Topic label generation in the popular science corpus

Mitrofanova Olga, Ten Lia and Athugodage Mark

Saint-Petersburg State University, Saint-Petersburg, Russia
o.mitrofanova@spbu.ru, st050030@student.spbu.ru,
m.athugodage@yahoo.com

Abstract. The paper presents results of experiments on topic label generation from web data and distributional semantic models. The procedure in question is required for topic label assignment in Russian popular science corpora. Topic modeling is performed by means of a series of algorithms including non-negative matrix factorization, latent Dirichlet allocation, biterm topic modeling. Our approach allows for reducing the shortcomings of conventional topic label assignment by choosing the first topical term as a topic label. We introduce an improved version of topic label generation as an ensemble of heterogeneous methods. Candidate labels are evaluated in course of human assessments. Results of our research allow us to verify the structure of scientific media sites and thus to improve their quality.

Keywords: topic modeling, topic label assignment, Russian corpora, scientific texts, human assessments

1 Introduction

Topic modeling is a way of building a semantic model of text corpora that determines interrelations of topics, documents and topical words. Topics are treated as hidden factors represented by clusters of topical words. Each document is associated with one or more topics with some probability or weight and the topics themselves may intersect: a certain word can be attributed to several topics [1]. Topic models help to improve the efficiency of procedures for extracting information from natural language texts, such as automatic headline generation, document clustering and classification, sentiment analysis and make a significant contribution to the training of AI systems [2, 3]. The scope of topic models is wide; they cover text corpora of different types and genres, among which are news [4, 5], social media texts [6-8], medical texts [9], financial texts [10], scientific texts [11], and fiction [12-15]. Our study is designed to solve the problem of studying the topical structure of popular scientific texts, which are of great demand in social media and educational sphere.

Frequently used topic modeling techniques include a group of algebraic models such as latent semantic analysis (LSA), non-negative matrix factorization (NMF) and probabilistic models such as probabilistic latent semantic analysis (pLSA), latent Dirichlet allocation (LDA), Pachinko allocation, Hidden Markov topic model. In practice, their multimodal extensions are often used that introduce additional corpus parameters. Such are authorship in author-topic model, addressee in author-recipient-topic model, relations between topics in hierarchical topic models, the presence of predefined topical words in Guided LDA, linguistic structures within topics in n-gram topic models, the possibility of generalization by introducing labels, changes in topical structure over time in dynamic topic models, etc. [1, 5, 7, 12, 15, 16, 17]. In recent years, a new class of topic models has emerged that combines probabilistic processes

and distributed vector models, for example, LDA2Vec, Top2Vec, embedded topic model, contextualized topic model, and BERTopic [18, 19]. The advantage of combined topic models is that they improve the quality of semantic representations and reduce the losses associated with the use of bag-of-words approach.

As a rule, topic modeling does not necessarily include topic label assignment as an obligatory procedure. Traditionally, topics are presented as a number and the first word or words with the highest probability or weight representing their attachment to the topic. There may be difficulties in understanding the output of topic modeling algorithms, especially for a non-specialist; labels are used to make topics easier to interpret. A label is a sequence of words that can capture the general meaning of a given set of topical words. The relevant labels are often manually assigned to topics based on subjective criteria. However, automatic selection of topic labels not only makes it easier to interpret the extracted word distributions, but also saves time and effort spent on manual indexing.

NLP provides several automatic methods for topic labeling; these methods are divided into three classes depending on label sources, types of algorithms involved, and label structure. The source for labels can be either internal, so that the labels are taken directly from the research corpus, or external, so that they are extracted from reference corpora, search engine output or knowledge bases (Wikipedia, WordNet). Algorithms of topic label assignment can be supervised or unsupervised. As regards their structure, labels can be unigrams, bigrams, etc. Label assignment through internal sources includes determining Kullbach-Leibler distance between word distributions and maximizing mutual information between candidate labels and topics [20]; rearrangement of relevant words in terms of their attachment to the topic [21]; ranking candidate labels using summarization algorithms [22]; extracting candidate label n-grams from documents most relevant to the topics, matching candidates to word vectors and letter trigrams, ranking candidates by similarity between topics and tag vectors [23]; finding documents closest to the topics, extracting individual terms and set expressions and ranking them according to information measures [24], etc. Among different approaches to assigning labels using external sources are term extraction from Google directory hierarchy (gDir) [25]; title extraction from Wikipedia or DBpedia and candidate label ranking candidate [26, 27]; using the web as a corpus for extracting candidate labels using Google search and ranking candidates with PageRank [28]; using Wikipedia titles as candidate labels and ranking candidates through neural embedding operations for words and documents [29], incorporating a formal ontology into a topic model for knowledge extraction (KB LDA) [30], using k-nearest neighbors clustering and hashing for quick label assignment to newly emerging topics [31], etc.

