Generating Sport Summaries: A Case Study for Russian

Valentin Malykh[1(✉)], Denis Porplenko[2], and Elena Tutubalina[1,3]

[1] Kazan Federal University, Kazan, Russian Federation
valentin.malykh@phystech.edu
[2] Ukrainian Catholic University, Lviv, Ukraine
[3] National Research University Higher School of Economics, Moscow, Russian Federation

Abstract. We present a novel dataset of sports broadcasts with 8,781 games. The dataset contains 700 thousand comments and 93 thousand related news documents in Russian. We run an extensive series of experiments of modern extractive and abstractive approaches. The results demonstrate that BERT-based models show modest performance, reaching up to 0.26 ROUGE-1F-measure. In addition, human evaluation shows that neural approaches could generate feasible although inaccurate news basing on broadcast text.

Keywords: Sport broadcast · Summarization · Russian language · Neural networks

1 Introduction

Nowadays, sports content is viral. Some events are trendy, watched by billions of people. Every day, thousands of events take place in the world that interest millions of people. The audience of online sports resources is quite broad. Even if a person watched a match, he is interested in reading the news, as there is more information in the news. Therefore, there is a great need for human resources to write this news or several for each sporting event. Media companies have become interested in cutting costs and increasing the quality of news [5]. Some well-known publications such as the Associated Press, Forbes, The New York Times, Los Angeles Times make automatic (or "man-machine marriage" form) generating news in simple topics (routine news stories) and will also introduce research to improve the quality of such news [5].

In this paper, we present the first attempt to apply state-of-the-art summarization models for automatic generation of sports news in Russian using textual comments. Our dataset is provided by a popular website `sports.ru`. The dataset consists of text comments which describe a game at a particular point in time. We explore several state-of-the-art models for summarization [10,13]. We note that recent research often evaluates models on general domain CNN/Daily Mail dataset [8], while the performance on domain-specific datasets, i.e. sport-related,

W. M. P. van der Aalst et al. (Eds.): AIST 2020, LNCS 12602, pp. 149–161, 2021.
https://doi.org/10.1007/978-3-030-72610-2_11

is not well studied. Our dataset differs greatly from the one used for training state-of-the-art approaches. Existing English CNN/DM/XSumm datasets [9,17,21] consist of public news and articles as input documents. The output summaries for them can contain either small summaries in one sentence (first sentence or news headline), summaries written by the same person after reading the input news or summaries which are obtained by the heuristic algorithmic way [16]. Our dataset contains broadcasts at the input and news as output sequences, written by different people in a different context.

In this work, we focus on the generation of news instead of the short summary with a game's results. Sports news describes the score of the match game, extend it with the details of the game including injuries, interview of coaches after the event, the overall picture or situation (e.g., "Dynamo with 14 points takes sixth place in the ranks."). Our contribution is two-fold: (i) we present a new dataset of sports broadcasts and also (ii) we provide the results of current summarization approaches to the news generation task, concluded with a human evaluation of the produced news documents.

2 Related Work

We would like to mention some works that use reinforcement training to solve abstract summarization problems. Paulus et al. proposed to solve the problem of summary generation in two stages: in the first, the model is trained with a teacher, and in the second, its quality is improved through reinforcement learning [20]. Celikyilmaz and co-authors presented a model of co-education without a teacher for two peer agents [3].

For the Russian language, there are recent works on abstract summarization, which mainly appeared in the last year. First of all, this is the work of Gavrilov et al. [4], which presented a corpus of news documents, suitable for the task of generating headings in Russian. Also, in this work, was presented the Universal Transformer model as applied to the task of generating headers; this model showed the best result for Russian and English. Some other works [7,24,25] was based on presented a corpus of news, which use various modifications of models based on the encoder-decoder principle.

Next, we will consider works that are directly related to news generation as a summary of the text. Here we want to highlight a study by the news agency Associated Press [5]. Andreas Graefe, in this study, talks in detail about the problems, prospects, limitations, and the current state in the direction of automatic generating news. In the direction of generating the results of sports events, there is little research. The first is a relatively old study based on the content selection approach performed on a task-independent ontology [1,2].

