Identification of argumentative sentences in Russian scientific and popular science texts

N V Salomatina¹, I S Pimenov² and E A Sidorova³

¹Sobolev Institute of Mathematics, Novosibirsk, Russia
²Novosibirsk State University, Novosibirsk, Russia
³A. P. Ershov Institute of Informatics, Novosibirsk, Russia

E-mail: salomatina_nv@live.ru

Abstract. In this study we analyze the applicability of specific machine learning algorithms to the task of detecting sentences containing argumentation in Russian text. We employ a collection of scientific and popular science texts with manually annotated argumentation to evaluate the quality of identifying argumentative sentences in terms of precision, recall, and $F$-measure. The experiment involves three algorithms: MNB, SVM, and MLP. The bag of words model is used for representing texts. Lemmas of words in analyzed sentences serve as features for the classification. We perform the automatic selection of informative features in accordance with Variance and $\chi^2$ criteria combined with the weight-based filtration of lemmas (via TF*IDF and EMI). The training set includes around 800 sentences, while the test set contains 180. The MNB algorithm demonstrates the highest $F$-measure and recall scores on almost all feature sets (maximal values reached equal 68.7% and 89% respectively), while the MLP algorithm shows the best precision for about half of feature selection variations (the maximal value is 72.5%).

1. Introduction

Research on automatic extraction of arguments from Russian texts is not particularly developed (among the few works are [1, 2, 3]), which is caused by the prior absence of, first, sets of texts in Russian language with annotated argumentation, and second, convenient tools for performing this type of annotation. One way to create argumentation corpora in Russian is through translating annotated texts in other languages, as done in [3] on the argumentation corpus of micro-texts in English and German (ArgMicro); this approach does not require the use of a special annotation tool. The set of web tools for argumentative annotation of texts, developed in the IIS SB RAS [4], provides a means to create thematic corpora, to visualize argumentative statements and argumentation schemes and to represent the argumentation structure of a text in form of a graph with two node types: information nodes (statements) and scheme nodes (reasoning models). These structures can then be extracted in a .json format for automatic processing.

The high labor-intensity of argumentative annotation (which is focused on deep textual structures) means that annotated texts are often insufficient for the efficient training of recognition systems even in the presence of annotated corpora (for example, in the English language). The presented work focuses on a small set of scientific and popular science texts, annotated primarily for analyzing specific reasoning models. In terms of genre characteristics, the chosen texts are distinct for their great lengths and diversity of linguistic expressions (which differentiates them from news articles, online
comments, essays and microtexts, frequently used in the construction of argumentation corpora for the English language).

Many applications implementing argumentation analysis rely on the automatic recognition of argumentation structures for the following tasks [5]:
1) processing of product reviews (for analyzing justifications of the expressed sentiments);
2) understanding and directing debates;
3) making decisions in decision support systems;
4) assessing the quality of texts (for example, student essays, scientific articles), etc.

In more than ten years of research on English corpora with argumentative annotation [6], the applicability of distinct machine learning algorithms has been evaluated in respect to specific subtasks of argumentation analysis, with the resulting scores becoming available for comparison.

The aim of the present work is 1) to analyze the applicability of certain machine learning methods (multinomial naive Bayes (MNB), support vector machine (SVM), and multilayer perceptron (MLP)) to Russian scientific and popular science texts for identifying sentences containing argumentation; 2) to conduct an experiment on a set of annotated texts and evaluate the quality of identifying argumentative sentences in terms of precision, recall, and F-measure.

2. Related works
Extraction of argumentation structures, both manual and automatic, is frequently performed in the following steps [7]:
- a) segmentation of the analyzed text into statements (clauses);
- b) classification of statements into argumentative and nonargumentative;
- c) identification of statements roles in the argumentation structure (the main thesis, claims, premises);
- d) identification of connections between argumentative statements in accordance with reasoning schemes and \ or the topic structure of the text.

