LLMs Out-of-the-Box Do Not Generate Context-Appropriate Word Order in Russian

Alina Shabaeva[♠♦], **John Frederick Bailyn**[♠], **Owen Rambow**[♠♦]
[♠]Department of Linguistics,
[♦]Institute for Advanced Computational Science
Stony Brook University
alina.shabaeva@stonybrook.edu

Abstract

In this paper, we examine whether LLMs can generate context-appropriate word order in Russian. Using a new corpus of movie scripts in Russian, with a particular focus on dialogues, we investigate how various semantic, morphosyntactic, and discourse features influence word order choice. We show that traditional machine learning can use these features to model different word order fairly well. We also examine whether LLM dialogue partners can generate context-appropriate word order in Russian. With both zero- and few-shot prompting, LLMs fail to generate word orders beyond the default subject-verb-object. Instead, in order to generate context-appropriate word orders, LLMs need to be fine-tuned or given explicit word order suggestions from a traditional machine learning method.

1 Introduction

Information Structure (IS) focuses on how speakers organize and present information within an utterance based on what they assume the listener knows (common ground), and what information will be new to them. It plays a crucial role in shaping sentence form, guiding how linguistic elements reflect shared knowledge, discourse assumptions, and the speaker's assessment of their interlocutor's mental state. In particular, word order variation often encodes how speakers evaluate their interlocutor's mental representation of the conversation, what they are familiar with and what is new information to them. IS is therefore closely tied to Theory of Mind (ToM) – the ability to infer others' cognitive states, such as beliefs (Premack and Woodruff, 1978; Apperly, 2010). To our knowledge, there has been no systematic investigation of whether LLMs can generate language which correctly reflects IS as it evolves in discourse.

Russian is a particularly interesting language for studying how syntax interacts with IS, as it encodes IS distinctions through syntactic reordering rather than prosody or morphology alone. For example, the simple transitive sentence *Маша читает книгу* (*Masha čitaet knigu*) 'Masha reads a book' can appear in all six permutations of S (subject), V (verb), and O (object): SVO, SOV, OSV, OVS, VSO, VOS. While having the same propositional meaning, these permutations are not necessarily appropriate in the same context (see an example in Appendix A).

The contributions of this paper are the following:

(1) Corpus-driven computational analysis and annotation: We introduce an annotated corpus of Russian movie scripts with extracted and computed linguistic features (Section 3), along with statistical validation of their significance (Section 4).

(2) Predictive modeling with machine learning: We conduct machine learning experiments in order to demonstrate that our features are effective in predicting different syntactic patterns (Section 5).

(3) Evaluation of LLMs: We show that LLMs display a strong bias toward canonical SVO word order, meaning that they do not exploit the discourse context and linguistic clues contained in it in word order choice. Using fine-tuning or explicit prompting can partially overcome this problem (Section 6).

2 Related Work

The Prague School made significant contributions in theorizing information structure (IS), introducing the concepts of functional sentence perspective, theme-rheme configuration, and communicative dynamism (Mathesius, 1939; Firbas, 1964).

Prince's fundamental analyses (Prince, 1981) likewise explain word order variation in terms of IS, distinguishing topic/focus from given/new information and showing how sentence form encodes discourse status.

In Russian, word order also has been widely explored, (1) within functionalist approaches (Kov-

Proceedings of the 27th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 453–472
August 2–5, 2026.

tunova, 1976; Krylova and Khavronina, 1986), which highlight the communicative function of sentence structure and its reflection of the speaker's intent, and (2) within generative syntactic frameworks, which explain permutations through syntactic configurations involving topic-focus projections (Bailyn, 1995, 2012).

While IS has been extensively examined from a theoretical point of view, computational modeling of this phenomenon remains relatively limited. For Czech, Hajič et al. (2020) present one of the foundational attempts to automate IS analysis, providing corpora annotated for theme-rheme distinctions and algorithms modeling discourse-driven constituent placement. For English and German, De Kuthy et al. (2018) develop a QUD-based annotation framework that models discourse and information structure, focusing on topic and focus identification. Moving beyond traditional hand-crafted features, Hou (2020) demonstrates that discourse context-aware BERT models could outperform state-of-the-art models in IS classification task, showing the benefits of neural approaches for capturing discourse information. This prior work focuses primarily on automating IS annotation and classification, while we show how various linguistic features (not limited to IS) affect word order variation.

Linearization has been extensively studied in natural language generation. Some approaches address this problem using only morphosyntactic information, without IS cues. For example, Bangalore and Rambow (2000) combine tree models with language models, and Bohnet et al. (2012) generate non-projective word orders across six typologically different languages. Cahill and Riester (2009) extend this work by incorporating information status features into generation ranking, while Zarrieß et al. (2012) investigate to what extent sentence-internal word order in German actually reflects discourse context, finding that external contextual features play a surprisingly limited role due to overlapping information within the target sentence. Additionally, the task has been included in shared tasks, in particular the Multilingual Surface Realization Shared Task (Mille et al., 2019), which defines shallow and deep tracks for linearization dependency structures.

While these approaches address the linearization task through structural or feature-based methods, we investigate whether LLMs can generate word order variation consistent with patterns observed in our corpus. More recent work on English constituent variation (Tur et al., 2025) compares several features (word length, token length, modifier weight, and syllable weight), showing how these factors influence surface order and contribute to syntactic flexibility. We adopt and extend this feature-based approach, and evaluate it on Russian, a language with flexible word order. To the best of our knowledge, no equivalent computational studies exist for Russian. And, crucially, there has been no systematic assessment whether LLMs can generate context-appropriate word order in any free word order languages.

3 Corpus

3.1 Corpus Preparation

In this study, our corpus is based on the Movie Scripts dataset[1] (Lisovsky, ca. 2019), which contains 20 preprocessed Russian text files. We chose this dataset because it provides relatively modern dialogues. A potential concern is that the scripts are translations of American films (e.g., *Reservoir Dogs*, *Pulp Fiction*, *Fight Club*), rather than texts originally written in Russian. However, the observed distribution of word order patterns closely matches findings from native Russian corpora (see Section 4), suggesting that translation does not affect the scripts' syntactic naturalness. Furthermore, this corpus consists of dialogue, and we could locate no other suitable dialogue corpora.

From the Movie Scripts dataset, we selected a subset of 10 scripts for manual annotation. These ten scripts were chosen arbitrarily rather than based on specific selection criteria. The resulting subset covers a range of genres (crime, comedy, drama, and thriller) and release years (1992-2012). The remaining ten scrips in the original dataset are comparable in length and were excluded purely due to practical reasons (cost of annotation). Including them in the corpus is left for future work.

Because of the specific genre of movie scripts, we annotated the corpus for content type (not for information structure). We created an annotation manual, and two independent annotators[2] along with the first author labeled the scripts using the INCEpTION tool (Klie et al., 2018). The annotation distinguishes four types of content: *speech* (what the characters say), *character name* (who is

[1] https://www.kaggle.com/datasets/lisovskey/moviescripts

[2] The annotators were recruited through word-of-mouth among Russian native speakers. We paid the equivalent of $20-$30 per hour depending on education level.

speaking), *comment* (information perceptible to the viewer and in most cases also the characters, e.g. visual and auditory cues), and *camera* (screen directions, e.g. *black screen* or *close-up*). Distinguishing between these categories is important because we wish to track what the characters say and perceive. The reason this task could not be done automatically is that each script has its own conventions, but each script can be annotated easily by humans. We have conducted an inter-annotator agreement evaluation by comparing 2133 tokens annotated by one of the external annotators and the first author (who also served as annotator for this corpus). The resulting Cohen's kappa is 0.87 with 92.6% agreement, indicating a high level of consistency between annotators.

We then used the Stanza NLP toolkit for morphosyntactic processing. Prior studies have demonstrated that Stanza achieves higher parsing accuracy than SpaCy and other pipelines across various languages (Qi et al., 2020), which we confirmed for our corpus as well.

We extracted sentences with transitive verbs, an overt nominative subject, and an overt accusative object. To maintain consistency, we retained only declarative sentences, filtering out questions, imperatives, and subjunctives (detected by Stanza). Additionally, we excluded sentences with verbs *звать* (*zvat'*) 'to call', *интересовать* (*interesovat'*) 'to interest', and *иметь* (*imet'*) 'to have', which involve quirky subjects[3] or idiomatic usage. To focus on dialogues, we restricted extraction to sentences labeled as speech. In total, we extracted 1,543 sentences that met our criteria.

3.2 Feature Extraction

From these extracted sentences we constructed a structured dataset. First, we derived **word order** information for both matrix and embedded clauses using dependency relations from Stanza. To test the accuracy of the parser, we manually annotated 100 sentences and found 99% agreement between the automatically assigned labels and the gold standard. We also added the preceding **context**, defined as the two most recent speech-labeled sentences before the target sentence. Camera directions and character names were excluded, as these are not accessible to the characters as part of their context, nor to the viewers. However, we retained comment-labeled sentences if they occurred between the two speech sentences or between the last speech sentence and the target, since they usually convey information available to the characters. We ran a grid search predicting word order from multiple features, including features that draw on the context, and the two-sentence context gave the best results. This finding aligns with text comprehension models that assume a limited working-memory buffer maintaining only the most recent information (Kintsch and van Dijk, 1978).

