MRIM at ImageCLEF2012. From Words to Concepts: A New Counting Approach

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. MRIM research group has participated in two tasks (ad-hoc image-based retrieval and case-based retrieval) of the ImageCLEF2012 Medical Retrieval track. In our contribution, we study the *frequency shift* problem that happens when using concepts instead of words as indexing terms. The main goal of our experiments is to check the validity of our new counting strategy of concepts (Relative Count), which is proposed as a solution to the *frequency shift* problem. In order to validate our new counting strategy, we compare the retrieval performance (represented by MAP) of some classical IR models using the classical counting strategy (count each concept as 1) with their performance using the new strategy. The results are promising, and using the new counting strategy shows a considerable gain in performance. We use in our experiments two supplementary resources: MetaMap as a text-to-concepts mapping tool, and UMLS as an external resource containing concepts.

1 Introduction

We present in this paper the contribution of the MRIM[1] research group in the ImageCLEF2012[2] Medical Image Retrieval task[3].

The main goal of our experiments and contribution is to study the side-effects of moving from the word-space to the concept-space on the classical Information Retrieval (IR) models. In other words, to study the side-effects of using concepts instead of words as indexing terms on the classical IR models. Concepts are entry IDs in a specific external resource, and each concept is associated to a set of strings that describe it.

However, in order to build a concept-based IR system, another component, for mapping documents and queries text into concepts, is needed. These mapping tools, e.g. MetaMap[4] [2], are imperfect, and they could map one piece of text

[1] Multimedia Information Modeling and Retrieval is a research group in LIG (Laboratoire d'Informatique de Grenoble) laboratory.
http://mrim.imag.fr/
http://www.liglab.fr/

[2] http://www.imageclef.org/2012

[3] http://www.imageclef.org/2012/medical

[4] http://metamap.nlm.nih.gov/

into no or several candidate concepts. For example, MetaMap maps a text like "*x-ray*" into six different UMLS[5] concepts.

Therefore and as classical IR models directly or indirectly depend on the shared terms $d \cap q$ between a document d and a query q in order to compute the Relevance Status Value $RSV(d,q)$ [4], we have what we called a *frequency shift* problem, because the number of shared terms $|d \cap q|$ between d and q changes in a non-homogeneous way when moving from the word-space to the concept-space. For example, using a mapping tool like MetaMap and an external resource like UMLS, if d and q share one word "*x-ray*" in the word-space, then they will share six different concepts in the concept-space. Whereas, if they share a noun-phrase of two words "*lung x-ray*" in the word-space, then they will share only one concept "*C0581647*" in the concept-space.

One solution to this problem is the supplementary disambiguation step that is actually achieved alongside the mapping process [3]. However, in this study we follow another strategy that proposes another concept counting mechanism. We do not count a concept as 1, instead of that, we give to each concept a relative count (in $\mathbb{R}^{*+}$) respecting the following two hypothesis:

- concepts that correspond to a longer text should receive larger count.
- the count of a concept should be inversely proportional to the ambiguity level of its corresponding text. The ambiguity level of a piece of text is determined by the number of concepts that is mapped into. The text that is mapped into a larger number of concepts is more ambiguous.

The goal is to finally satisfy the following condition for a piece of text TXT:

$$|TXT| = \sum_{c \in map(TXT)} count_c \tag{1}$$

where, $|TXT|$ is the number of words in TXT, $map(TXT)$ is the set of all candidate concepts of TXT, and $count_c \in \mathbb{R}^{*+}$ is the new relative count of the concept c and we will explain in the following sections the algorithm of computing it.

This year, ImageCLEF2012 contains four main tracks: 1) Medical Image Classification and Retrieval, 2) Photo Annotation and Retrieval, 3) Plant Identification, and 4) Robot Vision. Medical Image Classification and Retrieval track contains three tasks: 1) modality classification, 2) ad-hoc image-based retrieval which is an image retrieval task using textual, image or mixed queries, and 3) case-based retrieval: in this task the documents are journal articles extracted from PubMed[6] and the queries are case descriptions. We participated in the last two tasks: ad-hoc image-based retrieval and case-based retrieval. Table 1 shows some statistics on the data collections of the two tasks that we participated in. We only use the textual data.

