The Effective Relevance Link between a Document and a Query

Karam Abdulahhad, Jean-Pierre Chevallet, Catherine Berrut

The Effective Relevance Link between a Document and a Query

Karam Abdulahhad*, Jean-Pierre Chevallet**, and Catherine Berrut*

* UJF-Grenoble 1, ** UPMF-Grenoble 2, LIG laboratory, MRIM group
karam.abdulahhad,jean-pierre.chevallet,catherine.berrut@imag.fr

Abstract. This paper proposes to understand the retrieval process of relevant documents against a query as a two-stage process: at first an identification of the reason why a document is relevant to a query that we called the Effective Relevance Link, and second the valuation of this link, known as the Relevance Status Value (RSV). We present a formal definition of this semantic link between d and q. In addition, we clarify how an existing IR model, like Vector Space model, could be used for realizing and integrating this formal notion to build new effective IR methods. Our proposal is validated against three corpuses and using three types of indexing terms. The experimental results showed that the effective link between d and q is very important and should be more taken into consideration when setting up an Information Retrieval (IR) Model or System. Finally, our work shows that taking into account this effective link in a more explicit and direct way into existing IR models does improve their retrieval performance.

1 Introduction

Information Retrieval Systems (IRSs) are supposed to classify documents in two sets: the set of relevant documents to a query q, and the set of documents that are not relevant. An IRS computes a *machine* relevance that is supposed to be closed to a *human* relevance judgment, i.e. the judgment from the author of the query, also called *user relevance*. Moreover, IRSs usually compute a relevance score: a Relevance Status Value (RSV), against all documents, or against only those that are retrieved[1].

This distinction is very important because, that means there are two different notions in IR: the *relevance* notion and the *valuation* of this relevance computed by the RSV. Unfortunately, if the RSV is computed against all documents of a corpus, the first notion disappears. In this case, only a ranking of documents based on the RSV is computed.

For practical reason, many IRSs compute RSV only against the set of documents that share terms with the query: we interpret this as the minimal constraint often chosen to build the set of retrieved document is a non empty term

[1] The set of retrieved documents is implicitly the set of relevant documents from the machine point of view.

intersection between a retrieved document and a query. We feel that this poor constraint hides a more semantical constraint that a document should fulfill in order to be retrieved by a system: there should be a hidden *semantic link* and a reasoning process that could be followed to *demonstrate* this relevance. We call this link the *Effective Relevance Link* and we denoted it $d \leftrightarrow q$ in this paper. Hence the effective link between a document d and a query q is related to the reason or reasons that make d a candidate document to be retrieved for q. We also think that this Effective Semantic Link can be expressed in some logic. For example, we can state that for having $d \leftrightarrow q$, the system should explicitly has a *logical* reason like: if d is relevant to q, then there should exist a logical deduction chain that starts from d and ends at q, written as $d \rightarrow q$.

Information Retrieval (IR) models include more or less this effective link, each model in its own way. Vector Space Model (VSM) [18] assumes that both d and q are vectors in a specific term space and the link $d \leftrightarrow q$ is simply equivalent to a non empty terms intersection. Probabilistic Models (PM) [14] propose the probabilistic ranking principle for ranking documents by decreasing probability of their relevance to queries. Different estimations of the previous probability mean different variants of PM. Language Models (LM) [13] borrow their notion from the speech recognition community. LMs suppose that each document is a language and then they estimate the ability of document's language to reproduce the query.

Actually, the two notions are mixed in most of classical IR models: the effective link $d \leftrightarrow q$, or in other words, *why* d is a candidate answer for q, and *how much* $d \leftrightarrow q$ is strong, i.e. the relevance score $RSV(d, q)$. Therefore, we think one should study $d \leftrightarrow q$ and $RSV(d, q)$ that characterizes it, as separate notions.

Moreover, most IRS compute first an $RSV(d, q)$ and then deduce $d \leftrightarrow q$, for example by an implicit thresholding of the RSV value of the ordered documents list. We think the correct view should first be the study and computation of $d \leftrightarrow q$, and then computation of the strength of this link.

