# Automating Hierarchical Subject Index Construction for Scientific Documents

Elena I. Bolshakova[1] , Kirill M. Ivanov[2]

[1] Lomonosov Moscow State University,
National Research University Higher School of Economics, Moscow, Russia
eibolshakova@gmail.com
[2] Yandex, Moscow, Russia
ivanov.kir.m@yandex.ru

**Abstract.** Subject, or back-of-the-book index consists of significant terms with relevant page numbers of the text document, thus providing an easy access to its content. The paper describes methods developed for automating main stages of subject indexing for specialized texts: namely, term extraction, selection of the most important ones, detecting their reference pages, as well as recognizing semantic relations among selected index terms in order to structure them into hierarchy. The developed methods are intended for processing scientific documents in Russian and are based both on formal linguistics rules and unsupervised machine learning. Experimental evaluation of the methods have shown their sufficient quality to be built into computer subject indexing system.

**Keywords:** back-of-the-book indexing, hierarchical subject index, linguistic patterns and rules, automatic term extraction, recognition of term relations

## 1 Introduction

Subject, or back-of-the-book indexes are intended for reading large and medium-size text documents such as books, manuals, etc., especially in highly specialized domains. Typical subject index contains specific terms from the corresponding document, with reference pages, thereby facilitating navigation through the text and locating needed information. Such indexes are especially useful for readers of educational texts in difficult scientific and technical areas (textbooks, manuals, tutorials, etc.), since they represent key concepts of the text and also makes it easier to repeatedly read term definitions and other important fragments of texts.

To now, automatic back-of-the-book indexing is a little-researched area, although the first papers appeared long ago [17]. The main reasons are related with the complex nature of the problem and complexity of its subtasks. Nevertheless, the automation of these tasks is needed, because the high-laborious indexing work remains mainly manual, and useful subject indexes are absent in many modern textbooks and manuals, in particular, in texts of rapidly developing scientific and technical fields.

Among few works on automatic subject indexing, the most [7-9, 15] address extracting terms from a given text document, which is the central problem of index con-

struction. The statistics-based and machine learning methods proposed in [7, 9, 18] showed low precision and recall for term extraction (about 27-28%). The works [9, 14, 20] are mainly based on linguistic rules for term extraction, but do not provide proper evaluation of developed methods.

The other tasks of subject index construction are less investigated, including selection of page numbers relevant for reference and revealing subordinate relations of terms to form hierarchical indexes. Certain decisions of these tasks are implemented in two subject indexing systems: InDoc [20] and commercial system TExtract[1], which are oriented to English or French texts.

It should be pointed out that any subject indexing system will inevitably be semi-automatic, since there are no standards on structure and content of indexes, and the work of human indexer may be highly subjective. Another reason is insufficient accuracy of applied techniques from artificial intelligence and natural language processing. Therefore, the resulting index needs to be validated and edited by author of the document or expert in problem domain.

The main objective of our work is to propose a combination of methods and to study their applicability for automating construction of subject indexes for scientific texts with their reach terminology. Our approach is characterized by the following.

- Subject indexing is considered as complex problem comprising term extraction, selection among them of the most important ones, recognition of their semantic relations, and also detecting their reference pages. According to specificity of each subtask, we apply rule-based or machine learning techniques.
- Since widely-used statistical measures developed for corpus-based terminology extraction [13, 19] perform poorly for individual documents [16], our term extraction techniques, as well as a method for recognition of subordinate term relations are mainly based on formalized linguistic patterns and rules similar to [12]. For the other indexing subtasks we propose unsupervised machine learning, namely, clustering extracted terms and their occurrences in text.
- Our subject indexing methods are aimed to processing scientific documents, mainly educational texts containing many specific terms with their definitions. They account for various terminological features of scientific texts including typical contexts of their usage, thereby achieving efficiency of the methods.
- In contrast to most works [7-9, 15, 18, 20] dealing with indexing English or French documents, we consider texts in Russian. The developed rule-based methods continue our previous researches [2, 3], they are close to those in [9, 20], but are performed for Russian scientific texts. As a result, a representative set of rules with lexico-syntactic patterns of terms and their contexts was created.
- In order to improve index term detection (comparing with [1,7,8]), we have elaborated a selection procedure accounting for various factors of terms importance.

The present paper develops our recent work [4] by refining the selection procedure and by proposing methods for the other subtasks of constructing subject index.