Then, for each target sentence, we extracted the head, head lemma, and full constituent of the subject, of the object, and of the verb, and using both SpaCy or Stanza assigned further morphosyntactic features. We obtained semantic features (verb type and semantic roles of subject and object) by prompting the OpenAI GPT-3.5-turbo model. The complete list of features is as follows:

Morphosyntactic (previously discussed in (Zuban et al., 2021; Slioussar and Makarchuk, 2022; Seržant et al., 2025)):

- Part of speech (POS): *noun, pronoun, verb, etc.*
- Constituent sizes (number of tokens) for subject, object, and verb
- Verb aspect (*perfective* / *imperfective*)
- Presence of negation for the verb
- Clause type (*matrix* / *embedded* clause)

Phonological:

- Syllable weight: the number of syllables in subject, object, and verb constituents. Prior work has shown that it is a strong factor in predicting constituent shifting behavior in LLMs (Tur et al., 2025).

Semantic (previously discussed in (Titov, 2017; Laleko, 2022; Slioussar and Harchevnik, 2024)):

- Subject and object named entity types (extracted by Stanza): `PER` (Person), `ORG` (Organization), `LOC` (Location), `MISC` (Miscellaneous)
- Animacy of subject and object (see Appendix B for details).
- Verb type (*perception*, *communication*, *action*, *causative*, *psychological*, *motion*). This list was created according to general categories following (Levin, 1993).
- Predicate type (*2-place*, *3-place*)
- Semantic roles of subject and object (*Experiencer*, *Agent*, *Causer*, *Theme*, *Stimulus*, *Recipient*, *Patient*, *Beneficiary*). This list reflects the standard thematic roles discussed in the linguistic literature (Jackendoff, 1972).

Discourse:

- Discourse-related elements
- The coreference score

[3]Subjects in non-nominative case (dative, accusative, or genitive) that still syntactically behave as subjects.

• Is the target sentence preceded by a question?

The first two discourse features did not turn out to be useful for this particular dataset in predicting word order. First, we used overt discourse-related elements to detect focused or topicalized constituents (see Appendix C for details). Then, we developed a unitary accessibility score for subject and object separately: pronouns and personal names are treated maximally accessible by default, while the score for full noun phrases is computed based on the semantic distance between the target and possible antecedents in the preceding context, if present. Therefore, this score serves as a single measure of how strongly a constituent (subject or object) refers to the prior discourse. We found that it can be used to characterize different word orders very clearly. However, an ablation study showed that removing this score does not hurt word order prediction (plausibly, this is due to overlap with other features, such as part of speech). We conclude that while the accessibility score captures intuition about the contribution of context on word order choice, and allows for a clear presentation for human consumption, it is not crucial for machine learning (see Appendix D for further details).

4 Corpus Analysis

Our corpus contains examples of all six possible word orders. Table 1 presents their distribution, both individually and grouped, providing a foundation for further analysis. SVO occurs most frequently, followed by SOV, OSV, and OVS. The two remaining orders, VSO and VOS, appear only rarely. This distribution is consistent with previous studies of word order variation in Russian corpora (Billings, 2015; Kallestinova, 2007). Given the very low frequency of verb-initial patterns, we excluded them from further analysis unless stated otherwise.

Word Order	Count	Percentage	Binary
SVO	1123	72.8%	1123 (72.8%)
SOV	275	17.8%	
OSV	71	4.6%	
OVS	50	3.2%	420 (27.2%)
VSO	14	0.9%	
VOS	10	0.7%	

Table 1: Word order distribution in the corpus, with a binary column showing totals for SVO vs. all non-SVO patterns.

One of the most significant factors influencing word order choice is whether the sentence arguments are nominal or pronominal (Sirotinina, 1965; Slioussar and Makarchuk, 2022). Table 2 presents a detailed overview of subject and object **pronominality** across the four most frequent word orders in the corpus. SVO, the most frequent order, typically occurs with a pronominal subject (56.5%) or with both arguments pronominalized (23.4%). A similar tendency is observed in OVS, except that here it is the preverbal object that is pronominalized (62%). This distribution aligns with information structure theory: sentences tend to begin with given information, and pronouns typically refer to previously mentioned entities (Gundel et al., 1993; Lambrecht, 1994; Bailyn, 2012). In contrast, SOV is skewed toward having both arguments pronominalized (78.9%), while OSV is dominated by subject pronouns as is SVO (66.2%), indicating that this order involves distinct context requirements from the other cases of pre-verbal objects. Interestingly, both verb-final patterns almost never occur without at least one pronominal argument. Word order and pronominalization are significantly correlated according to the chi-square test, $\chi^2(9) = 569.3, p < .001$. We present a similar analysis for another important feature, **animacy** (see Appendix B for more detailed information).

WO	None	Subject only	Object only	Both
SVO	164 (14.6%)	634 (56.5%)	62 (5.5%)	263 (23.4%)
SOV	1 (0.4%)	30 (10.9%)	27 (9.8%)	217 (78.9%)
OSV	0 (0.0%)	47 (66.2%)	1 (1.4%)	23 (32.4%)
OVS	6 (12.0%)	5 (10.0%)	31 (62.0%)	8 (16.0%)

Table 2: Pronoun distribution across four word orders in the corpus.

5 Predicting Word Order

The objective of these experiments is to assess whether word order can be *predicted* based on the proposed features and to determine which of them have the greatest impact on the classification using machine learning methods. In our preliminary experiments, we tested four standard classifiers: Logistic Regression, Decision Tree, Random Forest, and XGBoost (Chen and Guestrin, 2016). Additionally, we experimented with multilayer perceptron (MLP) architectures with 1-3 hidden layers (from 64-32 to 256-128-64 units), with tuning details and architectural configurations summarized in Table 18 in Appendix E.

We removed features that did not show a statistically significant correlation with word order based

on a chi-square test: verb aspect, preceding question (which does not occur in our corpus), clause type, and focused object.

We excluded verb-initial sentences due to their low frequency. Thus, the resulting subset of the corpus used in these experiments consists of 1519 sentences. We split this subset randomly into training, development, and test sets with a 60/20/20 ratio. Table 3 shows the distribution of word orders across the splits. Note that the frequencies are remarkably similar in the three subsets. For standard classifiers,

Word Order	Train	Dev	Test
SVO	673 (73.9%)	225 (74.0%)	225 (74.0%)
SOV	165 (18.1%)	55 (18.1%)	55 (18.1%)
OSV	43 (4.7%)	14 (4.6%)	14 (4.6%)
OVS	30 (3.3%)	10 (3.3%)	10 (3.3%)
Total	**911**	**304**	**304**

Table 3: Distribution of word order classes across train, development, and test splits.

we determined hyperparameters using grid search with cross-validation on the training set. We trained the models without resampling and included in the evaluation reports per-class, macro- and weighted F1 scores.

5.1 Multiclass Classification: Predicting Specific Word Order

First, we conducted a multiclass classification experiment in which the objective was to predict one of the four possible word orders.

Table 4 reports the accuracy and the macro- and weighted F1 scores for our four classifiers and the best-performed MLP for the multiclassification task, compared against two baselines: (1) a majority baseline that always predicts the most frequent word order (SVO); and (2) a stratified random baseline (proportional to class distribution).

Model	Accuracy	Macro-F1	Weighted-F1
Majority Baseline	0.74	0.21	0.63
Random Baseline	0.61	0.28	0.61
LogReg	0.76	0.40	0.73
Decision Tree	**0.78**	**0.50**	**0.78**
Random Forest	0.77	0.46	0.74
XGBoost	0.77	0.47	0.75
3-layer-256-128-64 (MLP)	0.74	0.41	0.70

Table 4: Comparison of models' performance for the multiclass classification task on the test set.

Overall, the results are very similar across the models. However, the Decision Tree classifier performs slightly better. The detailed results of this model on the dev and test sets are provided in Table 17 in Appendix E. While it is not surprising that the model performs well on the dominant SVO class, it also captures meaningful signals for less frequent word orders. Although performance on the rare SOV and OSV classes remains limited on the test set (SOV: F1-score=0.63, OSV: F1-score=0.33), it outperforms baselines, indicating that the model learns from the linguistic features.

Table 18 in Appendix E presents a comparison of different neural architectures for the multiclass classification task. The detailed results of the best-performed model (3-layer-256-128-64) are provided in Table 19 in Appendix E. While the neural model outperforms both the majority and random baselines in accuracy and weighted-F1 scores, overall, it performs worse than standard classifiers. Notably, it completely fails to predict the rarest word orders (OSV and OVS). Plausibly, this is due to the limited amount of training data, which makes it challenging for the neural model to learn less frequent word order patterns.

5.2 Binary Classification: Canonical or Not

Second, we run a binary classification experiment in which the models had to distinguish between the canonical SVO order and all other patterns combined Table 5 shows the performance of our four standard classifiers and the best MLP architecture (2-layer-128-64) on the binary classification task.

Model	Accuracy	Macro-F1	Weighted-F1
Majority Baseline	0.74	0.43	0.63
Random Baseline	0.62	0.50	0.61
LogReg	0.77	0.65	0.75
Decision Tree	0.79	**0.74**	**0.80**
Random Forest	0.79	0.71	0.78
XGBoost	**0.80**	0.72	0.79
2-layer-128-64 (MLP)	0.76	0.64	0.74

Table 5: Comparison of models' performance for the binary classification task (SVO vs. non-SVO) on the test set.

As expected, results are higher than for multiclass classification, for both Macro-F1 and Weighted-F1. The Decision Tree classifier and XGBoost achieve very similar results. However, given slightly higher Macro-F1 score of the Decision Tree. See Appendix F for further discussion of these results.

5.3 Model Interpretation

In addition to solid performance, the Decision Tree classifier has the advantage of interpretability:

model behavior can be analyzed though feature importance, which indicate how strongly each feature influence the model's predictions. Table 6 shows the top ten features ranked by importance for both multiclass and binary classification tasks.

