Spam Detection via Constraint Programming

Evgeny Skvortsov
Simon Fraser University
evgenys@sfu.ca

ABSTRACT

The presented results are obtained by combining some previous approaches with modeling the spam detection as a constraint satisfaction problem. In this work we focus on hostgraph-based spam detection.

1. THE MODEL

An instance of a constraint satisfaction problem(CSP) is a triple $(V, D, \mathcal{C})$, where V is a set of variables, D is a set of their possible values and $\mathcal{C}$ is a set of constraints. Each constraint $c \in C$ is a pair $(\vec{s}, \rho)$, where $\vec{s}$ is some vector of variables and ρ is a set of tuples that are allowed combinations of values of variables $\vec{s}$.

We model web spam detection on a hostgraph $G = (V, E)$ as the following CSP:

- V — is a set of all hosts,
- $D = \{0, 1\}$, where 0 represents spam and 1 represents normal site.
- $\mathcal{C} = \mathcal{C}_E \cup \mathcal{C}_T \cup \mathcal{C}_D$, where
  - $\mathcal{C}_E = \left\{ \left((s,e), \begin{Bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \end{Bmatrix} \right) \middle| (s,e) \in E \right\}$,
  - $\mathcal{C}_T = \{(v, 1) | v \in \texttt{DMOZ}\}$, (constraints of trust)
  - $\mathcal{C}_D = \{(v, 0) | v \in \texttt{SPAM}\}$. (constraints of distrust)

It is easy to see that each constraint in $\mathcal{C}_E$ is an implication "If start of an edge **e** is a normal page then the end of the edge **e** is a normal page." Constraints in $\mathcal{C}_T$ are claiming that hosts from Open Directory are not spam and finally constraints in $\mathcal{C}_D$ are making hosts known as spam to be marked 0.

2. ASSEMBLING THE ALGORITHM

In practice we don't look for exact solution of the formulated CSP, but extend possible values of variables to the real segment $[0, 1]$ and run gradient descent on the penalty function computed based on violated constraints. We set $\texttt{SPAM} = V$, but give smaller weights to constraints in $\mathcal{C}_D$.

The values obtained after a fixed number of steps of the gradient decent are passed together with LinkRank and PageRank to a machine learning algorithm to estimate final probability of a particular page to be spam.

The machine learning classifier was built using WEKA package. It consists of several decision tree classifiers of different types, which outputs are combined through voting.

3. FUTURE WORK

Natural extension of the algorithm will be be using content based features for setting trust and distrust constraints.

4. ACKNOWLEDGMENTS

The author is grateful to Ian Andrew Bell, Einar Vollset and Weston Triemstra for multiple fruitfull discussions and comments on the work.

The work is supported by SomethingSimpler Systems and ACCELERATE BC program.