[1] http://www.texyz.com/textract/

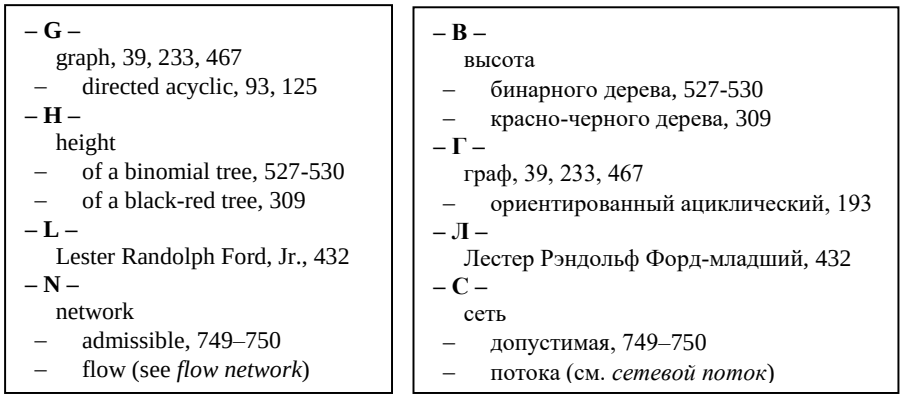
To implement the rule-based methods, we have exploited LSPL formal language [2] and its programming tools[2]. For evaluating the methods, we took several Russian medium-sized scientific texts, mainly on programming. For index term extraction and selection and for detecting reference pages, the developed methods were evaluated separately, since the methodology for evaluating the whole combination is unknown.

The experiments have showed rather good performance of them, in average 70-80 % of precision and recall for index term extraction, which exceeds the results of early statistics-based and machine learning methods [7, 8] and also of the recent one [1]. Overall, our methods are suitable for computer-aided subject indexing system.

The paper starts with explanation of back-of-the-book index structure and description of main tasks (stages) of its construction, and along the way a short overview of corresponding methods developed in related works is given. In the next sections, the methods proposed for all the tasks are sequentially considered and described, with experimental evaluation for the most of them. Finally, the conclusions are drawn.

## 2 Problems and Stages of Back-of-the-Book Indexing

Fragments of typical back-of-the-book indexes are presented in Figure 1. They contain index entries with specific terms from the text document (e.g., *graph*), proper names, and names of objects of the problem domain (such as *Lester Randolph Ford*). Index entries are associated with page numbers and page ranges that serve as pointers to important occurrences of the terms and names in the text.



**– G –**
graph, 39, 233, 467
– directed acyclic, 93, 125
**– H –**
height
– of a binomial tree, 527-530
– of a black-red tree, 309
**– L –**
Lester Randolph Ford, Jr., 432
**– N –**
network
– admissible, 749–750
– flow (see *flow network*)

**– В –**
высота
– бинарного дерева, 527-530
– красно-черного дерева, 309
**– Г –**
граф, 39, 233, 467
– ориентированный ациклический, 193
**– Л –**
Лестер Рэндольф Форд-младший, 432
**– С –**
сеть
– допустимая, 749–750
– потока (см. *сетевой поток*)

**Figure 1.** Fragments of Subject Indexes

Many of subject indexes are hierarchical, as the examples in Figure 1. Such indexes contain entries-headings representing generic concepts (e.g., *height* ) and subheadings (*height of a binomial tree*) that correspond to more specific concepts or particular objects. Such subordinate link between headings and subheadings often indicates generic-specific semantic relation of terms (hyperonym and hyponim term).

[2] http://lspl.ru/

Subject indexes may include cross-references (e.g., *see flow network*), which present synonymy semantic relations between terms.

The process of automatic construction of subject index for a given text document generally comprises 4 stages (tasks) [9, 15, 20]: 1) extracting single-word and multi-word terms by applying linguistics and statistic criteria; 2) selecting the more appropriate ones among extracted terms; 3) detecting semantic relations of the selected terms and structuring them into hierarchy; 4) constructing pointers to important locations of index terms in the text (page numbers).

The first stage produces only a flat list of words and word combinations. The standard term extraction techniques [13, 19] based on linguistic features of terms (grammatical patterns) and statistical measures of word occurrences do not guarantee extracted units to be true terms (e.g., non-term phrases of general lexicon like *key idea*), so resulted units are considered as *term candidates* and needed to be filtered.

