Development of methods and means for adapting an electronic interface for people with disabilities

A S Kuznetsova¹, V V Gilka¹, V V Gorbacheva², V L Rozaliev¹ and Duke F CH Okoroji¹

¹ Volgograd State Technical University 28, Lenina Avenue, Volgograd, 400005, Russia
² Volgograd Academy of Physical Culture 78, Lenina Avenue, Volgograd, 400005, Russia

E-mail: agnessakyz@yandex.ru

Abstract. The social policy of the state is for the most part aimed, at people, without any physical deviations. The needs for people with disabilities can be roughly divided into two groups of needs: those characteristic of ordinary citizens and the needs that cause illness. The development of these information technologies is a particularly valuable tool for people with disabilities; they can improve their quality of life. Despite the fact that there is still an acknowledged digital divide, it is now much smaller. For large numbers of people with disabilities, information technology is essential to develop the basic literacy and numeracy skills needed to participate more fully in society. For people with individual learning needs information technology is applicable to integrate into society through the process of socialization and control of unacceptable behavior.

1. Introduction
In daily life, people with disabilities face a number of problems related communicative, psychological characteristics when used with the environment, the availability of necessary physical objects, environment (housing, transport). Most of these problems arise due to social barriers between people with disabilities and society.

The social policy of the state is for the most part aimed, at people, without any physical deviations. The needs for people with disabilities can be roughly divided into two groups of needs: those characteristic of ordinary citizens and the needs that cause illness.

2. System analysis of the problem
The use of information technology is widely recognized as a daily part of the lives of many people and has an impact on many aspects of society, including education, training and employment, the economic prosperity of people and countries, as well as in the media and tourism and leisure. Significant efforts and investments have been invested to maximize the benefits of information technology in education and training in developed and developing countries.

The development of these information technologies is a particularly valuable tool for people with disabilities; they can improve their quality of life. Despite the fact that there is still an acknowledged digital divide, it is now much smaller.
Special education covers a wide variety of educational issues and can reach people of all ages. Some young people need support in order to be able to reach a basic level of communication with the people around them.

For large numbers of people with disabilities, information technology is essential to develop the basic literacy and numeracy skills needed to participate more fully in society. For people with individual learning needs information technology is applicable to integrate into society through the process of socialization and control of unacceptable behavior.

Global trends in meeting special educational needs have emphasized and developed the importance of not exacerbating the "feature" unnecessarily. The philosophy of inclusive or integrative education dominates many national systems - if students can, they should be placed in "regular" schools. It is obvious that such trends can lead to a significant burden on schools and higher education institutions. Many turn to information technology as a promising means of supporting special needs for people with disabilities in physical and (or) mental development, that is, deaf, hard of hearing, blind, visually impaired, with severe speech impairments, disorders of the musculoskeletal system.

State-funded special schools continue to exist, but the influence of philosophy and practice (inclusive education system) and its integration into education has made support for students with special needs a problem for a growing number of teachers (professors), schools, colleges and universities.

The commitment of these groups, however, actually turns out to be frustration arising from the lack of adequate resources, the limited number of staff trained to work with people with disabilities, staff training and competence in the context of special needs is a challenge facing secondary and higher education.

Many schoolchildren and students find themselves in an educational environment where teachers find it difficult to provide them with the individual support they often need. Their own impairments limit their participation, often without compensatory interventions or accessories to help them, lack of academic achievement, and low self-esteem give rise to the exact opposite of an integrative approach - educational and social isolation. In trying to solve such problems, schools, colleges and universities often manipulate their resources to provide specialized pedagogical support in the form of extra classes. However, even with the best of intentions, many of these tricks only highlight the problems of students and, in the end, contradict their desire to match their peers.

Schools, colleges, HEIs are looking for a panacea - it is either a teaching staff that can successfully master all the special needs in regular classrooms or a means by which students with special needs can successfully compensate for their own impairments without the stigma of abstinence and a significantly reduced perception of uniqueness.

