Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2019"

Moscow, May 29—June 1, 2019

AGRR-2019: AUTOMATIC GAPPING RESOLUTION FOR RUSSIAN

Smurov I. M., **Ponomareva M.**
ABBYY, Moscow, Russia

Shavrina T. O.
NRU HSE, Sberbank, Moscow, Russia

Droganova K.
Charles University, Faculty of Mathematics and Physics,
Prague, Czech Republic

The 2019 Shared Task on Automatic Gapping Resolution for Russian (AGRR-2019) aims to tackle non-trivial linguistic phenomenon, gapping, that occurs in coordinated structures and elides a repeated predicate, typically from the second clause.

In this paper we define the task and evaluation metrics, provide detailed information on data preparation, annotation schemes and methodology, analyze the results and describe different approaches of the participating solutions.

Key words: shared task, ellipsis, gapping, gapping resolution, Russian

1. Introduction

During the last two decades, just a few works have been dealing with ellipsis detection and resolution and almost exclusively for English. Most of these works address VP-ellipsis, which refers to the omission of a verb phrase whose meaning can be reconstructed from the context [Johnson 2001], for instance, in "Mary loves flowers. John does too" [Hardt 1997]; [Nielsen 2004]; [Lappin 2005]; [McShane and Babkin 2016]. [Anand and Hardt 2016] concentrate on sluicing, which refers to reduced interrogative clauses [Merchant 2001], for instance, in "Mary loves those flowers. I want to know why". [Schuster et al. 2018] and [Droganova and Zeman 2017] focus on gapping (i.e., an omission of a repeated predicate which can be understood from context [Ross 1970]).

To the best of our knowledge, there has been only one attempt to organize a shared task on ellipsis detection and resolution, specifically the shared task dedicated to VP-ellipsis detection and resolution for English, which was one of the SemEval-2010 tasks[1]. Unfortunately, the results of this shared task are not available.

Ellipsis exists in the majority of languages [Merchant 2001]. However, according to [Testelets 2011], a single rule that motivates elliptical constructions cannot be defined even within one language. In addition to the adversity of the construction itself, the phenomenon is naturally rare, thus research was conducted so far on rather small amount of data, not exceeding several hundreds of sentences; with the exception of [Anand and Hardt 2016], whose dataset consists of 4,100 sluicing examples from The New York Times subset of the Gigaword Corpus.

AGRR-2019 aims at detection and resolution of gapping constructions for Russian. For the purpose of the shared task we defined the task and evaluation metrics and developed a gapping dataset for Russian that consists of 7.5k sentences with gapping (as well as 15k relevant negative examples) and comprises data of various genres: news, fiction, social media and technical texts. We hope that the proposed methodology and dataset will encourage further development and regular comparison of systems for gapping detection and resolution.

2. Data

2.1. Linguistic Description

In this work we use the following terminology for gapping elements. We call the pronounced elements of the gapped clause **remnants**. Parallel elements found in full clause that are similar to remnants both semantically and syntactically are called **remnant correlates**. The missing material is called **the gap** [Coppock 2001].

Traditionally gapping is defined as an omission of a repeating predicate in non-initial composed and subordinate clauses where both remnants to the left and to the right remain expressed.

(1) Один имел силу солнца, другой — луны.
one had power sun other moon
'One had the power of the Sun, the other (had the power of) the Moon'

However Russian language allows a broader interpretation, thus it is important to mention the cases that were selected for the shared task and included into the gapping dataset for Russian.

The cases where the second remnant is missing and the second clause contains just one remnant are called stripping and can be considered a special case of gapping [Merchant 2016]. Canonical examples of stripping are limited to a small number of constructions (2)–(4). According to the [Hankamer and Sag 1976] who introduced the termin: "Stripping is a rule that deletes everything in a clause under identity with

[1] Task 4, descripion avaliable at http://semeval2.fbk.eu/semeval2.php?location=tasks#T14.

corresponding parts of a preceding clause except for one constituent (and sometimes a clause-initial adverb or negative)."