The filtering task is usually performed by evaluating and ranking the extracted term candidates with the aid of certain statistical measures (see [10, 13, 19]) and discarding the worst ones. The previous works on subject indexing based on such methods (even with machine learning) gave quite low precision and recall, about 27-28% [7]. Analogous techniques are applied for similar task of keywords extraction (terms denote concepts of problem domain, while keywords may be non-terms but represent main topics of the document) and also give low scores: the best reported in [11] are 35% of precision, 66% of recall, 45.7 % of F-measure. The recent term extraction method [1] based on grammatical patterns of terms along with terms clustering shows 35-67% of F-measure (precision 21-51% and recall 90%) in experiments with software requirements documents, which is also not effective enough for our applied task.

Therefore, for reliable index term detection we propose to sequentially select index terms from term candidates pre-extracted by lexico-syntactic patterns, making use of various term importance factors and also C-value termhood measure [10] performing well for texts in highly technical domains [16].

To hierarchically structure the selected index terms, headings and corresponding subheadings (and cross-references) are to be identified, which can be done in various ways. The only work [20] that proposed the way to automatically recognize generic-specific relations applies structural linguistics patterns similar to those in [10], as well as lexical similarity of multi-word terms that have common words (e.g., *acyclic directed graph* and *directed graph*). Our method exploits lexico-syntactic patterns for detecting such subordinate and also synonymy links, as well term clustering based on certain similarity measures, in order to reveal additional index terms and their semantic relations. Similar clustering techniques proposed in the paper [1] was specifically used for construction glossaries and differs from ours by similarity measures.

The last stage of index construction implies identifying important term occurrences truly relevant for indexing. We should note that importance and relevance may be understand in different ways [5], thus giving various automatic methods. The work [9] proposes text segmentation and selection of the most frequent terms in all segments, but the method was not evaluated. Our decision of the problem accounts for density of term occurrences in the text by clustering page numbers, and we have evaluated the proposed method.

## 3 Term Candidates Extraction

For our tasks, the collection of LSPL rules from the work [3], which encode linguistic information on structure and typical contexts of terms in Russian scientific texts was revised and supplemented, in order to extract from texts a more wide set of term candidates. The resulted set of rules encompasses three groups with various lexico-syntactic patterns:

- rules that specifies typical grammatical patterns of one- and multi-word scientific terms, by indicating part of speech of words (POS) and their grammatical characteristics (case, gender, etc.);
- rules formalizing typical contexts of definitions for new terms (*author's terms*), which are often encountered in scientific and educational papers (so defined terms are certainly be included in subject index being constructed);
- rules specifying typical contexts for introducing terminological synonyms and abbreviations (including synonyms for author's terms).

The first group includes, in particular, rules with grammatical patterns *A N* (e.g., *связный список – linked list*) and *N1 A N2<c=gen>* (*высота бинарного дерева – height of a binomial tree*), where *N, N1, N2* are nouns, *A* is an adjective, and *N2* is specified in genitive case (*c=gen*). For the second pattern, the extraction rule is:

*N1 A N2<c=gen> <A=N1> (N1) => A #N1 N2<c=gen>*

where the adjective of the first noun are grammatically agreed (*A=N1*). Symbol => denotes extraction of the recognized phrase (text item), in accordance with the pattern in the right-hand side of the rule (the sign # denotes lemmatization of the first noun).

The second group covers most of typical Russian-language phrases-definitions of terms in scientific texts. The rules include both particular lexical units (e.g., verbs *понимать, определять – mean, define*) and auxiliary pattern *Term* denoting phrase with grammatical pattern specified by the first group of rules. For example, the definition phrase *...под термином изменение климата будем понимать... (...under the term climate change we will mean ...)* is detected by the following rule:

*"под термином" Term<c= ins > "будем понимать" = > #Term*

where *Term* should be in instrumental case (*c=ins*), but extracted in normal form.

Rules of the third group recognizes and extracts pairs of term synonyms (of valid grammatical pattern), in such contexts as *...разрядность, или просто длина слова"* (*...bitness, or simply the length of a word ...*). In the following rule recognition relies on comma and lexical markers (words *или просто*, the latter *просто* is optional):

*Term1 "," "или" ["просто"] Term2 => # Term1 "-" #Term2*

Extraction by the described rules yields three sets of term candidates: $S_{gramm}$, $S_{auth}$, $S_{syn}$, respectively, and the sets are intersected, in particular, there are terms extracted by patterns of grammatical structure and also by patterns of term definition. Therefore, a procedure is necessary, for selecting unique and more significant term candidates for a subject index.

