BioDI: A New Approach to Improve Biomedical Documents Indexing

Wiem Chebil[1,2], Lina Fatima Soualmia[1], and Stéfan Jacques Darmoni[1]

[1] Normandie Univ, CISMeF Team, LITIS-TIBS EA 4108,
Rouen University and Hospital, France
[2] Research Unit MARS, Monastir University, Tunisia
wiem.chebil@yahoo.fr,
{Lina.Soualmia,Stefan.Darmoni}@chu-rouen.fr

Abstract. The partial match between biomedical documents and controlled vocabularies allows to find in the documents more terms variants than those existing in the dictionaries. However, it generates irrelevant information. We propose a new approach for indexing biomedical documents with the Medical Subject Headings (MeSH) thesaurus that aims to overcome the limitation of the partial match. In fact, our indexing approach proposes to restrict the stemming process in the step of pretreatment. The step of the descriptors extraction is based essentially on the vector space model and combines semantic and statistic methods to compute a score to estimate the relevance of a descriptor given a document. The knowledge provided by the Unified Medical Language System (UMLS) is used then for filtering. The filtering method aims to keep only relevant descriptors. The experiments of our approach that have been carried out on the OHSUMED collection, showed very encouraging results.

Keywords: Partial match, biomedical documents, stemming, MeSH term, term weight, UMLS.

1 Introduction

The permanent increase of biomedical documents in the internet makes the task of their manual indexing with the biomedical controlled vocabularies become more difficult. To replace the tedious task of the human indexers, several approaches of biomedical documents indexing were proposed. Some of these approaches were based on an exact match [1-3] between the controlled vocabularies and documents which allows to find in the document only terms in the dictionaries. Other approaches were based on a partial (or approximate) match [4-7] which allows to (i) find in the document other terms variants than those existing in the dictionaries by applying the stemming, which reduces words (in the document and in the controlled resource) to their stems (or roots) (e.g. reacts, reacting, reacted, are reduced to react), or lemmatization, which reduces words to their based form (e.g. operation, operated are reduced to operate) (ii) extract multi-word terms that share a subset of their words with the document. The terms extracted in the two cases (i) and (ii) may be relevant which

H. Decker et al. (Eds.): DEXA 2013, Part I, LNCS 8055, pp. 78–87, 2013.

leads to improve the recall. But also they may be irrelevant which leads to decrease the precision. For examples: in the case of (i) a short stem may be confused with an acronym such as *"kid"* which is an acronym of term *"Keratitis, Ichthyosis, and Deafness"* and also a stem of term *"kidding"*. In addition, the existing tools for the lemmatization may don't recognize the exact grammatical classes (verb, noun..) of the biomedical vocabulary. In the case of (ii) the term *"breast cancer"* in a document may yield the Medical Subject Headings (MeSH) [8] terms *"testicular cancer"* and *"stomach cancer"* because the three terms share the word "cancer" [9].

In this paper, we propose a new approach for indexing biomedical documents using MeSH thesaurus denoted Biomedical Document Indexing (BioDI) that aims to overcome the limitation of partial match based approaches. Our first contribution is to restrict the stemming process. In addition, to enhance the relevance estimation of a term[1], we compute a semantic, statistic and structure based score that gives an importance to the position of a word in the document as well as to the occurrence of the terms words in the same phrase. Another main contribution of our approach is to exploit the knowledge provided by the Unified Medical Language System (UMLS) [10] to filter the extracted descriptors. The filtering allows to keep relevant descriptors among those extracted in the case (ii).

The paper is organized as follow: the second section presents the related work. The section 3 details the steps of our indexing approach. In the section 4, we describe the experiments and the generated results that are discussed in the section 5. Finally, in the section 6 we conclude and present our future work.

2 Related Work

Several research approaches for indexing biomedical documents have been proposed. We focus on some of them. Pouliquen *et al.* [1] computed a statistic weight based on TF-IDF for each term automatically extracted from the document using a method based on NLP (Natural Language Processing). These terms are then matched to the terms of the ADM (assistance with the medical diagnosis) dictionary. Jonquet *et al.* [2] applied the Mgrep tool for extracting concepts from 200 biomedical ontologies, and computed a score for each generated annotation according to its origin (preferred term, non-preferred term, synonym term …etc.). Mukherjea *et al.* [3] developed BioAnnotator a new tool for indexing biomedical documents. It uses a parser to identify noun phrases from a document and then matches them to the UMLS concepts using a rule engine. Zhou *et al.* [4] proposed to annotate documents with only the most significant words in the UMLS Meta-thesaurus. Ruch [5] proposed an indexing approach denoted by Eagl that combined two models: the Vector Space Model (VSM) and a regular expression pattern matcher. The indexing technique of Aronson *et al.* [6] is based on three methods: the first uses MetaMap (software tool for English that allows mapping document to the UMLS concepts), the second is the tri-gram method and the last one is the KNN (the k-Nearest Neighbors). Majdoubi *et al.* [7] used the VSM to extract MeSH terms and then computed a statistic and semantic weight for ranking these terms.