In [5, 32, 33] the authors presented two approaches to topic label assignment for Russian corpora, namely candidate labels extraction from Yandex search engine (Labels-Yandex) and candidate labels extraction from Wikipedia by operations on word vector representations in explicit semantic analysis (Labels-ESA). Evaluation procedure showed that in most cases Labels-Yandex algorithm predicts correct labels but frequently relates the topic to a label that is relevant to the current moment, but not to a set of keywords, while Labels-ESA works out labels with generalized content.

In this paper, we propose a novel approach to topic label assignment, which is applicable in processing a popular scientific corpus that covers a wide range of topics. First, we discuss the problems of corpus building, filtering, and annotation. Second, we compare a set of topic modeling algorithms (LDA, NMF and BTM) that reveal topical structure of the corpus, analyze training hyperparameters and evaluation procedures. As the topical structure of the research corpus may not coincide with Wikipedia, we focus our attention at candidate labels extraction from the search engine and expand it by adding distributed vector representation model predictions and summarization procedures for topic label generation and ranking. Topic label verification is performed in course of a perceptual experiment, results of which are compared with the baseline worked out in previous research.

2 Topic modeling in the popular science corpus

2.1 Research corpus

The corpus developed within our study is a compilation of Russian texts sampled from *Elementy bolshoi nauki* [34], an online media outlet covering various aspects of natural sciences and technology. It contains 2,289 popular science articles published between 2010 and 2023, or approximately 3 million words.

As topic models typically need the data to be preprocessed, the necessary steps to be taken included lowercasing, tokenization with NLTK [35], lemmatization using pymorphy2 [36], and collocation extraction with Gensim module Phrases [37]. The latter was used as a means of improving topic coherence and overall topic distinctiveness as it keeps multiword expressions in the corpus instead of breaking them down into separate tokens. A total number of 5,389 unique noun phrases were extracted at this point. In addition, we removed all punctuation marks, digits, words including only latin characters as well as stop words based on a custom list of 1000 items. Only nouns and adjectives made it to the final version of the corpus as the most informative parts of speech regarding a document's content [38], with the size of the corpus reduced to 1.5 million tokens. Aside from the text itself, some metadata was also retrieved, including the title, the name of the author, the outlet, publication date, and the topics provided by the author. In each outlet, the in-built topics are essentially keywords describing the contents of the article and making navigation across the site easier. Next, we filtered out tokens that are too frequent or too rare to be informative, removing words that occur in less than 1% or more than in 20% to 70% of the documents depending on the model. As a result, the number of unique tokens ranged from 5,570 to 47,270.

2.2 Topic Modeling Results

The models built for the corpus were latent Dirichlet allocation (LDA), non-negative matrix factorization (NMF), and biterm topic model (BTM) [1, 3, 5, 16, 17, 39]. The intuition behind these algorithms is that no prior knowledge is needed for topic

extraction, although LDA typically requires detailed assumptions regarding the hyperparameters [40].

More specifically, LDA is a generative probabilistic model that uses word distributions for topic extraction. It is considered a three-level hierarchical Bayesian model comprising document level, topic level, and word level. At the document level, each document is represented as a finite mixture over a set of topic probabilities. At the topic level, each topic is represented as a finite mixture over an underlying set of words, and at the word level, each word is modeled as a distribution over topics. LDA typically requires three hyperparameters, or initial beliefs about the distribution: a number of topics, beta, and alpha. Alpha parameter represents document-topic density – with a higher alpha, documents are assumed to contain more topics, which results in a more specific topic distribution per document. Beta parameter represents topic-word density – with a higher beta, topics are supposed to be made up of more words in the corpus, which results in a more specific word distribution per topic.

To determine the optimal values for the hyperparameters, we performed a grid search until reaching the highest coherence score, also referred to as the quality of the extracted topic, with a value of 0.9 for both alpha and beta. The number of topics for LDA was chosen empirically beforehand and stood at 16. A fragment of the output is given below:

Topic 1: *частица, энергия, физика, электрон, измерение, детектор, атом, нейтрино, масса, фотон, ядро, протон, поле, коллайдер, квантовый (particle, energy, physics, electron, measuring, detector, atom, neutrino, mass, photon, nucleus, proton, field, collider, quantum)*

Topic 8: *ребёнок, аллель, отбор, выборка, страна, женщина, показатель, население, смертность, полиморфизм, корреляция, генофонд, индивид, старение, численность (child, allele, selection, sample, country, woman, rate, population, mortality, polymorphism, correlation, gene pool, person, ageing, number)*

Topic 3: *самец, самка, птица, яйцо, потомство, пол, гнездо, муха, колония, спаривание, половой, размножение, сперматозоид, птенец, отбор (male, female, bird, egg, offspring, sex, nest, fly, colony, mating, reproductive, reproduction, sperm, hatchling, selection)*

Alternatively, NMF is a non-probabilistic algorithm that employs a linear algebra approach for topic extraction. It breaks down (or factorizes) high-dimensional vectors into a lower-dimensional representation. The vectors can be represented by words, their raw counts or TF-IDF weights – a measure evaluating the statistical importance of a word in a collection of documents. In contrast to the simplistic bag-of-words approach used in LDA, which counts word occurrences disregarding any semantic information, the TF-IDF weighting generally assigns lower values to more frequent words in the corpus; a word is also considered important when it occurs rarely in the whole collection but frequently in a given document or a set of documents. Moreover, unlike LDA, NMF does not normally require hyperparameter tuning since the only parameter needed to be specified explicitly beforehand is the number of topics. In our case, the topics generated for NMF were more numerous, equaling 26.

