boolean model shown in the next section. For a more comprehensive extension of the Boolean model a more elaborate logic will be needed. For this purpose we adopt a fuzzy propositional logic advocated by Perfilieva and Novak [12], [13] (cf., also Hajek's presentation of Rational Pavelka logic in [10]). In this approach a more general structure of truth values than the one given by (6) is considered. It is a so-called complete residuated lattice:

$$\mathcal{L} = (L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1}) \quad (7)$$

which is a generalization of the conventional Boolean algebra used in classical logic. It is equipped with four binary operations: two lattice operations $\wedge$ and $\vee$, multiplication ($\otimes$) and residuation ($\rightarrow$). In general, for $L = [0, 1]$ the equivalences among various interpretations of $d \Rightarrow q$ listed in (4) as an indicator of relevance do not hold. In order to preserve them (i.e., to have soundness and completeness of this fuzzy propositional logic) some constraints have to be imposed on the choice of some operators of (7); cf. [12], [13].

Basically, J, a formal language of classical propositional logic, and a set, $\mathcal{P}_J$, of well-formed formulas over it, are assumed as a point of departure. The language, similarly to the classical case, consists of:

- a countable set of propositional variables $s_1, s_2, \ldots$
- a set of logical constants $\{\mathbf{a} \mid a \in L\}$ (including $\perp$ and $\top$ corresponding to $\mathbf{0}$ and $\mathbf{1}$, respectively)
- symbol of the logical connective of implication $\Rightarrow$
- brackets as auxiliary symbols.

(we will use letters $p, q, \ldots$ to denote compound formulae composed in a proper way of propositional variables and logical connectives).

Other logical connectives in addition to the implication may be derived in a usual way. An important extension consists in adding to the alphabet of language J logical constants for all the truth values of L. They will be treated themselves as elementary formulae (besides classical elementary formulae composed of a single propositional variable). Moreover, the concept of an evaluated formula is introduced as a pair (p, a) of a well-formed formula p and its syntactic evaluation a. The intuition behind the concept of an evaluated formula is such that it requires a formula p to be true to a degree equal at least a. In fact, such an evalutaed formula may be interpreted as a regular formula $\mathbf{a} \Rightarrow p$. This interpretation will be very useful for our purposes.

We skip other proof-theoretical aspects of this logic (inference rules, logical axioms) as we are mainly interested in semantics.

A truth valuation, υ, of formulae of language J is done again similarly as in the classical case:

$$\upsilon : \mathcal{P}_J \longrightarrow L \quad (8)$$

For an elementary formula a value is assigned directly and for a compound one by employing the truth-functionality of the logical connectives. In particular:

$$\upsilon(p \Rightarrow q) = \upsilon(p) \rightarrow \upsilon(q) \quad (9)$$
$$\upsilon(p \vee q) = \upsilon(p) \vee \upsilon(q) \quad (10)$$
$$\upsilon(p \wedge q) = \upsilon(p) \wedge \upsilon(q) \quad (11)$$
$$\upsilon(\mathbf{a}) = a \quad (12)$$

The truth values come now from $[0, 1]$ (in general, from L) instead of $\{0, 1\}$. New, special atomic formulae build of logical constants alone, $\mathbf{a}$, are always assigned a truth value a (i.e. a truth value which the logical constant $\mathbf{a}$ corresponds to).

IV. An extended fuzzy Boolean model

In Section II we have expressed the classical Boolean model of information retrieval in terms of the classical propositional logic. Now, we will re-interprete it in the context of fuzzy propositional logics discussed in the previous section. We will first employ the "naive" version of this logic what makes it possible to consider varying importance of particular index terms in the representation of documents. The second, more elaborate version of the fuzzy propositional logic provides a formal ground for the interpretation of importance weights both in documents and queries. This way, the advantages of vector space model may be incorporated into the Boolean model having more convincing theoretical foundations.

A. Weights in documents

First, we offer in the logical framework a formal treatment of importance weights assigned to index terms in the representation of documents. Thus, we assume the function F given by (1) to be now identical with the one used in the vector space model, i.e.,:

$$F : D \times T \longrightarrow [0, 1] \quad (13)$$

A popular choice for F might be a well-known $tf \times IDF$ scheme. Now, we adapt (5) to the fuzzy case. Hence, a document d is interpreted as a valuation υ_d such that:

$$\upsilon_d(s_j) = F(d, t_j) \quad (14)$$

The form of a query is preserved from the classical Boolean model, i.e., it is a formula p of language $\mathcal{P}_J$, exemplified by (3). The evaluation of a query for a document is obtained as the truth value of p under valuation υ_d. Thus, we immediately obtain flexible representation of documents and matching degree of a document against a query from the interval [0,1] (from L, more generally).

