LSIS/LIA at TREC 2012 Knowledge Base Acceleration

Ludovic Bonnefoy
LIA - University of Avignon
ludovic.bonnefoy@etd.univ-avignon.fr

Vincent Bouvier **Patrice Bellot**
LSIS - Aix-Marseille University
firstname.name@lsis.org

Abstract

This paper describes our joint participation in the TREC 2012 KBA task. The system is broken down as follows : first name variations of the entity topics are searched then documents containing at least one of them are retrieved. Finally documents go through two classifiers to categorize them as garbage, neutrals, relevant or centrals. This system got good results (3rd of 11) however first analyses tends to show that ranking is just a little bit better than random.

1 Introduction

TREC 2012 has seen a new track named *Knowledge Base Acceleration* (KBA) started. This new task requires to focus on new challenges such as finding good ways of managing large corpora along a timeline.

KBA is about retrieving information as well as assessing the importance of them in order to eventually feed databases with new correct statements, or even broadcast news in real time. To do so, a stream has to be continuously monitored along the time to detect changes (i.e., new upcoming things about a topic). This stream is actually a set of documents that could be fed of blogs, microblogs, news websites and so on. Hence, the corpus has three categories of documents:

News: come from public news websites;

Socials: come from blogs, forums;

Links: come from bitly database;

Timestamp as well as metadata are provided depending on the category of the document. As a first task this year, the purpose is to find documents about known entities and classify those documents into four different classes :

Garbage: not relevant, e.g. spam.

Neutral: Not relevant, i.e. no information could be deduced about the entity, e.g., entity name used in product name, or only pertains to community of target such that no information could be learned about entity.

Relevant: Relates indirectly, e.g., tangential with substantive implications, or topics or events of likely impact on entity.

Central: Relates directly to target such that you would cite it in the wikipedia article for this entity, e.g. entity is a central figure in topics/events.

Next section we detail preprocessing steps we did concerning the corpus and the topics. Second part presents the core of our method which relies on two classification steps and high-level features. Finally we present and discuss results and bring some perspectives.

2 Preprocessings

2.1 Corpus

The first challenge is how to deal with such a tremendous corpus and how to do it efficiently. We decided to index the corpus offline and, when running our system, to retrieve candidate documents by mean of a search engine.

2.1.1 Indexation

We first had to deal with how to index the corpus with respect to the KBA rules to treat the corpus as a stream. The corpus is already divided into folders (one folder per hour) so we naturally indexed each single folder separately.

To index the corpus, we used the state-of-the-art information retrieval platform Terrier[1]. Within a document it is possible to access either the *RAW* version of the data (e.g., body as HTML document) or to access the *CLEANSED* version (e.g., body as plain text). We decided not to deal with document structures so we only used the cleansed version for every data. Moreover only cleansed documents were evaluated by assessors.

For the indexation process we indexed documents' titles as well as their bodies.

2.1.2 Retrieval

We adopted a recall oriented approach for the document retrieval step : we wanted to retrieve all documents containing at least one mention of a given topic entity. The only restriction was that documents must contain the named entity without word permutations, missing words, etc. (to avoid to get documents about "Barak Obama" when looking for 'Aharon Barak").

2.2 The topics Pre-Processing / Pre-Treatments

A set of topics has been given to KBAers which contains persons or organizations important enough to have a Wikipedia page. The evaluation allows the participants to use Wikipedia dump done on the 1^{rst} of January 2012 in order to extract information for either training or any other usage. Since the given topics are just about a name (being the Wikipedia page title), we decided to try to find variants for each of the topics, in order to increase the amount of documents found.