In this work, we separate the two notions through introducing a formal-logical definition for $d \leftrightarrow q$. In addition, we study the influence of this separation on the performance of some IR methods.

The paper is structured as follow: In section 2, we present a more formal definition of $d \leftrightarrow q$ using the two notions *Exhaustivity* and *Specificity*. We show the importance of $d \leftrightarrow q$ and the attempts of using it in IR literatures, in section 3. Then in section 4, we show a practical view of $d \leftrightarrow q$. In section 5, we describe our concrete framework for integrating $d \leftrightarrow q$ in IR methods. We show our experiments for validating our hypothesis, in section 6. We conclude the paper in section 7.

2 A Tentative Formal-Logical Definition of $d \leftrightarrow q$

We introduced $d \leftrightarrow q$ in a subjective way. In order to build a more formal and general definition, we model it in a logical framework.

Many researchers argued that the retrieval process could be formalized as a logical implication from a document d to a query q noted $d \rightarrow q$. One of the earliest studies in this direction is the one of *Rijsbergen* [21], who introduced the use of logic as a theoretical IR foundation. He proposes to see d and q as a set of logical sentences in a specific logic. Then d is a candidate answer for q *iff* it logically implies q noted $d \rightarrow q$. In other words, if q is deducible based on d. However, IR is an uncertain process [5], because:

- The query q is an imperfect representation of the user needs;
- The document model d is also an imperfect representation of the semantic content of the document:
- Relevance judgment depends on an external factor, which is the user knowledge.

Therefore, another component, beside the logic, should be added to estimate the certainty of an implication and to offer a ranking mechanism. In other words, a measure P should exist and be able to measure the certainty of the logical implication between d and q, noted $P(d \rightarrow q)$. This formulation already split the matching computation into two stages: first establish the truth of the effective link $d \rightarrow q$ and then compute a score P, on this link.

This proposal exhibits a non symmetrical Effective Link, and leads to this question: are the two implications $d \rightarrow q$ and $q \rightarrow d$ the same? *Nie* [12], distinguishes the *Exhaustivity* of a document to a query $d \rightarrow q$, which means that d satisfies all *themes* of q, from the *Specificity* of a document to a query $q \rightarrow d$, which means that d's themes are all related to q.

In other words, *Exhaustivity* means that all themes of q should be referred in d. In this manner, suppose that we have a document d where $d \rightarrow q$ is valid, if we build another document d' by adding more themes to d then $d' \rightarrow q$ should be still valid. However, d is more relevant to q because it is more specific. Therefore, we need the other notion of "*Specificity*" in order to retrieve the most specific or precise document that already covers q [6]. The ideal case is when d contains all and only all the themes of q, that means, we can prove both $d \rightarrow q$ and $q \rightarrow d$.

Formal logic has a syntax but can also have semantics[2]. This semantic translates the formal sentences of that logic into another mathematical world. For example, we get the semantic of a logical sentence in the Propositional Logic by assigning a truth value (T or F) to each proposition in that sentence. Hence, each logical sentence s can have several translations or interpretations I_s depending on the truth values assignations. The subset M_s of I_s that make s true is called the set of *models* of s.

For any two sentences s_1 and s_2, we say that s_1 logically entails s_2, written as $M_{s_1} \models s_2$, or simply $s_1 \models s_2$, if s_2 is true in all models of s_1. In other words, any interpretation that makes s_1 true should also make s_2 true. Obviously, $\models$ is not commutative. In this manner, the Exhaustivity $d \rightarrow q$ could be translated into $d \models q$ whereas the Specificity into $q \models d$.

[2] We would like to warn the reader unfamiliar with logic formalisms that this notion of semantics (called formal semantics) is not related to "human" meaning.

