HYBRID APPROACH FOR WEB PAGE CLASSIFICATION BASED ON FIREFLY AND ANT COLONY OPTIMIZATION

Poonam Asawara, Dr Amit Shrivastava and Dr Manish Manoria

Department of Computer Science and Engineering SIRTS Bhopal, India

ABSTRACT:

The rapid development of the internet and web publishing techniques create numerous information sources published as HTML pages on World Wide Web. WWW is now a popular medium by which people all around the world can spread and gather the information of all kinds. But web pages of various sites that are generated dynamically contain undesired information also. This information is called noisy or irrelevant content. The need for innovative and effective technologies to help find and use the useful information and knowledge from a large variety of data sources is continually increasing. Web information has become increasingly diverse. In order to utilize the Web information better, people pursue the latest technology, which can effectively organize and use online information. Classification is one of the vital and important data mining techniques that grouped various items in a collection to predefined classes or groups. The main goal of classification is to exactly predict the target class for each case in the data. Web Page Classification is technique of data mining to discover classification of web pages. The information providers on the web will be interested in techniques that could improve the effectiveness of the web search engine. In this paper, the relationships among the techniques used in data mining are studied. A study of web usage is also done on optimization of this web classification.

Keywords: Random Forest Classifier, Ant Colony Optimization, Firefly, Web Page Classification.

[1] INTRODUCTION

Classification plays a vital role in many information management and retrieval tasks. On the Web, classification of page content is essential to focused crawling, to the assisted development of web directories, to topic-specific Web link analysis, to contextual advertising, and to analysis of the topical structure of the Web. Web page classification can also help improve the quality of Web search [1].

In this survey we examine the space of Web classification approaches to find new areas for research, as well as to collect the latest practices to inform future classifier implementations. Surveys in Web page classification typically lack a detailed discussion of the utilization of Web-specific features. In this survey, we carefully review the Web-specific features and algorithms that have been explored and found to be useful for Web page classification. The contributions of this survey are

— a detailed review of useful Web-specific features for classification;
— an enumeration of the major applications for Web classification; and  
— a discussion of future research directions \[2\]

At present, the numbers of Web-pages on World Wide Web are increasing significantly. The task to find Web-pages which present information satisfying our requirements by traversing hyperlinks is difficult. Therefore, we use search engines frequently on the portal site. There are two kinds of search engines, i.e., directory-style search engines such as Yahoo, and robot style ones such as goo, excite and altavista. The latter displays the lists of Web-pages which contain input keywords without checking themes characterizing respective Web-pages. For this reason these search engines are likely to provide misdirected Web-pages. On the other hand, in directory-style search engines, Web-pages stored in a database are classified with hierarchical categories compatible with their themes in order. This enables us to obtain Web-pages including information that meets our purpose by not only following input keywords but also traversing hyperlinks classifying Web-pages into categories in systematic order \[3\].

\[2\] **ANT COLONY OPTIMIZATION**

Swarm intelligence studies the collective behavior of unsophisticated agents that interact locally through their environment \[4\]. It is inspired by social insects, such as ants and termites, or other animal societies, such as fish schools and bird flocks. Although each individual has only limited capabilities, the complete swarm exhibits complex overall behavior. Therefore, the intelligent behavior can be seen as an emergent characteristic of the swarm. When focusing on ant colonies, it can be observed that ants communicate only in an indirect manner—through their environment—by depositing a substance called pheromone. Paths with higher pheromone levels will more likely be chosen and thus reinforced, while the pheromone intensity of paths that are not chosen is decreased by evaporation. This form of indirect communication is known as stigmergy, and provides the ant colony shortest-path finding capabilities.

ACO employs artificial ants that cooperate to find good solutions for discrete optimization problems \[5\]. These software agents mimic the foraging behavior of their biological counterparts in finding the shortest-path to the food source. The first algorithm following the principles of the ACO metaheuristic is the Ant System \[6\], \[7\], where ants iteratively construct solutions and add pheromone to the paths corresponding to these solutions.

Path selection is a stochastic procedure based on two parameters, the pheromone and heuristic values. The pheromone value gives an indication of the number of ants that chose the trail recently, while the heuristic value is a problem dependent quality measure. When an ant reaches a decision point, it is more likely to choose the trail with the higher pheromone and heuristic values. Once the ant arrives at its destination, the solution corresponding to the ant’s followed path is evaluated and the pheromone value of the path is increased accordingly. Additionally, evaporation causes the pheromone level of all trails to diminish.
gradually. Hence, trails that are not reinforced gradually lose pheromone and will in turn have a lower probability of being chosen by subsequent ants.