	Count	KBA 2012	LSIS 2012
total LSIS	44,351		
total KBA	52,244		
inter.	23,245	44.49%	52.41%
comp.	50,105	55.51%	47.59%

***Table 1:** KBA and LSIS result sets intersection and complements*

In order to find variants we parsed the whole Wikipedia corpus to find every links that pointed to the topic document from another document. We also consider the bold text contains in the first paragraph of a Wikipedia's article. Variants are weighted according to :

$$w(v_i) = \begin{cases} 1 \text{ if } v_i \text{ is the topic entity} \\ \frac{tf(v_i)}{\sum_{v_j \in V} tf(v_j)} \text{ otherwise} \end{cases}$$

Example 2.1 - Sample of found variants for Boris Berzovsky the businessman and the pianist :

```
---------------------------------------
Boris_Berezovsky_(businessman)
---------------------------------------
boris berezovsky               1,000000
boris abramovich berezovsky    1,000000

---------------------------------------
Boris_Berezovsky_(pianist)
---------------------------------------
boris berezovsky               1,000000
boris vadimovich berezovsky    1,000000
```

3 Classification

The output of the document retrieval step is a set of all documents containing at least one mention of one the entity's variants. According to table 1, $\approx 33\%$ of documents containing an entity mention are either garbage or neutral. In order to determine what

[1] http://terrier.org/

could make a document being either garbage, neutral, relevant or central, we decided to rely on a supervised approach. Such approach is made possible by the train corpus (from October to December 2011) and associated relevance judgments provided by organizers.

	garbage	neutral	relevant	central
contains mention	7991	3862	13971	7806
zero mention	15367	163	61	0

Table 2: *KBA corpus statistics*

3.1 Features

A lot of approaches already exists in order to filter documents in a stream according to a topic. However, most of them are topic specific ie each topic need is own classifier and so need an associated training set. This is a huge drawback because they are not easily adaptable to new topics.

In this work we were looking for features which capture topic independent phenomena which denote relevancy or centrallity (in the way it is defined for this task).

Intuitively, we came with three groups of features : time related features, document content related features and related entities features.

3.1.1 Time related features

Since we are monitoring topic activities on a chronological stream of documents, we thought interesting to look at the peaks that could arise suddenly after querying a certain amount of indexes. A scale has been determined arbitrarily that seems reasonable enough in term of time monitoring scale:

Daily report : Results are aggregated over 24hours so that there is enough time for documents to appear when a topic is making the *"buzz"*. The number of document found within a 24 hour scale is used as a feature for the algorithm.

6 previous days statistics : Statistics are gathered on a sliding window over a week (i.e., a queue of 7 days), where daily report represent the current day and other statistics are computed from the 6 previous days such as:

Number of mentions in previous days title: counts how many mentions there are in titles.

Number of mentions in previous documents: counts how many mentions there are in documents' bodies.

The average number of documents: The average number of documents where the whole week is considered.

σ of the amount of documents: The standard deviation computed from the number of documents found over a week. This features is helpful to detect peaks since the standard deviation will change brutally when the distribution of documents changes suddenly.

3.1.2 Document content related features

Even though those features are really interesting to observe it is not enough to assess whether a document is relevant or not, since it represents a set of documents and not a specific document. Moreover it may have some noise in the peak itself. So we decided to add features concerning the topic's mentions for each single document:

Mentions distribution: How the different variants of the mention are distributed along the document. We computed then the amount of mentions from 0% to 100% using a 10% step. In addition we add a specific feature for the mentions in title.

Tf-Idf: The document score given by a TF IDF computed by Terrier with variants as queries;

Cosine Similarity: A cosine similarity is computed between 1gram and 2gram words distribution of the document and the 1gram and 2gram words distribution created from the Wikipedia topic's page.

3.1.3 Related entities features

When dealing with monitoring particular topics, it's interesting to keep track of different relations the topic has with a particular entity (e.g., a person, an event,...). Related entities have been extracted from Wikipedia during the dump parsing using two different information:

- The page is linked by the topic's page;
- All named entities found in the topic's page

Relations : Count the amount of relations in document's title and body for each related entity;

3.2 Classifier

Two classifiers were used in cascade to determine if a document is either garbage/neutral, relevant or central. Documents to which the relevant/central class was assigned by the first classifier were then classify by the second one.

