Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2025"

April 23–25, 2025

The methodology of multi-criteria evaluation of text markup models based on inconsistent expert markup

Alexander Levikin
MSU
s02210450@gmail.com

Ildar Khabutdinov
Antiplagiat Company
khabutdinov@ap-team.ru

Andrey Grabovoy
Antiplagiat Company
grabovoy@ap-team.ru

Konstantin Vorontsov
MSU Institute for Artificial Intelligence
vorontsov@mlsa-iai.ru

Abstract

A wide class of natural language processing tasks is solved using markup. At the moment, the vast majority of models and datasets rely on a simple markup structure containing only fragments and labels. Moreover, simple classification metrics such as F_1, Precision, Recall are used to evaluate the model's accuracy. The problem with such metrics is that they do not take into account all aspects of the markup structure and that they are applicable only under the assumption of the existence of an ideal markup. This paper proposes a more general and universal markup structure that allows solving complex problems and builds a methodology for multi-criteria evaluation of text markup models based on inconsistent expert markup. After that, the application of the constructed method is considered to assess the quality of the model obtained within the winning algorithm of the "READ//ABLE" competition, which focused on building an effective essay markup system. The results demonstrate that the new markup structure and evaluation approach provides a more comprehensive and accurate assessment of model performance, addressing the limitations of traditional metrics by accounting for complex markup scenarios and expert inconsistencies.

Keywords: multi-criteria assessment methodology, inconsistent text markup.
DOI: 10.28995/2075-7182-2025-23-1066-1080

Методика многокритериального оценивания моделей разметки текста по несогласованным экспертным разметкам

Александр Левыкин
МГУ
s02210450@gmail.com

Ильдар Хабутдинов
Антиплагиат
khabutdinov@ap-team.ru

Андрей Грабовой
Антиплагиат
grabovoy@ap-team.ru

Константин Воронцов
Институт ИИ МГУ
vorontsov@mlsa-iai.ru

Аннотация

С использованием разметок решается широкий класс задач обработки естественного языка. На данный момент подавляющее большинство моделей и датасетов опираются на простую структуру разметки, содержащую лишь фрагменты и метки. Более того, для оценки точности модели используются простые метрики классификации, такие как F_1, Precision, Recall. Проблема таких метрик в том, что они не учитывают все аспекты структуры разметки, и в том, что они применимы лишь в предположении существования идеальной разметки. В данной работе описывается более общая и универсальная структура разметки, позволяющая решать комплексные задачи, и строится методика многокритериального оценивания моделей разметки текста по несогласованным экспертным разметкам. После чего рассматривается применение построенного метода для оценки качества модели, полученной в рамках конкурса "ПРО//ЧТЕНИЕ", целью которого являлось создание эффективной системы разметки эссе. Результаты показали, что новые структура разметки и подход к оценке обеспечивают более полную и точную оценку эффективности модели, устраняя ограничения традиционных метрик за счет учета сложных сценариев разметки и несогласованности действий экспертов.

Ключевые слова: методика многокритериального оценивания, несогласованные разметки.

1 Introduction

1.1 Motivation and Contribution

The field of Natural Language Processing encompasses a diverse range of tasks aimed at extracting, analyzing, and utilizing information from textual data. One fundamental approach to solving these tasks is through the use of markup systems. Markup enables the identification, classification and annotation of textual elements. These include detecting manipulation in news articles (Ott et al., 2011), identifying evaluative language in texts (Wiebe, 2000), classifying documents by topic or category (McCallum and Nigam, 1998), determining sentiment polarity (Pang et al., 2002), recognizing emotions expressed in text (Alrasheedy et al., 2022) and automating the evaluation of students' essays (Khabutdinov et al., 2024). By systematically tagging textual fragments with appropriate labels a structured representation is created.

Current approaches to markup evaluation often rely on datasets and metrics that vary in complexity and scope. For instance, the MultiCoNER (Malmasi et al., 2022; Fetahu et al., 2023) dataset uses the BIO scheme to annotate tokens within sentences, with quality metrics like Precision, Recall, and F1 scores (Buckland and Gey, 1994; Kawata and Kikui, 2019) for both tagging and mention detection. However, this approach fails to account for partially matching spans, assumes a single reference markup, and lacks support for complex tasks such as linking fragments or adding multiple tags and comments.

Another example is the RURED (Gordeev et al., 2020) dataset, which includes named entities and their relations within texts. It employs metrics such as Cohen's Kappa (Sim and Wright, 2005) to measure inter-annotator agreement. While this dataset supports links between fragments, it does not accommodate overtexts, fragment combinations, or multiple tags for a single fragment or link.

Both approaches highlight limitations in existing systems, particularly in handling complex structures, multi-annotator scenarios, and nuanced evaluation criteria. Addressing these gaps is critical for advancing markup systems and their applications.

In order to identify more complex and composite language techniques, such as multistep manipulation, it becomes necessary to take into account the connection between fragments and group them. In addition, there is a desire to add comments and overtexts to the selected fragments. All these wishes are taken into account in the built markup structure described in the next chapter. After building the model, the question arises about evaluating the quality of its work. The difficulty of evaluation lies in the fact that there is no ideal, absolutely correct markup, but only a set of expert markups that differ slightly from each other. Therefore, when evaluating a model, we are not talking about its quality, but only about its consistency and similarity with experts. This article will propose a methodology for multi-criteria evaluation of text markup models based on inconsistent expert markup.