[3] FIREFLY

Firefly Algorithm (FA) is an optimization technique, developed recently by Xin-She Yang at Cambridge University [8]. It is inspired by social behavior of fireflies and the phenomenon of bioluminescent communication. Fireflies can generate light inside of it. Light production in fireflies is due to a type of chemical reaction. It is thought that light in adult fireflies was originally used for similar warning purposes, but evolved for use in mate or sexual selection via a variety of ways to communicate with mates in flirtations. Although they have many mechanisms, the interesting issues are what they do for any communication to find food and to protect themselves from enemy hunters including their successful reproduction.

In general, the pattern of flashes is unique for a particular species of fireflies. The flashing light is generated by a chemical process of bioluminescence. However, two fundamental functions of such flashes are i) to attract mating partners or communication, and ii) to attract potential victim. Flashing may also be used for a protective warning mechanism. The light intensity at a particular distance from the light source follows the inverse square law. It means that, as the distance increases, the light intensity decreases.

Furthermore, the air absorbs light which becomes weaker and weaker as there is an increase of the distance. There are two combined factors that make most fireflies visible only to a limited distance that is usually good enough for fireflies to communicate each other. The flashing light can be formulated in such a way that it is associated with the objective function to be optimized. This makes it possible to formulate new metaheuristic algorithms. The main steps of FA described in.

![Figure 1. Firefly Algorithm](image-url)
[4] RANDOM FOREST CLASSIFICATION

The Random forest [9] is a meta-learner which consists of many individual trees. Each tree votes on an overall classification for the given set of data and the random forest algorithm chooses the individual classification with the most votes. Each decision tree is built from a random subset of the training dataset, using what is called replacement, in performing this sampling. That is, some entities will be included more than once in the sample, and others won't appear at all. In building each decision tree, a model based on a different random subset of the training dataset and a random subset of the available variables is used to choose how best to partition the dataset at each node. Each decision tree is built to its maximum size, with no pruning performed. Together, the resulting decision tree models of the Random forest represent the final ensemble model where each decision tree votes for the result and the majority wins.

There are two different sources of randomness in Random forest: random training set(bootstrap) and random selection of attributes. Using a random selection of attributes to split each node yields favorable error rates and are more robust with respect to noise. These attributes form nodes using standard tree building methods. Diversity is obtained by randomly choosing attributes at each node of a tree and using the attributes that provide the highest level of learning. Each tree is grown to the fullest possible without pruning until no more nodes can be created due to information loss. In Breiman’s early work[9], each individual tree is given an equal vote and later version of random forest allows weighted and un-weighted voting.

The random forest algorithm computes the out-of-bag error. The average misclassification for the entire forest is called as out-of-bag error which is useful for predicting the performance of the classifier without involving the test set example: cross-validation. The out-of-bag error of random forest depends on the strength of the individual trees in the forest and the correlation between them. With a less number of attributes used for split, correlation between any two trees decrease and the strength of a tree decreases. These two have reverse effect on error rates of random forest: less correlation increases the error rate while less strength decrease the error rate.

[5] LITERATURE REVIEW

Web page classification is significantly different from traditional text classification because of the presence of some additional information, provided by the HTML structure and by the presence of hyperlinks. In this paper we analyze these peculiarities and try to exploit them for representing web pages in order to improve categorization accuracy. We conduct various experiments on a corpus of 8000 documents belonging to 10 Yahoo! categories, using Kernel Perceptron and Naive Bayes classifiers. Our experiments show the usefulness of dimensionality reduction and of a new, structure-oriented weighting technique. We also introduce a new method for representing linked pages using local information that makes
hypertext categorization feasible for real-time applications. Finally, we observe that the combination of the usual representation of web pages using local words with a hypertextual one can improve classification performance. [10].

In this paper we use machine learning algorithms like SVM, KNN and GIS to perform a behavior comparison on the web pages classifications problem, from the experiment we see in the SVM with small number of negative documents to build the centroids has the smallest storage requirement and the least on line test computation cost. But almost all GIS with different number of nearest neighbors have an even higher storage requirement and on line test computation cost than KNN. This suggests that some future work should be done to try to reduce the storage requirement and on list test cost of GIS[11].