B. Weights in queries

Importance weights in the representation of a document make it possible to indicate which index terms are more, and which less relevant for this document. Similarly, a different importance may be assigned to various index terms or their combinations in a query. In order to represent such importance weights in the framework of our logical model we will exploit the logical constants and the evaluated syntax of the fuzzy propositional logic assumed (cf., Section III).

There is a fairly extensive literature on the topic. We will mainly refer to the work of Buell and Kraft [2], Dubois and Prade [17], Roubens and Fodor [18], Bordogna and Pasi [19] and Yager [4], but many others contributed too. We will focus on a query represented by a conjunction of propositional variables, i.e.:

$$q = s_1 \wedge \ldots \wedge s_K \tag{15}$$

and this may be easily generalized to a query where each s_j denotes any formula, not necessarily elementary one.

Next, let us assume that each index term t_j is assigned an importance weight w_j; $w_j \in [0, 1]$ and $\max_j w_j = 1$. Let us denote such a "weighted query" as q_w. Before we show how to express the weights within the logical framework, first, let us discuss how the importance weights in queries should be interpreted.

In the literature, cf., e.g., [19], three interpretations of the query weights w_j are considered:

1) relative importance,
2) thresholds of importance, and
3) ideal weights

According to the first interpretation, weight w_j of term t_j in a query indicates to what extent the appearance of an index term t_j in a document is important for the document to satisfy the query. If the weight is low (close to 0), then the absence of the term t_j in a document (i.e., low, possibly equal 0, weight of this term in the document) does not exclude the matching of this document against the query. If the weight of a term in a query is high (close to 1), then the document has to contain the term (i.e., have a high weight assigned to this term) to qualify for matching the query.

Due to the second interpretation [1], [2], the weights of particular index terms in the documents sought have to be higher than threshold values expressed with importance weights w_j. There are further possible interpretations depending on how the undersatisfaction of query terms is treated - a further discussion is given below. Herrera-Viedma [20] proposed a modified interpretation of query weights in this interpretation. Namely, high query weights are treated as above, i.e., as lower bounds on the weights of index terms in the documents, but low weights are treated as upper bounds of these weights, thus requiring they are lower.

The third interpretation [7] is somehow analogous to that assumed in the vector space model as the documents sought should be characterized by weights of the index terms similar to those specified in the query.

These various interpretations may be to some extent uniformly expressed due to the results obtained in the area of multicriteria decision making, fuzzy querying of databases and fuzzy information retrieval. Dubois and Prade [17] formalized the importance weight modelling in a way leading to a concise logical representation. Moreover, this representation is directly applicable in the framework of fuzzy logic as presented in Section III.

Due to Dubois and Prade [17] approach (gathering, aggregating and elaborating on the previous research of their own and other authors, e.g., Yager [4]), the matching degree, $\phi(q_w, d)$, of the query q_w and a document d should be calculated as (using notation assumed in this paper):

$$\phi(q_w, d) = \tag{16}$$
$$(w_1 \rightarrow \upsilon_d(s_1)) \wedge (w_2 \rightarrow \upsilon_d(s_2)) \wedge \ldots \wedge (w_K \rightarrow \upsilon_d(s_K))$$

where $\rightarrow$ and $\wedge$ are operators used in (7).

Now, various residuation operators $\rightarrow$ of the complete residuated lattice (7) yield a different behaviour of importance weights.

For the Kleene-Dienes implication:

$$x \rightarrow y = \max(1 - x, y)$$

(assumed in the original approach of Dubois and Prade) we recover the behaviour of the relative importance interpretation: components of (16) with the minimal (w_j = 0) and maximal (w_j = 1) importance weights are superfluous and critical, respectively, for the satisfaction of the overall query, while those with intermediate importance weights are influencing the satisfaction to some extent.

