mathematics

MDPI

Article

Virtual Dialogue Assistant for Remote Exams

Anton Matveev *,†, Olesia Makhnytkina †, Yuri Matveev, Aleksei Svischev, Polina Korobova, Alexandr Rybin and Artem Akulov

Information Technologies and Programming Faculty, ITMO University, 197101 Saint Petersburg, Russia; makhnytkina@itmo.ru (O.M.); yunmatveev@itmo.ru (Y.M.); svischev@itmo.ru (A.S.); pikorobova@itmo.ru (P.K.); arybin@itmo.ru (A.R.); 287550@niuitmo.ru (A.A.)

* Correspondence: aymatveev@itmo.ru

† These authors contributed equally to this work.

Abstract: A Virtual Dialogue Assistant (VDA) is an automated system intended to provide support for conducting tests and examinations in the context of distant education platforms. Online Distance Learning (ODL) has proven to be a critical part of education systems across the world, particularly during the COVID-19 pandemic. While the core components of ODL are sufficiently researched and developed to become mainstream, there is still a demand for various aspects of traditional classroom learning to be implemented or improved to match the expectations for modern ODL systems. In this work, we take a look at the evaluation of students' performance. Various forms of testing are often present in ODL systems; however, modern Natural Language Processing (NLP) techniques provide new opportunities to improve this aspect of ODL. In this paper, we present an overview of VDA intended for integration with online education platforms to enhance the process of evaluation of students' performance. We propose an architecture of such a system, review challenges and solutions for building it, and present examples of solutions for several NLP problems and ways to integrate them into the system. The principal challenge for ODL is accessibility; therefore, proposing an enhancement for ODL systems, we formulate the problem from the point of view of a user interacting with it. In conclusion, we affirm that relying on the advancements in NLP and Machine Learning, the approach we suggest can provide an enhanced experience of evaluation of students' performance for modern ODL platforms.

Keywords: virtual dialogue assistant; natural language processing; machine learning; online distance learning

Citation: Matveev, A.; Makhnytkina, O.; Matveev, Y.; Svischev, A.; Korobova, P.; Rybin, A.; Akulov, A. Virtual Dialogue Assistant for Remote Exams. *Mathematics* **2021**, *9*, 2229. https://doi.org/10.3390/math9182229

Academic Editor: Grigoreta-Sofia Cojocar

Received: 31 July 2021
Accepted: 6 September 2021
Published: 10 September 2021

1. Introduction

In this paper, we present a case for a Virtual Dialogue Assistant (VDA): an automated system intended to provide support for conducting tests and examinations in the context of distant education platforms. A *virtual dialogue assistant*, in this context and to the extent of our understanding, is not a common term; therefore, our approach for this presentation is first to introduce our model and its properties, define a case for it, and hypothesize about its utility. Then, we review our model in the context of the field of Natural Language Processing to draw similarities and distinctions with established problems researchers are working on, and suggest the role our model might play in the field. Finally, we explain our approach to various important problems we have to solve in order to, ultimately, assemble the model, and review our achievements and results at this point of our research.

The idea of distance education first appeared in the 18th century, and by the 20th century, there were schools focusing on it primarily [1]. The development of the World Wide Web in the 1990s revolutionized distance education, adapting it for the new reality of emerging post-bureaucratic organizations and globalization [2]. Since the beginning of the coronavirus pandemic in late 2019, according to the UNESCO Institute for Statistics data reports, more than 1.4 billion children in 186 countries were affected by school closures [3].

Worldwide, universities, especially the ones relying on revenue from international students, are facing major financial problems and are forced to either shut down or start offering their courses online [4]. However, it is not only educational institutions that have been online. According to reports, approximately 4% of corporations were using online learning in 1995, and presently, this number is at least 80% [5]. Various reports relying on data from major online education platforms, such as Coursera and EdX, present an evaluation of a compound annual growth rate for the digital education market at around 33% [6].

Our research targets some of the problems in the field of distance education, since we believe the data demonstrates that this field is rapidly growing and not only provides pure economic value, but also brings people convenience and the joy of learning, ultimately expanding opportunities for people to discover an interest in science and start performing their own scientific research, ultimately benefiting the community.

A significant impact of assessments on learning behavior was observed more than four decades ago, and even concepts like "assessment drives learning" were introduced [7]. Various forms of assessments were meticulously studied in the past few decades, and, it seems, there is no definite consensus on the exact relationship between forms of assessments and the quality of learning. Some researchers present cases that demonstrate a negative impact of assessment-driven education [8] and some point out that there is still much to be learned about the complex relationship between student learning and assessment [9].

However, even if someone has already formed an opinion on the role of assessments in the education process, online distance learning brings forward new circumstances that ought to be considered. Since distance learning brings an environment different from a traditional classroom, it is critical not to forget the essential roles of assessment: providing feedback to learners to remind them of what exactly they are doing and what they yet need to do to complete a course; offering a tool for evaluation of their progress; accumulating performance data to evaluate a course itself, if learners are performing poorly, there might be an issue with the course, not with the learners. It appears that overall, researchers seem to agree that various forms of assessment are necessary for distance learning, and new tools for assessment need to be developed, tested, and evaluated to support the modern learning environment [10,11].

It is also interesting to note that while the learning materials for a course might be universal for all students, the testing kit, in the ideal scenario, is different and adjusted for each student individually. In a one-to-one interview, a teacher is often equipped with past knowledge about a student and can manage the flow of the interview with the initial questions and the student's responses to previous questions. With distant examinations, there are several techniques, none of which seem to achieve the goal without significant drawbacks; for example, it is possible to prepare a large enough number of questions and present students with distinct subsets of questions that, however, does not mean the questions are adapted for each student, only that they are different; it is also possible to adjust selections of questions for a student with information about their intermediate indicators, if any, but even then, the possibility to tailor the following question based on the previous answers during the test is missing. It appears that no single solution is perfect; therefore, a compound approach is recommended [12].

Understanding that there is a demand for new tools for assessment in distance learning, in our research, we focus on investigating what features and qualities of such tools might be desirable and suggest an approach for designing, developing, and evaluating them. Specifically, our primary interest lies in the exponentially growing field of machine learning.

The field of Natural Language Processing (NLP) belongs to the intersection of linguistics and computer science. Beginning in the 1950s, the foundation of modern NLP was established by the end of the 20th century due to the significant increase in the amount of raw data and the development of traditional machine learning techniques. Over the past decade, modern machine learning techniques found their way into NLP. For our research, we do not limit ourselves to particular methods or algorithms, but instead borrow from the

rich history of the development of the field, noticing the advantages and disadvantages of various approaches and techniques.