3 Dataset

For the experiments, we used data provided by http://sports.ru. The data provided in the form of two text entities, these are the comments from a commentator who describes an event and the news. In the provided set, there are 8781 sporting events, and each event contained several comments and news; the news was published both before and after the sporting event. A description of each entity, examples, statistical characteristics, and preprocessing steps are described below.[1]

3.1 Broadcast

The provided data consists of a set of comments for each sporting event. Figure 1 shows examples of the comments. The comments contain various types of information:

- Greetings. E.g. “Добрый день” (“Hello”), “хорошего дня любителям футбола” (“good day to the football fans”);
- General information about the competition/tournament/series of games. E.g. “подниматься в середину таблицы” (“rise to the middle of the table”), “пятый раз в истории сыграет в групповом турнире” (“the fifth time in history [it] will play in the group tournament”);
- Information about what is happening in the game/competition. E.g. “пробил в ближний угол” (“[he] strucks into the near corner”, “удар головой выше ворот” (“a head hit above the goal”);
- Results/historical facts/plans/wishes for the players. Ex: “0:3 после сорока минут” (“[score in the game is] 0:3 after forty minutes”), “не забивали голов в этом сезоне” (“[they] didn’t score a goal this season”).

Also, each comment of a game contains additional meta-information: (i) the match identifier, (ii) the names of the competing teams (e.g. Real Madrid, Dynamo, Montenegro), (iii) the name of the league (Stanley Cup, Wimbledon. Men. Wimbledon, England, June), (iv) the start time of the game, (v) an event

Добрый день!
Наш сайт поздравляет всех, кто прошедшей зимой с нетерпением считал дни до старта российской премьер-лиги.
Наша первая текстовая трансляция чемпионата 2009 поможет Вам проследить за событиями, которые произойдут на стадионе “Локомотив”, где одноименная команда принимает гостей из “Химок”.

Good day!
Our site congratulates everyone who counted the days before the start of the Russian Premier League.
Our first text broadcast will help you follow the events that will take place at the Lokomotiv Stadium, where the team of the same name hosts guests from “Khimki”.

Fig. 1. Examples of comments for a same sport game.

[1] The owner of the dataset approved its publication, so it will be released shortly after the paper is published.

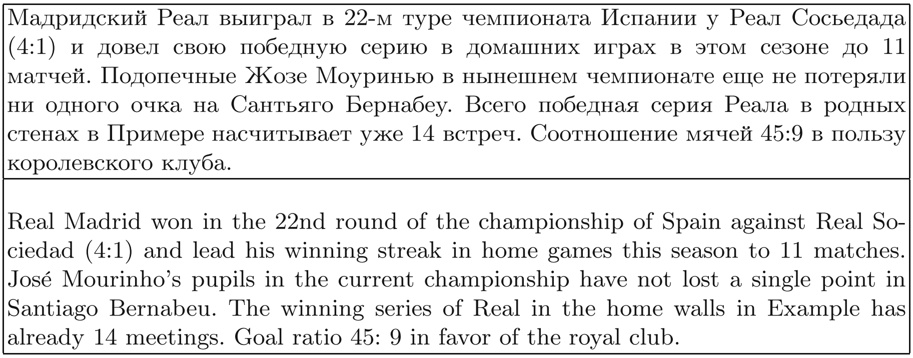
type (e.g. yellow card, goal), (vi) the minute in the sports game when the event occurred, and (vii) comment time.

We sorted by time and merged all the comments for one game into one large text and called it a broadcast. Before merging we cleaned the text from non-text info (like HTML tags). There are 722,067 comments in the dataset, of which we constructed 8,781 broadcasts. In the current study, we used only text information from the commentary. We would like to emphasize that some comments and news contain advertising or non-relevant information. In our experiments, we use some filtering described in Sect. 6.