It can be argued that information technologies will not only improve educational opportunities, they will open up for students with special needs that side of life that is accessible to students without special needs. Information technologies will also provide an opportunity for the development of education where there are difficulties in participation or, perhaps, teachers are not sufficiently prepared to work with students with special needs.

Some students with special needs suffer from physical or medical disabilities that limit movement and coordination. For example, students with cerebral palsy from birth or those who have suffered from spinal injuries often experience difficulties that can range from poor control of limb movements to complete lack of movement. Certain conditions negatively affect the student's ability to speak. Students with these types of problems may be highly capable intellectually, but are unable to perform physical functions such as speaking, writing, or other practical activities that underlie basic interaction and therefore learning and achievement. Many of them, however, will have some control, even very limited, over the limb, and great success would be achieved by using physical means (prosthetics) to interact with the computer. Dedicated keyboards, trackballs, switching systems, etc., allow the student to select operations on the computer (for example, choosing from a set of characters or on-screen symbols for communication) using the head, leg, elbow, or mouth, etc. An oculograph for tracking eye and face movements can be used as an indicator of intention and choice of actions on a computer.
People with profound or severe multiple disabilities may not be able to express emotions (such as laughter) or gestures (such as smiling) that might suggest a basis for communication and interaction. This severely limits their ability to participate in interactive learning opportunities, even with individual mentors who are familiar enough with their physical activities to discern transmitted signals.

Research in schools and universities shows that, according to teachers, about 80% of students with special needs do not have communication skills. They suggest that the low expectations resulting from such perceptions often limit the efforts of teachers or lecturer to engage students in the interactions that are necessary for education. Computers, however, open up opportunities for these students through their interactions with a multisensory environment. With the help of computerized switching technology, students can begin to learn very simple management and communication skills, which in turn can form the basis for higher order learning.

Much of learning depends on interacting with objects or other people, and sensory impairments such as partial or complete blindness or partial or profound deafness seriously affect the student’s ability to learn in the normal way. Quite a large number of pupils can suffer from visual and hearing impairments, and if they are not compensated at an early age, their acquisition of literacy and numeracy skills can be drastically undermined.

Obviously, this, in turn, may lead to slower progress than would be expected from other capable students. Hearing impaired students can be helped in a variety of ways through information technology. Information technology also plays an important role in enabling visually impaired students to compensate for some degree of poor or no vision. For some, this will mean a more accessible presentation of existing materials, while for others it will mean more effective means of translating ordinary visual information into speech or Braille.

Microsoft is an example of a major software company that has begun to take care of the needs of visually impaired users. Most of their applications now come with accessibility features, including keyboard customization tools for one-handed users, enhanced toolbar icon tools, enhanced text presentation, and accidental keystroke and repetition ignoring.

For such a range of information technology tools are available, including speech output packages for computers and text scanners, electronic braille recorders as well as scanners that enlarge printed texts and screen magnification systems.

3. Materials and methods

It is important for a user with disabilities to be able to quickly and intuitively access their history of navigating through hyperlinks from one page to another. The project described in article [19] builds a graphical map from the addresses visited by the user, with a control system and time of visit. This tool is designed for easy navigation for people with musculoskeletal disorders. The system is browser independent and uses a proxy server. The proxy server stores information in its cache that the client has access. If the user does not access some Internet page, this page is removed from the cache as unnecessary.[1]

The history is presented in the form of a tree, where each node is the URL of the page that the user visited. The user can also selectively display the headers of the nodes. When the computer mouse is in a tree node, the title and URL of the page are displayed next to the mouse. User can revoke the document with a double click. The user can save the map built during the navigation session. This project can be considered as an addition to standard web browsers. It is a solution to the problem of navigation and accessibility of websites.

The most popular and accessible is the "universal design" of a web resource, that is, in the case when the Internet resource is already adapted for people with disabilities, you do not need to use third-party programs other than an Internet browser to view it. Accordingly, the number of users of the site includes people with disabilities [2].

