NovaSearch at TREC 2015 Clinical Decision Support Track

André Mourão, Flávio Martins and João Magalhães

NOVA LINCS
Departamento de Informática
Faculdade de Ciências e Tecnologia
Universidade NOVA de Lisboa, Portugal
a.mourao@campus.fct.unl.pt, flaviomartins@acm.org,
jm.magalhaes@fct.unl.pt

Abstract. This paper describes the participation of the NovaSearch group at TREC Clinical Decision Support 2015. For this year's task, we extended our rank fusion experiments from last year's edition using a supervised Learning to Fuse technique.
Learning to Fuse is a technique that incrementally combines multiple runs that use different retrieval algorithms, relevance feedback schemes and query expansion data sources to create a better final rank.
We also experimented with query expansion using MeSH, SNOMed CT and Shingles thesaurus and tested a Journal based filtering technique to remove results from irrelevant journals. For Task B runs, we added the diagnosis information to the queries.

1 Introduction

TREC Clinical Decision Support Track goal is the "retrieval of biomedical articles relevant for answering generic clinical questions about medical records."[1]

This is the second edition of the track of the track and shares the same dataset and tasks with the 2014 edition, with the addition of a Task B that includes the diagnosis of the patient for the "treatment" and "test" queries. Our participation on this track follows our work on last year's track [3].

Section 2 details our usage of general and domain specific IR techniques. Section 3 describes our Learning to Fuse framework. Section 4 describes our journal filtering algorithm. Section 5 contains the results and discussion.

2 Medical text indexing and retrieval

Our indexing and retrieval system is based on Lucene, with support for additional retrieval functions, query expansion and pseudo-relevance feedback. [2] contains a detailed explanation of the full system; on this paper, we'll describe our new experiments for TREC CDS 2015.

[1] http://www.trec-cds.org/

2.1 Query expansion: MeSH, SNOMed and Shingles

Our baseline query expansion method is based on a SKOS formatted version of MeSH using Lucene-SKOS [1]. This year, we tested two additional methods:

- SNOMed CT 2015: we parsed the terms and relations of the Web Ontology Language (OWL) version of SNOMed CT International Release RF2 from January 2015, to make it work with Lucene-SKOS;
- Shingles: we created a n-gram (n=8) word based index that enabled expansion of query terms with neighbor terms from documents in the collection.

When expanding with MeSH and SNOMed, we appended all synonyms, alternative and preferential labels for all query terms with a weight of 0.7 (original query terms are weighed 1). When expanding with Shingles, we added the terms that appeared on more than one of the expansions of the individual query terms. Figure 1 shows some sample expansions for MeSH, SNOMed CT and Shingles.

58-year-old **woman** [*“women”*] with **hypertension** [*"blood pressure high"* | "high blood pressure disorder" | "hypertensive vascular degeneration" | "high blood pressure disorder"] and obesity presents with **exercise**-related [*"aerobic exercise"* | *"exercise physical"* | *"isometric exercise"* | "lesion"] episodic chest **pain** ["pain observations" | painful | "part hurts"] radiating to the **back** ["back structure excluding neck" | "entire back surface region"].

Fig. 1. Query expansion example. Bold terms represent query terms that were expanded; Blue and italic terms represent MeSH expansions; Green and underlined terms represent Shingle expansions; Red terms represent SNOMed CT expansions

Custom top-precision reranking When analyzing last year results, we discovered that query expansion lead to a slight decrease in the relevance of the top 2 documents, when compared to the original non-expanded run. Based on this observation, and for all expanded runs, we kept the top 2 results from the non-expanded run and followed by the results from the expanded run.

2.2 Rank fusion: Learning to Fuse

For last year's track, we selected a set of runs that applied different retrieval functions and combined them using a fusion algorithm. This year, we extended our unsupervised rank fusion method by adding a supervised step that selects what runs to combine, based on their performance on a relevant dataset. These runs can be based on different retrieval functions, expansion methods, etc. It

Algorithm 1 Learning to fuse (L2F) algorithm

let a rank be a list of documents (q_1,$id_{1,1}$,1),...,(q_1,$id_{1,n}$, n),...,(q_m,$id_{m,n}$,nm) where id is a document id, m is the number of queries and n the number of results per query,

Input: R: a sorted list of ranks returned by multiple retrieval systems. The list is sorted in descending order according to a evaluation metric met,

Input: met: evaluation metric that takes a rank and a set of relevance judgments and returns the value of that evaluation metric for that rank,

Input: $comb$: fusion algorithm that takes two ranks R_1 and R_2 and combines them into R_f, sorted according to rank and frequency across ranks (e.g. RRF, ISR),

Input: $tries$: natural number representing the number of iterations to run while the results are not improving,

Output: R_{best}: final combined rank.

```
$R_{current} \leftarrow R_1$
$R_{best} \leftarrow R_1$
$i \leftarrow 2$
$currentTries \leftarrow tries$
while $currentTries \geq 0$ and $i < len(R)$ do
    $R_{current} \leftarrow comb(R_{current}, R_i)$
    if $met(R_{current}) > met(R_{best})$ then
        $R_{best} \leftarrow R_{current}$
        $currentTries \leftarrow tries$
    else
        $currentTries \leftarrow currentTries - 1$
$i \leftarrow i + 1$
return $R_{best}$
```

works by sorting runs by performance on an evaluation metric and fuse additional runs if they keep improving the final result. Our approach is fully detailed in Algorithm 1.

