SpamChallenge 2007 :
France Telecom R&D Submissions

Tanguy Urvoy and Emmanuel Chauveau,
Pascal Filoche and Thomas Lavergne*
France Telecom R&D

In our article of Airweb 2006 workshop [4], we combined the approaches of HTML noise preprocessing (removing content), minhash fingerprinting and similarity clustering to spot dubious sets of web pages. For this challenge the idea is the same but we study more preprocessing and clustering strategies that we use to smooth the predictions of a classifier. We test two learning methodologies and sumbit two predictions.

Preprocessing

We test six different HTML preprocessings :

- *html_noise* removes any alpha-numeric characters;
- *html_noise_var_spaces* same process with removing of sequential spaces;
- *html_tags* keeps all tags content;
- *html_tags_and_noise* keeps noise in tags.

Two more preprocessing are used in order to compare those strategies with more standard filters :

- *html_to_words* outputs every alphanumeric character outside of tags;
- *full* outputs the initial document unmodified.

We combine these preprocessings with two LSH fingerprinting algorithms : Broder minhashing [2] and Charikar fingerprinting [3]. We obtain twelve fingerprints per web page hence twelve clusterings.

Clustering and smoothing

To compute these clusterings, we use a *"mutli-sort sliding window edge detection algorithm"* to approximate the similarity graph. The url clusters are the connected components of similarity graphs. The whole clustering process is described in Fig 1.

Given a spam prediction, a spam probability and an url cluster we produce the following smoothed cluster features :

- number of spam/ham pages in cluster (S, H);
- discrete spamicity : $\frac{\frac{S-H}{S+H}+1}{2}$;

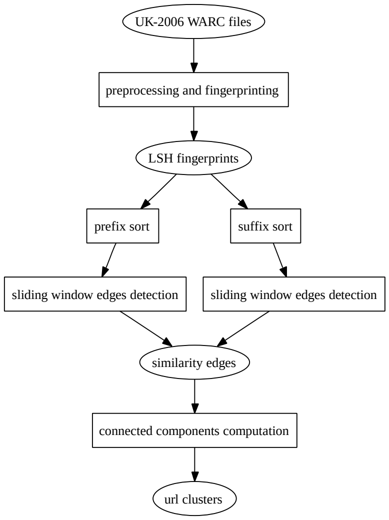
UK-2006 WARC files
preprocessing and fingerprinting
LSH fingerprints
prefix sort
suffix sort
sliding window edges detection
sliding window edges detection
similarity edges
connected components computation
url clusters

Figure 1: Process to obtain clusters from data using given preprocessing and LSH fingerprint.

- average spamicity and standard deviation;

To transform these cluster-features into host-features, we assign to each host :

- the features of the cluster which contain the most of its urls (dominant features);
- the weighted mean features for all clusters containing its urls (average features).

Features selection and classification

The informations provided by the clusterings in addition to the provided corpus based features (mostly direct, link based, transformed link based, and content based features, plus stacked graphical learning scores) are processed by the MODL (selective Bayesian) classifier [1] which propose a tag and a confidence score in this tag for each untagged host. We also use MODL to select and ponderate the most informative cluster-based features. We test two strategies and sumbit two predictions.

Submission 1. The first submission is described by Fig. 3. A first MODL classifier is trained on all features : direct, link based, transformed link based, content based features, and

Web Spam Challenge, May 8, 2007.

