



Hierarchical Interpretable Topical Embeddings for Exploratory Search and Real-Time Document Tracking

Anastasia Ianina, Moscow Institute of Physics and Technology, Russia

 <https://orcid.org/0000-0002-6822-2801>

Konstantin Vorontsov, Moscow Institute of Physics and Technology, Russia

 <https://orcid.org/0000-0002-4244-4270>

ABSTRACT

Real-time monitoring of scientific papers and technological news requires fast processing of complicated search demands motivated by thematically relevant information acquisition. For this case, the authors develop an exploratory search engine based on probabilistic hierarchical topic modeling. Topic model gives a low dimensional sparse interpretable vector representation (topical embedding) of a text, which is used for ranking documents by their similarity to the query. They explore several ways of comparing topical vectors including searching with thematically homogeneous text segments. Topical hierarchies are built using the regularized EM-algorithm from BigARTM project. The topic-based search achieves better precision and recall than other approaches (TF-IDF, fastText, LSTM, BERT) and even human assessors who spend up to an hour to complete the same search task. They also discover that blending hierarchical topic vectors with neural pretrained embeddings is a promising way of enriching both models that helps to get precision and recall higher than 90%.

KEYWORDS

Additive Regularization of Topic Model, BigARTM, EM-Algorithm, Exploratory Search, Hierarchical Multilevel Search, Information Retrieval, Segmentation-Based Search, Topic Modeling, Topical Embedding

INTRODUCTION

A fast and high-quality retrieval of relevant scientific and technological information becomes an important task in the era of new global challenges, such as a pandemic. The real-time monitoring of domain-oriented papers and news is impossible without fast processing of complicated search queries in order to detect semantically similar text documents without asking the user to formulate new queries. To navigate through a large amount of data query-document matching is not enough for acquiring the full picture of the problem domain which brings us to the idea of switching from known-item to exploratory search.

Exploratory search is a relatively new paradigm in information retrieval. It focuses on learning activities such as understanding new concepts and knowledge acquisition, investigation and analysis (Marchionini, 2006; White & Roth, 2009). Exploratory search setup implies that there is no exact query and unique result of search: a user may not be familiar with the terminology to google with or have no clear road map of the search domain. Current search systems aim to satisfy the needs of

DOI: 10.4018/IJERTCS.2020100107

Copyright © 2020, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

known-item search, but solving exploratory search problems using them may require much effort. A user has to formulate many short queries iteratively, gradually expanding the search domain by repeated steps of querying, browsing search results, and refining the query. The described explorative search demands may be fulfilled by completely different approaches to information seeking. Instead of conventional “googling” with a precisely formulated short text query, we use long text search queries. A document, a set of documents, or a document fragment may play a role of the query. Due to significant differences between exploratory and known-item search, standard Learning to Rank (Liu, 2009) techniques cannot be applied here. Besides, we focus on document-by-document search in which both query and documents are long texts.

We present an exploratory search approach based on probabilistic topic modeling (Blei, 2012; Blei, Ng, & Jordan, 2003; Hofmann, 1999). A probabilistic topic model extracts a set of latent topics from a collection of text documents. It represents each document with a vector of a discrete probability distribution over topics also called a topical embedding. We search for semantically similar documents by simply comparing the vectors of query and documents topical embeddings. This approach is similar to standard full text search based on inverted index with the exception that topics take the place of words. In this work, we are focusing on hierarchical multimodal topical embeddings. The hierarchy induces a cascade search, which starts with a search for generalized topics from low-dimensional vectors, then proceeds to search for more specific topics from higher-dimensional vectors. In experiments, we show that cascading increases both precision and recall of the search.

To get desirable topical representation of documents the topics should also be well interpretable and significantly different from each other. In order to combine these requirements with hierarchy and modalities we use additive regularization for topic modeling (ARTM) (Vorontsov, & Potapenko, 2015). As for technical implementation, we use an effective parallel implementation of the online EM-algorithm from open-source library BigARTM (Frei, & Apishev, 2016).

Compared to the previous work (Ianina, Golitsyn, & Vorontsov, 2017; Ianina, & Vorontsov, 2019), in this paper we continue to explore topical hierarchy and take a step further to merge topical embeddings with neural approaches. Thus, we create models that merge pretrained transformer-based representations and LSTM-based embeddings together with topical vectors and show the effectiveness of such a combination in terms of precision and recall of the search. Furthermore, we expand the experimental design by testing more search setups and more ways to compare topical embeddings. Also, we are moving from the conventional paradigm of document-by-document search and develop the segmentation-based search which divides query and document into thematically uniform text pieces and then compares all the text blocks to each other in order to get more accurate ranking.

One of the main limitations of our previous study is the absence of automatic evaluation techniques that do not require human labeling in order to prepare ground truth for the exploratory search queries. In this work we present a simple yet effective method for evaluating exploratory search quality on the open-sourced dataset of arXiv triplets (Dai et al., 2015). Unlike tech news datasets (habr.com in Russian and techcrunch.com in English) that we were using previously, arXiv triplets already have relevance assessments which makes it possible to evaluate exploratory search quality without additional manual labeling. Also the data from another domain (scientific articles vs. tech news) has different topical structure which possesses additional challenge for multi-criteria topic modeling.

Moreover, in this paper we present our topic-based exploratory search engine (arxiv-search.mipt.ru) for personalized search and recommendation of arXiv papers and discuss the integration of topical embeddings into the developed search service. It is a self-sufficient service that may be regarded as a service that helps scientists communicate and share their findings. The system helps the user to quickly assemble a collection of thematically relevant articles and then use this collection as a query to search for new articles.

The rest of the paper is organized as follows. In section “Probabilistic Topic Modeling” we introduce basic notation, define the additively regularized topic model for exploratory search and discuss a hierarchical topic model for the cascade topic-based search. In section “Topic-based

exploratory search” we describe a topic-based search algorithm and discuss evaluation techniques for an exploratory search task. In section “Experiments: Document-by-document Exploratory Search” we evaluate the search quality on two popular tech news media (TechCrunch in English and Habrahabr in Russian) based on manual human relevance assessments and compare search performed by assessors with our approach. After that in section “Segment-based Exploratory Search” we provide an extension of our method and discuss searching using text segments. We also propose combined embeddings that integrate topical and neural vectors into one searching algorithm and provide a detailed comparison with several baselines including TF-IDF, BM-25, word embeddings, BERT (Devlin, Chang, Lee, & Toutanova, 2018), CNN-based (He, Gimpel, & Lin, 2015) and LSTM-based approaches (Mueller, & Thyagara, 2016). In the end, we provide some technical details for reproducing our results, conclude and discuss the future work and potentials of the proposed topic-based search.

Related Work

Different ways of retrieving and transmitting data were previously studied in application to diverse real-time communication systems including online social networks (Chen, 2015), email services, instant messaging, web-search (Vuong, 2017) and recommender systems (Costa, 2011). Some of these methods propose to use topic modeling for extracting user’s topical activity context. We contribute to this idea by utilizing topic modeling for exploratory search in collective blogs and recommender systems.

However, topic modeling is a relatively new approach in the literature on exploratory search (Feldman, 2012; Jiang, 2014; Rahman, 2013; Singh, Hsu, & Moon, 2013) and even well-detailed surveys don’t mention it at all (Grant et al., 2015; Scherer, von Landesberger, & Schreck, 2013; Veas, & di Sciascio, 2015). The main reason of this is that the exploratory search community has been mostly focused on the user behaviour and understanding most common usage scenarios.

