Medical Case-based Retrieval by using a language model: MIRACL at ImageCLEF 2012

Jihen Majdoubi, Hatem Loukil, Mohamed Tmar, and Faiez Gargouri

Multimedia InfoRmation system and Advanced Computing Laboratory,
Higher Institute of Information Technologie and Multimedia, University of sfax,
Tunisia
{Jihen.Majdoubi,Hatem.Loukil,Mohamed.Tmar,Faiez.Gargouri}@isims.rnu.tn
http://www.isimsf.rnu.tn/

Abstract. This paper reports the experiment results of the MIRACL team in participating in the medical case retrieval task of ImageCLEF 2012. In this paper, we propose our contribution for conceptual indexing of medical articles which uses a language model for selecting the best representative descriptors for each article.

Keywords: conceptual indexing, medical article, language model

1 Introduction

Started from 2004, the ImageCLEFmed (medical retrieval task) aims at evaluating the performance of medical information systems, which retrieve medical information from a mono or multilingual image collection. The medical retrieval task of ImageCLEF 2012 uses a subset of PubMed Central containing 305,000 images. This task consists of three subtasks: modality classification, ad-hoc retrieval and case-based retrieval. In our work, we are particularly interested in the case-based retrieval task, which was firstly introduced in 2009. This is a more complex task, but one that is closer to the clinical workflow. In this task, a case description, with patient demographics, limited symptoms and test results including imaging studies, is provided (but not the final diagnosis). The goal is to retrieve cases including images that might best suit the provided case description. Unlike the ad-hoc task, the unit of retrieval here is a case, not an image. For the purposes of this task, a "case" is a PubMed ID corresponding to the journal article [1].
This paper describes the contribution of the MIRACL[1] team (Multimedia InfoRmation systems and Advanced Computing Laboratory) in its participation at the medical retrieval track.
Our proposed conceptual indexing approach consists of three main steps. At the first step (Term extraction), being given an article, Medical Subject Headings

[1] http://www.miracl.rnu.tn/

(MeSH[2]) thesaurus and the NLP tools, our indexing system extracts two sets: the first is the article's lemma, and the second is the list of lemma existing in the MeSH thesaurus. After that, these sets are used in order to extract the Mesh terms existing in the document. At step 2, these extracted terms are weighed by using the measures CSW and SW that intuitively interprets MeSH conceptual information to calculate the term importance. The step 3 aims to recognize the MeSH descriptors that represent the document by using the language model.
The rest of this paper is organized as follows: Section 2 describes our conceptual indexing approach. Submitted results will be presented and discussed in section 3. We conclude the paper in section 4 by outlining some perspectives for future work.

2 Our conceptual indexing approach

Our indexing methodology as schematized in Figure 1, consists of four main steps: (a) Pretreatment (b) term extraction (c) term weighing and (d) selection of descriptors. In the following, we describe the structure of MeSH vocabulary and then we detail the steps of our indexing method.

2.1 MeSH thesaurus

The structure of MeSH is centred on descriptors, concepts, and terms.

- Each term can be either a simple or a composed term.
- A concept is viewed as a class of synonyms terms. The preferred term gives its name to the concept.
- A descriptor class consists of one or more concepts where each one is closely related to each other in meaning. Each descriptor has a preferred concept. The descriptors name is the name of the preferred concept. Each of the subordinate concepts is related to the preferred concept by a relationship (broader, narrower).

2.2 Pretreatment

The first step is to split text into a set of sentences. We use the Tokeniser module of GATE [2] in order to split the document into tokens, such as numbers, punctuation, character and words. Then, the TreeTagger [3] stems these tokens to assign a grammatical category (noun, verb,...) and lemma to each token. Finally, our system prunes the stop words for each medical article of the corpus. This process is also carried out on the MeSH thesaurus. Thus, the output of this stage consists of two sets. The first set is the articles lemma, and the second one is the list of lemma existing in the MeSH thesaurus.
The figure 2 outlines the basic steps of the pre-treatment phase.

