Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2022"

Moscow, June 15–18, 2022

Between Denoising and Translation: Experiments in Text Detoxification

Sergey Pletenev
National Research University Higher School of Economics
Moscow, Russian Federation
alex010rey@gmail.com

Abstract

This paper describes a solution for the RUSSE Detoxification competition held as part of the Dialogue 2022 conference. The paper presents experiments based on autoregressive and non-autoregressive models. The following approaches are described in this paper: 1) Detoxification as a special case of the text style-transfer problem and the use of modern approaches to solve this task in Russian. 2) Using the Automatic Post-Editing algorithm as a task of translation from toxic to normative Russian text. The article provides an analysis of the listed models, their results in detoxification of sentences, as well an analysis of errors and reasons why the models gave such a diverse result.

Keywords: Sentence detoxification, pretrained language models, Non-autoregressive models, Russian language

DOI: 10.28995/2075-7182-2022-21-447-455

Между восстановлением текста и переводом: эксперименты по детоксикации текста

Плетенев Сергей
НИУ ВШЭ
Москва, Россия
alex010rey@gmail.com

Аннотация

В данной статье описано решение для соревнования по детоксикации предложений RUSSE Detoxification, проводящегося в рамках конференции Диалог 2022. В работе представлены эксперименты на основе авторегрессионных и неавторегрессионых моделей. В данной статье описываются следующие подходы: 1) Переопределение задачи детоксикации как частного случая задачи переноса стиля текста (style-transfer) и использование современных подходов для решения дан-ной задачи на русском языке. 2) Использование алгоритма автоматического пост-редактирования текста (Automatic Post-Editing) в качестве задачи перевода из токсичного в нормативный русский текст. В статье дан анализ перечисленных моделей, их результатов в детоксикации предложений, а также анализ ошибок и причин, по которым модели дали столь разнообразный результат.

Ключевые слова: Детоксикация предложений, Предобученные языковые модели, Неавторегрессионные модели, Русский язык

1 Introduction

With the widespread development of chats, social networks, and various forums, the need to classify and filter offensive content has emerged. There is a large class of articles(Wang et al., 2021)(Georgakopoulos et al., 2018) dealing with identifying and classifying offensive content sentences. But in addition to categorizing toxic sentences, there may be a requirement to detoxify sentences, i.e. to bring the text into a neutral, readable form. The task of detoxifying sentences seeks to reduce the offensive of the original sentence, but at the same time preserve the meaning and message of the text. The text detoxification problem can be reformulated as a subclass of the text style-transfer problem since the style transfer problem is a widely discussed and researched area of natural language processing.

The existing methods of text detoxification and style transfer are mostly made for the English language, which makes it difficult to transfer to other languages. For this purpose, the RUSSE Detoxification corpus(Dementieva et al., 2022) was developed to solve the detoxification problem in the Russian language. This paper describes the general problem statement and proposes a detoxification method based on RuT5 and describes in detail experiments with autoregressive (AR) and non-autoregressive models (NAR) for style transfer. We compare the capabilities of RuT5 (Raffel et al., 2019) models according to the baseline models of the competition, and explore different word alignment methods, combining different inference strategies and text preprocessing. The method was ranked 4th in the Automatic Evaluation and 1st in the Private Human Evaluation Leaderboard between models. To clarify, first place was awarded to "Human References", so the model received 2nd place overall. The article itself is organized as follows: Section 2 briefly describes previous research in style transfer; Section 3 describes the data used in the experiments; Section 4 describes the experiments; in Section 5 we discuss the results and provide an analysis of the proposed method and the generated the best model capabilities, In Section 6 we discuss the possible errors in the datasets and models that led to the disagreement of the scores; Section 7 concludes the article.

Our contribution is as follows:

1. We propose our method to detoxification using Seq2Seq models.
2. We adopt state-of-the-art style transfer models and evaluate them for the Russian language.
3. We publish experiments and our models for future research.[1]