3.2 News

News is a text message that briefly describes the events and results of a sports game. Unlike a brief summary, news can be published before and after the match. The news that took part in the experiments contains the following information:

- Comments and interviews of a player or a coach. E.g. “ребята отнеслись к матчу очень серьезно, я доволен” (“the guys took the game very seriously, I am satisfied”), “мы проиграли потому, что...” (“we lost because...”);
- Events occurring during the game. E.g. “боковой арбитр удалил полузащитника” (“side referee removes midfielder”), “полузащитник «Арсенала» Санти Касорла забил три мяча” (“«Arsenal» midfielder Santi Cazorla scores three goals”);
- General information about the competition/tournament/series of games. E.g. “сборные словакии и парагвая вышли в 1/8 финала” (“national teams of Slovakia and Paraguay reached the 1/8 finals”), “выполнить задачу на турнир выйти в четверть финал” (“[they] complete the mission for the tournament to reach the quarter-finals.”);
- Game results. E.g. “таким образом, счет стал 1:1” (“thus the score was 1: 1”), “счет в серии: 0-1” (“the score in the series: 0–1”);



Мадридский Реал выиграл в 22-м туре чемпионата Испании у Реал Сосьедада (4:1) и довел свою победную серию в домашних играх в этом сезоне до 11 матчей. Подопечные Жозе Моуриньо в нынешнем чемпионате еще не потеряли ни одного очка на Сантьяго Бернабеу. Всего победная серия Реала в родных стенах в Примере насчитывает уже 14 встреч. Соотношение мячей 45:9 в пользу королевского клуба.

Real Madrid won in the 22nd round of the championship of Spain against Real Sociedad (4:1) and lead his winning streak in home games this season to 11 matches. José Mourinho’s pupils in the current championship have not lost a single point in Santiago Bernabeu. The winning series of Real in the home walls in Example has already 14 meetings. Goal ratio 45: 9 in favor of the royal club.

Fig. 2. A sample news document for a sport game.

Each news document contains additional meta information: (i) title, (ii) time, (iii) sport game identifier. The data set contains 92,997 news documents. Figure 2 shows a sample news document.

4 Metrics

In our research, we used the ROUGE metric, which compares the quality of the human and automatic summary [12]. We have chosen this metric because it shows good statistical results compared to human judgment on the DUC datasets [19]. In ROUGE, a reference is a summary written by people, while the hypothesis (candidate) is an automatic summary. When calculating ROUGE, we can use several reference summaries; this feature makes more versatile comparisons (than comparing with only one reference summary). This metric bases on algorithms that use n-gram overlapping: the ratio of the number of overlap n-gram (between reference and candidate) to the total number of the n-gram in the reference.

$$ROUGE\text{-}N = \frac{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count_{match}\left(gram_n\right)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count\left(gram_n\right)}, \quad (1)$$

where n stands for the length of the n-gram, $gram_n$, and $Count_{match}(gram_n)$ is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries. We used a particular case of ROUGE-N: ROUGE-1, and ROUGE-2. We also use ROUGE-L, a specific case of the longest common sub-sequence in a reference and a hypothesis. The ROUGE metric could be considered a Recall, so we could extend it to Precision and F-measure in common manner. We denote them ROUGE-N-R, −P, and −F respectively.

5 Models

5.1 Oracle

This model generates an extractive summary that has the most value ROUGE between broadcast and news. We used the greedy search algorithm: we found the value of custom rouge (ROUGE-1-F + ROUGE-2-F) between each sentence from the broadcast and all the sentences in the news and selected the top 40 sentences. This algorithm stopped working in one of two cases: (1) the number of sentences is greater than the requested upper threshold (40 sentences) or (2) adding the next sentences does not increase ROUGE.

In this series of experiments, we decided to reduce the incoming sequence (broadcast) by applying the extractive approach techniques. We decided to apply the *Oracle* model, which selects sentences (in our case, 40 top sentences) from the broadcast, which have the maximum news relevance (gold reference). We used "bert-base-multilingual-uncased" model as encoder with $max\_pos = 512$. In this experiment, we trained two models that get a short output (the result of the Oracle model) as an input: (i) *OracleA* - a model trained with parameters as in section with parameters as model BertSumAbs and (ii) *OracleEA* - model trained with parameters as model BertSumExtAbs.

