L2 text recommendation system for Russian language

Nikolay Babakov[[1]](#footnote-0), Natalya Isupova[[2]](#footnote-1),Anastasiya A. Bonch-Osmolovskaya[[3]](#footnote-2)

Olga Eremina[[4]](#footnote-3)

**Abstract**

Language learning is a complicated process which includes many actions aimed to accumulate learner’s real-life language experience. Reading is one of the most important language-learning process, but searching texts which are appropriate for the current level of the learner is a very time consuming task. We propose a system for automatic text recommendation for Russian as L2 learners, based on evaluation of learners' language competence.

Though the system has been designed for Russian language the general principles of the system can be transferred to any language.

**Keywords**: text recommendation, L2 texts reading.

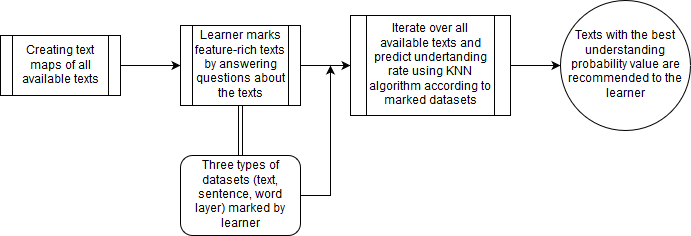
1. **Introduction**

Numerous L2 text recommendations approaches have been developed so far. The classic approach is based on learner passing CEFR-like test and the reading text with corresponding CEFR level. This approach is very straightforward, does not take many text complexity properties into account and most words positioning in definite CEFR level is quite an arguable question.

At the same time we have found some interesting approaches which used tests for further recommendation (Kurdi, M., 2018), but in that case a learner is supposed to read text and to manually score such parameters as syntaxis, morphology and etc. which are actually not so easy to evaluate.

We have tried to accumulate all experience of our colleagues and developed L2 text recommendation system which takes into account all domains of text reading complexity and provides the most accurate information about user knowledges in each domain which effects further recommendation (see Figure 1 ).

Figure 1. System outline



1. **Method**
   1. **Evaluation of learner language competence**
      1. *Textual markuping*

To collect the information about learner's language competence we use 10-15 texts with questions, related to them. The texts are ranged either within all possible CEFR levels or within the level the learner is supposed to have. There are several questions about each text learner should answer. The questions are designed in such way that there is one definite sentence in the text which is necessary and enough for correct answering to this question.

The texts for such a test are selected according to the amount of high frequency specific words contained inside each text. The ideal test should contain least amount of texts with such words which are most typical for the theme inside which we will perform the recommendation.

So when the user provides correct or incorrect answer to a question he/she namely puts “+” or “-” mark to the corresponding group of words which shows whether these words and the sentence is understood or not understood. Such a marking is distributed within all layers of text features and then used for collection of three datasets which thoroughly demonstrate learner knowledges

* + 1. *Learners competence datasets collection*

The next step is collection dataset which will project learner’s knowledges into three domains the text consists of: words, sentences and overall text features. All datasets’ vectors have target variable which shows whether the element of a text in corresponding layer is understood correctly or not.

*Collocations and words layer*

We use the collocations and words layer to find out the words or group of words a learner has understood or not understood when answered the questions.

Each word or collocation inside is marked with 0 (for incorrect answer) or 1 (fo correct) target variable

*Sentences layer*

Each sentence is represented as a vector, which includes percentage of non-trivial morphological features such as specifical parts of speech ( for Russian such parts of speech as participle) and other complicated language elements(which actually can vary for different languages, for Russian we used such elements as reflexive verbs or numerous nouns located next to each other with different case) and mean syntax dependencies. The target variable for each sentence is 1 for correct answer and 0 for an incorrect one.

*Text layer*

Each text is represented with a text complexity metrics such as LIX (Brown, J., Eskenazi, M., 2005) , type token ratio and also averaged syntax ratios of all sentences vectors.Target variable is the value within 0 and 1 which represents the percentage of correctly answered questions. For example, if there are ten question for a text and a lerner answered six of them correctly the corresponding target variable for such text will be 0.6

Table 1. Lerner knowledge dataset representation

|  |  |  |  |
| --- | --- | --- | --- |
| **element id** | **X** | **target variable** | **comment** |
| WORDS DATASET | | | |
| word\_1 | word2vec(He) | 1 | words from correctly answered sentence |
| word\_2 | word2vec(travels) | 1 |
| word\_3 | word2vec(alone) | 1 |
| word\_4 | word2vec(They) | 0 | words from incorrectly answered sentence |
| word\_5 | word2vec(were) | 0 |
| word\_6 | word2vec(foolished) | 0 |
| SENTENCES DATASET | | | |
| sentence\_1 | percentage of difficult to understand objects, POS and mean syntax dependencies | 1 | correct answer |
| sentence\_2 | 0 | incorrect answer |
| TEXTS DATASETS | | | |
| text\_1 | LIX, TTR, averaged sentences properties | 8/10 | percentage of correctly answered questions |
| text\_2 | 6/10 |

* 1. **Text recommendation process**
     1. *Text preprocessing*

Text recommendation process requires having all potentially recommended texts parsed to json-file

The text is preprocessed using UDpipe. We extract three types of features and create JSON text map which includes all of these features.