1.2 Method validation

We validate our approach at the "READ//ABLE" competition. The competitive task is to overcome a given technological barrier by building an algorithm for marking up Unified State Exam (USE) essays[1]. According to the procedure for conducting the final partitioning stage, it overcomes the technological barrier if the partitioning algorithm solves the competition problem and its average accuracy of algorithmic partitioning on the final sample is not worse than the average accuracy of expert partitioning calculated from expert partitions obtained under time-constrained conditions. We collaborate with the competition winner to explore the markup approach to the solution. The model architecture consists of various components to detect factual, logical, grammatical and speech (lexical violation) errors, as well as to highlight meaning blocks. Below is a brief description of the architecture.

Nowadays, one of the most effective open-source models in the Grammatical Error Correction task for the English language is the GECToR (Omelianchuk et al., 2020) model. For the grammar checker they have adapted the GECToR architecture for Russian and named it accordingly — RuGECToR (Khabutdinov et al., 2024). The choice of the architecture is due to the fact that it is easy to interpret and does not require a large amount of training data. The RuGECToR model is also utilised to check punctuation

[1] https://fipi.ru

compliance, despite the fact that punctuation compliance is not examined as part of the competition.

In order to verify compliance with speech norms, they use both rule-based and transformer-based (Vaswani et al., 2017) models. The classical model detects repetitions and tautologies in adjacent sentences, while the BERT-based (Devlin et al., 2019; Yang et al., 2019b; He et al., 2020; Warner et al., 2024; Liu et al., 2019) model finds more complex errors by classifying tokens.

The fact checker implements a pipeline for automated fact verification in text, combining document retrieval, segment extraction, and claim classification. The pipeline first uses an Anserini-based (Yang et al., 2017) search engine to retrieve relevant documents for a given claim. Extracted documents are processed by a Sentence Transformer (Reimers and Gurevych, 2019) to identify the most relevant segments based on cosine similarity between embeddings of the query and text passages. As knowledge bases they use collections of historical and literary documents, Wikipedia and news history. The BERT model for sequence classification then evaluates the relationship between the query and the retrieved segments.

The text logic error checker combines several approaches, the results of which are then aggregated.

The first approach has two main steps: candidate search and candidate classification. The candidate search starts with the comparison of candidate-reference pairs, where a candidate is a sentence in which a logical error is possible, and the reference is a fragment with which a logic violation occurs. Each pair is passed to the Question Answering (Yang et al., 2019a) BERT-based model input to refine the boundaries of the beginning and the end of the fragment. Then candidate-reference pairs with refined boundaries are fed to the input of the candidate classifier to get an error code or information that there is no logical error.

The second approach finds logical errors in the division of text into paragraphs, identifying cases where two paragraphs should be merged because they are logically related. Using the BERT model to evaluate the connectedness of paragraphs, the algorithm checks whether they can be merged without losing meaning.

The third approach also uses BERT-like models to predict the probability of logical succession between sentences in order to detect different types of errors. The first model evaluates the relationship between sentences using the Next Sentence Prediction (Shi and Demberg, 2019) task, and if the probability of logical succession between two sentences is low, it marks it as a logical sequence violation. The second model analyses the violation of causality between two sentences by binary classification.

The checker for meaning block detection in essays operates in several distinct stages. First, the input text is segmented into sentences. Next, the embeddings are passed through a BERT-base model to generate contextualized token representations. These representations are processed by a Conditional Random Field layer (Lafferty et al., 2001), which assigns a semantic label to each token based on its context and the subject-specific model. Predicted labels are aggregated to form contiguous spans representing meaning blocks. In the final stage, a post-processing step aligns the detected spans with sentence boundaries.

The evaluation results demonstrate that the algorithm achieves markup quality comparable to human annotators, particularly in overall consistency and tagging accuracy. While human annotators outperform the algorithm in fragment text consistency, the algorithm excels in maintaining consistent tagging and producing cohesive markup when fragments are aggregated. These findings highlight the algorithm's strengths in systematic tasks and suggest areas for improvement, particularly in nuanced text selection, to further align its performance with human capabilities.

2 Problem Statement

A generalized markup structure is proposed for consideration, which has the following form.

The L markup is a set of markup elements:

$$L = \{E_1, .., E_n\}, \tag{1}$$

where E_i is a markup element.

The markup element E is a triple:

$$E = (\{F_1, .., F_m\}, \{O_1^E, .., O_k^E\}, \{t_1^E, .., t_l^E\}), \tag{2}$$

where F_i is a fragment, O_i^E is a overtext, t_i is a tag (label).

Fragment F is a triple:

$$F = (s, f, \{t_1^F, .., t_v^F\}, \{O_1^F, .., O_q^F\}), \tag{3}$$

where $s, f \in \mathbb{R}$ are the beginning and end of the selected fragment, t_i is the tag, O_i^F is the text.

The O overtext is a two:

$$O = (C, \{t_1^O, .., t_p^O\}), \tag{4}$$

where C is a comment string, t_i is a tag. The superscripts F, E, or O emphasize that the tag/overtext refers to a fragment, markup element, or overtext, respectively.

A tag (the same as a label) t is an element of the tag dictionary, $t \in T$. The T tag dictionary is a set of words and phrases organized into a structure and used in markup, $T = \{t_1, .., t_n\}$.

To evaluate the consistency of the resulting markup, it is necessary to build a mapping $C(L_1, L_2)$:

$$C(L_1, L_2) : L_1 \times L_2 \longrightarrow [0, 1]. \tag{5}$$

3 Proposed Method

The comparison of the two markups' similarity is based on the F-measure:

$$F(A, B) = \frac{2|A \cap B|}{|A| + |B|}, \tag{6}$$

where A, B are some sets.

