

Cross Language Information Retrieval for Biomedical Literature

Martijn Schuemie*
Erasmus MC
m.schuemie@erasmusmc.nl

Dolf Trieschnigg†
University of Twente
trieschn@ewi.utwente.nl

Wessel Kraaij‡
TNO
kraaijw@acm.org

Abstract

This workshop report discusses the collaborative work of UT, EMC and TNO on the TREC Genomics Track 2007. The biomedical information retrieval task is approached using cross language methods, in which biomedical concept detection is combined with effective IR based on unigram language models. Furthermore, a co-occurrence method is used to select and filter candidate answers. On its own, the cross lingual approach and the filtering do not strongly improve retrieval results. However, the combination of approaches does show a strong improvement over the monolingual baseline.

1 Introduction

The TREC Genomics track focuses on literature disclosure in the genomics domain. This report discusses the collaborative work of the University of Twente, Erasmus Medical Center and TNO on the TREC Genomics Track 2007.

The 2007' task is to answer 36 biological information needs using a fixed collection of around 160.000 full-text articles from Highwire Press. The information needs require answers in the form of a particular entity type, ranging from “genes” to “signs or symptoms”. The systems should return passages, short strings of text from the original documents, containing the answer.

During last year we achieved reasonable (document retrieval) results using unigram language models without using any additional knowledge from biological databases. In this year's approach we focus on:

- A cross language approach to improve query language model estimation. A concept-tagged corpus is used as a parallel corpus to create a statistical translation dictionary (translating concepts to tokens and vice versa). The dictionary is used to translate concepts detected in the topic description to a query language model.
- Using a co-occurrence based method to determine and filter candidate answers.

In the following section the approach will be discussed in more detail. After that, the runs and achieved results are discussed.

2 Approach

Our approach consists of an *offline* and an *online* process.

The offline process, which is carried out beforehand, consists of the following steps:

*Biosemantics Group, Medical Informatics Department, ErasmusMC University Medical Center, Rotterdam, The Netherlands

†Human Media Interaction group, University of Twente, Enschede, The Netherlands

‡TNO Information and Communication Technology, Delft, The Netherlands

- Creation of the ontologies for each of the entity types used in the topics;
- Biomedical concept recognition on the collection to create a concept-tagged collection;
- Using the concept-tagged collection to create a statistical translation dictionary (consisting of concept-word pairs and a probability);
- Creation of an index of the valid passages in the collection.

The offline process is discussed in section 3. The online process, the retrieval process which is carried out for each of the topics, consists of the following steps:

- Use the dictionary to create a query language model for the topic;
- Find the best matching spans;
- Extract candidate answer concepts based on co-occurrence and select spans containing candidate concepts.

The online process is discussed in section 4.

3 Offline process

3.1 Collection preprocessing

The document collection preprocessing is similar to last year [6]. The HTML documents are split into sections with corresponding section titles, using several different templates to support the differences in document formatting. Texts within <TABLE>, <A> (hyperlink), and tags are ignored, thus also ignoring figure captions. Sentences are split using a simple algorithm developed by our team, using an unsupervised sentence boundary detection approach (based on the work of A. Mikheev [4]). For each sentence the byte offsets of the first and the last character in that sentence are reported. Last year the sections materials & methods, literature references and acknowledgements were removed. However, after observing that last year's ground truth data also contains relevant passages from these sections, we decided to include them for this year's participation.

3.2 Ontology creation

To detect the concepts in the documents, we needed an ontology that contains one or more terms for each concept. For every category of concepts a specific source of information was used:

GENES and PROTEINS We made no distinction between genes and proteins. For five organisms (Homo sapiens, Mus musculus, Rattus Norvegicus, Drosophila melanogaster, and Caenorhabditis elegans) we extracted the gene and protein names from Swiss-Prot¹ and Entrez-Gene². Entries with matching database identifiers were combined. Using data from HomoloGene³, homologs in different species were also combined into a single concept.

MUTATIONS, CELLS, and NEOPLASMS For these categories, we used the appropriate subtrees from the MeSH (Medical Subject Headings⁴) thesaurus.

DRUGS, DISEASES and SIGNS AND SYMPTOMS For these categories, we used the concepts belonging to the appropriate semantic types from the UMLS (Unified Medical Language System⁵) thesaurus.