On the other hand, exploratory search is often said to be one of the key applications in topic modeling literature, and searching for semantically similar documents is often used as an extrinsic criterion for the model validation (Andrzejewski, & Buttler, 2011; Wei, & Croft, 2006; Yi, & Allan, 2009). For example, in the paper (Veas, & di Sciascio, 2015) the flexibility and the possibility of visualization and navigating are said to be the key advantages of topic models for exploratory search. At the same time, the authors highlight several disadvantages: difficulties in topic interpretation, intricacy with modifying a topic model as new documents arrive, and high computational costs. These problems inherent in outdated methods have been successfully resolved during the last decade of topic modeling evolution. For instance, online algorithms for training topic models eliminate the necessity of high computational resources for training topic models. Such algorithms perform in linear time on huge data sets (Bassiou, & Kotropoulos, 2014; Mimno, Hoffman, & Blei, 2012; Vorontsov et al., 2015).

In this work, we are focusing on hierarchical topical embeddings for exploratory search and explore the process of cascade gradual search starting from high-level general topics and moving to more specific ones. Hierarchical ARTM is a well-known approach (Chirkova, & Vorontsov, 2016) which was proven to give desirable results for exploratory search task (Ianina, & Vorontsov, 2019). There are other approaches to hierarchical topic modeling including Hierarchical Latent Dirichlet Allocation that makes use of the Nested Chinese Restaurant Process (Blei, Griffiths, & Jordan, 2010) and Gaussian Hierarchical Latent Dirichlet Allocation (Yoshida, Hisano, & Ohnishi, 2020). Both methods rely on Bayesian techniques which makes it hard to build multi-objective topic models due to complicated variational inference if the prior is non-conjugate. Secondly, Dirichlet prior conflicts with natural assumptions of document and topic vectors sparsity. ARTM deals with these problems by setting requirements for a topic model in terms of optimization criteria rather than prior distributions. Due to the aforementioned reasons, we decided to use regularized Expectation-Maximization (EM) algorithm within ARTM framework instead of more complicated Bayesian inference.

Evaluation techniques for exploratory search possess several challenges too. Conventional search engines require multiple queries in order to get the whole picture of the domain and fulfill research-oriented information demand. In contrast, there is no iterative query reformulation in the topic-based exploratory search. Hence, we don't need complicated methods to evaluate the user behavior like those used in (Kraaij, & Post, 2006; Potthast, Hagen, Volske, & Stein, 2013; Shah, Hendaheba, & Gonzalez-Ibanez, 2016).

Real-Time Search Systems: Comparison With Competitors

- **SelkoAI (selko.ai):** This search system aims to identify relevant texts to fulfill search demands of user, solving the same task as we do. Selko is positioned as a tool for analysing tender and specification documents, system requirements, large subcontractor networks and distilling corporate knowledge. The system is able to find relevant quotations and highlight important pieces of text. However, they do not support complicated search queries or searching with several documents simultaneously. Another difference from our search engine is that we focus on facilitating communication and knowledge sharing in diverse scientific communities while Selko creators are determined to simplify processes in industry with automatic text analysis. Selko's indisputable advantage is their integration with Slack, MS Word, Excel and PowerPoint;
- **Deft (hello.shopdeft.com):** It is an online shopping search tool that enables you to find products with specific requirements across different eCommerce websites. The main similarity with our product is that both systems are able to search using a long text query. In addition, both systems are multimodal: you can search with different types of data (categories, tags, pictures, etc.). However, Deft is highly specialised: it was trained to find goods with certain features, not texts in general. On the other hand, our search engine is designed to work with textual data of any domain, but tested mainly on scientific papers.

However, all the mentioned studies have not led to the effective freely available solutions for exploratory search yet.

- **Our Solution (arxiv-search.mipt.ru):** We propose an exploratory search technique and show its effectiveness on a text collection of arXiv articles. Our product aims to fulfill the exploratory search demands of any type including long text queries formulated in natural language. Moreover, a user is able to form thematically coherent collections of articles and perform search using the whole collection instead of a separate document or a search query. We plan to add tools for sharing collections and discussing papers, which turns our search and recommendation service into a full-fledged communication system for researchers. Although our product is currently in the beta-test phase, achieved so far results are promising and may lead the search system to be integrated into various research-oriented communication systems to facilitate fast extraction of relevant information in huge knowledge bases. Another possible application is searching through dialogues, for example work-related correspondence in Slack, Confluence or emails.

Another challenging task that may be solved with our exploratory search engine is document monitoring. An ability to track recently published scientific articles without any additional effort is significant for researchers. Moreover, structuring, analysing and processing huge amounts of scientific literature is a cumbersome task which may be simplified using our topical search. The same problem appears not only in research, but also in news tracking. Nowadays, during pandemic, the importance of news monitoring grows rapidly. Our technology is able to analyze information from different sources and deliver only thematically relevant news articles in a timely manner. Moreover, the same technique may be used to analyse diverse knowledge bases (e.g. medical) to fasten the research and indicate the most valuable pieces of information.

Our technology is based on topic modeling. The tendency to merge topic modeling and exploratory search directions has only been outlined very recently. Our work follows this tendency too and bridges the gap between topic modeling and efficient techniques for exploratory search.

PROBABILISTIC TOPIC MODELING

Let us denote a finite set (collection) of multimodal documents by D and a finite set of modalities by M . Possible examples of modalities include words, bigrams, tags, categories, authors, etc. All the modalities are independent, and each modality m from M is defined by its term dictionary W_m .

Having term frequencies n_{dw} (the number of times the term w appears in the document d) a topic model retrieves a finite set of latent topics T from the text collection. Probabilistic topic model describes the observable term frequencies in each document by a probabilistic mixture of term distribution for the topics $\varphi_{wt} = p(w|t)$ weighted by topic probabilities for the documents $\theta_{td} = p(t|d)$:

$$p(d) = \sum_{t \in T} p(t) p(d) = \sum_{t \in T} \varphi_{wt} \theta_{td}$$

Learning the model parameters $\Phi = (\varphi_{wt})$ and $\Theta = (\theta_{td})$ from the data (n_{dw}) is a problem of stochastic matrix factorization. This problem is ill-posed, since the set of its solutions is generally infinite. In the additive regularization (ARTM) framework, the appropriate solution is found from the regularized log likelihood maximization under normalization constraints (Vorontsov et al., 2015):

$$\sum_{m \in M} \tau_m \sum_{d \in D} \sum_{w \in W_m} n_{dw} \ln \sum_{t \in T} \varphi_{wt} \theta_{td} + \sum_i \tau_i R_i(\Phi, \Theta) \rightarrow \max_{\Phi, \Theta}$$

$$\sum_{w \in W_m} \varphi_{wt} = 1; \varphi_{wt} \geq 0; \sum_{t \in T} \theta_{td} = 1; \theta_{td} \geq 0$$

where R_i are regularization criteria, τ_i are regularization coefficients, and τ_m are modality weights. The regularized variant of the EM-algorithm can be used to solve this optimization problem for any differentiable regularizers (Vorontsov et al., 2015; Vorontsov, & Potapenko, 2015). In our experiments we use the combination of three regularizers that are known to improve both the interpretability of topics and the search quality in terms of precision and recall (Ianina, Golitsyn, & Vorontsov, 2017): decorrelation on term distributions in topics, sparsifying topic distributions in documents and smoothing term distributions on topics.