Topic 6: *мантия, порода, земля, кратер, поверхность, млрд_год, минерал, марс, планета, образование, магма, базальт, слой, алмаз, древний (mantle, rock,*

earth, crater, surface, million_a_year, mineral, mars, planet, formation, magma, basalt, bed, diamond, ancient)

Topic 20: *опухоль, рак, метастаз, раковый_клетка, клетка, мутация, пациент, раковый_опухоль, терапия, лечение, ткань, ингибитор, опухолевый_клетка, рост, железа (tumor, cancer, metastasis, cancer_cell, cell, mutation, patient, carcinoma, therapy, treatment, tissue, inhibitor, tumor_cell, growth, gland)*

Topic 4: *самец, самка, спаривание, потомство, сперматозоид, ухаживание, яйцо, пол, поведение, половой_отбор, половой, особь, репродуктивный_успех, партнёр, популяция (male, female, mating, offspring, sperm, courtship, egg, sex, behavior, sexual_selection, reproductive, specimen, reproductive_success, partner, population)*

According to [39], BTM explicitly models the word co-occurrence patterns (i.e. biterms) in the whole corpus. In particular, it is most effective when performed on short texts, as word co-occurrence patterns there are sparse and not reliable. The documents used in our study were not short, with the average word count of 1,434, yet the model has been shown to outperform LDA even on normal texts [39]. A few of the 17 topics produced by BTM are listed below:

Topic 1: *галактика, масса, частица, звезда, энергия, физика, ядро, чёрный_дыра, вселенная, модель, нейтрино, детектор, наблюдение, эксперимент, вещество (galaxy, mass, particle, star, energy, physics, core, black_hole, universe, model, neutrino, detector, observation, experiment, matter)*

Topic 4: *физика, научный, человек, наука, университет, эксперимент, журнал, проект, теория, открытие, вопрос, институт, начало, решение, сша (physics, scientific, person, science, university, experiment, journal, project, theory, discovery, issue, institute, beginning, solution, usa)*

Topic 12: *самец, самка, особь, потомство, эксперимент, поведение, яйцо, популяция, птица, спаривание, пол, пара, муха, признак, маленький (male, female, specimen, offspring, experiment, behavior, egg, population, bird, mating, sex, pair, fly, feature, small)*

The output reveals a seemingly equal degree of topic interpretability across the models; all topics contain both common and technical terms, which are commonly found in popular science texts, and most of the terms within each topic seem to describe the same concept such as «cancer», «geology» or «elementary particles». Moreover, many topics in all three models contain roughly the same set of words or otherwise overlap (cf. the last topics in each model). The only major difference is the number of collocations, or n-grams, presented in a topic. In this respect, NMF appears to be the most sensitive to n-grams out of the three and thus potentially puts forward better results. To test this assumption, we then evaluated each model's performance in terms of topic coherence, which is based on the premise that words co-occurring more frequently are more likely to belong to the same topic [39]. Specifically, it measures the degree of semantic similarity between high scoring words in the topic, often using a PMI score. For BTM, we used the UMass coherence metric proposed by [41], which equaled -125.4 (that is considerably better than the baseline of -167.1, which represented a «good» topic according to the authors). For LDA, the UMass measure was implemented from [42] where it takes the values between -14 and 14 and in our case was equal to -1.75. In both

cases, numbers closer to zero indicate higher coherence. As for NMF, we implemented a custom approach using a Word2Vec model and word similarities. The coherence score thus obtained was 0.46 out of 1.0. Unfortunately, while all these results by themselves indicate a rather high semantic interpretability of topics, the lack of a common, out-of-the-box evaluation technique makes it impossible to compare the models directly, leaving human judgment the only reliable option.

3 Our approach to topic labeling

In this study, we used the following set of techniques for topic label generation: search engine topic labeling, topic labeling using Word2Vec, summarization-based topic labeling, and topic labeling with ChatGPT.

3.1 Search engine topic labeling

In this part of our work, we modified the approach introduced in [5, 32, 33]. The proposed idea is to use web-scraping techniques to generate candidate labels by extracting data from the Internet. Scraping is possible with either WebScraper [43] or Selenium [44]; in both cases, a robot-browser imitates human behavior by going through web pages [45]. Some other options for web scrapers include Nutch [46], Pyquery [47], Import.io [48], and Beautiful Soup [49]. Beautiful Soup is essentially the most notable of them as it is one of the basic Python libraries; however, it cannot go through web pages or type in input spaces [45].