Let L_1, L_2 be markups of the same document. The consistency $C(L_1, L_2)$ of markups L_1, L_2 is a weighted average of criteria, each evaluating a part of the markup:

$$C(L_1, L_2) = \sum_{i=1}^{n} w_i \cdot C_i(L_1, L_2), \tag{7}$$

where $w_1 + ... + w_n = 1$, and $w_i \geq 0$. Each criterion assesses the similarity of certain markup components, such as overtexts, fragments, tags, etc. Moreover, each criterion can be composite and itself be a weighted average of its criteria.

Since some components of the markup (markup elements, fragments, overtexts) have a composite structure, to compute the similarity of sets consisting of them, it is necessary to establish an accordance between an object from one set $x_i \in X$ to an object from another set $y_j \in Y$, i.e., find the most similar objects from the two sets and associate them:

$$A_{X,Y} = \{(i_1, j_1), .., (i_q, .., j_q)\}, \tag{8}$$

where each index pair (i, j) means that element x_i is associated with element y_j. The best accordance, i.e., one that maximizes the consistency of the two sets, is called optimal accordance:

$$A_{X,Y}^{opt} : C(X, Y; A_{X,Y}^{opt}) = \max_{A_{X,Y}} C(X, Y; A_{X,Y}) \tag{9}$$

Note that in formula (7), the consistency $C(L_1, L_2)$ of markups L_1, L_2 is computed with the optimal accordance of markup elements, fragments, and overtexts: $C(L_1, L_2) \equiv C(L_1, L_2; A^{opt})$. Further in formula notations, the dependence on the optimal accordance will be omitted. Similarly, the consistency of fragments and overtexts in the final consistency is calculated with their optimal accordance. Thus, in the process of computing the consistency of sets of objects with a composite structure, the task of finding the optimal accordance is solved.

In markup tasks, there is a set of documents $D = \{d_1, .., d_n\}$, and each document d contains markups: $d = \{L_1, .., L_m\}$. It is assumed that each document contains algorithmic markup L^{alg} and several expert markups. To evaluate the model, the following quantities are introduced: Average accuracy of algorithmic markup (AMA):

$$\text{AMA} = \frac{1}{|D|}\sum_{d \in D} \frac{1}{|d|-1} \sum_{\substack{L \in d,\\ L \neq L^{alg}}} C(L^{alg}, L) \tag{10}$$

Average accuracy of expert markups (EMA):

$$\text{EMA} = \frac{1}{|D|}\sum_{d \in D} \frac{2}{(|d|-1)(|d|-2)} \sum_{\substack{L_1, L_2 \in d\\ L_1, L_2 \neq L^{alg}}} C(L_1, L_2) \tag{11}$$

Relative accuracy of algorithmic markup (RMA):

$$\text{RMA} = \frac{\text{AMA}}{\text{EMA}} \tag{12}$$

If RMA ≥ 1, then it can be stated that the algorithmic markups are consistent with expert markups at least as well as expert markups are consistent with each other; in other words, it can be said that the markup algorithm works no worse than experts.

3.1 Consistency of markup elements accordance

This criterion calculated as the F-measure - the ratio of elements for which accordance was established — they "found a match" in the optimal accordance from the adjacent set:

$$C_A(L_1, L_2) = \frac{2 \cdot |A_L|}{|L_1| + |L_2|}, \tag{13}$$

where $A_L \equiv A^{opt}_{L_1, L_2}$ is the optimal accordance of markup elements L_1, L_2.

3.2 Tags consistency

This criterion is F-measure for sets of tags of markup elements:

$$C_T(L_1, L_2) = \frac{1}{|A_L|} \sum_{(i,j) \in A_L} \frac{2 \cdot |T_i \cap T_j|}{|T_i| + |T_j|}, \tag{14}$$

where $T_i = \{t^E_1, .., t^E_n\}$ is the set of tags of markup element E_i.

3.3 Overtexts consistency

This criterion calculated as a weighted average of criteria 3.3.1, 3.3.2, 3.3.3, averaged over all pairs from the accordance of markup elements:

$$C_O(L_1, L_2) = \frac{1}{|A_L|} \sum_{(i,j) \in A_L} \sum_{k=1}^{N} w_k \cdot C^O_i(E_i, E_j), \tag{15}$$

where $A_M \equiv A^{opt}_{L_1, L_2}$ is the optimal accordance of markup elements.

3.3.1 Consistency of overtexts accordance

This criterion calculated as F-measure - the proportion of overtexts for which accordance was established — they "found a match" in the optimal accordance from the adjacent set):

$$C^O_1(E_1, E_2) = \frac{2 \cdot |A_O|}{|E^O_1| + |E^O_2|}, \tag{16}$$

where $A_O \equiv A^{opt}_{E^O_1, E^O_2}$ is the optimal accordance of overtexts of markup elements E_1, E_2. E^O_i is the set of overtexts of markup element E_i.

3.3.2 Consistency of overtexts texts

This criterion is the average F-measure of similarity between overtexts texts (as bags of words) of overtexts:

$$C_2^O(E_1, E_2) = \frac{1}{|A_O|} \sum_{(i,j) \in A_O} \frac{2 \cdot |C_i^* \cap C_j^*|}{|C_i^*| + |C_j^*|}, \tag{17}$$

where C_i^* is the representation of overtext comment O_i as a bag of words. Additionally, lemmatization and conversion to lowercase are performed.

3.3.3 Consistency of overtexts tags

This criterion is F-measure for sets of tags of overtexts:

$$C_3^O(E_1, E_2) = \frac{1}{|A_O|} \sum_{(i,j) \in A_O} \frac{2 \cdot |T_i \cap T_j|}{|T_i| + |T_j|}, \tag{18}$$

where $T_i = \{t_1^O, .., t_n^O\}$ is the set of tags of overtext O_i.