This paper is structured as follows: Section 2 describes in details the problem that occurs when moving from the word-space to the concept-space, and it also

[5] Unified Medical Language System (http://www.nlm.nih.gov/research/umls/)
[6] http://www.ncbi.nlm.nih.gov/pubmed/

Table 1. Corpora statistics. *avdl* and *avql* are the average length of documents and queries. '*C*' for concepts and '*W*' for words. 'image2012' for ad-hoc image-based retrieval data collection and 'case2012' for case-based retrieval data collection.

Corpus	#d	#q	Used fields	Type	*avdl*	*avql*
image2012	306530	22	-	W	47.16	3.55
				C	104.26	9.41
case2012	74654	26	title+abstract	W	160.51	24.35
				C	376.14	63.73
			title+abstract+fulltext	W	2731.24.35	24.35
				C	-	-

presents our proposed solution. Section 3 presents all technical details of applying our proposed solution to ImageCLEF2012 test collections. It also shows our formal runs and the obtained results. We conclude in section 4.

2 Computing Relative Concept Count

Our algorithm depends on the output of MetaMap as a mapping tool and on the UMLS as an external resource containing the concepts. However, the algorithm could be easily generalized because most mapping tools [2][5] have the same general text-to-concept mapping mechanism [3].

For a textual document d or a query q, mapping tools (e.g. MetaMap) extract noun-phrases from the text and try to map them into one or more candidate concepts of a specific external resource (e.g. UMLS).

However, for a noun-phrase np, it is sometimes difficult to find concepts corresponding to the whole noun-phrase. Moreover, even if there are concepts corresponding to the whole noun-phrase np, it is useful to return some concepts corresponding to parts of np, because restricting our attention to the concepts that only correspond to the whole phrase could lead to miss some related concepts, or in other words, it could lead to lose in recall. Therefore, most mapping tools do not only depend on the exact match to find candidate concepts, but they also generate some variants[7] of the original noun-phrase [2], and then finding candidate concepts of all variants instead of only the original noun-phrase. For example, Table 2 shows the variants of the noun-phrase "*lobar pneumonia x-ray*" that are generated by MetaMap, their related candidate UMLS concepts, and the corresponding part of the original noun-phrase.

In this study, we regroup all variants that correspond to the same part of the original noun-phrase into only one variant. Therefore, Table 2 become Table 3.

[7] spelling variants, abbreviations, acronyms, synonyms, inflectional and derivational variants, or meaningful combinations of these.

Table 2. Variants of *"lobar pneumonia x-ray"* generated by MetaMap, their related candidate UMLS concepts, and the corresponding part of the original noun-phrase

Variants	Candidate concepts	Corresponding part
"lobar pneumonia x-ray"	-	*"lobar pneumonia x-ray"*
"lobar pneumonia"	*C0032300, C0155862*	*"lobar pneumonia"*
"lung x-ray"	*C0581647*	*"pneumonia x-ray"*
"lung"	*C0024109, C 1278908*	*"pneumonia"*
"pneumonia"	*C0032285*	*"pneumonia"*
"pulmonary"	*C2707265, C2709248*	*"pneumonia"*
"lobar"	*C1522010*	*"lobar"*
"lobe"	*C1428707*	*"lobar"*
"lobus"	*C0796494*	*"lobar"*
"x-ray"	*C0034571, C0043299, C0043309 C1306645, C1714805, C1962945*	*"x-ray"*

Table 3. Variants of *"lobar pneumonia x-ray"* after regrouping according to the corresponding parts of the original noun-phrase

Variants	Candidate concepts
"lobar pneumonia x-ray"	-
"lobar pneumonia"	*C0032300, C0155862*
"pneumonia x-ray"	*C0581647*
"pneumonia"	*C0024109, C1278908, C0032285 C2707265, C2709248*
"lobar"	*C1522010, C1428707, C0796494*
"x-ray"	*C0034571, C0043299, C0043309 C1306645, C1714805, C1962945*

2.1 Definitions

Our algorithm locally works at the level of noun-phrases not at the level of documents.