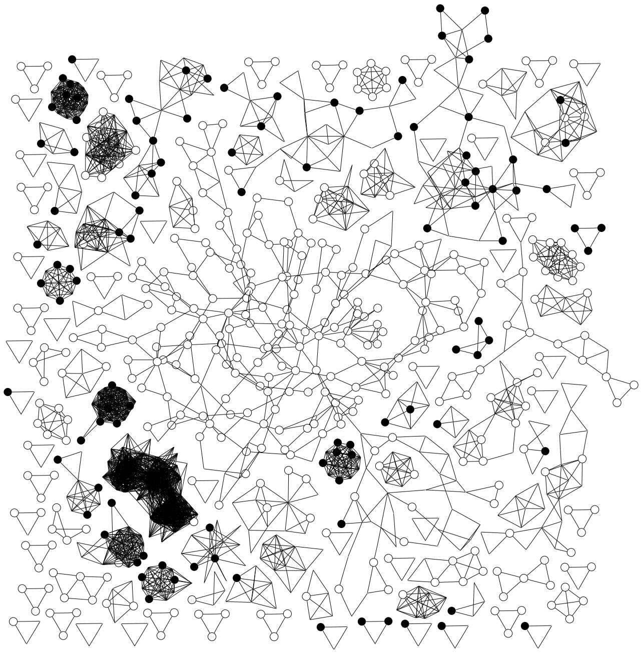

Figure 2: An HTML *html_noise_var_spaces* similarity graph : black nodes indicate spam labels and white nodes indicates normal labels, other are unknown.

stacked graphical learning scores. Its predictions are then smoothed according to each clustering. A second MODL classifier is trained on the cluster-based features to product the final prediction. We find 6863 normal and 2081 spam hosts and we expect a $F1$ measure around 0.75. This estimation is probably biased because we did not know the k-fold validation partition used for stacked graphical learning scores.

submission 2. The second submission is described by Fig. 4. We only inject the two passes stacked graphical learning score to complete the smoothed labels. A MODL classifier is then trained on cluster-based features and other corpus features except original stacked graphical learning scores. We find 6977 normal and 1967 spam hosts and we expect a $F1$ measure around 0.8. This estimation is probably also biased.

1. REFERENCES

[1] Marc Boullé. Modl: A bayes optimal discretization method for continuous attributes. *Machine Learning*, 65(1):131–165, 2006.

[2] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig. Syntactic clustering of the web. In *Selected papers from the sixth international conference on World Wide Web*, pages 1157–1166, Essex, UK, 1997. Elsevier Science Publishers Ltd.

[3] Moses Charikar. Similarity estimation techniques from rounding algorithms. In *STOC*, pages 380–388, 2002.

[4] T. Lavergne T. Urvoy and P. Filoche. Tracking web spam with hidden style similarity. In *International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, 2006.

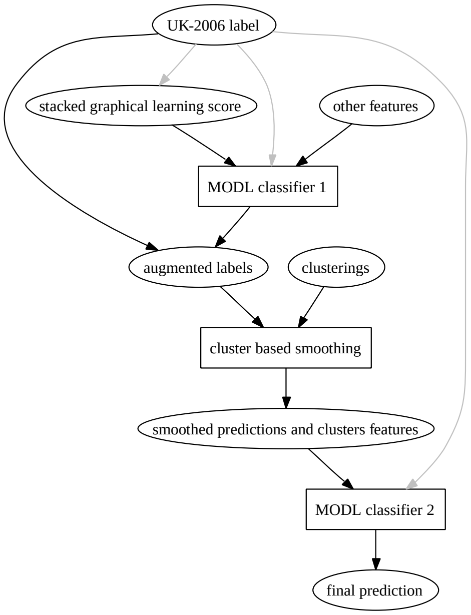

Figure 3: Process to tag urls for submission 1.

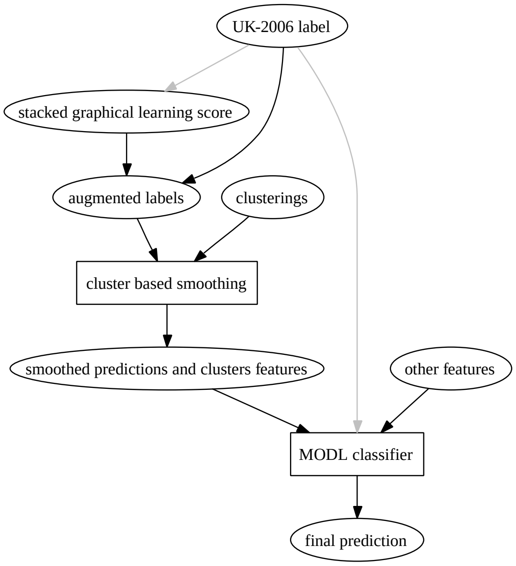

Figure 4: Process to tag urls for submission 2.