¹<http://www.expasy.org>

²<http://www.ncbi.nlm.nih.gov/sites/entrez?db=gene>

³<http://www.ncbi.nlm.nih.gov/sites/entrez?db=homologene>

⁴<http://www.ncbi.nlm.nih.gov/sites/entrez?db=mesh>

⁵<http://www.nlm.nih.gov/research/umls>

TOXICITIES We manually created a thesaurus for this category, containing 17 different toxicities.

MOLECULAR FUNCTIONS For this category, we used the appropriate subtree from the Gene Ontology⁶.

PATHWAYS We extracted pathway names from Medline using the following procedure: First, we selected a subset of documents using the query “gene OR protein”, limited to publications in the last five years. We then extracted passages that matched the pattern “the <x> pathway”, where <x> could be a maximum of four words. If <x> also occurred in the pattern “a <x> pathway” it was removed from the list, to correct for spurious matches such as “the unknown pathway” or “the complex pathway”. We furthermore manually checked the remaining list of potential pathway names.

STRAINS and ANTIBODIES We were not able to construct thesauri for the categories STRAINS and ANTIBODIES.

3.3 Concept recognition

Concept recognition was performed using our Peregrine tool [5]. Both text and ontology terms were tokenized. A token was considered to be a consecutive string of letters and/or numbers, all other characters were discarded. Tokens in the text were matched to the tokens in the ontology terms to detect concepts in the text. For all concepts except genes or proteins, the tokens were reduced to their normalized form using the LVG normalizer [3] prior to matching.

Because gene and protein names are often highly ambiguous, we used several simple rules to detect and possibly resolve ambiguous gene and protein names:

1. We first determined whether a term is ambiguous. A term is considered ambiguous if it refers to more than one gene in the dictionary, or when it is shorter than six characters and does not contain a number. A non-ambiguous term will automatically be assigned.
2. An ambiguous term will only be assigned if a synonym is found in the same document, or the term is the preferred name of the gene.

Because the simple disambiguation is rather strict, we also allowed ambiguous terms to be assigned if a keyword was found in the same document. A keyword is a word (i.e. a token) that occurs in any of the long-form names of the gene, and appears less than n times in the dictionary as a whole. We have achieved the best results with $n = 1,000$. For instance, in the term “Prostate Specific Antigen” the word “Prostate” appears less than 1,000 times in the dictionary. If the ambiguous synonym “PSA” is encountered in text, and the word “Prostate” is also encountered, the gene name is recognized.

Participation in the gene normalization task of the Biocreative 2 competition showed that this approach leads to a precision of 75% and a recall of 76% when linking human gene mentions in text to specific genes [5].

3.4 Statistical translation dictionary

The tagged corpus is used as a parallel corpus to create a statistical translation dictionary. The dictionary consists of {conceptid, token, probability}-triples, indicating the probability that a particular token t denotes a particular biomedical concept c . The dictionary is created by tokenizing the concept-tagged collection and by counting the assignment of a token to a concept. Figure 1 shows an example of tokenization and the assignment to concept ids.

The probability $P_{dict}(t|c)$ is estimated as follows:

$$P_{dict}(t|c) = \frac{freq(t, c)}{\sum_{t'} freq(t', c)},$$

⁶<http://www.geneontology.org>

where $freq(t, c)$ is the number of times a token in the tagged corpus is assigned to the concept c .

Figure 2 shows a partial entry list for the concept “TNF kappa B”.

<i>Original</i>	Regulation	by	egr-1,	c-jun	and	NFkB	transcription	factors.
<i>Tokens</i>	regulation		egr 1 egr1	c jun cjun		nfkB	transcript	factor
<i>Conceptid</i>	-1		2060534	2045920		2055956	-1	-1

Figure 1: Example of tokenizing a sentence and assigning concepts to them.

<i>Token</i>	<i>Count</i>	<i>P(t c)</i>
tnf	122234	0.452
factor	43857	0.162
necrosi	42060	0.156
tumor	39467	0.146
alpha	8938	0.033
tumour	4417	0.016
factoralpha	3145	0.012
tnfalpha	2813	0.010
necrosis	1824	0.007
cachectin	493	0.002
tnfa	325	0.001
dif	305	0.001
tumournecrosi	130	0.0005
...

Figure 2: Dictionary entries for the concept “TNF kappa B” (partially shown)

The dictionary can be used to determine the ambiguity and specificity of a token for a particular concept. How the dictionary is used is explained in section 4.1.