In order to determine the strategy for adapting the site for people with disabilities, it is necessary to accurately determine whether the problem of using the site is related to health limitations.
The architecture of the program includes two stages: parsing site styles and parsing site links. The first step is to parse style elements, create an equivalent style, and replace the source of the CSS file. At the second stage, the links are classified, if the link does not match the requirements for it, it is removed.

The main problems in Internet surfing for people with disabilities are reading information (for the visually impaired, the elderly, people with developmental retardation) and entering information from the keyboard (using a computer mouse) (visually impaired, the elderly, people with impaired coordination of movement).

The solution to the problem is to provide information in a simplified form and in an alternative form: a text message in the form of a voice message, an image using a text description. For convenient control of the Internet browser, it is required to minimize the user's interaction with the control devices familiar to the average user (keyboard, computer mouse). This problem can be solved by implementing voice control for the Internet browser.

There are several approaches to finding an article on the site. The following parsing approaches are distinguished:

1. Data extraction based on HTML document (text level). This approach includes three stages:
   1. Highlighting the code for the original web page;
   2. Extract required data from HTML code. After receiving the page code, the page should be processed, that is, to separate the text from the markup;
   3. Saving the obtained result. The data is recorded in a database or in a separate text file.

2. Extracting data based on a rendered document using computer vision. This algorithm is the most accurate, but quite complex and costly and includes two stages:
   5. Page visualization (rendering). At this point, the web page and site model is converted to an image.
   6. Extract the required data. Based on the algorithms of computer vision, content is searched for on a web page, in particular, pictures and their position on the page, text.

3. Extracting data on the basis and at the site level, when comparing pages of the same type and highlighting the differences between them (the blocks that will differ are the required content). This algorithm is used by most search engines.

Using formulas, tables or additional applications, you can detect the required piece of code on the page. This section will contain all the necessary information. Similar parts of the code are repeated on pages that have a similar structure. This approach allows you to automatically select and import all duplicate data. This classification of methods is at Figure 1.

![Figure 1. Approaches to finding an article on a web page.](image)

The most efficient approach is to extract data based on the HTML document structure. It is quite simple to implement and does not depend on the structure of third-party sites.

Since the sites are difficult to break into specific blocks due to the individual approach to the implementation of the site (the blocks are named differently on each site), it is supposed to give recommendations to sites wishing to work with people with disabilities (not specialized sites). For this, the developers will be supplied with a CSS file with the names of the classes. The developer himself will decide how his site will be implemented. In this browser, the original CSS file will be replaced.
The browser will ignore unnecessary blocks (for example, ads). In order to get rid of unnecessary blocks on the web page, blocks that are not described in the developed CSS file will be deleted.

The main elements of the page will be highlighted using the following effects: increasing the font, text contrast. Advertising and third-party content will take a secondary position on the web page.

Since dividing web resources into blocks is a rather laborious task. In the source file, each main block on the site is assigned a class value. The CSS file with the class names will be supplied to the developer. The developer himself will decide how his site will be implemented. The application will replace the original CSS file with a one designed for people with disabilities.

The application will rebuild the page according to the following standard:
- increase the font;
- removal of third-party images that do not carry a semantic load;
- change the color of the text of the main information to black;
- change the page background to white;
- changing the color of links on the page to blue, links will also be underlined;
- the site title is highlighted in bold text;
- the ad unit moves to the bottom of the page.

When you change the page font four times, the page structure is preserved.

In order to determine the strategy for adapting the site for people with disabilities, it is necessary to accurately determine whether the problem of using the site is related to health limitations.

A study was conducted in which 10 healthy people and 10 blind and half-blind people participated in the study. Both groups had equivalent socio-biological characteristics. Each group included 5 experienced internet users and 5 newbies.

The study made the following conclusions:

a) All users experienced difficulties with “overloaded” pages.

b) Important information should be at the top of the page.

c) It is necessary to keep the navigation on the site, and to warn the user with a sound signal, if they click correctly or incorrectly.

Method of parsing an article based on the distribution of letters on the site

This method is based on the density of distribution of Russian letters in the HTML code. Accordingly, this method works only on Russian-language sites.

