Improved Feature selection and redundancy computing -- THUIR at TREC 2004 Novelty track*

Liyun Ru, Le Zhao, Min Zhang, Shaoping Ma
State Key Lab of Intelligent Tech. & Sys., CST Dept, Tsinghua University, Beijing 100084, China
Lyru98@mails.tsinghua.edu.cn, zhaole@tsinghua.org.cn, {z-m, msp}@tsinghua.edu.cn

This is the third years that Tsinghua University Information Retrieval Group (THUIR) participates in Novelty task of TREC. Our research on this year's novelty track mainly focused on four aspects: (1) text feature selection and reduction; (2) improved sentence classification in finding relevant information; (3)efficient sentence redundancy computing; (4) effective result filtering. All experiments have been performed on TMiner IR system, developed by THU IR group last year.

1. Text feature selection and reduction

The work of text feature selection and reduction is used as the basis of both relevant and new steps. In novelty/passage retrieval, data sparseness problem is dominant. In a sentence, different terms act as different roles. What terms take most important information for a given user query? What kinds of feature are most useful to identify the core information? This is what we try to find out in our research on text feature selection and reduction. Three kinds of approaches have been carried out:

(1)Using Named Entity as significant features;

(2)Using POS-tagging information for feature selection;

(3)Using PCA transform for feature reduction.

The former two approaches take NLP information into account. In our experiments, both NE- and PCA- based approaches improve system performances, while POS-tagging information does no help. In the following, we'll give some more detailed descriptions of NE-based feature selection and PCA-based feature reduction.

1.1 Using Named Entity as significant features

The basic of NE-based approach is to recognize which phrase is Named Entity and what type of Named Entity it is. In our experiments, 34 types of Named Entities have been recognized[1], such as organization, person, location, date, country, occupation, animal, area, etc.

After the annotation of Named Entity, they are used in the following four ways:

Approaches	*NE*1	*NE*2	*NE*3	*NE*4
Reserve original words or not?	Discard	Discard	Reserve	Reserve
Use same or different tags for different NE types?	Different	Same	Different	Same

The experiments show that all NE based approaches we proposed will improve system performances and *NE*1 achieves best results.

1.2 Using PCA transform for feature reduction

PCA is a statistical tool for data analysis. It decorrelates second order moments corresponding to low frequency property, and identifies directions of principal variations in the

* Supported by by Chinese National Key Foundation Research & Development Plan (973) (Grant No.2004CB318108) and Chinese Natural Science Foundation (Grants No. 60223004, 60321002, 60303005).

[1] The NE recognition toolkit is provided by IBM China Research Center.

data. There are two advantages brought about by PCA, i.e. dimension reduction and noise reduction.

PCA is used in task2, task3 and task4 to solve data sparseness problem and reduce noise by finding the most dominant features. In task3 relevant step, query and all corresponding sentences of each topic are used to perform PCA transformation, and then the mean PCA-feature of query and relevant sentences of first 5 documents is taken as new feature of query.

Suppose PCA-feature of query is $Q = [q_1, \ldots, q_m]$, PCA-feature of each relevant sentence of first 5 documents is $S_k = [S_{k1}, \ldots, S_{km}]$, $k = 1, \ldots, K$, where K is the number of relevant sentences of first 5 documents. Then the new PCA-feature of query can be represented as the mean of query and all given relevant sentences:

$$Q' = [q'_1, \ldots, q'_m], \text{ where } q'_i = \frac{1}{K+1}(q_i + \sum_{k=1}^{K} s_{ki})$$

Then cosine similarities between all sentences and new query can be calculated and ranked.

For new step of task2, task3 and task4, only relevant sentences are used to PCA transformation. And overlap-based redundancy measurement is performed.

The experimental results show that this PCA-based feature subspace approach does helpful on finding relevance and sentence redundancy computing, which can be seen in Table 1.