For this purpose, we have estimated precision of term extraction for each group of rules, taking several textbooks of medium size in computer science (each is about 20 thous. words), which contain back-of-the-book indexes constructed by their authors (the problem encountered while performing experiments is the lack of human-built indexes in many Russian textbooks). The experiments have expectedly shown high recall but low precision (about 8-12%) for the first rule group, but for the second group rules the results were opposite, with high precision (91-95%), due to lexical markers used in them. We have formed a subset of very-high precision (VHP) rules from the second group, since for them extracted terms are to be obligatory included into subject index. The third group of rules show a rather good precision: 63-67%, and we also include the extracted terms into index.

## 4 Selection of Terms

The developed heuristics procedure iteratively forms a collection of index terms from pre-extracted sets of term candidates $S_{gramm}$, $S_{auth}$, $S_{syn}$ , aiming at reliable selection of the most important terms by accounting for the following factors.

- There are many non-terms among $S_{gramm}$, most of them can be filtered through applying lists of stopwords (auxiliary words and words of general lexicon);
- Terms extracted by very-high precision (VHP) extraction rules of term definitions should be selected first;
- According to Zipf's law, the most significant terms are units with an average frequency, so the most frequent and rare candidates must be discarded;
- Statistical measure C-value [10] estimating termhood (by accounting nesting of terms) and thus measuring term importance is useful for ranking selected terms;
- Terms occurred in content section (if any) of the text document or used in titles of its sections/subsections should be included in the index;
- Term candidates that are synonymous to already selected index terms can also be added to the subject index;
- Since index terms are often lexically similar (they refer to close concepts), a term candidate can be added into the index if it has common words (at least one) with any yet selected term (e.g., terms *second order predicate* and *logical predicate* are lexically similar).

The selection procedure encompasses three stages. The first stage involves filtering pre-extracted sets of term candidates with the aid of pre-compiled lists of stop words. The first list contains words that cannot themselves be terms (*план, начало – plan, start*, and so on), while the second list contains words that cannot be part of terms (e.g., *данный, низкий – given, low*). From all the sets $S_{gramm}$, $S_{auth}$, $S_{syn}$, their elements are excluded that: a) are encountered in the first list; b) contain words from the second list; c) consist of words from the first list. Thereby many collocations of common scientific lexicon such as *given plan* are discarded.

At the second stage, for all filtered term candidates, frequencies of their occurrences the text are calculated, and for frequencies of units from $S_{gramm}$ the percentiles

are calculated with the levels $p_1 = 0.4$ (rounding down) and $p_2 = 0.95$ (rounding up), respectively. Values $p_i$ are exploited as thresholds for eliminating unlikely candidates (both rare and frequent).

Then, the resulting set $R$ of subject index terms is incrementally formed by taking elements from the filtered $S_i$, through the following steps (initially $R$ is empty):

1. Term candidates from the set $S_{auth}$ obtained by VHP rules and with the frequency in the range $[p_1, p_2]$ are included in the set *R*.
2. Term candidates from the set $S_{gramm}$, with frequency in the range $[p_1, p_2]$ are added to *R*, provided that i) they are encountered in any title of sections/subsections in the text document (or list of content, if any) or ii) they have common words (at least one) with any term selected in Step 1.
3. Term candidates remaining in the set $S_{auth}$ (i.e. unconsidered in Step 1) and having common words (at least one) with any element from current $R$ are added to *R*.
4. Term candidates from the set $S_{auth}$ or $S_{syn}$, which are synonymous to a term from *R*, are added to *R*.
5. All pairs of synonyms from the set $S_{syn}$, whose total frequency is in the range $[p_1, p_2]$ for percentiles calculated for total frequencies of all synonymous pairs, are added to *R*.
6. Term candidates from the set $S_{gramm}$ with frequency in the range $[p_1, p_2]$, are added to *R*, provided they have common words with any element from current *R*.
7. Elements of *R* are ordered according their C-value and only the first $N_{top}$ elements (considered as more significant) are remained in the resulted index list.

The thresholds for percentiles $p_1$, $p_2$ and the order of the described steps were chosen experimentally. The value of $N_{top}$ is determined by the size of the source document, because the larger the text, the longer the list of candidate terms, and the less significant terms are located at its end. The value of $N_{top}$ may be about 50-90.