Feature	Imp.	Feature	Imp.
object_pos	0.47	object_pos	0.61
coref_score_object	0.09	predicate_type	0.10
obj_animacy	0.09	coref_score_object	0.06
predicate_type	0.08	has_negation	0.05
object_syllable_weight	0.05	verb_constituent_size	0.04
verb_constituent_size	0.05	object_syllable_weight	0.03
subject_role	0.04	object_ner	0.02
coref_score_subject	0.03	subject_role	0.02
object_ner	0.03	focus_subject	0.02
subject_syllable_weight	0.02	verb_type	0.01
Multiclass		**Binary**	

Table 6: Top-10 most important features identified by the Decision Tree classifer for the multiclass (left) and binary (right) classification tasks.

The analysis shows that in both tasks, the object's *part of speech* (POS) dominates, emphasizing its key role in determining word order. This is likely due to the differences between pronominal and nominal objects, which has been shown to correlate with word order variation. Additionally, according to Seržant et al. (2025), pronouns encode highly accessible referents that are already active in discourse or easily retrievable from memory, and therefore, tend to occur in preverbal position. Other features, such as *predicate type*, *object syllable weight*, and *object coreference score*, appear among the most important features in both tasks. Consequently, the models are capturing meaningful signals associated with a linguistically motivated set of features.

We emphasize that the main goal of these ML experiments was not only to evaluate the performance of different models, but to assess whether the proposed set of features captures meaningful signals. The results indicate that it does. While performance is not perfect, the models nevertheless learn patterns even for rare non-canonical word orders. Additionally, the feature importance aligns with linguistic expectations.

6 LLMs and Word Order Variation

6.1 Model Selection

For our final evaluations, we used GPT-3.5 Turbo (Brown et al., 2020) GPT-4o and GPT-4o-mini (OpenAI, 2024), and GPT-5-mini (OpenAI, 2025), as they consistently generated well-structured content that followed our instructions. We also tested alternatives such as LLaMA (Dubey et al., 2024) and mT5 (Xue et al., 2020), but their outputs were often incoherent. Additionally, we experimented with Russian LLMs, Saiga (Gusev, 2023) and RuGPT-3 (AI Forever, 2022). However, despite being trained on Russian corpora, both models performed poorly in our controlled text generation task, resulting in context copying and failure to follow instructions.

6.2 Prompt Design

We designed a prompt that required the model to continue a Russian movie script. Each prompt is based on a specific occurrence of a verb in the script. The LLM was instructed to generate exactly one subsequent sentence based on the given context (we use the same notion of context as described in Section 3). This new sentence was required to have a specified verb with a predefined aspect, tense and negation (if applicable), a provided subject constituent in the nominative case, and a provided object constituent in the accusative case. The clause type (matrix or embedded) in which these elements had to occur was provided too. The prompt emphasized that the generated sentence must be in the indicative mood, grammatically correct, and contextually appropriate. Additionally, we noted that although SVO is a canonical word order, other permutations are also possible in Russian. The exact prompt template used in our experiments is provided in Appendix G. All experiments described in this section use a zero-shot prompting setup, unless stated otherwise. The word order of the sentences generated by the LLMs was determined using Stanza.

We empirically determined that a temperature setting of 0.7 provides the optimal balance between diversity and consistency. Then, we selected GPT-3.5-turbo due to its low cost and evaluated the full corpus of 1543 sentences across multiple runs using the same prompt. To assess how many runs for each example are necessary, we compared the consistency scores, defined as the proportion of repeated generations assigned the same word order by the parser, for randomly sampled subsets of 3, 5 and all 10 runs per each sentence (see Table 7). Although five runs result in slightly higher variability compared to three runs, the difference is modest. Therefore, given the higher cost of ten runs and the minimal difference between three and five runs, we use three runs in all subsequent experiments, which

Number of runs	Mean	Std.	Min	Max
3	0.98	0.06	0.72	1.00
5	0.97	0.08	0.64	1.00
10	0.97	0.1	0.50	1.00

Table 7: Consistency measures for different numbers of sampled runs (on the entire corpus).

allows us to evaluate more data within the budget.

6.3 Model Evaluation and Error Analysis

Using the prompt discussed above (Appendix G), we evaluated three models on development and test sets (as in Section 5) by running each model three times with the same prompt. We use this setup as our baseline.

In the LLM setting, the objective is not only to evaluate sentence-level consistency with the word order in the reference corpus (as in classification tasks), but also to investigate whether the model's output reflects the word order distribution observed in the corpus. Therefore, we assess model performance using both word order distributions (which captures general LLM behavior) and sentence-level accuracy (which reflects per-sentence correctness). Importantly, matching the overall distribution does not necessarily entail high accuracy, and vice versa.

Across all experiments, some instances could not be evaluated for word order (*unknown* sentences) due to (1) generating incoherent output, and (2) failures to follow the instructions despite producing well-formed sentences. We manually inspected the latter error type and observed that these sentences still consistently follow an SVO pattern, without any variation. A detailed error analysis of GPT-4o *unknown* outputs, based on a sample of 50 instances (from the entire corpus experiment), is provided in Table 10. Obviously, the gold standard has no such sentences.

Detailed results for each model compared to the word order distribution in the corpus (gold) are shown in Table 8. All models demonstrate a strong preference for SVO pattern (around 84-90%), with minimal word order variation compared to the gold annotations. While the gold standard has 18.1% SOV sentences, no model has more than 6.4%, and no model generates any examples of OSV (4.6% in gold) or OVS (3.3% in gold). Furthermore, all models generate a high rate of of *unknown* sentences (labeled as *UNK*), between 7.7% and 9.4%. While GPT-3.5-turbo generates the highest percentage of non-canonical word orders, the more recent GPT-4o has the lowest percentage of *unknown* sentences. Additionally, GPT-4o achieves the highest accuracy compared to the gold annotations (see Table 9). Detailed per-word-order accuracy results are provided in Table 26 in Appendix J.

WO	Gold		GPT-3.5-turbo		GPT-4o-mini		GPT-4o	
	Count	%	Count	%	Count	%	Count	%
Development Set								
SVO	225	74.0%	733	80.4%	819	89.8%	777	85.2%
SOV	55	18.1%	58	6.4%	7	0.8%	55	6.0%
OSV	14	4.6%	–	–	–	–	–	–
OVS	10	3.3%	–	–	–	–	–	–
VOS	–	–	3	0.3%	–	–	–	–
VSO	–	–	1	0.1%	–	–	–	–
UNK	–	–	117	12.8%	86	9.4%	80	8.8%
Test Set								
SVO	225	74.0%	762	**83.6%**	822	90.1%	788	86.4%
SOV	55	18.1%	58	**6.4%**	4	0.4%	53	5.8%
OSV	14	4.6%	4	0.4%	1	0.1%	–	–
OVS	10	3.3%	–	–	–	–	–	–
VSO	–	–	2	0.2%	–	–	–	–
VOS	–	–	–	–	–	–	1	0.1%
UNK	–	–	86	9.4%	85	9.3%	70	**7.7%**

Table 8: Word order distributions for gold annotations and model outputs on the development and test sets, with counts and percentages separated.

Split	GPT-3.5	GPT-4o-mini	GPT-4o
Dev	65.6%	67.0%	**68.6%**
Test	67.9%	69.3%	**70.2%**

Table 9: Overall accuracy on the development and test sets.

Since we do not need training data for the zero-shot LLM experiments, we also separately analyzed model predictions across the entire corpus (1543 sentences) to understand general model behavior (see Table 23 in Appendix H). All models consistently show a strong preference for the canonical SVO order, generating it in more than 90% cases, as opposed to only 72.8% of cases in the corpus. Non-canonical orders are rare or absent.

We analyze errors (*unknown* sentences labeled as *UNK*) produced by GPT-4o in Table 10. We use GPT-4o as a representative LLM, errors observed in other models across all experiments fall into the same categories.

Furthermore, we experimented with GPT-5-mini across all four levels of reasoning. However, we limited the evaluation to the first 100 sentences in the corpus due to the model's higher cost. As shown in Table 25 in Appendix I, across all reasoning levels, GPT-5-mini generates mostly canon-

Error Type	Description	Count	%
Incoherent Output	Semantically incoherent or morphosyntactic errors.	18	36%
Argument Structure Mismatch	Subject, object, or verb mismatches with the prompt requirements.	22	44%
Morphosyntactic Mismatch	Morphosyntactic mismatches with the prompt requirements (e.g., aspect, mood, clause type, negation).	7	14%
Context Copying Error	Model copies context from the prompt instead of generating a new sentence.	3	6%

Table 10: Error types observed in GPT-4o generated sentences across the full corpus, where *unknown*-labeled sentences are categorized as errors (N = 50).

ical SVO (about 88-97%). Non-SVO patterns are almost absent: while SOV sentences appear very rarely (about 0.3-1.3%), other non-canonical word orders are never generated.

Interestingly, GPT-5-mini with a high level of reasoning shows the highest proportion of *unknown* sentences. Among these sentences, 32 instances (approximately 88.9%) correspond to empty outputs. This type of error occurs only at the high reasoning level. The other errors correspond to either incoherent outputs (2 sentences, 5.6%) or morphosyntactic mismatches (2 sentences, 5.6%), reflecting failures to meet the prompt requirements. *Unknown* sentences at other reasoning levels in the GPT-5-mini model have only the latter two types of errors.

6.4 Further Experiments with Prompt Engineering and Fine-Tuning

Next, we experimented with different prompting and training strategies for GPT-3.5-turbo and GPT-4o-mini. The results are presented in Table 11. Based on the experiments from Subsection 6.3, GPT-4o exhibits similar behavior, however it is more expensive to run. Thus, we excluded it from experiments described in this section.

First, we hypothesized that adding a note about word order in the prompt might bias the LLMs' outputs. Thus, to assess whether the models are reasoning about word order or just relying on surface-level statistical patterns, we removed this note from the prompt and rerun the experiment.

