

# Методы полуавтоматической суммаризации подборок научных статей

Власов Андрей Валериевич

Московский физико-технический институт  
Факультет управления и прикладной математики  
Кафедра интеллектуальных систем

Научный руководитель д.ф.-м.н. К. В. Воронцов

Выпускная квалификационная работа магистра

Москва,  
2020 г.

## Цель

создание технологии полуавтоматического реферирования тематических подборок научных статей (MAHS, Machine-Aided Human Summarization).

## Задачи

- 1 Декомпонировать задачу полуавтоматического реферирования на подзадачи машинного обучения.
- 2 Сформировать обучающую выборку «подборка → реферат» по коллекции научных статей.
- 3 Разработать алгоритмы обучения для основных подзадач:
  - генерация сценария реферата;
  - ранжирование фраз-кандидатов для продолжения реферата.
- 4 Оценить качество и выбор модели суммаризации.

## Прототип пользовательского интерфейса для написания реферата

SEARCH
About FAQ Sergey Kukharensko

PAPERS
SUMMARIZATION

SEARCH IN COLLECTION
Most recent Most quoted

### Collection of papers

- ▲ **BanditSum: Extractive Summarization as a Contextu...**  
25 SEP 2018 Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, Jacki
- ▲ **A Survey on Neural Network-Based Summarization...**  
19 MAR 2018 Yue Dong
- ▼ **SummaRuNNer: A Recurrent Neural Network based...**  
13 NOV 2016 Ramesh Nallapati, Feifei Zhai, Bowen Zhou
- ▲ **A Deep Reinforced Model for Abstractive Summariz...**  
11 MAY 2017 Romain Paulus, Caiming Xiong, Richard Socher
- ▲ **Neural Extractive Summarization with Side Informa...**  
14 APR 2017 Shashi Narayan, Nikos Papasarasantopoulos, Shay B. Cohen
- ▲ **Ranking Sentences for Extractive Summarization...**  
12 FEB 2018 Shashi Narayan, Shay B. Cohen, Mirella Lapata
- ▲ **Get To The Point: Summarization with Pointer-Gen...**  
14 APR 2017 Abigail See, Peter J. Liu, Christopher D. Manning

### Summary

**BanditSum**

A novel method for training neural networks to perform single-document extractive summarization without heuristically-generated extractive labels.

We call our approach BANDITSUM as it treats extractive summarization as a contextual bandit (CB) problem, where the model receives a document to summarize (the context), and chooses a sequence of sentences to include in the summary (the action).

A policy gradient reinforcement learning algorithm is used to train the model to select sequences of sentences that maximize ROUGE score.

The aim of this literature review is to survey the recent work on neural-based models in automatic text summarization.

We examine in detail ten state-of-the-art neural-based

### Recommended phrases

SummaRuNNer, a Recurrent Neural Network (RNN) based sequence model for extractive summarization of documents and show that it achieves performance better than or comparable to state-of-the-art.

Our model has the additional advantage of being very interpretable, since it allows visualization of its predictions broken up by abstract features such as information content, salience and novelty.

Another novel contribution of our work is abstractive training of our extractive model that can train on human generated reference summaries alone, eliminating the need for sentence-level extractive labels.