To experimentally evaluate efficiency of the described selection procedure, seven medium-sized (about 70-100 pages) educational scientific texts with human-built subject indexes were taken, the indexes were regarded as etalon sets of terms. The processed texts are devoted to programming systems (PS), programming languages (PL), formal grammars (FG), artificial intelligence (AI), discrete mathematics (DM). The results measured in precision (P), recall (R), and F-measure (F) are shown in Table 1. While evaluating, we took into account the coincidence of the concepts designated by formally different terms (such as *условная конструкция – условие*, *conditional construction – condition*), and $N_{top}$ contains all the selected terms.

One can notice that our methods of term extraction and selection demonstrates quite good performance: its recall (in average 0.78) is sufficient, and precision (in average 0.71) is acceptable, as well as F-measure, 0.74), they exceed the rates of the above-considered methods of term extraction for subject indexes [1, 6, 7]. For comparison, we also have processed and evaluated the manual devoted to academic writing (11,699 words), it can hardly be attributed to scientific or technical text. The precision proved to be 72% and recall 55% (F-measure, 62 %). The low recall may be partially explained by lack of explicit definitions of certain important but rare terms.

**Table 1.** Recall and Precision of Selection Procedure

| Text | Size (words) | Selected Index Terms | | P | R | F |
|---|---|---|---|---|---|---|
| | | # | Examples | | | |
| PS | 11,699 | 67 | *функциональное тестирование (functional testing)* | 0.70 | 0.81 | 0.75 |
| PL-1 | 21,060 | 140 | *ветвь условного выражения (branch of conditional expression)* | 0.74 | **0.84** | 0.79 |
| PL-2 | 29,301 | 208 | *левое согласование (left matching)* | 0.56 | 0.82 | 0.67 |
| PL-3 | 21,376 | 77 | *предикат второго порядка (second order predicate)* | 0.77 | 0.72 | 0.75 |
| FG | 15,890 | 73 | *нетерминальный символ (nonterminal symbol)* | **0.79** | 0.83 | **0.81** |
| AI | 19,471 | 98 | *алгоритм слепого перебора (blind search algorithm)* | 0.71 | 0.74 | 0.73 |
| DM | 20,786 | 222 | *компонента связности (component of connectivity)* | 0.73 | 0.71 | 0.72 |
| Mean | 19,940 | 126 | | **0.71** | **0.78** | **0.74** |

It should be noted that some extracted terms absent in the etalon subject indexes (such as term *proof tree* from the textbook on Prolog) are terms relevant for subject index, they may be omitted by human indexer because of subjectivity or intent to get a more short index. Thus, for subject index construction, recall is more crucial than precision: it is easier for human editor of the constructed index to discard some terms than to add new ones. Besides, to increase recall, the editor can change values $p_1$, $p_2$.

## 5 Identifying Subordinate Relations of Terms

To form hierarchical structure of subject index, subordinate links among pairs of selected terms are to be recognized, and corresponding headings and subheadings are to be formed. Our method of revealing subordinate relations makes use of information about structure of multi-word terms.

Admittedly, hyponym terms often originate from hyperonym terms by complementing them with qualifying words [6], e.g., *свертка* (*convolution*) – *левая свертка* (*left convolution*), *протокол передачи* (*transfer protocol*) – *протокол передачи почты* (*mail transfer protocol*). Accordingly, we determine potential hyperonyms (headings) based on grammatical patterns of compound terms, and particular LSPL rules with lexico-syntatic patterns are created for this purpose. Three examples of grammatical patterns and the corresponding grammatical patterns of potential heading are presented in Table 2.

In addition to grammatical patterns of headings, which were determined for all permissible multi-word terms, frequencies of terms in the document being processed are used, according to idea that any heading term should be more frequent than its subheadings.

**Table 2.** Grammatical patterns of heading and subheading

| Patterns of terms | Examples of terms | Grammatical patterns, examples of heading terms | |
|---|---|---|---|
| *A N* | *Числовой атом – Numeric atom* | *N* | *Атом – Atom* |
| *N1 N2<c=gen>* | *Ветвь функции – Function branch* | *N1* | *Ветвь – Branch* |
| | | *N2* | *Функция – Function* |
| *A1 A2 N* | *Ациклический ориентированный граф – Acyclic directed graph* | *N* | *Граф – Graph* |
| | | *A1 N* | *Ациклический граф – Acyclic graph* |
| | | *A2 N* | *Ориентированный граф – Directed graph* |

Specifically, to form headings and subheadings, the following procedure is performed, for each compound term $T$ of subject index. First, all words and phrases $\{T_p\}$, which can potentially become headings, are extracted from $T$ . Then, occurrence frequencies both for $T$ and for each element from $\{T_p\}$ are calculated. Finally, the following rule is applied: the word or phrase $T_{p_i}$ with the highest frequency among $\{T_p\}$ is chosen as heading, provided that its frequency exceeds the frequency of the term $T$. If there are several such elements, then the first one is selected, according to alphabetical order of the terms. In the case when the frequencies of all $\{T_p\}$ do not exceed the frequency of $T$, $T$ itself becomes the heading (without subheadings).