The more terms a document has, the less number of models validating that document exist. For example, with the indexing vocabulary $V = \{t_1, t_2, t_3, t_4, t_5\}$, one can have 2^5 different interpretations over V. If a document d is indexed by the terms $\{t_1, t_2, t_3\}$, then one can associate with d the set of 4 models M_d that make t_1, t_2, t_3 true. Another document d' indexed by $\{t_1, t_2, t_3, t_4\}$ is associated with the set $M_{d'}$ of only 2 models[3]. In this example any model of d' is also a model of d which means $M_{d'} \subseteq M_d$. In other words[4], if $d \models q$ then $d' \models q$.

By taking the uncertainty into account, the two notions Exhaustivity and Specificity could be rewritten as follow:

- **Exhaustivity** $P(d \rightarrow q)$: means to which limit M_d and $M_d \cap M_q$ are close, or in other words, $P(d \rightarrow q)$ could be equivalent to evaluate an other function $P(M_d, M_d \cap M_q)$. The best case is when $M_d \subseteq M_q$, which means $M_d = M_d \cap M_q$.
- **Specificity** $P(q \rightarrow d)$: means to which limit M_q and $M_d \cap M_q$ are close, or in other words, $P(q \rightarrow d)$ could be equivalent to $P(M_q, M_d \cap M_q)$. The best case is when $M_q \subseteq M_d$, which means $M_q = M_d \cap M_q$.

After this detailed description of $d \leftrightarrow q$ and after clarifying the potential interaction between Exhaustivity and Specificity, instead of calculating the relevance score between d and q as a degree of certainty of the logical implication $P(d \rightarrow q)$, now the relevance score is a function of the two implications [12] (1):

$$RSV\left(d, q\right) = F[P\left(d \rightarrow q\right), P\left(q \rightarrow d\right)] \tag{1}$$

3 $d \leftrightarrow q$ in IR Models

Many IR models, one way or another, try to integrate $d \leftrightarrow q$ in the process of computing the Relevance Status Value (RSV) between d and q. *Abdulahhad (et al.)* [1] exploit the semantic relations between document's concepts and query's concepts and use the attached weights of those relations for computing the final matching value between d and q.

Other studies *Rocchio* [16], *Salton (et al.)* [19] and *Buckley (et al.)* [4], need a second round of evaluation for integrating $d \leftrightarrow q$, through query reformulation using several prejudged documents. In fact after using RSV for sorting retrieved documents, they make the hypothesis that only some of them are really relevant, i.e. satisfies the effective relevance link. This technique is known as relevance feedback.

In classical bag-of-terms based IR methods, $d \leftrightarrow q$ is implicitly integrated. For example, in BM25 [15] and Pivoted Normalization Method [20], $d \leftrightarrow q$ appears timidly through the sum over shared terms ($\sum_{t \in d \cap q}$). The same thing

[3] One that makes t_1, t_2, t_3, t_4 true and t_5 false, and the other that make t_1, t_2, t_3, t_4 and t_5 true.

[4] Note that this is not necessarily true for all logical IR models. For example, this does not hold for the classical IR boolean model because documents are associated with only one interpretation.

for Language Models [13], but instead of sum, it is the product ($\prod_{t \in d \cap q}$). It is also true for Information-based methods [2] [7].

Several studies *Wilkinson (et al.)* [22] and *Rose (et al.)* [17] show that users prefer documents sharing more distinct terms with queries. Moreover, Fang (et al.) [10] determine several retrieval constraints for building effective retrieval methods. The second constraint $TFC2$ implies another constraint, which encourages promoting documents with more distinct query terms.

Historically, one of the earliest methods of ranking was the number of shared terms between d and q ($|d \cap q|$). This method is added to the Boolean Model in order to rank retrieved documents. In addition, the ranking formula of VSM could be restricted to $|d \cap q|$ when using binary weights for document and query terms (1 if t occurs in d, 0 otherwise).