As illustrated in Table 11, the word order distribution remains relatively stable compared to the previous results (our baseline), with SVO being slightly more preferable (around 89-91%), indicating only a small effect of the word order note. Detailed results for model performance on both dev and test sets using the modified prompt are provided in Table 27 in Appendix K.1. Table 28 reports the overall and per-word-order accuracy for the models for this setting.

Second, we used the predictions from our Decision Tree model (Section 5) to explicitly specify in the prompt which word order the LLM should use when generating a sentence (an example from the corpus in which Decision Tree predicts the correct output, unlike the LLM, is shown in Appendix K.2). While this setting does not yield major differences for GPT-3.5-turbo compared to the baseline, GPT-4o-mini shows better results, producing more diverse outputs and, most importantly, achieving the lowest proportion of *unknown* sentences across all experiments (only 2.7%). Detailed results on dev and test sets are presented in Table 29 in Appendix K.2, while overall and per-word-order accuracy are shown in Table 30.

Then, we explored a few-shot prompt. The examples for each possible word order with their corresponding contexts in this subcorpus (verb-initial constructions were excluded from the train/dev/test split) were taken from the training set, with one example per word order. We used the original prompt, including a note about word order. As shown in Table 11, on the one hand, this strategy works best for GPT-3.5-turbo, outperforming the baseline in generating various word orders. However, it yields the lowest overall accuracy (66.78%). In contrast, GPT-4o-mini performs worst, generating mostly SVO sentences (95.5%), but with the fewest *unknown* sentences (3.6%). Detailed model performance on the dev and test sets, as well as overall and per-word-order accuracy, are provided in Tables 31 and 32, respectively, in Appendix K.3.

Finally, we attempted to fine-tune both GPT-3.5-turbo and GPT-4o-mini. However, fine-tuning the first model was blocked by OpenAI safety filters (plausibly, this is because movie scripts contain policy-violating content). As noted in (Qi et al., 2023), even a few harmful examples can compromise the model safety alignment. By contrast, GPT-4o is described as having stronger safety mechanisms and mitigations than earlier models (OpenAI,

WO	Gold	Baseline (original prompt)		No WO Note		DT predictions		Few-shot		Fine-tuned
		GPT-3.5-turbo	GPT-4o-mini	GPT-3.5-turbo	GPT-4o-mini	GPT-3.5-turbo	GPT-4o-mini	GPT-3.5-turbo	GPT-4o-mini	GPT-4o-mini
SVO	74.0%	83.6%	90.1%	88.3%	90.4%	84.6%	85.6%	79.9%	95.5%	**71.7%**
SOV	18.1%	6.4%	0.4%	2.1%	0.9%	5.6%	9.3%	12.3%	0.5%	**13.5%**
OSV	4.6%	0.4%	0.1%	–	–	0.7%	1.9%	0.3%	0.3%	**3.0%**
OVS	3.3%	–	–	–	–	–	–	–	–	1.4%
VSO	–	0.2%	–	0.1%	–	0.8%	0.2%	–	–	0.2%
VOS	–	–	–	–	–	–	0.2%	0.1%	–	0.4%
UNK	–	9.4%	9.3%	9.5%	8.8%	8.3%	**2.7%**	7.3%	3.6%	9.8%
Accuracy	–	67.87%	69.30%	68.09%	69.74%	67.65%	73.36%	66.78%	71.82%	**73.57%**

Table 11: Word order distributions across baseline annotations and multiple prompting and training strategies (on the test set).

2024). Thus, we were able to fine-tune GPT-4o-mini on the training corpus using OpenAI's supervised fine-tuning (OpenAI, 2024). Following the standard fine-tuning procedure, we formatted the training data as a JSONL file of input-output pairs. Each training example reflected the structure of the prompt, containing preceding context, constituents, and morphosyntactic features, paired with the target word order label as the expected output.

Notably, the fine-tuned GPT-4o-mini model outperforms the baseline and all other experimental settings. It exhibits greater output diversity, with the proportion of SVO sentences closely matching the human gold standard (71.7% vs. 74%), and it generates more non-canonical word orders (with a distribution closely aligned to the corpus). Although the model still generates many *unknown* sentences, overall it demonstrates the highest syntactic diversity and the best overall accuracy (3.6%). Detailed per-word-order accuracy results for the fine-tuned model can be found in Table 33 in Appendix L.

Thus, fine-tuning is the most effective strategy for generating diverse, natural text. This aligns with Chakrabarty et al. (2025) on imitating literary style: the authors demonstrate that fine-tuned models outperform in-context prompting for both stylistic fidelity and and text quality, with experts favoring fine-tuned outputs.

7 Conclusion and Future Work

We introduce an annotated corpus designed for in-context word order variation analyses in colloquial Russian, which includes a set of linguistic features. Our evaluation of traditional machine learning reveals that our linguistic features play a crucial role in predicting word order. We then turn to LLMs. Our experiments show that LLMs struggle to generate context-appropriate word orders, i.e., they have not learned to exploit the linguistic features that (to a large extent) can predict word order, as we have shown. Importantly, this conclusion is based on word order distributions observed in our corpus, rather than on whether individual sentences match the gold data.

This gap cannot be explained by a failure of LLMs to generate correct Russian sentences. Although Russian is a language with complex morphology, complex agreement patterns, and complex case assignment rules, we observe that overall, the generated surface strings are typically grammatically correct (the instances with some errors vary between 3.6% and 8.3%). Thus, the LLMs have learned the core grammar of Russian in pre-training, although there is almost certainly far less training data available compared to English. Additionally, the LLMs have learned that non-SVO word orders are grammatical, as evidenced by their occasional production. What the LLMs have not learned is how the contextual factors affect grammatical choice. Put differently, LLMs are great at syntax and semantics (the structure and meaning of language), but not at pragmatics, the knowledge of how to use language in context when communicating with other agents. The reason for this gap is uncertain. One possible explanation concerns the genre distribution of the pre-training data. If the training corpus is dominated by genres such as news or scientific writing, where SVO order is more frequent (Billings, 2015), the model may have learned SVO order as a default. However, without access to the pre-training data, this assumption cannot be confirmed. Our work points at possible ways of remedying the failure: specialized fine-tuning, or using external knowledge for generation.

Limitations

In this paper, we limit our investigation in several ways. First, we restrict our analysis to Russian, and within this language we examine only transitive sentences as a pilot study. Second, our corpus

is relatively small and contains only movie scripts, which (1) are not representative of other genres, and (2) are translations of originally non-Russian films. Nonetheless, we observe similar word order distribution in the original data, and the native-speaker first author did not identify any unnatural word orders. Furthermore, it is well known that intonation interacts with information structure, and we have completely set aside intonation. Nonetheless, we consider this corpus to be a useful first step, and we hope that this paper will lead to more explorations of generation of contextually appropriate word order in LLMs.

Ethical Considerations

The total cost of all API-based experiments described in this study was approximately US$30, with a total usage of 16.1M input tokens and 1.5M outputs tokens.

We have released the corpus and code, which are available at `https://github.com/alinashabaeva/word_order_modeling`.

Acknowledgments

Alina Shabaeva and Owen Rambow are grateful for support from the Institute for Advanced Computational Science (IACS) at Stony Brook University. We thank three anonymous reviewers for thoughtful comments.

References

AI Forever. 2022. ruGPTs: Generative pre-trained transformer models for Russian. `https://github.com/ai-forever/ru-gpts`.

Ian Apperly. 2010. *Mindreaders: The Cognitive Basis of Theory of Mind*. Psychology Press.

Mira Ariel. 1985. The discourse functions of given information. *Theoretical Linguistics*, 12(s1):99–114.

AD Baddeley and GJ Hitch. 2007. Working memory: Past, present... and future. *The cognitive neuroscience of working memory*, pages 1–20.

John F Bailyn. 2012. *The syntax of Russian*. Cambridge University Press.

John Frederick Bailyn. 1995. *A configurational approach to Russian "free" world order*. Cornell University.

Srinivas Bangalore and Owen Rambow. 2000. Exploiting a probabilistic hierarchical model for generation. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*.

Stephanie Kay Billings. 2015. *A corpus-based analysis of Russian word order patterns*. Brigham Young University.

Bernd Bohnet, Anders Björkelund, Jonas Kuhn, Wolfgang Seeker, and Sina Zarriess. 2012. Generating non-projective word order in statistical linearization. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 928–939, Jeju Island, Korea. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Aoife Cahill and Arndt Riester. 2009. Incorporating information status into generation ranking. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 817–825, Suntec, Singapore. Association for Computational Linguistics.

Wallace Chafe. 1976. Givenness, contrastiveness, definiteness, subjects, topics, and point of view. *Subject and topic*.

Tuhin Chakrabarty, Jane C. Ginsburg, and Paramveer Dhillon. 2025. Readers prefer outputs of AI trained on copyrighted books over expert human writers. *arXiv preprint arXiv:2510.13939*.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794.

Kordula De Kuthy, Nils Reiter, and Arndt Riester. 2018. QUD-based annotation of discourse structure and information structure: Tool and evaluation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The LLaMa 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.

Jan Firbas. 1964. On defining the theme in functional sentence analysis. *Linguistiques de Prague*, 1:267–280.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Jeanette K. Gundel, Nancy Hedberg, and Ron Zacharski. 1993. Cognitive status and the form of referring expressions in discourse. *Language*, 69(2):274–307.

Ilya Gusev. 2023. rulm: A toolkit for training neural language models. https://github.com/IlyaGusev/rulm.

Jan Hajič, Eduard Bejček, Jaroslava Hlaváčová, Marie Mikulová, Milan Straka, Jan Štěpánek, and Barbora Štěpánková. 2020. Prague Dependency Treebank–Consolidated 1.0. *arXiv preprint arXiv:2006.03679*.

