*Workshop on Linguistic and Neurocognitive Resources (LiNCr2020)*, pages 28–37
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

# Language Models for Cloze Task Answer Generation in Russian

**Anastasia Nikiforova, Sergey Pletenev, Daria Sinitsyna,**
**Semen Sorokin, Anastasia Lopukhina, Nicholas Howell**
National Research University Higher School of Economics
Moscow, Russian Federation

## Abstract

Linguistics predictability is the degree of confidence in which language unit (word, part of speech, etc.) will be the next in the sequence. Experiments have shown that the correct prediction simplifies the perception of a language unit and its integration into the context. As a result of an incorrect prediction, language processing slows down. Currently, to get a measure of the language unit predictability, a neurolinguistic experiment known as a cloze task has to be conducted on a large number of participants. Cloze tasks are resource-consuming and are criticized by some researchers as an insufficiently valid measure of predictability. In this paper, we compare different language models that attempt to simulate human respondents' performance on the cloze task. Using a language model to create cloze task simulations would require significantly less time and conduct studies related to linguistic predictability.

## 1. Introduction

Nowadays language models are the most powerful instrument to transfer knowledge. Mostly pre-trained neural network models are more accurate in any type of task. This tendency in language processing - usage of language model (LM) weights as a part of base model weights - took place after word2vec announcing. Today there are three main ways to use pre-trained LM in different natural language processing tasks:

- Use pre-trained LM as universal embedder for text/sentence
- Fit pre-trained LM on new data (domain adaptation) and also use as an embedder
- Fit pre-trained LM as a part of a more complex and specific model

It is important to note that the resulting system is dependent on the quality of the underlying LM; thus strategies to compare models are also in demand. We propose a new comprehensive way to explore properties of different language models. Comparison of langauge models is not a new topic, and there are many different measures of their quality. Popular modern analyses are the Google analogy task for word2vec (Mikolov et al., 2013) and now GLUE tasks (Wang et al., 2018). However, we want to check the generative ability of several different types of LM and compare them.

One of the key terms in natural language understanding and speech generation is predictability. In cognitive linguistics, it implies a confidence degree of a language unit (word, part of speech, etc.) that can take next place in the sentence (or text). This property of the token in the context is usually measured in terms of the theory of probability, and it also has some well-known probabilistic properties. For example, the sum of the probabilities of all words which can or cannot (in terms of common sense) follow the left context is equal to one. A quarter-century ago these assumptions led to the emergence of the first artificial language models (ALM).

Research papers in the field of cognitive science have shown that correct prediction of the next word while reading a sentence simplifies the perception of a language unit and its integration into the context. Incorrect prediction can lead to a re-analysis of the context which is why language processing is slowed down. However, the types of dependencies between these two facts are still not well studied.

Nowadays, in linguistic and cognitive studies, to obtain data describing the probabilistic distribution of lexical units (for a specific context), artificial language models of various architectures are used or cloze tests are conducted. In a cloze test, participants asked to replace a missing language item in a sentence. The cloze test is frequently criticized for lack of coverage; nevertheless, in terms of common sense, it is cloze test which uses the so-called “human” linguistic mechanisms of speech generation to collect the data. The ALM, in contrast, is basically a set of various mathematical algorithms applied to the text corpus.

Our analysis and gold-standard is a Russian cloze test conducted by Laurinavichyute et al (Laurinavichyute et al., 2018) from an eye-tracking study. We take the results of the cloze task as a proxy for the underlying probability distribution of next-word continuations of partial sentences. The LM based on these answers we will call ”human-like”. So the uniqueness of the research is that we can compare artificial language models with the model which approximates real human expectations about the next word for a given sequence.

It is important to note that cloze tasks require a major time commitment and are financially expensive. One of the goals of this study is to find out whether the actual human respondents can be replaced by an artificial language model trained on a large corpus, or, whether language models can simulate human performance on this task. Our study compare several language models across four “levels” of prediction: lexical (distribution of surface forms), part-of-speech (distribution of morphological class), and two classes of semantic prediction.

Besides having importance in the field of neuro- and psycholinguistics, cloze task answer generation could also potentially be used for OCR and hand writing recognition, as mentioned in the paper by Kuperberg and Jaeger (Kuperberg and Jaeger, 2016).

## 2. Related Works

As Kuperberg and Jaeger claim in their “What do we mean by prediction in language comprehension?” (Kuperberg and

Table 1: Example of stimuli sentence from RNC in cloze task with probabilities of the next word. Stimuli: *"А промывать манную крупу перед тем, как варить ее, не пробовали?"* (English translation: *"Have you tried to rinse semolina before boiling it?"*)

| Stimulus | Next word | Predictability |
|---|---|---|
| А | промывать | 1,99E-07 |
| А промывать | манную | 9,95E-06 |
| А промывать манную | крупу | 0,091529563 |
| А промывать манную крупу | перед | 0,000779675 |
| А промывать манную крупу перед | тем | 0,015035226 |
| А промывать манную крупу перед тем, | как | 0,966154218 |
| А промывать манную крупу перед тем, как | варить | 0,011867962 |
| А промывать манную крупу перед тем, как варить | ее | 0,04090891 |
| А промывать манную крупу перед тем, как варить ее, | не | 0,146951959 |
| А промывать манную крупу перед тем, как варить ее, не | пробовали | 0,000829122 |

Jaeger, 2016), the reaction time is in direct proportion with the predictability of the word: the more predictable the word is the faster is the reaction. Moreover, predictability of a word or a context defines fixation time in eye-movement studies as a result of the language comprehension process. This implies that language comprehension must be predictive. The authors also state that as the previous context expands, the predictability of the next word increases leading to - in cloze tests - higher accuracy of predicting the next word, and - in eye-movement experiments - to shorter fixation duration.