The development of NLP begins with symbolic algorithms [13]: methods that use rules and grammars to parse and model syntactic structures to imitate how humans apply rules and grammar to sentence recognition. With the development of machine learning techniques, they found their way into NLP; the emergence of significantly large textual corpora allowed for the application of statistical methods to extract information, discover patterns, and predict missing data, that is, to build probabilistic models [13]. It was noted that, in general, symbolic techniques offer more compact and high-level representations than statistical models, but lack in the level of robustness and geerality [14]. In the last decade, the development of deep learning models relying on increasing computational resources (specifically, Graphical Processing Units) and parallelization, allowing for building artificial neural networks with billions of trainable parameters, transformed many fields of study and, along with even further increasing amounts of available raw and annotated data, made it possible to solve many complex problems using *brute force*; naturally, deep learning techniques are now widely applied to NLP problems as well [15]. It is not always the case that to solve a problem, it is necessary to employ the most powerful tool; often, a weaker tool might be more efficient. If a tool provides a satisfying outcome for a task but requires less computational resources or research and development time, it might be reasonable to utilize it over a more powerful tool. With that in mind, in our work, we attempt to avoid excessively narrowing down the list of techniques to consider and focus on exploring a wider range of methods, since our primary goal is not to achieve the best outcome for each specific problem but, instead, to demonstrate the potential power of a complex system that utilizes sufficiently accurate solutions to the specific problems and, by achieving synergy between them, produces a higher-level outcome.

Broadly speaking, NLP techniques might be categorized into three groups. Initially, the idea was to operate on manually constructed knowledge bases and hand-coded symbolic grammars or rules [16]. In the 1950s, researchers working on developing fully automatic systems for machine translation—one of the core problems in NLP—were highly optimistic about advances in formal linguistics and computational power and were expecting the problem to be effectively solved in less than a decade. Soon, however, the realization that the problem might be more complex than initially estimated occurred, with prominent researchers even expressing an opinion that the problem might be unsolvable in principle [17]. It appears that a rule-based approach might be appropriate and efficient for narrow situations but requires an exceeding amount of manual input for more general problems. In this work, we take advantage of this property of rule-based methods and apply them when we work with a problem, for which it is more important to demonstrate a proof-of-concept rather than the best possible accuracy.

Another category of NLP techniques is empirical for statistical methods. Compared to rule-based methods, the main difference is that the manual input is replaced with a combination of training data and a learning system that operates on the training data to produce a knowledge base. A knowledge base is a general term for various data structures that might be appropriate for particular problems [16], for example, ontologies. Some authors note that ontologies, even as a specific kind of knowledge base, might be too broad of a concept to specify their role explicitly; however, it is relatively clear that in virtually any case, they provide, at least, access to the meaning of terms, concepts, or relations [18]. In this work, we rely on the idea of ontologies for providing us with a tool for expressing the goal, the problem, and the reference solution. We expand on that further.

The state-of-the-art category of NLP techniques results from reaching the critical point in terms of computational resources and volumes of raw and annotated data in the past decade. For example, deep learning models approach hundreds of billions of parameters and terabytes of the total model size in a modern way [19]. Deep learning models already demonstrated exceptional performance for computer vision [20] and speech recognition problems, and more fascinating results appear to be on the way. The most common architec-

tures for deep learning models include recurrent neural networks (RNNs), convolutional neural networks (CNNs), recursive neural networks, and generative adversarial networks (GANs) [21]. Across various deep learning models, several concepts are proven to be core for most NLP problems.

The first is feature representations or embeddings. The aim of feature embedding is to encode text input into a form that focuses on highlighting particular characteristics of the input that are important to a specific problem. Depending on the input and the type of problem, embedding can be performed at the level of single characters, words, phrases, and even paragraphs.

The second concept is sequence-to-sequence modeling. In many problems related to human intelligence, data points are not independently and identically distributed, but instead depend on the preceding or following data points, and language is a straightforward example of sequential information. Moreover, for many NLP problems, not only is the input often sequential, but the output is also expected to have a similar nature, for example, in machine translation. One common approach to sequence-to-sequence modeling is to employ an encoder that consumes input and produces an intermediate output, and a decoder that transforms that output into the desired form.

Another important concept is reinforcement learning. There are several obstacles to sequence-to-sequence modeling, such as exposure bias and inconsistency between the train/test measurement; one approach to handle these issues is to apply reinforcement learning techniques and enhance a model to only rely on its own output, rather than the ground truth, and to directly optimize the model using the evaluation measure [22]. Training deep learning models for NLP problems requires significant computational and storage resources and is often performed by distributed networks of GPU servers [23]. For the purposes of our research, we do not aim to modify or reproduce any particular deep learning model; while it might be helpful to develop and train deep learning models to precisely fit to the requirements of our tasks, we believe, in our particular case, it is neither efficient nor necessary.

Among the three approaches, deep learning appears to have one clear advantage, and that is virtually unlimited scalability. Even a decade ago, many problems were deemed unapproachable by computers, for example, the game of Go, but recently the game was, essentially, solved with deep learning [24]. It appears all of the modern NLP problems can be, eventually, solved with deep learning [21]. However, at this moment, we believe it is not practical to simply delegate any problem to deep learning algorithms: the field of artificial intelligence is still relatively new, and most of the significant practical results were only produced in recent years. In our research, instead, we consider all three approaches and, if anything can be recognized as the key idea, it is the idea of ontologies. While deep learning solutions are powerful tools for achieving results, they are still mostly black boxes that provide little insight into the problems; on the other hand, while the idea of ontologies in itself is rather abstract, it provides an insight into how a solution to a problem might be constructed.

Originating from philosophy, the concept of ontology was adopted by computer science, particularly Artificial Intelligence and Computational Linguistics, either to refer to general ideas of conceptual analysis or domain modeling, or to describe a distinct methodology or architecture of a specific solution for data collection (mining), organization, and access [25]. Guizzardi [26] defends the notion that "the opposite of ontology is not non-ontology, but just bad ontology". This notion is vital to us, since it allows us to employ the concept of ontology as a basis for our reasoning about the fundamental nature of the problem we consider and solutions to it; in other words, we believe it is sufficient to demonstrate that if there exists a solution that shows acceptable performance with respect to an incomplete or imperfect ontology, then this solution has to be considered a candidate for the ultimate solution to the problem, given our transcendent goal of building "good" ontologies.