Each noun-phrase np is a sequence of words or a set of 2-tuples, where each tuple (w, i) contains a word $w \in W$ and the position $i \in \mathbb{N}^*$ of w in np. Any variant v of np is also supposed to be a noun-phrase. By this way, it is possible to attach to each noun-phrase np, a set V_{np}:

$$V_{np} = \{np\} \cup \{v_1, \ldots, v_j\}$$

where, $\{v_1, \ldots, v_j\}$ are the variants of np that are generated by the mapping tool.

We define the function wrd that returns the set of word-position tuples of a specific variant.

$$wrd : V_{np} \rightarrow 2^{W \times \mathbb{N}^*}$$

For example, suppose the variant $v =$"*pneumonia x-ray*" of the phrase "*lobar pneumonia x-ray*", then:

$$wrd(v) = \{(pneumonia, 2), (x - ray, 3)\}$$

We also define for any variant $v \in V_{np}$, $|v| = |wrd(v)|$ the number of words in v.

We define the function *map* that returns the candidate concepts of a variant $v \in V_{np}$.

$$map : V_{np} \to 2^C$$

where C is a set of concepts. For example, suppose the variant $v =$"*pneumonia x-ray*" of the phrase "*lobar pneumonia x-ray*", then:

$$map(v) = \{C0581647\}$$

We will remove from V_{np} all members v that do not have any candidate concepts $map(v) = \phi$. For example, in the case of "*lobar pneumonia x-ray*" noun-phrase, we will remove the noun-phrase itself because it is not mapped into any concept. Finally, V_{np} becomes:
V_{np}={"*lobar pneumonia*", "*pneumonia x-ray*", "*pneumonia*", "*lobar*", "*x-ray*"}

It is possible to define a partial order $<$ relation on the set V_{np} as follow:

$$\forall v_1, v_2 \in V_{np}, \quad v_1 < v_2 \quad \text{iff} \quad wrd(v_1) \subset wrd(v_2)$$

Therefore, it is possible to define a hierarchy HR_{np} on $V_{np} \cup \{R\}$, where:

- $v \in V_{np} \cup \{R\}$ are the nodes.
- R is an abstract root satisfying: $\forall v \in V_{np}, v < R$. Moreover, $|R| = 0$.
- The direct children $ch(v)$ of any node $v \in V_{np} \cup \{R\}$ is defined as follow: $\forall v_1, v_2 \in V_{np} \cup \{R\}$,

$$v_1 \in ch(v_2) \quad \text{iff} \quad v_1 < v_2 \quad \text{and} \quad \nexists v_3 \in V_{np}, v_1 < v_3 < v_2$$

- The direct parents $pr(v)$ of any node $v \in V_{np} \cup \{R\}$ is defined as follow:

$$\forall v_1, v_2 \in V_{np} \cup \{R\}, v_1 \in pr(v_2) \quad \text{iff} \quad v_2 \in ch(v_1)$$

For example, Fig. 1 shows the hierarchy of the noun-phrase "*lobar pneumonia x-ray*".

2.2 The Algorithm

The main goal of the algorithm is to compute the relative count of candidate concepts, as follow: for a noun-phrase *np*, distributing the number of words $|np|$ of this noun-phrase on the candidate concepts of *np* and its variants. The algorithm respects two hypothesis:

- the relative count of a concept is directly proportional to the number of words in the corresponding variant.

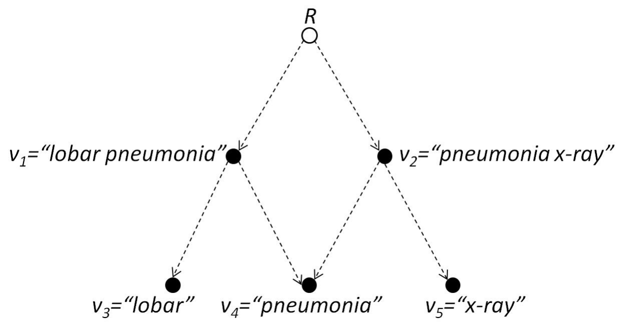

Fig. 1. The hierarchy of the noun-phrase "*lobar pneumonia x-ray*".