It was shown that the nested topic structure boosts topic-based exploratory search performance. In a hierarchical topic model each level is represented by a flat topic model. For topic hierarchy building we use a top-down level-by-level strategy proposed in (Chirkova, & Vorontsov, 2016) within the ARTM framework. The model divides topics into subtopics recursively (Zavitsanos, Paliouras, & Vouros, 2011): for each child level we find topic parents from the previous level using interlevel regularization. The regularizer claims parent topics to be well approximated by probabilistic mixtures of children's subtopics:

$$R(\Phi, \Psi) = \sum_{t \in T} \sum_{w \in W} n_{wt} \ln \sum_{s \in S} \varphi_{ws} \psi_{st}$$

where conditional probabilities $\psi_{st} = p(s|t)$ link subtopics s with parent topics t .

TOPIC-BASED SEARCH FOR DOCUMENT MONITORING

A necessity for fast and accurate retrieval of information having a complicated search demand appears in many applications from real-time news monitoring to embedded domain-specific search systems. In such scenarios, the precision and recall of search is paramount while a search procedure should be fast enough to work in a real-time setup. We propose a method that contributes to the evaluation and optimization of exploratory search quality. To achieve good quality together with low execution time, we propose several advances to the approach introduced in (Ianina, & Vorontsov, 2019). First, we designed a new fully-automated technique for evaluating exploratory search quality based on dividing a document into thematically coherent parts. Second, we present segmentation-based search (searching with thematically homogeneous blocks). Third, we incorporate both topical and neural embeddings into one searching algorithm. Finally, we expand our experiments to a huge dataset of arXiv articles and take a further step to move from monitoring technological news in collective blogs to tracking of scientific articles.

This section is organized as follows. First, we recall the basic algorithm of topic-based search. Then we enhance it with topical hierarchies and cascade search. Next we move to evaluation techniques and introduce experiment designs. After it, we discuss several important advances: segmentation-based search and blending with neural embeddings.

Topic-Based Exploratory Search

To make exploratory search a quick one-step procedure, we use searching techniques based on probabilistic topic modeling. First, we train a topic model of the text collection. Then in the inference phase the system gets a long text query q and learns its topic vector $p(t|q)$ in the same way as it is done for the documents in the collection. Next, the system ranks document vectors $p(t|d)$ by their similarity to the query vector and presents top k results to the user, where k is a hyperparameter that can be changed for different search scenarios. The effectiveness of such an approach has been demonstrated both for flat and hierarchical topic models (Ianina, Golitsyn, & Vorontsov, 2017; Ianina, & Vorontsov, 2019). However, hierarchical topic models yield better results due to their ability to gradually narrow the scope of the search.

In case of hierarchical topic-based search both query and document are represented as a sequence of topic vectors, one vector per level. We compare query and document topic vectors level-by-level starting from the top-level vectors of lower dimension and proceeding to the child-level vectors of higher dimensions. We take into account only topics in the child level connected with parent topics that were present with higher than threshold probability both in query and document topic vectors. This helps to discard irrelevant documents gradually specifying the query from general to specific topics. This cascade-style search emulates the humans' natural strategy of information seeking. The elimination of irrelevant documents at top levels increases the precision and speeds up the search process.

Another challenge is connected with the way we compare topic vectors with each other. In the section "Fine-tuning Topic Model" we discuss several similarity measures. Also we are moving from simple document-by-document search (comparing topical vectors of the whole texts) to segment-based search. We divide documents and queries into topically homogeneous batches (segments) and then measure the similarity between all the pieces of a query and document that need to be compared. Then we get the scores of the most similar text blocks and treat their weighted sum as final proximity score. We discuss this approach in more detail in section "Segment-based Exploratory Search".

Evaluation of Topic-Based Exploratory Search

To evaluate exploratory search quality, we introduce two evaluation techniques: the one involving two-stage human assessments of relevance and fully-automatic, based on self-search for document segments.

For the first approach, we constructed a set of long text queries by copy-pasting fragments from the sources outside the collection to avoid overfitting. Each query should be a text explaining the search intent and may contain distinct paragraphs of the articles on the same topic, related citations or even text blocks from Wikipedia. A query emulates a situation when a user is not familiar with a topic of search and tries to aggregate all the known so far information into one document.

Then we asked assessors to complete two tasks. First, they need to find within a given collection as many documents relevant to the query as possible having access to any search tools (Google, Yandex, etc.). Second, assessors are given the documents retrieved by our topic-based search engine and asked to label the documents as relevant or irrelevant to the query. Each query is processed by three assessors to reduce the variance of the result: we accept the document as relevant if the majority of assessors voted for it.

Having relevance assessments, we measure Precision@k and Recall@k for each query. Precision@k is the fraction of relevant documents among the first k documents found. Recall@k is the fraction of relevant documents found out of all the relevant documents. The calculation of Recall requires to know the set of all relevant documents for each query. We are approximating this set from below by joining the documents that were found by all assessors during both stages. Discussed evaluation method makes it possible to compare various topic models without additional assessments.

Although the aforementioned way of evaluation is reliable it cannot be easily expanded to other datasets due to the necessity to invent new queries and perform two-stage human-based assessments. Thus, it is hard to aggregate much queries for evaluation which brings us to the fully-automated, not dataset specific methods of evaluation.

The basic idea is to treat all the documents in the collection as queries and launch document-by-document search. In such a paradigm we will get as many queries as many documents are present in the collection which is more than enough to judge the model performance. Proceeding with this idea, we may cut the document into thematically uniform blocks and use them as the queries. Good models are expected to put the initial document to the top positions in the search results. Such a simple technique may be used as a proof-of-concept fast evaluation when the human assessment is not possible or too expensive.

Datasets

The experiments were based on three datasets: two tech news collections (Techcrunch.com in English and Habrahabr.ru in Russian), and also a dataset of scientific articles from arxiv.org. We decided to test our algorithm on both scientific papers and news from collective blogs to show the good generalizing ability to different data and possible applications of the technology to domain-specific search services and news monitoring.

The Habrahabr collection consists of 175143 articles. Articles contain terms of six modalities: 10552 word unigrams, 742000 word bigrams, 524 authors, 10000 commentators (authors of comments to the articles), 2546 tags, 123 hubs (categories).

The TechCrunch collection consists of 759324 articles. Articles contain terms of four modalities: 11523 word unigrams, 1.2 mln. bigrams (the tail of rare bigrams was deleted), 605 authors and 184 categories.

As for arxiv.org data, we used the dataset released by Dai et al., that contains automatically generated triplets of a query paper, a similar paper that shares keywords, and a dis-similar paper that does not share any keywords. The produced dataset contains 20000 triplets and is based on 963564 articles.

Exploratory Search for News Monitoring in Collective Blogs

The proposed search engine facilitates the usage of communication systems that are available within collective blogs. Fast and efficient retrieval of data makes intensifies the process of exchanging information in a blog. We tested our system on two datasets of articles from collective blogs (Habrahabr

and Techcrunch) and proved that exploratory search engine gives promising results in terms of better understanding, finding and structuring the information in the blog.

We applied the evaluation method based on human assessments to the Habrahabr and TechCrunch collections. For each collection, we composed 100 queries by copying text fragments taken from external sources such as stackoverflow.com, ixbt.com, and other IT-oriented blogs. Thus, each query is several coherent and thematically close paragraphs of text. More information on the composed queries can be found in Table 1.