### Prompters

Result

Experiment

Theory

Dataset

Annotate

Idea

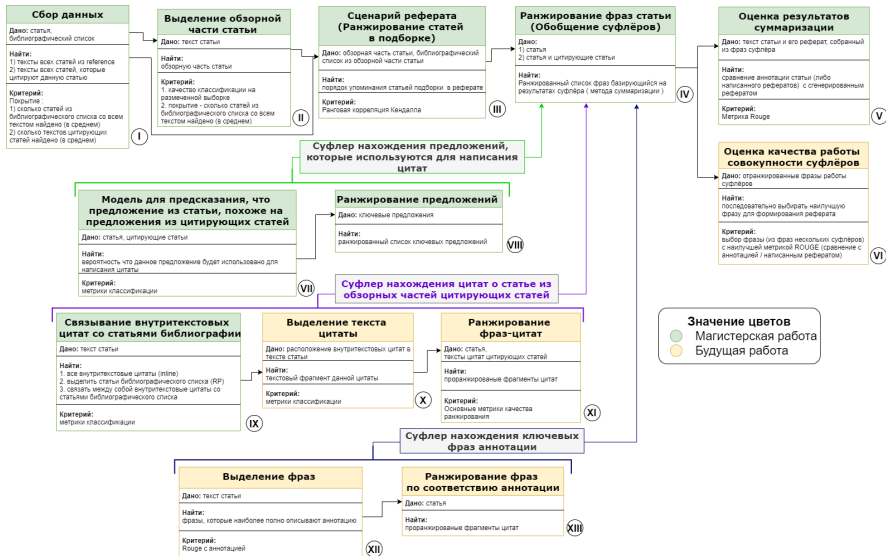
Motivation

Method

Conclusion

Citation

# Потоки обработки данных в системе реферирования



- 1 Формирование обучающей выборки:
  - сбор данных, выделение статей библиографии;
  - выделение внутритекстовых ссылок;
  - связывание внутритекстовых ссылок со статьями библиографии;
  - выделение обзорной части статьи.
- 2 Формирование сценария реферата (задача ранжирования).
- 3 Построение фраз-кандидатов для продолжения реферата:
  - выделение ключевых фраз в тексте статьи;
  - выделение текста цитаты;
  - выделение фраз об основных аспектах данной статьи;
  - ранжирование фраз-кандидатов.
- 4 Оценивание качества суммаризации.

## Обучающая выборка для решения задачи суммаризации

$$D = \{(x_i, y_i) : i = 1, \dots, n\}$$

$x_i$  – неупорядоченный список статей библиографии  $i$ -ой статьи,  
 $y_i$  – конкатенация обзорных сегментов, извлеченных из  $i$ -ой статьи.

## Задача формирования обучающей выборки по коллекции статей

- **Дано:** коллекция научных статей.
- **Найти** для каждой статьи:
  - список статей библиографии;
  - внутритекстовые ссылки на статьи библиографии;
  - обзорную часть статьи.

## Основные характеристики датасета S2ORC

- 1 Определены внутритекстовые цитаты
- 2 Определены статьи библиографического списка
- 3 Соединены внутритекстовые цитаты и статьи библиографии
- 4 Разделены тексты на параграфы, часть из которых имеет заголовков
- 5 Наличие метаданных и аннотаций.



Структура датасета

Всего статей	81.1M
с PDF	28.9M (35.6%)
с библиографией	27.6M (34.1%)
с полным текстом из GROBID	8.1M (10.0%)
с полным текстом из LaTeX	1.5M (1.8%)
с аннотациями издателя	73.4M (90.4%)
с PubMed идентификаторами	21.5M (26.5%)
с ArXiv идентификаторами	1.7M (2.0%)
с ACL идентификаторами	42k (0.1%)

Статистика по статьям S2ORC

## Постановка задачи

- **Дано:** структурированный текст статьи, поделённый на секции, каждая секция поделена на абзацы.
- **Найти:** подмножество абзацев, образующих обзорную часть статьи.
- **Критерий:**
  - 1 Качество классификации на размеченной выборке.
  - 2 Покрытие — доля статей из библиографического списка, процитированных в выделенной обзорной части.
- **Набор данных:** S2ORC.

## Обучающая выборка

Множество абзацев классифицированы на два класса:

- 1 обзорные разделы «Introduction», «Related work», «Background».
- 2 остальные.