For the Gödel implication:

$$x \rightarrow y = \begin{cases} 1 & \text{if } x \leq y \\ y & \text{otherwise} \end{cases}$$

and the Goguen implication:

$$x \rightarrow y = \min(y/x, 1)$$

we get threshold semantics for the importance weights. Namely, if index term t_j is related to a document to a degree higher or equal w_j, then the component of the query composed of t_j is satisfied to a degree equal 1. On the other hand, if the threshold w_j is not reached, then the matching degree falls below 1, according to the respective implication operator. The Goguen implication operator based formula provides for a continuous transition from the full satisfaction (degree equal 1) to lower degrees of matching.

Thus, semantics of the weighted queries have been studied in the literature very extensively and satisfactory solutions have been obtained. What we propose here is the syntax of importance weights in the queries which is consistent with the logical model of information retrieval. Thanks to the more sophisticated version of the fuzzy propositional logic discussed in Section III (cf. (7)) this syntactical extension is very natural and easy. Namely, a well-formed formula of this logic corresponding to the semantics expressed with (16) looks as follows:

$$q_w = (\mathbf{w}_1 \Rightarrow s_1) \wedge (\mathbf{w}_2 \Rightarrow s_2) \wedge \ldots \wedge (\mathbf{w}_K \Rightarrow s_K) \quad (17)$$

where $\mathbf{w}_i$'s are logical constants, while $\wedge$ denotes the conjunction connective whose semantics is provided by a lattice operator (7) denoted for simplicity with the same symbol $\wedge$.

Thus, we get a general syntactic counterpart of the semantic approaches to the modelling of importance weights in queries. This makes it possible to study various interpretations of van Rijsbergen's $d \Rightarrow q$ expression of relevance between a document d and a query q. This is especially important for logics lacking soundness or completeness properties.

In order to complete the formalisation of weighted queries in the logical framework we will briefly address two remaining problems: the syntax for importance weights meant as ideal weights and a query composed using the disjunction instead of the conjunction connectives. In both cases neither the formula (17) nor the expression (16) are valid.

In order to express the ideal weights interpretation of importance weights [19] we propose to replace (17) with:

$$q_w = (\mathbf{w}_1 \Leftrightarrow s_1) \wedge (\mathbf{w}_2 \Leftrightarrow s_2) \wedge \ldots \wedge (\mathbf{w}_K \Leftrightarrow s_K) \quad (18)$$

where the equivalence connective $\Leftrightarrow$ is to be modelled by the following operator $\leftrightarrow$:

$$x \leftrightarrow y = \min(x \rightarrow y, y \rightarrow x)$$

and $\rightarrow$ is the Goguen implication operator.

The $\leftrightarrow$ operator should be then treated as derived from other operators defined in the lattice (7). It may be easily verified that:

$$x \leftrightarrow y = \begin{cases} 1 & \text{if } x = y \\ x/y & \text{if } y > x \\ y/x & \text{if } y < x \end{cases}$$

and, thus, we obtain a reasonable characterization of the ideal weights interpretation.

It should be emphasized that a proper interpretation of importance weight should be carefully adopted as this does really makes a considerable difference. It may be illustrated on the following small example. Let us consider 3 documents and 3 index terms characterizing them in the following way:

	t_1	t_2	t_3
d_1	1.0	0.8	0.7
d_2	0.5	1.0	0.6
d_3	0.3	1.0	0.45

and a query in the form:

$$q = (s_1, 0.3) \wedge (s_2, 1.0) \wedge (s_3, 0.5)$$

We do not use here the notation corresponding to (17) as we want to take into account also an ideal weight interpretation that does not fit this scheme. Thus, in the above informal notation (s_j, w_j) denotes a propositional variable s_j with the $\mathbf{w}_j$ constant assigned what formally should be read as $\mathbf{w}_j \leftrightarrow s_j$ in case of an ideal weight interpretation and $\mathbf{w}_j \rightarrow s_j$ otherwise.

Then, depending on the importance weights interpretation we obtain a very different result:

weight interpretation	*order of documents*
relative importance	$d_1 \succ d_2 \succ d_3$
thresholds of importance	$d_2 \succ d_3 \succ d_1$
ideal weights	$d_3 \succ d_2 \succ d_1$

where $\succ$ denotes the ordering imposed on documents according to their matching degree against the query q.