The literature contains several different algorithms for cloze answer generation. In (Zhou et al., 2018) the authors state the importance of next word prediction in language modeling and its potential contribution to OCR and handwriting recognition. The authors enhance existing models with ELMo and BERT language models and train on the CLOTH dataset of cloze tests. BERT models show the highest performance (0.86 and 0.83 accuracy scores on test dataset for BERT Large and BERT Base respectively), as this model was initially trained to recover masked tokens in text. At the same time, the ELMo model's poor performance could be due to the lack of parameter tuning and the fact that ELMo was trained for the next sequence word prediction.

An LSTM-based model for cloze-style machine comprehension is proposed in (Wand et al., 2018). The model consists of document hierarchical structure and dynamic attention mechanism for building the representations between the document and the question. Despite the two-layer LSTM model with attention outperforms one-layer model, the final best accuracy score is still only 0.76 which could be improved by future modifications to the model.

# 3. Methodology

## 3.1. Cloze Probabilities

Cloze task described in (Taylor, 1953) is an experiment in which one or more words are removed from a sentence and the participants are asked to fill in the missing content. It is commonly carried out for assessing native speakers of a language, which is aimed to understand respondents' comprehension of a language and their ability to predict missing portions of written texts (Laurinavichyute et al., 2018). This experiment presumes that native speakers can understand context and vocabulary to identify the correct semantic field or part of speech of a missing word.

We used the dataset with cloze task answers from (Laurinavichyute et al., 2018). The dataset is based on 144 sentences randomly selected from the National Corpus of the Russian Language (RNC, ruscorpora.ru) - an online corpus of Russian texts with extensive search options. These sentences were slightly edited: the authors replaced rare infrequent words with more frequent ones and shortened the sentences when they exceeded the preset maximum length of 13 words. The stimuli sentences were subjected to the cloze task experiment. Respondents were asked to successively predict the next words for each context. An example of stimuli that were shown to the participants is presented in Table 1. with corresponding correct next words and calculated predictability scores.

Each context received from 10 to 100 responses, not all of which matched the correct word. The predictability of each next word was computed as the number of correctly predicted words divided by the total number of predicted words. The Laurinavichyute et al. article and the full list of sentences used in the study can be found.

## 3.2. Corpus-Based Probabilities

For computing corpus-based probabilities, different model types and training corpora were selected. The goal of these combinations is to represent some dependencies (if they exist) between model architecture, vocabulary and to compare results.

In this research, we were solving the task of language modeling - the task of predicting the next word given the corpus. Several models perform well of this task type, including HMM, LSTM, and BERT. We used pre-trained models on our data to predict the next word for each context. These models were trained on different corpora to see how corpora influence model performance.

*Hidden Markov Model*

Markov chain theory is increasingly used in real-world computing applications as it provides a convenient way to capture pattern dependencies in pattern recognition systems. For this reason, Markov chain theory is suitable for natural lan-

Table 2: Corpus statistics. RNC is the Russian News Corpus, and NCRL is the National Corpus of the Russian Language.

| **Name** | **Texts** | **Size** (GB) | **Mean length** |
|---|---|---|---|
| RNC | 470k | 2,928 | 176 |
| NCRL | 111k | 3,210 | 2341 |
| (agg) | 581k | 6138 | 1258 |

guage processing (NLP), where data consists of repeating sequences of symbols or words.
In this case, we are using bi- and tri-grams HMM not for PoS-tagging but the prediction of the next word. To eliminate out-of-vocabulary errors in our HMM models, we will use Good-Turing smoothing.

*LSTM*

A one-layer long short-term memory (LSTM) recurrent neural network model was used (Jozefowicz et al., 2016) to create a list of predictions for each word in the same 144 stimuli sentences. The dimensions of the model are 2048 for the hidden layer and 512 for the input and output layers.

*BERT*

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) is trained on a masked language modeling objective. Unlike a traditional language modeling objective of predicting the next word in a sequence given the history, masked language modeling predicts a word given its left and right context. Because the model expects context from both directions, it is not obvious how BERT can be used as a traditional language model (i.e. to evaluate the probability of a text sequence) or how to sample from it. We test several ideas: give the model all of the content, except masked word, and using the technique(Wang and Cho, 2019) to rework BERT as a classical language model.
For experiments, we used BERT trained on the Russian Wikipedia corpus (Wikipedia, 2019). To show differences between models, we fine-tuned BERT with our corpora.

## 3.3. Corpora

The Russian News Corpus (Shavrina and Shapovalova, 2017) includes newspaper articles published in the 2000s. The National Corpus of Russian Language (Apresjan et al., 2006) includes written texts from the middle of the 18th to the middle of the 20th century.
Some corpus statistics are presented in Table 3.3..
All of our models (except LSTM) were trained separately on the two corpora, and on their combination. Thus we examine these models:

- Hidden Markov Models
  - Bigram HMM on RNC
  - Bigram HMM on NCRL
  - Trigram HMM on RNC
  - Trigram HMM on NCRL
- BERT-based models
  - no fine-tuning
  - BERT fine-tuned on RNC
  - BERT fine-tuned on NCRL
  - BERT fine-tuned on both
- custom LSTM model
- hybrid model
  - Bigram HMM on NCRL + BERT fine-tuned on NCRL

The custom LSTM model was trained on a blinded NCRL (excluding the sentences chosen for stimuli) and the RNC. Overall, the training corpus consisted of 577 million tokens. The model was tested on 1000 sentences from the OpenCorpora project (Bocharov et al., 2011) with 1,9 million tokens from newspaper articles, Russian Wikipedia, texts from blogs, fiction, non-fiction, and legal documents.
Among all, there is a Bigram HMM on NCRL + BERT fine-tuned on the NCRL model, which is by structure a combination of a bigram HMM and a BERT model. These models were joined based on the best performance of both models: probability distributions of HMM are used for contexts with length less than 6 tokens, and BERT is used for longer contexts.

*Renormalization of probabilities*

To evaluate each model's predictions, we took the first 30 most probable words. Each probability was renormalized by dividing originally computed word-wise probability by the sum of probabilities of the first 30 words. This way the sum of probabilities the selected words would equal to 1.