[1] In this paper we denoted by terms all preferred and no preferred terms in MeSH thesaurus.

3 The Steps of Our Proposed Approach

Our approach BioDI is based on VSM [11] which was initially applied in Information Retrieval (IR) to compute a similarity between user's query and the document. In our approach, as in [5] and [7] the query is replaced by a term. Our method is composed of 4 steps: pretreatment, descriptors extraction, filtering and final ranking.

3.1 Step 1: Pretreatment

The step of pretreatment consists of 4 tasks: (i) dividing the document into phrases (ii) removing punctuation (iii) pruning stop words (iv) stemming. The last three tasks are applied also on MeSH terms. Let *"The binding of acetaldehyde to the active site of ribonuclease: alterations in catalytic activity and effects of phosphate."* a title of a document, after the pretreatment this title become *"bind acetaldehyd activ site ribonucleas alter catalyt activ effect phosphat"*. For the stemming, we chose to use PORTER Algorithm [12]. During the stemming process a short stem can be confused with an acronym. Thus, we propose to restrict the applying of the stemming process only on words that the length of their stems is equal or upper than a threshold Ts which is fixed experimentally (see section 4).

3.2 Step 2: Descriptors Extraction

The step of descriptors extraction begins with extracting all the preferred and no preferred terms. To do, we compute a similarity between each term and the document using cosine similarity. The terms candidates are those having a similarity upper or equal than a tuned Threshold Tcos. Then, we compute a weight for each extracted term. The final score of each selected term is the sum of its similarity with the document and its weight. After, the corresponding descriptors are assigned to the terms. The score of a descriptor is the score of its term. As in [13], if a descriptor corresponds to more than one term among the terms candidates, it will have the highest score. The term that gives its score to the descriptor is denoted the Representative Term (RT).

Similarity between a Term and a Document

Let $\{T_1 \ldots T_i \ldots T_z\}$ the set of MeSH terms. Each term T_i is composed of a set of words Ti $\{ wdt_1 .. wdt_k ... wdt_t \}$ with t the number of words in a term. T_i is represented by the vector VT ($WWT_1 \ldots WWT_k .. . WWT_t$), WWT_k is the weight of word wdt_k in the MeSH. The document DOC is represented by the vector VDOC ($WWDoc_1 \ldots WWDoc_k \ldots WWDoc_t$), $WWDoc_k$ is the weight of the word wdt_k in the DOC. The cosine similarity is computed then between VT and VDOC (1) and denoted Sim(T_i, DOC). We consider that WWT-WWDoc is the weight combination of wdt_k.

$$\text{Sim}(T_i, \text{DOC}) = \frac{\sum_{k=1}^{t} WWT_k \times WWDoc_k}{\sqrt{\sum_{k=1}^{t} (WWT_k)^2 \times (WWDoc_k)^2}} \quad (1)$$

Weight of Word in the Document (WWDoc)
For computing WWDoc, we use a weight based on the frequency of a word in the document that takes into consideration the position of the word in the document (in the title, in the abstract or in the paragraphs). This weight is denoted Word Average Frequency in the Document (WAFDoc) (2). We consider that the key words are more dissipate in the paragraphs and mixed with non relevant words (comparing to the abstract and title), while key words are more condensed in the title (comparing to abstract and paragraphs). Thus, we assign the following coefficients to each position in document: Position Coefficient (PC) =8 to the title, PC=4 to the abstract, PC=2 to the paragraphs.

$$\mathrm{WWDoc}_k = \mathrm{WAFDoc}_k = \frac{\sum_{p=1}^{r} FQ\left(wdt_k, P\right) \times PC_p}{\sum_{p=1}^{r} PC_p} \tag{2}$$