3.5 Span indexing

The preprocessed text collection is split into valid spans, as defined by the task. The spans are treated as documents in a unigram index. For the tokenization we use the method evaluated by Trieschnigg et al [7]. This method showed to improve document mean average precision with 27% compared to a basic tokenizer on the 2006 topics.

The method works as follows:

- The input text is split on whitespace, the resulting strings are treated as a group:
 - Extract tokens from the group consisting of either letters or digits.
 - If more than one token was extracted from a group, add the concatenation of these tokens as an additional compound token.
- For each obtained token: skip if it appears on a stopword list, do Porter stemming if the token does not contain uppercase letters.

4 Online process

Figure 3 gives an overview of the online process.

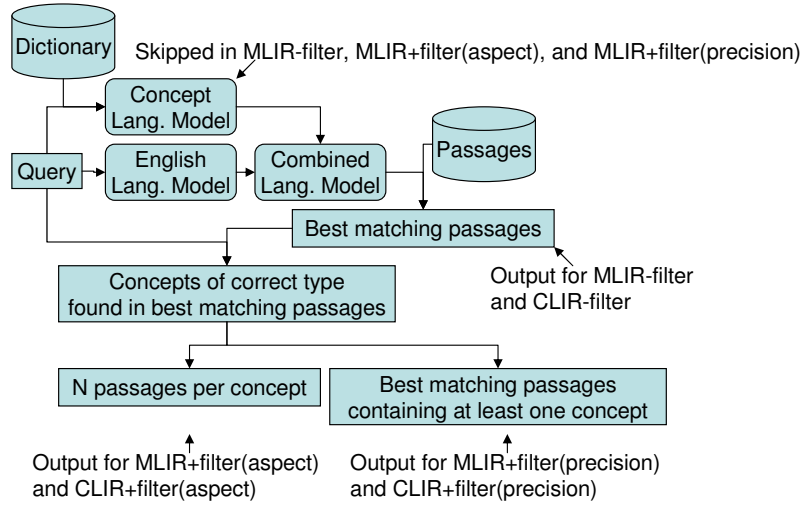


Figure 3: Overview of the online process

4.1 Query language model estimation and span retrieval

For the retrieval process, we use the topic description in two “languages”: Firstly, the original topic description in English. Secondly, the topic description in biomedical “concept language” as detected by the concept recognizer described in section 3.3. The topic description in “concept language” is translated to English using the statistical translation dictionary. A method similar to Xu [8] is used.

Our document retrieval is based on generative unigram language models. The language models of the indexed documents (in our case valid spans from complete articles) are compared to the language model of the query, and are ranked according to the likelihood of generating this query model. Both document and query model, $P(t|D)$ and $P(t|Q)$ respectively, assign a probability to the event of drawing a term t from it.

Our query model is a mixture of terms actually occurring in the original English query $P(t|Q_e)$, and the terms which can be generated from a language model based on the conceptual representation of the query $P(t|Q_c)$. The mixture query model looks as follows:

$$P(t|Q) = \alpha P(t|Q_e) + (1 - \alpha)P(t|Q_c), \quad (1)$$

where α is a mixture parameter, in our experiments fixed to 0.7 based on previous experimental results.

The English query model $P(t|Q_e)$ is simply estimated based on maximum likelihood:

$$P_{ML}(t|Q_e) = \frac{tf(t, Q_e)}{|Q_e|}, \quad (2)$$

where $tf(t, Q)$ is the term frequency of the term t in the query Q and $|Q|$ is the query length.

The conceptual query model $P(t|Q_c)$ is based on which terms are used to denote biomedical concepts detected in the query:

$$\begin{aligned} P(t|Q_c) &= \sum_{c \in Q_c} P(c, t|Q_c) \\ &= \sum_{c \in Q_c} P(t|c, Q_c)P(c|Q_c) \end{aligned}$$

where $c \in Q_c$ are the concepts detected in the query and $P(c, t|Q_c)$ is the probability that the term t is used to denote concept c in the query context. We simplify our model by dropping this dependence on the query context:

$$P(t|Q_c) \approx \sum_{c \in Q_c} P(t|c)P(c|Q_c),$$

where $P(t|c)$ denotes the probability that a term t is used to represent concept c . We assume $P(c|Q_c)$ is uniformly distributed and estimate $P(t|c)$ using the dictionary:

$$P_{ML}(t|Q_c) = \sum_{c \in Q_c} \frac{1}{|Q_c|} P_{dict}(t|c).$$

The retrieval status value (RSV), the function used to score a (span-)document D for a query Q , is defined as follows:

$$RSV(Q, D) = \sum_{t \in Q} P(t|Q) \log\left(1 + \frac{\lambda P(t|D)}{(1 - \lambda)P(t|C)}\right)$$