Step b) might be skipped for texts with the prevalence of argumentative statements (for example, texts of a specific genre or theme, such as discussions or legal texts). Popular science texts contain a significant portion of nonargumentative segments, which requires a preliminary extraction of the argumentative part for improving efficiency in solving subtasks c) and d).

Thematically specialized argumentation research is often conducted on legal texts [8, 9], while frequently chosen genres include essays and debates [10, 11, 12] (at times arguments for debates are created in advance via extraction from Internet [13]), scientific texts [7], news articles [14], and online comments [15].

At times, in classifying statements as argumentative or nonargumentative, researches do not segment text into clauses at the first step, but instead perform the analysis at the level of distinct sentences. Argumentative sentences appear as search objects in works [8, 9, 11]. Frequently, the extraction focuses only on arguments supporting (or attack) the theses directly [11, 13, 15]. Some works aim to identify argumentative sentences or clauses serving as claims or premises, for instance, [6, 12, 13].

Prognostic models are examined in works [16, 17], which evaluate respectively the probability of a conference acceptance for a given paper and the quality of product reviews.

For the identification of argumentative sentences (or clauses), the most frequently used machine learning algorithms are the MNB [7, 8, 9, 14], maximum entropy method [8, 9], SVM [3, 8, 9, 12, 14, 15], logistic regression [14], decision trees, random forest [3, 10, 18], and neural networks [6, 18].

As a rule, employed feature sets are based on several types of textual properties: structural, lexical, syntactical, and contextual, along with the usage of markers. The role of features might be taken by n-grams (n ≥ 1), word couples, part of speech tags, grammatical tags (tense or person of a verb etc.), number of punctuation marks [8, 9, 15], positions in the sentence (inside the main or the subordinate clause), and discourse markers [10, 11, 12]. Noted features may be enhanced in accordance with search object types (for example, types of arguments, types of supporting arguments) [11, 15], rhetoric
[10] or topic [7] structure of the text, etc. The most efficient combinations of features and machine learning methods are often chosen empirically.

Features are frequently assigned binary values. At times they can be weighted, for instance, with the TF*IDF statistic [11, 17]. In the present work, we employ a preliminary filtration of features based on TF*IDF and EMI weights, Variance and $\chi^2$ criteria, and their various combinations.

Application of machine learning methods (MNB, maximum entropy, SVM) to legal texts for identifying argumentative sentences [8, 9] demonstrates that the best results are reached by the MNB and maximum entropy algorithms trained on n-grams (n = 1, 2, 3), word couples, distinct parts of speech, keywords, punctuation, and text statistic. Precision as the measure of quality is dependent on the training set and reaches 73% and 80% for legal texts. Notably, legal texts are written in the highly formalistic language, which is not the case for popular science texts.

In [3], a Russian argumentation corpus of micro-texts is used for the task of automatic identification of pro and contra statements (the highest $F$-measure reached is 69.9%).

In the present work, we focus on analyzing applicability of three methods: MNB, SVM, and MLP. The former two are frequently mentioned in related works on argumentation mining.

3. The task of identifying argumentative sentence
A sentence is considered to be argumentative if it contains statements from the reasoning structure of the text (premises or claims) for supporting or attacking its main theses. The identification of argumentative sentences in texts presupposes the classification of the set of sentences $S = \{s_i\}$ from texts of the collection into two groups: $C = \{C_o, C_d\}$, where $C_o$ labels the class of sentences containing a premise or a claim, while $C_d$ marks the class of sentences without argumentation ($C_o \cap C_d = \emptyset$). The training set is employed to train a classifier for the precise classification $\hat{F}: S \times C \in \{0, 1\}$ that will be as close to the function $F(s_i, C) = \{0, i f s_i \notin C; 1, i f s_i \in C\}$ ($j = 1, 2$).

Learning and testing are performed on the set of expertly annotated examples (with the values of the function $F$ known): $S = S^T \cup S^T$, where $S^T$ is the training set, and $S^T$ is the test set. This enables the automatic evaluation of quality measures (precision $P$, recall $R$, and $F$-measure) on $S^T$.