[6] WORK DONE

![Flow Chart of Proposed Work](image.png)

Figure 2: Flow Chart of Proposed Work
1. Load web page dataset
2. Search some features
3. Main features (mf) and their relating sub features (sf) found
4. Sort rf with its associate mf
5. Allot separate numbers for every feature combinations
6. Optimize features with firefly optimization
7. Generate initial population of fireflies
   For i = 1 : n (firefly generations)
   For j = 1: i
   Move firefly towards bright light intensity
   Attractiveness varies with distance ‘r’
   Evaluate new solutions and update light intensity
   For i end
   For j end
   Rank fireflies and find the best
   For end
8. Firefly out value optimize with Ant Colony Optimization
   a. Initialize pheromone parameters
   b. Generate initial populations (ants)
   c. For each individual ant calculate fitness
   d. For each ant determine its best position
   e. If best global ant determined
   f. Update the pheromone trail
   g. Else
   h. Go to ‘b’
   i. Output is optimized values
9. Classify with Random forest classifier
10. Got comparative parameters

[7] RESULT ANALYSIS

This section is dealing with the implementation portion of the proposed work. For the same, this section in divided into two parts:

a. System Configuration
   The system on which these experiments are performed and verify the results:

Table 1: System Configuration

<table>
<thead>
<tr>
<th>Model</th>
<th>Sony Vaio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>Intel® Core™ i5-2450M</td>
</tr>
<tr>
<td>RAM</td>
<td>4GB</td>
</tr>
<tr>
<td>System Type</td>
<td>64 Bit Operating System</td>
</tr>
<tr>
<td>Windows</td>
<td>Windows 10 Home</td>
</tr>
<tr>
<td>MATLAB</td>
<td>R2014a</td>
</tr>
</tbody>
</table>
Table II shows the detail about the dataset used. According to the information, dataset is divided into four main categories. These main categories are further divided into sub-categories.

<table>
<thead>
<tr>
<th>Dataset Character</th>
<th>Dataset Category</th>
<th>Associated Theme</th>
<th>No. of associated files</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Account</td>
<td>Banking</td>
<td>200</td>
</tr>
<tr>
<td>B</td>
<td>Loans</td>
<td>Banking</td>
<td>200</td>
</tr>
<tr>
<td>C</td>
<td>Retail Finance</td>
<td>Banking</td>
<td>200</td>
</tr>
<tr>
<td>D</td>
<td>JavaLessons</td>
<td>Programming</td>
<td>200</td>
</tr>
<tr>
<td>E</td>
<td>Interpreter</td>
<td>Programming</td>
<td>200</td>
</tr>
<tr>
<td>F</td>
<td>Visual Basic</td>
<td>Programming</td>
<td>200</td>
</tr>
<tr>
<td>G</td>
<td>Galaxies</td>
<td>Science</td>
<td>200</td>
</tr>
<tr>
<td>H</td>
<td>Complexity</td>
<td>Science</td>
<td>200</td>
</tr>
<tr>
<td>I</td>
<td>Training</td>
<td>Sport</td>
<td>200</td>
</tr>
<tr>
<td>J</td>
<td>Formula One</td>
<td>Motor Sport</td>
<td>200</td>
</tr>
<tr>
<td>X</td>
<td>Basketball</td>
<td>Sport</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Total: 2200</td>
</tr>
</tbody>
</table>

Table III gives detail about the dataset used for training and testing phase. According to this table every category of the dataset is further partitioned into two categories.

<table>
<thead>
<tr>
<th>Dataset Category</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Relevant/Total</td>
<td>Relevant/Total</td>
</tr>
<tr>
<td>Banking</td>
<td>220/1100</td>
<td>190/1100</td>
</tr>
<tr>
<td>Programming</td>
<td>152/1100</td>
<td>140/1100</td>
</tr>
<tr>
<td>Science</td>
<td>160/1100</td>
<td>132/1100</td>
</tr>
<tr>
<td>Sport</td>
<td>346/1100</td>
<td>106/1100</td>
</tr>
</tbody>
</table>

Table IV: Comparison of Sensitivity

<table>
<thead>
<tr>
<th>Existing Work</th>
<th>Proposed Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.925</td>
<td>0.992657</td>
</tr>
</tbody>
</table>

Graph 1: Comparison of Sensitivity
Sensitivity measures the proportion of positives that are correctly identified as such (e.g., the percentage of banking pages which are correctly identified as having the categories and subcategories).

Comparison of Sensitivity

<table>
<thead>
<tr>
<th></th>
<th>Existing Work</th>
<th>Proposed Work</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensitivity</td>
<td>0.794993</td>
<td>0.836186</td>
</tr>
</tbody>
</table>

[8] CONCLUSION

This paper has discussed about three areas i) Web classification, ii) Optimization method (which is used by the web classification method to increase the efficiency or say decrease the complexity), iii) Random Forest Classification has been discussed. This article is helpful for evaders to have a deep insight into web page classification. This study motivates us to do further work in the area of optimized web page classification with the help of danger theory.

REFERENCES


[10] Daniele Riboni, D.S.I., University’ degli Studi di Milano, Italy, “Feature Selection for Web Page Classification”.