Unsupervised PCFG Inference from Russian Corpus of Phone Conversations

Liubov Kovriguina
ITMO University
Saint-Petersburg, Russia
lyukovriguina@corp.ifmo.ru

Alexander Shipilo
ITMO University, Saint-Petersburg State University
Saint-Petersburg, Russia
alexandershipilo@gmail.com

Ekaterina Sinelshchikova
ITMO University
ITMO University, Saint-Petersburg, Russia
sinel.katya@gmail.com

***Abstract*—Spontaneous speech full parsing still remains an unsolved task for the Russian language although a great amount of theoretical work has been done in the field of spontaneous speech syntax. The paper presents results on probabilistic context-free grammar induction from the unlabelled corpus of Russian spontaneous speech using the algorithm proposed by James Scicluna and Colin de la Higuera in 2014. The corpus contains 40 hours of speech (250 000 tokens). The exact task of the experiment was to learn syntactic structure of elementary discourse units that occur in spontaneous speech, make a benchmark for further development of spontaneous speech parsing algorithms and get statistics about elementary discourse units length and structure in spontaneous speech.**

I. Introduction

Voice control systems, dialogue systems and natural language interfaces consume spontaneous human speech. This results in the doubly noisy input: the noise comes from ASR systems and it also comes from feeble speech "understanding" algorithms that mostly use deadly simple approaches (n-grams or pattern search, keyword extraction, etc.) declaring that natural language understanding is performed. The last problem arises from the fact that spontaneous speech is very hard to parse and analyze, because it contains a large number of speech disfluencies, breaks, interruptions, unfinished and ellyptical phrases, grammatical mistakes, etc. This is the reason why NL parsers, trained on "written" texts, fail when given a spontaneous input. Spontaneous speech full parsing still remains an unsolved task for the Russian language although a great amount of theoretical work has been done in the field of spontaneous speech syntax. Spontaneous syntax characteristics and principles of speech generation demand a set of parsing rules which differs considerably from the set of rules used in the parsers processing "written" text.

To bridge this gap, we currently work on development of the parser for Russian spontaneous speech combining methods of grammar inference with semantic resources (knowledge bases and ontology of lingustic metadata that contains derivational, taxonomic and semantic relations between words of the Russian language). Ontology of linguistic metadata is now being developed using existing Tuzov semantic dictionary (in English see [1].

The whole project demands considerable efforts on spoken corpus design, evaluating and improving methods of speech repairs identification and removal, applying and evaluating existing supervised and unsupervised algorithms of learning a pool of dependency and constituency grammars, algorithms of semantic parsing, etc. For the Russian language no data and results have been published about quality of parsing procedures evaluated on spontaneous speech, therefore, we started the research with parsing spoken language using unsupervised algorithms in order to make a benchmark for further comparison of parsing procedures and algorithms applied to the spoken language. It seemed reasonable to start with the unsupervised algorithm running on the morphologically annotated corpus with minimal preprocessing to estimate the correctness of the parsing rules obtained on the unlabelled data. We used the algorithm for unsupervised probabilistic context-free grammar induction proposed by [2]. It is frequently mentioned in the literature that dependency grammars model syntactic structures of the languages with free or flexible word order (like Russian) better than phrase-structure grammars, but in the considered case the choice between dependency and constituent grammars is quite irrelevant, because, as it is mentioned above, the long-term goal is to consequently combine and evaluate different approaches to Russian spontaneous speech parsing running them on spoken corpora that undergo from none to thorough preprocessing to find the optimal way to generate valid syntactic/semantic structures for disfluent and noisy speech.

A. Experiment setup and paper structure

Main part starts from brief explanation of grammar inference principles and its general approaches (section 2). The experiment on testing COMINO for grammar inference for Russian spontaneous speech splits into three evident steps: 1) sample preparation, 2) adding modifications to the algorithm and its implementation, 3) evaluation. Sample preparation is discussed in section 3, where corpus annotating and algorithm input preparation (splitting the corpus into elementary discourse units) are described. Section 4 concerns the original COMINO algorithm and its implementation in the current experiments, evaluation and results are given in section 5.