Where $P_{ML}(t|D) = \frac{tf(t, D)}{|D|}$

$$P_{ML}(t|C) = \frac{\sum_{D \in C} tf(t, D)}{\sum_{D \in C} |D|}$$

$P(t|Q)$ is the probability of a term t to occur in the query language model. $P(t|D)$ is the probability of a term being sampled from the document D , which is based on the maximum likelihood estimate. $P(t|C)$ is the probability of a term being sampled from the background smoothing model C , which is based on maximum likelihood estimates on the collection. $tf(t, D)$ is the term frequency of a term in a document, i.e. the number of occurrences of the term in the document. $|D|$ denotes the document length, i.e. total number of tokens in the document. Finally, λ is the Jelinek-Mercer smoothing parameter which sets the relative influence of the background smoothing model. For more information see Hiemstra and Kraaij [1, 2].

4.2 Candidate entity selection and filtering

After retrieving a list of relevant spans, a number of possible answers is extracted. This is simply done by filtering the concepts with the desired entity type from the best scoring spans. Initial experiments with this approach showed that naively using these entities as candidate answers resulted in too general answers. Therefore, we sort the answers by the log likelihood that an entity in our list of relevant spans was not the result of drawing it from a random span in the collection. Initially, we use the top 50 results from our retrieval run to find candidate answers. If no candidate answers are found in this set, the next 50 results are considered repeatedly until at least a single matching concept is found, however no more than a 1000 hits are considered.

5 Runs and results

We submitted two automatic runs **EMCUT1** and **EMCUT2**. Run **EMCUT1** aimed at achieving a high aspect precision, by returning passages for different entities in a round-robin fashion. Run **EMCUT2** simply returns the best scoring passages from each document, which do contain the requested entity type. The results from the official runs are displayed in table 2.

After the workshop we carried out a number of additional experiments. Figure 3 shows the runs graphically, table 1 gives an overview of the results. The upper half of the table shows results in which we do not carry out candidate entity selection and filtering (named ‘-filter’). In the lower half of the table we do apply the approach described in section 4.2 (named ‘+filter’). As a baseline we use the untranslated query model based on maximum likelihood (named ‘MLIR’), as described in eq. 2. The CLIR system used the proposed mixture model based on both maximum likelihood and concept-translation (eq. 1). The following subsections discuss the results.

5.1 Candidate selection and filtering

One would expect that candidate selection and filtering would improve precision at cost of recall: if we detected that the passage contains the requested entity type, it’s more likely that it is relevant

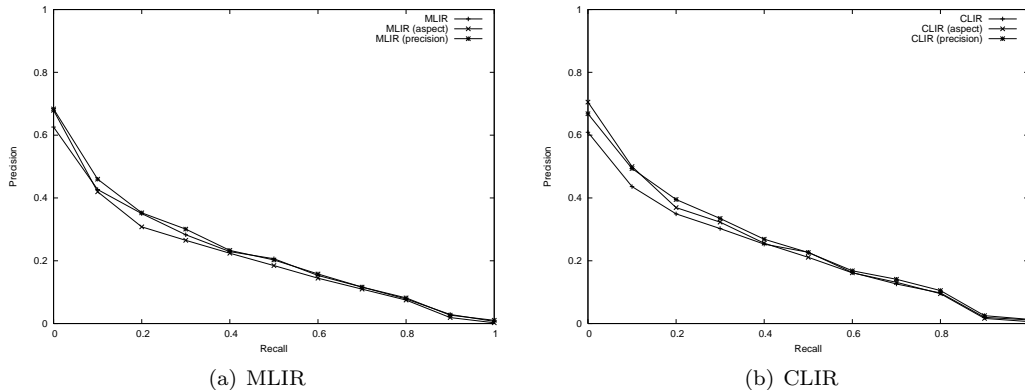


Figure 4: PR-curves of systems with (aspect/precision run) and without candidate selection and filtering.

to our query. However, if we miss a requested entity type during concept detection we might lose relevant answers as well. When we compare the upper half with the lower half of table 1, the precision based measures indeed confirm an increase in initial precision: precision at 5 and 10 retrieved relevant documents show a steady improvement over all runs. The average precision based measures show that especially the CLIR runs benefit from candidate selection and filtering. Also see the PR-curves for the baseline (figure 4(a)) and CLIR runs (fig. 4(b)).