2 Related works

We can categorize style transfer models into three types. The first type is the editing-based method (Li et al., 2018) (Shen et al., 2017), which edits the source sentence with several simple operations. The operations themselves consist of simply removing, replacing, or adding words to a sentence. The operations are usually trained separately and then constitute a pipeline. These methods are highly explainable and can be interpreted, but they usually need to locate and replace the stylist words, which hardly applies to complex tasks that require changes in sentence structures. The second type is sequence-to-sequence model because the detoxification task is similar to text generation tasks such as machine translation, summarization, or paraphrase generation. In this case, the model completely translates the text into a hidden representation of the model, and using a decoder generates new text sequentially, or autoregressive. This approach has shown good results in style transfer (John et al., 2019) and detoxification (Dale et al., 2021) tasks. But the main problem of such models is to preserve the original context, especially for long texts, which is a difficult task for seq2seq models. The third type of model (Huang et al., 2021) (Luo et al., 2019) combines the two previous approaches: on the one side, they create or learn a set of word alignments, and on the other hand, the sentence is generated end-to-end fashion. In addition, we adapted the Automatic Post-Editing (APE) method with Levenshtein transformer (Gu et al., 2019) to detoxification task. The APE consists of two steps: autoregressive(AR) generation using the seq2seq model and post-editing using an additional non-autoregressive(NAR) model.

3 Data

The organizers of the RUSSE Detoxification shared task has introduced a parallel detoxification dataset. The source sentences are Russian toxic messages from Odnoklassniki, Pikabu, and Twitter platforms. The target part of the dataset is the same messages which were manually rewritten by crowd workers to eliminate toxicity. Some toxic sentences contain multiple (up to 3) variants of detoxification. The dataset is divided into train, development, and test sets.

Dataset statistics:

- train: 3,539 toxic sentences with 1-3 detoxified versions;
- development: 800 toxic sentences with 1-3 detoxified versions;
- test: 1,474 toxic sentences with 1-3 detoxified versions.

[1] `https://github.com/A1exRey/DenoiseOrTranslation`

GYAFC Dataset	
Informal	**Formal**
Even the day after would be okay.	Even the following day would be alright.
well all ur missing is a million dollar smile..:)	Well, all you are missing is a smile worth a million dollars.
And I hear ya Fountain...same thing happened to me.	I hear you, Fountain, something similar happened to me.
altho, i dont really like girls all that much.	Although, I do not really like girls all that much.
RUSSE Detox Dataset	
Toxic	**Neutral**
это не наглость. это подлость! мерзавец папаколи	Это не наглость, это подлость
это твари а не люди <ThumbsDownEmoji>	Это плохие люди.
пошла на хер со своим гарантом дура	Оставьте всех в покое с гарантом.
ты зостовляиш миня стратать	ты заставляешь меня страдать

Figure 1: Examples from datasets

Due to fact that the amount of data is sufficient to train large pre-trained models but completely unsuitable for training models from scratch, we used an additional dataset (Dale, 2021) of paraphrases in Russian to train part of NAR models. This dataset contains 1 million sentences from the Russian part of Leipzig Corpora Collection 2019. Our choice of paraphrase is related to the success of such models in detoxification (Krishna et al., 2020).

Following (Kim and Rush, 2016), knowledge distillation from an autoregressive model can effectively simplify the training data distribution, which is essential for NAT-based models to learn good results. The easiest way of performing distillation is to follow the instructions of training a standard transformer model on the same data, and then decode the training set to produce a distillation dataset for NAT.

We use a pipeline to clean-up and prepare our datasets:

1. Remove punctuation marks from the text, except for the end of the sentence;
2. Remove emojis from text;
3. Remove unnecessary spaces in the sentence.

4 Experimental setup

The organizers of the competition provided several baselines:

1. **BertDelete** As a simple baseline this model is based on RuBert(Kuratov and Arkhipov,) model and toxic word dataset. Using lemmatized search on text, this algorithm mask-out all toxic words and then generate it with masked language model.
2. **RuT5** This baseline was simply trained on detoxification dataset with RuT5-base. No additional prefixes there used.
3. **ruPrompts** This baseline was trained on RuT5-base with model tuning via automatic prompt search.

For our experiments, we used NAST(Huang et al., 2021) model that performed best on the GYAFC (Rao and Tetreault, 2018) dataset. Grammarly's Yahoo Answers Formality Corpus (GYAFC) is the largest dataset for any style containing a total of 110K informal / formal sentence pairs in english. Differences between GYAFC and RUSSE Detox corpuses shown in Table 1. As we can see, both datasets have similar characteristics: missing or incorrect punctuation, emoji, grammatical errors, and toxic words.

NAST(Huang et al., 2021) is a non-autoregressive generator based on the observation of the word alignment: in style transfer tasks, most generated words can be aligned with the source words, where each pair of the aligned words is either identical or highly relevant. Additionally, we redesigned two non-autoregressive models to handle the detoxification task: LevT and DisCo.

text	Шустрая бурая лисица прыгает через ленивого пса														
src	Ш_ устра_ я бу_ рая ли_ си_ ца пры_ гает через лени_ вого п_ са														
0	Ш_	устра_	я	бу_	рая	ли_	си_	ца	пры_	гает	через	лени_	вого	п_	са
1	Ш_	устра_	я	бу_	рая	ли_	си_	ца	пры_	гает	через	лени_	вого	п_	са
2	Ш_	устра_	рый	бе_	рый	ли_	с	ца	пры_	гает	через	лени_	вого	п_	са
3	Ш_	устр_	рый	бу_	рый	ли_	с	пры_	гает	через	лени_	вого	п_	са	са
4	Ш_	устр_	рый	бу_	рый	ли_	с	пры_	гает	по	лени_	вым	п_	сам	[pad]

text	Шустрая бурая лисица прыгает через ленивого пса														
src	None														
0	Ш_	[mask]	я	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	п_	са
1	Ш_	[mask]	я	[mask]	ли_	с	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	[mask]	п_	са
2	Ш_	[mask]	я	[mask]	ли_	с	[mask]	[mask]	[mask]	[mask]	п_	са	,	п_	са
...	...														
12	Ш_	тур_	мо_	вая	ли_	су_	шка	прибе_	гает	к	не_	лов_	кому	п	су

Figure 2: The difference between NAR paraphrasers with and without src initialization

Disentangled Context (DisCo) transformer(Kasai et al., 2020) is a non-autoregressive sequence-to-sequence model. But unlike classical NAR architectures, where the model can only predict masked words, DisCo can predict all tokens simultaneously, which gives faster inference and improved model quality. It also uses a *easy-first algorithm*, in which each word is predicted by the words the model is most confident about. This decoding algorithm allows different contexts to be predicted in each iteration for all available tokens, allowing the decoding to stop when the model gets a good prediction. In this work we also use *mask-predict algorithm*, in which the number of iterations is always specified by some constant T.

The Levenshtein Transformer (LevT)(Gu et al., 2019) is a type of transformer that aims to transform text by sequentially adding, replacing, and deleting words. Hence, LevT is proposed to break the standardized decoding mechanism and replacing it with two basic operations — insertion and deletion. LevT is trained using imitation learning. The resulted model contains two policies and they are executed in an alternate manner. The authors argue that with this model decoding becomes more flexible. For example, when the decoder is given an empty token, it falls back to a normal sequence generation model. On the other hand, the decoder acts as a refinement model when the initial state is a low-quality generated sequence.

These models were initially trained on distilled russian paraphrase data, then fine-tuned on the detoxification corpus. In our case, as Teacher model we used for distillation rut5-paraphrase[2] which was evaluated on the paraphrase training dataset. A beam-search with size 3 was used as an additional parameter.

Because of the iterative generation of NAR models, it is possible to initialize them with any text. In experiments, we initialize them in three ways: **blank** – generation without initialization; **src** – generation with duplication of input text; **RuT5** – initialization with text obtained from autoregressive model RuT5. Similar methodology is used in LevT to solve APE task.

Evaluation is based on various metrics (Dale et al., 2021): **Style accuacy** (ACC) is based on pretrained toxicity classifier. **Content preservation** (SIM) is evaluated as the similarity of sentence-level embeddings of the original and transformed texts computed by the model. **Fluency** (FL) measured with the classifier of linguistic acceptability trained on the CoLA dataset. And **J** which is the multiplication of sentence-level style accuracy, content preservation, and fluency.

5 Results

For generation in all autoregressive and non-autoregressive models we use beam-search 12, with no limit on generation length. For non-autoregressive models we use the number of iterations equal to 16. For

[2] `https://huggingface.co/cointegrated/rut5-base-paraphraser`

model	Accuracy	Simularity	Fluency	J
RuT5-large	0.9475	0.8191	0.9107	0.7094
RuT5-baseline	0.796	0.827	0.837	0.560
RuT5-prompts	0.811	0.793	0.804	0.528
rubert-delete	0.558	0.887	0.852	0.406
NAST	0.8339	0.4983	0.7298	0.3074
LevT-blank	0.7327	0.0386	0.4135	0.0122
Disco-easy-first	0.1734	0.9321	0.9502	0.1435
Disco-mask-predict	0.3639	0.6685	0.7029	0.1707

Table 1: Test result without initialization

model	Accuracy	Simularity	Fluency	J
LevT-src	0.8262	0.4969	0.7071	0.2938
Disco-easy-first	0.4470	0.6611	0.7489	0.2255
Disco-mask-predict	0.3639	0.6685	0.7029	0.1707

Table 2: Test result with source initialization

generation with DisCo we use two different decoding methods: mask-predict and easy-first.

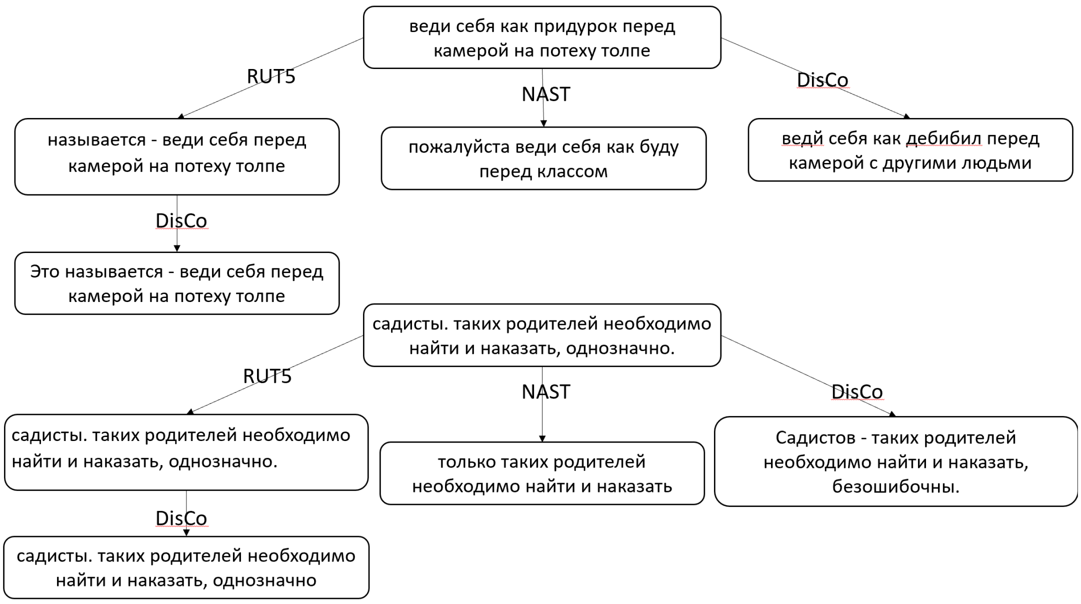

Figure 3: Visualisation of detoxification

We use HuggingFace Transformers[3] for RuT5 training and prediction. Each model is trained with following parameteres: encoder length 256, decoder length 256, batch size 3, 3 epochs, learning rate 5e-05, after each 1000 steps we evaluate our models with beam size 12.For NAR and paraphrase models training we use modified FairSeq[4] library.

As a strong baseline, we trained RuT5-large on the cleaned dataset. Table 1 shows the results of model generation without any additional initialization. The APE model showed the worst result, which is due to

[3]https://huggingface.co/transformers
[4]https://github.com/jungokasai/deep-shallow

model	Accuracy	Simularity	Fluency	J
RuT5-on-RuT5	0.9559	0.8043	0.9048	0.7007
RuT5-on-RuT5-beam1	0.9612	0.7979	0.9029	0.6982
LevT-mt	0.7881	0.6196	0.6195	0.3311
Disco-easy-first	0.9398	0.7645	0.8658	0.6314
Disco-mask-predict	0.8320	0.4789	0.3963	0.1570

Table 3: Test result with RuT5-large initialization

model	Accuracy	Simularity	Fluency	J
Human References	**0.888**	**0.824**	0.894	**0.653**
RuT5 (our)	0.794	**0.872**	**0.903**	**0.633**
RuT5 (baseline)	0.791	0.822	**0.925**	**0.606**
Ruprompts (baseline)	0.803	0.703	0.866	0.493
Delete (baseline)	0.387	0.705	0.726	0.162

Table 4: Private test with human evaluation

the fact that the model was originally trained not to generate a sentence from scratch, but only to rewrite the already prepared text. The best result was shown by the RuT5 model. On the other hand, none of these methods, even NAST, which showed state-of-the-art results on GYAFC dataset, could even beat the weakest baseline (bert-delete).

The table 2 shows results for non-autoregressive models with source text initialization. Compared to the previous table, all models except mask-predict showed a significant increase in quality. Probably, the reason for the low performance of mask-predict algorithm is connected to the fact, that large value of iterations for NAR model is set (16 iteration for each text), but this algorithm does not have an early-stopping mechanism. At the same time, easy-first has such an algorithm, so it gives us better results.

Next table 3 shows results for all of our models, which we initialized using the best results with RuT5-large. Additionally, we used the same model RuT5 again on the data from RuT5, with beam-search 1 and 12, but did not get any increase in the quality of the answers. The Disco-easy-first model also did not show any improvement in results. The model refused to complete most of the sentences, considering them as already good, while those sentences it stopped at we got a worse result than the original one. The levT model, although it got an additional increase, did not show high results. The Disco with mask-predict algorithm, like last time, showed a decrease in quality.

Since the automatic metrics (both reference-less classifiers and reference-based metrics) cannot reliably identify the best-performing model, competition organizers also conduct the manual evaluation of the private test set. Our best model (RuT5-large) was additionally tested with human evaluation. The result of this evaluation shown on table 4. Our model got first place, losing only to human evaluation.

6 Error Analysis

The first issue that could affect the quality of the models is the lack of data for paraphrasing. Although the training set has 1 million sentences, the encoder and decoder have the same dictionary, and the model is trained on the distilled data instead of the original data, the quality of the models indicates a lack of training. The second problem has to do with the language itself. The Russian language has a strong morphology, which can ruin the ability to link words to produce toxic-not-toxic pairs. This is the reason why the NAST model score is so low. The NAST model is very related to the generation of such pairs(Figure 4). The third problem is related to the dataset: in GYAFC dataset is much easier to extract individual words, while in RUSSE dataset there are some noisy data, which causes the tokenization to replace from 5% to 10% of all data with <unk>. As can be seen from the examples presented in

src	:-d:-d:-d ой бляя во дает блоханосец
src tokenized	:_ -_ d_ _ :_ -_ d_ _ :_ -_ d_ ой б_ ляя во дает бло_ ха_ но_ се_ ц
tgt	Ой, во даёт
tgt tokenized	О_ й_ , во даёт
src	Эй, слющяй, чо оскорбляешь срязю?
src tokenized	Эй_ , с_ лю_ щ_ яй_ , чо оскорб_ ля_ ешь с_ ря_ зю_ ?
tgt	Послушай, почему сразу оскорбляешь ?
tgt tokenized	Пос_ лу_ шай_ , почему сразу оскорб_ ля_ е_ ш_ ь ?

Figure 4: BPE tokenization for src and tgt sentences

Appendix A.1 when text is written only in capital letters BPE encoder generates a very large number of <unk> tokens, which prevents the NAR models from generating any meaningful text.

7 Conclusion

This paper presents our approach to detoxification in Russian language, which got J 0.633 and placed 2nd place (1st without Human References) on the private leaderboard. Our paper shows the effectiveness of the method. The error analysis also shows that the proposed approaches can perform better with additional datasets. But there is still the question of why larger models handle the task better. As a simple answer, let's try to reformulate the problem: detoxification as the sub-task of denoising. The input to the model is noisy text: it includes emoji, text with grammatical errors, toxic words, incorrect syntax, etc. Large models treat toxic words as noisy data, so the bert-delete baseline is relatively successful and does not generate a toxic word from the masked data again, but rather picks up a more neutral word. Yet larger models, such as RuT5, do this with all words. In addition, for such large models, there is no problem with tokenization so texts can be translated into tokenized form fully. This probably explains the high quality of sentence detoxification with the RuT5 model. In the future, we plan to adopt more NAR models for the style-transfer task.

References

David Dale, Anton Voronov, Daryna Dementieva, Varvara Logacheva, Olga Kozlova, Nikita Semenov, and Alexander Panchenko. 2021. Text detoxification using large pre-trained neural models. *CoRR*, abs/2109.08914.

David Dale. 2021. Paraphrasing russian texts: corpus, models, metrics, June. [Online; posted 28-June-2021].

Daryna Dementieva, Irina Nikishina, Varvara Logacheva, Alena Fenogenova, David Dale, Irina Krotova, Nikita Semenov, Tatiana Shavrina, and Alexander Panchenko. 2022. Russe-2022: Findings of the first russian detoxification task based on parallel corpora.

Spiros V. Georgakopoulos, Sotiris K. Tasoulis, Aristidis G. Vrahatis, and Vassilis P. Plagianakos. 2018. Convolutional neural networks for toxic comment classification.

Jiatao Gu, Changhan Wang, and Jake Zhao. 2019. Levenshtein transformer. *CoRR*, abs/1905.11006.

Fei Huang, Zikai Chen, Chen Henry Wu, Qihan Guo, Xiaoyan Zhu, and Minlie Huang. 2021. NAST: A non-autoregressive generator with word alignment for unsupervised text style transfer. // *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics: Findings*.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled representation learning for non-parallel text style transfer. // *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, P 424–434, Florence, Italy, July. Association for Computational Linguistics.

Jungo Kasai, James Cross, Marjan Ghazvininejad, and Jiatao Gu. 2020. Non-autoregressive machine translation with disentangled context transformer. // *Proc. of ICML*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. // *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, P 737–762, Online, November. Association for Computational Linguistics.

Yuri Kuratov and Mikhail Arkhipov. Adaptation of deep bidirectional multilingual transformers for russian language.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. // *North American Association for Computational Linguistics (NAACL)*.

Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. A dual reinforcement learning framework for unsupervised text style transfer. // *Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. // *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, P 129–140, New Orleans, Louisiana, June. Association for Computational Linguistics.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi S. Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. *CoRR*, abs/1705.09655.

Kehan Wang, Jiaxi Yang, and Hongjun Wu. 2021. A survey of toxic comment classification methods.

A Appendix

A.1 Generated examples

src	В такой ситуации виноваты пидарасы из Ростелекома у которых даже кошка может купить фильм с пульта. Дебилы блять.
t5	в такой ситуации виноваты люди из ростелекома, у которых даже кошка может купить фильм с пульта.
NAST	в такой ситуации виноваты люди из мешай у которых даже кошка может купить фильм с ещё
APE	в такой ситуации виноваты у которых даже кошка может купить фильм с пульта.
DisCo (easy-first)	В этой ситуации виноваты пидарасы из < unk>< unk>Ростелекома< unk>< unk>даже могут купить фильм с пульта
DisCo (mask-predict)	В этой ситуации у кокошки, есть есть кошки из стелекома, у которых даже есть возможность возможность кошка

src	и как земля таких уродов носит?
t5	и как земля их носит?
NAST	и как земля их носит как
APE	Ужие и как земля носит?
DisCo (easy-first)	И как земля, которые носят такие уроды?
DisCo (mask-predict)	И как их переносит, , как земля их носит?

src	а где ты денег столько взяла,уродина?
t5	а где ты денег столько взяла?
NAST	а где ты денег столько взяла
APE	УМИ: а где ты денег столько взяла?
DisCo (easy-first)	А где вы взяли столько денег,
DisCo (mask-predict)	Так где же вы взяли так много а г, , что взяла?

src	козел!!!!!!! ему бы там пожить. продумано до мелочей,как угробить людей за 10 жизни.
t5	ему бы там пожить. продумано до мелочей , как угробить людей за 10 жизни.
NAST	ему бы там не не до не не как не людей за не жизни
APE	ему бы там пожить. продумано до мело, как угробить людей за 10 жизней
DisCo (easy-first)	козел<unk><unk><unk><unk><unk><unk>! ему бы там пожить. продумано до мело <unk>,как угробить людей за 10 жизни.
DisCo (mask-predict)	Он бы бы там дожить до мельсамых разных блюдней вплоть до мелоза то, что что было продумано до мело<unk>, как оскорбить людей за 10 лет лет до конца

src	ПРОПУСТИЛА ГОДОВЩИНУ ПОБЕДЫ ЛИТЛ МИКС НА ИКС ФАКТОРЕ:((((СКА ТОЛЬКО Я ТАК МОГ
t5	пропустила слов нет, одни эмоции
NAST	пропустила прям хлеб литл всю на выходные я всю биологию только я так мог
APE	В ДЕке-: : : : : ——————ВИДЕК<unk>
DisCo (easy-first)	<unk>ПРОПУЧИНЫ <unk><unk>ЕЛЬ МИКС НА ИКС ФАКТОР<unk><unk><unk><unk><unk><unk>СК СКАЛЬКО Я ТАК МО<unk><unk><unk><unk>
DisCo (mask-predict)	<unk><unk><unk><unk>Нет, <unk><unk>я так мог