NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

Alena Fenogenova, Elizaveta Kuzmenko

A SEARCHING TOOL FOR RUSSIAN ERROR-ANNOTATED LEARNER ENGLISH CORPUS

BASIC RESEARCH PROGRAM

WORKING PAPERS

SERIES: LINGUISTICS
WP BRP 52/LNG/2016

SERIES: LINGUISTICS

Alena Fenogenova[1], *Elizaveta Kuzmenko*[2]

A Searching Tool for Russian Error-Annotated Learner English Corpus[3]

Abstract Learner corpora constitute an effective resource for specialists in fields of second language acquisition, foreign language teaching and corpus linguistics. They tend to get significant scholarly help from statistical tools of various kinds. However, for valuable usage of a corpus it should provide convenient and powerful tools for searching and manipulating data. In this paper we focus on searching tools, presented in *Russian Error-Annotated Learner English Corpus (REALEC)*, report our attempts to improve the format of the searching tools in our corpora. We also provide evidences that database search is much more efficient than common text search and demonstrate that search functionality in corpora is of great importance for research efficiency and extensive facilities.

JEL Classification: Z
Keywords: learner corpora, English as a second language, automatic search in corpora.

[1] National Research University Higher School of Economics. School of Linguistics. E-mail: alenush93@gmail.com

[2] National Research University Higher School of Economics. School of Linguistics. E-mail: lizaku77@gmail.com

[3] The article was prepared within the framework of the Academic Fund Program at the National Research University Higher School of Economics (HSE) in 2016 (grant 16-05-0057 «Learner corpus REALEC: Lexicological observations») and by the Russian Academic Excellence Project "5-100".

1 Introduction

Computerized learner corpora, an electronic collections of spoken or written foreign and second language learner data, constitute an effective resource for linguists and FLT specialists. It is a new brand of research which relates to the fields of second language acquisition (SLA), foreign language teaching (FLT) and corpus linguistics. The corpus-based studies of learner English is an area of research that attracts more and more scholarly interest [1], but, unfortunately, in Russian academic circles not much attention has been paid to this field so far.

Using learner corpora seems especially advantageous in scientific and pedagogical perspectives and has been proved to be highly effective [2]. Learner corpora tend to get significant scholarly help from statistical tools of various kinds. By combining insights from Second Language Acquisition theory and English Language Teaching practice with corpus linguistics methodology, researchers are able to describe interlanguage features, focus on language patterns that need more attention, offer new directions for language teaching. However, such corpora usually do not have convenient interfaces nor sufficient sets of searching tools, and creating them is not a trivial task. Lack of such tools is a disadvantage – it prevents researchers from pulling out relevant data from the corpus. In this paper we present the evidence from *REALEC(Russian Error-Annotated Learner English Corpus)* demonstrating that search functionalities in corpora are of great importance for research efficiency and extensive facilities.

The paper is outlined in the following way: in Section 2 we present an overview of the annotation tool *BRAT*, which the Russian Learner English Corpora is based on. Section 3 describes the corpus itself, its features and the problems that arise while working with it. In Section 4 we report our attempts to improve the format of the searching tools as well as provide evidence that the database search is much more efficient than common text search. Section 5 summarizes the work done and discusses our future plans for further improvement of searching tools in the corpus.

2 BRAT

The BRAT rapid annotation tool (BRAT) [4] is a special open-source text-annotation tool supported by Natural Language Processing (NLP) technology. It was developed to make annotation simple, intuitive, representative, accessible to non-technical users. *BRAT* is highly customizable system appropriate for different types of markup. It allows users to visualize annotated texts in real time, making it easy to see and edit error spans and other markup targets. It is commonly used for a variety of NLP tasks such as BioNLP task, Chunking, Language-Independent Named Entity Recognition, and many others. A set of large and reputable corpora were annotated using *BRAT*: Anatomical Entity Mention (AnEM) corpus, CellFinder corpus, Multi-Level Event Extraction (MLEE) corpus and its use in REALEC is well justified.

[4] `http://brat.nlplab.org`

BRAT implements a set of search functions: it allows users to search in the current document or in the whole collection for four different types of structures: text span annotations, relations, events, or simple text. The user can then select from searching for given text strings as whole words, for substrings, or for regular expression (regex). In regular expression mode, the given string is interpreted as a Perl Compatible Regular Expression that is matched against target text. For all these settings there is a simple user-friendly point-and-click interface. In addition, searching results can be displayed using keyword-in-context concordancing (with surrounding context of the specified size in a key word in context format).