Selenium is undoubtedly the simplest and the most effective Python library for web scraping. The procedure starts with initializing Google Chrome Webdriver – a specific driver sharing all functions and capabilities with Google Chrome browser. This robot is then tasked with searching for the topics obtained through traditional topic modeling techniques. The whole sequence of tokens, e.g. the whole topic, is directed into the search query. The robot looks through the first three pages of search results, collecting approximately 30 titles, with the number ranging depending on a topic's content. In Selenium, web page elements can be detected in multiple ways through XPATH, CSS selector, class name, text, and some others. XPATH and CSS selector are both precise, since there each web page element has its own address [50]; however, XPATH has been proved to be more efficient and stable while processing Google result pages. XPATH models each web page, which is by default an XML document, as a tree of nodes; there are different types of nodes, including element nodes, attribute nodes, and text nodes. XPATH defines a way to compute a string-value for each type of node [51, 52]. In our research, titles were retrieved with XPATH detector, which is built in the Webdriver.

After collecting the titles, we employed cleaning and lemmatization techniques; the former was necessary due to the titles having many redundant characters such as non-textual symbols and digits. All accented characters were transformed into their unaccented variants. The titles were lemmatized with the use of pymoprhy2. Generated labels were n-grams up to five units, although unigrams were also present. Some of the examples are *задание по химия*, *мозг от аксон до нейрон (chemistry homework , the brain from axon to neuron)*. Thus, a label was represented as a set of lemmatized tokens, while in reality it is an independent utterance. In order to get elaborate labels a search

engine was applied for a second time. Search engines often offer corrections to the search queries in case there are misspellings or typos; this function is useful when obtaining grammatically correct phrases without having to examine them manually. However, since Google offers corrections for Russian queries only if there are personalized settings in the browser and Google Chrome web browser is not personalized, we had to choose another search engine with Russian as a default language. Although there are many Russian-based search engines (Yandex, Mail.ru), the most efficient for our task was Rambler. We easily reproduced the algorithm originally implemented in relation to Yandex search engine (Labels-Yandex) as Rambler employs Yandex.XML technology. However, Yandex linguistic procedures fail to reconstruct original grammatical forms of lemmatized phrases; consequently, Yandex ranking results are corrected with the help of Rambler linguistic plug-ins.

The algorithm is similar to that of label generation described in [5, 32, 33]: all topical n-grams (10 items by default) are put into a search query in the Rambler search engine. The output is transmitted to TextRank calculator; further, candidate labels (both unigrams and lexical-grammatical constructions corresponding to frequent patterns) are selected according to TextRank values. The main change is that instead of collecting titles, the algorithm collects the corrected query, which is usually placed just below the search icon. Corrected query was taken with XPATH method, because BeautifulSoup (a standard library for searching elements on the page) does not see the element. The algorithm allowed us to get the following labels: *задание по химии*, *мозг от аксона до нейрона* (*chemistry homework, brain from axon to neuron*). However, it is necessary to note that the proposed method sometimes leaves the labels unchanged. Examples are *животное в сон*, *птица от яйца до взрослый* (*animal to sleep, bird from egg to adult*). Examples of topic generation for BTM model are given in Table 1. The labels we deemed acceptable are marked bold.

Table 1. Labels generated by Search Engine output processing

Topic	Search engine topic labeling
физика, научный, человек, наука, университет, эксперимент, журнал, проект, теория, открытие, вопрос, институт, начало, решение, США (physics, scientific, person, science, university, experiment, journal, project, theory, discovery, question, institute, start, solution, USA)	***наука в США и России, наука в США, метафизика и наука,*** *методология и метод,* ***метод и технология*** *(**science in the USA and Russia, science in the USA, metaphysics and science,** methodology and method, **method and technology**)*
человек, мозг, нейрон, животное, эксперимент, поведение, сигнал, мышь, испытуемый, песня, птица, информация, уровень, обучение, социальный (human, brain, neuron, animal, experiment, behavior, signal, mouse, test subject, song, bird, information, level, learning, social)	*сознание и мозг как мозг, нейронаука для медицина и психология, нейрон и душа, образование и наука, медицина и психология (**consciousness and brain** as a brain, **neuroscience for medicine and psychology**, **neuron and soul**, education and science, **medicine and psychology**)*
галактика, масса, частица, звезда, энергия, физика, ядро, чёрный_дыра, вселенная,	***нуклон синтез в вселенная, дыра в центре,*** *портрет в интерьере,*

модель, нейтрино, детектор, наблюдение, эксперимент, вещество (galaxy, mass, particle, star, energy, physics, core, black_hole, universe, model, neutrino, detector, observation, experiment, matter)	***вселенная и человек, физика и астрофизика (nucleon fusion into the universe, hole in the center**, portrait in the interior, **universe and man, physics and astrophysics**)*

3.2 Word2Vec topic labeling

Following [5, 7, 29] we used distributed semantic modeling for topic labeling. In this case, we trained a Word2Vec model [53] on our corpus to find *n* most similar words to the highest scoring words in a given topic. For each topic, a distributed representation of words was obtained using continuous bag-of-words (CBOW), one of the two model architectures available for Word2Vec (along with Skip-gram), which does not account for context or word order. A mean of the projection weight vectors of the given words was calculated and then compared to the word vectors in terms of cosine similarity. The words with high enough values were ranked from highest to lowest, the three most similar words considered potential topic labels. Some of the topics and their respective labels are listed in Table 2.