For example, for term *Ациклический ориентированный граф* (*Acyclic directed graph*), potential headings are: *граф* (*Graph*), *Ориентированный граф* (*Directed graph*), *Ациклический граф* (*Acyclic graph*). If term *Ориентированный граф* has the highest frequency, it becomes heading, and *Ациклический ориентированный граф* will be the corresponding subheading.

After selection of headings, a hierarchical structure for subject index terms is formed. From each subheading, its constituent part identical to the heading is deleted, and the rest is placed on the low level of the hierarchy. Resulting structure for the considered example is shown below:

***Rus.***: *Ориентированный граф*
*– ациклический*

***Eng.***: *Directed graph*
*– acyclic*

## 6 Revealing Additional Index Terms and Semantic Relations

To increase recall of index term selection, we propose to additionally use clustering of terms candidates from $S_{gramm}$, since such clustering can reveal groups of semantically related units, which together with already selected terms include significant terms that were missed at the stage of term selection.

Term clustering is almost not investigated, the paper [1] applies clustering merely for term extraction, with the purpose to construct glossaries for documents. Resulted clusters contain semantically related groups of terms, but their semantic links are not identified and not represented in constructed glossaries. Since for processing individ-

ual texts, automatic identification and classification of semantic links is a really difficult task, we believe that within a back-of-the-book indexing system it is reasonable to leave such classification work to human editor of the index being constructed, and the system only reveals groups of semantically related terms.

It should be noted that in our work, synonymy relations needed for establishing cross-references in subject index, are mainly identified at the stage of term extraction: pairs of synonyms introduced by the author of the text are recognized by lexico-syntactic patterns of the third group. Besides such obvious synonyms, term variants [6] , such as *пролог-интерпретатор* and *интерпретатор Пролога* (*Prolog interpreter* and *interpreter of Prolog*) are often encountered in scientific texts. As our experiments showed, such variants as well as another semantically related pairs are effectively detected by clustering.

In the experiments we applied Kmeans and DBSCAN clustering algorithms with context similarity measure that compares context words of two terms, from a window of size 4 (context is regarded as bag of words). Context similarity is evaluated with Jaccard index (the proportion of common context words in the set of all context words for the compared terms). Additionally we considered analogous measure with context words represented as vectors in distributional vector space, but we had to abandon it, since many words included in specific terms are absent in the known vector models RusVectores[3].

Results of Kmeans algorithm were better than for DBSCAN, and its hyperparameter, i.e. the number of clusters was experimentally selected so that the average cluster size was 5-10. Below we present three examples of clusters yielded by Kmeans with the context similarity measure:

1) *регулярная грамматика, формальная грамматика, конечный автомат, автомат – regular grammar, formal grammar, finite state automaton, automaton*;

2) *шлюз, маршрутизатор, коммутатор, коммуникационное оборудование – gateway, router, commutator, communication equipment*;

3) *простая рекурсия, хвостовая рекурсия, косвенная рекурсия, характеристика* – simple recursion, tail recursion, indirect recursion, characteristic**.

One can notice that in these groups there are pairs of terms with subordinate relation that can not be detected by our build-in lexico-syntactic patterns for this relation (*regular grammar* and *formal grammar*), as well as co-hyponims (*simple recursion* and *indirect recursion*), terms with certain association semantic relation (*regular grammar* and *finite state automaton*; *commutator* and *communication equipment*). Such pairs can be useful for enriching subject index. At the same time clusters may include elements semantically unrelated with the others (in the above example, such elements are marked with *), therefore a human should analyze them.

The experiments showed that most clusters contain semantically related terms. To enrich the set of index terms, only those clusters that include at least one yet selected index term are automatically detected and then are presented to the human editor to identify additional relevant terms among the elements of each cluster.