5.2 Concept cross translation

The numbers in table 1 show that the CLIR approach in general outperforms our baseline. Despite the reasonable average percentual increase, most of the differences are not significant. Figure 5 shows the per topic difference in (document) average precision ⁷. For 13 topics, no difference is witnessed because the concept recognizer did not detect any concepts, leading to no change in the query model. When no candidate selection takes place (fig. 5(a)), more than half of the affected topics is actually harmed by the cross language approach. On the other hand, a few topics (especially topics 209 and 229) benefit strongly from the CLIR approach.

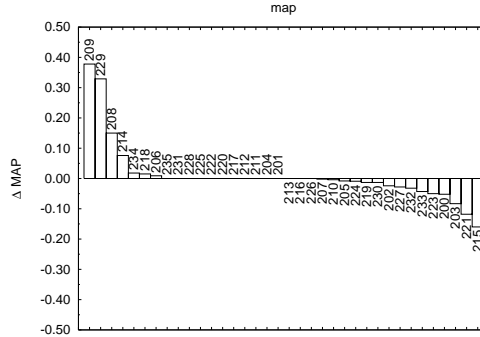
The runs which do candidate selection (fig. 5(b) and 5(c)) seem to benefit more from the CLIR approach. Half of the topics shows an increase in average precision, the other half a decrease. However, the increases are much more substantial than the decreases, leading to a higher mean average precision.

We have compared the submitted runs (CLIR+filter) to its baseline (MLIR+filter) and investigated the differences for each topic. The runs using the concept language model often perform better because the only concept recognised in the query is the most important part of the query, and extra weight is given to this part. For instance, in topic 207, only the concept ‘etidronate’ is found, so the search is biased towards this concept, leading to better results. However, sometimes the concept found is not the most important part of the query. For instance, in topic 224 the concept ‘lung cancer’ is detected, but ‘melanogenesis’ is not, even though this is an essential part of the query. Consequently, the performance on this topic is drastically reduced by incorporating the concept language model.

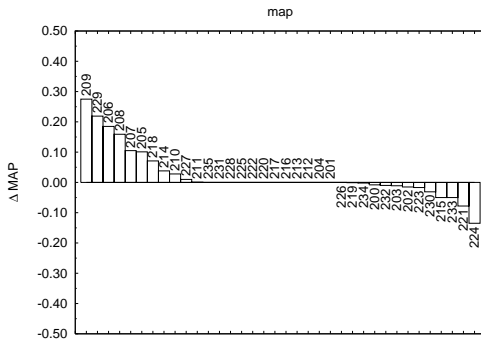
6 Conclusion

This year we approached TREC Genomics using a cross language IR (CLIR) techniques. Moreover, we filtered the retrieved spans on the presence of the concept types requested by the query.

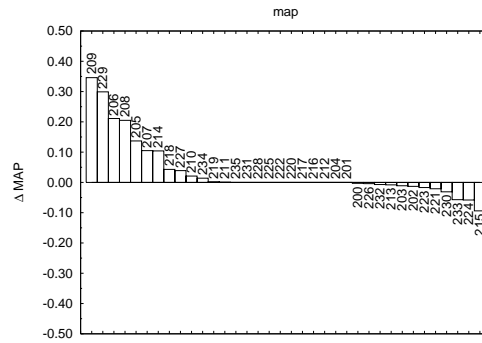
⁷Thanks to Edgar Meij for the script to generate them.



(a) No candidate selection



(b) Precision run



(c) Aspect run

Figure 5: Per topic difference in document average precision between CLIR and MLIR. Positive differences (CLIR > MLIR) are shown above the x-axis. The labels show the topic numbers.