5.2 Neural Models

An approach that we utilize in our research proposed [13] is called PreSumm. Yang Liu and Mirella Lapata in [13] proposed to encode the whole document, keeping its sense, to generate a compact conclusion. The abstract summarization is reduced to the neural machine translation problem: an encoder contains a trained model (BERT), and a decoder contains a randomly initialized BERT. If training two models - one pre-trained and other randomly initialized - at the same time with the same parameters, then one model can be overfitting, and the second can underfitting, or vice versa. The authors use different training parameters (learning rates and warmups).

BertSumAbs is a model for abstractive summarization. This model uses the NMT approach, pre-trained BERT as an encoder, and the randomly initialized transformer in the decoder. We used the abstract BertSumAbs model with bert-base-multilingual-uncased[2] as an encoder and randomly initialized BERT in the decoder. We also trained model *RuBertSumAbs* based on RuBERT model for encoder.

BertSumExtAbs is also using the NMT approach, but unlike BertSumAbs it uses the pre-trained BertSumExt on the extractive summarization task as an encoder. In this experiment, we used the double fine-tune stages for encoder: firstly, we are fine-tuning the model to the extractive summarization task, then we fine-tuning that model on the abstractive task [13]. For the first fine-tune stage, we used BERT "bert-base-multilingual-uncased", learning rate is $2 \cdot 10^{-3}$, dropout is 0.1, max_pos is 512, and 10000 warmup steps. Next, the trained model was used as an encoder for the abstractive summarization task with the same parameters as for BertSumAbs model. Inspecting the preliminary experiments, we realized that max_pos - truncates our incoming sequences; the model trains only 512 of the first broadcast tokens. According to the distribution of token lengths, this is quite small sequences to getting all vital information from the broadcast.

So we introduce *BertSumExtAbs1024* model. For it we used training parameters from previous experiments, with max_pos increased to 1024 and "bert-base-multilingual-uncased" as an encoder model. This model showed better results, compared with previous models with $max\_pos = 512$. The model trained 30,000 steps showed the best results. We hypothesize that we need to select sequences of higher dimensions or reduce the size of the input sequences while preserving the essential meanings and ideas of the entire broadcast.

We found out that generated news incorporated text that does not apply to the sports events; this text in common cases located at the end of the news. In this experiment, we eliminate sentences with such text. We call this model *BertSumAbsClean*. We deleted sentences that contained one of the specific words ("таблица"/"здесь."/"онлайн-трансляц") in broadcasts (source sequence) as well as in the news (target sequence). In broadcasts, a sentence with these words advertises online broadcasts on this site. In the news, sentences that contained

[2] https://github.com/google-research/bert/blob/master/multilingual.md.

these words referred either to another page or to a visualization (images/table); this information did not help to generate news and increase input sequences. Often such sentences have advertised mobile applications.

There are several works [16,26] showing good results on relatively large amounts of data, and weak results on the small ones. Our dataset contained nearly 8000 data samples, so we decided to increase our data corpus using augmentation. For this experiment, we decided to increase the amount of our data ten times using synthetically generated data based on existing data samples.

Our models for augmentation are based on the work [27]. The idea is to replace words in a broadcast on the words from another model. There are two models: thesaurus and static embedding. Both models receive a word at the input and return a list of words size of 10. Each word in the set is similar to the incoming word: with higher word similarity the higher position in the returned list is correlated. Next, we use a geometric distribution to select two parameters for the model (for each generated sample): the number of words to be replaced in the broadcast and the index of word in the returned list for each word, that should be replaced.

Next we describe two models sing augmentation techniques. For the first model, called *AugAbsTh*, we used a similarity graph model of words from a Russian language thesaurus project called Russian Distributional Thesaurus[3] [18]. The word similarity graph is a distribution thesaurus for the most frequent words of the Russian language, obtained on the embedding of words, which was built on the body of texts of Russian books (12.9 billion words). For the second model, called *AugAbsW2V*, we used word2vec [15] for vectorizing words and the cosine of the angle between the vectors, as a metric for word similarity. As a pre-trained model, we used a model trained on the Russian National Corpus [11]. Since there were several news items related to one broadcast in our dataset, we did not augment the news. We have set random news that was written after a sports game; for broadcasts with less than ten news, we repeated the news. Thus, we got two datasets with sizes of about 80,000 broadcasts each.