Table 1 Effects of PCA-based feature subspace approaches

Task	**Approach description**	**P**	**R**	**F**	**tag**
2new	Overlap threshold = 0.7 + tightness	0.46	0.96	0.606	THUIRnv0421
	PCA, cosine similarity, threshold = 0.8	**0.46**	**0.96**	**0.605**	**THUIRnv0423**
3rel	Long query	0.31	0.82	0.419	THUIRnv0411
	PCA, Filter when cosine-sim<0.15*top_sim	**0.34**	**0.76**	**0.433**	**THUIRnv0433**
4new	Overlap, threshold = 0.8	0.42	0.96	0.568	THUIRnv0443
	PCA, cosine similarity, threshold = 0.7	**0.43**	**0.92**	**0.572**	**THUIRnv0445**

2. Improved sentence classification

In TREC2003, we proposed a SVM-based sentence classification approach, which helped finding relevant information in task3 [1]. In this year, some improvements have been made.

Positive and negative examples were first selected from relevant sentences provided in first 5 documents. However, results may be not encouraging, if few positive examples are given. To solve this problem, top sentences returned by initial retrieval were added into positive examples set. Weights of positive and negative examples have been balanced, using inverse ratio of the numbers of the two kinds of examples. Then a SVM classifier is learned to find relevant information. Better results have been got in this way. We used a SVM package (version 2.4, by Chih-Wei Hsu, etc. [2]) to create the classifier.

It shows that this improved sentence classification does helpful on finding relevance, which can be seen in the following table 2.

Table 2 Effects of improved sentence classification using SVM

Approach description	P	R	F	tag
Long query (Baseline)	0.31	0.82	0.419	THUIRnv0411
SVM : radial basic function	0.24	0.59	0.289	un-official run
Improved SVM: radial basic function	**0.40**	**0.64**	**0.438**	**THUIRnv0434**

3. Efficient sentence redundancy computing

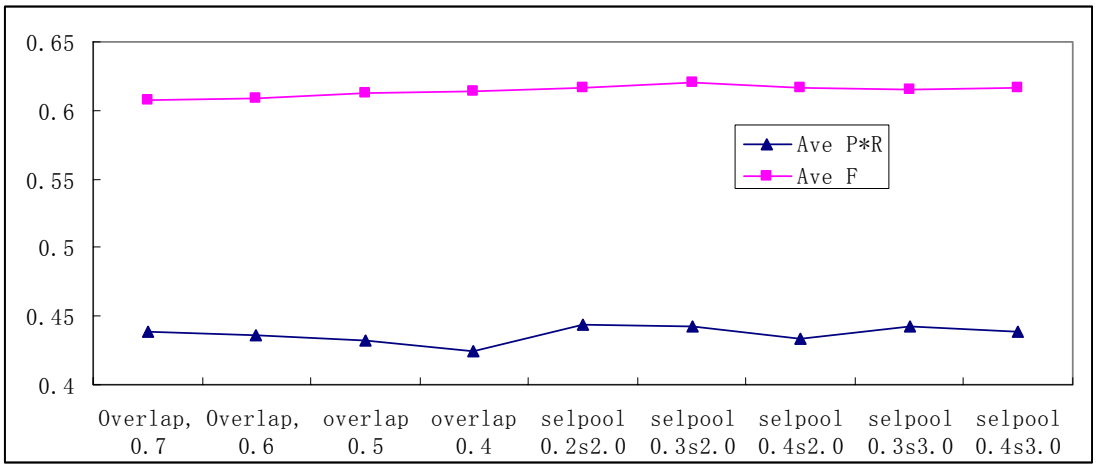
On sentence redundancy computing, both sentence-sentence overlapping and pool-sentence overlapping have been studied in TREC2002 and TREC2003. In this year, we proposed an improved overlap strategy based on selected pool with tightness factor.

3.1 Redundancy elimination based on selected pool

A selected pool is not a pool that contains all the previously appearing sentences. The previous sentences can be selected to the pool only if they have an overlap of more than some certain threshold to the current sentence. Once the pool is constructed, a comparison of overlap between the current sentence and the pool will be done.