Classifier 1: Class Garbage/Neutral and Class Relevant/Central

Classifier 2: Class Relevant and Class Central

4 Runs

Our team submitted 6 runs falling in one of the two different scoring methods used. Each classifier returns a score between 0 and 1 for a given document. A score of 0 or 1 denotes a total confidence in the class associated to a document by a classifier. Concerning the first classifier, a score below 0.5 signifies that it has assigned the garbage/neutral class to the document, a score higher corresponds to the relevant/central class. The second classifier gives a score between near 0 and near 1 where near 0 is relevant and near 1 is central.

"Yes" runs : for the 3 runs falling in this category we returned only documents classified as Relevant/Central by the first classifier. Then documents are scored according to :

$$score(d_i) = s(d_i, c_1) \times s(d_i, c_2)$$

where the score of the document d_i is given by a product of the scores given respectively from classifiers c_1 and c_2 having therefore $s(d_1, c_1) > 0.5$.

"All" runs: for these 3 runs all documents found are returned. Documents are scored according to :

$$score(d_i) = \begin{cases} s(d_i, c_1) \text{ if } s(d_i, c_1) < 0.5 \\ 0.5 + \frac{s(d_i,c_1) \times s(d_i,c_2)}{2} \text{ otherwise} \end{cases}$$

The 6 submitted runs are :

All : The two first runs are the result of Random Committee classifiers that uses the Weka framework;

RF : The next two other runs are the result of Random Forest classifiers that uses the Weka framework;

SRF : The last two other runs are the result of Random Forest classifiers that uses the Salford University System;

For each one run is for classifying Garbage/Neutral over Central/Relevant, where the other is for seperating Central from relevant.

5 Results

	F1	SU
bests	.359	.410
RF-Yes (200)	.342	.278
RF-All (600)	.330	.279
SRF-All (400)	.326	.216
SRF-Yes (200)	.322	.228
All-All (450)	.318	.188
All-Yes (50)	.306	.193
medians	.289	.220
means	.220	.311

Table 3: *Results for central judgments for F1 and SU measures*

Table 3 and 4 show performances of our runs against official judgments. At first look, our results are quite good. For central judgements: all out runs are above the median; our best run get to the third place. Concerning relevant/central evaluation, 4 of them are still above the median and far better than the mean scores.

However, if we look for instance to our best run for central evaluation (RF-Yes) with a cutoff of 0, the

	F1	SU
bests	.639	.635
RF-All (250)	.617	.600
SRF-All (250)	.614	.601
All-All (250)	.603	.586
RF-Yes (0)	.581	.588
medians	.553	.554
SRF-Yes (50)	.537	.568
All-Yes (0)	.543	.549
means	.405	.498

Table 4: *Results for central and relevant judgments for F1 and SU measures*

precision is equal to 0.276. Now, if we look back at Table 2 we can see that proportions of central documents compare to documents with a mention is 0.232. This shows that our ranking method performs just a little bit better than random. This is obvious when looking at RF-All for central and relevant judgements : the precision for a cutoff of 200 is 0.567 and random would have done 0.648.

In this work, we tried to find what could make a document relevant considering the time. Our first and strong assumption was to say:

"for each topic, a document is more likely to be relevant if a huge amount of document appears at the same time."

It seems to describe well the behavior of micro blogs such as tweeter where once a news about someone or something is released, many tweets appear about it, where almost no tweet where mentioning the topic beforehand.

Figure 1 gives gini variables importances for each features on the training data for Random Forests. It shows that the number of documents in the queue (i.e., in a week) do not really affect the decisions made by the classifier. However, this may be explain because the way we did it is not relevant. Investigations have to be done about it before strong conclusions can be made.

Still considering the gini variables, we found that it is highly valuable to look at relations. Indeed when a relation is already known the document is almost automatically accepted. Thus by keeping track of new relations we could probably improve the classifier precision. Moreover a relation between entities may help to determine which of the homonyms the document is about even though finding relations is an hard task and therefore only a few are found.