3.4 Fragments consistency

This creterion is composite and is calculated as the weighted average of criteria 3.4.1, 3.4.2, 3.4.3, 3.4.4, averaged over all pairs from the accordance of markup elements:

$$C_F(L_1, L_2) = \frac{1}{|A_L|} \sum_{(i,j) \in A_L} \sum_{i=1}^{N} w_k \cdot C_i^F(E_i, E_j), \tag{19}$$

where $A_L \equiv A_{L_1, L_2}^{opt}$ is the optimal accordance of markup elements, $N = 4$ in this case.

3.4.1 Consistency of fragments accordance

This criterion is calculated as F-measure - the proportion of fragments for which accordance was established — they “found a match” in the optimal accordance from the adjacent set:

$$C_1^F(E_1, E_2) = \frac{2 \cdot |A_F|}{|E_1^F| + |E_2^F|}, \tag{20}$$

where E_i^F is the set of fragments of markup element E_i, $A_F \equiv A_{E_1^F, E_2^F}^{opt}$ is the optimal accordance of fragments of markup elements E_1, E_2.

3.4.2 Consistency of fragments texts

This criterion is F-measure for selected text segments - ratio of doubled intersection length to the sum of their lengths:

$$C_2^F(E_1, E_2) = \frac{1}{|A_F|} \sum_{(i,j) \in A_F} \frac{2 \cdot |U_i \cap U_j|}{|U_i| + |U_j|}, \tag{21}$$

where $U_i = [s, f]_i$ is the selected text of fragment F_i.

3.4.3 Consistency of fragments tags

This criterion is F-measure for sets of tags of fragments:

$$C_3^F(E_1, E_2) = \frac{1}{|A_F|} \sum_{(i,j) \in A_F} \frac{2 \cdot |T_i \cap T_j|}{|T_i| + |T_j|}, \tag{22}$$

where $T_i = \{t_1^F, .., t_v^F\}$ is the set of tags of fragment F_i.

3.4.4 Consistency of overtexts fragments

This criterion is computed absolutely analogous to criterion 3.3.

3.5 Consistency of union of fragments texts

This criterion is F-measure for text fragments obtained by merging all fragments of a markup element:

$$C_{UF}(L_1, L_2) = \frac{1}{|A_L|} \sum_{(i,j)\in A_L} \frac{2 \cdot |U_i^* \cap U_j^*|}{|U_i^*| + |U_j^*|}, \tag{23}$$

where U_i^* is the union of texts (selected segments) of fragments of markup element E_i: $U_i^* = [s, f]_1 \cup ... \cup [s, f]_n$.

3.6 Consistency of union of fragments tags

This criterion is F-measure for two sets of tags, each obtained by merging sets of tags of fragments of its markup:

$$C_{UF}(L_1, L_2) = \frac{1}{|A_L|} \sum_{(i,j)\in A_L} \frac{2 \cdot |T_i^* \cap T_j^*|}{|T_i^*| + |T_j^*|}, \tag{24}$$

where $T_i^* = T_1 \cup ... \cup T_n$ is the union of tag sets of markup element E_i.

4 Experiments

The methodology described above was used in practice to evaluate the model built within the winning algorithm of the “READ//ABLE” competition. In the first subsection, we describe the structure of the competition, the markup and fields of the document. In the second subsection we describe how we generalized the described markup above for this competition. In the third subsection, we discuss the obtained results.

4.1 READ//ABLE description

The “READ//ABLE” competition is a technological challenge organized by the National Technology Initiative (NTI) in Russia. Launched in 2019, its goal is to stimulate the development of machine learning approaches capable of creating artificial intelligence systems that deeply understand text meaning and analyze cause-and-effect relationships across a wide range of topics.

The “READ//ABLE” competition is dedicated to the examination of USE essays for five school subjects: history, russian, english, literature and social. The competition’s technological barrier involves developing a robust software system that can identify errors in academic essays, matching the performance of a human specialist within a limited time frame. Participants are tasked with creating intelligent systems that detect errors in essays of up to 12,000 characters, with a processing time of no more than 60 seconds per essay.

In December 2022, the Russian company “Antiplagiat” was declared the winner. Their solution demonstrated a quality level of 100.14% compared to human experts, earning them the prize of 100 million rubles.

The competition has been conducted in multiple cycles, with each cycle comprising qualification and final trials. Additionally, several satellite contests focusing on specific sub-tasks have been held to support teams in developing comprehensive solutions.

In this subsection we want to describe the components of the algorithm, as well as the data structure.

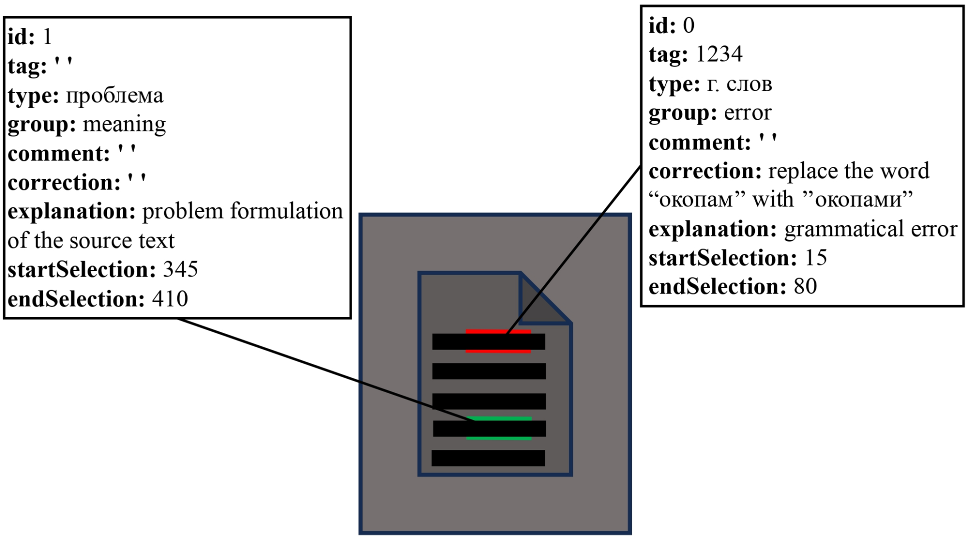

Figure 1: A description of the document markup from the competition “READ//ABLE” after execution of the algorithm.