## 3.4. Overview of Used Metrics

*Mean accuracy*

The metric is used to compute the mean of correct word prediction across all contexts. Range of values from 0 to 1. It was computed as a mean value of the array of accuracies.

*Absolute number of correct word predictions*

The metric represents the number of contexts for each prediction of the correct word is non-zero.

*Context consistency*

The metric represents the proportion of "context consistency". It can be interpreted as the answer to the next question: "How many contexts coincide assuming that prediction of the correct word (for each of them) is not equal to zero for a certain model pair?

*Kolmogorov-Smirnov test*

In our study, we used the two-sample Kolmogorov-Smirnov test to find out whether two underlying one-dimensional probability distributions of model predictions differ. The null hypothesis of the Kolmogorov-Smirnov test is: both samples of predicted words come from a population with the same probability distribution.
The Kolmogorov–Smirnov statistic is

$$D_{n,m} = \sup_x |F_{1,n}(x) - F_{2,m}(x)|,$$

where $F_{1,n}$ and $F_{2,m}$ are the empirical distribution functions of the first and the second sample respectively, and sup is the supremum function. The null hypothesis is rejected at level $\alpha$ if

$$D_{n,m} > c(\alpha)\sqrt{\frac{n+m}{nm}},$$

where n and m are the sizes of first and second sample respectively, and where c(α) is the inverse of the Kolmorogov distribution at α, which can be calculated as

$$c\left(\alpha\right) = \sqrt{-\frac{1}{2}\ln\alpha}.$$

The advantage of the Kolmogorov-Smirnov test is that, unlike the t-test, it can catch the difference between Gaussian distributions with similar means but different variances.

This metric was used to compare both lexical and word class probability distributions of cloze task, LSTM, HMM models pairwise. The results of the metric are listed in the Results section of this article.

*Kullback–Leibler Divergence*

Cross-entropy loss, or log loss, measures the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss increases as the predicted probability diverge from the actual label. The cross-entropy shows the difference between probability distributions p and q. Kullback and Leibler defined a similar measure now known as KL divergence. This measure quantifies how similar a probability distribution p is to a candidate distribution q. We used KL-divergence to compare several language models.

*Cosine Similarity*

Cosine similarity was used to measure the closeness of semantic vectors of predicted words between different models. It is widely used for calculating the distance between two words. In our study, cosine similarity was used to find the semantic probability of predicting the word which is semantically close to the target word.

## 3.5. Part-of-Speech Probabilities

Word class probabilities were computed as follows: each word in the model's vocabulary ($N = 500000$ most frequent words in the training corpus) and each word in the stimuli sentences was tagged for word class and morphological features using PyMorphy2 analyzer (Korobov, 2015) and the predictions of the model were compared to the annotation of the target words. A word class match was coded if the predicted and target word belonged to the same word class. The probability of a word class was computed by summing probabilities of all words in the model's vocabulary which had the morphosyntactic feature in question. For example, to estimate the word class probability in a sentence "A mobile __", where "phone" is the target word, we would sum up probabilities of all nouns in the model's vocabulary. For morphologically ambiguous words (e.g., рот 'mouth' in the nominative or accusative singular), all possible variants were considered in the probability estimation.

## 3.6. Probabilities for OBJECT-VERB-FUNCTIONAL-MODIFIER

We have also tried to use different tags for our part-of-speech tagging. Instead of using all of the tags, we thought we could use a more generalized set of object, verb, modifier, and functional word, because when a person mentally chooses the next word, they might not think in terms of the usual parts of speech, but choose generally an object, or a description of an object, or a verb, or just some functional word.

Firstly, we converted all of the modified Pymorphy tags into 4 general sets: 'ADJ', 'ADVB', 'NUMR' were generalized to 'MOD'; 'INFN' and 'PRED' - to 'VERB'; 'NPRO' to 'NOUN' (Object); 'PREP', 'PRCL', 'CONJ' and 'INTJ' to 'FUNC' for each context. Then, we have counted probabilities of these tags in the same manner as the usual parts of speech.

We noticed that the generalized probabilities were overall higher than with modified Pymorphy tags - further we will refer to it as OMVF (object-modifier-verb-functional).

## 3.7. Semantic Comparison

After lots of trial and error, it was decided to use semantic vectors for the comparison of cloze task results with other models in the semantic aspect. All words were mapped into a vector space of the model pre-trained on Wikipedia texts (Arefyev et al., 2015). The comparison itself needed to be dynamic because for each context a different amount of words should have been chosen.

To compare semantic vectors for each context, firstly, we cleared all of the words so there would be no digits, meaningless letters and punctuation. Then, we have built a function, which:

1) Extracts first 10 words (we have decided that that is the maximum amount of words for each context that would have meaningful probabilities as all of the words after the first ten for each model have their probabilities tends to zero) for each of 1219 contexts;

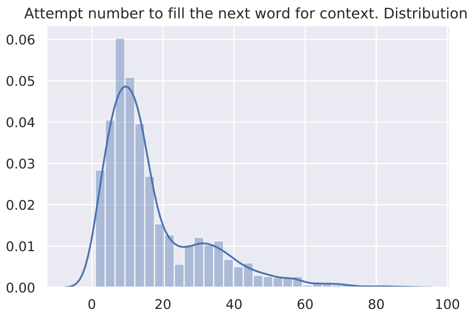
2) Computes the mean probability of remaining words;

3) Counts the difference between the probabilities of the first and last word and then the difference between mean probabilities of the previous word and the next one;

4) Decides what amount of words vectors to use for each context based on how different the first-last word difference and the mean probability difference is – if the latter is lesser than the former, the function checks the next difference, if not – the previous amount of words would be used for semantic comparison.

After the decision on the number of words was made, with the help of `gensim` in Python a vector for each word was extracted.

Then, MiniBatchKmeans (a modified version of the K-means algorithm, that uses mini-batches to reduce computation time, while at the same time trying to optimize the same goal function (Béjar, 2010)) algorithm in sklearn was used to find "cluster centers", or mean semantic vector for one model for each context. And at last, we computed cosine similarity (also with sklearn library) for each pair of semantic vectors for each pair of models.