4. Methods of the feature selection and classification
In our case, the argumentative classification presupposes performing the following stages:

- creating the collection of texts of the chosen genre and/or theme;
- performing the argumentative annotation of texts;
- constructing the training and test sets of texts;
- preprocessing texts, selecting and/or filtering features, constructing text models;
- training the classifier with the chosen method;
- applying the classifier to the test set and evaluating the quality of the classification.

The popular science collection is created from the texts of online publications in the corresponding genre, while the scientific collection is based on articles in linguistics and computer technologies from the Ru-RSTreebank corpus [19] (properties of their rhetoric structures have not been considered during the argumentative annotation). For the correctness of the training procedure, we try to approximately equalize the amounts of argumentative and nonargumentative sentences in the training set while preserving integrity of the source texts, which to a certain extent complicates the training set creation.

The preprocessing of texts includes lemmatization of words with the PyMorphy2 program and removal of words without cyrillic symbols. The bag-of-words model is then used to represent texts as vectors of normalized unigrams.

As the linguistic selection of features is time-intensive, in the present work we employ an automatic filtration of features. It is based on weighing vector components via TF*IDF and expected mutual information (EMI) measures, along with selecting unigrams by the use of Variance and $\chi^2$ criteria ([20]).
The TF-IDF measure reaches high values if the term frequently appears in a small set of documents, and low values if the term is encountered in many documents, which enables the filtration of function words and common words with generic meanings (while markers or their constituents might appear among these words, they are not likely to be as efficient in detecting argumentation). The EMI measure allows the filtration of lemmas that appear frequently in different classes, as its values correspond to the quantity of information provided by the presence or absence of the feature in the sentences of different classes; the maximal value is reached when the feature appears only in sentences of the same class.

The Variance criterion (Var) enables the removal of features reaching similar values (with the similarity defined by the chosen threshold) in all training sentences. The value of the $\chi^2$ criterion signifies the difference between the expected and observed probabilities of the feature and the class appearing, on the hypothesis of their independence.

We perform the automatic filtration of features through the use of functions from the Scikit-learn library [21], with the threshold values chosen empirically in accordance with recognition results.

For the classification of sentences, we have chosen MNB and SVM methods [21], frequently mentioned in related articles for solving analogous tasks, along with the MLP algorithm [21]. Deep configuration of algorithm parameters has not been performed as its values would be applicable only to the current dataset and would need to be reconfigured on its planned expansion.

The MNB algorithm. For evaluating the probability of the sentence $s$, belonging to the class $C_a$ or $C_b$, the MNB algorithm applies the Bayes theorem to the training set. Each sentence is represented as a multiset of lemmas (with assigned binary values for the chosen features), and its class is determined as the one with the highest posterior probability. Features (lemmas) are assumed to appear in the text independently of each other. In calculating probabilities, Laplace smoothing is used to compensate for the training set incompleteness.

The SVM algorithm. In the documents vector space, the algorithm locates the separating surface between two classes that is the most distant from all points of the training set. The radial basis function RBF: $\exp(-\gamma l|x−x'|^2)$ is chosen as the kernel (in the experiment, the quadratic polynomial function achieves lower recognition quality). Two parameters influence the SVM training with the RBF core: $C$ (regulates the balance between classification errors and the simplicity of the decision surface) and $\gamma$ (determines the significance of a single training example).

The MLP algorithm. The algorithm consists in training of neural network with the use of backpropagation. For the sentence $s$, components of its vector $(x_1, x_2, ..., x_m)$ form the input layer. As the activation function we use $f(x) = \max (0, x)$. By default, the perceptron consists of 100 layers and optimizes neuron weights by a quasi-Newton method for minimizing the logarithmic loss function. The classifier repeats iterations until reaching either convergence of the specified degree or the maximum number of iterations.