One example of this ontology-based approach is the Wolfram System. The web page for the Wolfram Data Framework says [27]:

> *As by far the largest computable knowledge system ever built, Wolfram Alpha has been in a unique position to construct and test a broad ontology.*

For example, met with a query "how many goats in Spain", the system produces [28] an interpretation of the query, a graph with livestock population historical data, and, ultimately, the result (3.09 million as of 2016). While the underlying knowledge base (ontology) of the system is not yet "perfect" and is constantly updated via data mining, it presents a valuable example of the capability and practicality of the ontology-based approach. This example is also important to us, since it is closely related to the fundamental problems we consider.

A virtual dialogue assistant for remote exams has at least two core features: question generation, and evaluation of students' answers. In the literature, those problems are approached in various ways.

There are methodological recommendations for composing exams and tests, including suggestions for managing the performance evaluation with automated systems. It is recommended to include an assortment of question types in an increasing difficulty progression:

1. Open question
   (a) Fill-in-the-blank question
   (b) Subjective question
2. Objective Test Question (OTQ)
   (a) True/False questions (statements that can be either true or false)
   (b) Closed or multiple-choice question (MCQ)
   (c) Sequencing questions (require sorting a set of items by some principle)
   (d) Matching questions (require interconnection of corresponding elements in two given sets)

Fill-in-the-blank questions are, generally, more straightforward to generate automatically than wh-questions. Fill-in-the-blank questions can be categorized as either a closed type [29,30] when there are candidates present, or an open type when they are not [31].

One obstacle for closed-type fill-in-the-blank question generation is the selection of reasonable but wrong potential answers.

At least in the past decade, there were multiple attempts to approach the closed question generation problem, resulting in a commonly acknowledged solution [32]. Modern research in this field primarily focuses on improving specific steps of the algorithm. The algorithm takes a text fragment as an input: usually, a segment of learning materials or specialized text. Some implementations of the algorithm work with formalized text structures, but, in general, the algorithm is supposed to work with arbitrary texts.

The first step of the algorithms is preprocessing of the input to obtain features that will be used at the next step. Two sources of input are considered here: (1) text in a natural language and (2) a structured representation of knowledge in a field with preset categories and relationships, for example, ontologies [30]. For texts in a natural language, the most common embeddings are statistical features (tf-idf), semantic embeddings, syntactic parsing, and POS-tagging.

The second step is the selection of a sentence that would be a source for a generated question. There are algorithms that can use a pattern to search for such sentences. A pattern might be a specific sequence of parts of speech. Another approach is machine learning, for example, summarizing [33]. The third step is key selection. The most common techniques are selection by word frequencies, pattern search, and standard machine learning models. The fourth step is question generation. The options are: to rephrase the sentence, to construct a question by applying a pattern, or to keep the original sentence but remove the key word. The fifth step is the selection of possible answer alternatives. The choice might be based on an ontology, synonyms, or nearest neighbors in the semantic space. The

final step is post-processing: additional operations required to fit the generated question to the requirements.

Modern implementations are primarily based on heuristics and patterns, which make them more stable and easier to interpret. On the downside, the large number of parameters makes it considerably challenging to set up. Using patterns for question generation and key word selection, on the other hand, limits the range of possible questions that might be generated.

For open-question generation—questions without a predetermined set of possible answers—the most common approaches are: (1) the open-closed question generation, which is a type of fill-in-the-blank question; and (2) generation of a question employing auxiliary phrases, such as "What is . . . ", "What constitutes . . . ", and so forth.

Answers to objective questions can be clearly interpreted and objectively evaluated as either correct or incorrect [34].

However, the development of neural network solutions based on transformers and the emergence of GPT-3 and BERT language models for the Russian language and mT5 multi-lingual model demonstrates the possibility of improvement for question-and-answer systems. One of the remaining issues is the processing of text segments with mathematical symbols. There are few papers on the topic of Mathematical Language Processing (MLP), and the research on the question generation for such text segments in the Russian language is yet to emerge. It appears that learning materials, particularly in STEM, rather often contain mathematical symbols and equations, meaning the solution to the problem would be desirable.

Answer evaluation is a critical part of the remote examination. The most complicated form of evaluation is the evaluation of answers to open questions. In a classroom, the examiner assesses how close the answer is to the correct answer, and one important criterion is how close the keywords are from the answer to the keywords of the correct answer. By utilizing vector representations of words in a vector space with semantic differences between words as a factor in the distance measure, it is possible to construct a system that can handle the diversity of a natural language and variability in phrasing. In this work, we are exploring algorithms for comparing dependency grammars employing vector representations of words.

In the context of NLP, relevance is a criterion used to evaluate the performance of a search engine; in other words, how close the output of the system is to what was prompted. This concept can be applied to question-and-answer and remote examination systems. For the document ranking problem, it is sufficient to have only a relevance measure; however, when there is only one document (the student's answer), it is necessary to establish a threshold to categorize the answer as either relevant or not relevant to the correct answer. For comparing a student's answer and the correct answer, we experiment with dependency grammars [35].

One relatively simple way to compare dependency grammars [36] is to count intersections of sub-trees representing semantic relationships; then the similarity between two text segments can be calculated as

$$E = \frac{|Q \cap T|}{|Q|}, \tag{1}$$

where E is similarity, Q is the set of semantic relationship tuples for the correct answer, and T is the set of semantic relationship tuples for the student's answer.

A more complicated approach is predicate matching [37]. In this case, it is not only the pairs of nodes, but all semantic relationships (so-called predicate relationships) at the verb that are considered.

Another method is fuzzy depth-first search [38]. This method also relies on dependency trees and can be summarized as traversing the trees simultaneously and adding or removing penalties depending on the item type and value.

Mathematical Language Processing is a field on the intersection of computer linguistic, formal languages theory, and, recently, machine learning and artificial intelligence. Mathe-

matical language is defined as any expression with digits, variables, or operators. Unlike natural language, mathematical language is consciously designed, since it was created by people. This field shares similar goals with NLP, and the main topics are:

1. Search for similar expressions;
2. Extraction of structural information, for example, variable names;
3. Transformation of expressions between different forms, for example, LaTeX and MathML.

