I started to learn Go (golang) programming language. As a first problem i decided to code "Hello, world!" a tool to extract useful content from a HTML page. How does it work. I use the goquery package to convert a HTML page to well structured DOM tree. Next step is to drop all nodes (tags) from a HTML page that are not useful at all. Like, script, style, head etc. Then for each mode of a tree (starting from a top level node) i repeat a function recursively. The function detects if &#65533;a node has subnodes. If yes, then the function checks how a text is distributed between child nodes of a node. If a text is distributed relatively&#65533;evenly, then current node is what we are looking for and a text from it is returned as a result. If a text is&#65533;gathered mainly in one of subnodes, then a function is executed for this subnode recursively. To detect if a text is distributed evenly to get total length of all text (ignoring HTML) in a node and then calculate a&#65533;Mean Deviation of lengths of a text in each subnode. Having this 2 values i am checking if a Mean Deviation is relatively small to a full text length. If yes , then the text is&#65533;distributed relatively&#65533;evenly. There is the code!&#65533;https://github.com/Gelembjuk/articletext Usage:
package main import ("fmt" "os" "github.com/gelembjuk/articletext") func main() { url := os.Args[1] text, err := articletext.GetArticleTextFromUrl(url) fmt.Println(text) }