[2] http://www.nlm.nih.gov/mesh

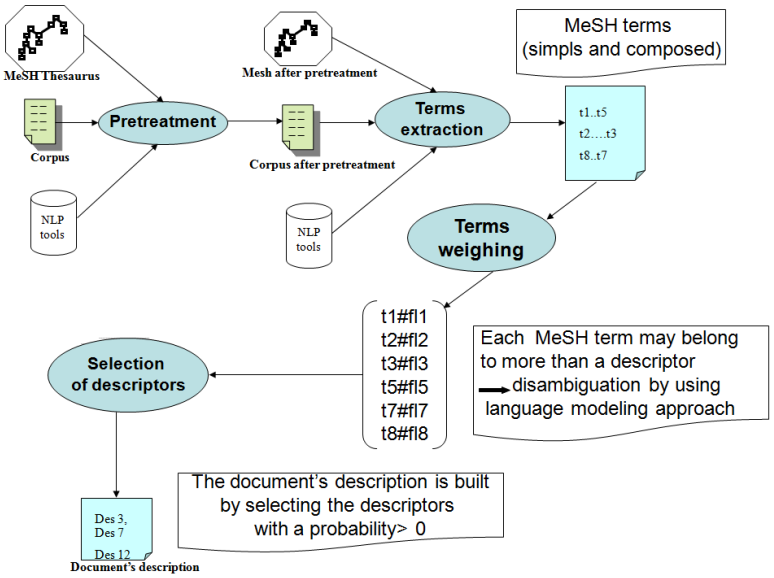

Fig. 1. Architecture of our indexing approach.

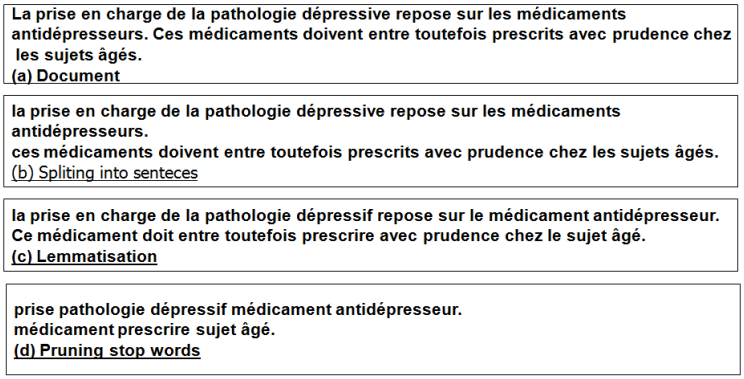

Fig. 2. Pretreatment step.

2.3 Term extraction

This step consists of finding the different Mesh terms existing in the set of terms generated by the pretreatment step. As mentioned above, a term MeSH can be either simple or composed. To extract the simple term, we project the Mesh thesaurus on the document by applying a simple matching. More precisely, each lemmatized term in the document is matched with the canonical form or lemma of MeSH terms. To recognize the composed terms, we have chosen to use YateA [4]. YateA (Yet Another Term ExtrAtor) is an hybrid term extractor developed in the project ALVIS. After text processing, YateA generates a file composed of two columns: the inflected form of the term and its frequency. For instance, as shown in figure 3 which describes the result of the term extraction process by using YateA, the term exercice physique occurs 6 times.

```
# Inflected form      Frequency
activité physique     16
activité sportive     9
exercice musculaire   8
exercice physique     6
effets bénéfiques     6
g de glucides   5
contrôle glycémique   5
insuffisance coronaire      5
index glycémique 4
risque cardiovasculaire     4
adaptation des doses  4
glycémie capillaire   4
sensibilité à l' insuline   4
fréquence cardiaque   3
hydrates de carbone   3
acides gras libres    3
autosurveillance glycémique 3
patient dnid    3
acides gras     3
dernier repas   3
profil lipidique 3
activité physique régulière 3
insuline rapide  3
```

Fig. 3. An excerpt of the result of YaTeA.

2.4 Term weighing

Given a set of extracted terms issued from the step of *Term extraction*, we calculate the terms weight by using two measures: the Content Structure Weight (CSW) and the Semantic Weight (SW) [5].

Content Structure Weight We can notice that the frequency is not a main criterion to calculate the CSW of the term. Indeed, the CSW takes into account the term frequency in each part of the document rather than the whole document.

For example, a term of the Title receives a higher importance ($*10$) than to a term that appears in the Paragraphs ($*2$). Table 2 shows the various coefficients used to weight the term locations. These coefficients were determined in an experimental way in [6].