Unlike NLP, MLP is at the early stages as a separate field of study. While machine learning techniques are already established for many NLP problems, the application of them to MLP is still not widely researched [39]. Some authors note that direct application of the NLP machine learning techniques does not produce results of similar quality [40]. The existing methods for feature extraction can be adopted for mathematical expression by applying them to the level of symbols; in other words, by considering an expression as a text segment and a mathematical symbol as a word (token) in terms of NLP. In this way, techniques for vector representation, distributed semantics, and universal language models can all be applied to MLP.

There are examples of adopting the TF-IDF method to MLP [39]. The research was conducted with classification and clustering problems for texts with mathematical expressions. Two datasets were used for training and testing: SigMathLing arXMLiv-08-2018 and NTCIR-11/12 MathIR arXiv. The advantages of this approach are:

1. Simple implementation;
2. Fixed length feature vector;
3. Relatively small text corpus for training.

The disadvantages are:

1. The context of an mathematical expression is not considered;
2. Insufficient representation.

In our approach to the problem, specifically, our focus on online education platforms, there is another interesting aspect to consider. Among various ODL platforms, there is a typical structure to the organization of learning materials or courses. Here, we are interested in one specific trait: a comment section. While details may vary between platforms, we generalize a model of an ODL platform and suppose that a course is split into sections in some sequential order, each containing a segment of text representing learning materials for that section of the course. We assume the students subscribing to a course have access to comment sections specific to the sections of the course. Here, we are interested in one specific feature of such a model: the emotional reaction of the students represented by the comments they submit. We do not assume whether a positive or a negative response to a segment of learning materials correlates to the quality of the materials; we are simply interested in investigating whether or not the presence of a reaction, followed by an adjustment to the question generation part of the system, may affect the performance evaluation of students. The methods and techniques for emotion detection that we investigated in one of our papers are related to this research [41].

One related topic that, however, that is beyond the scope of this paper is audio language processing. Notably, we want to mention on-device speech recognition [42] for the following reason: while traditionally, speech recognition is a resource-intensive task that virtually required a client-server implementation similar to Apple's Siri, modern techniques allow for on-device implementations, which not as much implies technical details, but presents a different view on architecture of solutions employing speech recognition as part of a pipeline; therefore, it allows us not to consider it as a task requiring specific accommodation, but simply as an ad hoc utility.

2. Methods

2.1. Semantic Relationships Extraction

Extraction of semantic relationships from unstructured text is essential for building knowledge bases, thesauruses, ontologies, and information search. Semantic relationships,

also called paradigmatic semantic relationships or, less often, lexical relationships, are relationships between lexical entities (words, phrases) within a restricted topic. Semantic relationships specifically describe connections and differences between words and phrases. There are several common types of relationships: synonyms, type-of relationships (hyponyms and hypernyms), part-of relationships (meronymy and holonymy), antonymy, and converses (relational antonyms).

Most often, ontologies, regardless of the language, are comprised of several entities: instances (concrete, such as people, buildings, etc., or abstract, such as numbers, feelings, etc.), concepts (groups or classes of objects), attributes (instance characteristics that have a name and a value for storing information), relationships and functions (describe dependencies between entities in the ontology), and axioms which represent restrictions.

For this research, we employ a dataset in the Russian language and experiment with extracting semantic relationships via rule-based and machine learning-based methods from it.

We picked the dataset RuSERRC described in Bruches et al. [43]. This dataset is a collection of abstracts to scientific papers in information technologies with 1600 annotated documents and 80 manually annotated with semantic relationships: CAUSE (X yields Y), COMPARE (X is compared to Y), ISA (X is a Y), PARTOF (X is a part of Y), SYNONYMS (X is a synonym of Y), USAGE (X is used for Y). For our purposes, we select the ISA, PARTOF, SYNONYMS, and USAGE relationships; however, we also enhance it with HYPON (hyponymy and hypernymy) and TERM (a term and its definition) semantic relationships. Examples of these records are shown in Table 1.

The total number of records for semantic relationships is 599: 99 for ISA in the original dataset and 14 added by us, 90 for PARTOF and 8 added, 33 for SYNONYMS and 13 added, 311 USAGE and 14 added; HYPON and TERM relationships were not present in the original dataset, and we added 15 and 51 of them, respectively.

Table 1. Examples of semantic relationship records.

<e1>Естественный язык</e1>(ЕЯ) - <e2>язык, используемый для общения людей и не созданный целенаправленно</e2>. *<e1>A natural language</e1>(NL) is <e2>a language which is used for people to communicate and which is not consciously designed</e2>.*	TERM
Примерами <e1>естественных языков</e1>являются <e2>русский</e2>, <e2>английский</e2>, <e2>китайский</e2>, <e2>казахский</e2>др. *Some examples of <e1>natural languages</e1>are <e2>Russian</e2>, <e2>English</e2>, <e2>Chinese</e2>, <e2>Kazakh</e2>, etc.*	ISA
К <e1>формальным языкам</e1>относят <e2>язык математической логики</e2>; <e2>языки программирования</e2>; <e2>языки, порожденные регулярными выражениями</e2>, <e2>конечными автоматики</e2>, <e2>грамматикой Хомского</e2>и др. *Some examples of <e1>formal languages</e1>are <e2>the language of mathematical logic</e2>; <e2>programming languages</e2>; <e2>languages emergent from regular expressions</e2>, <e2>closed-loop automations</e2>, <e2>Chomsky grammars</e2>, etc.*	HYPON
<e1>Языки, созданные целенаправленно</e1>, называют <e2>искуственными языками</e2>. *<e1>Consciously designed languages</e1> are called <e2>artificial languages</e2>.*	TERM

Table 1. *Cont.*

</e1>Лексический анализ</e1>(<e2>токенизация</e2>) - выделение в тексте слов, цифровых комплексов, знаков препинания, формул, и т.д. *</e1>Lexical analysis</e1>(<e2>tokenization</e2>) is a process of selecting words, digits, punctuation marks, equation, etc. from a text.*	SYNONYMS
При аналитическом выражении грамматических значений <e1>слова</e1>типично состоят из <e2>малого числа морфем</e2>, при синтетическом - из нескольких. *From the point of view of analytical expression of grammatical meaning, <e1>words</e1>typically consist of a <e2>small number of morphemes</e2>, but from a greater number of morphemes from a synthetic point of view.*	PART_OF
<e2>Разработка информационной системы</e2>по клещевой опасности на основе <e1>отнологии</e1>предметной области Предложен подход к разработке состава и структуры Интернет - ресурса на основе онтологии предметной области. *<e2>For development of an information system</e2>for acari ticks danger prevention based on <e1>ontologies</e1>, an approach to development and organization of a web-portal was suggested.*	USAGE

2.1.1. Rule-Based Approach

The sentiment extraction begins with preprocessing of source texts, their annotation, and a compilation of rule sets; each rule is then applied to each sentence, and, finally, the extracted semantic relationships are evaluated. When the recall for a rule is below 0.7, the rule is removed from the set. Via this process, for semantic relationships extraction, 84 rules were selected (see Table 2). Since the rules are fine-tuned to the dataset if new texts are added to the collection, the rules have to be reevaluated.

