ENERGY AWARE WEB BROWSING ON HAND HELD SMART DEVICES USING ATTENTION TIME PREDICTION

G.R. Hemalakshmi,
Assistant professor,
Department of computer science,
National Engineering College, Kovilpatti, India

S.Ananthi,
UG Scholar,
Department of computer science,
National Engineering College, Kovilpatti, India

G. Emima Pon Glory,
UG Scholar,
Department of computer science,
National Engineering College, Kovilpatti, India

Abstract: Web browsing on hand held smart devices consume enormous amount of energy especially during downloading of web pages. This energy consumption problem can be overcome by using the proposed methods. Downloading of webpages involve two phases, they are data transmission computation and layout computation. In our method it is intended to switch between various states like IDLE, DCH and FACH state in a way that changing the device among these states based upon the power and resource requirements. Then the attention time prediction technique is used to predict the user reading time of a web page and it checks whether the user reading time is greater than the threshold time if so then bring the device into low power state (i.e., FACH state) and prevent the high power consumption.

Keywords: Data transmission computation, Energyconsumption,Hand held smart devices, Layout computation, Reading time threshold, Web browser.

1. INTRODUCTION

Smartphones offer a most important and commonly used service i.e. web browsing. However, the current smartphone web browser wastes a lot of power when downloading webpages due to the special characteristics of the wireless radio interface. There are many researches in reducing the power consumption of smart phone based web browsing. In this paper, we focus on the two proposed techniques. Generally, there are two steps in downloading a webpage namely, Data transmission computation and Layout computation. In our proposed system, firstly we switch between various states like IDLE, DCH and FACH states in a way that, transforming the device among these states based upon the power and resource requirements. We allocate two third of the time slot to data transmission computation then the web browser can put the wireless radio interface into low power state, release the radio resource, and then run the remaining computations then one third of the time slot to the layout computation.

Secondly, we use attention time prediction to predict the user reading time of a web page. This web mining technique uses already existing collection of pages and their predicted reading time. It collects many features about the web page requested, such as the reading time, page size, layout design etc. once it gets downloaded. Then it performs content similarity analysis with the already existing collection of web pages and it checks whether the user reading time is greater than the threshold time if so then bringing the device in to low power state and prevent the more power consumption.

2. LITERATURE SURVEY

2.1 Energy-Aware Web Browsing on Smartphones

Bo Zhao reorganize the computation sequence of the web browser to retrieve all data in the webpage as fast as possible. In the original web browser design, the data transmission computation and the layout computation are mixed. His approach compares the energy-aware computation sequence with that of the original web browser for opening a webpage. At time slot 1, the original web browser spends its computation resource on both data transmission and layout computation. It processes one object and adds it to the DOM tree. In his approach, the browser focuses on data transmission and ignores layout. Thus, his approach can process more objects and add them to the DOM tree at time slot 1[1]. At time slot 2, his approach processes all the objects and builds the complete DOM tree. At time slot 3, no data transmission will happen, and his approach focuses on computing the page layout. Then he present a machine learning based approach called Gradient Boosted Regression Trees to predict the user reading time, by constructing statistical decision tree models from historical data based on which we can decide if the smartphone should switch to IDLE. Since different users have different reading patterns, he build the prediction engine for each user individually.

2.2 Energy Efficient Web Surfing for Smart Phones

Styling tasks apply the cascading algorithm and create the final style data structure for each node. A styling task must satisfy two dependencies before it can execute. First, it can only execute after the matching task working on the same node has completed execution, since the cascading algorithm uses the rules selected by the matching task. Second, a styling task working on a node can only execute after the styling task working on the node's parent has completed execution. By using two types of tasks, **Mulik Umesh** and **et al** proposed an algorithm called Parallel DOM Styling algorithm that performs rule-match a node before its parent is styled [2]. This basic version of the algorithm limits style sharing to parent child sharing. However, we subdivide style objects into sub styles containing related properties, and allow sharing at the sub style level, which increases the degree of available sharing. For example, if a child node uses the same font properties as its parent, then they can share the font sub style.