Yufang Hou. 2020. Fine-grained information status classification using discourse context-aware BERT. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6101–6112, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ray Jackendoff. 1972. *Semantic Interpretation in Generative Grammar*. MIT Press, Cambridge, Mass.

Elena Dmitrievna Kallestinova. 2007. *Aspects of word order in Russian*. The University of Iowa.

Walter Kintsch and Teun A. van Dijk. 1978. Toward a model of text comprehension and production. *Psychological Review*, 85(5):363–394.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9. Association for Computational Linguistics. Event Title: The 27th International Conference on Computational Linguistics (COLING 2018).

Irina Ivanovna Kovtunova. 1976. *Sovremennyj Russkij jazyk [Modern Russian language]: Porjadok slov i aktual'noe chlenenie predlozhenija*. Prosveshchenie, Moscow.

O. A. Krylova and S. A. Khavronina. 1986. *Poryadok slov v Russkom yazyke*. Russkij yazyk, Moscow.

Ivona Kučerová. 2012. Grammatical marking of givenness. *Natural Language Semantics*, 20(1):1–30.

Oksana Laleko. 2022. Word order and information structure in heritage and L2 Russian: Focus and unaccusativity effects in subject inversion. *International Journal of Bilingualism*, 26(6):749–766.

Knud Lambrecht. 1994. *Information structure and sentence form: Topic, focus, and the mental representations of discourse referents*, volume 71. Cambridge university press.

Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. University of Chicago press.

Maxim Lisovsky. ca. 2019. Movie scripts dataset. https://www.kaggle.com/datasets/lisovskey/moviescripts. Kaggle dataset, last updated in 2019. Accessed: 2025-01-03. Database: Open Database.

Vilém Mathesius. 1939. O tak zvaném aktuálním členění větném. *Slovo a slovesnost*, 5(4):171–174.

J.L. Mcdonald, K. Bock, and M.H. Kelly. 1993. Word and world order: Semantic, phonological, and metrical determinants of serial position. *Cognitive Psychology*, 25(2):188–230.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, and Leo Wanner. 2019. The second multilingual surface realisation shared task (SR'19): Overview and evaluation results. In *Proceedings of the 2nd Workshop on Multilingual Surface Realisation (MSR 2019)*, pages 1–17, Hong Kong, China. Association for Computational Linguistics.

OpenAI. 2024. Fine-tuning API Documentation. https://developers.openai.com/api/docs/guides/supervised-fine-tuning.

OpenAI. 2024. GPT-4o system card. *arXiv preprint*, arXiv:2410.21276.

OpenAI. 2025. GPT-5 system card. Technical report.

Allan Paivio. 1991. Dual coding theory: Retrospect and current status. *Canadian Journal of Psychology/Revue canadienne de psychologie*, 45(3):255.

David Premack and Guy Woodruff. 1978. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences*, 1(4):515–526.

Ellen F Prince. 1981. Toward a taxonomy of given-new information. *Radical pragmatics*.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108. Association for Computational Linguistics.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. 2023. Fine-tuning aligned language models compromises safety, even when users do not intend to! *arXiv*, abs/2310.03693.

Ilja A Seržant, Daria Alfimova, Petr Biskup, and Ivan Seržants. 2025. Efficient sentence processing significantly affects the position of objects in Russian. *Linguistics*, 1:38.

Olga B Sirotinina. 1965. *Porjadok slov v Russkom jazyke ('Word order in Russian')*. Saratov, Russia: Saratov State University.

Natalia Slioussar and Maria Harchevnik. 2024. Word order and context in sentence processing: Evidence from L1 and L2 Russian. *Frontiers in Psychology*, 15:1344366.

Natalia Slioussar and Ilya Makarchuk. 2022. SOV in Russian: A corpus study. *Journal of Slavic Linguistics*, 30(3):1–14.

V. D. Solovyev, Yu. A. Volskaya, and R. B. Akhtyamov. 2023. [The spectrum of associations to Russian abstract and concrete nouns]. *Nauchnyi rezultat. Voprosy teoreticheskoi i prikladnoi lingvistiki*, 9(1):153–173. (in Russian).

Elena Titov. 2017. The canonical order of Russian objects. *Linguistic Inquiry*, 48(3):427–457.

Ada Tur, Gaurav Kamath, and Siva Reddy. 2025. Language models largely exhibit human-like constituent ordering preferences. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, page 2498–2521. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mT5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Olga T Yokoyama. 1987. *Discourse and word order*. John Benjamins Publishing Company.

Sina Zarrieß, Aoife Cahill, and Jonas Kuhn. 2012. To what extent does sentence-internal realisation reflect discourse context? a study on word order. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 767–776, Avignon, France. Association for Computational Linguistics.

Yulia Zuban, Maria Martynova, and Sabine Zerbian. 2021. Word order in heritage Russian: Clause type and majority language matter. *Russian Linguistics*, 45:253–281.

A Word Order Variation: Examples

An example below, taken from (Kallestinova, 2007, p. 30), illustrates how different word orders can be used to answer question *(1a)*, with subtle differences in interpretation and preference.

1.a. Kto razbil vazu?
Who broke vase
'Who broke the vase?'

b. Vazu razbila Olja. *OVS*
vase-Acc. *broke* *Olya*-Nom.
'Olya-Foc. broke the vase.'

c. Razbila vazu Olja. *VOS*

d. **OLJA** razbila vazu. *SVO*

e. Razbila **OLJA** vazu. *VSO*

The question in *(1a)* asks about the subject, **who** performed an action. Examples *(1b-1e)* show possible word orderings with the focused subject *Olya*. All of them could be translated as *'It was Olya who broke the vase.*' In *(1b-1c)* the focused subject occur in the sentence-final position. Both sentences are licit in Russian, although OVS order is generally more natural as a neutral answer to the question. As noted by Kallestinova (2007), sentences *(1c)* and *(1e)* are emotive – focus there is marked by sentential stress. While both sentences emphasize Olya as the focused element, *(1d)* is more natural. Since such effects depend on prosody, these examples are beyond the scope of the current paper, as it does not examine speech. In written contexts, the SVO order *(1d)* can scarcely be interpreted as having a focused subject and therefore is less likely to occur.

B Animacy

Numerous cross-linguistic and experimental studies have highlighted the role of animacy in determining word order patterns, with animate referents tending to occur in clause-initial position (Yokoyama, 1987; Mcdonald et al., 1993).

B.1 Automatic Annotation of Animacy

We annotated **animacy** for subject and objects using SpaCy's morphological tags, supplemented with additional rules. First and second person pronouns were marked animate, while third pronouns were left unspecified due to their possible reference to inanimates. Entities labeled as PER by the NER were treated as animate. In addition, we created a list of pronouns (e.g., *кто* 'who' vs. *что* 'what') and assigned each an animacy label.

B.2 Corpus Statistics for Animacy

Table 12 presents the distribution of animacy patters across different word orders in the corpus. Cases where animacy could not be determined were excluded (332 cases for subjects, 21.86%; 180 cases for objects, 11.85%).

WO	Subject Anim	Subject Inanim	Object Anim	Object Inanim
SVO	791 (90.0%)	88 (10.0%)	304 (29.8%)	716 (70.2%)
SOV	197 (94.7%)	11 (5.3%)	102 (47.9%)	111 (52.1%)
OSV	50 (90.9%)	5 (9.1%)	14 (20.3%)	55 (79.7%)
OVS	28 (62.2%)	17 (37.8%)	17 (45.9%)	20 (54.1%)

Table 12: Distribution of subject and object animacy and non-animacy across word orders in the corpus.

The distribution of subject animacy indicates a consistent preference for subjects to be animate, regardless of word order. In contrast, objects show greater variability. In SVO and OSV, most objects are inanimate (70.2% and 79.7%). However, SOV and OVS display a more balanced distribution of object animacy, with subjects remain predominantly animate (94.7% and 90.9%).

Chi-square tests confirm that animacy distributions significantly differ across word orders for both subjects and objects. For subjects, there is a strong association between animacy and word order, $\chi^2(6) = 53.63, p < .001$, and for objects it is even more significant, $\chi^2(6) = 84.37, p < .001$. These results suggest that animacy for both subjects and objects is significantly associated with word order, with object animacy having a stronger effect on word order variation.

Additionally, we examined how subject-object animacy pairs distribute across word orders in our corpus (see Table 13).

WO	A–A	A–I	I–A	I–I
OSV	7 (13.2%)	41 (77.4%)	5 (9.4%)	0 (0.0%)
OVS	6 (17.6%)	11 (32.4%)	10 (29.4%)	7 (20.6%)
SOV	77 (47.8%)	74 (46.0%)	5 (3.1%)	5 (3.1%)
SVO	227 (28.5%)	491 (61.7%)	27 (3.4%)	51 (6.4%)

Table 13: Subject–object animacy pairs by word order. A–A = animate–animate; A–I = animate–inanimate; I–A = inanimate–animate; I–I = inanimate–inanimate.

While sentences with animate subjects or objects (A-A and A-I) are relatively common across all word orders, inanimate subjects are rare in SVO, SOV, adn OSV. A chi-square test confirms a significant association between word order and subject-object animacy pairs ($\chi^2(15) = 120.44$, $p < .001$).