- FQ(wdt_k, P): Frequency of wdt_k in the position P
- P=1:Title; P=2: Abstract; P=3: Paragraph
- $PC_{p:}$ The coefficient of the position P.
- r: The number of the positions coefficients

Weight Word in Term (WWT). We consider WF_k-IDF_k[2](Word frequency – Inverse document frequency) [11] is the weight of the word wdt_k in the term.

$$WWT_k = WF_k \times IDF_k \tag{3}$$

$$\mathrm{WF}_k = \frac{FWT_k}{\max_{e:1->t}\left(FWT_e\right)} \tag{4}$$

- t: is the number of words in a term
- FWT_k: Frequency of wdt_k in a term

We consider that the normalized frequency of a word wdt_k in the term is equal to its frequency in the descriptor containing this term because this term may be the RT of the descriptor. The frequency of a word in a descriptor is its frequency in all the terms of the descriptor. We consider also that the IDF of wdt_k is equal to the logarithm of the number of the descriptors containing in their terms at least one occurrence of wdt_k divided by the total number of the descriptors in MeSH.

$$\mathrm{IDF}_k = -\log\left(\frac{FWDM_k}{ND}\right) \tag{5}$$

[2] Instead of using "TF" (term frequency) we used "WF" (word frequency) because we consider that a term can be composed of one word or can be a multi-word term.

- ND: The total Number of Descriptors in MeSH
- FWM_k : The Frequency of the Word wdt_k in MeSH (the number of descriptors having at least one occurrence of wdt_k).

Weight of a Term in the Document (WTDoc)
We propose a new weight of a term T_i in the document denoted TAFDoc. This weight is based on WAFDoc and it is equal to the sum of the weights WAFDoc of all the words of T_i (the t words) divided by t. The results is majored by a coefficient cof>1 if all the t words of T_i are at least one time in the same phrase in the document. In fact, we hypothesize that words in the same phrase are more likely to cover the same meaning. The coefficient cof is experimentally tuned.

$$\mathrm{WTDoc}_i = \mathrm{TAFT}_i = \frac{\sum_{k=1}^{t} \mathrm{WAFDoc}\left(wdt_k\right)}{t} * cof \quad (6)$$

-cof >1 if the term words are in the same phrase at least one time in the document.
-cof= 1 if the term words are not in the same phrase

The Score of a Descriptor
The score of a descriptor is the maximum score of its terms (7). The term having the maximum score is the Representative Term (RT). The score of a term is the sum of its similarity with the document and its weight in the document (8).

$$Score(D) = \max_{j:1->n}(Score(T_j)) \quad (7)$$

$$\mathrm{Score}\,(\mathrm{T}_i) = \mathrm{WTDoc}_i + Sim(T_i, DOC) \quad (8)$$

n: The number of terms of a descriptor D

3.3 Step 3: Filtering

The aim of this step is to keep only the relevant descriptors among those having a multi-word RT that at least one of its words doesn't occur in the document. In fact, we classified the no extraction of these relevant descriptors as a category of indexing errors in [14]. This step consists of dividing the set of MeSH descriptors generated in the previous step into two sets of descriptors: the first set is denoted Principal Index (PI) and the second is denoted Secondary Index (SI). The PI contains the descriptors that their RT terms have all their words in the document. These Descriptors are denoted Principal Descriptors (PD). The SI contains the descriptors that their RT terms have a subset of their words in the document. These descriptors are denoted Secondary Descriptors (SD). We separate the PD and SD because we are based on the assumption that MeSH terms having all their words in the document are more likely to be correct. Then the relevant descriptors in SI are added from the SI to the PI. To do this task, first of all, the PD in PI are ranked using the score (7). Thus we have

PI= {PD_1,…PD_i …PD_v}, PD_i is a principal descriptor having the rank i and v is the number of PD in PI. Then, we propose to compute a score S for each SD (9). This score S is based on the co-occurrences of MeSH descriptors in MEDLINE and the semantic relations between MeSH descriptors provided by the semantic work of UMLS [10]. In fact, our assumption is that the SD is more likely to be correct if it is more co-occurrent or/and have more semantic relations with exactly the L first PD in PI that are considered the most relevant. L is the length of a window that contains the L first PD. For example, according to the proposed formula of S (9) if we fix L=1, that means S(SD) is equal to the sum of the number of co-occurrences and relations between the SD and the PD having the rank 1(PD_1). If L=2, that means S(SD) is equal to the sum of the number of the co-occurrences and the semantic relations between the SD and the two PD having the rank 1 and 2(PD_1 and PD_2). If SD doesn't co-occur or doesn't have any semantic relation with one of the L PD, or if the SD has a score S lower than a tuned threshold T, it isn't be added to PI. The threshold T was tuned according to the value of L.