Table 2. Examples of the selected rules (for demonstration purposes, verbs are translated from Russian into English, while the original structure and punctuation are kept intact).

	Examples of Rules	
Terms	$TERM—is $*	$* contraposed $TERM
	$TERM, $*,—is $*	$TERM performs $*
	$TERM,—is $*	$TERM is defined $*
	$TERM—is $*	$TERM marks $*
	$TERM—$*	$* $TERM means $*
	$TERM—$*	$TERM means $*
	$* $TERM—$*	$TERM is $*
	$* ($TERM—$*	$TERM , $*, is $*
	$TERM is called $*	$TERM is expressed by $*
	$* called $TERM	$TERM states $*
	$TERM means $*	$TERM (is performed by) $*
	$TERM solves $*	

One necessary step for information extraction is preprocessing. Here, we execute a typical pipeline: stop-word removal, tokenization, lemmatization, and, optionally, correction.

For performance evaluation, we calculate the F-score (Table 3).

Table 3. Performance of the rule-based approach.

	PART_OF	ISA	USAGE	HYPON	SYNONYMS	TERM
F-score	0.31	0.46	0.52	0.41	0.35	0.65
Precision	0.37	0.31	0.50	0.28	0.22	0.48
Recall	0.27	0.94	0.54	0.70	0.80	0.94

2.1.2. Machine Learning-Based Approach

In this work, we experimented with a neural network architecture from Bruches et al. [43]. This neural network has four layers. The first layer processes features from the pre-trained model Ru-Bert, where this model for the Russian language was trained on texts from the Russian segment of Wikipedia and news articles from the website, Lenta.ru. The volume of the pre-processed dataset is close to 6.5 gigabytes, 80% of it from Wikipedia. The second and the third steps apply a mask obtained from data annotations in such a way as to classify tokens into semantic relationship categories. The final layer outputs semantic relationship labels, and, finally, the data are passed through softmax. The training was performed with 10 and 20 epochs with the Adam optimizer and a starting learning rate of 0.0001.

The performance of the model with 20 epochs and a 0.0001 learning rate for each type of semantic relationship is shown in Table 4.

Table 4. Performance of the machine learning-based approach for each type of semantic relationship.

	PART_OF	ISA	USAGE	HYPON	SYNONYMS	TERM
F-score	0.78	0.95	1.00	0.86	0.86	0.29
Precision	0.78	1.00	1.00	0.81	0.81	0.33
Recall	0.78	0.90	1.00	0.91	0.91	0.25

2.1.3. Example of Question Generation

The semantic relationships extracted via both of the described methods can then be used to generate questions (Table 5).

Table 5. Examples of questions generated via extracted semantic relationships.

Question Pattern	Question in Natural Language
What is $TERM?	Что такое естественный язык?
Which kinds of $TERM there are?	Какие есть виды языков?
What are some examples of $TERM?	Какие есть примеры эльфийских языков?
Which parts $TERM consists of?	Что является частями анализа языка?
What is $TERM a part of?	Частью чего является Синтез языка?

For alternative (incorrect) answers, we used parts of other records of the same semantic relationship type. Here is a full example of a test constructed via question generation based on a text segment:

> Естественный язык (ЕЯ)—язык, используемый для общения людей и не созданный целенаправленно. Примерами естественных языков являются русский, английский, китайский, казахский и др. Языки, созданные целенаправленно, называют искусственными языками. На данный момент их уже больше 1000, и постоянно создаются новые. Обработка естественного языка (Natural Language Processing, NLP)—общее направление искуственного интеллекта и математической лингвистики. Оно изучает проблемы компютерного анализа и синтеза естественных языков.
>
> Вопрос к отрывку №1:
>
> Что такое естественный язык?

1. Язык, используемый для общения людей и не созданный целенаправленно. 2. какое-любо конкретное значение, которое может принимать данный признак (ключ). 3. лингвистическеие процессоры, которые друг за другом обработывают входной текст.

Что такое искуственный язык?

1. язык, созданный целенаправленно, 2. удаление значетельной части "морфолонгического шума"и омонимичности словоформ 3. явление, при котором синтаксические конструкции имеют близкие значения и способны в определенных контекстах заменять друг друга.

Natural language (NL) is a language used to communicate between people and is not purposefully created. Examples of natural languages are Russian, English, Chinese, Kazakh, and so forth. Languages created purposefully are called artificial languages. At the moment, there are already more than 1000 of them, and new ones are constantly being created. Natural Language Processing (NLP) is a general area of artificial intelligence and mathematical linguistics. It studies the problems of computer analysis and synthesis of natural languages.
Question for excerpt 1:
What is natural language?
1. The language used to communicate between people and not purposefully created;
2. Any specific value that a given attribute (key) can take;
3. Linguistic processors, which one after another process the input text.
What is artificial language?
1. Language created on purpose;
2. Removal of a significant part of the "morpholongic noise" and homonymy word forms;
3. A phenomenon in which syntactic constructions have close meanings and are capable of replacing each other in certain contexts.

2.2. Deep Learning-Based Question Generation

For this problem, we experiment with neural network architectures based on transformers for three tasks: key selection, question generation, and selection between possible answers. The workflow is the following: a question is generated based on a text fragment, where this question and the text fragment are then used to generate an answer, and finally, the text fragment, the question, and the answer are used to generate alternative answers. We choose the pre-trained models ruGPT and mT5, and we pick the smallest variations (2 and 4 gigabytes, respectively) since we are primarily interested in proving the concept rather than achieving the highest possible precision. mT5 is a model often employed for seq2seq problems, and it contains an encoder that translates the input text into a latent vector space, and a decoder that takes the output of the encoder and its own output from the previous pass to produce a new output. ru-GPT-3 is a language model that only employs the decoder part of the original transformer architecture. Since the question generation from a context and the generation of an answer from a context and a question are sqe2sqe problems, we expect mT5 to produce more accurate output; however, we notice that question generation with ruGPT-3 produces a more stable and coherent result. For fine-tuning, we selected the RuBQ (3000 records) and SberQuAD (50,000 records) datasets, and for evaluation, we calculate perplexity (Table 6).