– the relative count of a concept is inversely proportional to the ambiguity level of the corresponding variant. The variant that is mapped into a larger number of concepts is more ambiguous.

The input is the set V_{np} that contains all variants of np, and the number of words $|np|$ in np. The output will be a set CC_{np} of 2-tuples, and each tuple contains one of the candidate concepts and its associated relative count. The detailed algorithm is shown in Algorithm 1.

2.3 Example

Assume np is the noun-phrase "*lobar pneumonia x-ray*". The list of variant V_{np} is: $V_{np} = \{v_1, v_2, v_3, v_4, v_5\}$ (see Fig. 1). Figure 1 also shows the hierarchy HR_{np} of the set $V_{np} \cup \{R\}$. Initially, we have $\alpha_R = |np| = 3$ and $\forall v \in V_{np}, \alpha_v = 0$. We scan the nodes in breadth-first order, that means, nodes will be scanned in the following order $< R, v_1, v_2, v_3, v_4, v_5 >$.

By following the Algorithm 1, Table 4 shows the output set CC_{np}. By consulting Table 4, we can see: $\sum_{(c,count_c) \in CC_{np}} count_c = |np| = 3$.

3 Experiments

The main goal of our experiments is to show the validity of our new way of counting. We check this validity through comparing the retrieval performance of some classical IR models using classical concept counting method, to their performance using our new way of counting.

We use the Mean Average Precision (MAP) metric as an indicator to the retrieval performance of IR models.

3.1 Retrieval Models

In order to check the validity of our new way of counting, we use several classical IR models:

Algorithm 1: RelativeCount

input : V_{np}, $|np|$
output: $CC_{np} \subseteq C \times \mathbb{R}^{*+}$

1 $CC_{np} = \{\}$;
2 construct the hierarchy HR_{np} of the set $V_{np} \cup \{R\}$;
3 attach to each node n' in HR_{np} a value $\alpha_{n'}$, where $\alpha_{n'}$ is the total amount that is received from $pr(n')$ and should be distributed on the candidate concepts of n' and its children $ch(n')$, starting by $\alpha_{n'} = 0$;
4 n is the current node, starting by $n = R$;
5 set $\alpha_R = |np|$;
6 scan HR_{np} in a breadth-first way, starting from the current node $n = R$;
7 **begin**
8   **for** *each child* $n'_i \in ch(n)$ **do**
9   compute the amount $\alpha_{n'_i}$ that should be transferred from n to n'_i:
        $\alpha_{n'_i} = \alpha_{n'_i} + \frac{\alpha_n \times |n'_i|}{|n| + \sum_{n'_j \in ch(n)} |n'_j|}$;
10   **end**
11   **if** $n \neq R$ **then**
12   compute the amount a_n that should be distributed on the candidate concepts $map(n)$ of the current node n: $a_n = \frac{\alpha_n \times |n|}{|n| + \sum_{n'_j \in ch(n)} |n'_j|}$;
13   **for** *each candidate concept* $c_i \in map(n)$ **do**
14   compute the relative count: $count^{np}_{c_i} = \frac{a_n}{|map(n)|}$;
15             $CC_{np} = CC_{np} \cup \{(c_i, count^{np}_{c_i})\}$;
16   **end**
17   **end**
18   change n to the next node according to the breadth-first scan order;
19   go to line 7;
20 **end**

- from probabilistic framework: we choose BM25 (2) [7].
- from language models framework: we choose Dirichlet model DIR (3) [9], and Jelinek-Mercer model JM (4) [9].
- from vector space framework: we choose Pivoted Normalization Method PIV (5) [8], and a version of TFIDF model (6) [1].

where, $tf_{t,d}$ is the term frequency of the indexing term t in the document d, $tf_{t,q}$ is the term frequency of the indexing term t in the query q, $|d|$ is the document length, $|q|$ is the query length, $avdl$ is the average document length, N is the total number of documents in the corpus D, n_t is the number of documents that contain the indexing term t, $d \cap q$ are the shared indexing terms between d and q, and $p(t, D)$ is the probability of t given the corpus language model D.

s, k_1, b, k_3, λ, and μ are all parameters. They usually have the following values: $s = 0.2$ [8]. $k_1 = 1.2$, $b = 0.75$, and $k_3 = 1000$ [6]. $\lambda = 0.1$ for short queries or $\lambda = 0.7$ for long queries and $\mu = 2000$ [9].