Table 1. Weighing coefficients

term location	Weight of the location
Title (T)	10
Keywords (K)	9
Abstract (A)	8
Paragraphs (P)	2

The CSW of the term t_i in a document d is given as follows:

$$CSW(t_i, d) = \frac{\sum\limits_{A \in T,K,A,P} f(t_i, d, A) \times W_A}{\sum\limits_{A \in T,K,A,P} f(t_i, d, A)} \quad (1)$$

Where:

- W_A is the weight of the location A (see Table 2),
- $f(t_i, d, A)$ is the occurrence frequency of the term t_i in the document d at location A.

For example, the term *tumeur* exists in the document d_{1683}: 1 time in the title, 2 times in the abstract and 9 times in the Paragraphs,

$$CSW(tumeur, d_{1683}) = \frac{1 * 10 + 2 * 8 + 9 * 2}{1 + 2 + 9}$$

.

Semantic Weight (SW) The Semantic Weight of term t_i in the document d depends on its synonyms existing in the set of Candidate Terms ($CT(d)$) generated by the term extraction step. To do so, we use the $Synof$ function that associates for a given term t_i, its synonyms among the CT(d).
Formally the measure SW is defined as follows:

$$SW(t_i, d) = \frac{\sum\limits_{g \in Synof(t_i, CT(d))} f(g, d)}{|Synof(t_i, CT(d))|} \quad (2)$$

For a given term t_i, we have on the one hand its Content Structure Weight ($CSW(t_i, d)$) and on the other its Semantic Weight ($SW(t_i, d)$), its Local Weight

$((LW(t_i, d)))$ is determined as follows:

$$LW(t_i, d) = \frac{CSW(t_i, d) + SW(t_i, d)}{2} \tag{3}$$

By examining the equation 3, we can notice that the terms (simple or composed) are weighted by the same way. Despite the several works dealing with the weighing of composed terms, there is so far no weighing technique shared by the community [7]. In our approach, we applied the weighing method proposed by [8]. According to [8], for a term t composed of n words, its frequency in a document depends on the frequency of the term itself, and the frequency of each sub-term. For this purpose, it proposes the measure cf is defined as follows:

$$cf(t, d) = f(t, d) + \sum_{st \in subterms(t)} \frac{length(st)}{length(t)}.f(st, d) \tag{4}$$

where:

- $f(t, d)$: the occurrences number of t in the document d.
- $Length(t)$ represents the number of words in the term t.
- $subterms(t)$ is the set of all possible terms MeSH which can be derived from t.

For example, if we consider a term "cancer of blood", knowing that "cancer" is itself also a MeSH term, its frequency is computed as:

$$cf(cancer\ of\ blood) = f(cancer\ of\ blood) + \frac{1}{2}.f(cancer)$$

Consequently, in an attempt to take into account the case of composed terms, we calculate the csw measure as follows:

$$CSW(t_i, d) = \frac{\sum\limits_{A \in T,K,A,P} f(t_i, d, A) \times W_A}{\sum\limits_{A \in T,K,A,P} f(t_i, d, A)} + \sum_{st \in subterms(t_i)} \frac{length(st)}{length(t_i)}.f(st, d) \tag{5}$$

where: f(st,d) is the occurrences number of st in the document d.
It's important to note that in the case of simple terme, $subterms(t_i) = \emptyset$. Consequently the formulas presentd by equations 5 and 1 are equivalent.

Finally, the weight of a term t_i in a document d_j $(Weight(t_i, d_j))$ is calculated as follows:

$$Weight(t_i, d_j) = LW(t_i, d_j).ln(N/df) \tag{6}$$

where:

N: the total number of documents,
df (document frequency): the number of documents which term t_i occurs in.

2.5 Selection of descriptors

A term MeSH may be located in different hierarchies at various levels of specificity, which reflects its ambiguity. In the last years, due to the amount of ambiguous terms and their various senses used in biomedical texts, term ambiguity resolution becomes a challenge for several researchers [9][10][11]. Differently from the proposed works in the literature, our method assign the appropriate descriptor related to a given term by using the language model approach.
In our approach, to determine for an ambiguous term, its best descriptor, we have adapted the language model of [12] by substituting the query by the Mesh descriptor. Thus, we infer a language model for each document and rank Mesh descriptors according to their probability of producing each one given this model. We would like to estimate $P(des|d)$, the probability of generation a Mesh descriptor des given the language model of document d. For a collection D, document d and MeSH descriptor (des) composed of n concepts, the probability $P(des|d)$ is done by :

$$P(des|d) = P(d). \prod_{c_j \in relatedtoDes(des,d)} (1-\lambda)\,.P(c_j|d) + \lambda.P(c_j|D) \qquad (7)$$