It is interesting to notice that an increase in batch size improves the performance. Since ruGPT-3 appears to be a better fit for our problem, next, we adapt it via zero-shot training: feeding a generative model with some patterns for it to fulfill it with actual data.

Table 6. Performance of the mT5 and ruGPT-3 models.

Model	Problem	Perplexity (Training Set)	Perplexity (Test Set)	Batch Size	Iterations	Dataset
ruGPT-3	Question	1049	2657	4	2959	RuBQ
ruGPT-3	Answer	1356	316	2	5917	RuBQ
mT5	Question	1357	1176	1	11,847	RuBQ
ruGPT-3	Question	157,259	102,679	1	85,328	SberQuAD
ruGPT-3	Question	6113	-	2	2096	SberQuAD
ruGPT-3	Question	2491	-	4	1048	SberQuAD

We experiment with the *Large* version of the model with 760 million parameters trained by a team from Sberbank with the supercomputer Christofari. Over 600 gigabytes of data (mostly in the Russian language) was used for training the model: literary fiction, Russian and English segments of Wikipedia, news web-portals, public categories of Pikabu.ru, a collection of articles for a popular science web-portal 22century.ru, materials from banki.ru, Omnia Russica text corpus, snapshots of GitHub and StackOverflow. The ratio between data in Russian and English used by the authors for training the model is around 9 to 1. The model utilizes 2048 tokens of a context window and full attention blocks. Currently, this is the most powerful model for the Russian language. The architecture of the model is a transformer decoder with 24 blocks, 16 heads each. The model is trained by predicting the next word in a sequence. Since we are working with a seq2seq problem, to generate output (a question), we need to feed the model with a proper prompt. Here is an example of a prompt and a response of the model:

(rationale)

[question word, key word]

-question

-answer

(rationale)

[question word, key word] (затравка) (NLTK – специализированная среда для автоматической обработки текстов, созданная для работы с Python)

[Что, NLTK]

-Что такое NLTK?

-Специализированная среда для автоматической обработки текстов, созданная для работы с Python. (В состав NLTK входит коллекция корпусов и словарные базы данных.)

[Что, состав]

(generation)

-Что включает в себя NLTK?

-NLTK включает в себя: -PyLib (Публичная библиотека Python, содержащая библиотеку, написанную на языке Python для обработки PHP ... (конец генерации, предел длины последовательности)

(rationale)
[questionword, keyword]
-question
-answer
(rationale)
[questionword, keyword]
(seed) (NLTK is a specialized environment for automated text processing with Python)
[What, NLTK]
-What is NLTK?

-a specialized environment for automated text processing with Python.
(NLTK contains collections of text corpora and dictionaries.)
$[What, contains]$
(***generation***)
-What are the parts of NLTK?
-NLTK contains PyLib (a public Python library that contains a module for PHP processing . . . (finish generation, exceeded length limit)

2.3. Answer Evaluation

One common simple approach to the problem of answer evaluation relies on comparing word or lemmas symbol-by-symbol. We expand on this idea and attempt to search for semantically similar segments. This approach enhances the fuzzy string search, similar to how predicate comparison enhances the method for enumerating intersections of sets of semantic relationships.

Here is an explanation of the steps required to compare the answer given by a student to the correct answer. We assume that text fragments are represented by dependency trees; similar to depth-first search, both dependency trees are traversed: missing an edge results in a penalty, and the cost is an adjustable value based on several parameters, such as the length of text fragments and the complexity of the dependency trees; also, the cost may differ for the student's answer and the correct answer.

At each step, the nodes are compared by calculating the dot product between vector embeddings for the given words and scaling with an additional adjustable parameter; for example, the distance between the node and the root of the tree may imply a relative significance of a particular sub-tree.

Between all paths from the root to a node, the one with the highest cumulative similarity is selected. Those values can then be used to fine-tune the threshold for making a decision.

There are various ways to construct dependency trees; for our purposes, we chose to work with objects that hold attributes and pointers to other objects, such as word embeddings and links from parent nodes to children nodes. Additional attributes may include, for example, tags for parts of speech.

For experiments, we presented several teachers with a text fragment from learning materials about linguistics and asked them to formulate questions, and the teachers came up with a total of 26 questions. Then, we asked four students to answer the questions, and finally, we asked the teachers to arrange the answers from the most relevant to the least relevant (adding the correct answer forefront). During testing, we investigated the influence of the following parameters and the performance of the algorithm:

1. The scaling parameter for the distance between nodes;
2. The penalty for missing an edge in the dependency tree of the correct answer;
3. The penalty for missing an edge in the dependency tree of the student's answer;
4. The normalization coefficient for decision-making.

For this particular dataset, we acquired the following values:

1. The scaling parameter w_1 for the distance between nodes is calculated as

$$w_1 = \frac{1}{depth}, \tag{2}$$

   where *depth* is the distance between the root and a node
2. The penalty w_2 for missing and edge in the dependency tree of the correct answer is 2
3. The penalty w_3 for missing and edge in the dependency tree of the students answer is 0.6
4. The normalization coefficient w_4 for decision-making is calculated as

$$w_4 = \frac{1}{1 + \alpha(len_1 + len_2)}, \tag{3}$$

where $\alpha = 0.05$, len_1 and len_2 are lengths of the correct and the students' answers, respectively.

Examples of answer evaluations are displayed in Table 7.

Table 7. Evaluations of answers (*distance* from the correct answer).

	Answer 1 (Correct)	Answer 2	Answer 3	Answer 4	Answer 5
Question 1	0.0	0.3	0.3	0.6	0.9
Question 2	0.0	0.5	0.6	0.9	0.9
Question 3	0.0	0.6	0.6	0.7	0.5
Average	0.0	0.7	0.8	1.0	1.4

It is important to note that, even with fine-tuning, for this dataset, we cannot establish thresholds that would allow separating segments for each answer in order (the segment for the first answer, the segment for the second answer, etc.). There might be various reasons for that, however, reviewing them here would be beside the point, since it was not the goal in the first place. For this particular problem, we look for an increase in value with a decrease in relevance while establishing a threshold (or lack thereof), which is of methodological interest.