[3] https://rusvectores.org/ru/

## 7 Determining Reference Pages

Every term of subject index should be associated with page numbers or/and page ranges (for example: 5, 81-83) that indicate occurrences of the term in the text document. Some terms may be quite often used in the text, and it is not reasonable to include references to all pages with their occurrences. Usually, only significant places of term usage are detected and correspondent pages are placed into the subject index.

Evidently, pages with detected definitions of terms is significant, so we necessarily include them to a subject index being constructed. We determine significance of other pages, depending on occurrences frequency of the given term on these pages, which may be regarded as "density" of term usage.

To evaluate the density of occurrences for a particular term in the text, we propose to cluster the multi-set of page numbers for pages with occurrences of the term (a page number is repeated if the term is encountered several times in it) and then to form page ranges for each resulted cluster. In general case, each resulted cluster contains neighboring pages, which are concatenated into page range, but with the following reasonable restriction. The maximum permissible distance between two neighboring pages in page range is equal to $M$ ($M = 1\ or\ 2$), otherwise, the range may include more than $M$ pages without occurrences of the term, and this is unacceptable for reader of the text. The number $K$ ($K = 2, 3, 4, 5$) delimiting the number of page references (it also should be reasonable) is additional parameter of our method for determining page references.

For a given term, the steps of our method are as follows.

1. All occurrences of the term in the document are recognized (disregarding the exact form they take), and multi-set $S_{page}$ of page numbers for term occurrences is formed: if the term is used several times on a page, then its number is added to $S_{page}$ as many times as term is encountered, for example: $S_{page} = \{12, 23,23,25,28,29,29,30,31,31,31,33,50,51,70, 90\}$.
2. $S_{page}$ is divided into clusters, with the DBSCAN density-based clustering algorithm and the parameter $M$; in our example $M = 1$ and six clusters are formed: {12} {23,23,25} {28,29,29,30,31,31,31,33} {50,51} {70} {90}.
3. Clusters are ordered by cardinality of multi-sets, in our example: {28,29,29,30,31,31,31,33} {23,23,25} {50,51} {12} {70} {90}; and then the first $K$ ($K = 2$) clusters are taken**:** {28,29,29,30,31,31,31,33}, {23,23,25}.
4. In each such cluster, the repeated elements (page numbers) are deleted, and the remaining ones are sorted in ascending order: {28,29,30,31,33} {23,25}.
5. In the case when the page with definition of the given term (if any) is absent in these clusters (for example, 12), corresponding one-element cluster is added: {28,29,30,31,33} {23,25} {12}.
6. The clusters are sorted by ascending order of their first element numbers: {12}{23,25}{28,29,30,31,33}.
7. Each cluster with more than one element is converted to a page range while one-element clusters give separate pages: *12, 23-25, 28-33.*

Since ways for estimating the quality of selecting reference pages were not proposed in the related works, we experimentally evaluated recall of our method, i.e. the degree of coverage of the pages indicated in the author's subject indexes with clusters of pages yielded by our method. For evaluation, texts of the same scientific textbooks were taken. The obtained coverage rates are from 84.4% to 94.3% (depending on particular text), which is quite good quality.

## 8 Conclusion and Future Work

In this paper we have proposed the methods for automating all the stages and tasks of constructing back-of-the-book index for an individual text document, including term extracting and filtering, detecting semantic relations of terms and important occurrences of index terms in the document. The methods were implemented and evaluated within a prototype subject indexing system with open code[4] for Russian text documents. At all stages of subject index construction, the user of the system can set and change necessary parameters of the methods, can indicate a text fragment to be processed and then verify and edit the results.

The evaluation of the proposed methods have shown their quite good performance, in particular, our technique of term extraction and selection gives considerable increase of precision and recall in comparison with the previous related works. In our opinion, it is mainly due to built-in knowledge about terms in scientific and educational texts, which was formalized as the set of rules with lexico-syntactic patterns and used in combination with the heuristics about term importance.

On the way towards high-quality indexing tools, further experiments and improvements for our methods are needed, below we indicate some of them:

- To test and refine the heuristic selecting procedure for documents from another problem domains;
- To create procedures for extraction of named entities significant in text of certain problem domains (for example, in texts on programming these are names of built-in program function);
- To develop additional methods to automatically recognize semantic relations of terms based on models of distributional semantics.