# 4. Results

## 4.1. Quantitative and Qualitative Analysis of Cloze Task Language Model

First of all, it is necessary to establish how many different answers on average are available for each context in a language model based on a cloze task, since it is necessary to determine how many of the most likely words will be explored in artificial language models and test the hypothesis about the dependence of predictability on the length of the context.

The mean quantity of predictions for the language model built on cloze-task results is 17 words.

In Table 4.1., contexts with minimum variance in filling are listed. It is worth emphasizing that in all these cases there was no variance in respondents' answers, i.e. all respondents gave the same one answer for these contexts. What is more, this predicted word was the original word from the corpus.

We classified these contexts based on their constraining ability:

- Semantically constraining contexts (contexts #3, 4, 6, 8)
- Syntactically constraining contexts (context #1)
- Idiomatically constraining contexts (contexts #2, 5, 7, 9, 10)

The maximum number of different answers were received for the "на болотах" ("On the Swamps") context - 87 words, which is explained by the absence of any limiting semantic properties of the context.

In this regard, the study of the artificial language model distribution is meaningless, as it will always be uniform, to say, the indicator for each context in similar histograms will be equal to the size of the vocabulary.

Another important aspect of the study of the linguistic model of the cloze task is the relationship between the length of the context and the probability of predicting (i.e., predictability of) the correct word.

The regression line reflects a high value of predictability for contexts of length both less and more than five lexical units. However, the lack of correlation is worth emphasizing. We received the Pearson correlation coefficient score of 0.323 and Spearman correlation coefficient of 0.363.

According to the results of our experiment, the closest probability distribution of the correct word of all the models was achieved with BERT trained on the literary corpus.

In this case, there is practically no correlation: Pearson correlation score is 0.09, and Spearman correlation is 0.08.

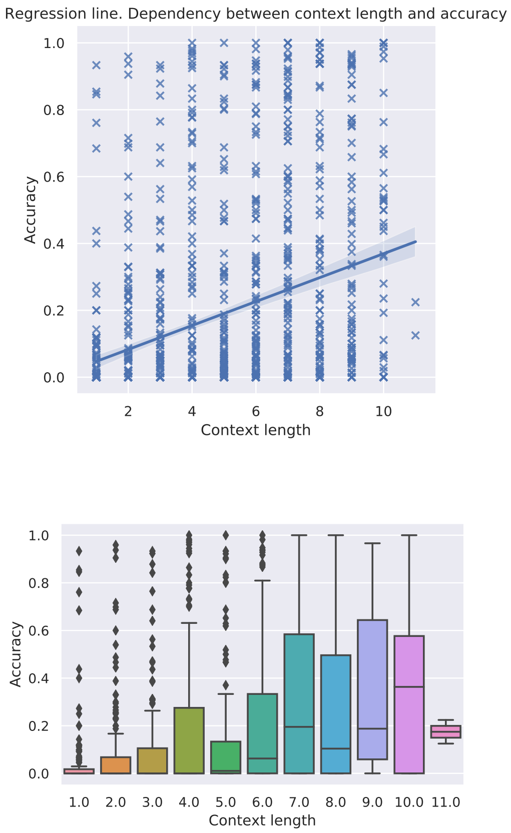


Figure 1: Context length vs. lexical accuracy.

Each model was evaluated by two measures: mean accuracy of model predictions and an absolute number of correct word predictions. For computing mean accuracy, the mean of correct answer probabilities was taken. In case of an absolute number of correct word predictions, a model achieved +1 score if there was at least one correct answer among all predictions.

## 4.2. Model Comparison on the Lexical Level

*Mean Accuracy*

Figure 2 below shows a bar chart of the mean accuracy scores of each model on the lexical level. As the goal of our study was to build an algorithm, which would be the closest approximation of the cloze task results (18% accuracy), we can see that BERT (not a language model one) model scored better than the others. Interestingly, all HMM model architectures showed low results on the lexical level.

It is noticeable that BERT mean accuracy results are higher than the cloze task score. This can be explained by the fact that the model was trained on a large number of written texts and thus had a higher chance to guess the correct word. Following this assumption, it is possible to infer respondents' active vocabulary size is lower than the model's vocabulary. Also, we can make a hypothesis that the process of word retrieval by humans and by the model is performed differently, as respondents do not always respond with the most probable answer.

*Absolute Accuracy*

In terms of the absolute number of predicted words, in the

Table 3: Contexts in which minimum variance is observed in filling by one lexical unit.

1. Какие главные лекарства должны входить (в)
*What are the main drugs that should be included (in)*
2. В современном обществе семья и школа оказывают большое (влияние)
*In modern society, family and school have a large (influence)*
3. Зачем ему звонить если откликается спокойный женский (голос)
*Why would he call if a calm female (voice) answers*
4. Ирине досталась отдельная комната в двухкомнатной (квартире)
*Irina got a separate room in a two-room (apartment)*
5. Они не ели целый (день)
*They haven't eaten all (day)*
6. Во избежание ожогов надо нанести на лицо небольшое (количество)
*To avoid burns on the face, apply a small (amount)*
7. Дрозды и скворцы начали вить семейные гнезда неподалеку друг от (друга)
*Blackbirds and starlings began to twist family nests not far from each (other)*
8. Собаку виновницу случившегося приказали сечь хотя в чем была ее (вина)
*The dog responsible for the incident was ordered to be beaten, although it wasn't really her (fault)*
9. С нескрываемой едкой иронией отзываются они друг о (друге)
*With undisguised caustic irony, they speak of each (other)*
10. Олень бродил среди берез жевал талый (снег)
*The deer wandered among the birches chewing melting (snow)*

Figure 2: Mean accuracy histogram, lexical level.

