Detecting Spam in Web Corpora

Vít {Baisa, Suchomel}

NLP Centre
MU

RASLAN 2012

Introduction

- Recently, many fake webpages have appeared on Web.
- It is bad source for purpose of crawling webpages.
- How to get rid of the *spam* on Web?
- We will describe this unwanted content,
- an experiment with filtering it using frequencies of all possible n-grams and
- evaluation of this approach giving approx. 75 % accuracy.

Why Web Spam Is Harmful

- To increase Page rank on Google, webpages full of hyperlinks and low quality text are generated and put online.
- Since fake webpages are aimed at attracting attention on a theme (*viagra*, *loan*, *mortgage*, *insurance*) it gives them higher frequency at the expense of the others.
- It is unwanted content in both search results and web corpora.
- E.g. *viagra* is 100× more frequent in recently (2012) crawled corpus than in its predecesor from 2008.

Comparing Keywords in 2012 and 2008

	lemma	**2012**	**2008**	**RFR**
1	**loan**	360.1	51.7	6.97
2	online	462.4	119.2	3.88
3	your	4194.4	1660.2	2.53
4	**insurance**	263.1	56.8	4.63
5	**credit**	321.7	119.9	2.68
6	buy	421.3	175.7	2.40
7	**mortgage**	132.4	22.9	5.78
8	product	502.6	219.6	2.29
9	brand	164.3	41.8	3.93
10	website	261.9	94.5	2.77
. . .				
21	**debt**	150.9	48.5	3.11

Keywords from focus corpus enTenTen12 (2012), reference corpus enTenTen08 (2008).

Spam Techniques

- Dumping of a large number of unrelated terms,
- generating text from scratch,
- inserting spam words into copied webpages,
- gluing text segments (s, p) from various sources.

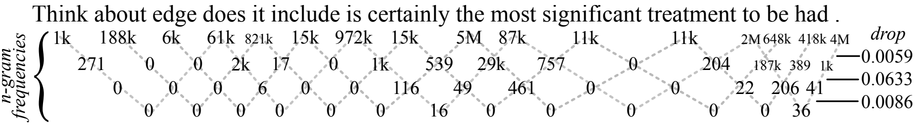
This paper deals with the first two types. The third and partially the fourth type may be detected and removed within deduplication.

Another problem and challenge is to detect *content farms* which are often made manually and contain natural language and so are out of scope of the following approach.

Detecting Spam Using All N-grams Frequencies

- It was shown that high number of frequent n-grams correlates with fluency of a text (BLEU).
- Given all n-grams from a reference corpus we evaluate suspicious (possibly spam) texts by counting frequencies of all n-grams from them.
- From a given sentence or paragraph we extract one vector with *frequency-drops* on various n-gram levels.
- Such vectors are then used for machine learning and automatic classification of spam and no-spam.

The Approach On Figure



Think about edge does it include is certainly the most significant treatment to be had .

n-gram frequencies

1k 188k 6k 61k 821k 15k 972k 15k 5M 87k 11k 11k 2M 648k 418k 4M *drop*

271 0 0 2k 17 0 1k 539 29k 757 0 204 187k 389 1k — 0.0059

0 0 0 6 0 0 116 49 461 0 0 22 206 41 — 0.0633

0 0 0 0 0 0 16 0 0 0 0 0 36 — 0.0086

Getting All N-grams From Corpora

- Even there are $O(n^2)$ of all n-grams we are able to count their frequencies in almost linear time.
- We used algorithm described by Church and Yamamoto which exploit suffix array and longest common prefix array.

distinguish	at	all	between	the	personal	...
distinguish	at	least	between	the	meaning	...
distinguish	at	least	four	sources	in	...
distinguish	at	least	three	cases	:	...
distinguish	at	once	from	the	destruction	...
distinguish	at	the	outset	between	the	...

Part of suffix array built form BNC, n-grams starting with *distinguish*.

Evaluation Data

- For evaluation we prepared set of 1045 sentences in 406 paragraphs.
- Paragraphs were manually annotated as spam or not-spam.

text category	**class**	**% doc**
nice text	OK	37.0 %
low quality text (possibly a content farm)	OK	18.5 %
fluency slightly broken (sentence or paragraph stitching)	spam	9.8 %
fluency broken (sentence or phrase stitching)	spam	13.0 %
not fluent (triplets of words stitching)	spam	14.1 %
nice text, unrelated words (spam terms weaving)	spam	7.6 %

Classification of texts from the collection of 92 web documents containing word *loan*.

Evaluation Metrics

$$precision = \frac{tp}{tp + fp}$$

$$recall = \frac{tp}{tp + fn}$$

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

$$f\text{-}score = 2 \times \frac{precision \times recall}{precision + recall}$$

Results

	BASE	**sD**	**sT**	**SVM**	**SVM^c**	**SVM$_a$**	**SVM$_a^c$**	**^{2}SVM$_a^c$**
prec	48.53	55.36	49.72	31.31	83.84	72.73	84.85	89.90
rec	100.0	97.03	90.91	50.00	63.36	64.29	62.22	68.46
f-sc	65.35	70.50	64.29	38.51	72.17	68.25	71.79	77.73
acc	48.53	59.41	50.98	51.47	68.63	67.16	67.65	75.00

Conclusion & future work

Conclusion

- We must deal with spam content.
- Computation of feature vector is very fast (2.10^6 tokens per minute).
- Classification was done on paragraph level.
- Classifications should be done by more annotators and native speakers.
- Use oficial evaluation data from various competitions.

Future work

- Creating more test data (SciGen, bullshit ipsum, etc.)
- Domain-specific reference corpus.
- Bigger corpus (BNC used).

Is This Spam?

It is a means for this company to essentially create believe in with potential clients, and raises a warning sign if the membership rights is lost for reasons unknown.

If a human being has a payday loan through money mart canada branch, are they able to get a nd they decidedly love people who are desperate so yes you can get check cash instant loan now cash instant loan now online.