Table 4. The output set CC_{np} of the noun-phrase "*lobar pneumonia x-ray*"

V_{np}	CC_{np}
v_1="*lobar pneumonia*"	*(C0032300,0.375), (C0155862,0.375)*
v_2="*pneumonia x-ray*"	*(C0581647,0.75)*
v_3="*lobar*"	*(C1522010,0.125), (C1428707,0.125), (C0796494,0.125)*
v_4="*pneumonia*"	*(C0024109,0.15), (C1278908,0.15), (C0032285,0.15)* *(C2707265,0.15), (C2709248,0.15)*
v_5="*x-ray*"	*(C0034571,0.0625), (C0043299,0.0625), (C0043309,0.0625)* *(C1306645,0.0625), (C1714805,0.0625), (C1962945,0.0625)*

$$RSV(d,q) = \sum_{t \in d \cap q} ln \frac{N - n_t + 0.5}{n_t + 0.5} \times \frac{(k_1 + 1) \times tf_{t,d}}{k_1 \times \left((1-b) + b \times \frac{|d|}{avdl}\right) + tf_{t,d}} \times \frac{(k_3+1) \times tf_{t,q}}{k_3 \times tf_{t,q}} \tag{2}$$

$$RSV(d,q) = |q| \times ln \frac{\mu}{|d| + \mu} + \sum_{t \in d \cap q} tf_{t,q} \times ln \left(1 + \frac{tf_{t,d}}{\mu \times p(t,D)}\right) \tag{3}$$

$$RSV(d,q) = |q| \times ln(\lambda) + \sum_{t \in d \cap q} tf_{t,q} \times ln \left(1 + \frac{1-\lambda}{\lambda} \times \frac{tf_{t,d}}{|d| \times p(t,D)}\right) \tag{4}$$

$$RSV(d,q) = \sum_{t \in d \cap q} \frac{1 + ln\left(1 + ln\left(tf_{t,d}\right)\right)}{(1-s) + s\frac{|d|}{avdl}} \times tf_{t,q} \times ln \frac{N+1}{n_t} \tag{5}$$

$$RSV(d,q) = |d \cap q| \times \sum_{t \in d \cap q} tf_{t,q} \times \frac{tf_{t,d}}{tf_{t,d} + \frac{|d|}{avdl}} \times \frac{N}{n_t} \tag{6}$$

There are two different views for documents and queries. The first one is the classical view, where both documents and queries are bags of indexing terms (e.g. words or concepts). The classical way of counting (count each concept or word as 1) is compatible with this view, and the previous definitions of IR model components correspond to this classical view. We will present the second view, which is convenient to the relative count of concepts. In this view, some components of IR models should be redefined.

Most mapping tools extract noun-phrases from the text of documents and queries. Therefore, any document d or query q is a sequence of noun-phrases:

$$d = < np_1, \ldots, np_{n_d} > \qquad q = < np_1, \ldots, np_{n_q} >$$

We redefine the set CC_{np} on the level of documents and queries instead of the noun-phrase level:

$$CC_d = \bigcup_{np_i \in d} CC_{np_i} \qquad CC_q = \bigcup_{np_i \in q} CC_{np_i}$$

$$CC_{d\cap q} = \{(c,r)|(c,r) \in CC_q, \exists(c_i,r_i) \in CC_d, c = c_i\}$$

According to our way of counting (Relative Count), the components of the IR models become:

$$tf_{c,d} = \sum_{(c,r_i)\in CC_d} r_i \qquad tf_{c,q} = \sum_{(c,r_i)\in CC_q} r_i \qquad tf_{c,D} = \sum_{d_i \in D} tf_{c,d_i}$$

$$|d| = \sum_{(c_i,r_i)\in CC_d} r_i \qquad |q| = \sum_{(c_i,r_i)\in CC_q} r_i \qquad |D| = \sum_{d_i \in D} |d_i|$$

$$|d \cap q| = \sum_{(c_i,r_i)\in CC_{d\cap q}} r_i$$

3.2 Data

We only use the textual part of the data collections of ad-hoc image-based retrieval and case-based retrieval tasks of the ImageCLEF 2012 Medical Image Classification and Retrieval track. Table 1 shows some statistics about these two data collections.