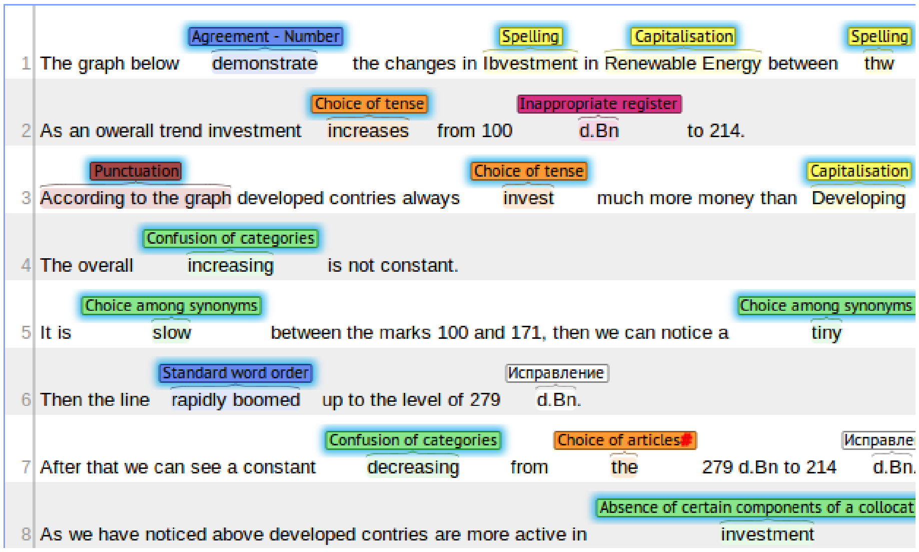
Nevertheless, *BRAT* was not originally created for managing and searching large text collections, because it was developed primarily as an annotation tool. Therefore, it does not support high quality searching tools. At the same time, search in corpora has a number of features that make its architecture rather specific [3]: for example, wide context and different annotation attributes should be taken into account while making search queries. Mere text search is not sufficient for various goals of using large corpora, and the reasons for it will be discussed in detail in the next section. The speed of processing leaves much to be desired as well. Besides, *BRAT* is fully supported only in *Google Chrome* browser.

3 REALEC: Russian Error-Annotated Learner English Corpus

REALEC is composed of English essays written by native Russian speakers, namely, students of the National Research University Higher School of Economic. The essays get error-annotated by ELT experts, and the creators of the corpus employed *FreeLing*[5] lemmatizer and part-of-speech tagger modules to assign POS and lemmas to word tokens of essays uploaded into the corpus. At the moment the corpus contains more than 360 thousand word tokens, 17 thousand annotated errors in almost 1300 essays. The corpus is available online at `http://realec.org`. REALEC error annotation scheme is hierarchical, and it consists of 4 layers: error type, error cause, linguistic negative impact caused by the error and the influence of the error on general understanding of the text. Qualified English teachers mark the essays and annotate them in accordence with the error classification inctruction, which was proved to be reliable [4]. *BRAT* framework was applied for annotation interface and visualization. You can seen an example of a text annotated in brat on Figure **??**.

One notable weakness of *BRAT* is its search interface, which is not very rich. Although a user can look for a particular entity (error type) in a document or in the whole collection, he or she cannot combine several entity types in the query, nor set multiple constraints, nor search for grammar, lemmas and their combinations as is conventional in all modern corpora. Even more frustrating is

[5] `http://nlp.lsi.upc.edu/freeling/`



1 The graph below demonstrate the changes in Ibvestment in Renewable Energy between thw

2 As an owerall trend investment increases from 100 d.Bn to 214.

3 According to the graph developed contries always invest much more money than Developing

4 The overall increasing is not constant.

5 It is slow between the marks 100 and 171, then we can notice a tiny

6 Then the line rapidly boomed up to the level of 279 d.Bn.

7 After that we can see a constant decreasing from the 279 d.Bn to 214 d.Bn.

8 As we have noticed above developed contries are more active in investment

Figure 1. English essay annotated with brat

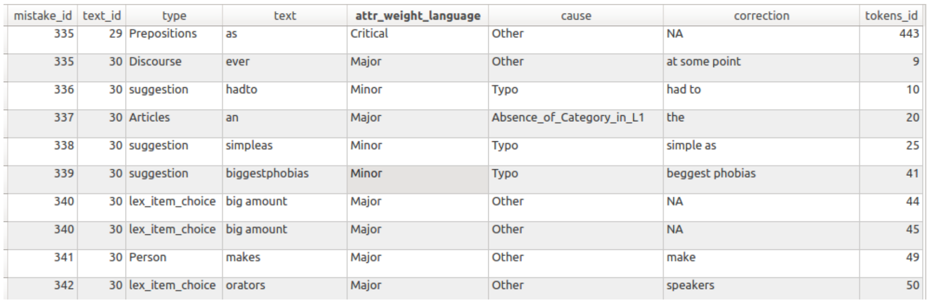
the fact that *BRAT* does not support any statistical information, thus, saving search results and downloading them is unavailable. As a result, right from the start of setting up the corpus there was a need for search tools to be improved [5]. The present paper aims to report the advancement of this improvement, the approach taken and results achieved in the next section.