Field Name	**Description**
id	Unique fragment number
group	“error” or “meaning”; type “error” indicates a localized error, type “meaning” evaluates reasoning blocks
type	Indicates the type of error or meaning block
tag	A string of unique letters or numbers linking related fragments; may be absent if localized
comment	Details the error if not present in the error classifier; otherwise, left empty
correction	Provides a corrected version of the fragment without errors
explanation	A detailed commentary applying to the highlighted fragment
startSelection	Fragment start position
endSelection	Fragment end position

Table 1: Description of fields for annotation of document fragments.

The document before evaluation contains two fields: meta information and essay text. After the essay has been checked, the criteria and document markup fields are added to the document. Fig. 1 shows an example of document markup. Table 1 describes the fields of the markup fragment.

The final grade is automatically calculated from the obtained markup according to the USE criteria.

Segmentable errors can be categorised into four general types — grammatical, speech (inappropriate or redundant use of words in context), logical and factual errors. It is also an additional task to segment the meaning blocks.

The main stages of the Essay Checking System are depicted in Fig. 2. Essay Checking System receives the essay document, when user send it for evaluation. Then it goes to Entrypoint — the main component, that routes the document to the checkers. The Essay checking system consists of five checkers, each of which is responsible for a specific task.

Most of the algorithms were trained on data that was provided by the competition organisers. The data were marked up by the USE experts. The internet was also parsed for essay texts to train unsupervised approaches.

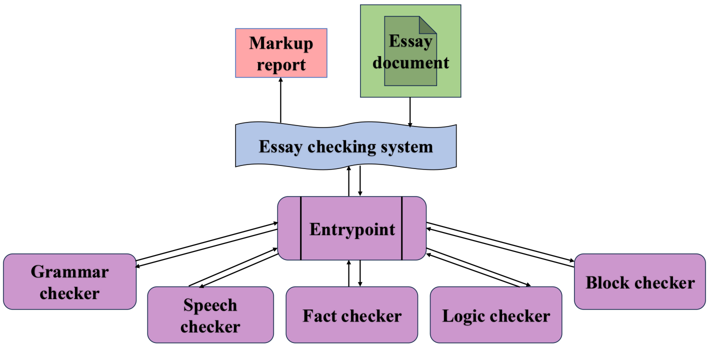

Figure 2: A figure depicting the process of producing document markup. The picture shows the main Entrypoint component, which sends the text of the document to the appropriate checkers to find errors or select meaning blocks, and then aggregates their results.

4.2 Evaluation metrics

The pairwise accuracy $M(X, Y)$ of annotation X relative to annotation Y is calculated as the weighted average of seven metrics $M_1(X, Y), \ldots, M_7(X, Y)$ with weights $w_1, \ldots, w_7$:

$$M(X, Y) = \frac{\sum_{i=1}^{7} w_i M_i(X, Y)}{\sum_{i=1}^{7} w_i}$$

Weights w_i determine the significance of each metric, with $w_i = 0$ excluding a metric.

Essay Score Prediction Accuracy

Measures the match between essay scores derived from annotations X and Y:

$$M_1(X, Y) = \left(1 - \frac{\sum_i |K_i(X) - K_i(Y)|}{\max K}\right) \cdot 100\%$$

Fragment Detection Accuracy and Recall

Evaluates matching fragments in annotations X and Y. Let us introduce a set D of fragment pairs (i, k) such that each $x_i \in X$ corresponds to at most one y_k and each $y_k \in Y$ corresponds to at most one x_i:

$$\text{Precision} = \frac{|D|}{n}, \quad \text{Recall} = \frac{|D|}{m}, \quad M_2(X, Y) = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Code Prediction Accuracy

Proportion of matched fragments in document D with identical codes:

$$M_3(X, Y) = \frac{1}{|D|} \sum_{(i,k) \in D} [\text{type}(x_i) = \text{type}(y_k)]$$

Subtype Prediction Accuracy

Proportion of matched fragments in document D with identical error subtypes:

$$M_4(X, Y) = \frac{1}{|D|} \sum_{(i,k) \in D} [\text{subtype}(x_i) = \text{subtype}(y_k)]$$

Fragment Localization Accuracy

Measures overlap using the Jaccard index:

$$M_5(X,Y) = \frac{1}{|D|} \sum_{(i,k)\in D} \frac{|x_i \cap y_k|}{|x_i \cup y_k|}$$

Correction Accuracy

Proportion of fragments with identical corrections:

$$M_6(X,Y) = \frac{1}{|D|} \sum_{(i,k)\in D} [\text{correction}(x_i) = \text{correction}(y_k)]$$

Explanation Accuracy

The average expert judgement of explanation accuracy across all markup fragments that have explanations. This is the only metric based not on comparison with the markup, but on experts' evaluations. Experts score each explanation in the tested algorithmic markup from 0 to 5 points. The total score is made up of answers to the following questions regarding the given explanation:

1. It is most likely to be understood by the author of the essay.
2. It correctly explains the essence of the error or gives a relevant reference to the source.
3. It leaves no opportunity for appeal.
4. It refers to the text of the work and specifically to the highlighted fragment
5. It solves the pedagogical problem and helps to avoid similar mistakes in the future.