5. The experiment of argumentative classification
The annotated set used in the experiment includes 96 texts: 10 of which are scientific and 86 are popular science. Each text has been annotated by one expert. A large part of popular science texts has been annotated for the analysis of specific argumentation schemes. Annotations created for this purpose do not always reflect the argumentation structure of a text in a complete or coherent (in respect to the resulting graph) manner, at times the main thesis remains unmarked. Consequently, the experiment focuses on 22 annotated texts in which the noted tendencies are either absent or minimized via the contraction of the text to its annotated section (preserving the integrity of paragraphs).

We conduct the experiment to evaluate the identification of sentences, which contain arguments, with the MNB, SVM, and MLP methods. Precision, recall, and $F$-measure scores are calculated for assessing performance of algorithms, and most prominent classification errors are remarked further. The experiment also examines possible ways of optimizing the feature space.
5.1. Experimental data
The training $S^a$ and the test $S^t$ sets (based on 16 and 6 texts respectively) are formed with almost equal amounts of argumentative (from the $C_a$ class) and nonargumentative (from the $C_d$ class) sentences. Quantitative characteristics of the $S^a$ and $S^t$ sets are provided in table 1.

| Table 1. Quantitative characteristics of experimental data. |
|---|---|---|
| SL | 392 | 390 | 782 |
| ST | 91 | 89 | 180 |

During the preprocessing of sets, all words formed by the sequences of non-Cyrillic symbols are the removed from texts, while the others are normalized into their lemmas.

5.2. Evaluating the argument identification
At the training stage, we reduce the dimensionality of the feature space via weighting lemmas (with TF*IDF and EMI metrics) and employing Variance and $\chi^2$ criteria, along with their combinations. Classifiers are trained and tested on features (lemmas) selected in accordance with chosen weights and criteria. After the filtration of features, they are assigned binary values on distinct samples (sentences), with each value determined by the presence or absence of the lemma in the sentence.

5.2.1. Learning. The maximum size of a feature set (without filtration) equals 3302 different lemmas. We examine 15 ways of filtering features. Sizes of different feature sets $V_i (i = 1,...,15)$, filtration conditions, and filtration thresholds are given in table 2 ($p_1, p_2, p_3$ denote values of the corresponding weights or criteria).

| Table 2. Filtration conditions and thresholds. |
|---|---|---|
| Filtration type | Threshold | $|V_i|$ |
| --- | --- | --- |
| TF*IDF $> p_1$ | $p_1 = 0.5$ | 3164 |
| EMI $> p_2$ | $p_2 = 0.001$ | 800 |
| TF*IDF $> p_1$ & EMI $> p_2$ | $p_1 = 0.5$ ; $p_2 = 0.001$ | 718 |
| Var $> p_3$ | $p_3 = 0.01$ | 333 |
| Var $> p_3$ & TF*IDF $> p_1$ | $p_3 = 0.01$ ; $p_1 = 0.5$ | 202 |
| Var $> p_3$ & EMI $> p_2$ | $p_3 = 0.01$ ; $p_2 = 0.001$ | 168 |
| Var $> p_3$ & TF*IDF $> p_1$ & EMI $> p_2$ | $p_3 = 0.01$ ; $p_1 = 0.5$ ; $p_2 = 0.001$ | 88 |
| $\chi^2$ (best) | 25% best | 825 |
| $\chi^2$ (best) & TF*IDF $> p_1$ | 25% best ; $p_1 = 0.5$ | 791 |
| $\chi^2$ (best) & EMI $> p_2$ | 25% best ; $p_2 = 0.001$ | 200 |
| $\chi^2$ (best) & TF*IDF $> p_1$ & EMI $> p_2$ | 25% best ; $p_1 = 0.5$ ; $p_2 = 0.001$ | 179 |
| $\chi^2$ (best) | 10% best | 330 |
| $\chi^2$ (best) & TF*IDF $> p_1$ | 10% best ; $p_1 = 0.5$ | 317 |
| $\chi^2$ (best) & EMI $> p_2$ | 10% best ; $p_2 = 0.001$ | 80 |
| $\chi^2$ (best) & TF*IDF $> p_1$ & EMI $> p_2$ | 10% best ; $p_1 = 0.5$ ; $p_2 = 0.001$ | 71 |