5.3 Extractive Models

In our experiments, we apply two models to the implementation of the TextRank algorithm, the first based on the PageRank[4] algorithm, the other on the Gensim TextRank.[5] These approaches differ in the similarity function of two sentences. The PageRank-based model uses cosine distance between vectors of sentences (below we describe the algorithm of getting vector of the sentence); the Gensim model based on the BM25 algorithm. For the PageRank model, we preprocessed the broadcast text. We split the text into sentences using NLTK [14], then split it to words, and lemmatize each word using pymystem3 [22]. To vectorize the words, we used two different pre-trained models: the word2vec model, trained on

[3] https://nlpub.mipt.ru/Russian_Distributional_Thesaurus.

[4] http://bit.ly/diploma_pagerank.

[5] https://radimrehurek.com/gensim/index.html.

the Russian National Corpus [11] (*PageRank W2V*), and the FastText model, trained on a news corpus [23] (*PageRank FT*). To vectorize a sentence, we average all the word vectors. Then, we calculated the cosine distance between all sentences, build a similarity matrix, converted it to a graph, and applied the PageRank algorithm. The PageRank parameters remained by default from the library. After that, we selected sentences with a maximum page ranking and formed a summary from them. For the *TextRank* model, we used raw broadcast text and parameter $ratio = 0.2$, which adjust the percentage size of the summary compared to the source text (20% of the sentences from the source text will be in summary).

We also used another extractive approach called the *LexRank* algorithm. We use our implementation of LexRank algorithm[6] since the existing implementation has memory issues. We chose the top 10 sentences, the rest of the parameters used were set by default.[7]

6 Experiments

For our experiments, we selected the news document with minimum length and the ones written after the match. The results of the experiments are presented in Table 1.

All algorithms from the TextRank experiments showed very similar results: ROUGE-1-F is the same in all its variants. ROUGE-2 showed the worst results among all ROUGE metrics. *TextRank* works better than *PageRank W2V* and *PageRank FT*: ROUGE-1-F less than 0.1. We could conjecture that is due to different people describe sports commentary and news in different formats, styles and situations: the commentator describes the emotionally sporting game online, with details; the author of the news, calmly and dryly reports the results or takes an interview from the game player or coach. Therefore, these texts, when comparing, use different words, word forms, expressions. This property leads to an insignificant ROUGE metric based on the overlapping of common words.

The experiment with the *LexRank* approach showed the same result as the TextRank for ROUGE-1-F, higher by 0.01 in ROUGE-L-F and lower by 0.02 in ROUGE-2-F. This algorithm is very similar to TextRank; therefore their results are pretty close to each other.

We cut off long broadcasts and news for neural models: the maximum length of the broadcast was 2500, and the length of the news 200. To train the model, we used the NVIDIA Tesla P100 video card and split our dataset into shards, with a size of 50 examples. Next, we will describe different experiments with different approaches, parameters.

The best ROUGE results show the model that has been trained in 50,000 steps. We noticed that the model tended to overfit after 50,000 iterations. The ROUGE value in this experiment was the highest compared to all extractive

[6] https://github.com/DenisOgr/lexrank/pull/1/files.
[7] https://pypi.org/project/lexrank/.

experiments: ROUGE-1-F is greater than 0.13, ROUGE-2-F is 0.07, ROUGE-L-F is 0.13; we made this comparison using the value of the models that showed the highest result, except for the Oracle model.

RuBertSumAbs model showed lower ROUGE results compared to the BertSumAbs model: ROUGE-1 F is lower by 0.04, ROUGE-2-F, and ROUGE-L F are less by 0.01 and 0.05, respectively. We assume that the reason for this is the encoder model: "bert-base-multilingual-uncased" generates contextual vectors better than "RuBERT".