If the examiner considers that the fragment is not an error or does not require an explanation, then it is expected to give zeros in all questions, and the mark for this explanation should be zero. The explanation in the expert markup automatically gets maximum. In order to reduce labour costs, expert checking of explanations is only carried out during the Final Tests.

$$M_7(X,Y) = \frac{\text{Expert Score}}{\text{Maximum Score}} \cdot 100\%$$

Optimistic Accuracy

Optimistic relative pairwise accuracy of the algorithmic markup of a single essay, when compared to the entire set {E} of expert markups of that essay:

$$M_{\text{opt}}(A,\{E\}) = \frac{\max_E M(A,E)}{\min_{E,E'} M(E,E')} \cdot 100\%$$

Average Accuracy

The average relative pairwise accuracy of the algorithmic markup of one essay, when compared to the entire set {E} of expert markups of that essay:

$$M_{\text{avg}}(A,\{E\}) = \frac{\text{avg}_E M(A,E)}{\text{avg}_{E,E'} M(E,E')} \cdot 100\%$$

Overall Relative Accuracy (OTAR)

Combines optimistic and average accuracy using parameter H:

$$\text{OTAR} = \frac{H \cdot \text{avg}_E M(A,E) + (1-H) \cdot \max\limits_E M(A,E)}{H \cdot \text{avg}_{E,E'} M(E,E') + (1-H) \cdot \min\limits_{E,E'} M(E,E')} \cdot 100\%$$

The prerequisite for winning the competition is overcoming the technological barrier of OTAR > 100%.