$$S(SD) = \sum_{i=1}^{L} CF(SD, PD_i) + \sum_{i=1}^{L} NR(SD, PD_i) \quad (9)$$

CF: Co-occurrence Frequency; NR: Number of the semantic Relations

3.4 Step 4: Final Ranking

The SD selected in the previous step will be added to PI, the final index (FI) is thus constructed. The descriptors of FI are re-ranked using the score (7).

4 Experiments and Results

To test our approach we selected randomly 6,000 citations among the OHSUMED collection[3] composed of 4,591,015 MEDLINE citations. Each selected citation is composed of title and an abstract. The content of the title is merged with the content of the abstract when indexing the citations. We don't consider the sub-headings in our approach. To evaluate BioDI, we used the classical measures of Precision (P), Recall (R) and F-score (Fs). The precision is the number of correct descriptors divided by the total number of descriptors automatically generated. The recall is the number of correct descriptors divided by the number of descriptors manually extracted. F-score combines precision and recall with an equal weight [15].

4.1 Evaluation of the Terms Extraction

The different cases experimented in order to fix the adequate value of Ts are: Ts>=3, Ts >=4, Ts>=5 and Ts>=6. We experimented also the stemming without considering the stem length and the case where we didn't stem the words. For each of these cases, we applied the cosine similarity between the MeSH terms and the document and we

[3] http://trec.nist.gov/data/t9_filtering.html

tested the performance of the proposed weight combination WFIDF-WAFDoc as well as others combinations: 1-1 (assigning 1 to the weight of word in the document if the word exist in the document, 0 else), IDF-WFIDF, WFIDF-WFIDF. When computing the cosine similarity a big number of terms are extracted, thus, only those having a similarity upper than a tuned threshold Tcos equal to 0.8 were selected as candidates for indexing the document. In order to generate the results of these experiments we affected for each extracted term its correspondent descriptor because the manual indexing has been carried out using descriptors. The table 1 illustrates the obtained results of the experiments described above.

Table 1. Results of terms extraction[4]

	1-1 (or 0)	IDF-WFIDF	WFIDF-WFIDF	WFIDF -WF	WFIDF-WAFDoc
	P-R-Fs	P-R-Fs	P-R-Fs	P-R-Fs	P-R-Fs
A	0.180-0.30-0.225	0.174-0.310-0.199	0.175-0.330-0.228	0.177-0.340-0.232	0.179-0.360-0.239
B	0.170-0.32-0.222	0.161-0.320-0.214	0.163-0.350-0.222	0.165-0.400-0.233	0.168-0.410-0.238
C	0.159-0.48-0.238	0.148-0.520-0.223	0.150-0.521-0.230	0.155-0.535-0.240	**0.158-0.570-0.246**
D	0.121-0.520-0.196	0.113-0.550-0.187	0.115-0.560-0.190	0.117-0.580-0.194	0.119-0.600-0.198
E	0.112-0.57-0.187	0.106-0.605-0.180	0.107-0.610-0.182	0.109-0.620-0.185	0.110-0.630-0.187
F	0.100-0.60-0.171	0.090-0.615-0.157	0.092-0.620-0.160	0.094-0.630-0.164	0.099-0.650-0.172

A: Without stemming, B: Ts>= 6, C: Ts>=5, D: Ts>=4, E: Ts >=3, F: Stemming without considering the length of word stem.

4.2 Experiments and Results of Generating the PI

The aim of these experiments is to compute the precision, recall and f-score of the PI where descriptors are ranked using the score (7) that takes into account the similarity between the MeSH terms and the document and also the weight of the terms in the document. In the first experiment (section 5.2) we evaluated the performance of proposed similarity. In this experiment we tested the performance of the proposed weight TAFDoc through two experiments. First of all, we varied the value of the coefficient cof and we compute the TAFDoc. We carried out this test, in order to find the best value of the coefficient cof. Then, we evaluated the performance of BM25 term weighting model used in [16] to compute the weight of concepts, which is compared

[4] We kept three numbers after the point because the results are very close to each other.