Table 2. Labels generated by Word2Vec model

Topic	Word2Vec topic labeling
*мутация, хромосома, **аллель**, старение, ребёнок, частота, днк, изменчивость, выборка, **генотип**, **фенотип**, потомок, мать, полиморфизм, вредный (mutation, chromosome, **allele**, ageing, child, frequency, dna, variability, sample, **genotype**, **phenotype**, descendant, mother, polymorphism, detrimental)*	***фенотип, аллель, генотип (phenotype, allele, genotype)***
*днк, мышь, опухоль, рнк, **фермент**, **белок**, ткань, рак, клеточный, мутация, синтез, заболевание, аминокислота, кровь, иммунный (dna, mouse, tumor, rna, **enzyme**, **protein**, tissue, cancer, cellular, mutation, synthesis, disease, amino acid, blood, immune)*	***фермент, белок,** вирус (**enzyme, protein,** virus)*

It is clear that the labels thus obtained are words synonymous to the original topical terms, yet they ultimately fail to describe the overall content of a given topic, making it more difficult to interpret. Instead, we searched for a method that would yield a general word or phrase summarizing the meaning of the entire topic.

3.3 Summarization-based topic labeling

In this section, we propose summarization as a new way to generate topic labels. For this purpose, a set of labels was obtained through an abstractive summarization T5 model for Russian. The model is based on Google's mT5-base [54]; it was fine-tuned by David Dale (known as «cointegrated» on HuggingFace Hub). Summarization model is a useful tool for finding the most important labels out of 10 or more. The main problem is that any summarization model is by its nature a text2text-generation model;

therefore, it generates a text with sentences, not just a set of n-grams [55]. To tackle the issue one can put a comma or a dot after each n-gram, e.g.

галактика, масса, частица, звезда, энергия, физика, ядро, чёрный дыра, вселенная, модель, нейтрино, детектор, наблюдение, эксперимент, вещество (galaxy, mass, particle, star, energy, physics, nucleus, black_hole, universe, model, neutrino, detector, observation, experiment, substance)

This significantly reduces the chance that the summarization model will generate a full sentence. The structure, such as the one presented above, was passed to the model. The model generates the most important n-grams, separating them by comma − the way it was in the input. Thus, putting a comma between is a successful strategy to prevent a model from generating a full sentence. The other problem is that some n-grams are repeated several times: in the previous example (see Table 3), the unigram *ядро (core)* occurs twice. This problem can be solved by increasing a repetition penalty in Transformers pipeline, but in this case, one would need to customize it; adjusting settings for each topic individually is generally not a good idea. The other possible solution, the one that was chosen for this research, is transforming a Python list into a Python set. It is also worth mentioning that the summarization model might generate a label that is not initially presented in the topic, although this was extremely rare. Acceptable labels in Table 3 are marked bold.

Table 3. Labels generated by an abstractive summarization model

Topic	Summarization-based topic labeling
галактика масса частица ***звезда*** *энергия физика* ***ядро чёрный_дыра*** *вселенная модель нейтрино детектор наблюдение эксперимент вещество (galaxy mass particle* ***star*** *energy physics* ***nucleus black_hole*** *universe model neutrinos detector supervision experiment substance)*	***звезда, ядро,*** *ядро,* ***чёрный_дыра,*** *все (****star, nucleus,*** *nucleus,* ***black_hole,*** *all)*
клетка *белок* ***нейрон*** *рецептор организм тип белка молекула животное* ***ядро*** *ген вещество ткань сигнал клеточный (****cell*** *protein* ***neuron*** *receptor organism protein type molecule animal* ***nucleus*** *gene substance tissue signal cellular)*	***человек, клетка, нейрон,*** *нейрон,* ***ядро*** *(****human, cell, neuron,*** *neuron,* ***nucleus****)*

3.4 ChatGPT topic labeling

Finally, ChatGPT topic labeling was employed to verify and generalize topic labels obtained at previous stages. For this purpose, a set of labels was generated with ChatGPT, a chatbot developed by OpenAI [56]. Specifically, the bot was asked to a) produce one or more general expressions that would cover the meaning of a given topic and to b) choose the most important word within the topic. The same was asked regarding the labels obtained via search engines; additionally, if there were more than one general expression, the bot was tasked with selecting the most important one. At

this point, a total of 59 topics provided by LDA, BTM, and NMF as well as 59 labels for each of them were given to the bot. As a result, a set of three different labels was assigned to each topic. Some of the examples are presented in Table 4.