C Discourse-related Marking of Subjects and Objects: Algorithm and Statistics

We propose a rule-based algorithm for detecting subjects and objects marked by overt discourse cues associated with topic- or focus-related interpretations. The algorithm identifies three types of markers: preposed elements *только* (*tol'ko*) 'only', *лишь* (*liš'*) 'only', *даже* (*daže*) 'even', *именно* (*imenno*) 'exactly', *как раз* (*kak raz*) 'just/exactly', the postposed *же* (*že*) 'indeed', and the enclitic *-то* (*-to*) (this enclitic has no obvious glosses in English). For example, in the sentence *Этого-то я и опасаюсь* (*Etogo-to ja i opasajus'*) 'This is exactly what I am afraid of', the enclitic *-to* attached to the object *etogo* and functions as a discourse cue that signals emphasis. To avoid ambiguity, the algorithm excludes indefinite pronouns with similar surface forms by checking their lemmas (e.g., *кто-то* (*kto-to*) 'someone', *что-то* (*što-to*) 'something', *где-то* (*gde-to*) 'somewhere', *какой-то* (*kakoj-to*) 'some kind of'). For each sentence, the code checks relevant elements in the immediately adjacent positions to the subject and object constituents (and their heads). The output consists of binary labels indicating whether overt discourse cues are present for the subject and object.

However, this feature is scarcely represented in our corpus: out of 1519 sentences, the algorithm detected only 15 (1.0%) subjects with discourse-related elements, and 3 (0.2%) objects. Nevertheless, a chi-square test shows a statistically significant association between subjects marked by discourse cues and word order ($\chi^2(3) = 9.58$, $p = 0.022$), while the association between objects marked by discourse cues and word order is not significant ($\chi^2(3) = 6.44$, $p = 0.092$), probably due to their very limited occurrence in the corpus.

D Coreference Score

D.1 Definition

We define the coreference score as a unitary measure of how strongly a subject, object, and verb refer to prior discourse. This feature is particularly crucial for languages lacking articles like Russian, where referential status can only be inferred from the context. The coreference score feature is motivated by previous work on information structure and theme/rheme distinctions (Prince, 1981; Lambrecht, 1994), which shows that speakers tend to place given information earlier in a sentence and new information towards the end. Later work (Kučerová, 2012) presents a comparative study of three Slavic languages – Czech, Russian, and Serbo-Croatian, showing that in these languages givenness is always grammatically marked through word order.

To compute the referential status of subject, object, and verb with respect to the discourse, we

combined multiple tools targeting different types of coreference. In addition, we incorporated linear distance between the target and its most similar antecedent, motivated by psycholinguistic evidence that more recent entities remain more cognitively active (Chafe, 1976; Ariel, 1985; Baddeley and Hitch, 2007). To integrate these signals, we optimized category weights by grid search. The resulting algorithm works as follows:

(1) If the target is a **pronoun** or a **personal name** (NER=PER), the algorithm immediately assigns a score of **1.2** (determined by grid search).

(2) If the target is neither a pronoun nor a personal name, we use Stanza to determine whether the **same lemma** occurs in the preceding context. If the same lemma is found, we assign it a base weight of **0.8** (determined by grid search) if it appears in the speech-labeled context, and **0.6** (determined by grid search) if it occurs in the comment-labeled context. Next, to model the effect of distance on salience, we multiple the score by the exponential decay formula:

$$\text{decay} = e^{-\frac{\text{position}}{\text{total number of tokens}}}$$

(3) If the first two options are not found, we use **FastText embeddings** (Grave et al., 2018) to find semantically similar words. The similarity $\geq$ threshold is 0.5 (determined by grid search). The final score is calculated as **0.5** for speech and **0.2** for comment (determined by grid search), multiplied by (1-δ) where δ denotes the distance between the target vector and one of a related word. The smaller δ is (the more similar words), the higher the final score. This value is further adjusted by applying an exponential decay.

(4) For nouns, we also consider an abstractness feature using pre-calculated **abstractness** scores from a Russian dictionary, which was developed by the Laboratory of Quantitative Linguistics at Kazan Federal University and is available as an open-source resource (Solovyev et al., 2023). The intuition behind this method is that more concrete words are cognitively more accessible (Paivio, 1991). If the abstractness score is below a threshold of 0.14 (determined by grid search), we multiply it by a weight of **0.1** (determined by grid search). Next, we compare this value to the FastText-based score (if available). The higher of the two is retained as the final score.

(5) Finally, we take into account **definiteness**, as (Kučerová, 2012) shows that it impact word order in Slavic languages. In our setting, definiteness is incorporated in the coreference score feature and can be modeled by a determiner or a possessive pronoun that precede subject or object (e.g., *мой* 'my', *его* 'his', or *этот* 'this'). Our original idea was to boost the final score (if these markers were present) by multiplying it by a boosting score. However, the score determined by grid search turned out to be **1**, indicating no effect. A closer inspection of the data suggests that this outcome is because these constructions are extremely rare in our corpus. In datasets where such cases occur more frequently, this feature may have a stronger effect and result in a higher optimal boost.

As described above, tokens appearing in speech are assigned higher weights than those in comments. If both modalities are present, their scores are summed. See Appendix D.4 for details.

In preliminary experiments, we also had a verb coreference score. However, ANOVA showed that word order has a minimal impact on verb coreference score. Additionally, it has never occurred in the list of the most important features in ML classification experiments. One possible explanation is the crude computation of verb coreference score, which relied only on identical lemma and embeddings similarity. In contrast, subject and object can be linked to the context in more ways, making their scores more reliable. Therefore, we excluded verb coreference score from the corpus.

D.2 Coreference Score: Corpus Analysis

Table 14 represents average coreference scores for subjects and objects across four different word order patterns in the corpus.

Word Order	Subject	Object
OSV	1.200	0.538
OVS	0.449	0.970
SOV	1.133	1.079
SVO	1.030	0.437

Table 14: Average coreference scores for subjects and objects across word orders.

As shown in Table 14, subject scores are overall higher than object and verb scores, with the exception of OVS order where the object score is higher than the score of the postposed subject. This aligns with linguistic theories, as sentences typically begin with given information. The outlier is OSV for which the object score is quite low, presumably because topicalization has different pragmatic constraints on movement than the other types of move-

ment. Interestingly, verb scores remain consistently low across all orders.

We conducted one-way ANOVAs to further examine the relationship between word order and coreference scores. The results show a significant effect on coreference scores for subjects, objects, and verbs. However, the strength of the effect differed. Word order had a stronger impact on object coreference scores ($F = 77.41$, $p < .001$) than on subject coreference scores ($F = 30.15$, $p < .001$).

D.3 Coreference Score: Ablation Study for Machine Learning Experiments to Predict Word Order

Since the *coreference score* feature is novel and designed to capture subtle discourse-level information, we conducted an ablation study to determine its contribution. Thus, we removed the coreference scores for both subjects and objects and re-evaluated performance of the Decision Tree. The results for the multi-class setup are presented in Table 15.

Class	Precision	Recall	F1-score	Support
Development Set (Ablation: No Coreference Score)				
SVO	0.85	0.85	0.85	225
SOV	0.47	0.64	0.54	55
OSV	1.00	0.36	0.53	14
OVS	0.00	0.00	0.00	10
Accuracy	–	–	0.76	304
Macro-F1	0.58	0.46	0.48	304
Weighted-F1	0.76	0.76	0.75	304
Test Set (Ablation: No Coreference Score)				
SVO	0.87	0.89	0.88	225
SOV	0.56	0.69	0.62	55
OSV	1.00	0.29	0.44	14
OVS	0.00	0.00	0.00	10
Accuracy	–	–	0.80	304
Macro-F1	0.61	0.47	0.49	304
Weighted-F1	0.79	0.80	0.78	304

Table 15: Per-class performance of the **Decision Tree** classifier for the multiclass classification task on the development and test sets without coreference score features.

Comparing the two test set evaluations (Table 17 and 15), the model without *coreference scores* achieves only slightly better overall accuracy (which rises from 0.78 (using all features) to 0.80 (without coreference scores) but the macro-F1 score decreases only from 0.50 to 0.49. Interestingly, the OVS pattern shows the opposite result: its F1 drops from 0.15 (with coreference scores) to 0.00 (without), suggesting that coreference score might help capture distinctive patterns of rarer word orders like OVS.

When comparing the binary classification results with the *coreference score* feature (see Table 20) and without (see Table 16), we observe a very slight but consistent improvement after ablation. Specifically, accuracy increases from 0.79 to 0.81, and macro-F1 from 0.74 to 0.76. This indicates that the *coreference score* features provide little benefit in this simplified binary setting. Nonetheless, as discussed above, they still may be helpful in the multi-class classification tasks, where distinguishing between non-canonical word orders requires more nuanced cues.

Class	Precision	Recall	F1-score	Support
Development Set (Ablation: No Coreference Score)				
SVO	0.87	0.80	0.83	225
non-SVO	0.53	0.66	0.59	79
Accuracy	–	–	0.76	304
Macro-F1	0.70	0.73	0.71	304
Weighted-F1	0.78	0.76	0.77	304
Test Set (Ablation: No Coreference Score)				
SVO	0.88	0.85	0.87	225
non-SVO	0.61	0.68	0.65	79
Accuracy	–	–	0.81	304
Macro-F1	0.75	0.77	0.76	304
Weighted-F1	0.81	0.81	0.81	304

Table 16: Per-class performance of the **Decision Tree** classifier for the binary classification task (SVO vs. non-SVO) on the development and test sets without coreference score features.