Without candidate entity selection (-filter)							
		Document measures			Passage MAP		Aspect
		MAP	P@5	P@10	Pass.	Pass. 2	MAP
MLIR		0.210	0.394	0.358	0.045	0.046	0.121
CLIR		0.220 (4%)	0.389 (-1%)	0.375 (5%)	0.050 (11%)	0.053 (16%†)	0.113 (-7%‡)
With candidate entity selection (+filter)							
		Document measures			Passage MAP		Aspect
		MAP	P@5	P@10	Pass.	Pass. 2	MAP
MLIR (precision)		0.218	0.439	0.392	0.067	0.033	0.150
CLIR (precision)	EMCUT2	0.240 (10%†)	0.444 (1%)	0.425 (9%)	0.074 (10%)	0.038 (13%)	0.150 (-0%)
MLIR (aspect)		0.200	0.428	0.378	0.058	0.030	0.136
CLIR (aspect)	EMCUT1	0.234 (17%)	0.467 (9%)	0.419 (11%)	0.069 (18%)	0.037 (23%)	0.153 (13%)

Table 1: Comparison of different retrieval systems. Between parentheses the change with the system’s baseline (previous row); † and ‡ indicate a significant difference based on a Wilcoxon signed-rank test ($p < 0.05$ and $p < 0.005$ respectively).

topic	entity type	document		passage		passage2		aspect		ground truth		
		EMCUT1	EMCUT2	EMCUT1	EMCUT2	EMCUT1	EMCUT2	EMCUT1	EMCUT2	# asp.	# pass.	# docs
200	[PROTEINS]	0.291	0.289	0.048	0.044	0.006	0.005	0.111	0.097	233	320	193
201	[MUTATIONS]	0.036	0.021	0.006	0.006	0.002	0.002	0.003	0.001	6	37	12
202	[DRUGS]	0.000	0.001	0.000	0.000	0.000	0.000	0.000	0.000	21	53	43
203	[CELL OR TISSUE TYPES]	0.538	0.538	0.025	0.025	0.003	0.003	0.106	0.106	181	321	147
204	[CELL OR TISSUE TYPES]	0.294	0.307	0.093	0.092	0.020	0.019	0.313	0.356	24	164	74
205	[SIGNS OR SYMPTOMS]	0.269	0.243	0.030	0.020	0.014	0.009	0.093	0.112	13	93	65
206	[TOXICITIES]	0.375	0.375	0.048	0.048	0.019	0.019	0.090	0.090	17	38	19
207	[TOXICITIES]	0.143	0.143	0.004	0.004	0.000	0.000	0.075	0.075	7	15	12
208	[BIOLOGICAL SUBSTANCES]	0.288	0.274	0.116	0.044	0.086	0.026	0.083	0.035	11	22	16
209	[BIOLOGICAL SUBSTANCES]	0.446	0.419	0.404	0.442	0.248	0.265	0.504	0.398	9	78	11
210	[MOLECULAR FUNCTIONS]	0.193	0.192	0.047	0.052	0.025	0.028	0.102	0.081	18	71	57
211	[ANTIBODIES]	0.415	0.415	0.010	0.010	0.013	0.013	0.026	0.026	23	57	42
212	[GENES]	0.327	0.350	0.272	0.291	0.161	0.165	0.456	0.472	94	358	133
213	[GENES]	0.549	0.609	0.194	0.243	0.122	0.140	0.453	0.424	115	377	185
214	[GENES]	0.650	0.630	0.170	0.195	0.107	0.125	0.402	0.288	39	209	98
215	[PROTEINS]	0.254	0.283	0.072	0.070	0.046	0.044	0.095	0.102	56	137	73
216	[GENES]	0.060	0.053	0.019	0.012	0.007	0.004	0.020	0.015	10	42	34
217	[PROTEINS]	0.050	0.041	0.005	0.010	0.002	0.003	0.006	0.012	34	38	34
218	[GENES]	0.366	0.335	0.101	0.091	0.060	0.052	0.197	0.139	61	163	74
219	[DISEASES]	0.003	0.001	0.001	0.000	0.000	0.000	0.005	0.000	34	22	16
220	[PROTEINS]	0.224	0.276	0.043	0.046	0.066	0.068	0.514	0.519	4	16	6
221	[PATHWAYS]	0.570	0.523	0.108	0.084	0.033	0.028	0.185	0.164	80	183	87
222	[MOLECULAR FUNCTIONS]	0.083	0.069	0.051	0.054	0.018	0.018	0.068	0.078	63	57	42
223	[STRAINS]	0.146	0.146	0.013	0.013	0.007	0.007	0.094	0.094	11	18	8
224	[GENES]	0.001	0.001	0.000	0.000	0.000	0.000	0.000	0.000	1	3	3
225	[BIOLOGICAL SUBSTANCES]	0.071	0.333	0.055	0.167	0.023	0.074	0.067	0.333	1	1	1
226	[PROTEINS]	0.436	0.497	0.085	0.159	0.024	0.039	0.269	0.215	11	152	57
227	[GENES]	0.238	0.235	0.068	0.077	0.034	0.040	0.154	0.148	132	281	172
228	[GENES]	0.034	0.032	0.007	0.006	0.006	0.003	0.017	0.033	11	15	14
229	[SIGNS OR SYMPTOMS]	0.553	0.565	0.222	0.210	0.114	0.105	0.783	0.804	18	150	57
230	[PATHWAYS]	0.173	0.173	0.070	0.070	0.034	0.034	0.055	0.055	17	82	29
231	[TUMOR TYPES]	0.025	0.020	0.002	0.000	0.001	0.000	0.009	0.000	7	16	13
232	[DRUGS]	0.081	0.067	0.005	0.003	0.005	0.001	0.026	0.010	42	93	57
233	[GENES]	0.006	0.012	0.001	0.002	0.000	0.001	0.006	0.016	1	19	16
234	[GENES]	0.099	0.087	0.036	0.048	0.007	0.004	0.068	0.043	455	609	483
235	[GENES]	0.117	0.081	0.031	0.030	0.009	0.008	0.070	0.053	110	182	107
	average	0.233	0.240	0.069	0.074	0.037	0.038	0.153	0.150	54.72	124.78	69.17
	TREC median	0.187	0.187	0.049	0.049	0.028	0.028	0.108	0.108			