II. Grammatical inference approaches

Grammatical Inference (grammar learning) is a subfield of theoretical computer science. It "provides principled methods for developing computationally sound algorithms that learn structure from strings of symbols. The relationship to computational linguistics is natural because many research problems in computational linguistics are learning problems on words, phrases, and sentences" [3]. Grammatical inference methods are applicable to any discrete sequences of symbols or strings, therefore, they can be used in the broadest range of

research tasks, from pattern recognition in computer vision to process mining. Heinz, de la Higuera and van Zaanen brilliantly remark, that "there are many other ways to look at the learning of (natural language) grammars". For instance, one area of research aims to develop cognitive models of language learning. In this area, properties of theoretical models are compared against the performance of human learners. It still is not clear exactly what a cognitively realistic model of language learning should include [3]. In respect to the development of natural language parsers grammar inference models are crucial to build an adequate set of parser rules, which can rely on any available annotation or perform on unlabelled data. The grammar inference algorithm should be given access to an annoated corpus or unlabelled data. With this information the algorithm tries to build a formal representation capable to analyze the structure of sentences (analysis) in natural language and generate sentences in natural language (synthesis). Two typical scenarios of grammar learning is learning from text where only strings from the language are given to the learner (this approach is known as *batch learning*), and learning from an informant where the strings from which one is to learn are labeled with 1 or 0 depending on the fact that they belong or not to the language. These two approaches are directly correlating with unsupervised and supervised learning. Unsupervised approaches are usually less accurate, but time-saving as they need no annotation which requires informant's efforts, which are sometimes burdening (syntactic tagging, as well as parsing, is one of the most laborious and complex NLP tasks especially in the part of relation labeling). Among the unsolved problems of grammar inference remains the need of algorithms that can deal with noisy data: the usual benchmarks that appeared since the late 90s were concerned with learning large automata from positive (sentences that are valid and appear in the language) and negative (grammatically ill-formed sentences) data, but in all cases this data has to be noise free. Grammar inference algorithms share most algorithms and techniques with data mining and applied statistics: machine learning, pattern recognition, tabu search, sequence modeling, etc.

III. Spontaneous speech corpus tagging

For the experiment a corpus of spontaneous phone conversations gently provided for the research by the Center of Speech Technologies (http://speechpro.ru/) was used. The corpus contains signal, phoneme tagging, pause borders and transcripts of phone interviews (dialogues + monologues). Speech disfluencies are not numerous in this corpus because of the high professional and social status of the dictors. This corpus is private and can not be freely accessed. Currently we are building an open corpus of spontaneous Russian speech with multilevel annotations (speech repairs, EDU segmentation, syntactic tagging, etc.) to ensure comparability of results in future. Transcripts to 40 hours of speech from the above mentioned corpus were manually segmented into elementary discourse units and sentences. To fasten the process of splitting into elementary discourse units each sound file with corresponding transcription was segmented into smaller ones by pauses. A more detailed procedure of splitting the speech into elementary discourse units is given below.

Total number of elementary discourse units used in the experiment is 84000. Then each EDU was pos-tagged using Mystem tagger (https://tech.yandex.ru/mystem/) that allows to resolve homonymy. Speech repairs were left in the corpus except those which could be qualified as hesitation fillers, like

"ну", "вот", "как бы", "как-то так", "какой-то такой", "там э", "ээ", "а". The last were removed before the pos-tagging procedure. Even this simple preprocessing improved the results of grammar inference, because these words influenced contexts distribution.