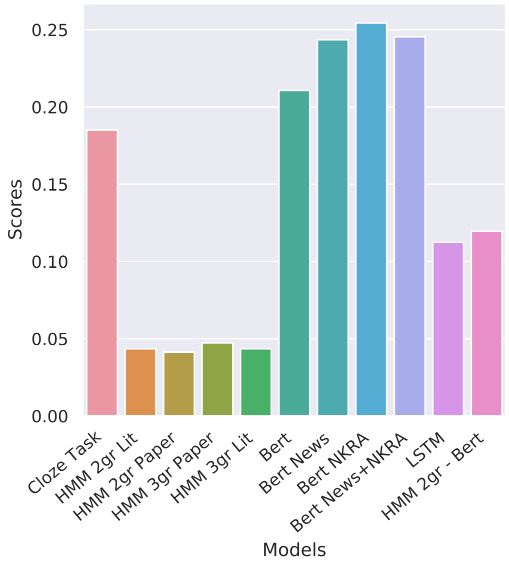
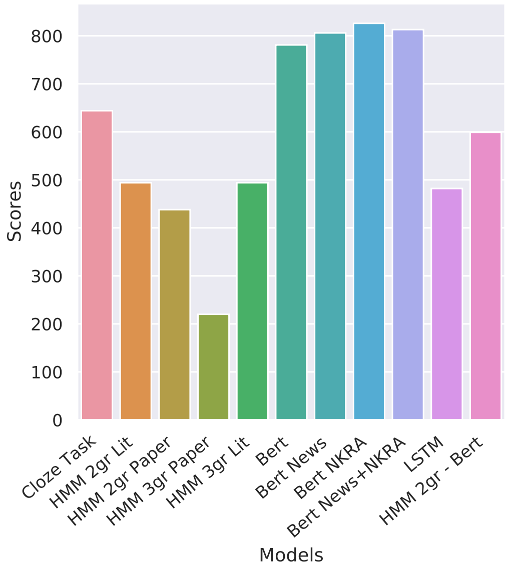


Figure 3: Absolute accuracy histogram, lexical level.



cloze task around 625 contexts were given at least one correct prediction. The closest to that are the results of the bigram HMM model combined with BERT (with around 545 contexts with at least one correct prediction) and raw BERT (with about 725 contexts with at least one correct prediction).

*Model Consistency*

Next, we compared models' performances using an inclusion-exclusion principle to find the percentage of overlapping answers between different models. The result of this comparison is shown on the heat map below.

The heat map reflects information on pairwise model comparison, however, we are mainly interested in how close the models are to the cloze task model. The comparison showed that the models with the the largest overlap with the cloze task are bigram HMM model combined with BERT (58% of overlap) and BERT trained on RNC.

*Kolmogorov-Smirnov Tests*

At the next step, we performed Kolmogorov-Smirnov testing to find the similarity in the probability distributions of the model predictions. Figure 5 also reflects the sum value of the pairwise comparison of the contexts using Kolmogorov-Smirnov testing. For convenience, all the values were normalized. Observations show that the closest probability distribution is seen in the LSTM model. However, as we see, there is a variation in metric values from 0.9 to 6.2 for different models. Thus, we can assert that the difference between all models and a cloze task is rather large and in some way unacceptable.

Figure 4: Overlap heatmap, lexical level.

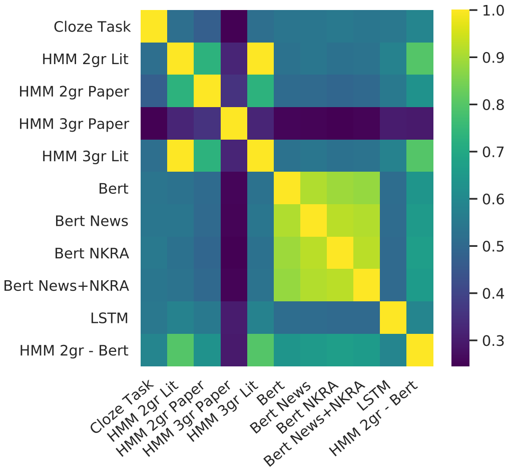
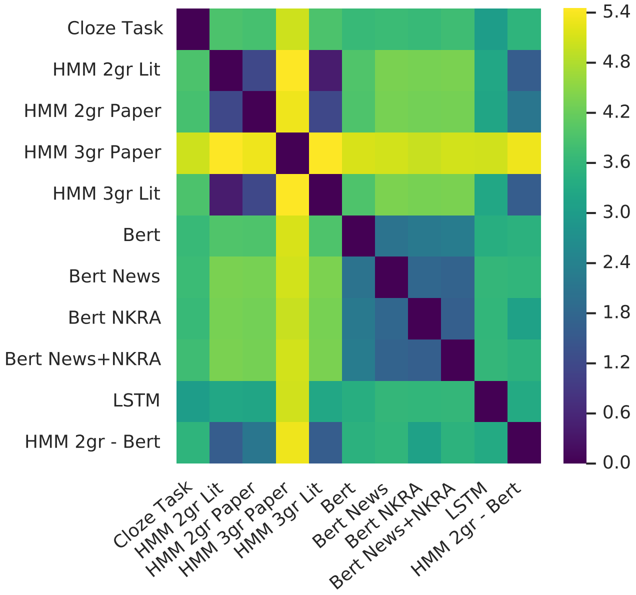


Figure 5: Kolmogorov-Smirnov tests, lexical level.



*Cosine Similarity*

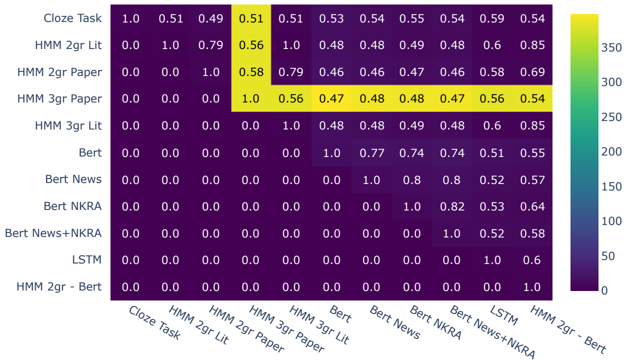
To measure model predictions on the semantic level, the cosine similarity between each context's predicted words centroid vector was found. The number of words was selected dynamically for each context by maximizing vector significance with the minimum words. Figure 6 reflects the results.