Table 4. Labels generated by ChatGPT

Topic	Topic labels (Google)	The most important word within the topic	The most important word within the labels	The most important word for general expressions
физика, научный, человек, наука, университет, эксперимент, журнал, проект, теория, открытие, вопрос, институт, начало, решение, США (physics, scientific, person, science, university, experiment, journal, project, theory, discovery, issue, institute, beginning, solution, USA)	*наука в США и Россия, наука в США, метафизика и наука, методология и метод, метод и технология (science in the USA and Russia, metaphysics and science, methodology and method, method and technology)*	*наука (science)*	*методология и метод (methodology and method)*	*научные исследования (scientific research)*
опухоль, рак, метастаз, раковый_клетка, клетка, мутация, пациент, раковый_опухоль, терапия, лечение, ткань, ингибитор, опухолевый_клетка, рост, железа (tumor, cancer, metastasis, cancer_cell, cell, mutation, patient, carcinoma, therapy, treatment, tissue, inhibitor, tumor_cell, growth, gland)	*перспектива в лечение, лечение, гормонотерапия при раке, важность, опухолевый рост (promise in treating, treatment, hormone therapy for cancer, importance, tumor growth)*	*раковый_клетка (cancer_cell)*	*лечение (treatment)*	*лечение рака (cancer treatment)*

Generally, the more plausible labels were obtained by retrieving the most important word or word phrase within the general expressions produced by ChatGPT. To verify the results, we used an evaluation procedure based on [5] by asking 19 human assessors to rate the generated labels on a scale from 0 to 2, where 0 indicates that a label does

not cover the content of a topic, 1 indicates that a label somewhat covers the content of a topic, and 2 indicates that a label covers the content of a topic completely. Average weights were calculated for each group of labels; labels with a mean rating ≥ 1.5 were considered good and ≤ 0.5 were considered bad. The results are shown in Table 5. As expected, the labels were generally deemed satisfactory, with the most important words among general expressions receiving the highest values. Expanded topic labeling procedure with the best weight 1.52 outperforms introduced in [5] where Labels-Yandex get high weight 1.4 and Labels-ESA get medium weight 0.98 (given maximum threshold 2).

Table 5. The average ratings for each type of labels

The most important word within the topic	The most important word within the labels	The most important word among general expressions
1.16	0.95	**1.52**

4 Conclusion

In this paper, we present modifications of previously developed techniques of topic label assignment and demonstrate the applicability of these techniques in the task of structuring popular science texts in corpora obtained from the web-sources. In our case, topic modeling was performed by means of non-negative matrix factorization, latent Dirichlet allocation, and biterm topic modeling. Topic labels generated with the help of search engine topic labeling, topic labeling with Word2Vec, summarization-based topic labeling, and topic labeling using ChatGPT complement each other, as there are few intersections in the sets of topic labels. Thus, our work introduces an improved version of topic label generation as an ensemble of methods combining inner and outer sources of labels. Further development of our research deals with application of multimodal topic modeling with label assignment for online scientific resources.

Acknowledgements. Research is performed with partial support of SPbSU Scientific Project № 75254082 «Modeling the communicative behavior of residents of a Russian metropolis in the socio-speech and pragmatic aspects with the use of artificial intelligence methods», RSF grant № 21-78-10148 «Modeling the meaning of a word in individual linguistic consciousness based on distributive semantics».

References

1. Daud, A., Li, J., Zhou, L., Muhammad, F.: Knowledge discovery through directed probabilistic topic models: a survey. In: Proceedings of Frontiers of Computer Science in China. P. 280−301. (2010). URL: https://www.researchgate.net/publication/215904200_Latent_Dirichlet_allocation_LDA_and_topic_modeling_models_applications_future_challenges_a_survey
2. Milkova, M.A.: Topic models as a tool of «distant reading». In: Digital economy. № 1(5). P. 57−70. (2019). [in Russian]. URL: http://digital-