A. Splitting the corpus into elementary discourse units

There is no unity between researchers on the definition and principles of identifying elementary discourse units [4]: "Researchers in the field have proposed a number of competing hypotheses about what constitutes an elementary discourse unit. While some take the elementary units to be clauses (Grimes, 1975; Givon, 1983; Longacre, 1983), others take them to be prosodic units (Hirschberg and Litman, 1993), turns of talk (Sacks, 1974), sentences (Polanyi, 1988), intentionally defined discourse segments (Grosz and Sidner, 1986), or the "contextually indexed representation of information conveyed by a semiotic gesture, asserting a single state of affairs or partial state of affairs in a discourse world," (Polanyi, 1996, p.5)". Principles of segmenting disctor's utterances into EDU were adopted as in Kibrik and Podlesskaya [5]. These authors elaborated the following cues to identify elementary discourse units: pausing, tempo, loudness, intonation, and accent placement. In the paper [6] Kibrik stresses that "identified EDUs display a remarkable correlation with independently established semantic and syntactic units, that is clauses" and provides data on the percentage of clausal EDU in English (60%), Russian (68%), Japanese (68%) and some other languages. In Table I we provide a distribution of elementary discourse units by the number of verbs in the EDU (evidence from the above mentioned corpus of spoken conversations). In the processed corpus percent of EDU containing at least one finite verb is smaller - 44%, probably due to the large number of answers like "yes" and "no" in the phone conversations.In Table II

TABLE I. EDU distribution by the number of verbs in the EDU

N verbs	frequency
0	47020
1	30696
2	5376
3	762
4	126
5	10
6	10

distribution of EDU by its length is shown (evidence from the same corpus) which can be used in the algorithms of automatic speech segmentation into EDU/clauses/sentences.

IV. Probabilistic context-free grammar definition in the COMINO Algorithm

Probabilistic context-free grammars (also called sometimes *Stochastic*) are CFG with probabilities added to rules. C.Manning and H.Schütze write that "PCFGs are the simplest and most natural probabilistic model for tree structures, the mathematics behind them is well understood, the algorithms for them are a natural development of the algorithms employed

TABLE II. EDU LENGTH DISTRIBUTION

N tokens	frequency
1	19389
2	16845
3	15779
4	12357
5	8447
6	4938
7	2946
8	1500
9	793
10	455
11	230
12	118
13	75
14	75
15	54

with HMMs, and PCFGs provide a sufficiently general computational device that they can simulate various other forms of probabilistic conditioning [7]. Scicluna and de la Higuera use the standard notion of CFG: "a context-free grammar (CFG) is a 4-tuple $\langle N, \Sigma, P, I\rangle$, where *N* is the set of non-terminals, Σ the set of terminals, *P* the set of production rules and *I* a set of starting non-terminals (i.e. multiple starting non-terminals are possible). The language generated from a particular non-terminal A is $L(A) = \{w \mid A \stackrel{*}{\Rightarrow} w\}$ and the language generated by a grammar G is $L(G) = \cup_{S\in I} L(S)$. A CFG is in *Chomsky Normal Form* (*CNF*) if every production rule is of the form $N \to NN$ or $N \to \Sigma$." [2, p. 1353].

A probabilistic context-free grammar (PCFG) is a CFG with a probability value assigned to every rule and every starting non-terminal [2, p. 1354].

Scicluna and Higuera apart from the notion of CFG and PCFG start from introducing the cornerstone notion of their approach to grammar inference - notion of a "congruence relation". Below is a broad citation with the formal definition of the congruence relation and its accessible explanation: " *A congruence relation* $\sim$ on Σ^* is any equivalence relation on Σ^* that respects the following condition: if $u \sim v$ and $x \sim y$ then $ux \sim vy$. The congruence classes of a congruence relation are simply its equivalence classes. The congruence class of $w \in \Sigma^*$ w.r.t. a congruence relation $\sim$ is denoted by $[w]_\sim$. The set of contexts of a substring w with respect to a language L, denoted *Con(w,L)*, is $\{(l, r) \in \Sigma^* \times \Sigma^* \mid lwr \in L\}$. Two strings *u* and *v* are syntactically congruent with respect to L, written uv, if $Con(u, L) = Con(v, L)$. This is a congruence relation on Σ^*. The *context distribution* of a substring *w* w.r.t. a stochastic language (L, ϕ), denoted $C_w^{(L,\phi)}$, is a distribution whose support is all the possible contexts over alphabet σ (i.e. $\Sigma^* \times \Sigma^*$) and is defined as follows:

$$C_w^{(L,\phi)}(l, r) = \frac{\phi(lwr)}{\Sigma_{l',r'\in\Sigma^*}(l'wr')}$$

Two strings *u* and *v* are stochastically congruent with respect to (L, ϕ), written $u \cong (L, \phi)v$, if $C_u^{(L,\phi)}$ is equal to $C_v^{(L,\phi)}$. This is a congruence relation on Σ^* " [2, p. 1354]".