From the previous presentation, we can see that in spite of the importance of $d \leftrightarrow q$, represented by $d \cap q$, it is not sufficiently integrated in the classical IR methods. In this study, we try to explicitly integrate the $d \leftrightarrow q$ in the process of estimating the retrieval score between d and q: $RSV(d, q)$, in order to build a more precise and effective retrieval method.

4 $d \leftrightarrow q$ and Weighting

In all IR models, e.g. Language Models [13], Probabilistic Models [14], Vector Space Models (VSM) [18], etc. the weight of an indexing term t is usually estimated depending on three sources of information:

1. The document d is usually used for estimating the descriptive power or the local weight w_t^d of t in d. For example, the term frequency of t in d.
2. The query q: the weight w_t^q is whether manually assigned by users or estimated through the term frequency of t in q.
3. The corpus or document collection D is used for estimating the discriminative power w_t^D of t in D. For example, the Inverse Document Frequency (IDF), or the smoothing component of Language Models [23].

In general, at the time of computing the matching value between a specific document d and a specific query q:

1. The value of w_t^d is independently estimated of q. For certain d and t, w_t^d is constant whatever is q.
2. The value of w_t^q is independently estimated of d. For certain q and t, w_t^q is constant whatever is d.
3. The value of w_t^D is independently estimated of both d and q. For certain D and t, w_t^D is constant whatever is d and q.

We think that this is an insufficient modeling because each weight is independently computed from the effective link $d \leftrightarrow q$. Hence, we propose the matching score computation to take into account the $d \leftrightarrow q$ in an explicit manner, in addition to w_t^d , w_t^q and w_t^D.

We illustrate this problem using one of the classical IR methods, the Pivoted Normalization Method [20] (2).

$$RSV(d,q) = \sum_{t \in d \cap q} \frac{1 + ln\left(1 + ln\left(tf_{t,d}\right)\right)}{(1-s) + s\frac{|d|}{avdl}} \times tf_{t,q} \times ln\frac{N+1}{n_t} \tag{2}$$

where $tf_{t,d}$ is the term frequency of t in d, $tf_{t,q}$ is the term frequency of t in q, s is a constant (normally $s = 0.2$), $|d|$ is the length of d, $avdl$ is the average document length in the corpus, N is the total number of documents in the corpus, and n_t is the number of documents that contain t.

$$\begin{gathered} RSV(d,q) = \textstyle\sum_{t \in d \cap q} w_t^d \times w_t^q \times w_t^D \\ w_t^d = \frac{1+ln(1+ln(tf_{t,d}))}{(1-s)+s\frac{|d|}{avdl}} \qquad w_t^q = tf_{t,q} \qquad w_t^D = ln\frac{N+1}{n_t} \end{gathered} \tag{3}$$

(3) shows that w_t^d is independent of q, w_t^q is independent of d, and w_t^D is independent of both d and q.

As most of IR methods are based on the bag-of-terms paradigm, the most evident indication to $d \leftrightarrow q$ could be the shared terms between d and q: $d \cap q$, because what makes d a candidate answer for q is having shared terms with q: $d \cap q \neq \emptyset$. The shared terms compose the ground where both d and q interact with each other. Without shared terms ($d \cap q = \emptyset$), there is no explicit link between d and q, hence d is not potentially a relevant document. Actually this is not quite correct because of the *term-mismatch* problem [8], where two terms are used for expressing on the same meaning, e.g. flat vs. apartment. However, the term-mismatch problem is out of the scope of this study.

5 Revisiting the VSM with $d \leftrightarrow q$

The Vector Space Model (VSM) is a well known model that can benefit from an explicit integration of the Effective Relevance Link. Before revisiting the VSM model, let's analyze the relationship between the logical description of $d \leftrightarrow q$ and a term set representation.