The body of the article is detected as follows: the number of Russian characters in the article is counted, then a graph of the density of distribution of Russian characters in the article is built. Then, by calculating and constructing the normal distribution (Gaussian distribution), we can guess which part of the HTML page is more likely to be considered content.

The title of the article in this method is the <title> meta tag. This meta tag is extracted from the HTML code. If it is not present, it is considered that the article has no title.

After defining the content area, all extraneous elements that are not text, such as Java scripts, comments, iframe, are removed in the found area. Tables are also deleted from the body of the table article. Pictures are not removed from the content area.

Method for highlighting an article on a website.

The use of several methods at the same time in a certain order allows you to recognize an article on the site with one hundred percent accuracy, regardless of the structure of the site.

At the first stage, all elements that participate in the page structure (for example, a div) are selected, with the text contained in them, if any. The main task at this stage is to get such a flat list: DIV element -> text in it.

In the case when an element has nested elements (the presence of nested div elements) their content is discarded. The nested div elements, which are also children, will be examined and processed in turn.

This method is based on one important assumption. Elements that are semantically intended to mark up a structure (div) are only used to mark up the structure, not to mark up paragraphs. Otherwise, only the largest paragraph of the article will be highlighted. For the top 100 news and education sites, this assumption is correct.[2]
After this selection, a set of elements is obtained in which you need to recognize the article. Further, using various algorithms, each element will be assigned a coefficient of the probability of having an article.

For all elements in all elements of the DOM (Document Object Model) tree, there are elements that have repeating patterns in their attributes (for example, class or id). There are two types of nodes in the DOM tree: element nodes and text nodes. Element nodes are formed by tags. Accordingly, some nodes will be nested within others. Exclusively these nodes form the very structure of the tree.

Test nodes (text nodes) are formed by texts within elements, which are designated as #text. A text node must contain only a line of text; it is always at the lowest level and has no children.

A repeating pattern has two meanings in this context:

a) the same set of class elements
b) the same substring of text in id.

All such elements, including those of descendant elements (nested elements), and those elements that are included in the set for must be pessimized, that is, they are assigned a certain reduction factor, which in turn depends on the number of repetitions found.

In the site menu and in the columns located on the site, most often there are only links (internal - links for navigating to your own resources or external - links to third-party sites). Consequently, the content of these blocks is clearly not an article on the web page.

All similar elements that will be found on a given web page are given a pessimizing coefficient.

The markup in blocks is just as important when searching for the body of an article. The higher the number of different types of markup in a block (text wrapping, list, span), the less likely this block is an article. Advertising is a good example. Advertising is not at all like an article, as it is entirely composed of a list. The ratio of the layout of the text to the amount of text is important. The lower this ratio, the more likely it is that this block of text is an article.

Another excellent metric is the number of dots (i.e. sentences) in a given block of text. As a rule, in headings, menus, lists, periods at the end of phrases are not put. In the body of the article, there are many times more sentences completed with a dot. From these considerations, we can conclude that the more dots in a block of text, the more likely it is that this block of text is an article.

Counting dots in a block of text works great as a checklist for menus and various lists. If these blocks pass the previous checks, then they will be more likely eliminated at this stage.

One of the main methods, and the final step in parsing an article, is comparing the length of the text in different blocks. Counting the number of blocks that contain text of the same length is important. A large number of blocks with text of approximately the same length is a sign of the absence of an article in these blocks. Therefore, it is easiest to select the block that contains the most text. While this method works 90 percent of the time, it is not as perfect as it sounds and has a distinct disadvantage in special cases.

For example, in the case when the comment to the text of the article has a size commensurate with the article. To avoid such special cases, this algorithm is used at the final stage after unwanted elements such as comments are pessimized, for example, for repeating patterns in id or in a class using other algorithms.[4]

At the output, after each process there are some results that are processed further. In the figure, the following outputs are numbered:

- structured groups of extracted elements;
- flat list of items (excluding children);
- groups of elements with an item availability factor;
- a group of element nodes and text nodes;
- elements without repeating patterns;
- structured by the number of links inside, a group of elements;
- elements without blocks, with the same markup;
- structured by the number of points, a group of elements.