*Kullback–Leibler Divergence*

Due to the fact that all of our models are word-level, and in order to lower the casing variability we've combined all our vocabularies into one. For this compound vocabulary, we calculated the KL divergence of our models.

Table 4.2. shows the scores for three different models with bigram HMM trained on NCRL showing the best results. Unfortunately, this heat map does not show us changes in the language model (LM) distances context-lengthwise. Top 3 lengthwise LM distances from cloze are shown in Table 4.2.. As we see, the bigram models start to increase the distance from 1 to 6 context length, but BERT, on the contrary, starts to lower the distance up to 6 context length. For this case, we merged the bigram model and BERT at length equal to 6.

Figure 6: Cosine Similarity between each pair of models .



| | Cloze Task | HMM 2gr Lit | HMM 2gr Paper | HMM 3gr Paper | HMM 3gr Lit | Bert | Bert News | Bert NKRA | Bert News+NKRA | LSTM | HMM 2gr - Bert |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Cloze Task | 1.0 | 0.51 | 0.49 | 0.51 | 0.51 | 0.53 | 0.54 | 0.55 | 0.54 | 0.59 | 0.54 |
| HMM 2gr Lit | 0.0 | 1.0 | 0.79 | 0.56 | 1.0 | 0.48 | 0.48 | 0.49 | 0.48 | 0.6 | 0.85 |
| HMM 2gr Paper | 0.0 | 0.0 | 1.0 | 0.58 | 0.79 | 0.46 | 0.46 | 0.47 | 0.46 | 0.58 | 0.69 |
| HMM 3gr Paper | 0.0 | 0.0 | 0.0 | 1.0 | 0.56 | 0.47 | 0.48 | 0.48 | 0.47 | 0.56 | 0.54 |
| HMM 3gr Lit | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.48 | 0.48 | 0.49 | 0.48 | 0.6 | 0.85 |
| Bert | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.77 | 0.74 | 0.74 | 0.51 | 0.55 |
| Bert News | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.8 | 0.8 | 0.52 | 0.57 |
| Bert NKRA | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.82 | 0.53 | 0.64 |
| Bert News+NKRA | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.52 | 0.58 |
| LSTM | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 | 0.6 |
| HMM 2gr - Bert | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 |

Table 4: KL scores for three different models, all trained on the National Corpus of the Russian Language. "Context provided" is in number of tokens.

| Context | Bigram HMM | BERT | LSTM |
|---|---|---|---|
| 1 | 1.34 | 2.17 | 2.01 |
| 2 | 1.57 | 2.10 | 1.78 |
| 3 | 1.84 | 2.07 | 1.84 |
| 4 | 1.79 | 1.93 | 1.81 |
| 5 | 1.91 | 2.02 | 1.92 |
| 6 | 1.86 | 1.88 | 1.87 |
| 7 | 1.99 | 1.77 | 1.89 |
| 8 | 1.73 | 1.69 | 1.80 |
| 9 | 1.97 | 1.55 | 1.63 |
| 10 | 2.46 | 1.71 | 1.80 |
| 11 | 2.79 | 2.68 | 2.17 |

## 4.3. Model Comparison on the PoS Level

*Mean Accuracy*

The performance of all models, except for LSTM, at the parts of speech level, has significantly and proportionally increased. This is due to a decrease in the set of classes for which classification occurs. At this stage, the first 30 words of each model were tagged for parts of speech. Overall, there were 16 word classes. Notably, BERT linguistic models have the highest scores.

*Absolute Accuracy*

Table 5: Distance in the KL-metric between the cloze task and language models.

| Model | KL-distance to cloze |
|---|---|
| HMM | 1.79 |
| BERT | 1.93 |
| LSTM | 1.84 |
| HMM + BERT | 1.71 |

Figure 7: Mean accuracy histogram, part-of-speech level.

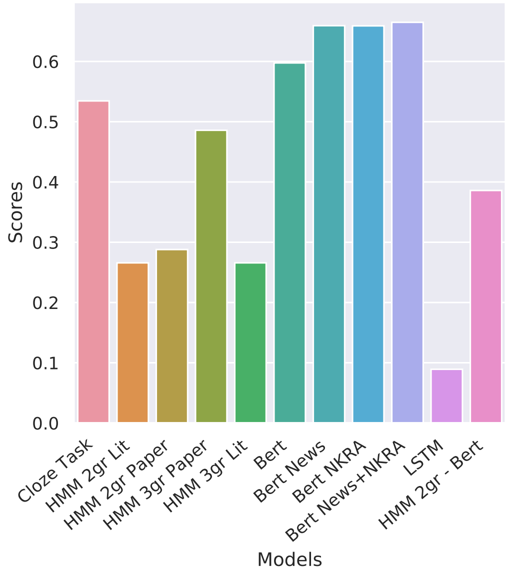
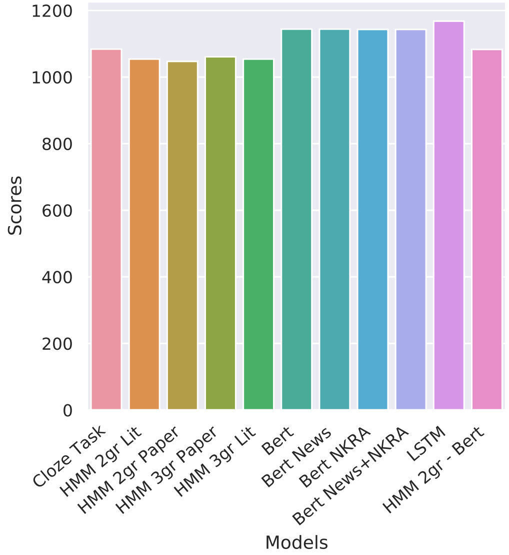


Figure 8: Absolute accuracy histogram, part-of-speech level.