4 Search system in REALEC

Most learner corpora do not have impressive, practicable and robust searching tools. This is not surprising, because it is not a trivial task, which is accounted for by the fact that large corpora have normally several layers of markup, including indexes, types of errors, tags and target objects. Thus, to get a relevant result, a user's query passes through a chain of different query layers. The architecture of the corpus may be complex, and it needs an integrated and extensible query system derived from different knowledge sources, access strategies, and usage situations.

So, the architecture of a large corpus consists of at least two parts: *logical* and *physical*[6]. Queries allow the user to describe the desired data on a physical level, leaving it with the database management system on a logical level to carry out planning, optimizing, and performing the physical operations necessary to produce the result.

Some of learner corpora still provide concordancing tools which are insufficient and do not implement the "separating" logic. Normally, corpora are static (they are seldom updated), which means that frequent search needs are simple and can be resolved by CQP-like technology [3]. CQP (corpus query processor) is a flexible and efficient workbench for managing and querying large text corpora. It provides access to multilevel annotations.



mistake_id	text_id	type	text	**attr_weight_language**	cause	correction	tokens_id
335	29	Prepositions	as	Critical	Other	NA	443
335	30	Discourse	ever	Major	Other	at some point	9
336	30	suggestion	hadto	Minor	Typo	had to	10
337	30	Articles	an	Major	Absence_of_Category_in_L1	the	20
338	30	suggestion	simpleas	Minor	Typo	simple as	25
339	30	suggestion	biggestphobias	Minor	Typo	beggest phobias	41
340	30	lex_item_choice	big amount	Major	Other	NA	44
340	30	lex_item_choice	big amount	Major	Other	NA	45
341	30	Person	makes	Major	Other	make	49
342	30	lex_item_choice	orators	Major	Other	speakers	50

Figure 2. Table "tags" in database

In our corpus we have taken the decision to divide the query system. We converted REALEC into the database format of SQL. It is a relational storage model which contains separate tables of data that are related to each other and can be manipulated by means of SQL queries. The structure of our database is presented on Figure 3.

The main entity in our database is Text, which is natural for documents collection. The only property of the Text entity is *text name.*

Every text in its turn is made up of tokens, therefore, the next entity in our database is Token. It is important to point out that tokens in the database table are ordered in the same way they appear in text. This helps to extract contextual information by taking some number of tokens before and after the token in a search query. The Token entity has two properties: its POS tag (and available morphological information) and the id of an Error entity, if this token is included in some error span. A Token can be a part of several errors, but it also can belong to no errors at all.

The last entity in the database is Error. The properties of an error are the following:

- error type;
- language weight, which denotes how serious an error is from the linguistic point of view;
- understanding weight, which denotes how gravely the error affects understanding of a text;
- cause of an error;
- suggested correction.

If we want to find which words appear to be part of an error of a particular type, we collect all the necessary cases from the table of Error entities, take