This selected pool method is the combination of sentence-sentence comparison and pool-sentence overlapping approaches. It overcomes certain drawbacks of the sentence-sentence overlap comparison which certainly excludes the case in which multiple sentences appearing before the current sentence together made the current sentence redundant; and drawbacks of the simple pool-sentence comparison, in which the pool constructed using all the previously appearing sentences will certainly contain too much noise introduced by the sentences not related to the current sentence. The effects of using selected pool are shown in Figure 1.

Figure 1. Effects of using overlap-based and selected-pool-based redundancy measurement



overlap, X: sentence-sentence overlap approach, X is the redundancy threshold.

selpool XsY: selected pool approach, X is the redundant threshold, and Y is the selection threshold for a sentence to be added in the pool.

3.2 A drawback of the overlap based novelty judgment and a remedy – tightness restriction of overlap based methods.

In overlap-based methods, such as sentence-sentence overlap novelty detection and selected pool overlap, only the number of overlapping terms is taken into account, not concerning term position information. In experiments, however, many sentences with an overlap of nearly 1 are real novel ones. Why? Because a previous sentence that has many terms overlapped a latter short one, with the overlapping terms scattering separately in the previous sentence. When this occasion occurs, the latter sentence (the overlapped one) is usually not redundant as it appears.

Therefore we consider the window of overlapped terms in the previous sentence (referred as w_0) and that in the latter one (referred as w_i). If $w_0 > T*w_i$, the previous sentence is not potent. T is called tightness factor. For selected pool method, tightness can also be used in the process of

adding sentences to the pool.

We call this method a tightness restriction on overlapping (referred as tightness in the following). This tightness method is proved helpful in both novelty 2003 and novelty 2004 top 5 documents . The recall is improved with a comparatively small decrease in precision, and a steady increase in F-measure is obtained.

Table 3 Effects of using tightness restriction of overlap-based method

Novelty 2004 task2	**#ret**	**Ave P**	**Ave R**	**Ave P*R**	**Ave F**	**Difference:**
overlap 0.7	6965	0.462	0.950	0.439	0.608	68 in 253 are novel[2]
overlap 0.7 tightness	7218	0.456	0.965	0.441	0.606	
2004 task4, in top5 docs	**#ret**	**Ave P**	**Ave R**	**Ave P*R**	**Ave F**	
overlap 0.7	974	0.694	0.964	0.668	0.786	12 in 24 are novel
overlap 0.7 tightness	998	0.686	0.981	0.675	0.789	
selpool 0.3s5.0	965	0.694	0.959	0.665	0.784	13 in 25 are novel
selpool 0.3s5.0 tightness	990	0.687	0.977	0.672	0.788	
Novelty 2003 task2	**#ret**	**Ave P**	**Ave R**	**Ave P*R**	**Ave F**	
overlap 0.7	13303	0.719	0.972	0.698	0.815	108 in 218 are novel
overlap 0.7 tightness	13521	0.716	0.979	0.700	0.816	

4. Document and sentence similarities fusion

The main difference between this year's and last year's novelty task is that each topic of this year will include zero or more irrelevant documents in addition to 25 relevant documents. This year's task more likely happens in real world information retrieval applications. Therefore the role of document and sentence similarities is one of the interesting points in our study.

Three observations of document and sentence similarities have been carried out.

(1) Sentence-based filtering: similarities of original documents are ignored. Each sentence is taken as individual document, and those with small similarity scores or at last n percent are filtered out.

(2) Document-based filtering: Sentences are taken as parts of one document. After initial retrieval on sentence level, top sentences are used to generate a relevant documents list, and then sentences not belonging to relevant documents are discarded.

(3) Fusion of sentence and document similarity: Searching using sentences and documents respectively, then getting a final score for each sentence by fusing sentence similarity and the corresponding document similarity, and performing result filtering.

In our experiments, sentence filtering and the fusion of sentence and document are helpful, while document-based filtering does not, which can be seen in the following table 4.