economy.ru/images/easyblog_articles/520/DE-2019-01-06.pdf?ysclid=lefaetif1z601986150
3. Vorontsov, K.V.: Probabilistic topic modeling: ARTM regularization theory and the BigARTM open source library. (2023). [in Russian]. URL: http://www.machinelearning.ru/wiki/images/d/d5/Voron17survey-artm.pdf
4. Karpovich, S.N.: Russian-language corpus of texts of SKTM-ru for topic model development. In: Proceedings of the International Conference «Corpus Linguistics-2015». Saint-Petersburg. (2015). [in Russian].
5. Mitrofanova, O., Kriukova, A., Shulginov, V., Shulginov, V.: E-hypertext Media Topic Model with Automatic Label Assignment. In: Recent Trends in Analysis of Images, Social Networks and Texts: 9th International Conference, AIST 2020, Revised Supplementary Proceedings. Communications in Computer and Information Science, vol. 1357. Springer. P. 102−114. (2021). URL: https://doi.org/10.1007/978-3-030-71214-3_9
6. Mamaev, I.D, Mitrofanova, O.A.: Automatic Detection of Hidden Communities in the Texts of Russian Social Network Corpus. In: A. Filchenkov, J. Kauttonen, L. Pivovarova (Eds.). Artificial Intelligence and Natural Language: 9th Conference, AINL 2020, Helsinki, Finland, October 7–9, 2020, Proceedings. Communications in Computer and Information Science. Vol. 1292. Springer. P. 17−33. (2020). URL: https://doi.org/10.1007/978-3-030-59082-6_2
7. Mitrofanova, O., Sampetova, V., Mamaev, I., Moskvina, A., Sukharev, K. Topic modeling of the Russian corpus of Pikabu posts: Author-topic distribution and topic labeling. In: CEUR Workshop Proceedings, 2813. P. 101−116. (2021).
8. Nikolenko, S.I., Koltcov, S., Koltsova, O. Topic modeling for qualitative studies. In: Journal of Information Science. Vol. 43. P. 1-15. (2017).
9. Khawaji, K., Almubark, I., Almalki, A., Taylor, B.: Similarity Matching for Workflows in Medical Domain Using Topic Modeling. In: 2018 IEEE World Congress on Services (SERVICES). San Francisco, CA, USA. P. 19-20. (2018).
10. Dowling, M., Piepenbrink, A., Saqib, A., Helmi, H.: Machine learning in finance: A topic modeling approach. (2019). URL: https://arxiv.org/ftp/arxiv/papers/1911/1911.12637.pdf
11. Vorontsov, K.V., Voronov, S.O.: Automatic Filtering of Russian Scientific Content using Machine Learning and Topic Modeling. In: International Conference on Computational Linguistics and Intellectual Technologies «Dialogue–2015». Moscow. (2015). URL: http://www.dialog-21.ru/media/2135/vorontsov.pdf
12. Mitrofanova, O.A., Sedova, A.G.: Topic Modeling in Parallel and Comparable Fiction Texts (the case study of English and Russian prose). In: Information Technology and Computational Linguistics (ITCL 2017). Association for Computing Machinery. (2017).
13. Rhody, L.M.: Topic Modeling and Figurative Language. In: Journal of Digital Humanities. Vol. 2(1). (2012). URL: http://journalofdigitalhumanities.org/2-1/topic-modeling-and-figurative-language-by-lisa-m-rhody/
14. Sherstinova, T., Mitrofanova, O., Skrebtsova, T., Zamiraylova, E., Kirina, T.: Topic modeling with NMF vs. expert topic annotation: the case study of Russian fiction. In: L. Martínez-Villaseñor, H. Ponce, O. Herrera-Alcántara, F.A. Castro-Espinoza (Eds.). Advances in Computational Intelligence. 19th Mexican International Conference on Artificial Intelligence, MICAI 2020, Proceedings. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Vol. 12469. Springer. P. 134−151. (2020). URL: https://doi.org/10.1007/978-3-030-60887-3_13

15. Nokel, M.A., Lukashevich, N.V.: Topic models: adding bigrams and taking account of the similarity between unigrams and bigrams. In: Numerical methods and programming. Vol. 16. Issue 2. P. 215–234. (2015).
16. Zamiraylova, E., Mitrofanova, O.: Dynamic topic modeling of Russian fiction prose of the first third of the XXth century by means of non-negative matrix factorization. In: R.Piotrowski's Readings in Language Engineering and Applied Linguistics. PRLEAL-2019: Proceedings of the III International Conference RWTH Aahen University. CEUR Workshop Proceedings. Vol. 2552. P. 321–339. (2019).
17. Greene, D., Cross, J.: Exploring the Political Agenda of the European Parliament Using a Dynamic Topic Modeling Approach. In: Political Analysis. Vol. 25. P. 77–94. (2016). URL: https://arxiv.org/pdf/1607.03055.pdf
18. Dieng, A.B., Ruiz, F.J.R., Blei, D.M.: Topic Modeling in Embedding Spaces. (2019). URL: https://arxiv.org/abs/1907.04907
19. Grootendorst, M.: BERTopic: Neural topic modeling with a class-based TF-IDF procedure. (2022). URL: https://arxiv.org/pdf/2203.05794.pdf
20. Lau, J.H., Newman, D., Karimi, S., Baldwin, T.: Best Topic Word Selection for Topic Labeling. In: COLING'10 Proceedings of the 23rd International Conference on Computational Linguistics, Stroudsburg, PA, Association for Computational Linguistics. P. 605–613. (2010).
21. Aletras, N., Stevenson, M., Court, R.: Labeling Topics using Unsupervised Graph-based Methods. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Baltimore, Maryland, ACL. P. 631–636. (2014).
22. Mei, Q., Shen, X., Zhai, C.: Automatic labeling of multinomial topic models. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD'07, New York, New York, USA, ACM Press. P. 490–499. (2007).
23. Cano Basave, A.E., He, Y., Xu, R.: Automatic Labeling of Topic Models Learned from Twitter by Summarisation. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Stroudsburg, PA, USA, Association for Computational Linguistics. P. 618–624 (2014).
24. Kou, W., Li, F., Baldwin, T.: Automatic Labeling of Topic Models using Word Vectors and Letter Trigram Vectors. In: Proceedings of the Eleventh Asian Information Retrieval Societies Conference (AIRS 2015). № 1. P. 253–264. (2015).
25. Nolasco, D., Oliveira, J.: Detecting Knowledge Innovation through Automatic Topic Labeling on Scholar Data. In: 49th Hawaii International Conference on System Sciences (HICSS), Koloa, HI: IEEE Computer Society. P. 358–367. (2016).
26. Magatti, D., Calegari, S., Ciucci, D., Stella, F.: Automatic labeling of topics. In: ISDA 2009 9th International Conference on Intelligent Systems Design and Applications, Pisa, IEEE. P. 1227–1232. (2009).
27. Lau, J.H., Grieser, K., Newman, D., Baldwin, T.: Automatic Labeling of Topic Models. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Vol. 1. Stroudsburg, PA. Association for Computational Linguistics. P. 1536–1545. (2011).
28. Hulpus, I., Hayes, C., Karnstedt, M., Greene, D.: Unsupervised graph-based topic labeling using DBpedia. In: Proceedings of the Sixth ACM international conference on Web search and data mining WSDM'13. P. 465–474. (2013).