Thresholds have been assigned empirically chosen values, in whose neighborhood each classifier reaches about the same quality scores, which, in turn, decrease on distancing from the chosen value. As each vector is filled with 99% zeroes on average (due to small lengths of sentences serving as samples), the TF*IDF value of the analyzed feature is compared against the threshold at every sentence, and the feature is recognized as informative if it exceeds the threshold at least once.
Similarly, the sparseness of vectors influences the choice of a relatively low value for the EMI threshold: for each given feature, its EMI value is calculated over the whole vector in accordance with its frequencies in argumentative and non-argumentative sentences. The Variance threshold filters lemmas that have the same value in more than 99% of the samples (while for a given sentence, the average relative frequency of its lemmas across the training set equals 0.5%). The $\chi^2$ threshold is configured to select the fixed percent of features that are most strongly dependent on their argumentation class. Threshold values remain the same on combining several filtration conditions.

Practically, all of the reasoning models which appear in annotated texts, are represented both in the training and test set. Out of twenty-two distinct models, just two are absent in the training set, and five appear only in the training set (one such model, VerbalClassification, is used mostly in scientific texts, and other six models are generally rare).

5.2.2. Recognition. Quality scores are calculated for each algorithm (MNB, SVM, MLP) across all feature sets ($V_i$). The results of the experiment are presented in table 3 (recognition scores are indicated as percent values). The bold font marks two highest scores for each algorithm, separately for precision, recall and F-measure.

| $|V|$ | MNB | SVM | MLP | MNB | SVM | MLP | MNB | SVM | MLP |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 3302 | 56.86 | 53.75 | 55.95 | 63.74 | 47.25 | 51.65 | 60.10 | 50.29 | 53.71 |
| 3164 | 58.33 | 47.17 | 55.29 | 61.54 | 54.95 | 51.65 | 59.89 | 50.76 | 53.41 |
| 825 | 54.55 | 48.53 | 47.13 | 59.34 | 36.26 | 45.05 | 56.84 | 41.51 | 46.07 |
| 800 | 53.06 | 55.07 | 60.53 | 57.14 | 41.76 | 50.55 | 55.03 | 47.50 | 55.09 |
| 333 | 56.67 | 51.43 | 56.10 | 56.04 | 39.56 | 50.55 | 56.35 | 44.72 | 53.18 |
| 330 | 57.14 | 49.21 | 59.74 | 65.93 | 34.07 | 50.55 | 61.22 | 40.26 | 54.76 |
| 791 | 54.21 | 48.11 | 55.56 | 63.74 | 56.04 | 60.44 | 58.59 | 51.78 | 57.89 |
| 718 | 52.59 | 57.14 | 59.77 | 67.03 | 35.16 | 57.14 | 58.94 | 43.54 | 58.43 |
| 317 | 56.82 | 55.93 | 59.42 | 82.42 | 36.26 | 45.05 | 67.26 | 44.00 | 51.25 |
| 200 | 54.55 | 57.63 | 54.79 | 72.53 | 37.36 | 43.96 | 62.26 | 45.33 | 48.78 |
| 168 | 52.58 | 62.90 | 72.50 | 56.04 | 42.86 | 63.74 | 54.26 | 50.98 | 67.84 |
| 80 | 51.68 | 58.11 | 60.27 | 84.62 | 47.25 | 48.35 | 64.17 | 52.12 | 53.66 |
| 179 | 56.83 | 56.67 | 64.15 | 86.81 | 37.36 | 37.36 | 68.70 | 45.03 | 47.22 |
| 88 | 50.97 | 59.18 | 63.64 | 86.81 | 31.87 | 30.77 | 64.23 | 41.43 | 41.48 |
| 71 | 53.29 | 60.66 | 64.58 | 89.01 | 40.66 | 34.07 | 66.67 | 48.68 | 44.60 |