We use two types of indexing terms:

- words (W): we eliminate the stop words and stem the remaining words using Porter algorithm to finally get the list of words that indexes documents and queries.
- concepts (C): we use MetaMap for mapping the documents and queries text content into UMLS concepts.

3.3 Formal Runs

The initial of our runs in the formal campaign is "LIG_MRIM_xxx". However, Tables 5 and 6 show the name and the description of our runs. The best MAP in the ad-hoc image-based retrieval task (only text) is *0.2182*, and the best MAP in the case-based retrieval task (text only) is *0.1690*.

3.4 Results and Discussion

In this section we show the validity of our new method of concept counting (RelativeCount), through comparing the retrieval performance of IR models with or without using the relative count. The only type of indexing terms that is used in this section is concepts. We count concepts in two ways: the classical count (count each concept as 1) and the relative count (Algorithm 1).

Table 7 shows the results of applying IR models to the image2012 collection (see Teble 1).

Table 8 shows the results of applying IR models to the case2012 collection (see Teble 1). We only map the title and the abstract parts of documents into concepts

Table 5. Our formal runs in the ad-hoc image-based retrieval task. (TFIDF*) means the TFIDF model after removing $|d \cap q|$. (TFIDF**) means the TFIDF model after removing $|d \cap q|$ and *avdl*

run name	IR model	term type	count type	MAP
IB_TFIDF_W_avdl_DintQ	TFIDF	W	-	0.1586
IB_FUSION_TFIDF_W_TB_C_avdl_DintQ	TFIDF	W + C	relative	0.1432
IB_FUSION_JM01_W_TB_C	JM($\lambda = 0.1$)	W + C	relative	0.1425
IB_TB_PIVv2_C	PIV	C	relative	0.1383
IB_TFIDF_C_avdl_DintQ	TFIDF	C	classic	0.1345
IB_TB_JM01_C	JM($\lambda = 0.1$)	C	relative	0.1342
IB_TB_BM25_C	BM25	C	relative	0.1165
IB_TB_TFIDF_C_avdl	TFIDF*	C	relative	0.1081
IB_TB_DIR_C	DIR	C	relative	0.0993
IB_TB_TFIDF_C	TFIDF**	C	relative	0.0900

Table 6. Our formal runs in the case-based retrieval task. (TFIDF*) means the TFIDF model after removing *avdl*. (TFIDF**) means the TFIDF model after removing $|d \cap q|$. (TFIDF***) means the TFIDF model after removing $|d \cap q|$ and *avdl*. The concepts are extracted from the title and abstract of each document

run name	IR model	term type	count type	MAP
CB_FUSION_DIR_W_TA_TB_C	DIR	W + C	relative	0.1508
CB_FUSION_JM07_W_TA_TB_C	JM($\lambda = 0.7$)	W + C	relative	0.1384
CB_TFIDF_W_DintQ	TFIDF*	W	-	0.1036
CB_TA_TB_JM07_C	JM($\lambda = 0.7$)	C	relative	0.0908
CB_TA_TB_BM25_C	BM25	C	relative	0.0895
CB_TA_TB_DIR_C	DIR	C	relative	0.0893
CB_TA_TB_PIVv2_C	PIV	C	relative	0.0865
CB_TA_TFIDF_C_DintQ	TFIDF*	C	classic	0.0789
CB_TA_TB_TFIDF_C_avdl	TFIDF**	C	relative	0.0692
CB_TA_TB_TFIDF_C	TFIDF***	C	relative	0.0646

Tables 7 and 8 show that we have a considerable gain in retrieval performance when using the relative count instead of the classical one. The gain is clearer in the case of short documents and queries (image2012) than the case of long documents and queries (case2012).

Tables 5 and 6 show that using words as indexing terms is still more effective (from the retrieval performance point of view) than using concepts.