The model *BertSumExtAbs1024* did not show significant improvements, compared to the best models, where we used $max\_pos = 512$. The values of ROUGE-1-P and ROUGE-L-P are higher by 0.01. We want to note that this model was trained for 30,000 steps, and this is 20,000 steps lower than BertSumAbs. We have seen that increasing the input sequence from 512 to 1024 did not produce significant improvements, according to the ROUGE metric. We assume that this property of ROUGE metric: the overlap words between summary and "gold" news do not increase while increasing the input sequence in PreSumm approaches.

The *Oracle* models from this experiment showed approximately the same results (among themselves): ROUGE-1-F, ROUGE-2-F are the same, and ROUGE-L-F is 0.01 more for OracleEA than for OracleA. Therefore, we will make a comparison of other models with the best model for ROUGE in this experiment. Also, these models were trained on different numbers of steps: OracleA at 30,000 and OracleEA at 40,000, respectively.

As for *BertSumAbsClean* we have noticed that metric ROUGE decreased compared to previous experiments, and we got the best model by ROUGE, the trained model only 20000 steps (this is the lowest number of training steps in our experiments). Comparing to the oracle models ROUGE-1-F and ROUGE-L-F metrics are less than 0.02, and ROUGE-2-F is less than 0.004. We hypothesize that deleted sentences were increasing our ROUGE: "gold" and generated news had advertisements and "referred" sentences, and they increase the ROUGE.

Both the augmented models indicated significant improve performance of our task and was training on 100000 steps. However, the *AugAbsTh* model showed a higher ROUGE score than the *AugAbsW2V*: the scores of ROUGE-1-F, ROUGE-2-F, and ROUGE-L-F are higher by 0.04. This indicates that using synonyms to generate words in our task is more robust and significantly better than using word2vec embeddings. AugAbsTh model has outperformed the best previous model BertSumExtAbs1024 as well as the oracle models. The score of ROUGE-1-F and ROUGE-2-F are higher by 0.05 and ROUGE-L-F scores higher by 0.07 compared to BertSumExtAbs1024. Comparing with the OracleA model, AugAbsTh has ROUGE-1-F score higher on 0.01, ROUGE-2-F on 0.03, and ROUGE-L on 0.04 accordingly. This suggests that increasing the data corpus using real or "similar to real" data will increase the performance of the models.

Table 1. ROUGE scores from all models.

Method/Metric	*ROUGE-1*			*ROUGE-2*			*ROUGE-L*		
	P	R	F	P	R	F	P	R	F
Oracle	0.2	0.22	0.21	0.02	0.02	0.02	0.18	0.20	0.19
OracleA	0.23	**0.30**	0.25	0.09	0.13	0.10	0.22	**0.28**	0.22
OracleEA	0.23	0.29	0.25	0.09	0.12	0.10	0.21	0.27	0.21
PageRank W2V	0.06	0.15	0.08	0.00	0.00	0.00	0.06	0.15	0.06
PageRank FT	0.06	0.13	0.08	0.00	0.00	0.00	0.06	0.13	0.06
TextRank	0.05	0.17	0.08	0.00	0.01	0.00	0.05	0.16	0.06
LexRank	0.07	0.10	0.08	0.00	0.00	0.00	0.07	0.09	0.06
BertSumAbs	0.19	0.25	0.21	0.07	0.10	0.07	0.17	0.23	0.18
RuBertSumAbs	0.14	0.26	0.17	0.04	0.10	0.06	0.13	0.24	0.13
BertSumExtAbs	0.18	0.25	0.2	0.06	0.10	0.07	0.17	0.23	0.17
BertSumExtAbs1024	0.20	0.25	0.21	0.07	0.10	0.08	0.18	0.23	0.18
BertSumAbsClean	0.18	0.24	0.19	0.07	0.07	0.06	0.18	0.18	0.16
AugAbsTh	**0.26**	**0.30**	**0.26**	**0.12**	**0.14**	**0.13**	**0.23**	**0.28**	**0.25**
AugAbsW2V	0.23	0.26	0.22	0.08	0.10	0.09	0.19	0.25	0.21

7 Human Evaluation

The effectiveness of ROUGE was previously evaluated [6,12] through statistical correlations with human judgment on the DUC datasets [19]. To judge the news, we asked five annotators to rate the news by four dimensions: relevance (selection of valuable content from the source), consistency (factual alignment between the summary and the source), fluency (quality of individual sentences), and coherence (collective quality of all sentences). We chose five random news from different models (with the different number of training steps). We chose only abstractive models since these models have shown better performance compared to the extractive ones. The summary score for each dimension is obtained by averaging the individual scores. The comparison results are displayed in Table 2.