29. Bhatia, S., Lau, J.H., Baldwin, T.: Automatic Labeling of Topics with Neural Embeddings. In: 26th COLING International Conference on Computational Linguistics, 2016. P. 953–963. (2016).
30. Allahyari, M., Pouriyeh, S., Kochut, K., Arabnia, H.R.: A Knowledge-based Topic Modeling Approach for Automatic Topic Labeling. In: International Journal of Advanced Computer Science and Applications. 8(9). P. 335–349. (2017).
31. Mao, X., Hao, Y.-J., Zhou, Q., Yuan, W., Yang, L., Huang, H.: A Novel Fast Framework for Topic Labeling Based on Similarity-preserved Hashing. In: COLING 2016. P. 3339–3348. (2016).
32. Kriukova, A., Erofeeva, A., Mitrofanova, O., Sukharev, K.: Explicit Semantic Analysis as a Means for Topic Labeling. In: Artificial Intelligence and Natural Language Processing: 7th International Conference, AINL 2018, St. Petersburg, Russia, October 17–19, 2018, Proceedings / ed. by D. Ustalov, A. Filchenkov, L. Pivovarova, J. Žižka, Springer, Cham. P. 167–177. (2018).
33. Erofeeva, A., Mitrofanova, O.: Automatic Topic Label Assignment in Topic Models for Russian Text Corpora. In: Structural and Applied Linguistics. Vol. 12. Saint-Petersburg. P. 122−147. (2019). [in Russian].
34. URL: https://elementy.ru/
35. URL: https://www.nltk.org/
36. URL: https://pypi.org/project/pymorphy2/
37. URL: https://radimrehurek.com/gensim/
38. Cheng, X., Yan, X., Lan, Y., Guo, J.: BTM: Topic Modeling over Short Texts. In IEEE Transactions on Knowledge and Data Engineering. Vol. 26. № 12. P. 2928−2941. (2014).
39. Martin, F., Johnson, M.: More Efficient Topic Modeling Through a Noun Only Approach. In: Proceedings of Australasian Language Technology Association Workshop, P. 111−115. (2015).
40. Egger, R., Yu, J.: A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts. In: Frontiers of Sociology. Vol. 7. (2022).
41. Mimno, D., Wallach, H., Talley, E., Leenders, M., McCallum, A. Optimizing Semantic Coherence in Topic Models. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. P. 262−272. (2011).
42. Röder, M., Both, A., Hinneburg, A. Exploring the Space of Topic Coherence Measures. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM'15). P. 399−408. (2015).
43. URL: https://webscraper.io/
44. URL: https://www.selenium.dev/downloads/
45. Bo Zhao. Web Scraping. In: Encyclopedia of Big Data. Springer International Publishing (2017)
46. URL: https://nutch.apache.org/
47. URL: https://pypi.org/project/pyquery/
48. URL: https://www.import.io/
49. URL: https://pypi.org/project/beautifulsoup4/
50. Chapagain, A.: Hands-On Web Scraping with Python. Packt Publishing. (2019).
51. XML Path Language (XPath) Version 1.0 (renderx.com). URL: https://www.renderx.com/~renderx/portal/Tests/xmlspec/xpath.pdf
52. Olteanu, D., Meuss, H., Furche, T., Bry, F. XPath: Looking Forward. In: EDBT 2002 Workshops, LNCS 2490. P. 109–127. (2002).

53. Mikolov, T., Chen, K., Corrado, G.S., Dean J.: Efficient Estimation of Word Representations in Vector Space. (2013). URL: https://arxiv.org/abs/1301.3781
54. Xue, L. et al. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. (2021). URL: https://arxiv.org/abs/2010.11934
55. Narang, Sh. et al.: WT5?! Training Text-to-Text Models to Explain their Predictions. (2020). URL: https://arxiv.org/abs/2004.14546
56. URL: https://openai.com/blog/chatgpt