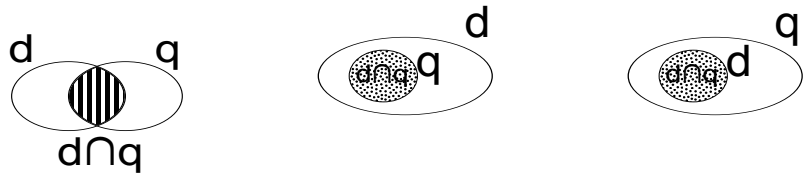
In the previous section, we rewrote the two implications $d \rightarrow q$ and $q \rightarrow d$ using $M_d \cap M_q$. One can associate a set of terms d with a set of models M_d in the following way: M_d are the models where each term of d is true. Hence adding a term to the set d reduces the model set M_d. Moreover, if $q \subseteq d$ then $M_d \subseteq M_q$, and finally $M_d \cap M_q$ is equivalent to $d \cap q$: see Fig. 1.

For example, with the vocabulary set $\{t_1, t_2, t_3, t_4\}$, given the document $d = \{t_1, t_2, t_3\}$. Then using the VSM notation (with 1 for true):

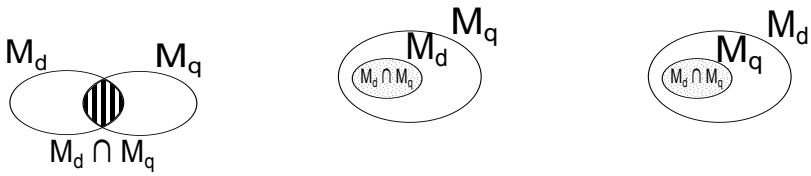
$$M_d = \{\langle 1,1,1,0\rangle, \langle 1,1,1,1\rangle\}$$

If $q = \{t_1, t_2\}$ then:

$$M_q = \{\langle 1,1,0,0\rangle, \langle 1,1,0,1\rangle, \langle 1,1,1,0\rangle, \langle 1,1,1,1\rangle\}$$



(a) Normal case (terms) (b) Exhaustivity (terms) (c) Specificity (terms)



(d) Normal case (models) (e) Exhaustivity (models) (f) Specificity (models)

Fig. 1. The different cases of interaction between d and q

In this example $q \subseteq d$ whereas $M_d \subseteq M_q$. In this case the document cover the query in an exhaustive manner. See Figs. 1(b) and 1(e).

According to *Nie* [12], the RSV value between d and q could be estimated as a function F of the degree of certainty of the two implications (1).

According our definition of Exhaustivity and Specificity, the (1) could be rewritten as follow (4):

$$RSV\left(d,q\right) = F[P\left(M_d, M_d \cap M_q\right), P\left(M_q, M_d \cap M_q\right)] \tag{4}$$

By assuming that documents and queries are sets of terms instead of assuming that they are logical sentences with sets of models, (1) could be written as (5). There is always a possibility to go from a logical sentence to a set of terms and vice-versa [11], through: 1- using the Propositional Logic, 2- assuming that each term is a proposition, and 3- assuming that each document is a logical sentence of conjunctive propositions or it is a set of terms.

$$RSV\left(d,q\right) = F[P\left(q, d \cap q\right), P\left(d, q \cap d\right)] \tag{5}$$

Actually, we need a concrete framework for computing the $RSV(d,q)$. Therefore, we need to realize the following abstract elements:

– The document d.
– The query q.
– The shared terms $d \cap q$.
– The function F.
– The uncertainty measure P.

In IR field, there are many frameworks for doing that, e.g. Vector Space Framework [18], Probabilistic Framework [14], Language Models [13], etc. In this study, we choose the Vector Space Framework. Therefore, the previous abstract elements become:

– The document $\overrightarrow{d}$ is a vector in term space T. For each term $t \in T$, there is a correspondent component in $\overrightarrow{d}$: w_t^d, where $w_t^d > 0$ is the weight of t in d if t occurs in d or 0 otherwise.
– The query $\overrightarrow{q}$ is a vector in term space T. For each term $t \in T$, there is a correspondent component in $\overrightarrow{q}$: w_t^q, where $w_t^q = 1$ if t occurs in q or 0 otherwise.
– The shared terms $\overrightarrow{d \cap q}$ is a vector in term space T. For each term $t \in T$, there is a correspondent component in $\overrightarrow{d \cap q}$: $w_t^{d \cap q}$, where $w_t^{d \cap q} = 1$ if t occurs in both d and q or 0 otherwise.
– The function F: there are many choices for F, e.g. sum, weighted sum for favoring Exhaustivity over Specificity or vice-versa, product, etc. In this study, we suppose that both Exhaustivity and Specificity are equally important and we choose the product ($\times$).
– The uncertainty measure P: in Vector Space Framework there are many choices for computing the distance between two vectors [9]. Here, we choose the inner-product measure.

Finally, $P(q, d \cap q)$ is the distance between $\overrightarrow{q}$ and $\overrightarrow{d \cap q}$, same for $P(d, d \cap q)$. The (5) could be rewritten as follow (6):

$$RSV\,(d,q) = \left(\overrightarrow{q} \cdot \overrightarrow{d \cap q}\right) \times \left(\overrightarrow{d} \cdot \overrightarrow{d \cap q}\right) \tag{6}$$

where $(\cdot)$ is the inner-product (dot-product). Then the retrieval formula becomes (7):

$$RSV\,(d,q) = \left[\sum_{t \in T} w_t^q \times w_t^{d \cap q}\right] \times \left[\sum_{t \in T} w_t^d \times w_t^{d \cap q}\right] = |d \cap q| \times \sum_{t \in d \cap q} w_t^d \tag{7}$$

where $|d \cap q|$ is the number of shared terms between d and q.

The only remaining component that should be clarified is the weight of a term t in a document d or w_t^d. Several weighting formulas exist e.g. Pivoted Normalization [20], BM25 [15], DFR [2], TF-IDF, etc. Here we will use a version of the *TF-IDF* formula. Our final retrieval formula becomes (8):

$$RSV\,(d,q) = |d \cap q| \times \left[\sum_{t \in d \cap q} \frac{tf_{t,d}}{tf_{t,d} + |d|} \times \frac{N}{n_t}\right] \tag{8}$$

6 Experiments

To validate our hypothesis about the utility of integrating the $d \leftrightarrow q$ into IR models, we apply (8) on corpuses and compare the performance against the performance of some classical IR methods. We use for the comparison the Mean Average Precision (MAP) metric.

6.1 Experiments Setup

We use in our experiments three different corpuses and three types of indexing terms.

The types of indexing terms: each type of indexing terms represents a different facet of documents and queries.

- 5Grams ($5G$) / 4Grams ($4G$): we used five-characters-wide / four-characters-wide window for extracting 5grams / 4grams with shifting the window one character each time.
- Words (W): we eliminated the stop words and stemmed the remaining words using Porter algorithm to get finally the list of words that indexes documents and queries.
- Concepts (C): we mapped the text into UMLS's concepts using MetaMap, where UMLS[5] is a multi-source knowledge base in the medical domain. Whereas, MetaMap[6] [3] is a tool for mapping text into UMLS concepts.

Corpuses: we validate our hypothesis against three corpuses. One from ImageCLEF2010[7] and two from ImageCLEF2011 (Table 1):

Table 1. Corpuses statistics. *avdl* and *avql* are the average length of documents and queries.

Corpus	#d	#q	Type	*avdl*	*avql*
image2010	77495	16	5G	627.23	29.88
			W	62.12	3.81
			C	157.27	12.0
image2011	230088	30	5G	468.86	32.1
			W	44.83	4.0
			C	101.92	12.73
case2011	55634	10	4G	30380.17	192.4
			W	2594.5	19.7
			C	5752.38	57.5

[5] Unified Medical Language System. http://www.ncbi.nlm.nih.gov/bookshelf/br.fcgi?book=nlmumls
[6] http://metamap.nlm.nih.gov/
[7] http://www.imageclef.org/

- *image2010*: contains short medical documents and queries.
- *image2011*: also contains short medical documents and queries. However, it is larger than image2010.
- *case2011*: contains long medical case description documents and long queries.