Table 2. Human evaluation results.

Model/Metric	Relevance	Consistency	Fluency	Coherence
OracleA	0.46	0.60	0.78	0.78
BertSumExtAbs1024, 10k steps	0.36	0.58	0.56	0.46
BertSumExtAbs1024, 30k steps	0.26	0.34	0.70	0.54
BertSumAbs	0.28	0.50	0.56	0.58
BertSumAbsClean	0.28	0.56	0.72	0.70

Analyzing the data from Table 2, we want to emphasize that the values of Fluency and Coherence are generally higher than the values of Relevance and

Consistency. This suggests that the models from our experiments generate pretty high-quality and linked sentences, but worse select events from the broadcast. The highest scores of Fluency and Coherence have OracleA and BertSumAbsClean models. News generated by BertSumAbs and OracleA models have the highest scores of Relevance. Concluding this experiment, we did not observe any visual relationships between human judgment and the ROUGE metric. We also want to notice that we received some comments from annotators regarding the quality of the news. Most of the comments were aimed at the fact that the quality of the sentences is pretty good, but the news does not review important events or reviews non-existent events from the broadcast.

8 Conclusion

In this paper, we have investigated the task of generating news based on sports commentary with state-of-the-art approaches for summarization. The main challenges of this novel dataset are: 1) the average size of one document differs greatly from texts in existing general domain corpora [9,17,21], 2) domain of texts is sport-related that includes diverse information about a match game, 3) news are written in a language other than English. Unlike expectations, the state-of-the-art neural models show modest performance for our task. We obtained the maximum value 0.26 by ROUGE-1-F score using BERT as an encoder. We found out that increasing data corpus using text argumentation based on thesaurus gives a substantial improvement: we increase data per ten times, and the ROUGE-1-F score has gone up on 0.05 in the absolute difference in comparison with best no augmentation score.

The quantitative analysis opens up several future research directions. First, we plan to increase the number of documents in our dataset by transforming the comments of the sporting event from audio sources, which are more popular than textual. Second, the effective application of transformers as an encoder suggests continuing experiments with other types of transformers, like GPT-2 or different BERT-based architectures. Finally, we could explore a custom evaluation metric based on the main characteristics of a game: overall score or main events.

We hope that this work will foster the research in text generation in Russian and for narrow domain texts in general.

Acknowledgements. The work of the first author was funded by RFBR, project number 19-37-60027. The final work on the manuscript carried out by Elena Tutubalina was funded by the framework of the HSE University Basic Research Program and Russian Academic Excellence Project “5–100”.

References

1. Bouayad-Agha, N., Casamayor, G., Mille, S., Wanner, L.: Perspective-oriented generation of football match summaries: old tasks, new challenges. ACM Trans. Speech Lang. Process. **9**(2), 3:1–3:31 (2012). https://doi.org/10.1145/2287710.2287711

2. Bouayad-Agha, N., Casamayor, G., Wanner, L.: Content selection from an ontology-based knowledge base for the generation of football summaries. In: Proceedings of the 13th European Workshop on Natural Language Generation, pp. 72–81. Association for Computational Linguistics, Nancy, France, September 2011. https://www.aclweb.org/anthology/W11-2810
3. Celikyilmaz, A., Bosselut, A., He, X., Choi, Y.: Deep communicating agents for abstractive summarization (2018)
4. Gavrilov, D., Kalaidin, P., Malykh, V.: Self-attentive model for headline generation. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) ECIR 2019. LNCS, vol. 11438, pp. 87–93. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-15719-7_11
5. Graefe, A.: Graduate school of Journalism. Tow Center for Digital Journalism, C.U.G.S., GitBook: Guide to Automated Journalism (2016). https://books.google.com.ua/books?id=0iPbjwEACAAJ
6. Graham, Y.: Re-evaluating automatic summarization with BLEU and 192 shades of ROUGE. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 128–137. Association for Computational Linguistics, Lisbon, Portugal, September 2015. https://doi.org/10.18653/v1/D15-1013, https://www.aclweb.org/anthology/D15-1013
7. Gusev, I.: Importance of copying mechanism for news headline generation (2019)
8. Hermann, K.M., et al.: Teaching machines to read and comprehend. CoRR abs/1506.03340 (2015). http://arxiv.org/abs/1506.03340
9. Hermann, K.M., et al.: Teaching machines to read and comprehend. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 1693–1701. Curran Associates, Inc. (2015). http://papers.nips.cc/paper/5945-teaching-machines-to-read-and-comprehend.pdf
10. Klein, G., Kim, Y., Deng, Y., Senellart, J., Rush, A.M.: OpenNMT: open-source toolkit for neural machine translation. CoRR abs/1701.02810 (2017). http://arxiv.org/abs/1701.02810
11. Kuratov, Y., Arkhipov, M.: Adaptation of deep bidirectional multilingual transformers for Russian language (2019)
12. Lin, C.Y.: ROUGE: a package for automatic evaluation of summaries. In: Text Summarization Branches Out, pp. 74–81. Association for Computational Linguistics, Barcelona, Spain, July 2004. https://www.aclweb.org/anthology/W04-1013
13. Liu, Y., Lapata, M.: Text summarization with pretrained encoders (2019)
14. Loper, E., Bird, S.: NLTK: the natural language toolkit. In: Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. ETMTNLP 2002, vol. 1, p. 63–70. Association for Computational Linguistics, USA (2002). https://doi.org/10.3115/1118108.1118117
15. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space (2013)
16. Nallapati, R., Zhou, B., dos santos, C.N., Gulcehre, C., Xiang, B.: Abstractive text summarization using sequence-to-sequence RNNs and beyond (2016)
17. Narayan, S., Cohen, S.B., Lapata, M.: Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 1797–1807. Association for Computational Linguistics, Brussels, Belgium, October–November 2018. https://doi.org/10.18653/v1/D18-1206, https://www.aclweb.org/anthology/D18-1206

18. Panchenko, A., Ustalov, D., Arefyev, N., Paperno, D., Konstantinova, N., Loukachevitch, N., Biemann, C.: Human and machine judgements for Russian semantic relatedness. In: Analysis of Images, Social Networks and Texts: 5th International Conference, AIST 2016, Yekaterinburg, Russia, 7–9 April 2016, Revised Selected Papers, pp. 221–235. Springer International Publishing, Yekaterinburg, Russia (2017). https://doi.org/10.1007/978-3-319-52920-2_21
19. Over, P.: An introduction to DUC-2001: intrinsic evaluation of generic news text summarization system (2001)
20. Paulus, R., Xiong, C., Socher, R.: A deep reinforced model for abstractive summarization (2017)
21. Sandhaus, E.: The New York times annotated corpus LDC2008t19 (2008)
22. Segalovich, I.: A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine, pp. 273–280 (2003)
23. Shavrina T., Shapovalova, O.: To the methodology of corpus construction for machine learning: « taiga» syntax tree corpus and parser. In: Proceedings of CORPORA2017, International Conference, Saint-Petersbourg (2017)
24. Sokolov, A.: Phrase-based attentional transformer for headline generation. In: Computational Linguistics and Intellectual Technologies (2019)
25. Stepanov, M.: News headline generation using stems, lemmas and grammemes. In: Computational Linguistics and Intellectual Technologies (2019)
26. Tan, J., Wan, X., Xiao, J.: From neural sentence summarization to headline generation: a coarse-to-fine approach. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence, pp. 4109–4115. IJCAI 2017. AAAI Press (2017). http://dl.acm.org/citation.cfm?id=3171837.3171860
27. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification (2015)