In absolute values, there is also a tendency in higher accuracy of BERT models, which can be interpreted as follows. BERT as a language model correctly predicts a part of the next word, but the words themselves, rather close to the context, have a low probability. Moreover, in many cases, they have almost a uniform distribution equal to 0.033.

*Model Consistency*

The consistency of a part of speech prediction differs significantly from the lexical level. However, when compared with the cloze task model, we do not see a strong resemblance with one of the artificial models. That is, there are contexts for which a person can predict the part of speech of the next word correctly, while the models are not able to do the same.

## 4.4. Model comparison on the OMVF level

*Mean Accuracy*

Figure 9: Overlap heatmap, part-of-speech level.

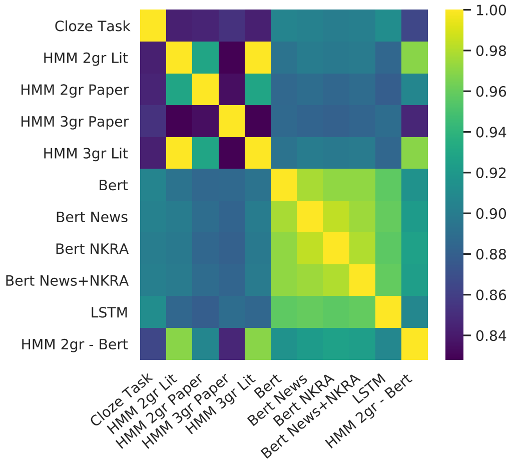
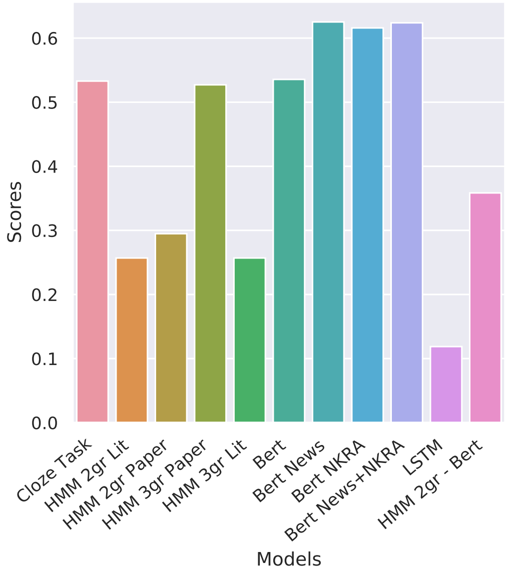


Figure 10: Mean accuracy histogram, object-verb-functional-modifier level.



Reducing the number of classes by 4 times does not lead to an improvement in the average accuracy. Although we can see that the models perform similarly as on the original part-of-speech model.

*Absolute Accuracy*

Absolute values at the OMVF level of generalization cease to reflect any properties of the models. This is due to the metric calculation algorithm. The model's indicator increases by one each time when there is a correct answer and its probability is not equal to 0. Accordingly, the graph reflects that in more than 95% of cases the correct tag is present. It can be noted that for a model with a random tag generator, this threshold would be 25%. Such differences in magnitudes suggest productivity at the OMVF level.

*Model Consistency and Kolmogorov-Smirnov Test on OMVF*

Consistency at the object-verb-functional-modifier level and Kolmogorov-Smirnov Test are is shown in Figure 4.4. and 4.4. correspondingly.

Figure 11: Absolute accuracy histogram, object-verb-functional-modifier level.

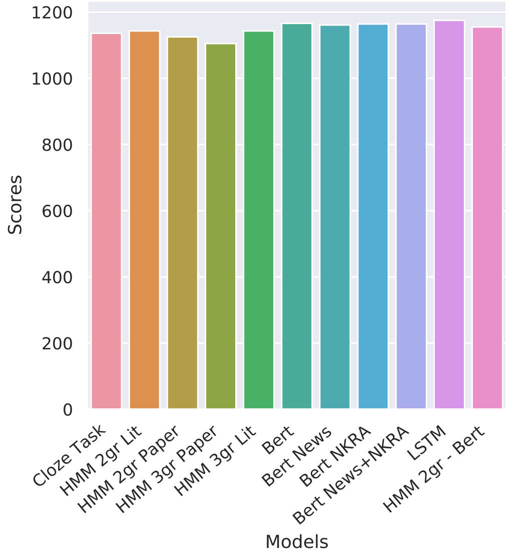


Figure 12: Overlap heatmap, object-verb-functional-modifier level.

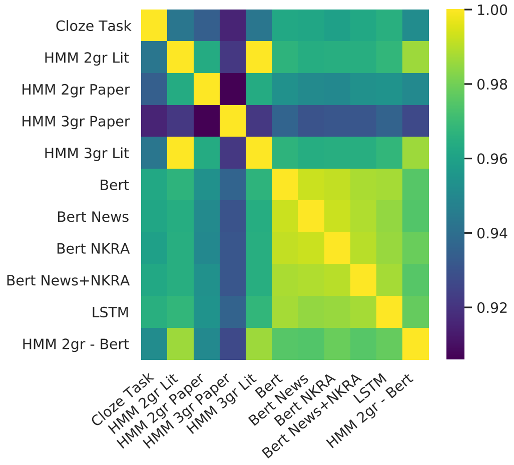
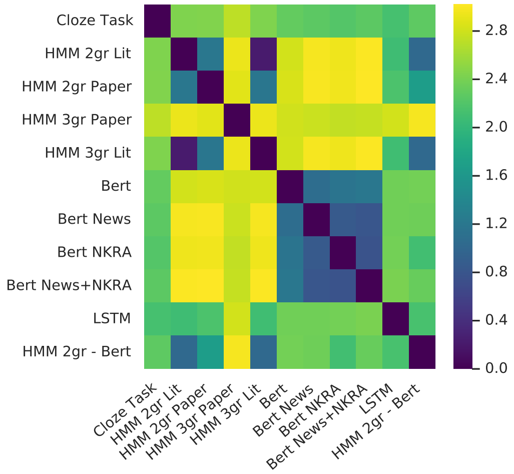


Figure 13: Kolmogorov-Smirnov test heatmap, object-verb-functional-modifier level.