The output of the whole algorithm is the body of the article.
Figure 2. The order of methods for highlighting an article on the site.

Development of models and methods for providing content in an alternative form (text content in audio format and vice versa, graphic image in text format). Web accessibility is directly dependent on the tools used to develop the website and on assistive technologies. By using the HTML markup language, you can create a web page layout that adheres to established standards.

Assistive technologies include screen readers, magnifiers, alternative keyboards, and more. The most used is a screen reader. Such programs read text from a web page in a linear fashion. One of the ways to adapt web resources for people with visual disabilities is presented in the article. The architecture of the program includes two stages: parsing site styles and parsing site links. The first step is to parse style elements, create an equivalent style, and replace the source of the CSS file. At the second stage, the links are classified, if the link does not match the requirements for it, it is removed.

Approaches to finding an article on a web page

There are several approaches to finding an article on the site. The following parsing approaches are distinguished:

a) Data extraction based on html document (text level). This approach includes three stages:
- highlighting code for the original web page;
- extract required data from HTML code. After receiving the page code, the page should be processed, that is, to separate the text from the markup;
- saving the obtained result. The data is recorded in a certain database or in a separate text file.

b) Extracting data based on a rendered document using computer vision. This algorithm is the most accurate, but quite complex and costly and includes two stages:
- page visualization (rendering). At this point, the web page and site model are converted to an image.
- extract the required data. Based on the algorithms of computer vision, content is searched for on a web page, in particular, pictures and their position on the page, text.

c) Extraction of data on the basis and at the site level, when comparing the same type of pages and highlighting the differences between them (the blocks that will differ are the required content). Most search engines use this algorithm.

Using formulas, tables or additional applications, you can detect the required piece of code on the page. This section will contain all the necessary information. Similar parts of the code are repeated on pages that have a similar structure. This approach allows you to automatically select and import all duplicate data.

The most efficient approach is to extract data based on the html document structure. It is quite simple to implement and does not depend on the structure of third-party sites.

Classification of web portals based on latent semantic analysis. Most of the information on Internet portals is unstructured data. These are the data that users interact with. As a solution to this problem, there are portals, programs that classify the information presented and automatically group the data, making an abstract of these articles. A prime example is the Yandex News portal.

Good input data for latent semantic analysis are news portals, which do not differ much in structure from educational portals. The article has the following structure: heading - contains the topic of the article, consists of one small sentence, basic information - includes the main topic (revealed in the first two paragraphs).

To solve this problem, it should be built a thematic model. The thematic model describes the belonging of a document to a certain topic. Mathematically, the task is reduced to the parallel classification of a number of documents on the same set of topics. When building a given model, some designations and assumptions are formalized. Let's denote by D a set of documents that contain a set of documents d, and n is the number of required documents in the set (1).

Where D - a set of documents, d - a set of documents, n - the number of required documents.

D={d_1,d_2,d_3,….d_n}
(1)

Each document d contains some set of words Wd (2). Where d - a subset of documents, Wd - a set of words.

W_d={w_1,w_2,w_3,….w_d}
(2)

Documents d also have a given topic t, which is included in the topic subset T (3). Where d - a subset of documents, t - a subset of topics.

T={t_1,t_2,t_3,….t_n}
(3)

The value of the matrix (Formula 4) will be the thematic model. Where Φ - the value of the matrix, p - the distribution of documents, w - the set of words, t - the topic.

Φ=lp(w|t)Φ=lp(w|t)
(4)

One of the first approaches in dealing with the structure of an HTML page is to build a DOM (Document Object Model) tree. DOM is the object model used for XML / HTML documents.

According to this model, each document is a hierarchy, a tree. Each tag in an HTML document forms an element node. The tags that are nested in this node are child nodes. The text is represented by nodes of the "text" type.