IR models: from one side, the performance of TF-IDF$_{d\cap q}$ (8) is compared to the performance of the same formula but without $|d \cap q|$ component (TF-IDF). We did that for showing the positive effect of integrating $d \leftrightarrow q$ into weighting formulas. From another side, we compare the performance of TF-IDF$_{d\cap q}$ to the performance of Pivoted Normalization Method PIV (2), BM25 method (9), and Dirichlet language model DIR (10), where $p\,(t, D)$ is the probability of t given by the collection language model D. Through this comparison, we show the validity of our hypothesis.

s, k_1, b, k_3, and μ are all constants. They usually have the following values: $s = 0.2$ [20]. $k_1 = 1.2$, $b = 0.75$, and $k_3 = 1000$ [10]. $\mu = 2000$ [23].

$$RSV(d,q) = \sum_{t\in d\cap q} ln\frac{N-n_t+0.5}{n_t+0.5} \times \frac{(k_1+1)\times tf_{t,d}}{k_1\times\left((1-b)+b\times\frac{|d|}{avdl}\right)+tf_{t,d}} \times \frac{(k_3+1)\times tf_{t,q}}{k_3\times tf_{t,q}} \tag{9}$$

$$RSV(d,q) = \sum_{t\in d\cap q} tf_{t,q} \times ln\left(1+\frac{tf_{t,d}}{\mu\times p\,(t,D)}\right) + |q| \times ln\frac{\mu}{|d|+\mu} \tag{10}$$

6.2 Results and Discussion

Table 2. The experimental results of applying (8) to the three corpuses and using the three types of terms.

		image2010		image2011		case2011	
Type	Formula	MAP	gain	MAP	gain	MAP	gain
$5G$ / $4G$	TF-IDF$_{d\cap q}$	0.3165	+16%	0.1474	+26%	0.0755	+26%
	TF-IDF	0.2739		0.1169		0.0599	
W	TF-IDF$_{d\cap q}$	0.3332	+14%	0.2069	+51%	0.1044	+33%
	TF-IDF	0.2916		0.1368		0.0786	
C	TF-IDF$_{d\cap q}$	0.3248	+13%	0.1672	+13%	0.1605	+19%
	TF-IDF	0.2883		0.1484		0.1347	

Table (2) shows the experimental results of (8), applying on the three corpuses and using the three types of terms. Table (2) shows that using $|d\cap q|$, or in other words, explicit $d \leftrightarrow q$ integration into IR methods, improves considerably the average precision. This conclusion is valid for all corpuses and all types of terms. That means, our hypothesis is valid for short and long documents and queries, in addition, it is also valid for different facets of documents and queries.

Table 3. The experimental results of applying (8) and some classical IR methods (2, 9, and 10) to the three corpuses and using the three types of terms.

Type	Formula	image2010	image2011	case2011
$5G$ / $4G$	TF-IDF$_{d\cap q}$	**0.3165**	**0.1474**	0.0755
	PIV	0.2872	0.1069	0.0759
	BM25	0.2733	0.1302	0.0062
	DIR	0.2947	0.1241	**0.0775**
W	TF-IDF$_{d\cap q}$	**0.3332**	**0.2069**	0.1044
	PIV	0.2992	0.1546	0.1023
	BM25	0.2745	0.1995	0.0964
	DIR	0.2960	0.1534	**0.1295**
C	TF-IDF$_{d\cap q}$	**0.3248**	**0.1672**	**0.1605**
	PIV	0.2530	0.1096	0.1037
	BM25	0.2123	0.1552	0.0956
	DIR	0.2455	0.1228	0.1036

Table (3) show the experimental results of (8) and some classical IR methods (2, 9, and 10). They show that for all types of terms and for all corpuses, (8) performs better than the other formulas, except when using words with long documents and queries. In other words, even a simple non-parametric formula (8) performs better than classical IR methods, through simple integration of $|d \cap q|$ into the formula, where $|d \cap q|$ is an indication to $d \leftrightarrow q$.