(2) The man stole the car after midnight, **but not** the diamonds. [Merchant 2016]

(3) Abby can speak passable Dutch, and Ben, **too**. [Wurmbrand 2013]

(4) Все мы любим Мамбу и Сережа **тоже**.
All we love Mamba and Serezha too
'All of us love Mamba, and Serezha ~~loves it~~ too'

Such examples were not included in the corpus. The set of constructions for Russian that implement stripping seems to be broader than for English. Therefore we encountered wide variety of examples that go beyond the canonical examples. Examples (5) and (6) illustrate the cases when arguments of the elided verb do not fully correspond to the arguments of the pronounced verb, thus some of the arguments of the elided verb (highlighted in bold) do not have correlates. We consider such examples gapping with one remnant and include them in the corpus.

(5) Добавляем муку, крахмал и разрыхлитель, а **в конце** — сметану.
add flour starch and baking.powder and in end sour.cream
'We add flour, starch and baking powder, and at the end ~~we add~~ sour cream.'

(6) Рост цен составил 11,9 процента (**за 2009 год** — 4,4 процента)
growth prices amounted.to 11.9 percent in 2009 year 4.4 percent
'The prices growth amounted to 11.9 percent (in 2009 ~~it amounted to~~ 4.4 percent)'

Elements remaining after predicate omission can be of different nature. Consider the following examples where remnants are predicates (7), preposition phrases (8), adverbs (9), adjectives (10) possibly with their dependents.

(7) Одно может вдохновлять, а другое вгонять в тоску.
one can inspire and other put in melancholy
'One thing can inspire and the other ~~can~~ put you in a melancholy mood.'

(8) Советую вам поменьше думать о проблемах, и побольше
Recommend you less think about problems and more
об их решении.
about their solution
'I recommend you to think less about problems, and ~~think~~ more about solving them.'

(9) Вначале они играли интересно, потом прескучно.
at.first they played interesting.ADV after extremely.boring.ADV
'At first they played interesting, then ~~they played~~ extremely boring.'

(10) Сердце ее было слишком чистым, чувства слишком искренними.
heart her was too pure feelings too sincere
'Her heart was too pure and her feelings ~~were~~ too sincere.'

While collecting the corpus, one of our main goals was to make it diverse. Along with grammatical diversity briefly described above, we intended to make the corpus heterogeneous both lexically and topically. We discuss how different genres contribute to the corpus in the next section.

2.2. Obtaining the Data

Reasonable amount of data is crucial to train a system utilizing machine learning techniques. At the same time, gapping is a relatively rare syntactic phenomenon: according to our data, no more than 5 sentences out of 10,000 contain gapping. Furthermore, annotation is a laborious process and existing corpora do not exceed several hundred examples. Thus, for the purpose of the shares task our priority was to collect as much data as possible. For this reason we opted to validate automatically obtained markup instead of annotating sentences from scratch.

Compreno parser [Anisimovich et al. 2012] was used to provide syntactic analysis for several millions of sentences. This parser includes a template-based module for gapping detection [Bogdanov et al. 2012] which allowed us to identify sentences with gapping elements. Such sentences were selected and automatically annotated using bracket markup (see subsection **Dataset Format**).

Over 22,500 sentences were shared among 11 assessors. Assessors were asked to evaluate the automatically obtained annotations, classifying each sentence into one of the following classes:

[0] no gapping, no markup is needed;
[1] correctly annotated;
[2] incorrectly annotated;
[3] difficult to analyse.

Each sentence was evaluated by two assessors. If both assessors considered a sentence class 1, it was added to the corpus as a positive example.

Since the markup was only evaluated without correcting it, we managed to collect a reasonably large corpus in a relatively short time.

To serve the training purpose, the corpus has to include negative examples. We considered two types of negative examples to select more relevant sentences. The first type comprises problematic negative sentences on which Compreno parser false-positively predicted gapping (labeled with 0 by both assessors). Introducing negative examples of this type supposedly would allow a system to improve upon the results of the source parser. The second type comprises sentences not shorter than 6 words that contain dash or comma, and a verb. We made the negative class twice as large as the positive one.

We intended to produce a corpus comprising a variety of genres. The main part of the corpus consist of fiction, technical texts and news. We deliberately added texts from social media and balanced their proportion in both positive and negative classes, so they form 25% of the corpus.

All obtained sentences were split in development set and training set in proportion 1:5. For the final submission, the participants were allowed to train their systems on training set and development set jointly.