The DOM tree represents a document as a tree object that can be modified using scripts. Figures 11 and 12 show the appearance of the page source code and the DOM tree of this page.
Both types of nodes are highlighted in this tree:
1) tags that form nodes elements. Some nodes contain other nodes. Such a tree structure is formed exclusively due to such nodes.
2) text nodes forms the text located inside the nodes. In the figure, they are designated as #text. A text node can only contain a line of text. This node cannot have children and is always located at the lowest level in the tree structure.

When processing an incorrect HTML document, the browser automatically corrects it for correct display. The same when building the DOM tree. For example, the top <html> tag will always be completed in the tree. If the HTML file contains only one word, the browser will place it in the <html> and <body> tags.

When constructing a DOM tree, the browser handles minor errors, such as closing tags.
According to the standard, all tables must contain the <tbody> tag, but if there is no such tag in the table, the browser adds this tag on its own. The DOM also displays comments that do not affect the appearance of the HTML document. In total, the DOM distinguishes between twelve different node types. In dealing with the text selection method, only elements (the main site structure) and text nodes
are taken into account.  

*A method of highlighting text based on surface analysis.*

In addition to the actual content, all web pages include navigation elements, some templates or ad units. These templates are usually not related to the main content, which degrades the accuracy of text searches. This approach is based on two features. The first is the word count and the second is link density. These functions correspond to stochastic textual models that are embedded in the field of quantitative linguistics.

Approaches to text pattern detection are based on the use of the segment characteristics of the DOM tree, which includes frequently used segments or blocks on the site. Using a combination of different approaches, it is possible to increase the percentage of discovered articles up to 40-50 percent. Simplify the structure of the DOM tree for building and displaying a site style tree, which is then used to build a solution and reduce entropy when a template is found at the level of the DOM tree. Pattern detection is inextricably linked to web page segmentation. After that, a static model is built at the segment level, using the density coefficients of the text of the subsequent blocks to determine the page levels.

This algorithm uses a machine learning method to build classifications. The method detects the largest contiguous text area with the fewest HTML tags and marks it as "full text". This method is based on one assertion that the density of tags in the document template is higher than in the full-text content, that the main content is longer than the text content. Using this approach, a representative set of features is analyzed to automatically classify patterns. One of the incentives for using this algorithm is that with the help of such algorithms, the search on the Internet is improved (removing stop words, determining the nausea of the test).

A combination of functions that are used to classify segments of web pages are used to identify snippets of text as full text title, listing, text navigation. These approaches work at four different levels: individual text blocks (elements), a full HTML document (a sequence of one or more text blocks, taking into account structural information), a rendered image of the document, and a complete website. The first two levels can be parsed locally for each document, while the next two levels require external information, such as images, CSS, and the style of the web page.

Segmentation of web pages and classification of text within a document is based on the use of structural elements on web pages, such as individual HTML tags, certain classes, CSS styles. The more CSS is used on a web page, the less important the HTML structure is. To avoid this specific problem, which can lead to overshooting of a certain set of data, as well as an increase in page processing time, only the following structural features are considered. There are the presence of a specific heading tag (H1, H2, H3, H4, H5, H6), paragraph tag (P) and an anchor section tag (A) as an HTML element that encloses a specific text block.

Since the definition of a template checks the text not at the text level, but at the functional level, this method does not consider words separately as signs of classification. Text objects are treated at a higher level that is independent of domain and specific language. Average word length, average sentence length (sentence boundaries are determined by a simple pattern-based heuristic check for full stops, question marks, exclamation marks, periods), and absolute word count. Another important source for the classification task is the local context, that is, the absolute and relative position of the text block in the document. If the granularity of the segmentation is high, it is likely that the full text is followed by the full text and the pattern is followed by the pattern. In addition, when there is a significant amount of text template, the main content is usually surrounded by the template (header, footer, left navigation, right navigation, and the like), and not vice versa (i.e. even if the last text block contains a sentence, if it is a copy or some kind of error, which are not considered template).