## Признаки

- Густота ссылок =  $\frac{\text{количество цитат}}{\text{количество символов в абзаце}}$ ;
- Количество последовательных предложений, включающих не менее 1 цитаты;
- Позиция секции в статье =  $\frac{\text{порядковый номер секции}}{\text{количество секций}}$ ;
- Усредненная позиция внутритекстовых цитат в каждой цитате =  $\frac{\frac{1}{n} \sum_{i=1}^n \text{позиция } i \text{ внутритекстовой цитаты}}{\text{количество символов секции}}$ ,  
где  $n$  — количество внутритекстовых цитат абзаца.

## Результаты вычислительного эксперимента

Модель	Accuracy	Покрытие
Базовая модель (количество ссылок)	61%	56.7%
Gradient Boosting	82%	59.6%

## Постановка задачи

- **Дано:** неупорядоченная подборка статей
- **Найти:** порядок упоминания этих статей в реферате
- **Критерий:** коэффициент ранговой корреляции Кендалла на обучающей выборке
- **Набор данных:** S2ORC

## Формирование обучающей выборки

$i$ -й объект генерируется по  $i$ -й статье из коллекции:

- $x_i$  — множество статей, процитированных в обзорной части;
- $y_i$  — порядок упоминания этих статьей в обзорной части.

## Признаки

- год публикации статьи
- количество цитирований статьи / авторов статьи
- цитируемость журнала или конференции
- наличие индексации (ACL, Pibmed, DOI, arXiv)

## Постановка задачи ранжирования для формирования сценария

- $a(d, c, w)$  — модель ранжирования статей  $c$ , цитируемых в  $d$
- $w$  — вектор параметров модели
- $c \prec c'$  — порядок цитирования « $c$  раньше  $c'$ » в статье  $d$ .

$$Q(w) = \sum_d \sum_{c \prec c'} \log(1 + \exp(a(d, c', w) - a(d, c, w))) \rightarrow \min_w$$

## Результаты вычислительного эксперимента

Модель	$\tau$
Базовая модель (год выхода статьи)	0.10
CatBoost с попарным подходом	0.48

Пусть имеется сценарий реферата — ранжированная подборка.

**Суфлёр** — это функция нашей системы, которая выводит пользователю ранжированный список фраз о заданной статье из сценария для продолжения реферата.

## Примеры суфлёров

- 1 Ключевые фразы из аннотации и текста данной статьи.
- 2 Фразы, используемые в других статьях при цитировании данной статьи.
- 3 Фразы из данной статьи, похожие на фразы из цитирующих статей.
- 4 Фразы об основных аспектах данной статьи:
  - метод • результат • идея • эксперимент • вывод • и т.п.

**Задача построения суфлёра**, который ранжирует фразы из данной статьи, похожие на фразы из цитирующих её статей.

## Постановка задачи

- **Дано:**
  - ① статья
  - ② множество цитирующих её статей
- **Найти:** ранжированный список фраз
- **Критерий:** качество суммаризации по метрике ROUGE, которая сравнивает сгенерированный реферат с несколькими рефератами, написанными людьми
- **Набор данных:** CL-SciSumm

Цитирующая статья	Исходная статья
Japanese Named Entity Recognition Using Structural Natural Language Processing	Global Features
Rishel Sankar Graduate School of Information Science and Technology, University of Tokyo rsankar@isip.t.u-tokyo.ac.jp	Sasha Kurzban Graduate School of Information Science and Technology, University of Tokyo skurzba@isip.t.u-tokyo.ac.jp
<p><b>Abstract</b></p> <p>This paper presents an approach that uses structural information for Japanese named entity recognition (NER). Our NER system is based on Support Vector Machine (SVM), and extracts two types of structural information: co-occurrence, coreference relations, syntactic, semantic and syntactic features, which are obtained from structural analysis. We evaluated our approach on NER data and obtained a higher F1 measure than existing approaches that do not use structural information. We also conducted experiments on NER, NE tags and on NE annotated sentences and compared their accuracy.</p>	<p><b>4.2 Global Features</b></p> <p>Context from the whole document can be important in classifying <b>ссылочный промежуток</b> may appear in abbreviated form when it is mentioned again later. Previous work deals with this problem by correcting inconsistencies between the named entity classes assigned to different occurrences of the same entity (Borthwick, 1999; Mikheev et al., 1998). We often encounter sentences that are highly ambiguous in themselves, without some prior knowledge of the</p>
<p>On <b>Цитирующее предложение (цитата)</b> use, various NER systems have explored global information and reported their effectiveness. In (Malouf, 2002; Chieu and Ng, 2002), information about features assigned to other instances of the same token is utilized. (Ji and Grishman, 2005) uses the information obtained from coreference analysis for NER. (Mohit and Hwa, 2005) uses syntactic features in building a semi-supervised NE tagger.</p>	<p><b>Ссылка</b></p> <p><b>Ссылочный промежуток 1:</b> Previous work deals with this problem by correcting inconsistencies between the named entity classes assigned to different occurrences of the same entity</p>