Where:
RelatedtoDes (respectively RelatedtoCon) is the function that associates for a given descriptor des (respectively concept con) and a document d, the concepts (respectively terms) MeSH which are related to des (respectively con) in d. In the equation 7, we need to estimate two probabilities:

1. $P(c|D)$: the probability of observing the concept c in the collection D:

$$P(c|D) = \frac{f(c,D)}{\sum_{c' \in D} f(c',D)} \qquad (8)$$

   where $f(c,D)$ is the frequency of concept c in the collection D.
2. $P(c|d)$: the probability of observing a concept c in a document d:

$$P(c|d) = \frac{f(c,d)}{|concepts(d)|} \qquad (9)$$

   Where

$$f(c,d) = \prod_{t_j \in relatedtoCon(c,d)} LW(t_j,d)$$

Finally, to assign the appropriate sense (Best Descriptor (BD)) related to an ambiguous term (t_i) in the context of document (d_j), we retain the descriptor which maximizes $P(des|dj)$.

3 Results and discussion

The goal of our experiments is to answer the following question: Can our conceptual indexing approach improve the information retrieval process. these experiments are performed on the Case-based 2012 collection. This collection is based on a dataset containing the over 300,000 images of 75000 articles of the biomedical open access literature. 26 case-based topics are also provided where the retrieval unit is a case, not an image.
In order to make clear these experiments, we first present the experimental process and the techniques used for validation. Finally, we discuss the obtained results.

3.1 Experimental process

Our experimental process is undertaken as follows:

- Our process starts by dividing each article into a set of sentences. After tokenisation, lemmatisation of the corpus and the Mesh terms is ensured by TreeTagger[3]. Finally, a filtering step is performed to remove the stop-words.
- For each document d_j, of a test corpus, we determine the set of Candidate Terms(CT(d_j). After that, each term of this set will be weighed to determine its imprtance in d_j.
- For each document d_j, we select the set of Best Descriptor $BD(d_j)$.

Thus, each document d is presented as follows: $\overrightarrow{d} = (d_1, d_2 \dots d_n)$
where d_i is the probability of descriptor i in the document (see equation 7).
We can note that this indexing process is also performed on queries: after extracting the pertinent descriptors, the querie is presented as follows: $\overrightarrow{q} = (q_1, q_2 \dots q_n)$
where q_i is the weight (0 or 1 depending on whether the descriptor belongs or not to the query) of descriptor i in the query.

3.2 Experimental results

To determine the relevance of a document d_j to a query q: we apply 6 RSV (Retrieval Status Value) measures:

1. Okapi BM25:

$$rsv(d,q) = \sum_{j=1}^{n} \log \frac{N - n(q_j) + 0.5}{n(q_j) + 0.5} . \frac{f(q_j,d).(k_1+1)}{f(q_j,d) + k_1.(1 - b + b.\frac{|D|}{avgdl})}$$

   Where:

   - N: total number of documents in the collection.
   - $n(q_j)$: number of documents containing the descriptor j.

- $f(q_j, d)$: frequency of descriptor j in document d.
- k_1 et b: experimental parameters[3].
- $avgdl$: average length of documents.

2. Cosine measure:

$$rsv(\overrightarrow{d}, \overrightarrow{q}) = cos(\boldsymbol{d}, \boldsymbol{q}) = \frac{\boldsymbol{d}.\boldsymbol{q}}{|\boldsymbol{d}|.|\boldsymbol{q}|} = \frac{\sum_{k=1}^{n} d_k \times q_k}{\sqrt{\sum_{k=1}^{n} d_k^2 . \sum_{k=1}^{n} q_k^2}}$$

3. Dice coefficient:

$$rsv(d, q) = \frac{2 \times \sum_{k=1}^{n} d_k.q_k}{\sum_{k=1}^{n} d_k^2 + \sum_{k=1}^{n} q_k^2}$$

4. Jelinek measure:

$$RSV(d, q) = \prod_{des_i \in Des} (w_{ij})^{fi}$$

where:
- Des is the set of ds MeSH descriptors,
- w_{ij} is the weight of the descriptor des_i in the document d_j,
- fi is the frequency of the descriptor des_i in the querie q.