2.3 Efficient Energy Consumption Approach during Browsing in Android Smartphones

Ashwini M. Sonwane suggests a method called image formats comparison and Optimization to reduce power consumption in smart phones that says JPEG images is appreciably less expensive than other formats such as GIF and PNG for comparable size images on the Android browser [3]. For example, try to translate all images on the Facebook web site to JPEG and check the result to obtain considerable energy savings. Their conversion easily changes working of JavaScript on the page, without altering the user experience.

2.4 Bartendr: A Practical Approach to Energy-aware Cellular Data Scheduling

Aaron Schulman and et al, introduced an approach called Bartendr which reduces power consumption issues in cellular data scheduling. Bartendr strives for energy efficiency by scheduling communication during periods of strong signal. To accomplish this, it predicts signal strength minutes into the future. For example, Bartendr can predict efficient times to wake up and sync email, and intervals when data should be downloaded [4]. First, they describe how Bartendr uses prior tracks of signal strength to predict future signal strength. Then, they compare tracks to alternate methods of signal prediction based on location and history. Finally, they present algorithms that use the predicted signal strength to efficiently schedule syncs and streaming media.

2.5 Fast and Parallel Webpage Layout

Leo A. Meyerovich and et al introduce new algorithms for CSS selector matching, layout solving, and font rendering, which represent key components for a fast layout engine. Their sequence of algorithms is the following Step 1 (selector matching) determines, for every HTML node, which style constraints apply to it. Steps 2, 4-6 (box and text layout) solve layout constraints. Each step is a single parallel pass over the HTML tree. Consider a node's font size, which is constrained as a concrete value or a percentage of its parent's: step 2 shows that once a node's font size is known, the font size of its children may be determined in parallel. Note text is on the DOM tree's fringe while boxes are intermediate nodes. Step 3 (glyph handling) determines what characters are used in a page, calls the font library to determine character constraints (e.g., size and kerning), and renders unique glyphs. Step 7 (painting or rendering) converts each shape into a box of pixels and blends it with overlapping shapes. They found parallelism within every step, and, in some cases, even obtained sequential speedups. While their layout process has many steps, it essentially interleaves four algorithms: a CSS selector matcher, a constraint solver for CSS-like layouts, and a glyph renderer [5]. They can individually examine the first three algorithms, where they achieve speedups from 3x to 80x. Beyond the work presented here, they are applying similar techniques to related tasks between steps 1 and 2 like cascading and normalization, and GPU acceleration for step 8, painting.

2.6 A User-Oriented Webpage Ranking Algorithm Based on User Attention Time

Songhua Xu and et al. proposed a new webpage ranking algorithm which is personalized. Their idea is to rely on the attention time spent on a document by the user as the essential clue for producing the user-oriented webpage ranking. The prediction of the attention time of a new webpage is based on the attention time of other previously browsed pages by this user [6]. To acquire the attention time of the latter webpages, they developed a browser plugin which is able to record the time a user spends reading a certain webpage and then automatically send that data to a server. Once the user attention time is acquired, they calibrate it to account for potential repetitive occurrences of the webpage before using it in the prediction process. After the user's attention times of a collection of documents are known, their algorithm can predict the user's attention time of a new document through document content similarity analysis, which is applied to both texts and images. They evaluate the webpage ranking results from our algorithm by comparing them with the ones produced by Google's PageRank algorithm.