2.4. System Architecture

To conform to the requirements of the specific problem we investigated, a virtual dialogue assistant for remote exams has to be considered in the context of design and technical implementation for online education platforms. The essential part is to provide clear external interfaces encapsulating internal processes and ensure the internal processes are manageable, maintainable, and scalable. To achieve that, our solution is to follow modern design principles for building information systems, separating the system into distributable modules implementing business-logic subroutines and front-end modules, providing application programming interfaces (API) for requesting specifications of datatypes and scenarios supported by the system and queuing tasks, such as generating question, evaluating answers, and so forth. One notable advantage of this approach is that the system can be fine-tuned to use-cases supported by a given online education platform. For any modern information system, its distributed nature (also implying scalability) is not as much a feature, but a necessity; therefore, considering its architecture, it is vital to make sure it can provide those. One of the key measurable characteristics of a system to evaluate its scalability is the complexity of interfaces: internal and external. In our description of the components of the system, we implicitly demonstrate that the components exhibit a high level of decoupling, proving confidence in that even with the increased sophistication of particular modules, it ought not to produce a necessity for a significant increase in complexity of interfaces.

3. Discussion and Conclusions

The goal of this research was to investigate solutions for enhancing the process of evaluation of students' performance at online education platforms. We purposely do not imply our interest in developing a general question-and-answer system, as such a proposition would necessarily require a much deeper and broader analysis of areas beyond the scope of this work. While restrictions, by definition, limit freedom, they also provide guarantees, allowing for a more focused view on a problem. Here, we limited ourselves to looking into traditional and modern approaches to find whether and in what manner they may be applicable for building a system for evaluation of students' performance, and we restricted the environment in which this evaluation is ought to be performed—specifically, online education platforms. With the environment specified, we now review various approaches and techniques for both solving specific sub-problems and organizing them together into a solution offering new opportunities or improving the existing ones. To our

conclusion, it appears that both traditional techniques, requiring affordable computational resources and manual labor, and modern state-of-the-art methods relying on significantly large computational and storage resources, allow for building a system that can significantly improve performance evaluation at online education platforms. We believe that further improvements are primarily to be either of a quantitative nature (storage, computation, data mining, etc.) or of a methodological one.

Author Contributions: Conceptualization, A.M., O.M. and Y.M.; methodology, A.M., O.M. and Y.M.; software, A.M., A.S., P.K., A.R. and A.A.; validation, A.M., O.M., A.S., P.K., A.R. and A.A.; formal analysis, A.M., O.M. and Y.M.; investigation, A.M., O.M., A.S., P.K., A.R. and A.A.; resources, O.M. and Y.M.; data curation, O.M.; writing—original draft preparation, A.M. and O.M.; writing—review and editing, Y.M.; visualization, A.M. and O.M.; supervision, A.M., O.M. and Y.M.; project administration, A.M. and O.M.; funding acquisition, A.M. and O.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financially supported by ITMO University, 197101 Saint Petersburg, Russia.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Mclsaac, M.; Gunawardena, C. Distance education. In *Handbook of Research for Educational Communication and Technology: A Project of the Association for Educational Communication and Technology*; Routledge: London, UK, 1996; pp. 403–437.
2. Rumble, G. Re-inventing distance education, 1971–2001. *Int. J. Lifelong Educ.* **2001**, *20*, 31–43.
3. Global Monitoring of School Closures. Available online: https://en.unesco.org/covid19/educationresponse (accessed on 14 May 2021).
4. Witze, A. Universities will never be the same after the coronavirus crisis. *Nature* **2020**, *582*, 162–164. [CrossRef]
5. E-Learning Global Market Trajectory & Analytics. Available online: https://www.strategyr.com/market-report-e-learning-forecasts-global-industry-analysts-inc.asp (accessed on 14 May 2021).
6. Digital Education Market by Learning Type (Self-paced and Instructor-led Online Education), Course Type, End-user (Individual Learners and Academic Institutions, Enterprise and Government Organizations), and Geography (North America, Europe, APAC and RoW)-Forecast up to 2026. Available online: https://reports.valuates.com/market-reports/INFO-Othe-4I48/digital-education (accessed on 20 May 2021).
7. Raupach, T.; Brown, J.; Anders, S.; Hasenfuss, G.; Harendza, S. Summative assessments are more powerful drivers of student learning than resource intensive teaching formats. *BMC Med.* **2013**, *11*, 31–43. [CrossRef]
8. Kirkpatrick, R.; Zang, Y. The Negative Influences of Exam-Oriented Education on Chinese High School Students: Backwash from Classroom to Child. *Lang. Test. Asia* **2011**, *1*, 36. [CrossRef]
9. Newble, D. Revisiting 'The effect of assessments and examinations on the learning of medical students'. *Med Educ.* **2016**, *50*, 498–501. [CrossRef] [PubMed]
10. Perera-Diltz, D.M.; Moe, J.L. Formative and Summative Assessment in Online Education. *J. Res. Innov. Teach.* **2014**, *7*, 130–142.
11. Chaudhary, S.V.S.; Dey, N. Assessment in Open and Distance Learning System (ODL): A Challenge. *Open Prax.* **2013**, *5*, 207–216. [CrossRef]
12. Halova, E.Y.; Kobilarov, R.G. Advantages and Disadvantages of the Test Method for Checking and Evaluating of the Knowledge, the Skills and the Habits of Students. *AIP Conf. Proc.* **2010**, *1203*, 1325–1328. [CrossRef]
13. Indurkhya, N.; Damerau, F.J. *Handbook of Natural Language Processing*, 2nd ed.; CRC Press: Boca Raton, FL, USA, 2010.
14. Clark, A.; Fox, C.; Lappin, S. *The Handbook of Computational Linguistics and Natural Language Processing*; John Wiley & Sons: Hoboken, NJ, USA, 2010; doi:10.1002/9781444324044. [CrossRef]
15. Otter, D.W.; Medina, J.R.; Kalita, J.K. A Survey of the Usages of Deep Learning in Natural Language Processing. *arXiv* **2019**, arXiv:cs.CL/1807.10854.
16. An overview of empirical natural language processing. *AI Mag.* **1997**, *18*, 13.
17. Hutchins, W.J. Machine Translation: A Brief History. In *Concise History of the Language Sciences*; Elsevier: Amsterdam, The Netherlands, 1995. [CrossRef]
18. Estival, D.; Nowak, C.; Zschorn, A. Towards ontology-based natural language processing. In Proceeedings of the Workshop on NLP and XML (NLPXML-2004): RDF/RDFS and OWL in Language Technology, Barcelona, Spain, 25 July 2004. [CrossRef]
19. Huang, Y.; Cheng, Y.; Bapna, A.; Firat, O.; Chen, M.X.; Chen, D.; Lee, H.J.; Ngiam, J.; Le, Q.V.; Wu, Y.; Chen, Z. GPipe: Efficient training of giant neural networks using pipeline parallelism. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 103–112.
20. Alexeev, A.; Kukharev, G.; Matveev, Y.; Matveev, A. A Highly Efficient Neural Network Solution for Automated Detection of Pointer Meters with Different Analog Scales Operating in Different Conditions. *Mathematics* **2020**, *8*. [CrossRef]