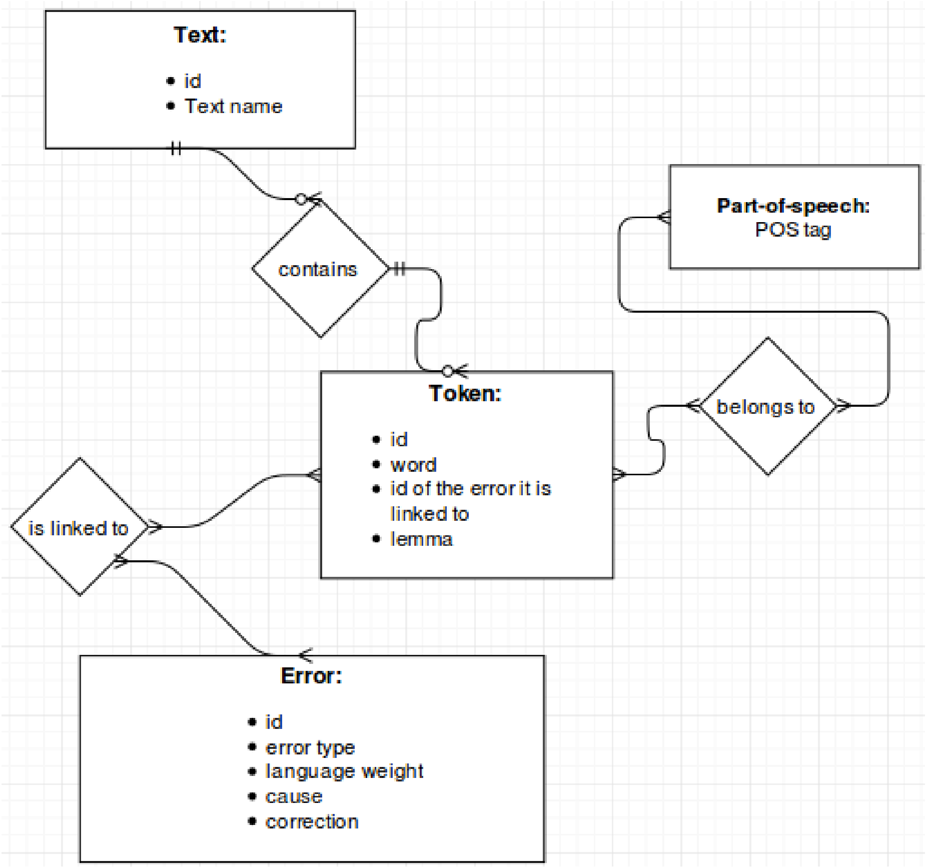


Figure 3. Entity relations scheme of the database for our corpus

context into account and take corresponding tokens from the table of Token entities.

All annotated texts are processed by the special script in Python language which parses texts and annotations and creates relative tables for the database. All in all, we created three tables (Figure 2):

1. *tags* with information about error type, text sequence, correction, cause of the error, mistake/token/text ids and weight of an error;
2. *texts*, which contains id and the name of the document;
3. *tokens* consisting of tokens, ids, grammatical information and lemmas.

Directing SQL-queries to the tables in the database allows quick, efficient and precise execution of queries of any complexity. Table 1 visually presents the comparison of functional facilities between the old version of searching tool in *REALEC* and the new one.

Table 1. Comparative table of available functions in old and new versions of REALEC

Functions	old version	new version
search for tokens	+	+
search for error types	+	+
search for lemmas	+	+
search for error attributes	-	+
search for error types and lemmas	-	+
search for error types in context	-	+
search grammar	-	+
display of statistics	-	+
save results and statistics	-	+
user-friendly interface	+	-

5 Conclusions and future work

To sum up, the searching tool currently implemented in REALEC is based on the *BRAT* interface, and it is not efficient enough and has to be improved. We decided to follow the trends in corpora architecture and create a database managing system to perform quick and precise search queries. Now it is possible to store a big amount of data, process the queries more efficiently, get all the necessary statistics and save search results. Naturally, common users cannot perform this kind of search themselves, because they need to know how to write queries in SQL. Therefore, our future plans are to create a user-friendly interface which will allow users to make intuitive queries which will further lead to performing SQL-queries. We believe that such architecture is going to greatly improve searching options in REALEC.

Acknowledgement This research was supported by *the National Reasearch University Higher School of Economics*. We thank our colleague *Olga Vinogradova* for her valuable comments.

References

1. Granger, S., Gilquin, G., Meunier, F.: Twenty Years of Learner Corpus Research. Looking Back, Moving Ahead: Proceedings of the First Learner Corpus Research Conference (LCR 2011). Volume 1. Presses universitaires de Louvain (2013)
2. Granger, S., Dagneaux, E., Meunier, F., Paquot, M., et al.: The international corpus of learner english. version 2. handbook and cd-rom. (2009)
3. Ide, N., Brew, C.: Requirements, tools, and architectures for annotated corpora. In: Proceedings of data architectures and software support for large corpora, Citeseer (2000) 1–5
4. Kuzmenko, E., Vinogradova, O., Kutuzov, A.: Evaluating inter-rater reliability for hierarchical error annotation in learner corpora. (2015)
5. Kuzmenko, E., Kutuzov, A.: Russian error-annotated learner english corpus: a tool for computer-assisted language learning. NEALT Proceedings Series Vol. 22 (2014) 87

6. Christ, O.: A modular and flexible architecture for an integrated corpus query system. arXiv preprint cmp-lg/9408005 (1994)

Contact details

1. Alena Fenogenova

National Research University Higher School of Economics (Moscow, Russia), School of Linguistics.

E-mail: alenush93@gmail.com

2. Elizaveta Kuzmenko

National Research University Higher School of Economics (Moscow, Russia), School of Linguistics.

E-mail: lizaku77@gmail.com

Any opinions or claims contained in this Working Paper do not necessarily reflect the views of HSE.