The annotation of the test set was evaluated by the organizers: it contains ten times less examples than joined training and development sets with the same distribution of genres and the same ratio of positive to negative classes.

In addition, we released optional training materials, that comprise 115,563 examples of noisy data with the same proportion of positive and negative examples. The annotation was obtained automatically by Compreno without further manual validation.

Table 1: Number of examples by class; vk stands for social media texts

		0		1		sum
dev	vk	670	2,760	326	1,382	20,548
	other	2,090		1,056		
train	vk	2,860	10,864	1,366	5,542	
	other	8,004		4,176		
test	vk	343	1,365	185	680	2,045
	other	1,022		495		
sum		14,989		7,604		22,593

2.3. Dataset Format

We have two versions of annotation schemata. The first schema provides human-readable format useful for analysing and evaluating the annotation. Square brackets are utilized to mark all gapping elements (whole NP, VP, PP etc. for remnants and their correlates and the predicate controlling the gap). The gap is marked with **V**. The syntactic head of the predicate that corresponds to the ellided predicate is marked with **cV**.

- V—the gap
- cV—the head of the VP that controls the gap
- R1—the first remnant
- cR1—correlate of the first remnant
- R2—the second remnant
- cR2—correlate of the second remnant

The sentence (10) would have the following bracket annotation (11).

(11) [$_{\text{cR1}}$ Сердце ее] [$_{\text{cV}}$ было] [$_{\text{cR2}}$ слишком чистым], [$_{\text{R2}}$ чувства] [$_{\text{V}}$]
heart her was too pure feelings
[$_{\text{R2}}$ слишком искренними].
too sincere
‘Her heart was too pure and her feelings (were) too sincere.’

While the bracket format is convenient for human analysis, it is less suitable as input for automatic systems. Thus we utilize the alternative format: information concerning every sentence is represented by 8 columns. The first column contains plain text, which serves as input for automatic systems. The second column contains 0 or 1 depending on the presence of gapping. The rest of the columns correspond to gapping elements (cV, cR1, cR2, V, R1, R2) and contain character offsets for annotation borders for each gapping element if it is present in the sentence. Consider an example (12).

(12) **text**
Сердце ее было слишком чистым, чувства слишком искренними.

class	**cV**	**cR1**	**cR2**	**V**	**R1**	**R2**
1	10:14	0:9	15:30	39:39	31:38	39:57

2.4. Assessment Analysis

In this section we provide analysis of the examples that were labeled as class [2] or [3] by the assessors. Tables 2 and 3 show the confusion matrices of assessors' marks.

Table 2. Assessment analysis for the subcorpus of technical and fiction texts

	0	1	2	3
0	1,533	138	129	136
1	240	5,301	1,021	237
2	213	451	1,600	281
3	307	177	117	108

Out of 11,989 sentences 44% were considered correctly annotated and 13% were unanimously considered to have no gapping.

Table 3. Assessment analysis for the social media subcorpus

	0	1	2	3
0	1,817	232	174	118
1	154	1,900	142	46
2	75	130	360	21
3	139	53	36	25

Out of 5,422 sentences 35% were considered correctly annotated and 34% were unanimously considered to have no gapping.

The annotators classified slightly more than half of the automatically annotated examples as correctly annotated or having no gapping at all. Out of the rest of examples the most interesting are the examples unanimously attributed to class 2—incorrect annotation of sentence with gapping—and class 3—problematic sentences that are difficult to analyse.

Let us illustrate cases frequently encountered in these two classes. All sentences are given with automatic annotation, which has errors that show the bias of the source system and the corpus.

The following cases are common for class 2:

- Gapping with more than two remnants

(13) В Виннице больше оставаться было нельзя, [$_{\text{cR1}}$ семья] [$_{\text{cR2}}$ самолётом]
in Vinnitsa longer stay was impossible family plane
[$_{\text{cV}}$ отправилась] **в Россию**, а [$_{\text{R1}}$ я] [$_{\text{V}}$] [$_{\text{R2}}$ поездом **на восток**].
travaled to Russia and I train to east
'It was impossible to stay any longer in Vinnitsa, and the family traveled by plane to Russia, while I took a train to the east.'

Among other cases listed below, this is the only case that always gets erroneous annotation due to the limitations of the rule-based algorithm for gapping detection in Compreno.