## Структура датасета CL-SciSumm

**Ссылочный фрагмент (промежуток)** — предложение исходной статьи, которое используется для составления цитаты.

Суфлёр строит ранжированный список фраз из данной статьи, похожие на фразы из цитирующих её статей:

## 1 Вычисление набора признаков:

### 1 Косинусная мера близости между векторными представлениями:

- предложений модели (TF-IDF, LDA, LSI, HDP)
- предложений (усредненных по словам) (W2V, WMD)

### 2 Сравнения набора N-грамм:

- количество общих биграмм
- ROUGE (-1, -2, -l f1)
- Sequence Matcher ratio

### 3 Позиционные признаки:

- Позиция предложения в исходной статье =  $\frac{\text{порядковый номер предложения}}{\text{количество предложений в исходной статье}}$
- Позиция предложения в секции исходной статьи
- Позиция секции в исходной статье

## 2 Обучение классификатора, предсказывающего, является ли предложение из исходной статьи ссылочным промежуточком:

- Random Forest
- SVM
- XGBoost
- CatBoost
- MLP

### 3 Суммаризация: Ранжирование по вероятности предложений из исходной статьи

- топ-1 реферат (отбор наилучших предложений в реферат до достижения ограничения слов)
- топ-3 реферат (наилучший по ROUGE реферат из трех сгенерированных, как топ-1 реферат)

### 4 Оценка результатов суммаризации

- $r$  - реферат, написанный аннотатором датасета CL-SciSumm
- $s$  - реферат, сгенерированный системой

$$ROUGE_N(s) = \frac{\text{количество перекрывающихся } N\text{-грамм}(r, s)}{\text{количество } N\text{-грамм в } r}$$

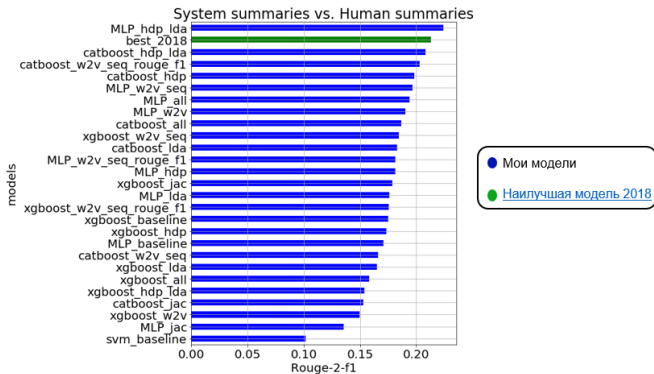
$$ROUGE_L = \frac{(1 + \beta^2)R_{LCS}P_{LCS}}{R_{LCS} + \beta^2 P_{LCS}}, \quad R_{LCS} = \frac{LCS(X, Y)}{|X|}, \quad P_{LCS} = \frac{LCS(X, Y)}{|Y|}$$

где  $LCS(X, Y)$  длина максимальной общей подпоследовательности рефератов  $X$  и  $Y$ ,  $\beta = \frac{P_{LCS}}{R_{LCS}}$ .

Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. 2004

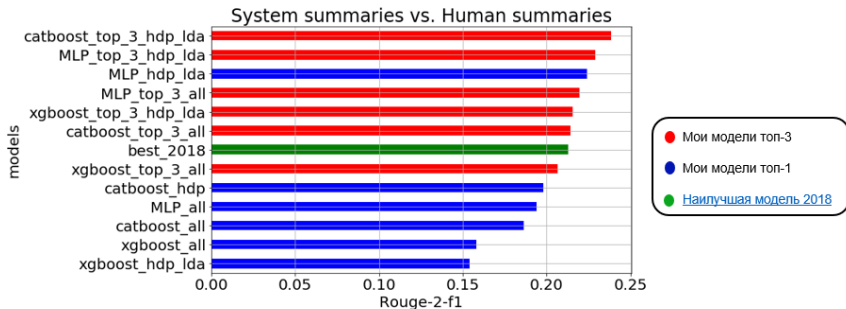


# Результаты для топ-1 рефератов



- наилучшие признаки, которые дают улучшение для всех классификаторов:
  - w2v ● wmd ● lda ● hdp ● seq\_match
- наилучшие классификаторы на различных наборах признаков:
  - многослойный перцептрон ● catboost

# Результаты для топ-3 рефератов



- Метрика ROUGE для топ-3 рефератов лучше, чем для топ-1 рефератов на 15-19% в зависимости от классификатора
- Лучший классификатор для генерирования топ-3 рефератов совпадает с топ-1 рефератов

- 1 Предложена декомпозиция задачи полуавтоматического реферирования подборок научных статей на 13 подзадач классификации, ранжирования и суммаризации текстов.
- 2 Создан прототип системы полуавтоматического реферирования.
- 3 В экспериментах показано преимущество предложенных методов по сравнению с известными решениями.

## Направления дальнейших исследований

- 1 Реализовать полный набор суфлёров
- 2 Предложить критерий качества суммаризации с учётом всех суфлёров и действий пользователя
- 3 Внедрить полуавтоматическое реферирование в поисково-рекомендательный сервис <https://arxiv-search.mipt.ru/>

## Сгенерированный топ-1 реферат

News article headlines are a rich source of paraphrases; they tend to describe the same event in various different ways, and can easily be obtained from the web. We compare two methods of aligning headlines to construct such an aligned corpus of paraphrases, one based on clustering, and the other on pairwise similarity-based matching. We show that the latter performs best on the task of aligning paraphrastic headlines. In the study described in this paper, we make an effort to collect Dutch paraphrases from news article headlines in an unsupervised way to be used in future paraphrase generation. News article headlines are abundant on the web, and are already grouped by news aggregators such as Google News. We aim to build a high-quality paraphrase corpus. Where previous work has focused on aligning news-items at the paragraph and sentence level (Barzilay and Elhadad, 2003), we choose to focus on aligning the headlines of news articles. Part of the data in the DAESO-corpus consists of headline clusters crawled from Google News Netherlands in the period April–August 2006. Using headlines of news articles clustered by Google News, and finding good paraphrases within these clusters is an effective route for obtaining pairs of paraphrased sentences with reasonable precision. We have shown that a cosine similarity function comparing headlines and using a back off strategy to compare context can be used to extract paraphrase pairs at a precision of 0.76.

## Реферат аннотатора

This paper talks about Clustering and matching headlines for automatic paraphrase acquisition. For this purpose it is necessary to have a monolingual corpus of aligned paraphrased sentences. We compare two methods of aligning headlines to construct such an aligned corpus of paraphrases, one based on clustering, and the other on pair wise similarity-based matching. News article headlines are abundant on the web, and are already grouped by news aggregators such as Google News. It is clear that k-means clustering performs well when all unclustered headlines are artificially ignored. In the more realistic case when there are also items that cannot be clustered, the pair wise calculation of similarity with a back off strategy of using context performs better when we aim for higher precision.