D.4 Coreference Algorithm

Algorithm 1 Coreference Score Evaluation

Input: Target word (lemma of subject, object, or verb), constituent, context, POS, NER, abstractness dictionary, FastText model

Output: Coreference score ≥ 0 and method used

Decay Function: decay $= e^{-\frac{\text{position}}{\text{total tokens}}}$

Step 1: Pronoun / Person Entity
if *POS = Pronoun* ***or*** *NER = PER* **then**
  score $\leftarrow$ 1.2 method $\leftarrow$ pronoun_or_person
  return (score, method)

Step 2: Lemma Match
if *lemma occurs in context* **then**
  if *found in speech* **then**
    score $\leftarrow 0.8 \times$ decay
  else
    score $\leftarrow 0.6 \times$ decay
  method $\leftarrow$ lemma_match

Step 3: Semantic Similarity (FastText)
if *FastText model available* **then**
  find most similar token in context with similarity sim **if** $sim \geq 0.5$ **then**
    if *in speech* **then**
      score_sem $\leftarrow 0.5 \times sim \times$ decay
    else
      score_sem $\leftarrow 0.2 \times sim \times$ decay
    if *score_sem* $>$ *score* **then**
      score $\leftarrow$ score_sem method $\leftarrow$ semantic_similarity

Step 4: Abstractness Adjustment
if *lemma in abstractness dictionary & POS=NOUN & NER≠PER* **then**
  score_abstr $\leftarrow 0.1 \times score\_abstr$ **if** *score_abstr* $>$ *score* **then**
    score $\leftarrow$ score_abstr method $\leftarrow$ abstractness

Step 5: Determiner Boost
if *determiner precedes head & method* $\neq$ *pronoun / person* **then**
  score $\leftarrow$ score $\times 1.0$; `// boost factor is neutral`

return (score, method)

E Multiclass Classification

Table 17 presents the detailed performance of the Decision Tree classifier on both development and test sets for the multiclass classification task, compared against the majority and random baselines.

Class	Precision	Recall	F1-score	Support
Development Set				
SVO	0.89	0.83	0.86	225
SOV	0.48	0.75	0.58	55
OSV	0.71	0.36	0.48	14
OVS	1.00	0.10	0.18	10
Accuracy	–	–	0.77	304
Macro-F1	0.77	0.51	0.52	304
Weighted-F1	0.81	0.77	0.77	304
Test Set				
SVO	0.88	0.85	0.87	225
SOV	0.53	0.78	0.63	55
OSV	0.75	0.21	0.33	14
OVS	0.33	0.10	0.15	10
Accuracy	–	–	0.78	304
Macro-F1	0.62	0.49	0.50	304
Weighted-F1	0.80	0.78	0.78	304
Baselines (Test Set)				
Majority Baseline (SVO)	–	–	0.21	304
Random Baseline (stratified)	–	–	0.28	304

Table 17: Per-class performance of the **Decision Tree** classifier for the multiclass classification task on the development and test sets, with majority and stratified random baselines.

Tables 18 and 21 show the results of our experiments with multilayer perceptron (MLP) architectures of varying depth and width, ranging from one to three hidden layers with hidden unit sizes from 64-32 to 256-128-64. We used batch normalization for all models, dropout (0.3-0.4), and L2 regularization ($\lambda \in [0.001, 0.01]$). Models were trained with the Adam optimizer with learning rate of 0.001 and early stopping , with softmax and sigmoid output layers used for the multiclass and binary classification tasks, respectively.

Architecture	Accuracy	Macro-F1	Weighted-F1
3-layer-256-128-64	**0.74**	**0.41**	**0.70**
2-layer-256-128	0.74	0.36	0.68
1-layer-64	0.73	0.29	0.68
1-layer-128	0.75	0.25	0.66
2-layer-128-64	0.73	0.25	0.66

Table 18: Comparison of **neural architectures** for the multiclass classification task (dev set).

Table 19 reports per-class performance of the best-performing MLP model (3-layer-256-128-64) on the test set.

Class	Precision	Recall	F1-score	Support
OSV	0.00	0.00	0.00	14
OVS	0.00	0.00	0.00	10
SOV	0.52	0.20	0.29	55
SVO	0.77	0.96	0.85	225
Accuracy	–	–	0.75	304
Macro-F1	0.32	0.29	0.29	304
Weighted-F1	0.66	0.75	0.68	304

Table 19: Per-class performance of the **3-layer-256-128-64** MLP for the multiclass classification task (test set).

F Binary Classification

Class	Precision	Recall	F1-score	Support
Development Set				
SVO	0.88	0.82	0.85	225
non-SVO	0.57	0.70	0.63	79
Accuracy	–	–	0.79	304
Macro-F1	0.73	0.76	0.74	304
Weighted-F1	0.80	0.79	0.79	304
Test Set				
SVO	0.88	0.83	0.86	225
non-SVO	0.59	0.68	0.63	79
Accuracy	–	–	0.79	304
Macro-F1	0.73	0.76	0.74	304
Weighted-F1	0.81	0.79	0.80	304
Baselines (Test Set)				
Majority Baseline (SVO)	–	–	0.43	304
Random Baseline (stratified)	–	–	0.50	304

Table 20: Per-class performance of the **Decision Tree** classifier for the binary classification task (SVO vs. non-SVO) on the development and test sets, with majority and stratified random baselines.

The Decision Tree achieves an accuracy of 0.79 and a macro-F1 score of 0.74 on the test set, outperforming both the majority baseline (0.74 accuracy, 0.43 macro-F1) and the random baseline (0.62 accuracy, 0.50 macro-F1). While strong performance on the canonical SVO class (F1=0.86) is expected, it is noteworthy that the model achieves a solid score for the non-SVO category (F1=0.63). As expected, the binary setup substantially improves macro-F1 compared to the multi-class task (0.74 vs. 0.50). These results suggest that the classifier is not simply overfitting to the dominant order but is capturing systematic patterns in rarer orders, providing evidence of a meaningful signal.

Table 21 compares neural architectures for the binary classification task on the development set. The 2-layer-128-64 MLP achieves the highest scores. As shown in Table 22, the model performs well for the canonical SVO class (F1=0.86), however, performance on the non-SVO class is substantially lower (F1=0.23). Similar to the multiclass setup, this may be due to the limited size of the data.

Architecture	Accuracy	Macro-F1	Weighted-F1
2-layer-128-64	**0.76**	**0.64**	**0.74**
1-layer-64	0.75	0.60	0.72
1-layer-128	0.73	0.58	0.70
3-layer-256-128-64	0.74	0.55	0.69
2-layer-256-128	0.73	0.46	0.64

Table 21: Comparison of neural architectures for the binary classification task (SVO vs. non-SVO) (dev set).

Class	Precision	Recall	F1-score	Support
SVO	0.76	0.98	0.86	225
Non-SVO	0.73	0.14	0.23	79
Accuracy	–	–	0.76	304
Macro-F1	0.75	0.56	0.55	304
Weighted-F1	0.76	0.76	0.70	304

Table 22: Per-class performance of the **2-layer-128-64** MLP for the binary classification task (SVO vs. non-SVO) (test set).

G Prompt Template Used for LLM Experiments

Prompt Template

```
You are working with a movie script in Russian. Below
is a context from the script, where:
  SPEECH indicates character dialogue
  COMMENT indicates scene description or comment

Context:
{context}

Continue this movie scene by generating a new sentence
that naturally follows this context.  The generated
sentence MUST contain:

  - the verb "{verb_lemma}" (with the specified aspect,
tense, and negation, if applicable),
  - "{subject}" as the subject constituent (nominative
case), and
  - "{object}" as the object constituent (accusative
case).

REQUIREMENTS:
1) You must use the exact given words:
  - Subject: "{subject}"
  - Verb: "{verb_lemma}"
  - Object: "{object}"
2) These words must be used within the same clause as
direct arguments of the verb.
3) Generate exactly one sentence in Russian.  Do not
add translations, comments, or explanations.
4) The sentence must:
  - be in the indicative mood,
  - be natural continuation of the context,
  - be grammatically correct,
  - not be a question, command, or use the subjunctive
mood,
  - not be a copy of the context.

NOTE ON WORD ORDER:
While  Russian  allows  flexible  word  order,
Subject-Verb-Object  (SVO)  is  the  most  common
and natural pattern.  Prefer SVO unless the context
(pragmatic or stylistic features) or morphosyntactic
properties  require  a  different  order  for  emphasis,
information structure, style, or emotion.

OUTPUT:

Provide only one generated Russian sentence.
```

H Model Behavior Analysis

Table 23 shows the word order distribution produced by three models (GPT-3.5-turbo, GPT-4o-mini, and GPT-4o), compared with the corpus data (gold).

Order	Gold	GPT-3.5-turbo	GPT-4o-mini	GPT-4o
SVO	3369 (72.8%)	3930 (84.9%)	4275 (92.4%)	4218 (91.1%)
SOV	825 (17.8%)	295 (6.4%)	62 (1.3%)	227 (4.9%)
OSV	213 (4.6%)	4 (0.1%)	5 (0.1%)	17 (0.4%)
OVS	150 (3.2%)	–	2 (0.0%)	–
VSO	42 (0.9%)	13 (0.3%)	–	1 (0.0%)
VOS	30 (0.6%)	5 (0.1%)	1 (0.0%)	–
UNK	–	382 (8.3%)	284 (6.1%)	166 (3.6%)

Table 23: Word order distribution on the entire corpus and across GPT-3.5-turbo, GPT-4o, and GPT-4o-mini models.

Table 24 reports accuracy for each non-canonical word order, which remains relatively low across all models (approximately 42-47%). Although GPT-3.5 generated more diverse outputs in terms of word orders compared to the other models, its accuracy on these outputs is the lowest.

Model	Word Order	Correct / Total	Accuracy (%)
GPT-3.5	SOV	133 / 295	45.1
	OSV	0 / 4	0.0
	VSO	0 / 13	0.0
	VOS	0 / 5	0.0
	Overall	133 / 317	**42.0**
GPT-4o-mini	SOV	30 / 62	48.4
	OSV	2 / 5	40.0
	OVS	0 / 2	0.0
	VOS	0 / 1	0.0
	Overall	32 / 70	**45.7**
GPT-4o	SOV	109 / 227	48.0
	OSV	7 / 17	41.2
	VSO	0 / 1	0.0
	Overall	116 / 245	**47.4**

Table 24: Accuracy of generated non-canonical word orders across the full corpus. Accuracy reflects the proportion of generated non-canonical word orders that match the gold word order.