In conclusion, the effective link between d and q ($d \leftrightarrow q$) is a very important component, and it should be correctly exploited for improving the performance of IR methods.

7 Conclusion

We study in this paper the explicit integration of the effective link $d \leftrightarrow q$ into an IR matching model. We have presented a formal definition of $d \leftrightarrow q$ based on logical framework through two notions: Exhaustivity and Specificity. Those notions describe an interesting relevance link between d and q. According to Exhaustivity and Specificity, the best answer for a query q is the most specific (smallest) document that fully contains q.

We revisit the Vector Space Model, and test the effect of integrating $d \leftrightarrow q$ into the matching formula. Experimental results on three test corpuses show that our hypothesis about the importance of integrating $d \leftrightarrow q$ into IR models is valid. We also validated our hypothesis against three types of indexing terms and we get similar positive results.

The next steps of this work concern the revisiting of other IR models like the probabilistic and language models, and some experimentation on other test collections, not specifically in the medical domain.

References

1. Karam Abdulahhad, Jean-Pierre Chevallet, and Catherine Berrut. Solving concept mismatch through bayesian framework by extending umls meta-thesaurus. In *la huitième édition de la COnfrence en Recherche d'Information et Applications (CORIA 2011), Avignon, France*, March 16–18 2011.
2. Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357–389, October 2002.
3. Alan R. Aronson. Metamap: Mapping text to the UMLS metathesaurus, 2006.
4. Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic Query Expansion Using SMART: TREC 3. In *TREC*, 1994.
5. Y. Chiaramella and J. P. Chevallet. About retrieval models and logic. *Comput. J.*, 35:233–242, June 1992.
6. Yves Chiaramella, Philippe Mulhem, and Franck Fourel. A model for multimedia information retrieval. Technical report, 1996.
7. Stéphane Clinchant and Eric Gaussier. Information-based models for ad hoc ir. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 234–241, New York, NY, USA, 2010. ACM.
8. Fabio Crestani. Exploiting the similarity of non-matching terms at retrievaltime. *Inf. Retr.*, 2(1):27–47, February 2000.
9. S. Dominich. *Mathematical Foundations of Information Retrieval (Mathematical Modelling: Theory and Applications)*. Springer, 1 edition, March 2001.
10. Hui Fang, Tao Tao, and ChengXiang Zhai. A formal study of information retrieval heuristics. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 49–56, New York, NY, USA, 2004. ACM.
11. David E. Losada and Alvaro Barreiro. A logical model for information retrieval based on propositional logic and belief revision. *The Computer Journal*, 44:410–424, 2001.
12. J. Nie. An outline of a general model for information retrieval systems. In *Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '88, pages 495–506, New York, NY, USA, 1988. ACM.
13. Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 275–281, New York, NY, USA, 1998. ACM.
14. S. E. Robertson. Readings in information retrieval. chapter The probability ranking principle in IR, pages 281–286. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.
15. S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '94, pages 232–241, New York, NY, USA, 1994. Springer-Verlag New York, Inc.
16. J. Rocchio. *Relevance Feedback in Information Retrieval*, pages 313–323. 1971.
17. Daniel E. Rose and Curt Stevens. V-twin: A lightweight engine for interactive use. In *TREC*, 1996.

18. G Salton, A Wong, and C S Yang. A vector space model for automatic indexing. *Communications of the ACM*, (18):613–620, 1975.
19. Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.
20. Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '96, pages 21–29, New York, NY, USA, 1996. ACM.
21. C. J. van Rijsbergen. A non-classical logic for information retrieval. *Comput. J.*, 29(6):481–485, 1986.
22. Ross Wilkinson, Justin Zobel, and Ron Sacks-Davis. Similarity measures for short queries. In *TREC*, 1995.
23. Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 334–342, New York, NY, USA, 2001. ACM.