Table 4 Effects of result filtering (novelty 2004 task1)

Approach description	P	R	F	tag
NE + Long query +LCE (MI, win=10) (Baseline)	0.26	0.95	0. 381	un-official run
Sentence Filtering	**0.31**	**0.81**	**0.409**	**THUIRnv0412**
Fusion of sentence and document filtering	**0.29**	**0.84**	**0.404**	**THUIRnv0413**
Document filtering	0.27	0.82	0.381	THUIRnv0415

[2] Tightness restriction will return more sentences than the overlap method it is based on. This difference is the number of novel sentences in all the sentences tightness method returned more than the basis of overlap.

5. Submitted official runs

Task	Approach description	P	R	F	tag
1rel[1]	Long query	0.31	0.82	0.419	THUIRnv0411
	Filter results when sen-sim < 0.3 * top_sim	0.31	0.81	0.409	THUIRnv0412
	Filter results when DSF-sim < 0.4 * top_sim	0.29	0.84	0.404	THUIRnv0413
	Filter results of last Dynamic Percent	0.28	0.81	0.392	THUIRnv0414
	Filter results of irrelevant documents	0.27	0.82	0.381	THUIRnv0415
1new[2]	Selected pool, 0.3s5.0	0.15	0.74	0.228	THUIRnv0411
	Selected pool, 0.3s5.0	0.14	0.73	0.220	THUIRnv0412
	Selected pool, 0.3s5.0	0.14	0.75	0.215	THUIRnv0413
	Selected pool, 0.3s5.0	0.13	0.72	0.209	THUIRnv0414
	Selected pool, 0.3s5.0	0.13	0.75	0.209	THUIRnv0415
2new	Overlap threshold=0.7 + tightness	0.46	0.96	0.606	THUIRnv0421
	Selected pool + tightness	0.46	0.96	0.606	THUIRnv0422
	PCA, cosine sim, threshold = 0.8	0.46	0.96	0.605	THUIRnv0423
	Selected pool	0.46	0.95	0.608	THUIRnv0424
	POS tag :nv, Selected pool + tightness	0.45	0.93	0.589	THUIRnv0425
3rel	NE + long query +filter when DSF-sim <0.2*top_sim	0.35	0.75	0.434	THUIRnv0431
	NE+PCA, Filter when cosine-sim<0.15*top_sim	0.34	0.77	0.431	THUIRnv0432
	PCA, Filter when cosine-sim<0.15*top_sim	0.34	0.76	0.433	THUIRnv0433
	SVM classification	0.40	0.64	0.438	THUIRnv0434
	Long query, filter when sen-sim<0.1*top_sim	0.36	0.67	0.435	THUIRnv0435
3new[2]	Selected pool + tightness	0.14	0.68	0.219	THUIRnv0431
	Selected pool + tightness	0.14	0.71	0.217	THUIRnv0432
	Selected pool + tightness	0.14	0.69	0.218	THUIRnv0433
	Selected pool + tightness	0.17	0.59	0.231	THUIRnv0434
	Selected pool + tightness	0.15	0.61	0.226	THUIRnv0435
4new[3]	Selected pool + tightness	0.42	0.97	0.567	THUIRnv0441
	Overlap + tightness	0.42	0.97	0.566	THUIRnv0442
	Overlap	0.42	0.96	0.568	THUIRnv0443
	PCA1, cosine sim, threshold = 0.7	0.43	0.91	0.569	THUIRnv0444
	PCA2, cosine sim, threshold = 0.7	0.43	0.92	0.572	THUIRnv0445

[1] Baseline of THUIRnv0412, THUIRnv0413, THUIRnv0414 is NE+Long query+LCE (MI,win=10), P=0.26, R=0.95, F=0.381

[2] In task1 and task 3, we aim to compare different approaches of finding relevant information; hence the same method has been used for different runs in new step.

[3] PCA1: data set is the whole relevant sentence;

PCA2: data set =new sentences of first 5 documents + all relevant sentences of remainder documents.

Reference

[1] M. Zhang, etc, THUIR at TREC 2003: Novelty, Robust and Web, in proceedings of TREC2003, pp556~567

[2] Chih-Wei Hsu, etc., SVM package, http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html