I Evaluation of GPT-5-mini

Table 25 shows the distribution of word orders in the corpus and in sentences generated by GPT-5-mini across four reasoning levels (using the first 100 instances from the entire corpus, including rare verb-initial patterns).

WO	Gold	Min	Low	Med	High
SVO	73 (73.0%)	283 (94.3%)	282 (94.0%)	291 (97.0%)	263 (87.7%)
SOV	13 (13.0%)	1 (0.3%)	4 (1.3%)	3 (1.0%)	1 (0.3%)
OSV	8 (8.0%)	–	–	–	–
OVS	4 (4.0%)	–	–	–	–
VSO	1 (1.0%)	1 (0.3%)	–	–	–
VOS	1 (1.0%)	–	–	–	–
UNK	–	15 (5.0%)	14 (4.7%)	6 (2.0%)	36 (12.0%)

Table 25: Word order distribution in the dataset and sentences generated by GPT-5-mini across four reasoning levels (first 100 prompts).

J Accuracy for Development and Test Sets

Table 26 reports per-word-order accuracy for GPT-3.5-turbo, GPT-4o-mini, and GPT-4o on the dev

and test sets.

Word Order	Development Set								
	GPT-3.5			GPT-4o-mini			GPT-4o		
	Correct	Total	Accuracy	Correct	Total	Acc.	Correct	Total	Acc.
SVO	572	675	84.7%	609	675	90.2%	596	675	88.3%
SOV	26	165	15.8%	2	165	1.2%	30	165	18.2%
OVS	0	30	0.0%	0	30	0.0%	0	30	0.0%
OSV	0	42	0.0%	0	42	0.0%	0	42	0.0%
Overall	598	912	65.6%	611	912	67.0%	626	912	68.6%

Word Order	Test Set								
	GPT-3.5			GPT-4o-mini			GPT-4o		
	Correct	Total	Acc.	Correct	Total	Accuracy	Correct	Total	Acc.
SVO	597	675	88.4%	629	675	93.2%	613	675	90.8%
SOV	22	165	13.3%	3	165	1.8%	27	165	16.4%
OVS	0	30	0.0%	0	30	0.0%	0	30	0.0%
OSV	0	42	0.0%	0	42	0.0%	0	42	0.0%
Overall	619/912 (67.9%)			632/912 (69.3%)			640/912 (70.2%)		

Table 26: Per–word-order accuracy for GPT-3.5-turbo, GPT-4o-mini, and GPT-4o on the development and test sets.

K Prompt Engineering

K.1 Excluding Word Order Instructions from the Prompt (Zero-Shot Setup)

Table 27 shows generated word order distributions without a note about word order in the prompt for GPT-3.5-turbo and GPT-4o-mini on the dev and test sets. Table 28 presents overall and per-word-order accuracy for GPT-3.5-turbo and GPT-4o-mini without a note about word order onthe dev and test sets.

WO	Generated Word Order Distribution (No WO Note)			
	GPT-3.5-turbo		GPT-4o-mini	
	Dev	Test	Dev	Test
SVO	798 (87.5%)	805 (88.3%)	834 (91.4%)	824 (90.4%)
SOV	14 (1.5%)	19 (2.1%)	6 (0.7%)	8 (0.9%)
VSO	–	1 (0.1%)	–	–
UNK	100 (11.0%)	87 (9.5%)	72 (7.9%)	80 (8.8%)
Total	912	912	912	912

Table 27: Generated word order distributions without a note about word order for GPT-3.5-turbo and GPT-4o-mini on the development and test sets.

WO	Accuracy					
	GPT-3.5-turbo			GPT-4o-mini		
	Correct	Total	Acc.	Correct	Total	Acc.
Development Set						
SVO	606	675	89.8%	624	675	92.4%
SOV	5	165	3.0%	3	165	1.8%
OVS	0	30	0.0%	0	30	0.0%
OSV	0	42	0.0%	0	42	0.0%
Total	611	912	67.0%	627	912	68.8%
Test Set						
SVO	613	675	90.8%	632	675	93.6%
SOV	8	165	4.8%	4	165	2.4%
OVS	0	30	0.0%	0	30	0.0%
OSV	0	42	0.0%	0	42	0.0%
Total	621	912	68.1%	636	912	69.7%

Table 28: Overall and per–word-order accuracy for GPT-3.5 and GPT-4o-mini without a note about word order on the development and test sets.

K.2 Using Decision Tree Predictions in the Prompt

Context:

- Не то все пропустим.
  Ne to vsyo propustim.
  'Otherwise, we will miss everything.'
- Мы ничего не пропустим.
  My nichego ne propustim.
  'We won't miss anything.'

Gold (from the corpus):

Они все пропустят. *SOV*
Oni vsyo propustyat.
'They will miss everything.'

Generated by GPT-4o-mini (original prompt with a word order note):

Они пропустят все. *SVO*
Oni propustyat vsyo.
'They will miss everything.'

Decision Tree prediction:

SOV (matches gold)

In the corpus example above, using our set of features, the Decision Tree classifier correctly predicts SOV word order, which is more natural in this case. This preference could be explained by mor-

phosyntactic and IS factors: the object is a pronoun and consequently discourse-given, which trigger earlier, preverbal position. In contrast, using the original prompt including a note about word order, the GPT-4o-mini model consistently generates the SVO order (3/3 runs).

Table 29 reports generated word order distributions and Table 30 shows overall and per-word-order accuracy for GPT-3.5-turbo and GPT-4o-mini on the dev and test sets, using Decision Tree predictions in the prompt.

WO	Generated Word Order Distribution			
	GPT-3.5-turbo		GPT-4o-mini	
	Dev	Test	Dev	Test
SVO	759 (83.2%)	772 (84.6%)	765 (83.9%)	781 (85.6%)
SOV	63 (6.9%)	51 (5.6%)	92 (10.1%)	85 (9.3%)
OSV	9 (1.0%)	6 (0.7%)	21 (2.3%)	17 (1.9%)
OVS	1 (0.1%)	–	–	–
VSO	7 (0.8%)	7 (0.8%)	3 (0.3%)	2 (0.2%)
VOS	–	–	–	2 (0.2%)
UNK	73 (8.0%)	76 (8.3%)	31 (3.4%)	25 (2.7%)
Total	912	912	912	912

Table 29: Generated word order distributions for GPT-3.5-turbo and GPT-4o-mini on the development and test sets (using Decision Tree predicitons in the prompt).

	Accuracy					
	GPT-3.5-turbo			GPT-4o-mini		
WO	Correct	Total	Acc.	Correct	Total	Acc.
Development Set						
SVO	590	675	87.4%	601	675	89.0%
SOV	25	165	15.2%	46	165	27.9%
OVS	1	30	3.3%	0	30	0.0%
OSV	9	42	21.4%	13	42	31.0%
Total	625	912	68.5%	660	912	72.4%
Test Set						
SVO	594	675	88.0%	615	675	91.1%
SOV	20	165	12.1%	44	165	26.7%
OVS	0	30	0.0%	0	30	0.0%
OSV	3	42	7.1%	10	42	23.8%
Total	617	912	67.7%	669	912	73.4%

Table 30: Overall and per–word-order accuracy for GPT-3.5-turbo and GPT-4o-mini on the development and test sets (using Decision Tree predictions in the prompt).

K.3 Few-Shot Setting

Table 31 shows word order distributions generated by GPT-3.5-turbo and GPT-4o-mini on the dev and test sets in the few-shot setting, while Table 32 reports total and per-word-order accuracy for the same models.

WO	Generated Word Order Distribution			
	GPT-3.5-turbo		GPT-4o-mini	
	Dev	Test	Dev	Test
SVO	702 (77.0%)	729 (79.9%)	871 (95.5%)	871 (95.5%)
SOV	119 (13.0%)	112 (12.3%)	1 (0.1%)	5 (0.5%)
OSV	–	3 (0.3%)	–	3 (0.3%)
OVS	5 (0.5%)	–	–	–
VSO	2 (0.2%)	–	–	–
VOS	–	1 (0.1%)	–	–
UNK	84 (9.2%)	67 (7.3%)	40 (4.4%)	33 (3.6%)
Total	912	912	912	912

Table 31: Generated word order distributions for GPT-3.5-turbo and GPT-4o-mini on the development and test sets (few-shot setting).

	Accuracy					
	GPT-3.5-turbo			GPT-4o-mini		
WO	Correct	Total	Acc.	Correct	Total	Acc.
Development Set						
SVO	560	675	83.0%	567	675	84.0%
SOV	49	165	29.7%	42	165	25.5%
OVS	5	30	16.7%	0	30	0.0%
OSV	0	42	0.0%	0	42	0.0%
Total	614	912	67.3%	609	912	66.8%
Test Set						
SVO	639	675	94.7%	652	675	96.6%
SOV	1	165	0.6%	3	165	1.8%
OVS	0	30	0.0%	0	30	0.0%
OSV	0	42	0.0%	0	42	0.0%
Total	640	912	70.2%	655	912	71.8%

Table 32: Total and per–word-order accuracy for GPT-3.5-turbo and GPT-4o-mini on the development and test sets (few-shot setting).

L Fine-tuning GPT 4o-mini

Table 33 displays per-word-order accuracy for the fine-tuned GPT-4o-mini on the test set.

Word Order	Correct	Total	Accuracy
SVO	572	675	84.7%
SOV	68	165	41.2%
OSV	21	42	50.0%
OVS	10	30	33.3%
Overall	**671**	**912**	**73.6%**

Table 33: Per-word-order accuracy for fine-tuned GPT-4o-mini on the test set.