The third row of the table shows the results of identifying argumentative sentences without filtering features. The following five rows demonstrate scores reached by the separate application of each filter, while the rest show the results of combined filtration. The first column indicates the sizes of feature sets after filtration (filtration conditions are shown in table 2). For all algorithms on aggregate, using one filter separately improves precision in half of cases, recall and F-measure in their third. In terms of precision, MNB reaches the highest values on separate filtration through either TF*IDF or $\chi^2$ (10% best), in terms of recall and F-measure, on combined filters. Reducing the numbers of feature yields, as a rule, a drop in recall; exceptions are SVM with the TF*IDF filtration and MNB with the $\chi^2$ (10% best) one. The F-measure value for MNB increases only on the use of $\chi^2$ (10% best); SVM scores improve with the TF*IDF filtration; MLP reaches higher scores on filtering through either $\chi^2$ (10% best) or EMI.

Simultaneous use of several filters affects classification scores in different ways. Combining filters does not improve the precision of MNB, yet almost all combinations benefit the precision of SVM and MLP (best results are provided by $\{\text{Var} \& \ EMI\}$ and $\chi^2$ (10% best) & TF*IDF & EMI$)$. Recall improves, for MNB, on filtering with $\{\chi^2(10\% \text{ best}) \& \text{TF*IDF} \& \text{EMI}\}$ or $\{\chi^2(10\% \text{ best}) \& \text{EMI}\}$; for
SVM, on using \{TF*IDF\} or \{χ\^2(25% best) & TF*IDF\}; for MLP, with \{χ\^2(25% best) & TF*IDF & EMI\}, F-measure reaches higher values on filtering with, for MNB, \{χ\^2(25% best) & TF*IDF & EMI\} or \{χ\^2(10% best) & EMI\}; for SVM, \{Var & EMI\} or \{χ\^2(10% best) & EMI\}; and for MLP, \{Var & EMI\} or \{TF*IDF & EMI\}. To summarize, filtration of features improves the overall quality, yet also requires configuring conditions for each algorithm and for each score type.

It is important to note that, as a rule, each classifier reliably identifies either argumentative (MNB) or nonargumentative (SVM and MLP) sentences. Comparable values in identifying the sentences of both classes are rarely achieved. The highest precision score (72.50%) belongs to the MLP algorithm on 168 features where it also reaches its best recall and F-measure. The highest recall (89.01%) and F-measure (68.70%) are reached by the MNB method on the sets of 71 and 179 features respectively.

While we are yet to address the efficiency of using linguistic-based features, preliminary results demonstrate that, for example, employing lemmas of specified parts of speech (adjectives, verbs, and adverbs) improves SVM scores in a more prominent manner than those of the MNB and MLP.

The analysis of classification errors demonstrates that the largest part of incorrectly identified the sentences is constituted by sentences either without markers or with ambiguous markers (znachit; poskol’ku; govorit’ o tom, chto; ...) which appear frequently both in argumentative and nonargumentative training samples. At the same time, those argumentative sentences corresponding to reasoning models absent in the training set have been identified correctly in the test set due to containing rhetoric and argumentative markers (mozhet byt’; tem samym; otsyuda mozhno naiti; okazyvaetsya, chto; dlya togo, chtoby; napisano, chto; ...).

Results of the conducted experiment are not directly comparable with values presented in related works due to language and genre differences in data, and also to the usage of different quality scores. At the same time, in terms of precision, our best result (72.5% with MLP on 168 features) approaches the value reached in [8] (73%).

6. Conclusion
In this work we analyze the applicability of three machine learning algorithms (MNB, SVM, MLP) for identifying argumentative sentences in scientific and popular science texts. We show that these methods can be employed in the shortage of annotated training data and linguistic expertise.

The performance of algorithms is evaluated across different feature sets constructed via the filtration of lemmas in accordance with the TF*IDF and EMI measures, Variance and χ\^2 criteria. We demonstrate that the filtration of features improves precision in half of the cases, recall and F-measure in their one-third for all three algorithms on aggregate, and share observations on filtering features for each algorithm. The classification quality approaches values reached in related works on larger collections of texts in English language.