Generally saying, two substrings *u* and *v* are congruent if their sets of left and right contexts are the same, and these substrings are stochastically congruent if the distributions of these contexts are also the same. Here arise two essential moments to discuss: 1) what is the context size and 2) which criterion to choose to affirm that contexts distributions are the same? Authors of the algorithm note, that "due to the problem of sparsity with contexts (in any reasonably sized corpus of natural language, very few phrases will have more than one occurrence of the same context), only local contexts were considered in their experiments. The local contexts of substring *w* are the pairs of first *k* symbols (or words for natural language) preceding and following *w*. The lower *k* is, the less sparsity is a problem, but the empirical context distribution is less accurate. For natural language corpora, *k* is normally set to 1 or 2" [2, p. 1355]. Context's length can be increased for artificial languages. To test the algorithm on Russian spontaneous speech corpus, we set the left and right context to 1. We used Mann-Whitney-Wilcoxon test to determine whether contexts' distributions of any two substrings come from one population without assuming them to follow the normal distribution.

COMINO (this is the name of the algorithm developed by Scicluna and de la Higuera) induces the grammar from a positive sample S. It includes the following basic steps [2, p. 1355]":

1) Inducing the stochastically congruent classes of all the substrings of S; At the beginning, each substring (or phrase for natural language) in the sample is assigned its own congruence class (line 2). Then, pairs of frequent congruence classes are merged together depending on the distance between their empirical context distribution, which is calculated on local contexts (that is, on contexts set to 1 or 2).
2) Selecting which of the induced classes are non-terminals and subsequently building a CFG. This is a crucial step when the algorithm knows nothing about the alphabet of the language or it is desirable to consider an n-gram or a collocation as a terminal. In our experiment, the set of terminals is known since the very beginning, therefore, the step of discriminating terminals from non-terminals was skipped. While building the context-free grammar probabilities are assigned in such a way that the smallest possible grammar is built.
3) Assigning probabilities to the induced CFG. This step was done using the Inside-Outside algorithm as in the cited paper.

In the next section details about the algorithm implementation, modifications and the explanation of choices is given.

A. COMINO Algorithm implementation and modifications

Each elementary discourse unit of the corpus has ungergone automatic POS-tagging with Mystem tagger. The input of the algorithm are sequences of part-of-speech tags corresponding to the the word forms in the elementary discourse unit. For example, the utterance " кто знает, что вот на меня так вот произвело впечатление такое большое" (English approximate translation: *"who knows, well, what impressed me at that time so much, a big impression though"*) will be rewritten as POS-tags sequence the following way: "*SPRO V CONJ PART PR SPRO ADVPRO PART V S APRO A*". Then, each input POS-tag sequence was split into all possible substrings and left and right contexts and their probabilities were generated for each generated substring. As a result,

we got an array of substrings assigned with their contexts and contexts' probabilities. At the next step we determined which substrings were stochastically congruent and merged them into one congruence class. Contexts' distributions were compared using Mann-Whitney-Wilcoxon test while Scicluna and de la Higuera used used L1-Distance and Pearson's chi-squared test. Our choice is explained by the fact that there is no evidence that contexts are distributed normally and it is preferable to use a non-parametric test. The next step - frequent congruence classes selection - differs from Scicluna and de la Higuera approach. COMINO's authors define a frequent congruence class as "one whose substring occurrences in the sample add up to more than a pre-defined threshold n. Infrequent congruence classes are ignored due to their unreliable empirical context distribution. However, as more merges are made, more substrings are added to infrequent classes, thus increasing their frequency and eventually they might be considered as frequent classes" [2, p. 1355]. As it can be concluded from this definition, the very task is to filter out frequent congruence classes from the array of congruence classes, but nothing is said about the procedure of threshold choosing. Therefore, we decided to rank elements of congruence classes according to their frequency in the sample and analyze their rank distribution. The distribution can be seen in Fig.1.