## References


1. Arora, C., Sabetzadeh, M., Briand, L., Zimmer, F.: Automated Extraction and Clustering of Requirements Glossary Terms. IEEE Transactions on Software Engineering, Vol.43, Issue 10, pp. 918–945 (2016)
2. Bolshakova, E., Efremova, N., Noskov, A.: LSPL-Patterns as a Tool for Information Extraction from Natural Language Texts. In: New Trends in Classification and Data Mining, pp. 110–118. ITHEA, Sofia (2010)


[4] https://github.com/ivanov-kir-m/SISTool

3. Bolshakova, E., Efremova, N. : A Heuristics Strategy for Extracting Terms from Scientific Texts. In: Analysis of Images, Social Networks and Texts. Fourth Int. Conference AIST, CCIS, Vol. 542, pp. 285–295. Springer, Heidelberg (2015)
4. Bolshakova, E., Ivanov, K.: Term Extraction for Constructing Subject Index of Educational Scientific Text. In: Computational Linguistics and Intellectual Technologies: Papers from the Annual Int. Conference "Dialogue". Issue 17 (24), pp.143–152. Moscow (2018)
5. Christina, S., Oktaviani, E.: Identifying the relevant page numbers that referred by the back-of-book index using syntactic similarity and semantic similarity. In: Proceedings of 2017 Second Int. Conference on Informatics and Computing (ICIC), pp. 1–6 (2017)
6. Cram, D., Daille, B.: TermSuite: Terminology extraction with term variant detection. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics—System Demonstrations, p. 13–18. Berlin, ACL (2016)
7. Csomai, A., Mihalcea, R.: Investigations in Unsupervised Back-of-the-Book Indexing. In: Proceedings of the Florida Artificial Intelligence Research Society Conference, pp. 211–216 (2007)
8. Csomai, A., Mihalcea, R.: Linguistically Motivated Features for Enhanced Back-of-the Book Indexing. In: Proc. Ann. Conf. of the Association for Computational Linguistics, ACL/HLT, Vol. 8, pp. 932–940 (2008)
9. Da Sylva, L.: Integrating Knowledge from Different Sources for Automatic Back-of-the-Book Indexing. In: Proceedings of the Annual Conference of CAIS/Actes du congrès annuel de l'ACSI (2013)
10. Frantzi, K., Ananiadou, S., Mima, H.: Automatic Recognition of Multi-Word Terms: The C-value/NC-value method. In: Nikolau, C. et al. (Eds.) International Journal on Digital Libraries, vol. 3(2), pp. 115–130 (2000)
11. Hasan, K.S., Ng, V.*:* Automatic keyphrase extraction*:* a survey of the state of the art. In: Proceedings of the 52th Annual Meeting of the ACL, pp. 1262–1273 (2014)
12. Hearst, M.A.: Automated Discovery of WordNet Relations. In: FellBaum, C. (Ed.) WordNet: An Electronic Lexical Database, pp. 131–151. MIT Press, Cambridge (1998)
13. Korkontzelos, I., Ananiadou, S.: Term Extraction. In: Oxford Handbook of Computational Linguistics (2nd Ed.). Oxford University Press, Oxford (2014).
14. El Mekki,T., Nazarenko, A.: An application-oriented terminology evaluation: the case of back-of-the book indexes. In: Proc. of the Workshop on Terminology Design: quality criteria and evaluation methods (LREC-TermEval), pp.18–21.Genoa, Italy (2006)
15. Reinholt, K., Lukon, S., Juola, P.: A Machine-Aided Back-of-the-Book Indexer. In: Proceedings of DHCS 2010, Chicago, Illinois (2010)
16. Sajatovic, A., Buljan, M., Snajder, J., Basic, B. D.: Evaluating Automatic Term Extraction Methods on Individual Documents. In: Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019), pp. 149–154. Florence, Italy (2019)
17. Salton, G.: Syntactic Approaches to Automatic Book Indexing. In: Proceedings of the 26th annual meeting of the ACL, pp. 204–210. Morristown, NJ, USA (1988)
18. Wu, Z. et al.: Can Back-of-the-Book Indexes be Automatically Created? In: Proc. of the 22nd ACM Int. Conference on Information & Knowledge Management. ACM (2013)
19. Zhang, Z., Iria, J., Brewster, C., Ciravegna, F.: A comparative evaluation of term recognition algorithms. In: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC08), pp. 2108–2113. Marrakech, Morocco (2008)
20. Zargayouna, H., El Mekki, T., Audibert, L., Nazarenko, A. IndDoc: an Aid for the Back-of-the-Book Indexer. The Indexer, 25(2), pp. 122–125 (2006)