21. Torfi, A.; Shirvani, R.A.; Keneshloo, Y.; Tavaf, N.; Fox, E.A. Natural Language Processing Advancements By Deep Learning: A Survey. *arXiv* **2021**, arXiv:cs.CL/2003.01200.
22. Keneshloo, Y.; Shi, T.; Ramakrishnan, N.; Reddy, C.K. Deep Reinforcement Learning for Sequence-to-Sequence Models. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*. [CrossRef]
23. Hazelwood, K.; Bird, S.; Brooks, D.; Chintala, S.; Diril, U.; Dzhulgakov, D.; Fawzy, M.; Jia, B.; Jia, Y.; Kalro, A.; et al. Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective. In Proceedings of the 2018 IEEE International Symposium on High Performance Computer Architecture (HPCA), Vienna, Austria, 24–28 February 2018. [CrossRef]
24. Wang, F.Y.; Zhang, J.J.; Zheng, X.; Wang, X.; Yuan, Y.; Dai, X.; Zhang, J.; Yang, L. Where does AlphaGo go: From church-turing thesis to AlphaGo thesis and beyond. *IEEE/CAA J. Autom. Sin.* **2016**, *3*. [CrossRef]
25. Guarino, N. *Formal Ontology in Information Systems: Proceedings of the 1st International Conference, Trento, Italy, 6–8 June 1998*; IOS Press: Trento, Italy, 1998; Volume 46.
26. Guizzardi, G. Ontology, Ontologies and the "I" of FAIR. *Data Intell.* **2020**, *2*, 181–191. [CrossRef]
27. Wolfram Data Framework. Available online: https://www.wolfram.com/data-framework (accessed on 15 July 2021).
28. how many goats in spain. Available online: https://www.wolframalpha.com/input/?i=how+many+goats+in+spain (accessed on 15 July 2021).
29. Das, B.; Majumder, M.; Phadikar, S.; Sekh, A. Automatic generation of fill-in-the-blank question with corpus-based distractors for e-assessment to enhance learning. *Comput. Appl. Eng. Educ.* **2019**, *27*, 1485–1495. [CrossRef]
30. Rakić, K. The proposal of the intelligent system for generating objective test questions in controlled natural language for domain knowledge based on ontology. In Proceedings of the 2016 International Conference on Smart Systems and Technologies (SST), Osijek, Croatia, 12–14 October 2016; pp. 135–138. [CrossRef]
31. Marrese-Taylor, E.; Nakajima, A.; Matsuo, Y.; Yuichi, O. Learning to Automatically Generate Fill-In-The-Blank Quizzes. In Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications; Association for Computational Linguistics, Melbourne, Australia, 19 July 2018; pp. 152–156. [CrossRef]
32. CH, D.R.; Saha, S.K. Automatic Multiple Choice Question Generation From Text: A Survey. *IEEE Trans. Learn. Technol.* **2020**, *13*, 14–25. [CrossRef]
33. Kurtasov, A. A System for Generating Cloze Test Items from Russian-Language Text. In Proceedings of the Student Research Workshop Associated with RANLP 2013, Hissar, Bulgaria, 9–11 September 2013; INCOMA Ltd. Shoumen, BULGARIA: Hissar, Bulgaria, 2013; pp. 107–112.
34. Wang, D.; Zhao, Y.; Lin, H.; Zuo, X. Automatic scoring of Chinese fill-in-the-blank questions based on improved P-means. *J. Intell. Fuzzy Syst.* **2021**, *40*, 5473–5482. [CrossRef]
35. Batura, T.; Charintseva, M. *Basics of Text Information Processing*; A.P. Ershov Institute of Informatics Systems (IIS) SB RAS: Novosibirsk, Russia, 2016.
36. Solovyev, A. Syntactic and Semantic Models and Algorithms in Question Answering. In Proceedings of the 13th All-Russian Scientific Conference "Digital libraries: Advanced Methods and Technologies, Digital Collections", RCDL 2011, Voronezh, Russia, 19–22 October 2011.
37. Schlaefer, N. *A Semantic Approach to Question Answering*; Verlag Dr. Müller: Riga, Latvia, 2011.
38. Solovyev, A. Dependency-based algorithms for answer validation task in Russian question answering. In *Language Processing and Knowledge in the Web*; Springer: Berlin/Heidelberg, Germany, 2013; Volume 8105. [CrossRef]
39. Scharpf, P.; Schubotz, M.; Youssef, A.; Hamborg, F.; Meuschke, N.; Gipp, B. Classification and clustering of arxiv documents, sections, and abstracts, comparing encodings of natural and mathematical language. In Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020, Virtual Event, China, 1–5 August 2020. [CrossRef]
40. Krstovski, K.; Blei, D.M. Equation Embeddings. *arXiv* **2018**, arXiv:stat.ML/1803.09123.
41. Bogoradnikova, D.; Makhnytkina, O.; Matveev, A.; Zakharova, A.; Akulov, A. Multilingual Sentiment Analysis and Toxicity Detection for Text Messages in Russian. In Proceedings of the 2021 29th Conference of Open Innovations Association (FRUCT), Tampere, Finland, 12–14 May 2021. [CrossRef]
42. Laptev, A.; Andrusenko, A.; Podluzhny, I.; Mitrofanov, A.; Medennikov, I.; Matveev, Y. Dynamic acoustic unit augmentation with bpe-dropout for low-resource end-to-end speech recognition. *Sensors* **2021**, *21*, 3063. [CrossRef] [PubMed]
43. Bruches, E.; Pauls, A.; Batura, T.; Isachenko, V. Entity Recognition and Relation Extraction from Scientific and Technical Texts in Russian. In Proceedings of the 2020 Science and Artificial Intelligence Conference (SAI Ence), Novosibirsk, Russia, 14–15 November 2020. [CrossRef]