Table 1. Statistics on the manually composed exploratory search queries

	Habrahabr	TechCrunch
Number of queries	100	100
Min. query length (words)	93	75
Max. query length (words)	455	392
Average query length (words)	262	195
Overlap (number of assessors evaluating the same query)	3	3

According to the aforementioned evaluation method, an assessor was asked to find as much as possible relevant to the query documents within the text collection. Such a task took from 5 to 65 minutes to complete with an average value equal to 30 minutes. Importantly, there is no obvious dependence between the time spent by an assessor and the quality of the search (Fig. 1). On the contrary, our topical search takes no longer than 0.1 sec. and gives even better quality, which makes the algorithm applicable for real-time news monitoring systems.

To exhibit experiment in detail we depicted search results for every query in the form of scatter plot. For each query there are two points on the plot: one for manual human-based search (circles) and one for topic-based search (triangles). On average, precision for our best topic-based model (hierarchical ARTM with 3 levels, for more detail please refer to the last section) is 7% higher while recall is 10% higher than the same metric for manual human search. The difference in precision and

Figure 1. The time (min.) spent by assessors to process each query

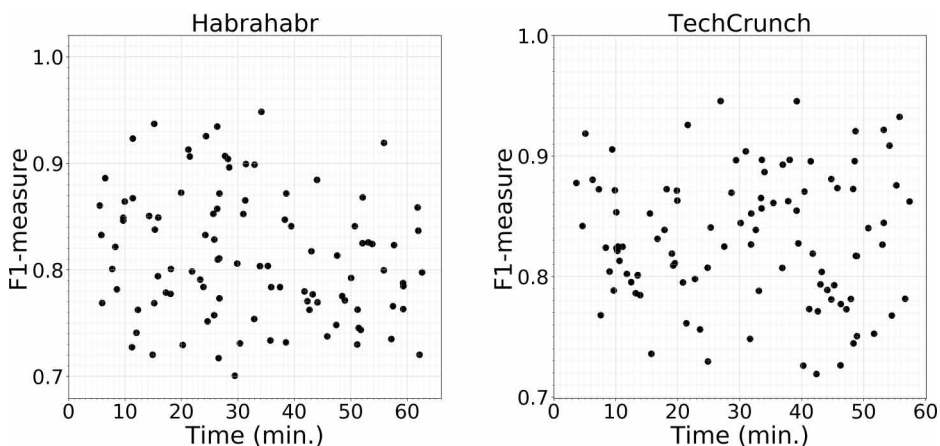
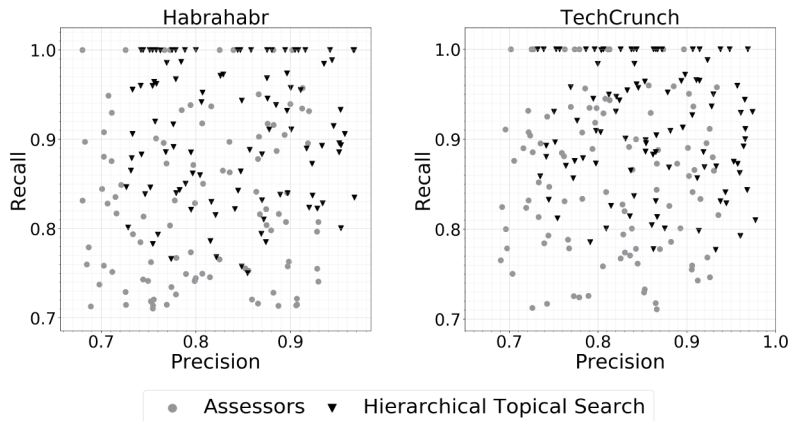


Figure 2. The quality of assessors' and hierarchical topic-based search



recall between assessors search and topic-based search was tested to be statistically significant with Mann–Whitney test. P-values were less than 0.01 for all the experiments.

The highest recall we got for the topic-based search is 1.0 for 26 queries out of 100 for Habrahabr and 29 queries out of 100 for TechCrunch. This means that our search engine is able to find documents that were missed out even by human annotators. Moreover, topic-based search gives a significant advancement in time: it produces an answer in less than 1 second while human assessors spend up to 65 minutes on the same task. Thus, topic-based exploratory search obtains higher precision and recall and performs significantly faster than human assessors.

Segment-Based Exploratory Search for Scientific Articles Tracking

Apart from collective blogs, exploratory search engine may be easily embedded into many real-time research-oriented communication systems. To prove the competitiveness of our algorithm in such applications we utilized a collection of triplets of articles from arXiv released by Dai et al. This dataset contains automatically generated triplets of a query paper, a similar paper with shared keywords, and a dis-similar paper that does not share any keywords. Then we built a hierarchical topic model using only titles and abstracts of the articles from the mentioned dataset. The first layer of the hierarchy was composed according to arXiv categories.

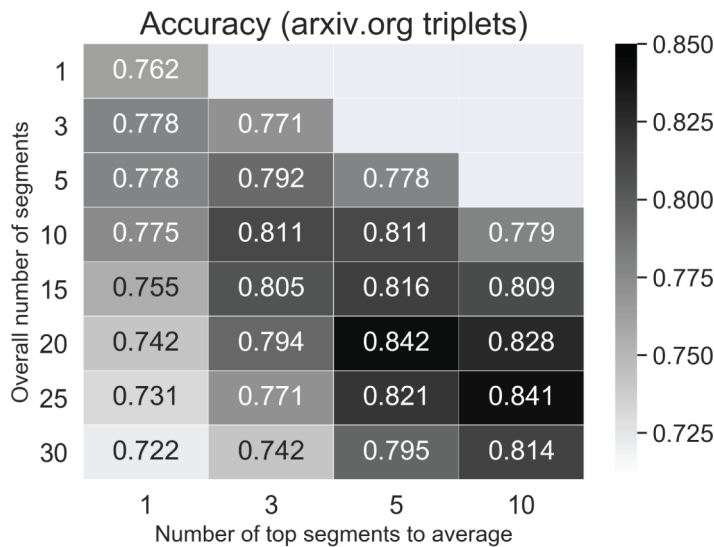
As far as the dataset contains 20,000 triplets, we were able to evaluate 20,000 exploratory search queries in the following way. First, for every query we got a list of documents from the topic search engine. Second, we estimated the number of documents from the topic search output that are also mentioned as relevant to the query in the dataset. Finally, we calculated the average ratio of documents appeared both in the search engine output and second part of the corresponding to the query triplets. The accuracy of search measured in the aforementioned way was 84%.

To escalate the experiment even further, we designed another scheme for measuring documents affinity. Full-text article may be topically heterogeneous and impure which possesses a certain challenge for topical search. To overcome this issue we advise to divide text into topically homogeneous batches and then measure the proximity between all the pieces in the “many-to-many” manner instead of comparing the topic vectors for the whole texts. After such a comparison there are several scenarios of aggregating the final proximity score for the document from the scores of its parts. First, we may judge two documents by its most similar units and give the pair of documents the same score as for its two closest segments. Second, we may average the scores for n most similar segments or even count the weighted average according to the lengths of the segments in case of different-sized segments.