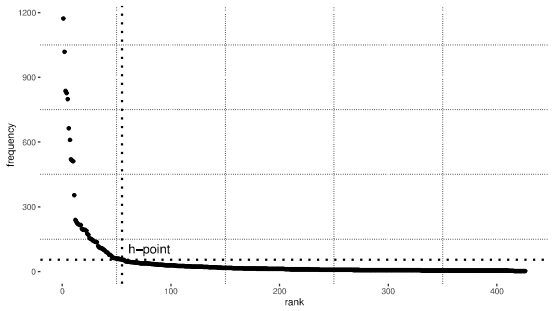

Fig. 1. Rank distribution of congruence classes

Apparently, this is a long-tail distribution, where typically about 75-80% of elements have minor absolute frequencies (5 or less). Distributions like this can be separated using the h-point criterion, introduced to quantitative linguistics by Altmann and Popescu in 2009 [8]. The *h-point* is defined as the point at which the straight line between two (usually) neighboring ranked frequencies intersects the $r = f(r)$ line [8, p. 24], see Fig.2:

$$h = \begin{cases} r, & if\ \exists\ r = f(r) \\ \frac{f(i)r_j - f(j)r_j}{r_j r_i + f(i) - f(j)}, & if\ \exists\ r = f(r) \end{cases}$$

In other words, the h-point is that point at which $r = f(r)$ (r – rank, $f(r)$ – absolute frequency of the token having rank r). If there is no such point, one takes, if possible, two $f(i)$ and $f(j)$ such that $f(i) > r_i$ and $f(j) < r_j$ (i, j are indexes for the neighboring frequencies and neighboring ranks). The h-point seems to be an important indicator in rank-frequency phenomena. In respect to lexical statistics, h-point shows the border between autosemantics and synsemantics. Regarding the problem of choosing frequent grammatical contexts, we speculated, that h-point will separate contexts involved in building sentence syntactic structure, from contexts involved in conveying stylistic and pragmatic characteristics of the utterance (anyhow, this is a way to set the threshold automatically, but the interpretation of the idea needs further experiments). Enlarged part of the congruence classes distribution can be seen in Fig.3.

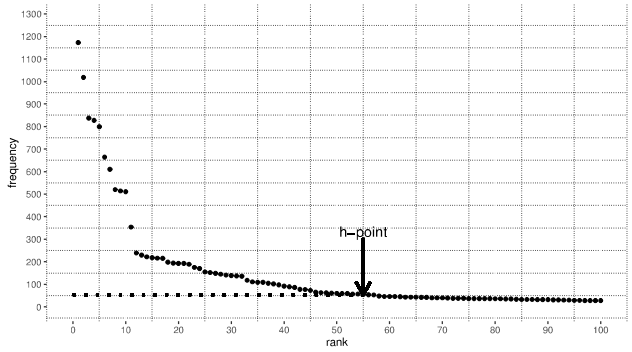

Fig. 2. Enlarged part of the congruence classes distribution

For the considered distribution h-point equals 55, it means that only 55 congruence classes were selected to build the grammar. However, these 55 classes solely cover 74.5% of occurrences of congruence classes elements in the sample. The process of building rules for PCFG starts with splitting congruence classes elements, which are non-terminals, using terminals and smaller elements of congruence classes. This process is described quite clearly in the paper of Scicluna and de la Higuera [2, p. 1356]: "for every congruence class $[w]$ and for every string w in $[w]$ ($|w| = n$), the following conditional statement is added to the formula:

$$v(w) \Rightarrow (v(w_{1,1}) \wedge v(w_{2,n})) \vee (v(w_{3,n})) \vee ... \vee (v(w_{1,n-1}) \wedge v(w_{n,n}))$$

where $v(x)$ is the variable corresponding to the congruence class $[x]$ and $w_{i,j}$ is the substring of w from the i^{th} to the j^{th} symbol of w. This statement is representing the fact that if a congruence class $[w]$ is chosen as a non-terminal then for each string in $w \in [w]$, there must be at least one CNF rule $A \rightarrow BC$ that generates w and thus there must be at least one division of *w* into $w_{1,k}w_{k+1,n}$ such that *B* corresponds to $[w1, k]$ and C corresponds to $[w_{k+1,n}]$. The grammar extracted from the solution of this formula is made up of all the possible CNF production rules built from the chosen non-terminals". For example, let's consider an element of the congruence class "ADJ PR N ADJ" (ADJ - adjective, PR - preposition, N - noun). It can be split as following:

ADJ PR N ADJ ⇒ ADJ + PR N ADJ

ADJ PR N ADJ ⇒ ADJ PR + N ADJ

ADJ PR N ADJ ⇒ ADJ PR N + ADJ

When the loop runs for all terminals and all non-terminals constituting nonterminals, frequency of all splits is count. The most frequent split is written to the list of grammar rules. The loop stops when all splits have equal frequency. Examples of the generated rules are given in the next section.

V. Results and discussion

A. Evaluation approaches to grammar inference algorithms

Grammar inference algorithms can be evaluated in different aspects (formal, logic, empirical, etc.). In this exact task we have to evaluate parsing accuracy.

According to van Zaanen and Geertzen [9], evaluation methods "can be divided into four groups: *looks-good-to-me* approaches analyze the output of GI systems manually. Rebuilding a-priori known grammars use, often small, "toy" grammars to generate sequences, which are used as input for the GI system. The output of the system is then compared against the original grammar. The *language membership* method measures the ability to classify sequences based on language membership. This measures language equivalence (weak equivalence). The performance in this method is expressed by two metrics: precision, which shows the effectiveness to decide whether a sequence is in the language or not and recall, which measures coverage. Finally, *comparison against a treebank* uses a treebank, a collection of sequences with their derivation, as a "gold standard". The plain sequences (generated by removing the structure from the treebank sequences) are used as input and the output of the GI system is compared against the original structure". In this task we will use comparison against a treebank and compare the annotation produced by the algorithm to the manual annotation of the gold standard.

B. Evaluation, results and discussion

The algorithm produced a grammar of 94 rules modeling the structure of elementary discourse units, most of them are adequate (rules for parsing verb phrases, noun phrases and prepositional phrases). Below are rules generated at the first 10 iterations of the algorithm:

1. $PR + N \rightarrow PP$
2. $SPRO + V \rightarrow S$
3. $A + N \rightarrow NP$
4. $V + ADV \rightarrow VP$
5. $APRO + N \rightarrow NP$
6. $N + V \rightarrow S$
7. $PART + V \rightarrow VP$
8. $PR + SPRO \rightarrow PP$
9. $CONJ + SPRO \rightarrow$ false phrase / elliptical phrase
10. $ADVPRO + V \rightarrow VP$

Among the wrong rules are rules joining adverbial pronouns and nouns, sequences of particles into one phrase. Some of the generated rules are either applicable to describe ellyptical sentences or can be considered wrong, i.e rule joining personal pronoun and adverb in a phrase (like "я быстро", a sentence with verb ellypsis in Russian).

To evaluate the algorithm performance a manually tagged sample of 1000 elementary discourse units from the corpus described above was used. Immediate constituents' borders and types were annotated for each elementary discourse unit in the sample. An example is shown below:
Russian text: Хотелось бы в Венецию / Мечта детства...
English translation: I would like to Venice / A childhood dream...
Manually annotated phrase structure: (S (VP V PART) (PP PR N)) / (NP N N)

Morphological analysis was performed using mystem tagger (https://tech.yandex.ru/mystem/), types of constituents were annotated using the Penn Treebank annotation scheme. In the example above N - noun, PR - preposition, PART - particle, V - verb ; VP - verb phrase, NP - noun phrase, PP - prepositional phrase, S - sentence. Unfinished phrases, speech repairs, sentences and phrases with ellypsis appeared frequently in the tagged dataset and additional annotation tags were ascribed to them: UP - unfinished phrase, SR - speech repair, EP - phrase with ellypsis. When comparing the COMINO algorithm tagging against the manual tagging wrong tags given by the algorithm to these three types of phrases were not considered as mistakes. Such easing of requirements was made due to the inability of the current algorithm to classify such phrases.

