Using TF-IDF Weight Ranking Model in CLINSS as Effective Similarity Measure to Identify Cases of Journalistic Text Re-use

Yurii Palkovskii, Alexei Belov

Zhytomyr State University, MARS p.e., Plagiarism Detector Accumulator Project
palkovskiy@yandex.ru

Abstract. Journalistic text reuse is one of the emerging issues of the multilingual news space. 2012 CLITR event address this particular issue and this paper tries to outline one of the possible methods that can be applied to detecting parallel stories in different languages. The focus our research is made on detecting similarity between two texts that are written in different language pairs: English -Hindi and English-Gujarati. The developed application prototype does not discriminate between story detection and fragment detection, treating both cases in the same manner. The approach used implied the usage of automatic language translation - Google Translate web service to normalize one of the input texts to the target comparison language and apply ranking model that includes several filters, each of which adds ranking points to the final score. Four filters are the cross-linked TF-IDF similarity scores between different parts of the input pair, the fifth filter is a sliding window based TF-IDF comparer and finally, the Date Filter. Though the developed complex approach is purely statistical it showed promising result and can be further improved by applying machine learning algorithms for meta-parameters adjustments. This short paper covers the basic principles that we utilized to develop the comparer that will be able to effectively detect the above mentioned similarities regardless of the exact language pair under analysis.

Keywords: NLP, Similarity Detection, News Similarity Detection, Cross Lingual Similarity, Text Mining, Text Rewrite Detection, Text Reuse Detection

adfa, p. 1, 2011.

1 Introduction

One of the research priorities of our group is the experimentation with different methods of finding similarities between different types of input texts. That is why we got highly interested in the CL!NTR\FIRE 2012 initiative and decided to try our hand at detecting cross-lingual similarities within the Hindi-Gujarati-English language pairs. It was evident that tackling shared information quantum that contains the same sense, but having different text forms, completely excluded the possibility of effective using n-gram detection approach. This particular task resembles the type of heavily obfuscated plagiarism detection, both simulated and artificial at which we focused our efforts at CLEF\PAN2011 and CLEF\PAN2012 [3,4].

2 Methods

Before construction our own prototype we closely studied both motioned works mentioned at CL!NSS website - one by Dragos Munteanu [1] and Daniel Marcu and the other one by Emma Barker and Robert Gaizauskas [2].
We used Google Translate web frontend to manually translate both sets from English to Gujarati and Hindi (for purely technical reasons we didn't manage to launch an automatic API conversion - mostly due to the absence of the updated v.2 Google Translate API wrapper for dot net platform).

Each pair of documents in comparison stage get "final similarity score" by adding several sub scores for each particular comparison aspect.

We used the following aspects:

1. TF-IDF score comparing OD_Title with SD_Title.
2. TF-IDF score comparing OD_Text with SD_Text.
3. TF-IDF score comparing OD_Title with SD_Story.
4. TF-IDF score comparing OD_Story with SD_Title.
5. Maximum TF-IDF score comparing OD_Text with SD_Text, via each sentence vs each sentence.
6. If OD_Date falls into 10 days range in comparison with SD_date, 0,5 score is added to the final score.

As long as no automatic assessment script is available we decided to use heuristic weight based ranking model.

Final ranking algorithm is a straightforward sum of all values:

```
Public Function _getFSimScore() As Double
    Dim i_DateScoreDelta As Integer = 0
    If Me.i_DateDifference <= 10 Then
        i_DateScoreDelta = 0.5
    End If
    Return Me.dbl_tfidf_od_story_vs_sd_title + _
           Me.dbl_tfidf_od_title_vs_sd_story + _
           Me.dbl_tfidf_story_vs_story + _
           Me.dbl_tfidf_title_vs_title + _
           i_DateScoreDelta
End Function
```

We harvested most frequently used words from Hindi and Gujarati and removed them during preprocessing stage, top 5000 words from each language, excluding English non-translated entries.

Text preprocessing stage included exclusion of all symbols that are non alphanumeric. Sentence split made by the following splitters: "!?.…". Stemming was not applied.

Taking into consideration our previous experience of participation in CLEF\PAN2011 and CLEF\PAN2012 [3,4], we used the concept of "heavy caching" every intermediate result for every document pair at preprocessing stage, so that we ran two separate stages separately for data retrieval and data analysis, thus boosting the prototype "tune-in" and general program development.

Total prototype runtime is about 24 hours on a 6 core Intel 990Ex with 6 GB RAM on Windows 8 in C#\vb.net on Vertex 3 SSD drive.

3 Evaluation

No automatic evaluation script was available at the training stage, so we decided to adjust the input meta-parameters by an educated guess and check the effectiveness by artificially injecting 10 different news events harvested by hand from the latest news feeds (BBC and CNN) into the training set and then check what the performance will be, evaluating by the final position of the injected pairs. All the injected news stories fell into the top 10 results produced by the application prototype. So we decided that selected parameters satisfy the initial requirements.

Taken into consideration the final results obtained (the 1-st result at CL!TR) we may state that the suggested approach can perform as a starting baseline and it can be further improved by adding machine learning strategies to achieve much better results.

4 Conclusions

Future research can be focused on the investigation of what ranking filters and comparison methods work at each particular case. Introducing an automated evaluation framework makes possible to employ a wide range of automated parameter adjustment models and dramatically boosts "tuning-in" stage of any practical approach. Thus the exact influence of the each particular ranking filter can be cleanly evaluated. We believe that combining purely statistical methods with some semantic similarity measurement will be beneficial and will probably get even better results. Further study of such hybrid systems is a key priority in the research of our group and we hope to further investigate the opportunities for tackling "story detection"\"fragment detection", "focal\non focal event subdivision", "derived\non derived content" and Text Reuse Classification aspects.

References

1. Barron-Cedeno, Alberto and Paolo Rosso. 2009. On Automatic Plagiarism Detection based on n-grams Comparison. In Mohand Boughanem, Catherine Berrut, Josiane Mothe, and Chantal Soule-Dupuy, editors, ECIR 2009, volume 5478 of LNCS, pages 696–700, Toulouse, France. Springer.
2. Dragos Munteanu and Daniel Marcu (2005). Improving Machine Translation Performance by Exploiting Comparable Corpora. Computational Linguistics, 31 (4), pp. 477-504, December.
3. Emma Barker and Robert Gaizauskas (2012). Assessing the Comparability of News Texts. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12).
4. International Competition on Plagiarism Detection. 2009. http://www.webis.de/pan-09/competition.php.
5. Potthast, M., Stein, B., Barrón-Cedeño, A., Rosso, P.: An evaluation framework for plagiarism detection. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. pp. 997–1005. Association for Computational Linguistics (2010).
6. Potthast, M., Barrón-Cedeño, A., Eiselt, A., Stein, B., Rosso, P.: Overview of the 2nd International Competition on Plagiarism Detection. In: Braschler et al. [1]
7. Potthast, Martin et.al. (editors). 2009. PAN Plagiarism Corpus PAN-PC-09. http://www.webis.de/research/corpora.