The most difficult part here is to provide good text segmentation algorithm. One of the possible solutions is TopicTiling, an LDA-based text segmentation algorithm (Riedl, & Biemann, 2012). While applying this method and also accustoming it to work on the base of ARTM is out of scope for this article, we propose a simplified version of this approach. We divide text into m parts of the similar size. The size is measured in sentences so that no sentences would be divided between different segments. Then we compare topic vectors of each segment to with the vectors of all the segments in the other text and apply one of the scoring techniques mentioned earlier. Thus, we get as much exploratory search queries as much texts are present in the collection and turn from exploratory search scenario to document-by-document search or even recommendation system mode. In Figure 3 we show search accuracy for different ways of scoring the final segment-based proximity measure between two documents.

We conducted the same experiment for Habrahabr and Techcrunch collections and found out that for these datasets segment search helps to raise quality just a little bit (Fig. 4). Unlikely for arXiv triplets collection, the best models were trained with top- n segments equal to overall number of the segments (cells on the diagonal). This may be caused by the data structure: scientific articles from arxiv.org are much longer and diverse in terms of topical representation, while tech news tends to have 1-2 main topics and much simpler structure which does not need to be divided into pieces.

Figure 3. Accuracy of topic search over segments on the arxiv.org triplets data

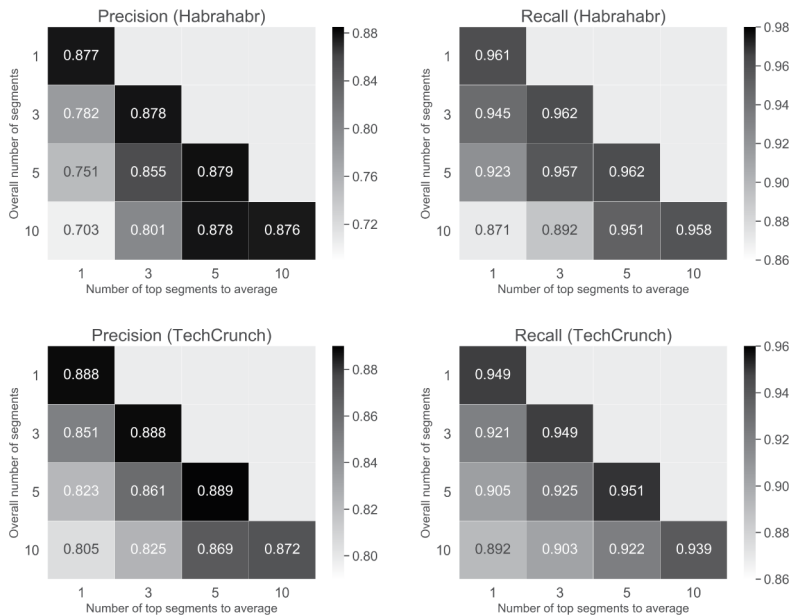


Comparison With Baselines

In this section, we provide a detailed comparison of topic-based search and other approaches applied to news and scientific articles monitoring tasks. The Mann–Whitney test confirmed that the differences between baselines and ARTM-based models are significant (p-values were less than 0.02 for all the experiments). All the results for ARTM-based models and baselines are shown in Fig.5:

1. **TF-IDF and BM-25:** TF-IDF similarity search is a simple but strong competitor because it uses all the information about term frequencies. We used a TF-IDF vectorizer from scikit-learn library (Pedregosa et al., 2011). Meta-information (tags, categories, authors) was taken into account as

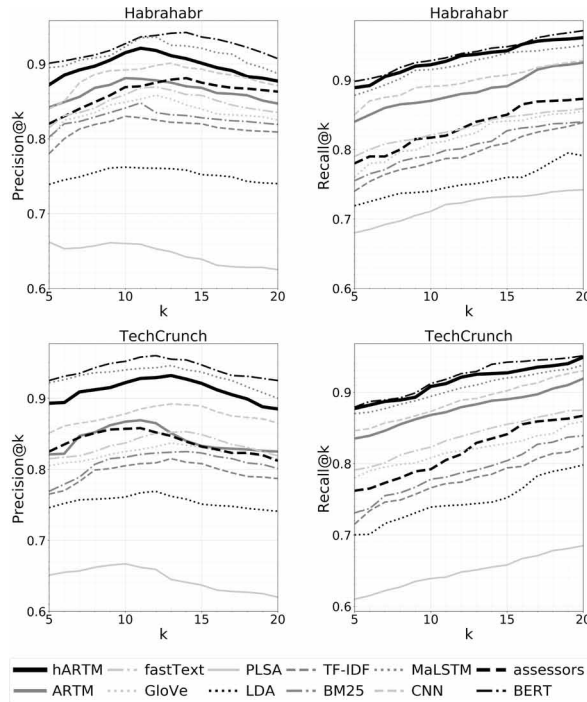
Figure 4. Precision and recall of hierarchical topic search over segments on the Habrahabr and TechCrunch



well as n-grams that we extracted using TopMine (El-Kishky, Song, Wang, Voss, & Han, 2014). Also we used ranking function Okapi BM25 which performs just slightly better than TF-IDF baseline. Topic-based search appeared to perform better than both TF-IDF and BM-25 in terms of precision and recall. Our embeddings have another important advantage: topic vectors contain much less dimensions than TF-IDF representations;

2. **Other topic models (PLSA (Hofmann, 1999) and LDA (Blei, Ng, & Jordan, 2003)):** Both of them perform worse than ARTM-based search. Interestingly, the gap between LDA and ARTM or PLSA and ARTM is much bigger than the same difference between ARTM and other baselines. This brings us to the point that tuning the model with regularizers is a significant step on the way to interpretable topical vectors;
3. **CNN-based approach (He, Gimpel, & Lin, 2015):** In this approach each sentence is modeled with a convolutional neural network that extracts features at multiple levels of granularity. Then the representations are compared using L2 and cosine metric (as well as in our approach). We reproduced the results from the paper for every sentence from our datasets (both for queries and documents) and then aggregate per-sentence representations to get vectors for the whole texts;
4. **Word Embeddings:** Word embeddings are widely used in searching for semantically close documents (Roy, Ganguly, Bhatia, Bedathur, & Mitra, 2018). We tried pretrained *GloVe.840B.300d* for English texts (Pennington, Socher, & Manning, 2014), *RusVectors (skip-gram)* trained on Russian Wikipedia for Russian texts (Kutuzov, & Kuzmenko, 2016) and also *fastText* (Bojanowski, Grave, Joulin, & Mikolov, 2017) vectors. All the mentioned approaches showed comparable with manual human search quality, but hierarchical ARTM outperformed both *GloVe* and *fastText*;
5. **MaLSTM (Siamese adaptation of LSTM) (Mueller, & Thyagara, 2016):** Although this technique is used for measuring sentences similarity, we have expanded its field of applicability to measuring distances between small texts (queries and documents) and used it as a baseline;
6. **BERT (Devlin, Chang, Lee, & Toutanova, 2018):** Recently released pretrained transformer-based models (like BERT, RoBERTa, Transformer-XL, GPT, GPT-2, etc.) have shown promising results on a wide range of tasks. Here we take just one model from the list to use it as a baseline. We took pretrained BERT model from the library *transformers* ('bert-base-uncased') (Wolf

Figure 5. Comparison between search performed by assessors, ARTM-based search and the baselines



et al., 2019) for English texts and BERT from DeepPavlov library (Burtsev et al., 2019) for Russian texts. Then we averaged the [CLS] vectors from BERT over sentences to get document embeddings.

Combining Topic-Based Search With Baseline Models

Our topic-based search outperforms nearly all the baselines and performs just slightly worse than BERT and LSTM baseline in terms of precision and with the same quality in terms of recall. Bridging the gap between the topic-based search and other mentioned baselines may be done by combining both approaches into one searching algorithm.