The *tries* parameter was introduced to avoid local minima, by allowing the result to get slightly worse at one iteration, if it leads to a better result after adding additional ranks. It was set to 3. The selected evaluation metric for ranking was infNDCG and the fusion algorithm was RRF. The models were trained with last year queries and relevance judgments.

The set of possible ranks was a combination of all of the following:

- *Retrieval function:* **TFIDF**, **BM25+**, **BM25L** or Language Model (**LM**)
- *Search Fields:* full text (**f**) or full text + title + abstract (**fat**)
- *Query expansion:* no expansion (**NoEXP**), MeSH (**MSH**), SNOMed CT (**SNO**), Shingles (**SHI**)
- *Pseudo-Relevance Feedback:* true (**PRF**) or false (**NoPRF**)

The runs selected by the algorithm are available in Table 1. Best 2014 is the set of runs that led to our best result on TREC CDS 14. L22F A and B are the runs selected by the Learning to Fuse algorithm for Task A and B, respectively. BM25+, BM25L and TFIDF retrieval functions, MeSH expansion and PRF seem to be present in the bulk of the selected runs, although both selections also improved when Language Model based runs and runs without Query Expansion were added to the query.

Table 1. Runs selected by the Learning to Fuse algorithm (L2F A, L2F B) and best last years run (Best 2014)

Best 2014	L2F A	L2F B
BM25+, fat, MSH, PRF	BM25+, f, NoEXP, PRF	BM25+, f, NoEXP, NoPRF
BM25L, fat, MSH, PRF	BM25+, fat, MSH, PRF	BM25+, f, SNO, PRF
TFIDF, fat, MSH, PRF	BM25+, fat, NoEXP, PRF	BM25+, fat, SNO, NoPRF
LM, fat, MSH, PRF	BM25L, f, MSH, PRF	BM25L, fat, MSH, PRF
	BM25L, fat, MSH, PRF	TFIDF, f, MSH, PRF
	TFIDF, f, MSH, PRF	TFIDF, f, NoEXP, PRF
	TFIDF, f, NoEXP, PRF	TFIDF, f, SHI, PRF
	TFIDF, fat, MSH, PRF	TFIDF, f, SNO, PRF
	TFIDF, fat, SNO, PRF	TFIDF, fat, MSH, PRF
	LM , f, SNO, PRF	TFIDF, fat, SNO, PRF
		LM, f, SNO, PRF

2.3 Journal-based filtering

The PubMed Central collection contains an huge amount of articles and journals that are not relevant for case-based clinical support systems. These articles and journals may be related to non-human subjects, gene sequencing, amongst others.

We created a filter that removes from the ranks articles from certain journals deemed irrelevant, for each query type. The filters are based on the Vowpal Wabbit classifier and trained using journal articles Bag-of-Words (BoW), after stemming and stop word removal. For each query type, we selected a sample of relevant and non-relevant documents (evaluated on last year's tracks), aggregated document BoW into journal BoW and trained a model using the relevance judgments as the ground truth.

3 Results and discussion

Table 2 contains a summary of the techniques used in each run. All our runs are based on the case summaries.

Table 3 contains the results of the techniques used in each run. The Learning to Fuse approach on Task A lead to a slight increase in performance when compared to the 2014 selection and to a single, non-fused run.

Table 2. NovaSearch run summary. **Fusion**: selected fusion algorithm; **QE**: Query Expansion; **PRF**: Pseudo Relevance Feedback; **JF**: Journal Filtering; **Diag**: Diagnosis appended to the query. × marks that all the runs applied the technique. Retrieval functions in italics are a set of runs. ∗ marks that one or more of the runs that were combined applied the technique; more information on Table 1.

	Run id	Retrieval func.	Fusion	QE	PRF	JF	Diag	Notes
Task A	1	*Best 2014*	RRF	×	×			Fusion Baseline
	2	BM25L	—	×	×			Custom Re-Ranking
	3	*L2F A*	RRF	∗	∗			Learning to fuse
Task B	4	BM25L	—	×	×		×	Custom Re-Ranking
	5	*L2F B*	RRF	∗	∗		×	Learning to fuse
	6	*L2F B*	RRF	∗	∗	×	×	L2F + filtering

For Task B, the binary relevance on journal filtering algorithm was too aggressive and biased towards last years data. The results may improve when testing the algorithm with this year's relevance judgments.

This document will be updated with additional experiments and discussion when the relevance judgments are released.

Table 3. TREC CDS 2015 results for NovaSearch runs. Bold values represent our best results.

	Run id	infAP	infNDCG	R-prec	P@10
Task A	1	0.0490	0.2242	0.1927	**0.3567**
	2	0.0475	0.2208	0.1801	0.3467
	3	**0.0509**	**0.2295**	**0.1964**	**0.3567**
Task B	4	0.0665	0.2964	0.2282	0.4567
	5	**0.0783**	**0.3207**	**0.2637**	**0.4933**
	6	0.0675	0.2992	0.2179	0.4900

References

1. Haslhofer, B., Martins, F., Magalhães, J.: Using skos vocabularies for improving web search. In: Proceedings of the 22nd international conference on World Wide Web companion. pp. 1253–1258. International World Wide Web Conferences Steering Committee (2013)
2. Mourão, A., Martins, F., Magalhães, J.: Multimodal medical information retrieval with unsupervised rank fusion. Computerized Medical Imaging and Graphics 39, 35 – 45 (2015), `http://www.sciencedirect.com/science/article/pii/S0895611114000664`, medical visual information analysis and retrieval
3. Mourão, A., Martins, F., Magalhães, J.: Novasearch at trec 2014 clinical decision support track. In: Proceedings of the 2014 Text Retrieval Conference. pp. 1–4 (2015)