3. SYSTEM DESIGN

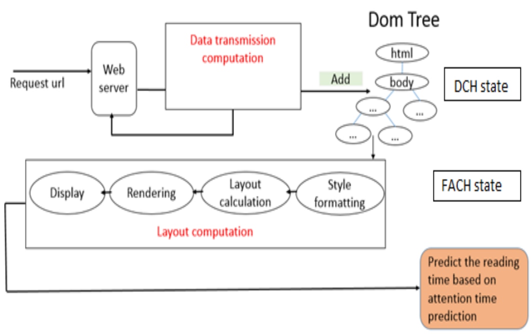

Fig.3.1 Energy aware architecture of web browser

Data transmission computation

HTML, JavaScript and CSS are the three kinds of data transmission sources. Content objects such as HTML files, JavaScript files, images, and flashes, are referenced by

URLs. Therefore, the web browser will fetch and include those objects to the DOM tree as a node. For JavaScript code, it is either transformed to HTML code and then fetches content objects, or it will get content objects from the web server directly. The HTML code is parsed and if there exists JavaScript code, it is sent to the interpreter engine. If it has URLs for objects like images, HTML files, and flashes, these objects are requested.

Layout computation

All data transmission computations are done only when HTML and JavaScript files are processed and all the CSS files are scanned. Then, the layout computation is processed which does not cause data transmission. Parsing is done for 1/3 of webpage content, then a simplified intermediate display is drawn over there. This display does not need CSS rules, style format, or images, a little layout computation is needed. Since it does not need to wait for any information from the CSS files, it can be rendered faster than the normal system.

Attention Time Prediction

Based on the attention time of other previously browsed pages by the user, the attention time prediction of the new webpage is calculated. A new browser plugin is developed that records the time a user spends reading a certain webpage and then automatically send that data to a server. After the user attention time is calculated, repetitive occurrences of the webpage are made before using it for prediction. Once the user's attention times for a collection of documents are predicted, user's attention time of a new document is predicted using document content similarity analysis.

4. METHODOLOGY

It uses the energy aware approach for reducing the power consumption of smartphones and also it uses the attention time prediction to predict the user reading time.

Steps:

1. Begin to open a webpage
2. Data transmission computation is done.
3. Layout computation is done
4. Collect features x = {x1,…. x10}
5. Webpage opens and Waits for seconds.
6. Get Tr from the prediction model with x if (Tr> Td) OR (Tr>Tp AND mode == power) then switch to IDLE state

Prediction model:

1. We denote the training sample set as {tatt (u, di) |i = 1, • • •, n} where n is the number of documents u has read so far, which are represented as di (i = 1, • • •, n).
2. When a new document dx arrives, we calculate the similarity between dx and all the documents in the training set.
3. We then select k documents which have the highest similarity with dx.
4. In our current experiment, k is set as min (10, n), where n is the size of the current training sample set.
5. Finally predict the attention time for dx.

Advantages of our system

- This approach can reduce the power consumption during web page loading.
- This system allows effective usage of back bone network.
- Our solution can reduce the webpage loading time by 16%, and increase the network capacity by 20.6%.

5. CONCLUSION

This approach can reduce the power consumption of smartphone by more than 30% during web browsing. Moreover, our solution can reduce the webpage loading time by 16%, and increase the network capacity by 20.6%.

6. REFERENCES

[1] Bo Zhao, Wenjie Hu, Qiang Zheng, and Guohong Cao (2014) - Energy-Aware Web Browsing on Smartphones‖ , IEEE Transactions on Par-allel and Distributed Systems.

[2] MulikUmesh, Bora Sanket, ChavanAnkita, BandeJyoti (November-2015) -- Energy efficient web surfing for smartphones, International Journal of Scientific & Engineering Research.

[3] Ashwini M. Sonwane, (May 2015) -- Efficient Energy Consumption Approach during Browsing in Android Smartphones, International Journal of Emerging Technology and Advanced Engineering.

[4] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshpande, C. Grunewald, K. Jain, and V. N. Padmanabhan (2010) -- Bartendr:
a practical approach to energy aware cellular data scheduling, University of Maryland ,Microsoft Research India.

[5] L. A. Meyerovich and R. Bodik (2010) -- Fast and parallel webpage layout, in Proc. Int. World-Wide Web Conf.

[6] Songhua Xu,Yi Zhu, Hao Jiangand Francis C.M. Lau (2008) -- A User-Oriented Webpage Ranking Algorithm Based on User Attention Time, Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence.