# RUAccent: Advanced System for Stress Placement in Russian with Homograph Resolution

**Denis Petrov**
d.petrov.research@gmail.com

## Abstract

This paper presents a novel approach to the problem of stress placement in Russian text, with a particular focus on resolving homographs. We introduce a comprehensive system that combines morphological analysis, context-aware neural models, and a specialized "Ё-fikator" to accurately place stress in Russian words, including those with ambiguous pronunciations. Our system outperforms existing solutions, achieving a 0.96 accuracy on homographs and 0.97 accuracy on non-homograph words.

## 1 Introduction

Accurate stress placement is crucial for natural-sounding text-to-speech (TTS) systems, particularly in languages with complex stress patterns such as Russian. The challenge is further compounded by the presence of homographs — words that are spelled identically but have different meanings and stress patterns. Resolving these ambiguities is essential for producing intelligible and contextually appropriate synthesized speech.

In Russian, homographs can be categorized into several types:

1. Homographs that change meaning based on their morphological and syntactic features (e.g., рЕки/рекИ, where рЕки is the nominative plural form of "река" (river), and рекИ is the genitive singular form).

2. Homographs that can only be disambiguated using surrounding context (e.g., зАмок/замОк - castle/lock).

3. Ё-homographs (e.g., все/всё - all/everything).

Each type presents unique challenges for stress placement and requires specialized techniques for resolution.

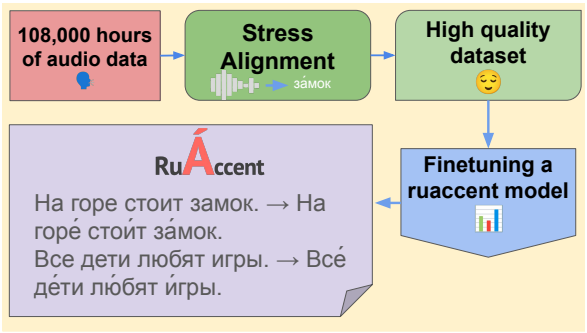


Figure 1: Audio alignment pipeline

Previous approaches to stress placement in Russian have often struggled with homographs, leading to misinterpretations and unnatural-sounding output. Our work addresses this gap by developing a system that not only places stress accurately on standard words, but also resolves homographs based on their context. The visualization of our system is presented in Figure 1.

The rest of this paper is organized as follows: Section 2 provides an overview of related work. Section 3 describes our methodology, including data preparation, model development, and system architecture. Section 4 presents our experimental results and comparative analysis. Section 5 concludes the paper and summarizes our key findings. Section 6 addresses the current limitations of our system, highlighting areas for potential improvement. Section 7 discusses the ethical considerations related to our use of audio data in this research. Section 8 outlines potential directions for future work, including possible adaptations of our approach to other Slavic languages.

The main contributions of this paper are:

- A novel system for Russian stress placement that achieves 0.96 accuracy, surpassing existing solutions.

- A comprehensive approach to homograph resolution, addressing each of the three types



*Proceedings of the 31st International Conference on Computational Linguistics*, pages 6642–6648
January 19–24, 2025. 

with specialized techniques.

- The development and analysis of improved datasets for training stress placement models, addressing limitations in existing corpora.
- A detailed comparative analysis of our system against current state-of-the-art solutions.

## 2 Related Work

The challenge of stress placement in Russian, particularly with regard to homograph resolution, has been the subject of several studies in recent years. However, most existing solutions have limitations in their ability to handle complex cases.

### 2.1 Rule-based Approaches

Early attempts at automated stress placement (Yakovenko et al., 2018) and (Kalinovskiy, 2024) in Russian relied heavily on rule-based systems and dictionaries. While these methods worked well for common words, they struggled with rare words, and especially certain types of homographs, which require contextual understanding.

These approaches showed some success with homographs that differ in morphological features, as these could often be disambiguated based on part-of-speech or grammatical form. However, these systems failed when dealing with contextual homographs, where the stress difference does not correlate with morphological differences.

A classic example of this limitation is the word pair "зАмок" (castle) and "замОк" (lock). In this case, where the correct stress placement depends entirely on the word's meaning in context, traditional rule-based systems were unable to make accurate predictions.

This limitation highlighted the need for more advanced approaches that could incorporate semantic and contextual information in the stress placement process.

### 2.2 Machine Learning Approaches

While more recent work has leveraged machine learning techniques, such as the systems developed by (Ponomareva et al., 2017) and (Kalinovskiy, 2024) that could resolve a limited number of homographs. The limitations of this earlier machine learning-based solution highlight the need for more advanced and comprehensive methods to address the challenge of stress placement and homograph resolution in Russian text-to-speech systems.

## 3 Methodology and Evolution of RUAccent

Our approach to Russian stress placement and homograph resolution evolved through several iterations, each building upon the lessons learned from the previous version. Here, we detail the progression of RUAccent[1] and our data collection and preparation methods.

### 3.1 RUAccent Version 1: Initial Data Sources

For our initial version, we utilized Common Crawl (Abadji et al., 2022), a large web corpus, as the primary data source. This process involved:

1. Filtering Common Crawl data for Russian-language content using Fasttext language detector (Joulin et al., 2016)
2. Extracting texts with existing stress markings using regular expressions.

However, we found that the data obtained from Common Crawl (Abadji et al., 2022) was often of very low quality, low variance and low homographs coverage.

### 3.2 RUAccent Version 2: RNC Data

In our second iteration of RUAccent, we primarily relied on the Common Crawl (Abadji et al., 2022) extractions for training data. However, we encountered significant limitations:

- Limited representation of certain types of homographs
- Imbalance between prose and poetry samples (see Figure 2)

To address these issues, we:

1. Filtered and cleaned the RNC (Savchuk et al., 2019) data:
   - Removed all non-Russian sentences
   - Filtered out sentences shorter than 2 words
   - Removed exact duplicate sentences
2. Augmented the dataset with additional prose samples from Common Crawl (Abadji et al., 2022), which underwent the same cleaning procedure

[1] https://github.com/Den4ikAI/ruaccent

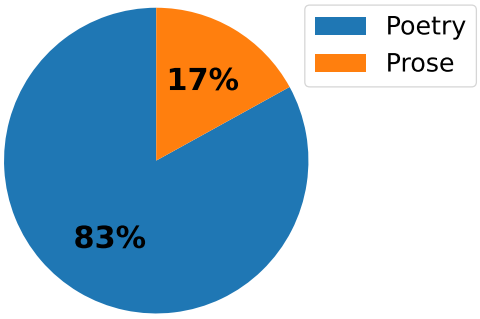


Figure 2: Composition of texts in the Russian National Corpus

### 3.3 RUAccent Version 3: Synthetic Data Generation

To address the limitations of both the RNC (Savchuk et al., 2019) and Common Crawl (Abadji et al., 2022) datasets, we have implemented synthetic data generation techniques in our third iteration:

1. Utilized the previous RUAccent model to generate initial stress placements
2. Employed RuPosTagger (Koziev, 2020) morphological analyzer to generate stress placements in morphological homographs

This approach allowed us to generate a large volume of diverse data, particularly for underrepresented homographs and challenging cases.

### 3.4 RUAccent Version 4: Incorporating Audio Data

In our latest iteration, we significantly expanded our dataset by incorporating audio data. We collected 108,000 hours of diverse audio content, including podcasts, audiobooks, YouTube videos, and radio records. This wide range of audio sources ensured a rich and varied dataset for training our models.

Our audio annotation process followed this algorithm:

1. Audio Input: We fed the audio into WhisperX (Bain et al., 2023) to obtain a textual transcription with word-level timestamps
2. Homograph Identification: We extracted the timestamps of words identified as homographs
3. Stress Classification: These homograph instances were then passed to our classifier, which predicted the correct stress variant

### 3.5 Audio-Based Stress Annotation

To process audio data, we developed an audio-text alignment system comprising a text encoder and an audio encoder. The text encoder, based on RoFormer (Su et al., 2023), was designed to support stress marking, while the audio encoder utilized a wav2vec (Baevski et al., 2020) model pre-trained on ASR tasks with stress information. We employed the Common Voice (Ardila et al., 2020) dataset for fine-tuning, augmenting it with stress markings generated by our RUAccent-turbo2 model.

The text encoder was trained on 200 GB of stress-marked text using AMLM (Autoregressive Masked Language Modeling) and NSP (Next Sentence Prediction) tasks, similar to the Canine (Clark et al., 2022) model.

For training the final classifier, we leveraged various Text-to-Speech (TTS) synthesizers capable of stress control, such as Silero (Team, 2021) and vosk-tts (Shmyrev and Team, 2023). This approach provided diverse audio samples for training. We applied contrastive learning to align representations from both text and audio modalities. During this process, all layers of the text encoder were unfrozen, while only the last two layers of the audio encoder were trainable.

This audio-based approach enabled us to cross-verify stress placement in spoken language and identify stress based on phonetic cues in audio data. Through iterative improvement of our data collection and preparation methods, we created a comprehensive, diverse, and high-quality dataset for training and testing our homograph resolution models.

### 3.6 System Architecture

Our system is comprised of three main components, each designed to handle specific aspects of the stress placement task:

1. Homograph Resolver
2. Word stress placer
3. "Ё-fikator"

#### 3.6.1 Homograph Resolver

For homographs that require broader contextual understanding, we developed a series of transformer-based models:

- Ruaccent-big: Trained on RNC (Savchuk et al., 2019) corpus
- Ruaccent-turbo: A more compact model trained on a significantly larger volume of data
- Ruaccent-turbo2: A finetuned Ruaccent-turbo model on extended corpus
- Ruaccent-turbo3: Trained using Audio Alignments

Ruaccent-big is ruBERT[2] (Zmitrovich et al., 2023) model, finetuned on RNC (Savchuk et al., 2019) corpus. Ruaccent-turbo, trained on 200 GB of text data annotated using various pipelines. Despite having only 80 million parameters, Ruaccent-turbo outperforms our previous best model, Ruaccent-big. Table 3 shows the comparison of the models on the top 200 homographs, with **Ruaccent-turbo3** achieving the highest accuracy. Also, we evaluated proprietary solutions.

#### 3.6.2 Homograph Resolver Training Process

Throughout the process, we continuously evaluated and refined our models to improve performance on real-world texts. This included further refinement of our pipeline for stress placement in regular words, incorporating neural network-based approaches for enhanced accuracy.

The architecture of our model consists of a transformer encoder with a linear layer on the head.

For the training of our models, we used the following hyperparameters: a learning rate of 2e-5, 2 training epochs, a batch size of 256 and a constant learning rate scheduler.

The training process was conducted on two RTX 4090 GPUs over a period of two weeks.

#### 3.6.3 "Ё-fikator"

To handle the case of "Ё-homographs", we developed a specialized model. This model is based on ruDistillBert[3] (Kolesnikova et al., 2022). We trained it using texts from Wikipedia.

[2] https://huggingface.co/ai-forever/Rubert-base

[3] https://huggingface.co/DeepPavlov/distilrubert-tiny-cased-conversational-5k

| Model | Accuracy |
|---|---|
| BERT (APE) | 0.951 |
| BERT (ALiBi) | 0.964 |
| **BERT (RoPE)** | **0.972** |

Table 1: Comparison of positional embeddings for word stress placement

The model was trained in a Named Entity Recognition (NER) style, where the "YO" label indicated that a word should be "Ё-ficated" (i.e., the letter "е" should be replaced with "ё"). This approach allowed the model to learn contextual cues for proper "Ё-fication" of words.

#### 3.6.4 Comparison of Positional Embeddings for Word Stress Placement

In the development of our RUAccent system, we required a model for accurate stress placement in regular words. During our experiments, we observed that the choice of positional embeddings had a significant impact on the model's performance in this task.

We compared three types of positional embeddings:

1. Absolute Positional Embeddings (APE) (Vaswani et al., 2023), as used in the classical BERT architecture (Devlin et al., 2019).
2. Attention with Linear Biases (ALiBi) (Press et al., 2022), an alternative positional embedding method.
3. Rotary Position Embeddings (RoPE) (Su et al., 2023), as used in the RoFormer (Su et al., 2023) architecture.

The results of our experiments are presented in Table 1:

The RoFormer (Su et al., 2023) model, which utilizes RoPE, achieved the highest accuracy of 0.972, outperforming both the classical BERT with APE (0.951) and BERT with ALiBi embeddings (0.964).

We attribute the superior performance of RoPE to several factors:

1. Shift invariance: RoPE encode the relative position of tokens, allowing the model to recognize stress patterns regardless of their absolute position in the word.

2. Expressiveness: RoPE use sine and cosine functions to rotate embeddings based on position, providing rich signals about the word's syllabic structure.

These properties make RoPE particularly well-suited for capturing the relative positions of syllables and their stress patterns within words. By leveraging RoPE in our RUAccent system, we were able to achieve state-of-the-art performance on the word stress placement task.

Our findings highlight the importance of selecting the appropriate positional embedding mechanism based on the specific task at hand. While APE and ALiBi embeddings have proven effective in various NLP tasks, RoPE demonstrably outperform them in the context of word stress placement. This insight guided our choice to use RoFormer (Su et al., 2023) with RoPE as the foundation for the word stress placement component of RUAccent.

## 4 Results and Evaluation

To comprehensively evaluate our system, we measured its performance across various components using a diverse and high-quality test set. This evaluation provided a multifaceted view of our system's capabilities, focusing on general stress placement accuracy as well as performance on homographs. For our test set, we selected samples from audio annotations with a confidence score above 0.99, ensuring a high level of reliability in our evaluation data. The size of our test corpus was substantial, comprising 2 million sentences. This extensive dataset allowed us to conduct a thorough assessment of our system's performance across a wide range of linguistic contexts.

### 4.1 Comparison of Stress Placers on non-homograph words

We compared our system with other stress placers on non-homograph words to evaluate the overall accuracy. Table 2 presents the results of this comparison.

Our RUAccent[4] system outperformed Silero[5] (Team, 2021), StressRNN[6] (Ponomareva et al., 2017), and RUSS Deberta[7] (Gusev, 2023), demonstrating its superior accuracy in stress placement.

[4] https://github.com/Den4ikAI/ruaccent

[5] https://github.com/snakers4/silero-models

[6] https://github.com/dbklim/StressRNN

[7] https://github.com/IlyaGusev/russ

| Stress Placer | Accuracy |
|---|---|
| Russtress | 0.673 |
| Ru Word Stress Deberta | 0.931 |
| Silero | 0.952 |
| **RUAccent** | **0.972** |

Table 2: Accuracy comparison of stress placers on non-homograph words

| Model | Accuracy |
|---|---|
| StressRNN | 0.0584 |
| ruaccent-big | 0.8886 |
| ruaccent-tiny | 0.9063 |
| ruaccent-turbo | 0.9089 |
| ruaccent-turbo2 | 0.9118 |
| sber-proprietary | 0.9191 |
| ruaccent-tiny2 | 0.9580 |
| **ruaccent-turbo3** | **0.9637** |

Table 3: Comparison of models

### 4.2 Homograph Resolver Accuracy

Table 3 presents the accuracy of our models compared to existing solutions.

Our final model, Ruaccent-turbo3, achieved the highest accuracy of 0.9638, outperforming existing solutions and our previous best model.

### 4.3 Qualitative Analysis

To demonstrate the practical effectiveness of RUAccent in handling various types of homographs, we present several examples below:

1. Contextual homographs: Он увидел зАмок на горе. / Он увидел замОк на двери. (He saw a castle on the hill. / He saw a lock on the door.)

RUAccent correctly identified the stress placement based on the context, distinguishing between "castle" and "lock".

2. Morphological homographs: Белые стЕны окружали город. / Около стенЫ старого дома лежал маленький котенок. (The white walls surrounded the city. / A little kitten lay by the wall of the old house.)

The system accurately placed stress on different syllables based on the grammatical case (nominative plural vs. genitive singular).

3. Ё-homographs: Все ушли домой. / Всё было готово к празднику. (Everyone went home. / Everything was ready for the celebration.)

RUAccent correctly differentiated between "" (all/everyone) and "" (everything), applying the

appropriate stress and letter choice.

4. Rare contextual homographs: Детей нужно вЫкупать перед сном. / Не выкупАть рофл. (The children need to be bathed before bed. / Don't buy out the ROFL.)

This example showcases RUAccent's ability to handle extremely rare and complex cases. The system correctly identified the stress in "" (to bathe) and distinguished it from the highly unusual phrase " " (to buy out the ROFL), demonstrating its robustness in dealing with modern internet slang and unexpected contexts.

These examples illustrate RUAccent's capability to handle a wide range of homograph types, from common contextual differences to more nuanced semantic distinctions. This demonstrates the system's robustness and practical applicability in real-world scenarios.

## 5 Conclusion

In this paper, we presented an approach to Russian stress placement with a focus on homograph resolution. Our system, combining morphological analysis and context-aware neural models, achieves state-of-the-art performance with 0.96 accuracy on homographs and 0.97 accuracy on non-homograph words.

## 6 Limitations

Despite the significant advancements introduced by RUAccent, several limitations still persist, which leave room for future improvements:

1. **Difficulty with Low-Resource Homographs**: Certain homographs, particularly those that appear infrequently in training data or belong to niche lexical categories, may still pose problems for accurate stress placement.

2. **Ambiguity in Complex Sentences**: In sentences with highly complex syntactic structures, the system's performance may degrade due to challenges in correctly interpreting long-range dependencies. As a result, it may incorrectly resolve stress patterns in contexts where multiple interpretations are plausible.

3. **Inconsistent Stress in Multiple Occurrences**: The model may struggle when encountering multiple instances of the same word with different meanings and stress patterns within a single text. For example, in the Russian sentence "На горе стоит зАмок, на двери которого висит замОк" (There is a castle on the hill, with a lock hanging on its door), the word "замок" appears twice with different stress patterns and meanings. Such cases can be challenging for the model to consistently and accurately resolve.

Addressing these limitations in future iterations of the system will further enhance its robustness.

## 7 Ethical Considerations

All audio data used in our study was obtained from the following sources:

- Audiobooks in the public domain,
- Podcasts licensed under Creative Commons,
- YouTube videos licensed under Creative Commons, used in compliance with the specified licensing terms,
- Public radio broadcasts.

We ensured that all data usage complies with applicable licensing and fair use policies.

## 8 Future Work

Future work will focus on further refining the system's performance on rare and domain-specific cases, as well as exploring integration with broader natural language processing applications. We believe that the methodologies and insights presented in this paper will contribute significantly to the ongoing development of accurate and context-aware text processing systems for Russian and potentially other languages with complex stress patterns.

### 8.1 Adaptation to Other Slavic Languages

The principles and architecture of RUAccent have potential applications beyond Russian. We propose extending our system to other Slavic languages, particularly those with similar stress placement challenges:

- Ukrainian: Like Russian, Ukrainian has mobile stress patterns and homographs. Adapting RUAccent to Ukrainian could involve retraining on Ukrainian corpora and adjusting for language-specific features.
- Belarusian: With its own set of stress rules and homographs, Belarusian presents an interesting challenge for adaptation.

Adapting RUAccent to these languages would involve:

1. Collecting word with stress markings corpora.
2. Collection of homograph dictionaries, collection of data for morphological analyzers and collection of texts with accents.
3. Collecting texts for pretraining homograph resolution module.
4. Developing language-specific homograph resolution strategies

## References


Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus. *Preprint*, arXiv:2201.06642.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. *Preprint*, arXiv:1912.06670.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Preprint*, arXiv:2006.11477.

Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. 2023. Whisperx: Time-accurate speech transcription of long-form audio. *Preprint*, arXiv:2303.00747.

Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2022. <scp>canine</scp>: Pre-training an efficient tokenization-free encoder for language representation. *Transactions of the Association for Computational Linguistics*, 10:73–91.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Ilya Gusev. 2023. russ: Package for word stress detection. https://github.com/IlyaGusev/russ.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Ilya Kalinovskiy. 2024. Multilingual text parser. https://github.com/just-ai/multilingual-text-parser.

Alina Kolesnikova, Yuri Kuratov, Vasily Konovalov, and Mikhail Burtsev. 2022. Knowledge distillation of russian language models with reduction of vocabulary. *arXiv preprint*.

Ilya Koziev. 2020. rupostagger. https://github.com/Koziev/rupostagger.

Maria Ponomareva, Kirill Milintsevich, Ekaterina Chernyak, and Anatoly Starostin. 2017. Automated word stress detection in russian.

Ofir Press, Noah A. Smith, and Mike Lewis. 2022. Train short, test long: Attention with linear biases enables input length extrapolation. *Preprint*, arXiv:2108.12409.

S. O. Savchuk, T. A. Arkhangelskiy, A. A. Bonch-Osmolovskaya, O. V. Donina, Yu. N. Kuznetsova, O. N. Lyashevskaya, B. V. Orekhov, and M. V. Podryadchikova. 2019. Russian national corpus 2.0: New opportunities and development prospects.

Nikolay Shmyrev and Vosk Team. 2023. vosk-tts: Simple tts based on vits with some old ideas. https://github.com/alphacep/vosk-tts.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2023. Roformer: Enhanced transformer with rotary position embedding. *Preprint*, arXiv:2104.09864.

Silero Team. 2021. Silero models: pre-trained enterprise-grade stt / tts models and benchmarks. https://github.com/snakers4/silero-models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need. *Preprint*, arXiv:1706.03762.

Olga Yakovenko, Ivan Bondarenko, Mariya Borovikova, and Daniil Vodolazsky. 2018. Algorithms for automatic accentuation and transcription of russian texts in speech recognition systems. In *International Conference on Speech and Computer*, pages 768–777. Springer.

Dmitry Zmitrovich, Alexander Abramov, Andrey Kalmykov, Maria Tikhonova, Ekaterina Taktasheva, Danil Astafurov, Mark Baushenko, Artem Snegirev, Tatiana Shavrina, Sergey Markov, Vladislav Mikhailov, and Alena Fenogenova. 2023. A family of pretrained transformer language models for russian. *Preprint*, arXiv:2309.10931.