- Lack of markup in some of multiple clauses that contain a gap

(14) [$_{cR1}$ В Петербурге] делами [$_{cV}$ ведал] старший сын [$_{cR2}$ Фёдор],
in St.Petersburg business involved eldest son Fedor
[$_{R1}$ в Казани] —[$_{V}$] [$_{R2}$ Иван], **в Ростове** **и Рыбинске** — **Дмитрий,**
in Kazan Ivan in Rostov and Rybinsk Dmitry
в **Самаре** — **Михаил**.
in Samara Mikhail.
'In St. Petersburg the eldest son Fedor was involved in business, in Kazan—Ivan, in Rostov and Rybinsk—Dmitry, in Samara—Mikhail.'

- Particular type of gapping when the correlate clause semantically generalizes over instances described in following clauses

(15) [$_{cR1}$ Два Ангела] [$_{cV}$ уселись] на плечах: один— [$_{cR2}$ на левом],
two angels sat on shoulders one on left
а [$_{R1}$ второй] —[$_{V}$] [$_{R2}$ на правом].
and second on right
'Two Angels sat on their shoulders: the first set on the left and the second on the right.'

This type of gapping is not limited to semantic relations between the clauses. The main clause may lack the correlates of some remnants, e.g. *в правую руку, в левую руку* in (16).

(16) [$_{cV}$ Возьмите] лист [$_{cR1}$ бумаги] и два карандаша разного цвета:
take piece paper and two pencils different colour
один [$_{cR2}$ в правую руку], [$_{R1}$ другой] —[$_{V}$] [$_{R2}$ в левую].
one in right hand another in left
'Take a piece of paper and two different coloured pencils: one in the right hand, the other in the left.'

- The correlates may remain unmarked in case of coordinated predicates in the full clause.

(17) **Ты** продолжала молчать и оценивающе [$_{cV}$ смотрела]
you kept be.silent and appraisingly looked
[$_{cR2}$ на меня], а [$_{R1}$ я] [$_{V}$] [$_{R2}$ на тебя].
at me and I at you
'You kept silent and were looking at me appraisingly, while I ~~was looking~~ at you.'

- Incorrectly predicted boundaries of gapping elements. In (18) the unknown word *Суне* may be the reason for the erroneous prediction.

(18) Тётя Яна [$_{cV}$ купила] [$_{cR1}$ **своей**] [$_{cR2}$ **Суне** сказки], а [$_{R1}$ себе] [$_{V}$] [$_{R2}$ прописи].
Aunt Yana bought her Suna fairy.tales and for.herself copybook
'Aunt Yana bought a book of fairy tales for Suna and a copybook for herself.'

- When gapping appears deeper in the syntactic tree, the main clause of the whole sentence may be erroneously predicted as correlate.

(19) Поэтому-то [$_{cR2}$ Евангелие] и [$_{cV}$ советует] нам благословлять,
That.is.why Gospel PART advises us to.bless
а не проклинать, так как **благословение** **приносит** **благо**,
and not to.curse because blessing brings good
а [$_{R1}$ проклятье] — [$_{V}$] [$_{R2}$ беду и несчастье].
and curse misfortune and grief
'That is why the Gospel advises us not to curse but to bless, because blessing brings good, and curse ~~brings~~ misfortune and grief.'

- The pair of a remnant and its correlate are missing in annotation

(20) [$_{cR1}$ Кто-то из нас] [$_{cV}$ выживает] **благодаря**, а [$_{R1}$ кто-то] **вопреки**.
Somebody of us survive due.to and somebody despite.of
'Some of us survive due to something, and some, despite of something.'

- The analysis that is syntactically possible, but semantically doubtful and causes incorrect sentence interpretation. In (21) the correlate of R1 is erroneously detected due to morphological homonymy of *слова мысли* (it is interpreted as NomPl), thus the correlate of the predicate is predicted incorrectly as well.

(21) [$_{cR1}$ Евангелие] [$_{cV}$ призывает] человека **привести** свои **дела**
Gospel encourages person to.bring their deeds
[$_{cR2}$ в соответствие со словами], [$_{R1}$ слова] [$_{V}$] [$_{R2}$ в соответствие
into.line with words words into.line
с мыслями], а [$_{R1}$ мысли] — [$_{V}$] [$_{R2}$ в соответствие со Словом Божиим].
with thoughts and thoughts into.line with word of.God
'The Gospel encourages a person to bring their deeds into line with words, ~~to bring~~ their words into line with thoughts, and ~~to bring~~ their thoughts into line with the Word of God.'