We constructed a search algorithm by simply blending vector similarities for different approaches into one score in the following way:

$$score(q, d) = \alpha \cdot sim(emb_1(q), emb_1(d)) + (1 - \alpha) \cdot sim(emb_2(q), emb_2(d))$$

where sim is any similarity measure (cosine similarity, for example), $emb_1(q)$ and $emb_1(d)$ are vectors obtained by the first approach (e.g. topic vectors) and $emb_2(q)$ and $emb_2(d)$ are vectors from the second approach (e.g. fasttext embeddings). In such a manner, we combined our best topic model (3-level hierarchical ARTM) with all the baselines and found out that blending *any* baseline with the topic model boosts its performance. The results are presented in Tables 2 and 3. Here we present the result for the best found parameter α ; in case of weak baseline, α may be equal to 0. We highlighted the models that in combination with hARTM (or ARTM) may give better result than both the baseline and hARTM (or ARTM).

It is important to note, that combining hARTM with other baselines results in performance better than the strongest part of the blend (no matter was the topic model much better or slightly worse than the baseline). This result is promising in terms of enhancing close document search with BERT-based embeddings, which established itself as a highly competitive model. Importantly, combining BERT with other baselines from the list gave no significant gain in quality. On the other side, combining hARTM with not so strong baselines (TF-IDF, BM-25) results in no considerable advances: all the blended with TF-IDF models were worse than pure ARTM models.

For arXiv dataset TF-IDF baseline performs much better than the same model for Habrahabr and TechCrunch datasets. It even slightly beats fasttext and GloVe baselines. This may be caused by arXiv triplets dataset design: it was constructed automatically by suggesting that relevant articles

Table 2. Comparison between ARTM-based search, baselines and the blended models (ARTM + baseline, hARTM + baseline) for Habrahabr text collection

Model	Precision			Recall		
	Baseline	Baseline +ARTM	Baseline +hARTM	Baseline	Baseline +ARTM	Baseline +hARTM
TF-IDF	0.809	0.847	0.877	0.839	0.925	0.961
BM-25	0.819	0.847	0.877	0.840	0.925	0.961
GloVe	0.825	0.849	0.877	0.855	0.926	0.961
fasttext	0.835	0.851	0.879	0.859	0.926	0.961
CNN	0.871	0.872	0.881	0.928	0.929	0.963
MaLSTM	0.887	0.887	0.895	0.950	0.950	0.965
BERT	0.907	0.907	0.910	0.971	0.971	0.981
ARTM	0.847	-	-	0.925	-	-
hARTM	0.877	-	-	0.961	-	-
Assessors	0.863	-	-	0.873	-	-

Table 3. Comparison between ARTM-based search, baselines and the blended models (ARTM + baseline, hARTM + baseline) for Techcrunch text collection

Model	Precision			Recall		
	Baseline	Baseline +ARTM	Baseline +hARTM	Baseline	Baseline +ARTM	Baseline +hARTM
TF-IDF	0.787	0.825	0.885	0.824	0.919	0.949
BM-25	0.801	0.825	0.885	0.839	0.919	0.949
GloVe	0.815	0.826	0.885	0.859	0.919	0.949
fasttext	0.819	0.831	0.887	0.875	0.920	0.949
CNN	0.865	0.865	0.890	0.930	0.930	0.950
MaLSTM	0.888	0.888	0.918	0.938	0.938	0.951
BERT	0.890	0.890	0.928	0.951	0.951	0.962
ARTM	0.825	-	-	0.919	-	-
hARTM	0.888	-	-	0.949	-	-
Assessors	0.812	-	-	0.867	-	-

Table 4. Comparison between ARTM-based search, baselines and the blended models (ARTM + baseline, hARTM + baseline) for arXiv triplets text collection

Model	Accuracy (whole text search)			Accuracy (search over top-5 segments)		
	Baseline	Baseline +ARTM	Baseline +hARTM	Baseline	Baseline +ARTM	Baseline +hARTM
TF-IDF	0.750	0.751	0.765	0.749	0.750	0.845
BM-25	0.752	0.754	0.767	0.752	0.752	0.846
GloVe	0.741	0.741	0.762	0.742	0.742	0.842
fasttext	0.745	0.745	0.762	0.747	0.747	0.842
CNN	0.760	0.760	0.763	0.759	0.759	0.843
MaLSTM	0.832	0.832	0.834	0.837	0.837	0.843
BERT	0.843	0.843	0.844	0.845	0.845	0.848
ARTM	0.697	-	-	0.715	-	-
hARTM	0.762	-	-	0.842	-	-

share the same keywords with the query. Simple TF-IDF model catches such patterns very well, even without additional neural network models upon it.

The crucial part of this experiment is to find the best blending coefficient α . To do this we performed grid search with step 0.1 which then was narrowed down to step 0.05 near the optima. Next we present the results of grid search over parameter α for four mixed models (BERT + hARTM, MaLSTM + hARTM, fasttext + hARTM, TF-IDF + hARTM) in Figure 6.

Combining other baselines with each other resulted in no noticeable improvements. Any mixture of baselines without introducing ARTM resulted in performance limited by the strongest model in the blend.

Fine-Tuning Topic Model

In this section, we will share technical details on topic models training and hyperparameter search for our models. Here we present the results only for arXiv dataset. More information on Habrahabr and Techcrunch models fine-tuning may be found in (Ianina, & Vorontsov, 2019).

The process of tuning topic model parameters includes several steps. First of all, we tested several similarity measures between query and documents from the collection: cosine similarity, Euclidean distance, Manhattan distance, Hellinger distance, Kullback-Leibler divergence. The set of similarity measures expands the one from (Mikhailova, Diurdeva, & Shalymov, 2017). For each of them we measured the accuracy of search on arXiv triplets data. We also provide grid search results for segment-based search on arXiv data because it showed much better performance than document-by-document search. For all the collections and experiment designs cosine similarity showed the best results (table 5).

The next challenge is to find an optimal number of levels and topics on each level for the model. Models with more or equal then 4 levels have pure interpretation and lead to very low search quality (precision < 0.72 , recall < 0.65). Flat unilevel models are competitive but still show worse search quality than hierarchical counterparts (Fig.5). This makes us choose between 2-level and 3-level models with different number of topics at each level. To find the best model we need to evaluate the quality of the overall model, not every level in alienation. Our grid search included 75 combinations of parameters but here we present only the best shots for arXiv collection (table 6). The first level of hierarchy was fixed (we used proposed by arxiv.org categories), so we show grid search results only for second and third levels of hierarchy (table 9).