Finally, let us consider the problem of a query being a disjunction of weighted propositional variables. It turns out that it may be formally expressed as:

$$q_w = (\neg\mathbf{w}_1 \Rightarrow_c s_1) \vee (\neg\mathbf{w}_2 \Rightarrow_c s_2) \vee \ldots \vee (\neg\mathbf{w}_K \Rightarrow_c s_K) \quad (19)$$

where $\Rightarrow_c$ is a *coimplication* connective. This connective may be treated as a derived one whose semantics in the context of the lattice (7) may be provided by an operator $\rightarrow_c$ defined as follows [18]:

$$x \rightarrow_c y = \neg(\neg x \rightarrow \neg y)$$

Thus, for the Kleene-Dienes, Gödel and Goguen implication operators the corresponding coimplication operators are defined, respectively:

$$x \rightarrow_c y = \min(1 - x, y)$$

$$x \rightarrow_c y = \begin{cases} 0 & \text{if } x \geq y \\ y & \text{otherwise} \end{cases}$$

$$x \rightarrow_c y = \begin{cases} 0 & \text{if } x = 1 \\ \max\{0, \frac{y-x}{1-x}\} & \text{otherwise} \end{cases}$$

This completes our presentation of the logical formalism of weighted queries based on a fuzzy propositional logic.

C. Aggregation operators

Such an extended Boolean model as presented in the previous subsections may be further refined with respect to the logical connectives used. In the literature such extensions have been proposed (cf., e.g., [19]) and are motivated by a potential inadequacy of t-norm and t-conorm operators for the modelling of users' information needs. This may be exemplified by a query represented in our logical formalism as:

$$q = s_1 \wedge s_2 \wedge s_3 \wedge s_4 \wedge s_5 \quad (20)$$

This query might be an imperfect representation of the user needs regarding documents that are related to the index terms $t_1, \ldots t_5$. If all index terms are really required, then (20) is right. However, it may easily be the case that the user is not sure how to express his or her needs that may, in fact, be satisfied by a document related to just 3 or 4 of the index terms $t_1, \ldots t_5$. More generally, he or she may find interesting a document related to *most* of the index terms from among $t_1, \ldots t_5$. If this latter requirement is the real need of the user, then a query in the form (20) might be completely inadequate. Notice, that it is enough that only one s_i is valuated to 0 for a given document to make the whole query to be matching this document to degree 0. In this respect, the fuzzy extension of the fuzzy Boolean model as described earlier does not change anything.

This problem has been addressed by Salton et al. [21] and their approach may be illustrated on the following example (20). They suggested that the relevance of a document should be computed as inversely proportional to the distance between two vectors $[w_1, w_2, w_3, w_4, w_5]$ and $[1, 1, 1, 1, 1]$, where the former represents document d, gathering weights of the index terms $t_1, \ldots, t_5$ in it, while the latter represents our exemplary query (20). The distance between the vectors is to be computed using a p-norm (more often referred to as an l-norm) for a selected value of parameter p. For $p = 1$ we obtain the solution similar to the vector space model, and for $p = \infty$ we obtain a simple fuzzy model with the minimum and maximum operators representing the conjunction and disjunction.

This deficiency of both the classical and fuzzy logical connectives has been also observed in the context of decision making, database querying etc. A number of other *aggregation operators* has been proposed as substitutes. Now, they may be studied in the framework of a rigorous, formally defined fuzzy logical model of information retrieval.

V. Conclusion

We have proposed an extension to the classical Boolean model of information retrieval. This extension refers to a new theoretical advancements regarding fuzzy propositional logic. In the paper we have provided formal logical interpretations for all elements of the information retrieval model. We have used both some results well-known in the literature and some original solutions. The latter include: a new interpretation of a document as a valuation in terms of propositional logic; syntactical representation of weighted queries in the language of fuzzy propositional logic presented in [12], [13], [10] and a logical representation of ideal weights interpretation of importance weights in queries via equivalence operator.

A comprehensive interpretation of all elements of information retrieval model in terms of fuzzy propositional logic makes it possible to address some other problems inherent to a classical logical model (cf., e.g., [22]). This requires a further study. Another area for further research is the question of interpretation of the proposed model depending on the implication operator assumed in the lattice (7).

References

[1] Radecki T., "Fuzzy set theoretical approach to document retrieval," *Information Processing and Management*, vol. 15, no. 5, pp. 247–260, 1979.

[2] Buell D. and Kraft D.H., "Threshold values and Boolean retrieval systems," *Information Processing & Management*, vol. 17, pp. 127–136, 1981.

[3] Kraft D.H. and Buell D.A., "Fuzzy sets and generalized Boolean retrieval systems," *International Journal on Man-Machine Studies*, vol. 19, pp. 45–56, 1983.