This algorithm considers several heuristic features. The first is the absolute number of words that start with an uppercase letter or are all capitalized. And the next is the ratio of these words in comparison with the total number of words and the ratio of full stops to the total number of words, the number of markers associated with the date, time, special characters such as a vertical bar (these characters are found in the text of the navigation template). The density of links (in percentage) is also
calculated.

The density of the text for each specific block is also estimated in addition to the density of the channel. A measure of text density is used to segment web pages.

Calculation of text density is in Formula 6. Text density $\delta (b)$ calculates the number of tokens $T(b)$ in a specific text block $b$, divided by the number of lines $L(b)$, covered after text wrapping with a fixed column width $w_{max}$ (an empirically estimated optimal value [40]). Due to the side effect of the unfilled last line, the last line is ignored unless it is the only line in the segment. Where $T(b)$ - the number of tokens, $\delta (b)$ - the density of the text, $b$ - the block, $L(b)$ - the number of lines.

$$T'(b) = \{ t \mid t \in T(l), l_{first}(b) \leq l < l_{last}(b) \}$$

$$\delta(b) = \begin{cases} \frac{|T'(b)|}{|L(b)|}, & |L(b)| > 1 \\ 1, & |L(b)| = 1 \end{cases}$$

Web pages are segmented into atomic blocks of text, which are annotated with characteristics and based on these, classified by content or pattern. Atomic text blocks are sequences of character data separated by one or more HTML, with the exception of tags like <a> - for calculating link density.

| Class                      | Block | Words | Tokens  |
|----------------------------|-------|-------|---------|
| Total                      | 72662 | 520483| 644021  |
| Boilerplate                | 79%   | 35%   | 46%     |
| Any content                | 21%   | 65%   | 54%     |
| Heading                    | 1%    | 1%    | 1%      |
| Full text of the article   | 12%   | 51%   | 42%     |
| Additional text            | 3%    | 3%    | 2%      |
| User comments              | 1%    | 1%    | 1%      |
| Related content            | 4%    | 9%    | 8%      |

The distribution of classes is dominated by full-text and template (Boilerplate), the remaining four classes quantitatively play a secondary role, as shown in Table 1. User class comments were only scored to quantify the comment text relative to the rest of the full text. Due to the highly distorted distribution of the original six text classes, the emphasis is on dividing the classes into two similar groups. The first includes the template and content, the second contains the title, full article text, additional text, user comments, related content. The four-class problem is generally more difficult to solve, so a lower margin of error is expected.

Weka explores the increment of information for each function and evaluates machine learning classifiers based on decision trees. The classification accuracy is measured by evaluating the recall accuracy, the false positive rate coefficient and the dimensionless point curve. All scores are normalized based on the number of words in a block, that is, large blocks are weighted higher than small blocks.

As you can see above, the distribution of classes is dominated by full-text and template (Boilerplate), the other four classes quantitatively play a minor role. User class comments were only scored to quantify the comment text relative to the rest of the full text. Due to the highly distorted distribution of the original six text classes, the emphasis is on dividing the classes into two similar groups. The first includes the template and content, the second contains the title, full article text, additional text, user comments, related content. The four-class problem is generally more difficult to solve, so a lower margin of error is expected.

4. Implementation of the program prototype

After conducted studies of the lead-tin-base bronze structure of the BrO10S10 grade, it is possible to conclude that the modification with superdispersed powders influences considerably its structure. It is particularly typical for low concentrations of the modifier (up to 0.25%).
Extracting information from Internet resources using Ruby. Internet resources are not limited to serving real users. Many popular websites support an API through which a computer program interacts with a website, for example, to collect some information and statistics. Web scraping (screen scraping) is a technology with the help of which HTML pages are converted at the syntactic level into more convenient forms for consumption. It is possible to retrieve structured information from Internet content using APIs.

There are several common web scraping approaches. Some of these approaches convert HTML to JSON, making it much easier to retrieve data from a page. Some solutions allow you to define the presented content as an HTML hierarchy with structured data. Nokoqiri's solution allows you to parse HTML and XML documents using the Ruby language. The pjscreape's solution is used for JavaScript and is command line based. This solution can parse the received web page, including the JavaScript code of this page. Beautiful Soup's solution is used for Python and integrates well with the Python environment.