- The parser may miss some remnants in coordinated clauses that contain a gap. In this case some remnants may be erroneously merged together and form one remnant instead of two that would correspond to different correlates.

(22) [$_{cV}$ Нарезать] [$_{cR1}$ лук и шампиньоны **полукольцами, куриное филе**]
to.slice onion and champignons half.moons chicken fillet
[$_{cR2}$ кубиками], а [$_{R1}$ картофель] [$_{V}$] [$_{R2}$ полосками].
cubes and potatoes sticks
'Slice the onion and champignons into half moons, ~~dice~~ the chicken fillet into cubes, and ~~cut~~ the potatoes into sticks.'

- Coordinated correlates or remnants are not predicted as an entire gapping element

(23) [$_{cR1}$ Раньше] я [$_{cV}$ хотела] [$_{cR2}$ любви] **и замуж**, а [$_{R1}$ сейчас]
previously I wanted love.NOUN and married.ADV and now
[$_{R2}$ кожанку и джип].
leather.jacket and jeep
'Previously, I wanted love and to get married, and now ~~I want~~ leather jacket and jeep.'

Both assessors considered approximately 1% of all the automatically annotated examples problematic. The cases where the markup was inapplicable rather then wrong or the assessors could not mark an example as lacking any kind of gap are the following:

- Canonical stripping (with *тоже, нет* etc.)

(24) Пронумеруйте тakты, а то [$_{\text{cR1}}$ глаза] [$_{\text{cV}}$ могут] сместиться,
number.IMP bars because eyes may shift
а [$_{\text{R1}}$ цифры] — **нет**!
and numbers not
'Number the bar lines, because the eyes may shift, but not the numbers'

[$_{\text{cR1}}$ Мертвых] [$_{\text{cR2}}$ мы] охотно [$_{\text{cV}}$ принимаем] сюда, но [$_{\text{R1}}$ живых] — [$_{\text{V}}$]
dead.NOUN we gladly accept here but living.NOUN
[$_{\text{R2}}$ **дудки**]!
no.EXCLAM
'We gladly accept the dead here, but the living—not on your life'

- Stripping with less typical markers

Они [$_{\text{cV}}$ оказывают] психологическую поддержку [$_{\text{cR2}}$ жертвам
they provide psychological support victims
землятресений], **в особенности** [$_{\text{V}}$] [$_{\text{R2}}$ детям].
earthquakes especially children
'They provide psychological support to earthquake victims, especially to children'

- Conjunction rather then gapping

(25) [$_{\text{cR2}}$ Рисовать рисунок] [$_{\text{cV}}$ надо] на кальке, а затем [$_{\text{V}}$]
to.draw picture should on tracing.paper and then
[$_{\text{R2}}$ вырезать как показано на картинке].
cut.out as is.shown on figure
'One should draw the picture on tracing paper and then cut it out as the figure shows.'

(26) Меня [$_{\text{cV}}$ попросили] [$_{\text{cR1}}$ привезти] вас [$_{\text{cR2}}$ сюда], а [$_{\text{R1}}$ самому поехать]
me asked bring you here and myself go
[$_{\text{V}}$] [$_{\text{R2}}$ куда-нибудь еще].
somewhere else
'I was asked to bring you here and to go somewhere else myself.'

3. Shared Task Set-Up

Training data were released on January 26th 2019, automatic noisy data were released a week after. Participants had approximately a month to create solutions (system submissions were due February 23rd) and the results were announced on March 5, 2019.

Further details on the task schedule, evaluation, and results are available on the task web site at: https://github.com/dialogue-evaluation/AGRR-2019.

3.1. Shared Task

We offered participants two tracks concerning different technological limitations:

1. Closed track—an open-source track, convenient for research groups and student teams. Participants of the track were allowed to train their models only on open-access data (open source dictionaries, word embeddings, open universal embedders, open parsing systems, etc.). To verify the results, participants placed their code and models on github making it publicly available both for organizers and other teams.
2. Open track—no restriction on data and systems used; recommended for participants from industry who present their products. Track participants were allowed to bring any data for learning beyond the provided data and use their own commercial programs. Github sharing was not required.