The evaluation was done using the Unlabelled brackets F1 (UF1) score as given in the paper of J.Scicluna and Colin de la Higuera [2, p. 1358]. We obtained the 69.2 UF1 score that is lower than 85.1 upper bound result given by COMINO algorithm. This upper bound COMINO result was obtained on Wall Street Journal corpus, not on the corpus or spontaneous dialogues or monologues Still, this resulresut hedefinilts scan not be directly compared.etely seems important for several reasons, therefore, t. Firstly, we obtained a baseline for unsupervised grammar learning for Russian spontaneous speech. Also, taking in consideration the fact, that speech repairs and other disfluencies influence the parsing quality badly, 69.2 parsing quality is an admittable result. Mistakes generated by COMINO algorithm are mainly arise from wrong parsing parentheses, phrases with ellypsis and coordinative compounds. Sequences of particles and hesitation fillers are also parsed improperly since they convey mainly pragmatics rather than syntactic relations.

It is a well-known fact that both immediate constituents and dependency structures have different limitations on syntactic structure modeling. Immediate constituents model, that is used in formal grammars, is unsuitable to describe discontinuous syntactic relations, but can easily describe coordinative compounds in comparison to dependency structures. The results we obtained show that hybrid structures (combining phrases and dependencies in one parse tree [10]) are more appropriate for spontaneous speech. However, for some utterances it is better to generate semantic and pragmatic representations skipping the parsing stage because surface syntax constructions are too elliptical and vague. In further work we plan to combine phrase structure and dependency annotation to ensure its appropriateness for evaluation algorithms which implement different models of syntax and experiment with evaluating the parsing quality after speech repairs removal.

VI. Conclusion

This part presents results of making a benchmark for the task of testing grammar inference algorithms applied to Russian spontaneous speech. This problem is considered important because there in no parser for Russian spontaneous speech, from one side, and there is no data about performance of different grammar inference algorithms on Russian corpora, especially on spontaneous speech corpora. Future work in this domain is really large, because algorithms can be developed and evaluated not only in respect of performance, but also in

respect to time efficiency (that is a critical point for dialogue systems development), syntactic structure models (dependency trees, immediate constituents), metadata the parser uses (it can rely on grammar, semantics, or both), noisiness of the input (input transcripts may contain speech repairs, or the last may be removed). Obtaining precise results on each of these factors will allow, as we hope, to build a parser, capable to analyse spontaneous speech accurately and effectively.

ACKNOWLEDGMENT

This work was financially supported by the Russian Fund of Basic Research (RFBR), Grant 16-36-60055.

REFERENCES

[1] V.A. Tuzov. Computer Semantics of Russian. 2002. "http://www.wseas.us/ e-library/conferences/skiathos2002/papers/447-263.pdf".

[2] James Scicluna and Colin de la Higuera. PCFG Induction for Unsupervised Parsing and Language Modelling. 2014. "http://emnlp2014.org/papers/ pdf/EMNLP2014141.pdf".

[3] Jeffrey Heinz, Colin de la Higuera, and Menno van Zaanen. *Grammatical Interference for Computational Linguistics*. 2008.

[4] Lynn Carlson and Daniel Marcu. *Discourse Tagging Reference Manual*. "https://www.isi.edu/ marcu/ discourse/tagging-ref-manual.pdf".

[5] A.A. Kibrik and V.I.Podlesskaya. *Stories about dreams. Corpus based research of spoken Russian discourse (in russian) Rasskazi o snovideniyah Korpusnoe issledovanie ustnogo russkogo diskursa*. 2009.

[6] American Psychological Association. *The Problem of Non-Discreteness and Spoken Discourse Structure*. American Psychological Association, Washington, DC, 1983.

[7] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MTI Press, 2009.

[8] Ioan-Iovitz Popescu and Gabriel Altmann. *Some aspects of word frequencies*. 2009.

[9] Menno van Zaanen and Jeroen Geertzen. *Problems with Evaluation of Unsupervised Empirical Grammatical Inference Systems*, pages 301–303. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[10] Fei Xia and Martha Palmer. *Converting Dependency Structures to Phrase Structures*. 2014.