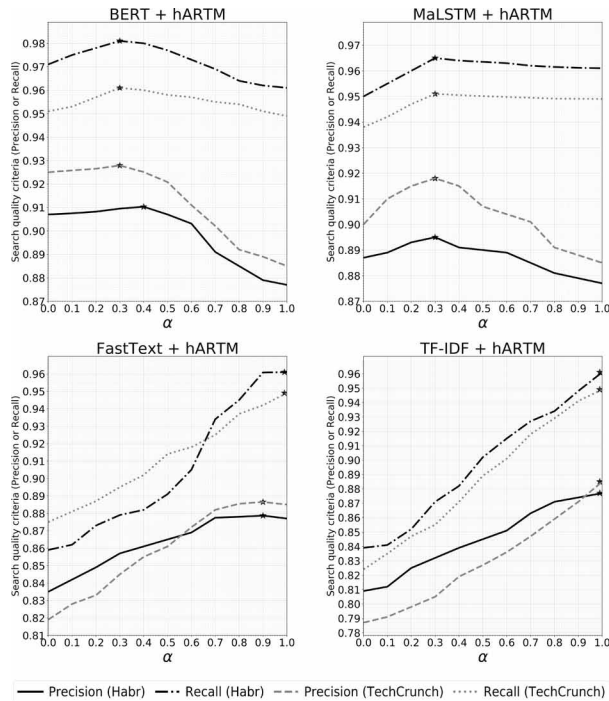
Figure 6. Grid-search over blending parameter α for mixed models BERT + hARTM, MaLSTM + hARTM, fasttext + hARTM, TF-IDF + hARTM

Table 5. Accuracy for topic search by top-n segments with different similarity measures: Euclidean, Cosine, Manhattan, Hellinger, Kullback-Leibler for arXiv (number of segments is fixed and equal to 20)

n	arXiv				
	Eu	cos	Ma	He	KL
1	0.621	0.742	0.703	0.692	0.713
3	0.645	0.794	0.721	0.711	0.732
5	0.657	0.842	0.729	0.727	0.755
10	0.638	0.828	0.725	0.715	0.741

Another important topic model feature is a set of regularizers. We discovered that the decorrelation regularizer contributes most to the search quality, but all other regularizers (τ -sparsing, Φ -smoothing, interlevel connections sparsing) considerably improve the search quality too. Model with no regularization gives much worse result than all the baselines (Fig.5). For more information about regularizer trajectories and topic model fine-tuning please refer to Ianina, & Vorontsov, 2019.

CONCLUSION

In this paper, we investigate exploratory search with long text queries for simplifying the process of search within research-oriented communication systems such as collective blogs and shared scientific knowledge bases. Also exploratory search is applicable to real-time news monitoring, which

Table 6. Accuracy for 3-level hierarchical topic search by top-n segments on arXiv triplets data. Number of segments is fixed and equal to 20. The first level of hierarchy is fixed and based on arXiv categories.

n	40			50					60		
	130	150	170	190	200	210	220	230	250	270	290
1	0.673	0.682	0.704	0.722	0.735	0.742	0.739	0.731	0.719	0.691	0.658
3	0.684	0.697	0.723	0.765	0.784	0.794	0.790	0.782	0.754	0.715	0.672
5	0.691	0.705	0.731	0.820	0.836	0.842	0.838	0.830	0.772	0.745	0.681
10	0.687	0.701	0.728	0.805	0.819	0.828	0.821	0.815	0.768	0.724	0.675

significance has increased now due to the necessity to track news regarding pandemic. Our research directly contributes to creation of a fast and accurate real-time document monitoring system by utilizing low-dimensional topical representations of texts as well as cascade interlevel search with hierarchical topical embeddings. This iterative level-by-level search emulates exploratory search nature with its gradual query rephrasing in order to clarify search intent. We proved topic search competitiveness over manual human-based search in terms of precision and recall and discovered that hierarchical topic model is able to find documents that were missed out even by assessors.

We compared our method with several baselines and showed that our model is much better (TF-IDF, BM-25, fasttext, GloVe, CNN-based methods, PLSA, LDA) or comparable in quality (LSTM, BERT) to them. Moreover, we enhanced our method by blending it with the baselines and discovered that introducing topic vectors to neural models, like LSTM or BERT, increases its quality by up to 3% in terms of precision and recall. This fact designates a possible direction of advancing popular pre-trained neural models in other than exploratory search domains by mixing it with topic embeddings.

Except for document-by-document search, we introduced a segmentation-based search, a simple alternation of the initial method to search over long documents with intricate topical structure. We showed its effectiveness on a search task within scientific articles from arxiv.org.

The main contribution of this paper compared with the previous study is the implementation of the topic-based search as a self-sufficient product (available at arxiv-search.mipt.ru). It is a personalized search system for tracking arXiv articles. The successful performance of the proposed technology on three diverse datasets with various experiment designs makes it possible to consider the technology for application to real-time research-oriented communication systems to simplify the process of knowledge acquisition and discovery. The proposed exploratory search engine may be used to facilitate data investigation in collective blogs and scientific communities, track scientific or news articles in real-time manner, organize and effectively search for domain-specific information (manuals, sets of requirements, lists of tasks) in work-related communication services, like Slack, Confluence or email storages.

ACKNOWLEDGMENT

The work was supported by the Government of the Russian Federation (Agreement N° 05.Y09.21.0018).

REFERENCES

- Andrzejewski, D., & Buttler, D. (2011). Latent Topic Feedback for Information Retrieval. *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '11*, 600–608.
- Bassiou, N. K., & Kotropoulos, C. L. (2014). Online PLSA: Batch updating techniques including out-of-vocabulary words. *IEEE Transactions on Neural Networks and Learning Systems*, 25(11), 1953–1966. doi:10.1109/TNNLS.2014.2299806 PMID:25330420
- Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4), 77–84. doi:10.1145/2133806.2133826
- Blei, D. M., Griffiths, T. L., & Jordan, M. I. (2010). The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the Association for Computing Machinery*, 57(2), 1–30. doi:10.1145/1667053.1667056
- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3, 993–1022.
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 135–146. doi:10.1162/tacl_a_00051
- Burtsev, M. (2019). *Open-source AI library DeepPavlov*. Retrieved from: <https://github.com/deepmipt/DeepPavlov>
- Chen, H., Jin, H., & Zhang, F. (2015). CBL: Exploiting community based locality for efficient content search service in online social networks. *IEEE Transactions on Services Computing*, 10(6), 868–878. doi:10.1109/TSC.2015.2501821
- Chirkova, N. A., & Vorontsov, K. V. (2016). Additive regularization for hierarchical multimodal topic modeling. *Journal Machine Learning and Data Analysis*, 2(2), 187–200. doi:10.21469/22233792.2.2.05
- Costa, A., & Roda, F. (2011). Recommender systems by means of information retrieval. *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, 1-5. doi:10.1145/1988688.1988755
- Dai, A.M., Olah, C., & Le, Q.V. (2015). *Document embedding with paragraph vectors*. CoRR abs/1507.07998
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding*. arXiv preprint arXiv:1810.04805
- El-Kishky, A., Song, Y., Wang, C., Voss, C. R., & Han, J. (2014). Scalable topical phrase mining from text corpora. *Proceedings of the VLDB Endowment*, 8(3), 305–316. doi:10.14778/2735508.2735519
- Feldman, S. E. (2012). The answer machine. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 4(3), 1–137. doi:10.2200/S00442ED1V01Y201208ICR023
- Frei, O., & Apishev, M. (2016). Parallel non-blocking deterministic algorithm for online topic modeling. In *Proceedings of the AIST Conference (Analysis of Images, Social networks and Texts)*, (vol. 661, pp. 132-144). Springer International Publishing Switzerland, Communications in Computer and Information Science (CCIS). doi:10.1007/978-3-319-52920-2_13
- Grant, Ch. E., George, C. P., Kanjilal, V., Nirkhiwale, S., Wilson, J. N., & Wang, D. Zh. (2015) Topic-Based Search, Visualization, and Exploration System. *Proceedings of the FLAIRS Conference*, 43–48.
- He, H., Gimpel, K., & Lin, J. (2015). Multi-perspective sentence similarity modeling with convolutional neural networks. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1576-1586. doi:10.18653/v1/D15-1181
- Hofmann, Th. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, (pp. 50-57). New York: ACM.
- Ianina, A., Golitsyn, L., & Vorontsov, K. (2017). *Multi-objective topic modeling for exploratory search in tech news*. In *Communications in Computer and Information Science, AINL-6: Artificial Intelligence and Natural Language Conference* (Vol. 789). Springer International Publishing.