gray background: >= median

bold: highest score

Table 2: Performance scores for runs EMCUT1 and EMCUT2

On its own the CLIR approach gives varying results: some topics benefit from the reweighting of important query terms and the expansion with tokens related to the detected biomedical concepts. Other topics are actually harmed by the approach because the most important concepts were not detected. In many cases no concepts were detected because we limited our concept detection to concept types requested by the topics.

For the filtering we can say the same: on its own only the initial precision is improved by filtering on requested concept type.

Interestingly, the combination of CLIR and filtering does show a clear improvement over the baseline. Our hypothesis is that the filtering makes up for poor queries from the CLIR approach and benefits from improved CLIR queries.

Future work will be to further examine the cross language approach to biomedical IR. We expect better results when the initial concept recognition is more complete.

7 Acknowledgements

This work was part of the BioRange programme of the Netherlands Bioinformatics Centre (NBIC), which is supported by a BSIK grant through the Netherlands Genomics Initiative (NGI).

References

- [1] Djoerd Hiemstra and Wessel Kraaij. Twenty-one at TREC-7: Ad hoc and cross language track. In Ellen M. Voorhees and Donna K. Harman, editors, *The Seventh Text REtrieval Conference (TREC-7)*, volume 7. National Institute of Standards and Technology, NIST, 1999. NIST Special Publication 500-242.
- [2] Djoerd Hiemstra and Wessel Kraaij. A language modeling approach for TREC. In Ellen M. Voorhees and Donna Harman, editors, *TREC: Experiment and Evaluation in Information Retrieval*. MIT press, 2005.
- [3] A. McCray, S. Srinivasan, and A. Browne. Lexical methods for managing variation in biomedical terminologies. In *Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care*, 1994.
- [4] Andrei Mikheev. Periods, capitalized words, etc. *Comput. Linguist.*, 28(3):289–318, 2002.
- [5] M.J. Schuemie, R. Jelier, and J.A. Kors. Peregrine: Lightweight gene name normalization by dictionary lookup. In *Proceedings of the Biocreative 2 workshop*, Madrid, April 2007.
- [6] D. Trieschnigg, W. Kraaij, and M.J. Schuemie. Concept based document retrieval for genomics literature. In *Proceedings of TREC 15*, 2007.
- [7] R.B. Trieschnigg, W. Kraaij, and F.M.G. de Jong. The influence of basic tokenization on biomedical document retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 803–804, New York, NY, USA, 2007. ACM Press. extended abstract.
- [8] Jinxi Xu, Alexander Fraser, and Ralph M. Weischedel. TREC 2001 cross-lingual retrieval at BBN. In E.M. Voorhees and D.K. Harman, editors, *The Tenth Text REtrieval Conference*, 2001.