* + 1. *Recommendation algorithm*

The recommendation algorithm matches the vectros collected during competence evaluation with the vectors of text analysis. K-Nearest-Neighbours approach is applied to predict understanding rate for each potentially recommended text.

The recommendation algorithm automatically iterates over each text, extracts same feature levels (words layer, sentences layer, text level) and looks for each word’s, each sentence’s and text features vector’s top-5 most similar (according to cosine similarity) vectors in the evaluation dataset. The target variables of these top-5 most similar vectors are averaged proportionally to the similarity value and are used to predict understanding of each word, sentence of the text and the potentially recommended text as a whole. The process of matching the variables between two datasets is illustrated in the table 2

Table 2. Text understanding prediction overview for all three layers

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Dataset type** | **Prediction object** | **Result** | **Prediction result meaning** | **Recommendation reference** |
| words dataset | word | W, +-val [0...1] | the value corresponds to the normalized tf-idf which is positive for correct understanding and negative for incorrect | percentage of correctly understood words values |
| sentences dataset | sentence | S, [0...1] | 0 for non-understanding  1 for understanding | percentage of correctly understood sentences |
| text dataset | text | T, [0...1] | percent of correctly understood sentences | percentage of correctly answered questions |

So finally for each analyzed text we get a vector with three numbers which range from 0 to 1. The result can be illustrated as follows [0...1,0...1,0...1]

It is assumed that for good understanding of the text it is necessary to be familiar with 80% of the material or grammar rules. So the closer each calculated value to 0.8 the more likely the texts is recommended. We call this value Understanding Deviation and mark it as UD. The formula for calculating UD shown below

(1)

where

W - percentage of words in examined text with correct understanding prediction

S - percentage of sentences in examined text with correct understanding prediction

T - percentage of abstract text related questions with correct answer prediction

FInally when all texts are analyzed there is UD corresponding to each text in a text database. The texts are sorted by value and three texts with the least standard deviation are recommended to the learner.

1. **Model performance evaluation**

We use two different approaches for evaluating of the of the recommended text understanding. The first one is providing text related questions, referring to definite sentences of the text. The second approach is to ask the user to mark the sentences which include the biggest amount of the predefined features within some scale like “Don’t understand the sentence at all … Understand general sense … Fully understand”. Choosing of definite approach depends on time availability of test designers. The first approach which implies questions to the recommended texts is more time consuming because the questions should be prepared manually.

After applying any of the named feedback model we get the same knowledge datasets but it has different application. It is used for model performance evaluation.

The model is assumed to work well if the learner answers 80% of questions related to the recommended text or mark 80% of sentences from the text as “Fully understand”

The model itself can be tested by providing learners with the recommended texts, collecting their answers and calculating standard deviation from 0.8.

So in fact the formula is the same with the previous one but we handle not predicted but real values.

(2)

W\* - percentage of correctly understood words’ tf-idf normalized value in recommended text

S\* - percentage of correctly understood sentences in recommended text

T\* - percentage of correctly answered text related questions

Obviously, the less UD\* value is the better this system has worked. That is why the final metric of this is R-square-like (let’s name it UR-square which stand for Understanding Rate) and is calculated using the following formula

(3)

For now we have performed tests with 20 learners (each of them got three texts recommended for reading) and the UR-square achieved turned out to be 0.75

1. **Conclusions**

In this paper we presented new approach for L2 text recommendation according to learners skills which are checked using most natural way - texts questions answering.

The next steps will be applying more tests with learners of different backgrounds, applying the approach to other languages and tuning the test parameters to reach higher accuracy.

**References**

Brown, J., Eskenazi, M. (2005) *Student, text and curriculum modeling for reader-specific document retrieval.* Proceedings of the IASTED International Conference on Human-Computer Interaction, Phoenix

Kurdi, M. (2018). *A reading recommendation system for ESL learners based on linguistic features*. Proceedings of the 31th International Florida Artificial Intelligence Society Conference, Melbourne, Florida

1. Nikolay Babakov, Moscow, Russian Federation, bbkhse@gmail.com [↑](#footnote-ref-0)
2. Natalya Isupova, Moscow, Russian Federation, nata\_isupova@inbox.ru [↑](#footnote-ref-1)
3. Anastasiya A. Bonch-Osmolovskaya, Moscow, Russian Federation, abonch@hse.ru [↑](#footnote-ref-2)
4. Olga Eremina, Moscow, Russian Federation, oeremina@hse.ru [↑](#footnote-ref-3)