5. Jaccard measure:

$$rsv(d, q) = \frac{\sum_{k=1}^{n} d_k.q_k}{\sum_{k=1}^{n} d_k^2 + \sum_{k=1}^{n} q_k^2 - \sum_{k=1}^{n} d_k.q_k}$$

6. Overlap measure:

$$rsv(d, q) = \frac{\sum_{k=1}^{n} d_k.q_k}{min\left(\sum_{k=1}^{n} d_k^2, \sum_{k=1}^{n} q_k^2\right)}$$

Table 2 depicts the results of our submitted runs for the Case-based retrieval task in terms of MAP, P@10 and P@30.
By examining the table 2, we can note that the run *R1_MIRACL* shows the best rates in terms of MAP, P@10 and P@30. This result may be explained by the performance of the BM25 measure that takes into account term frequency and document size.
As we can see in table 2, the least effective results are generated by the run *R2_MIRACL* by usin the Cosine measure.Indeed, in this run, our indexing system does not extract any relevant document.

[3] In this experiment b=0, 75 and k_1 was fixed at 1, 6

Table 2. Results of our submitted runs for the Case-based retrieval task

Method	RunID	MAP	P@10	P@30
BM25	R1_MIRACL	0,0421	0,0538	0,0462
COSINE	R2_MIRACL	0	0	0
DICE	R3_MIRACL	0,012	0,0192	0,0218
JELINEK	R4_MIRACL	0,0196	0,0308	0,0282
JACCARD	R5_MIRACL	0,0024	0,0038	0,0013
OVERLAP	R6_MIRACL	0,0111	0,0192	0,0128

As shown in table 2, the results generated by the runs "*R3_MIRACL*" and "*R6_MIRACL*" are very similar.
R5_MIRACL perform worse than *R4_MIRACL* in all metrics. For example, the value of MAP generated by *R4_MIRACL* is equal to 0,0196. Concerning *R5_MIRACL*, it generates 0,0024 as value of MAP.

4 Conclusion

This article describes the conceptual retrieval approach of the MIRACL team for the ImageCLEF 2012 medical retrieval track, especially the case-based retrieval task. The results obtained by our submitted runs prove that our indexing method is useful to enhance the semantics of the document, which could be an interesting evidence to improve the retrieval effectiveness of medical retrieval systems. Our future work aims at incorporating a kind of semantic smoothing into the langage modeling approach. We also plan to use several semantic resources in the indexing process. We believe that multi-terminology based indexing approach can enhance the IR performance.

References

1. Müller, H., de Herrera, A.G.S., Kalpathy-Cramer, J., Fushman, D.D., Antani, S., Eggel, I.: Overview of the imageclef 2012 medical image retrieval and classification tasks. In: CLEF. (2012)
2. Cunningham, M., Maynard, D., Bontcheva, K., V.Tablan: Gate: A framework and graphical development environment for robust nlp tools and applications. ACL (2002)
3. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. International Conference on New Methods in Language Processing. Manchester (1994)
4. Aubin, S., Hamon, T.: Improving term extraction with terminological resources. In: Advances in Natural Language Processing. Volume 4139 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2006) 380–387
5. Majdoubi, J., Tmar, M., Gargouri, F.: Using the mesh thesaurus to index a medical article: Combination of content, structure and semantics. In: KES (1). (2009) 277–284

6. Gamet, J.: Indexation de pages web. Report of dea, universit de Nantes (1998)
7. Baziz, M., Boughanem, M., Aussenac-Gilles, N., Chrisment, C.: Semantic cores for representing documents in ir. In: Proceedings of the 2005 ACM symposium on Applied computing. SAC '05, ACM (2005) 1011–1017
8. Baziz, M.: Indexation conceptuelle guide par ontologie pour la recherche d'information. PhD thesis, Univ. of Paul sabatier (2006)
9. Andreopoulos, B., Alexopoulou, D., Schroeder, M.: Word sense disambiguation in biomedical ontologies with term co-occurrence analysis and document clustering. IJDMB **2**(3) (2008) 193–215
10. Stevenson, M., Guo, Y., Gaizauskas, R., Martinez, D.: Knowledge sources for word sense disambiguation of biomedical text. In: BioNLP '08: Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing, Association for Computational Linguistics (2008) 80–87
11. Duy, D., Lynda, T.: Sense-based biomedical indexing and retrieval. In: NLDB. (2010) 24–35
12. Hiemstra, D.: Using Language Models for Information Retrieval. PhD thesis, University of Twente (2001)