Table 2. Results of generating PI with varying cof and comparing TAFDoc to BM25

	BM25	TAFDoc		
		cof=1	cof=1.5	cof=1.6
P-R-Fs(rank1)	0.61-0.17-0.26	0.68-0.19-0.28	**0.71-0.21-0.31**	0.70-0.18-0.28
P-R-Fs(rank10)	0.17-0.43-0.23	0.23-0.43-0.28	**0.29-0.40-0.33**	0.28-0.37-0.23
P-R-Fs(rank15)	0.19-0.47-0.25	0.21-0.45-0.27	**0.25-0.43-0.30**	0.24-0.40-0.29

to the performance of TAFDoc. For each one of the two experiments, a new score (7) was computed with keeping always the proposed similarity, and PI is re-generated.

Table 2 presents the results of these experiments at ranks1, 10 and 15[5].

4.3 Evaluation of the Filtering Step and Final Ranking

In order to evaluate the step of filtering we generate final results for different values of L. For each value of L a new value of T is experimentally tuned. These results are shown in table 3.

Table 3. Results after filtering and final ranking at rank 1, 10 and 15

	L=1/T=70	L=2/T=50	L=3/T=10	L=4/T=4	L=5/T=5	L=6/T=6
P-R-Fs(rank1)	0.71-0.21-0.31	0.71-0.21-0.31	**0.71-0.21-0.31**	0.71-0.21-0.31	0.71-0.21-0.31	0.71-0.21-0.31
P-R-Fs(rank10)	0.31-0.52-0.38	0.35-0.51-0.40	**0.41-0.50-0.45**	0.37-0.48-0.42	0.35-0.45-0.38	0.30-0.40-0.34
P-R-Fs(rank15)	0.26-0.55-0.34	0.32-0.54-0.39	**0.36-0.52-0.42**	0.34-0.49-0.39	0.30-0.48-0.35	0.27-0.45-0.33

4.4 Evaluation of Some Other Approaches

To highlight the effectiveness of our indexing approach, we compared the performance of BioDI to the performance of some other approaches. In fact, we evaluated MaxMatcher [4], and Eagl [5] which are partial match based approaches and BioAnnotator [3] which is an exact match based approach. The results of this evaluation are detailed in table 4.

Table 4. Evaluation of MaxMatcher, Eagl and BioAnnotator at ranks 1, 10 and 15

	MaxMatcher	Eagl	BioAnnotator	BioDI
P-R-Fs(rank1)	0.69-0.18-0.27	0.62--0.18-0.27	0.70-0.14-0.22	0.71-0.19-0.29
P-R-Fs(rank10)	0.32-0.46-0.37	0.25-0.40-0.30	0.33-0.24-0.26	0.41-0.50-0.45
P-R-Fs(rank15)	0.27-0.50-0.35	0.17-0.54-0.25	0.29-0.27-0.26	0.36-0.52-0.42

[5] We didn't test other ranks upper than 15 because the average number of keywords in MEDLINE citations is 15 [5].

5 Discussion

The table 1 shows that, for all the weights combinations, the precision of terms extraction is higher without stemming, and then it decreases when the stemming is applied with considering Ts. The more Ts decrease the more the precision also decreases. In addition, we can observe that the recall is very low without applying stemming and its value is significantly higher when Ts>=6. Moreover, according to the values of f-score we can deduce that the stemming process performs well when Ts>=5. When analyzing table 2, we can see that the performance of the VSM is better (according to the f-score value) when applying the weight combination WFIDF-WAFDoc than the 4 others weights combinations though 1-1(or 0) gives a slightly higher precision. We can deduce also that WAFT when combined with WF-IDF performs well than WF and WF-IDF. The table 2 shows that the best results of generating PI when applying the weight TAFDoc are scored when cof=1.5. We can conclude also according to table 2 that TAFDoc is more effective than BM25. These results show the well interest of: (i) taking into account the word position in the document (ii) giving more importance to terms having their words in the same phrase. According to the table 3 (final results), we can observe that there is no change in the performance of BioDI after PI's expansion when the first descriptor is retrieved. Nonetheless, at rank 10 and 15 an improvement of results can be seen. Obviously, descriptors having a part of the words of their RT doesn't occur in the document don't have the best weight. We can see also, that the expansion method performs better at L=3 than at the other values of L. In addition, when L=6 we have a remarkable decrease of results. Indeed, at L>5 it's less possible to find a SD which is co-occurent or have semantic relations with exactly the L first PD. The evaluation of Maxmatcher, Eagle and BioAnnotator (table 4) confirms the effectiveness of BioDI which out performs the three other approaches in the different ranks and in term of precision, recall and F-score when L is equal to 3, 4 and 5. Thus, we can deduce that the performance of our approach is closely dependent on the parameters L, cof and Tcos that must be well tuned to allow BioDI to outperform the other approaches.