The last two tests reflect a contradictory trend: LSTM shows a greater resemblance to the cloze task. Here it is necessary to comment on probabilistic distribution. Calculation of statistical tests based on probability distributions of the models makes the metrics far from objective, since the variation of the number of lexical units to consider significantly impacts the final output. Thus the results may highly differ for 25 and 30 lexical units. Moreover, artificial models allow us to reflect distributions for the large vocabulary, whereas the vocabulary of the cloze task is very limited and has many random outliers when the context is not constraining the next word on any level. It is disputable whether such cases should be eliminated from the vocabulary during research or not.

## 5. Conclusion

One of the important results of this study is the development of a certain set of methods (tools) for comparing the generative properties of language models. The starting point is an unrestricted set of contexts and the probabilistic distribution of words for each of them. This data can be obtained from all kinds of language models. Also, the re-normalized value of frequency from the corpus can be used as a language model for these purposes in further research. From our point of view, the most relevant metrics (from presented above) are Mean (Absolute) accuracy is prediction and Cosine similarity measure computed for each pair of models.

We tested this methodology on the following models: Cloze-task-based model, hidden Markov model (with the different n-grams), LSTM and BERT. From the results, we have noticed that the last one can predict the next word more accurately, while LSTM - Cloze task pair shows that semantic directions of k-first words for given context are more similar. However, based on all of these metrics scores, we can conclude that Cloze-task-based model cannot be replaced by any of the artificial language models presented in this paper for eye-tracking experiments. In addition, it is worth noticing that predictability scores computed within the Cloze task reflect the real-world situation, but this is beyond the scope of our study and could potentially be used in a different neurolinguistic experiment.

## 6. Bibliography

Al-Anzi, F. and Abu Zeina, D. (2017). Statistical markovian data modeling for natural language processing. *International Journal of Data Mining & Knowledge Management Process*, 7(1):25–35.

Apresjan, J., Boguslavsky, I., Iomdin, B., Iomdin, L., Sannikov, A., and Sizov, V. (2006). A syntactically and semantically tagged corpus of russian: State of the art and prospects 1. In *Proceedings of LREC*, pages 1378–1381.

Arefyev, N., Panchenko, A., Lukanin, A., Lesota, O., and Romanov, P. (2015). Evaluating three corpus based semantic similarity systems for russian. In *Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодно Международной конференции Диалог (Москва, 27 – 30 Мая 2015)*, volume 2, page 116–128. РГГУ Москва.

Bocharov, V., Bichineva, S., Granovsky, D., Ostapuk, N., and Stepanova, M. (2011). Quality assurance tools in the opencorpora project. In *Компьютерная* *лингвистика и интеллектуальные технологии: По материалам ежегодно Международной конференции Диалог (Бекасово, 25 – 29 Мая 2011)*, pages 101–109. РГГУ Москва.

Béjar, J. (2010). K-means vs mini batch k-means: A comparison. *Universitat Politècnica de Catalunya*.

Chen, S., Beeferman, D., and Rosenfeld, R. (1998). Evaluation metrics for language models. *Carnegie Mellon University*, pages 1–6.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep birdirectional transformers for language understanding.

Frisson, S., Harvey, D., and Staub, A. (2017). No prediction error cost in reading: Evidence from eye movements. *Journal of Memory and Language*, pages 200–214.

Fu, C., Li, Y., and Zhang, Y. (2019). Atnet: Answering cloze-style questions via intra-attention and inter-attention. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*.

Halser, E., Stahlberg, V., Tomalin, M., and et al. (2017). A comparison of neural models for word ordering. *ACL Anthology*, pages 208–212.

Hofmann, M., Biemann, C., and Remus, S. (2017). Benchmarking n-grams, topic models and recurrent neural networks by cloze completions, eegs and eye movements. *ResearchGate*, pages 1–17.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. (2016). Exploring the limits of language modeling.

Korobov, M. (2015). Morphological analyzer and generator for russian and ukrainian languages.

Kuperberg, G. and Jaeger, T. (2016). What do we mean by prediction in language comprehension? *Language, Cognition and Neuroscience*, pages 32–59.

Laurinavichyute, A., Sekerina, I., Alexeeva, S., and et al. (2018). Russian sentence corpus: Benchmark measures of eye movements in reading in russian. *Behavior Research Methods*.

Luke, S. and Christianson, K. (2016). Limits on lexical prediction during reading. *Cognitive Psychology*, pages 22–60.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Murgue, T. and Higuera, C. (2004). Distances between distributions: Comparing language models. *Structural, Syntactic, and Statistical Pattern Recognition*, pages 269–277.

Shavrina, T. and Shapovalova, O. (2017). To the methodology of corpus construction for machine learning: «taiga» syntax tree corpus and parser. In *Corpus Linguistics*.

Smith, N. and Levy, R. (2011). Cloze but no cigar: The complex relationship between cloze, corpus, and subjective probabilities in language processing. *Cognitive Science Society*, pages 1637–1642.

Staub, A., Grant, M., Astheimer, L., and A., C. (2015). The influence of cloze probability and item constraint on cloze task response time. *Journal of Memory and Language*, pages 1–17.

Taylor, W. (1953). Cloze procedure: A new tool for measuring readability. *Journalism Quarterly*, pages 415–433.

Wand, Shuohang, , and Jiang, J. (2018). An lstm model for cloze-style machine comprehension. In *Computational Linguistics and Intelligent Text Processing: 19th International Conference, CICLing*.

Wang, A. and Cho, K. (2019). Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2018). Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*.

Wikipedia. (2019). Russian wikipedia.

Zhou, J., X., J., and Yang, J. B. (2018). Cloze answer generator. *CS 224N Project*.