Concerning our formal contribution in the ad-hoc image-based retrieval task, we got a middle-rank. We still far from the best formal run. However, our contribution in the case-based retrieval task was more encouraging. We are ranked the second in the final list. We made a late fusion between two result set: 1- one resulting from applying DIR model to the case2012 corpus using words as indexing terms, and 2- another one resulting from applying DIR model to case2012 corpus using concepts as indexing terms.

Table 7. Comparing the retrieval performance of IR models using the classical count with their performance using the relative count (Algorithm 1). The corpus is image2012. The last column shows the gain in performance when using the relative count comparing to the performance of using the classical count

IR model	count type	MAP	Gain
PIV	classic	0.1071	+27%
	relative	0.1360	
BM25	classic	0.1034	+10%
	relative	0.1142	
DIR	classic	0.0861	+13%
	relative	0.0969	
JM($\lambda = 0.1$)	classic	0.1022	+29%
	relative	0.1318	
TFIDF	classic	0.1322	+7%
	relative	0.1410	

4 Conclusion

We present in this paper the contribution of the MRIM research group in the ImageCLEF2012 Medical Image Retrieval task. We describe the *frequency shift* problem that happens when moving from the word-space to the concept-space. The source of this problem is the heterogeneous change in the frequency of indexing terms when moving from the word-space to the concept-space.

We propose a solution to the frequency shift through a new counting strategy. Our counting strategy (Algorithm 1) depends on the hierarchy that could be built from the output of mapping tools. It also depends on the following two hypotheses:

– the relative count of a concept is directly proportional to the number of words in the corresponding text.
– the relative count of a concept is inversely proportional to the ambiguity level of the corresponding text.

For validating the effectiveness (from the retrieval performance point of view) of our new counting strategy, we participated in the ImageCLEF2012 campaign, more precisely, in the ad-hoc image-based retrieval and case-based retrieval tasks. Our experiments only depend on the textual data. For mapping text into concepts, we use MetaMap as a mapping tool and UMLS as an external resource containing concepts. In the case-based retrieval task, we only map the *title* and *abstract* of each document (we do not map the *fulltext*).

In the ad-hoc image-based retrieval task, we got a middle-rank. We still far from the best formal run. However, our contribution in the case-based retrieval task was more encouraging. We are ranked the second in the final list.

Moreover, the supplementary results that we present in this paper show a considerable gain in retrieval performance when applying our counting strategy

Table 8. Comparing the retrieval performance of IR models using the classical count with their performance using the relative count (Algorithm 1). The corpus is case2012. The last column shows the gain in performance when using the relative count comparing to the performance of using the classical count

IR model	count type	MAP	Gain
PIV	classic	0.0789	+10%
	relative	0.0865	
BM25	classic	0.0847	+7%
	relative	0.0895	
DIR	classic	0.0825	+8%
	relative	0.0893	
JM($\lambda = 0.7$)	classic	0.0863	+5%
	relative	0.0908	
TFIDF	classic	0.0830	+2%
	relative	0.0847	

(Algorithm 1) comparing to the classical counting strategy (count each concept as 1).

References

1. Karam Abdulahhad, Jean-Pierre Chevallet, and Catherine Berrut. The Effective Relevance Link between a Document and a Query. In *23rd International Conference on Database and Expert Systems Applications (DEXA 2012), Vienna, Austria*, pages 206–218, sep 2012.
2. Alan R. Aronson. Metamap: Mapping text to the umls metathesaurus, 2006.
3. Jean-Pierre Chevallet, Joo Hwee Lim, and Thi Hoang Diem Le. Domain knowledge conceptual inter-media indexing, application to multilingual multimedia medical reports. In *ACM Sixteenth Conference on Information and Knowledge Management (CIKM 2007), Lisboa, Portugal*, November 6–9 2007.
4. 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.
5. Christopher Dozier, Ravi Kondadadi, Khalid Al-Kofahi, Mark Chaudhary, and Xi Guo. Fast tagging of medical terms in legal text. In *Proceedings of the 11th international conference on Artificial intelligence and law*, ICAIL '07, pages 253–260, New York, NY, USA, 2007. ACM.
6. 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.
7. 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.

8. 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.
9. 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.