6 Conclusion and Future Work

We presented in this paper our indexing approach that proposes to improve the partial match between biomedical documents and the controlled vocabularies. Our main contributions are: (i) restricting the stemming process to the words that their stem length is equal or upper than 5 (ii) computing a new score to estimate the relevance of a MeSH descriptor given a document. This score takes into account the position of a word in the document and gives more importance to terms having all their words in the same phrase (iii) filtering the index using the semantic and statistic resources of UMLS in the aim of keeping only relevant descriptors among those having a subset of their RT in the document. The several experiments carried out on the OHUMED corpus showed that BioDI allows improving partial match as well as exact match between biomedical documents and biomedical terminologies. We aim after these encouraged results to test the proposed approach with computing the score (9) between SD and all possible combinations of the first PD. In addition, we aim to

compare our approach to more others approaches. We are working also on applying our approach on the corpus of the catalog and index of french-language health internet resources (CISMeF)[6].

References

1. Happe, A., Pouliquen, B., Burgun, A., Cuggia, M., Beux, P.L.: Automatic concept extraction from spoken medical reports. I. J. Medical Informatics 70(2-3), 255–263 (2003)
2. Jonquet, C., LePendu, P., Falconer, S.M., Coulet, A., Noy, N.F., Musen, M.A., Shah, N.H.: NCBO Resource Index: Ontology-based search and mining of biomedical resources. J. Web Sem. 9(3), 316–324 (2011)
3. Mukherjea, et al.: Enhancing a biomedical information extraction system with dictionary mining and context Disambiguation. IBM Journal of Research and Development 48(5/6), 693–701 (2004)
4. Zhou, X., Zhang, X., Hu, X.: MaxMatcher: Biological concept extraction using approximate dictionary lookup. In: Yang, Q., Webb, G. (eds.) PRICAI 2006. LNCS (LNAI), vol. 4099, pp. 1145–1149. Springer, Heidelberg (2006)
5. Ruch, P.: Automatic assignment of biomedical categories: toward a generic approach. Bioinform. J. 22(6), 658–664 (2006)
6. Aronson, A.R., Mork, J.G., Gay, C.W., Humphrey, S.M., Rogers, W.J.: The NLM indexing initiative's medical text indexer. Med. Health Info. 11(1), 268–272 (2004)
7. Majdoubi, J., Tmar, M., Gargouri, F.: Using the MeSH thesaurus to index a medical article: combination of content, structure and semantics. In: International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, KES, vol. (1), pp. 277–284 (2009)
8. Nelson, S.J., Johnson, W.D., Humphreys, B.L.: Relationships in Medical Subject Heading. In: Relationships in the Organization of Knowledge, pp. 171–184. Kluwer Academic Publishers (2001)
9. Trieschnigg, D., Pezik, P., Lee, V., et al.: MeSH Up: effective MeSH text classification for improved document retrieval. Bioinformatics 25(11), 1412–1418 (2009)
10. Bodenreider, O.: The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Research 32(4), 267–270 (2004)
11. Singhal, A.: Modern information retrieval: a brief overview. IEEE Data Eng. Bull. 24(4), 35–43 (2001)
12. Porter, M.: An algorithm for suffix stripping. Program 14(3), 130–137 (1981)
13. Couto, F.M., Silva, M.J., Coutinho: Finding genomic ontology terms in text using evidence content. BMC Bioinformatic 6, (S-1) (2005)
14. Chebil, W., Soualmia, L.F., Dahamna, B., Darmoni, S.J.: Automatic indexing of health documents in French: Evaluating and analysing errors. IRBM BioMedical Engineering and Research 33(2), 129–136 (2012)
15. Manning, C.D., Schütze, H.: Fondations of statistical natural language processing, pp. 534–536. MIT Press, Cambridge (1999)
16. Dinh, D., Tamine, L.: Towards a context sensitive approach to searching information based on domain specific knowledge sources. Web Semantics: Science, Services and Agents on the World Wide Web 12-13, 41–52 (2012)

[6] `http://www.chu-rouen.fr/cismef/`