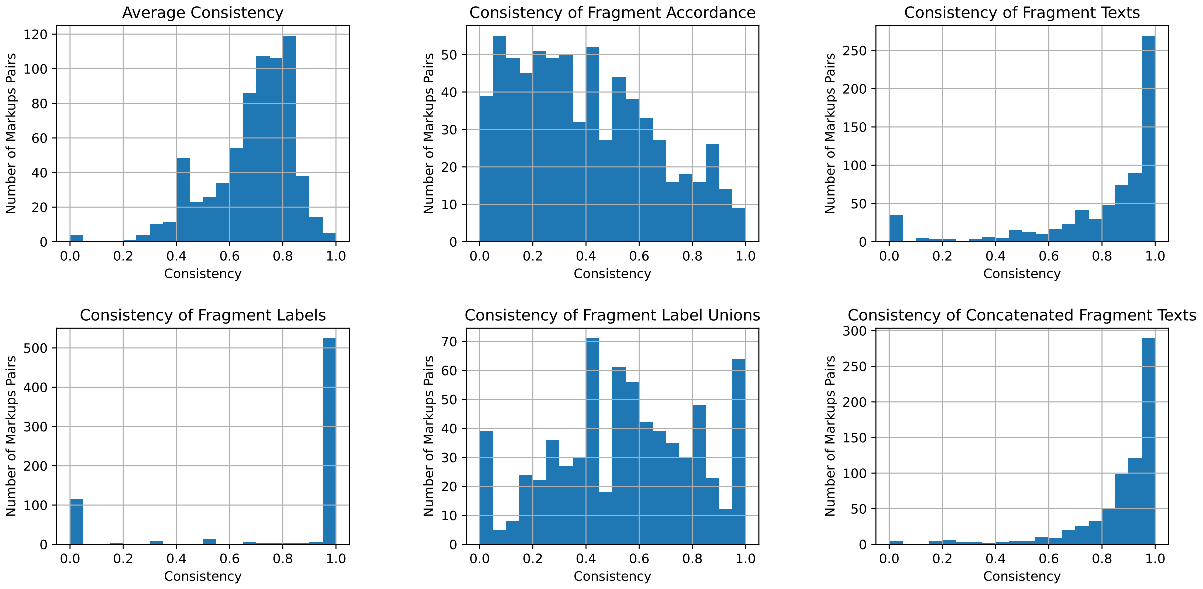

Figure 3: Consistency criteria for pairs of annotations made by annotators

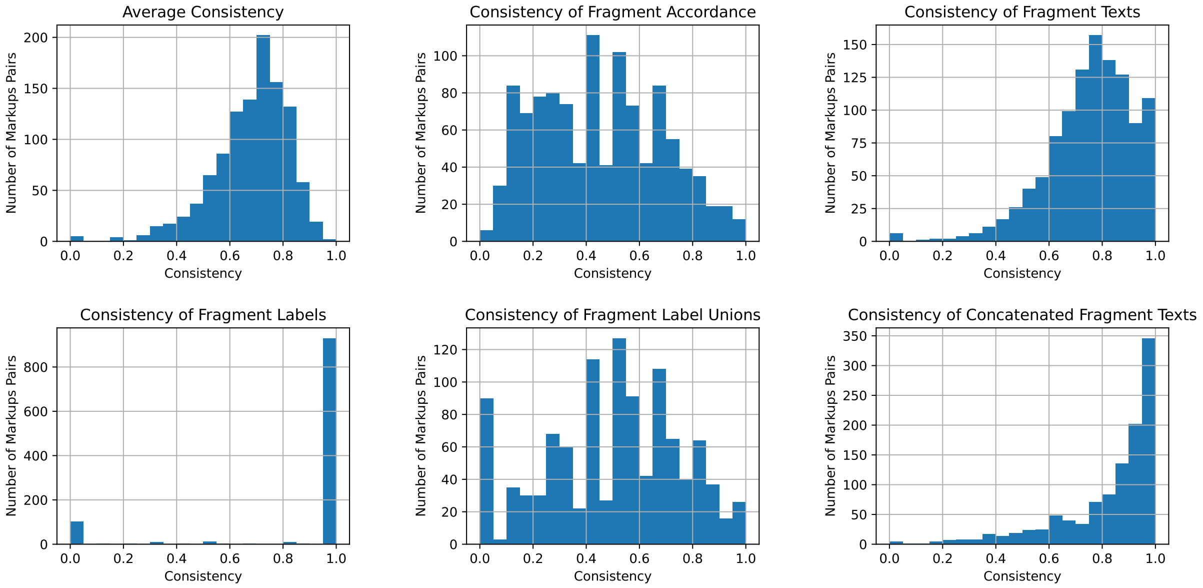

Figure 4: Consistency criteria for pairs of annotations, one made by the algorithm and the other by a human

4.3 READ//ABLE winning algorithm evaluation

The markup structure within the competition was simple, as the markup contained only one markup element, which consisted of fragments with tags, and the overtexts left by the model were not considered since the annotators did not leave them. In section 4.2, we showed a special case of approbation of the developed metrics from section 3. These metrics were applied to validate the competition. The winning system scored an accuracy of 100.14% compared to the average markup of the USE expert — the OTAR metric.

A set of 500 documents was considered, with 1595 annotations of these documents made within the “READ//ABLE” competition. Among them, 1005 annotations were made by annotators, and 500 were

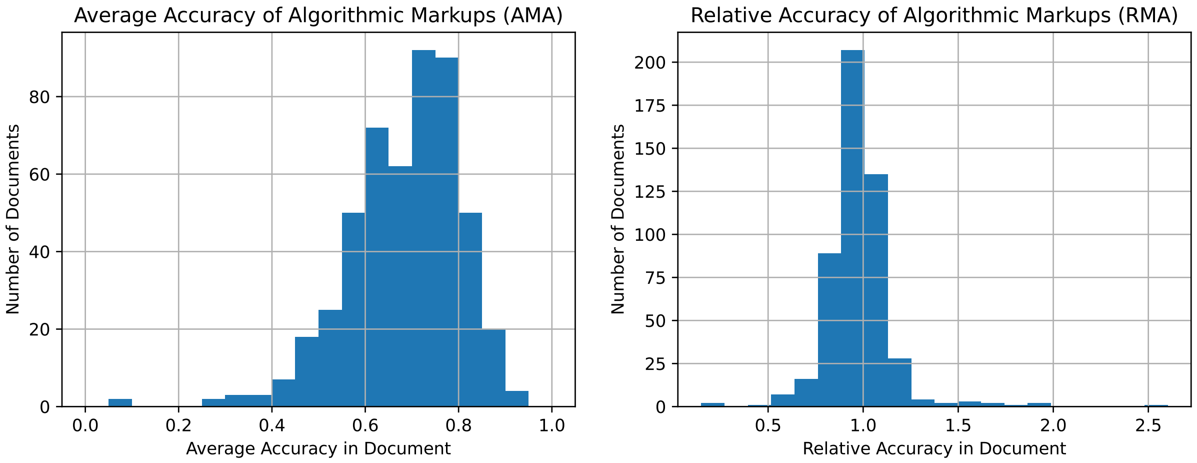

Figure 5: AMA and RMA metrics for the model obtained within the “READ//ABLE” competition

made by the model. Histograms of each of the consistency criteria and their average value are presented in Fig. 3 and Fig. 4. Fig. 5 presents the histogram of AMA and RMA metrics.

4.4 Results discussion

The evaluation of algorithmic markup compared to human annotators reveals several key insights into the model’s consistency, accuracy, and potential limitations. In order to conclude about the consistency of the algorithm with the experts and the experts with each other we examine the histograms in Fig. 3 and Fig. 4.

The histogram for "Consistency of Fragment Texts" shows that human annotations exhibit higher clustering around higher consistency values, whereas the algorithm demonstrates more significant variation. This suggests that while the algorithm is effective, there are cases where it either extracts incorrect fragments or fails to identify certain errors consistently.

Comparing human annotations with the algorithm’s output in terms of fragment tags, we observe that the algorithm achieves a relatively high level of accuracy. The histogram for "Consistency of Fragment Labels" suggests that the algorithm’s tagging process aligns well with human annotators, though minor discrepancies exist. Specifically, human annotators tend to be more rigid in their label selection, while the algorithm exhibits greater variability. One explanation is that the algorithm relies on probabilistic methods or learned patterns rather than strict rule-based tagging. While this allows it to generalize well, it also introduces occasional misclassifications.

The histogram for "Consistency of Fragment Label Unions" indicates that both the algorithm and human annotators demonstrate significant variability. It suggests that the algorithm performs on par with human annotators, indicating that its generalization capability is relatively strong. This is an encouraging result because it demonstrates that even though the algorithm is not perfect at fragment identification on a case-by-case basis, its overall trend aligns with human judgments.

In the "Consistency of Fragment Accordance" histogram, we notice a relatively wide distribution, with many of instances having low consistency values. This suggests that even human annotators exhibit notable differences in how they mark fragments, meaning that essay annotation is inherently subjective. This result emphasizes the need to use consistency metrics in markup tasks, since in the case of standard metrics, it is not obvious what to use as ground truth.