Let's say it is required using the Nokogiri tool, it is required to use web scraping to determine some metric from the database, which is reflected in the HTML page. The data is located inside the HTML <span> tag, with a specific ID for the tag. In this method, the <span> tag will be requested and the resulting content from the processed link will be returned.

CrunchBase provides a REST API that allows you to access more required data than the previous tool. In this method, the requested URL is determined, after which a GET request from the HTTP class is used to receive the response. The received response is parsed in the form of a JSON object, and the element is accessed directly using the data structure of the Ruby language.

There are advantages to using the API in each case. When working with scraping, you don't have to go deep into the HTML code to get the structure and highlight the data for processing. With the use of Nokogiri technology, it is quite easy to find the necessary data in the structure. But in the event that the structure of the HTML code is changed, a significant modification of the script may be required in order to adapt it for parsing the new structure of the document. The API-based approach works well as long as the contrast of the API is substantial and well-visible.

Another important advantage of this approach is that the base of the interface allows you to access all the data that is accessed through the interface through a JSON object that is returned upon request. The amount of data that a website provides via HTML is much less.

This program is implemented in the C# language. It uses third-party libraries: Nreadability, Microsoft Speech Platform - Runtime, Microsoft Speech Platform - Runtime Languages, Microsoft Speech Platform.

In accordance with the plan of the technical task, the developed program performs the following functions:
- go to the URL address;
- highlighting the title of the article and the body of the article;
- change the color scheme of the article;
- voice playback of websites.

The program should detect an article on the web portal and only display it when you enable "Article Mode". Voice reproduction of the article is user-defined.

The selection of the title and body of the article is done using the library Nreadability. To do this, a block of text is selected from the website, which is presumably an article, the title of the web page is selected (it usually coincides with the title of the article on the site). If the block contains images, the image data will also be included in the article. Voice playback of websites is done using the Microsoft Speech Platform. Voice reading is carried out from the first block of text on the site and further in order. Changing the color scheme occurs by changing the CSS of the source file.

Article mode is available only for those sites that have recognized a block of text that meets the requirements of the article (the requirements are described in the article search methods). Text voice playback is available for all websites containing blocks of text.
5. Conclusion
Accessible technology plays an important role in engaging students with disabilities, both educationally and socially, by embracing disability and ensuring that students are not left behind or left behind. This enhances learning opportunities for all students, not just students with disabilities. There are some barriers to the technology available, but sometimes solutions can be found.

This paper provides a systematic analysis of communication problems for people with disabilities, a systematic analysis of problems, opportunities for their solution. Federal and world bills were considered to improve the quality of life for people with disabilities. The classification of nosological groups of people with disabilities and the distribution of information technologies that facilitate the use of a computer for each nosological group are also considered.

To solve this problem, it was decided to implement an Internet browser with the "reading mode" function, which allows you to recognize the title of the article, the body of the article on the site, highlight them, thus getting rid of unnecessary content on the page for comfortable reading and navigation. This program also allows you to provide text content in an alternative voice form.

6. Acknowledgments
The reported study was conducted with partial financial support by RFBR and the administration of the Volgograd region, projects number: 19-07-00020, 18-07-00220, 19-47-343001, 19-47-340009, 19-47-340003, 19-47-340013

References
[1] Law "On Ratification of the Convention on the Rights of Persons with Disabilities" http://kremlin.ru/events/president/news/15196
[2] Orlova Y A 2016 Models and methods of information support for communication of people with disabilities p 352
[3] Rozaliev V, Guschin R, Orlova Yu, Zaboleeva-Zotova A and Berdnik V 2019 The method for searching emotional content in images based on low-level parameters with using neural networks Advances in Intelligent Systems and Computing 848 260-265
[4] Rozaliev V, Nikitin N, Orlova Yu and Zaboleeva-Zotova A 2019 Results of using neural networks to automatically creation musical compositions based on color image Advances in Intelligent Systems and Computing 875 148-157