Example 5.1 - example of disambiguation using relations between entities :

Say for instance we know that *"Boris Berezovsky"* (BB) (the pianist) plays at a specific concert hall. The extracted relation is:

$$\text{Boris Berezovsky} \rightarrow \text{PLAYS} \leftarrow \text{concert hall}$$

where *"Boris Berezovsky"* is the relation entity 1 (RE_1), *"concert hall"* is the relation entity 2 (RE_2) and *"PLAYS"* is the link l between the two elements. Let considers that in a future document appearing 2 month later the same relation appears. Since we know from a previous relation that Boris Berezovsky played at this concert hall, so it's more likely the pianist (and not the businessman) that plays back there 2 month later. So this feature might become a criteria that really helps disambiguating homonyms topics.

In addition, similarity between topics wikipedia's pages and the found documents has been also revealed by the decision tree classifier. This means that somehow a document that is about a specific topic may be similar to a document that has general overview of the topic. It could be interesting to measure the similarity between documents that appears the same day. Thus documents that echo the same information could be aggregated or at least have a heavier weight.

6 Conclusion

This task is a very challenging task as expected and therefore very interesting. This first year allows us to comprehend what is behind KBA. From now on we have learned a lot concerning the way we have to treat the stream corpus as well as the topics. We also know, since relations might be a key point, that having a system that we can rely on to find relation would probably improve our final results.

Moreover there are plenty of data we have not used (e.g, metadata, category of documents, the source,...) although it might be also important to consider a document for instance depending on the reliability of the source.

In this report we presented features that can be used to capture relevancy mainly independently of the entity evaluated. One of the natural perspective is to figure out how to combine them with topic specific approach in order to see how they can contribute to improve their results.

There are still many improvements that can be done, and since this is only the first years, it will make the following years quite promising.

```
OVERALLCOUNT_RELATED_ENTITIES_MENTION        100.00000
                  COSINE_SIMLARITY_1G        100.00000
                  COSINE_SIMLARITY_2G        100.00000
     COUNT_MENTION_IN_60_70%_DOCUMENT         88.33264
                         STAT_MENTION         73.70804
                  COUNT_RELATED_CITED         72.44485
     COUNT_MENTION_IN_10_20%_DOCUMENT         70.76657
               AVG_DOCUMENTS_IN_QUEUE         66.24520
          COUNT_SENTENCE_WITH_MENTION         65.93995
                 COUNT_RELATED_LINKED         65.86051
      COUNT_MENTION_IN_0_10%_DOCUMENT         57.96971
     COUNT_MENTION_IN_20_30%_DOCUMENT         55.61225
     COUNT_MENTION_IN_40_50%_DOCUMENT         51.39779
    COUNT_MENTION_IN_90_100%_DOCUMENT         49.98843
     COUNT_MENTION_IN_80_90%_DOCUMENT         49.00893
     COUNT_MENTION_IN_70_80%_DOCUMENT         45.33999
     COUNT_MENTION_IN_30_40%_DOCUMENT         30.62796
                        TERRIER_SCORE         25.70346
             COUNT_MENTION_IN_DOCUMENT         19.88681
     COUNT_MENTION_IN_50_60%_DOCUMENT         18.27151
    COUNT_MENTIONS_IN_PREVIOUS_TITLES         12.17387
 COUNT_MENTIONS_IN_PREVIOUS_DOCUMENTS         10.05529
      COUNT_MENTIONS_IN_CURRENT_TITLE          8.25128
                     BIGRAM_FREQ_AS_1          7.58489
  COUNT_MENTION_IN_PREVIOUS_SUMMARIES          6.93097
           STD_DEV_DOCUMENTS_IN_QUEUE          6.51563
                    TRIGRAM_FREQ_AS_1          6.42411
             COUNT_DOCUMENTS_IN_QUEUE          5.07882
       PREVIOUS_WITH_MENTION_IN_QUEUE          4.77457
```

Figure 1: *Gini variable importance for features on training data for Garbage/Neutral vs. Relevant/Central*