In Fig. 5 the RMA histogram shows that the model’s accuracy is centered around 1, meaning that it agrees with human annotators on average as much as they agree with each other. However, the presence of some extreme cases where the RMA deviates significantly suggests that there are specific instances where the algorithm either outperforms or underperforms compared to human annotators. In cases where

RMA > 1, the algorithm likely follows more rigid, rule-based logic that leads to greater consistency than human annotators, who may be influenced by subjectivity. In cases where RMA < 1, the algorithm struggles with contextual nuances that humans naturally interpret more accurately.

Fig. 5 shows that the AMA values are concentrated around higher accuracy levels, there is still some distribution toward lower values, indicating cases where the algorithm struggles. These outliers likely correspond to edge cases where human judgment plays a significant role, such as ambiguous errors or unconventional phrasing in essays.

The results show that the algorithm achieves human-comparable annotation accuracy, though inconsistencies in fragment selection highlight the subjectivity of human markup. While the model performs well in structured error detection, it struggles with context-sensitive cases and hierarchical relationships between errors. Future improvements should focus on refining contextual understanding and integrating expert feedback to enhance annotation consistency. We also see the need to use consistency metrics as essay evaluation is very subjective, and if standard NLP metrics are used, it is not clear what counts as true.

5 Conclusion

Evaluating the quality of textual markup in tasks involving subjective and context-dependent annotations remains a significant challenge in Natural Language Processing. Standard evaluation metrics often fail to capture the nuanced differences between human and algorithmic annotation, especially when multiple valid interpretations are possible. This is crucial in such applied tasks as essay evaluation, personalized writing feedback, grammar and style correction, and intelligent tutoring systems.

In this study, we introduced a multi-criteria evaluation method for assessing markup consistency and quality, which allows for detailed comparison between human annotators and automated algorithms, taking into account all the features of its generalized structure. This method was applied to the "READ//ABLE" competition, providing valuable insights into the strengths and weaknesses of the evaluated language model.

Our analysis revealed that the algorithm demonstrates quality comparable to human annotators in fragment tagging consistency and overall markup cohesion. However, the model exhibited lower performance in fragment text consistency, suggesting that while it excels in systematic and structural tasks, there is room for improvement in handling the subtleties of text extraction.

The possibility of including/excluding additional criteria and changing the weighting coefficients ensures the adaptability of the evaluation method to a wide range of markup structures and tasks. This makes it a robust tool for assessing algorithms in various Natural Language Processing problems.

References

Mashary Alrasheedy, Ravie Muniyandi, and Fariza Fauzi. 2022. Text-based emotion detection and applications: A literature review. P 1–9, 10.

Michael K. Buckland and Fredric C. Gey. 1994. The relationship between recall and precision. *J. Am. Soc. Inf. Sci.*, 45:12–19.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. // Jill Burstein, Christy Doran, and Thamar Solorio, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, P 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Besnik Fetahu, Zhiyu Chen, Sudipta Kar, Oleg Rokhlenko, and Shervin Malmasi. 2023. MultiCoNER v2: a large multilingual dataset for fine-grained and noisy named entity recognition. // Houda Bouamor, Juan Pino, and Kalika Bali, *Findings of the Association for Computational Linguistics: EMNLP 2023*, P 2027–2051, Singapore, December. Association for Computational Linguistics.

Denis Gordeev, Adis Davletov, A. Rey, G. Akzhigitova, and G. Geymbukh. 2020. Relation extraction dataset for the russian. P 348–360, 01.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *CoRR*, abs/2006.03654.

Naotaka Kawata and Genichiro Kikui. 2019. Mention detection method for entity linking. // *2019 IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD)*, P 41–45.

I. A. Khabutdinov, A. V. Chashchin, A. V. Grabovoy, A. S. Kildyakov, and U. V. Chekhovich. 2024. Rugector: Rule-based neural network model for russian language grammatical error correction. *Program. Comput. Softw.*, 50(4):315–321, July.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. // *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, P 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. cite arxiv:1907.11692.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022. MultiCoNER: A large-scale multilingual dataset for complex named entity recognition. // Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na, *Proceedings of the 29th International Conference on Computational Linguistics*, P 3798–3809, Gyeongju, Republic of Korea, October. International Committee on Computational Linguistics.

Andrew McCallum and Kamal Nigam. 1998. A comparison of event models for naive bayes text classification. // *AAAI Conference on Artificial Intelligence*.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR – grammatical error correction: Tag, not rewrite. // Jill Burstein, Ekaterina Kochmar, Claudia Leacock, Nitin Madnani, Ildikó Pilán, Helen Yannakoudakis, and Torsten Zesch, *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, P 163–170, Seattle, WA, USA → Online, July. Association for Computational Linguistics.

Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. 2011. Finding deceptive opinion spam by any stretch of the imagination. // *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, P 309–319, USA. Association for Computational Linguistics.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. // *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, P 79–86. Association for Computational Linguistics, July.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. // *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11.

Wei Shi and Vera Demberg. 2019. Next sentence prediction helps implicit discourse relation classification within and across domains. // Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, P 5790–5796, Hong Kong, China, November. Association for Computational Linguistics.

Julius Sim and Chris C Wright. 2005. The kappa statistic in reliability studies: Use, interpretation, and sample size requirements. *Physical Therapy*, 85(3):257–268, 03.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. // *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, P 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference.

Janyce Wiebe. 2000. Learning subjective adjectives from corpora. // *AAAI/IAAI*.

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. // *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, P 1253–1256, New York, NY, USA. Association for Computing Machinery.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019a. End-to-end open-domain question answering with BERTserini. // Waleed Ammar, Annie Louis, and Nasrin Mostafazadeh, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, P 72–77, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le, 2019b. *XLNet: generalized autoregressive pretraining for language understanding*. Curran Associates Inc., Red Hook, NY, USA.