- Ialina, A., & Vorontsov, K. (2019). Regularized Multimodal Hierarchical Topic Model for Document-by-Document Exploratory Search. In *Proceeding of The 25st Conference of FRUCT (Finnish-Russian University Cooperation in Telecommunications) Association, the seminar on Intelligence, Social Media and Web (ISMW)*. Helsinki, Finland: IEEE. doi:10.23919/FRUCT48121.2019.8981493
- Jiang, T. (2014). Exploratory search: a critical analysis of the theoretical foundations, system features, and research trends. In *Library and Information Sciences* (pp. 79–103). Springer. doi:10.1007/978-3-642-54812-3_7
- Kraaij, W., & Post, W. (2006). Task based evaluation of exploratory search systems. In *Proceedings of SIGIR workshop on Evaluating Exploratory Search Systems (EESS)*, (pp. 24–27). ACM.
- Kutuzov, A., & Kuzmenko, E. (2016). WebVectors: a toolkit for building web interfaces for vector semantic models. In *International Conference on Analysis of Images, Social Networks and Texts*, (pp. 155–161). Springer.
- Liu, T. Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3), 225–331. doi:10.1561/15000000016
- Marchionini, G. (2006). Exploratory search: From finding to understanding. *Communications of the ACM*, 49(4), 41–46. doi:10.1145/1121949.1121979
- Mikhailova, E., Diurdeva, P., & Shalymov, D. (2017). N-gram based approach for text authorship classification: Metric selection. *International Journal of Embedded and Real-Time Communication Systems*, 8(2), 24–39. doi:10.4018/IJERTCS.2017070102
- Mimno, D., Hoffman, M., & Blei, D. (2012). *Sparse stochastic inference for latent Dirichlet allocation*. arXiv preprint arXiv:1206.6425
- Mueller, J., & Thyagara, A. (2016). Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16*, (pp. 2786–2792). AAAI Press.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., & Vanderplas, J. et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2825–2830.
- Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543. doi:10.3115/v1/D14-1162
- Potthast, M., Hagen, M., Volske, M., & Stein, B. (2013). Exploratory search missions for TREC topics. *Proceedings of the EuroHCIR*, 1033, 7–10.
- Rahman, M. (2013). Search engines going beyond keyword search: A survey. *International Journal of Computers and Applications*, 75(17), 1–8. doi:10.5120/13200-0357
- Riedl, M., & Biemann, C. (2012). TopicTiling: a text segmentation algorithm based on LDA. In *Proceedings of ACL 2012 Student Research Workshop*, (pp. 37–42). Association for Computational Linguistics.
- Roy, D., Ganguly, D. S., Bhatia, S., Bedathur, S., & Mitra, M. (2018). Using word embeddings for information retrieval: How collection and term normalization choices affect performance. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*, (pp. 1835–1838), New York: ACM. doi:10.1145/3269206.3269277
- Scherer, M., von Landesberger, T., & Schreck, T. (2013). Topic Modeling for Search and Exploration in Multivariate Research Data Repositories. *Research and Advanced Technology for Digital Libraries: International Conference on Theory and Practice of Digital Libraries, TPD 2013*, 370–373.
- Shah, Ch., Hendaheba, Ch., & Gonzalez-Ibanez, R. (2016). Rain or shine? Forecasting search process performance in exploratory search tasks. *Journal of the Association for Information Science and Technology*, 67(7), 1607–1623. doi:10.1002/asi.23484
- Singh, R., Hsu, Y. W., & Moon, N. (2013). Multiple perspective interactive search: A paradigm for exploratory search and information retrieval on the web. *Multimedia Tools and Applications*, 62(2), 507–543. doi:10.1007/s11042-011-0910-2

International Journal of Embedded and Real-Time Communication Systems

Volume 11 • Issue 4 • October-December 2020

Tai, K. Sh., Socher, R., & Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (Vol. 1, pp. 1556-1566). Association for Computational Linguistics. doi:10.3115/v1/P15-1150

Tan, Y., & Ou, Z. (2010). Topic-weak-correlated latent Dirichlet allocation. *The 7th International Symposium Chinese Spoken Language Processing (ISCSLP)*, 224-228.

Veas, E. E., & di Sciascio, C. (2015). Interactive Topic Analysis with Visual Analytics and Recommender Systems. *The 2nd Workshop on Cognitive Computing and Applications for Augmented Human Intelligence (CCAAHI2015), International Joint Conference on Artificial Intelligence (IJCAI)*.

Vorontsov, K. V., Frei, O., Apishev, M., Romov, P., Suvorova, M., & Yanina, A. (2015). Non-bayesian additive regularization for multimodal topic modeling of large collections. In *Proceedings of the 2015 Workshop on Topic Models: Post-Processing and Applications*, (pp. 29-37). New York: ACM.

Vorontsov, K. V., & Potapenko, A. A. (2015). Additive regularization of topic models. *Machine Learning. Special Issue on Data Analysis and Intelligent Optimization with Applications*, 101(1), 303–323.

Vuong, T., Jacucci, G., & Ruotsalo, T. (2017). Proactive information retrieval via screen surveillance. *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1313-1316.

Wei, X., & Croft, W. B. (2006). LDA-based document models for ad-hoc retrieval. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 178-185.

White, R. W., & Roth, R. A. (2009). *Exploratory Search: Beyond the QueryResponse Paradigm*. In *Synthesis Lectures on Information Concepts, Retrieval, and Services*. Morgan and Claypool Publishers.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... Brew, J. (2019). *Huggingface's transformers: State-of-the-art natural language processing*. ArXiv, abs/1910.03771

Yi, X., & Allan, J. (2009). A Comparative Study of Utilizing Topic Models for Information Retrieval. *Lecture Notes in Computer Science*, 5478, 29–41. doi:10.1007/978-3-642-00958-7_6

Yoshida, T., Hisano, R., & Ohnishi, T. (2020). *Gaussian Hierarchical Latent Dirichlet Allocation: Bringing Polysemy Back*. arXiv preprint arXiv:2002.10855

Zavitsanos, E., Paliouras, G., & Vouros, G. A. (2011). Non-parametric estimation of topic hierarchies from texts with hierarchical Dirichlet processes. *Journal of Machine Learning Research*, 12, 2749–2775.

Anastasia Ianina graduated from Moscow Institute of Physics and Technology (MIPT) with a Master's degree in Computer Science in 2018. Now she is a PhD student at MIPT working on Natural Language Processing tasks (topic modeling for exploratory search and article recommendation). Previously she was working at Yandex, Machine Intelligence Laboratory at MIPT and self-driving department at Lyft.

Konstantin Vorontsov was born in Moscow, USSR in 1971, graduated from Moscow Institute of Physics and Technology in 1994, defended PhD in 1999, doctor of physical and mathematical sciences since 2010, professor of Russian Academy of Sciences since 2015. Now he is head of Machine Intelligence Laboratory at MIPT. Full details on personal page <http://www.MachineLearning.ru/wiki?title=User:Vokov>.