All the systems participating in the shared task have chosen closed (open-source) track. All the models are publicly available on participants' github (links can be found at AGRR github page).

The participants were offered 3 different gapping tasks:

1. Binary presence-absence classification—for every sentence, participating systems must decide if there is a gapping construction in it.
2. Gap resolution—for every sentence with gapping, participating systems must predict the position of the elided predicate and the pronounced predicate in the antecedent clause.
3. Full annotation—for every sentence with gapping, participating systems must predict the linear position of the elided predicate and positions of its remnants in the clause with the gap, as well as the positions of remnant correlates and pronounced predicate in the antecedent clause.

3.2. Metrics

For the binary classification task we have decided to use standard metrics: precision, recall and f-measure (the participants' submissions were ranked according to the latter one). For the tasks 2 and 3, we have decided to avoid using standard metrics that require gold-standard tokenization. Our main motivation was to allow participants to use any available syntactic parser (since tokenization is often a part of a syntactic parsing pipeline, choosing any particular tokenization could have potentially made some parsers less suitable for the shared task than others). Given this reasoning, for gap resolution and full annotation tasks we have chosen symbolwise f-measure as the main metric. More specifically:

- true-negative samples for binary classification task do not affect total f-measure;
- for true-positive samples, symbolwise f-measure is obtained for each relevant gapping element separately, thus generating 6 numbers for full annotation task and 2 numbers for gap resolution task (if the evaluated sentence is either false-positive or false-negative, all the generated numbers are equal to 0);
- the obtained f-measures are macro-averaged on the whole corpus.

For instance, if the gold standard offset for particular gapping element is 10:15 and the prediction is 8:14, we have 4 true positive chars, 1 false negative char and 2 false positive chars, and the resulting f-measure equals 0.727.

It should be noted that the evaluation results on task 1 are always greater or equal to the results on tasks 2 and 3 (and while f-measure on task 2 may theoretically be lower than f-measure on task 3, the former is normally expected to be higher than the latter). This feature correlates with the hierarchy of the tasks: each subsequent task requires solving the previous ones (i. e. tasks 2 and 3 have nonzero annotations only on the sentences with gapping and task 3 provides richer annotation than task 2).

3.3. Results

Research groups from various Russian universities (MIPT, MSU, HSE, IITP, NSU), participants from two IT companies and several independent researchers have taken part in the competition, making 9 teams in total.

Binary classification and gap resolution tasks were equally popular among the participants (all teams have submitted solutions for the tasks); all teams but one have also participated in full annotation task. Final results are shown in Table 4 (sorted by gap resolution score). The implemented solutions are described in detail in the next section.

Table 4. The official results of the AGRR-2019 shared task

team	binary			gap resolution	full
	precision	recall	f-measure	f-measure	f-measure
fit_predict	0.969	0.95	**0.959**	**0.905**	**0.892**
EXO	0.899	0.964	**0.931**	**0.815**	**0.786**
Koziev Ilya	0.774	0.903	0.834	**0.677**	**0.647**
Derise	0.801	0.906	**0.850**	0.665	0.622
Meanotek	0.891	0.781	0.832	0.635	0.514
МГУ-DeepPavlov	0.934	0.644	0.762	0.601	0.587
vlad	0.778	0.915	0.841	0.574	
MorphoBabushka	0.763	0.619	0.683	0.466	0.440
nsu-ai	0.485	0.123	0.196	0.037	0.036

Some participants have submitted their solutions after the deadline (but before the release of the test data). These solutions were not scored alongside the official results of the AGRR-2019 shared task. These results are given in a separate table (see Table 5).

Table 5. Results of after-deadline submissions

team	binary			gap resolution	full
	precision	recall	f-measure	f-measure	f-measure
МГУ-DeepPavlov	0.973	0.646	0.776	0.617	0.599
МГУ-DeepPavlov	0.898	0.934	0.916		
МГУ-DeepPavlov[2]	0.97	0.712	0.821	0.658	0.653
EXO	0.946	0.946	**0.946**	**0.859**	**0.836**
Meanotek	0.815	0.939	0.872	0.727	0.688

3.4. Methods

All participants but one (МГУ-DeepPavlov) have reduced gap resolution and full annotation tasks to sequence labeling task. The most fruitful approaches were to enhance standard BLSTM-CRF architecture [Lample et al. 2016]; [Ma and Hovy 2016], to pretrain LSTM-based language model or to use transformer-based solutions [Vaswani et al. 2017]; [Devlin et al. 2018].