[4] Yager R.R., "A note on weighted queries in information retrieval systems," *Journal of the American Society for Information Science*, vol. 38, pp. 23–24, 1987.

[5] Ogawa Y. and Morita T. and Kobayashi K., "A fuzzy document retrieval system using the keyword connection matrix and a learning method," *Fuzzy Sets and Systems*, vol. 39, pp. 163–179, 1991.

[6] Kraft D.H. and Buell D.A., "Fuzzy sets and generalized Boolean retrieval systems," in *Readings in Fuzzy Sets for Intelligent Systems*, Dubois D., Prade H., and Yager R.R., Eds. San Mateo: Morgan Kaufmann Publishers, 1992.

[7] Bordogna G., Carrara P., and Pasi G., "Query term weights as constraints in fuzzy information retrieval," *Information Processing & Management*, vol. 27, pp. 15–26, 1991.

[8] Kraft D.H. and Bordogna G. and Pasi G., "An extended fuzzy linguistic approach to generalize Boolean information retrieval," *Journal of Information Sciences*, vol. 2, no. 3, pp. 119–134, 1994.

[9] Bordogna G., Carrara P., and Pasi G., "Fuzzy approaches to extend Boolean information retrieval," in *Fuzziness in Database Management Systems*, Bosc P. and Kacprzyk J., Eds. Heidelberg: Physica Verlag, 1995, pp. 231–274.

[10] Hajek P., "On the metamathematics of fuzzy logic," in *Discovering the World with Fuzzy Logic*, Novak V. and Perfilieva I., Eds. Heidelberg New York: Physica-Verlag, 2000, pp. 155–174.

[11] Novak V. and Perfilieva I., Eds., *Discovering the World with Fuzzy Logic*. Heidelberg New York: Physica-Verlag, 2000.

[12] Novak V., Perfilieva I., and Močkoř J., *Mathematical Principles of Fuzzy Logic*. Boston: Kluwer, 1999.

[13] Perfilieva I. and Novak V., "Fuzzy logic on the basis of classical logic," in *Selected Topics on Information Technology*, Kacprzyk J., Krawczak M., and Zadrożny S., Eds. EXIT, 2002, pp. 83–130.

[14] van Rijsbergen C. J., "A new theoretical framework for information retrieval," in *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*, Rabitti F., Ed., Pisa, Italy, 1986, pp. 194–200.

[15] van Rijsbergen C.J., "A non-classical logic for information retrieval," *The Computer Journal*, vol. 29, no. 6, pp. 481–485, 1986.

[16] Sebastiani F., "A note on logic and information retrieval," in *MIRO'95 Proceedings of the Final Workshop on Multimedia Information Retrieval*. Glasgow, Scotland: Springer, 1995.

[17] Dubois D. and Prade H., "Using fuzzy sets in flexible querying: why and how?" in *Flexible Query Answering Systems*, Andreasen T., Christiansen H., and Larsen H.L., Eds. Kluwer Academic Publishers, 1997, pp. 45–60.

[18] Fodor J. and Roubens M., *Fuzzy Preference Modelling and Multicriteria Decision Support*, ser. Sereis D: System Theory, Knowledge Engineering and Problem Solving. Kluwer academic Publishers, 1994.

[19] Bordogna G. and Pasi G., "Application of fuzzy sets theory to extend Boolean information retrieval," in *Soft Computing in Information Retrieval*, Crestani F. and Pasi G., Eds. Heidelberg New York: Physica Verlag, 2000, pp. 21–47.

[20] Herrera-Viedma E., "Modeling the retrieval process of an information retrieval system using an ordinal fuzzy linguistic approach," *Journal of the American Society for Information Science and Technology (JASIST)*, vol. 52, no. 6, pp. 460–475, 2001.

[21] Salton G. and Fox E.A. and Wu H., "Extended Boolean information retrieval," *Communications of ACM*, vol. 26, no. 11, pp. 1022–1036, 1983.

[22] Lalmas M., "Logical models in information retrieval: Introduction and overview," *Information Processing & Management*, vol. 34, no. 1, pp. 19–33, 1998.

[23] Kraft D.H. and Bordogna G. and Pasi G., "Fuzzy set techniques in information retrieval," in *Fuzzy Sets in Approximate Reasoning and Information Systems*, ser. The Handbook of Fuzzy Sets Series, Bezdek J.C., Dubois D., and Prade H., Eds. Norwell: Kluwer Academic Publishers, 1999, vol. 3.