The inclusion of new annotated texts into the collection and implementation of the linguistic analysis of features must result in improved quality values in identifying argumentative sentences. It will also enable a comparative study of argumentation in scientific and popular science texts.

Acknowledgments
The study was carried out within the framework of the state contract of the Sobolev Institute of Mathematics (project no. 0314-2019-0015).

References
1. Achmadeeva I et al 2019 Indicator Patterns as Features for Argument Mining https://ieeexplore.ieee.org/document/8958295 (27.04.2021)
2. N V Salomatina et al 2021 J. Phys.: Conf. Ser. 1715 012013
3. Fishcheva I and Kotelnikov E 2019 Cross-Lingual Argumentation Mining for Russian Texts Proc. Of the 8th International Conference “Analysis of Images, Social Networks and Texts” (Kazan) pp 134–144
4. Sidorova E et all 2020 Platforma dlya issledovaniya argumentacii v nauchno-populyarnom
diskurse Ontologiya Proektirovaniya 10 (38) (in Russian) pp 489–502

[5] Lawrence J and Reed C 2019 Argument Mining: A Survey Computational Linguistics 45 (4) pp 765–818

[6] Daxenberger J et al 2017 What is the Essence of a Claim? Cross-Domain Claim Identification Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing pp 2055–66

[7] Lawrence J et al 2014 Mining arguments from 19th century philosophical texts using topic based modelling Proc. of the First Workshop on Argumentation Mining (Baltimore, Maryland) (ACL) pp 79–87

[8] Moens M-F et al 2007 Automatic Detection of Arguments in Legal Texts Proc. of the 11th International Conference on Artificial Intelligence and Law (NY, USA) ACM Press pp 225–230

[9] Mochales Palau R and Moens M-F 2009 Argumentation mining: The detection, classification and structure of arguments in text Proc. of the 12th Int. Conf. on Artificial Intelligence and Law (Barcelona) ACM NY pp 98–109

[10] Madnani N et al 2012 Identifying high-level organizational elements in argumentative discourse Proc. of the 2012 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Montreal) pp 20–28

[11] Xinyu Hua and Lu Wang 2017 Understanding and Detecting Supporting Arguments of Diverse Types Proc. of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers) pp 203–208

[12] Stab C and Gurevych I 2014 Identifying argumentative discourse structures in persuasive essays Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (Doha) pp 46–56

[13] Levy R et al 2014 Context Dependent Claim Detection. Proc. of the 25th International Conference on Computational Linguistics: Technical Papers (Dublin, Ireland) pp 1489–1500

[14] Sardianos C et al 2015 Argument Extraction from News Proc. of the 2nd Workshop on Argumentation Mining (Denver, Colorado) pp 56–66

[15] Park J and Cardie C 2014 Identifying Appropriate Support for Propositions in Online User Comments Proc. of the First Workshop on Argument Mining (Baltimore, Maryland USA) pp 29–38

[16] Accuosto P and Saggion H 2019 Transferring knowledge from discourse to arguments: A case study with scientific abstracts Proc. of the 6th Workshop on Argument Mining (Florence, Italy) pp 41–51

[17] Passon M et al 2018 Predicting the Usefulness of Amazon Reviews Using Off-The-Shelf Argumentation Mining Proc. of the 5th Workshop on Argument Mining, (Brussels, Belgium) pp 35–39

[18] Yamen A et al 2017 Unit segmentation of argumentative texts Proc. of the 4th Workshop on Argument Mining (Copenhagen) pp 118–128

[19] Pisarevkaya D et al 2017 Towards building a discourse-annotated corpus of Russian Proc. of the International Conference «Dialogue 2017» Computational Linguistics and Intellectual Technologies (Moscow) pp 194–204

[20] Manning C and Schutze H 2000 Foundation of Statistical Natural Language Processing The MIT Press Cambridge Massachusetts p 680

[21] Scikit Learn Homepage https://scikit-learn.org/stable/supervised_learning.html (25.05.2021)