The methods used are summarized in Table 6.

Table 6. Methods of the AGRR participants

team	architecture	token features	sequence labeler	additional features
fit_predict	Trasformer (BERT)	BERT (pretrained)	Custom FSA-based postprocesser	Joint model resolving both full annotation and binary classification; Noisy data (not validated by assessors) was used.
EXO	BLSTM + MultiHead self-attention	BERT (pretrained)	NCRF++ (n-best CRF implemented in [Yang and Zhang 2018])	Joint model resolving both full annotation and binary classification.
Koziev	BLSTM	Word2vec + CharRNN(CNN)	CRF	Separate models for binary classification and full annotation tasks.
Derise	BiGRU, Transformer	fastText	None	Separate models for binary classification (BiGRU) and full annotation (Transformer) tasks.
Meanotek	2 layer LSTM	Character-level LM (LSTM)	None	Full annotation task model was trained; Binary classification is resolved with heuristics on full annotation.

[2] These results were further improved two weeks after the end of AGRR-2019, when all the gold answers and the systems of the other participants were available. We do not consider these results relevant for the shared task and thus do not to include them into this paper.

team	architecture	token features	sequence labeler	additional features
МГУ-Deep-Pavlov	Rule-based; BLSTM model (submitted after deadline)	ELMo, UDPipe, Morphological features	Not sequence labeling approach	Rule-based system (scored system) BLSTM model (submitted after deadline) uses dot-product similarity to determine if a pair of tokens are a particular pair of gapping elements (cV and V, V and R1 etc)
vlad	ULMFit + linear decoder	ULMFit (pretrained)	None	Separate models for binary classification (MLP) and full annotation (linear decoder) tasks.
Morpho-Babushka	BERT	BERT (pretrained) + Pymorphy2	None	Separate models for binary classification and full annotation tasks.
nsu-ai	BERT	BERT (pretrained)	None	Joint model, separate outputs for class (one per sentence) and each gapping element label (one per token).

Most participating systems did not use any token-level features other than word embeddings, character-level embeddings, or language model embeddings [Peters et al. 2018]; [Devlin et al. 2018]; [Howard and Ruder 2018]. Of particular note is that neither of the top-scoring systems use morphological or syntactic features. While it may be theorized that using such features could yield some improvements, we suppose that language model embeddings (especially when coupled with self-attention like in the top two systems) contain most syntactic information relevant for ellipsis resolution.

4. Conclusion

In this paper we have introduced Automatic Gapping Resolution for Russian (AGRR-2019), the first shared task centered on gapping. We have outlined the design of the dataset used for the shared task and provided a brief assessment analysis. We have defined three tasks on gapping detection and resolution as well as evaluation metrics. Finally, we have presented the official results of the shared task.

The two most important features of our dataset are its diversity and size. Russian language allows rather broad interpretation of gapping (see section Linguistic Description for details). Furthermore, we were able to increase the diversity of our corpus not only by varying its genre composition, but also by including a substantial social media component (see details in section Obtaining the Data). The size of our corpus (7.5k sentences with gapping and 15k relevant negative sentences) is sufficient for successful gapping resolution with ML-methods (as shown in Results). To the best of our knowledge no other publicly available dataset contains a comparable amount of gapping examples.

The task attracted considerable attention from a number of researchers, but only nine teams have submitted their solutions. Nevertheless the participants have demonstrated that gapping can be successfully resolved using sequence-labeling techniques (the best solution has achieved 0.96 in gapping classification task, 0.91 in gap resolution task and 0.89 in full annotation task). A surprising observation is that rich morphological and syntactic features are not necessary to achieve satisfactory results on gapping resolution.

We hope that AGRR-2019 has provided useful insights both for researchers interested in improving parsing quality and those who study theoretical aspects of gapping. We believe that it is a small step towards fully resolved ellipsis.

5. Acknowledgments

Authors are thankful to Alexey Bogdanov for help with linguistic analysis and useful advices.

The work was partially supported by the GA UK grant 794417.

References

1. *Anand P. and Hardt D.* (2016). Antecedent selection for sluicing: Structure and content. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 1234–1243.
2. *Anisimovich K. V., Druzhkin K. Ju., Minlos F. R., Petrova M. A., Selegey V. P. and Zuev K. A.* (2012). Syntactic and semantic parser based on abbyy compreno linguistic technologies. In Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialog” [Komp’iuternaia Lingvistika i Intellektual’nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii “Dialog”], Bekasovo, Vol. 2, pp. 90–103.
3. *Bogdanov A.* (2012). Description of gapping in a system of automatic translation. In Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialog” [Komp’iuternaia Lingvistika i Intellektual’nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii “Dialog”], Bekasovo, Vol. 2, pp. 61–70.
4. *Coppock E.* (2001). Gapping: In defense of deletion. In Proceedings of the Chicago Linguistics Society, Vol. 13, pp. 133–148.
5. *Devlin J., Chang M. W., Lee K. and Toutanova K.* (2018). Bert: Pretraining of deep bidirectional transformers for language understanding. In arXiv preprint arXiv:1810.0480.
6. *Droganova K. and Zeman D.* (2017). Elliptic Constructions: Spotting Patterns in UD Treebanks. In Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017), 48–57.
7. *Hankamer J. and Sag I.* (1976). Deep and surface anaphora. In Linguistic Inquiry, 7:391–426.
8. *Hardt D.* (1997). An empirical approach to VP ellipsis, Computational Linguistics, MIT Press, Vol. 23(4), pp. 525–541.
9. *Howard J. and Ruder S.* (2018). Universal language model fine-tuning for text classification. In Association for Computational Linguistics

10. *Johnson K.* (2001). What VP ellipsis can do, and what it can't, but not why, Citeseer
11. *Lample G., Ballesteros M., Subramanian S., Kawakami K. and Dyer C.* (2016). Neural Architectures for Named Entity Recognition. In NAACL-HLT.
12. *Lappin S.* (2005). A sequenced model of anaphora and ellipsis resolution, Anaphora Processing: Linguistic, Cognitive, and Computational Modelling. Amsterdam: John Benjamins, pp. 3–16.
13. *Ma X. and Hovy E.* (2016). End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics.
14. *McShane M. and Babkin P.* (2016). Detection and resolution of verb phrase ellipsis, LiLT (Linguistic Issues in Language Technology), Vol. 13.
15. *Merchant J.* (2001). The syntax of silence: Sluicing, islands, and the theory of ellipsis, Oxford University Press on Demand.
16. *Merchant J.* (2016). Ellipsis: A survey of analytical approaches. University of Chicago, Chicago, IL.
17. *Nielsen L. A.* (2004) Verb phrase ellipsis detection using automatically parsed text. In Proceedings of the 20th international conference on Computational Linguistics, Association for Computational Linguistics, p. 1093.
18. *Peters M. E., Neumann M., Iyyer M., Gardner M., Clark C., Lee K. and Zettlemoyer L.* (2018). Deep contextualized word representations. In Proceedings of NAACL.
19. *Ross J. R.* (1970). Gapping and the order of constituents. In Manfred Bierwisch and Karl Erich Heidolph, editors, Progress in linguistics: A collection of paper, De Gruyter, 43:249–259.
20. *Schuster S., Nivre J. and Manning C.* (2018). Sentences with Gapping: Parsing and Reconstructing Elided Predicates, arXiv preprint arXiv:1804.06922.
21. *Testelets Ya. G.* (2011). Ellipsis in Russian: Theory versus Description. Typology of Morphosyntactic Parameters [Ellipsis v russkom yazyke: teoreticheskiĭ i opisatel'nyĭ podkhody], Typology of Morphosyntactic parameters, MSUH, pp. 1–6.
22. *Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł. and Polosukhin I.* (2017) Attention is all you need. In Advances in Neural Information Processing Systems, pp. 5998–6008.
23. *Wurmbrand S.* (2013). Stripping and topless complements. Ms., University of Connecticut.
24. *Yang J. and Zhang Y.* (2018). NCRF++: An Open-source Neural Sequence Labeling Toolkit. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics.