Improving a Domain Specific English ASR System for a Russian Native Speaker with Kaldi

Pedro González Bascoy[1] *, Maike Brauer*[1]

[1]University of Stuttgart

st168913@stud.uni-stuttgart.de, st169226@stud.uni-stuttgart.de

Abstract

Automatic speech recognition plays an essential role in human-machine interaction (HMI). In this globalized era, users may choose to use non-native languages for HMI when systems do not support them or to access the benefits of well-resourced languages, e.g. English. This work explores a number of modifications to adapt an ASR system for English of Russian native speakers. Models are built using the open-source Kaldi toolkit, and their results are compared to a baseline constructed using the `mini_librispeech` corpus. Our best speech recognizer is able to outperform our baseline system by about 46 percentage points.

1. Introduction

Automatic speech recognition, in other words having a system that is able to automatically understand human speech, has been a realm of research for several decades. The capability of the systems have improved with new methods, ranging from recognizing isolated words to continuous speech – from acoustic-phonetics based methods to incorporating finite-state machines [1], to a whole new level.

The goal of this project is to build an ASR system which is improved to correctly recognize English spoken by a Russian native speaker. With this in mind, we used the open-source toolkit Kaldi [2]. It targets acoustic modeling research and features a finite state transducer based framework written in `C++`. Currently, it competes against toolkits such as HTK [3] and RWTH [4], featuring better support for weighted finite state transducer and linear algebra. Conveniently, Kaldi incorporates several components of a speech recognition system separately. The user can make individual decisions on what methods to use, making it therefore a flexible system that can be adapted to the user needs.

In order to decide which modules should be tuned, it becomes necessary to inspect how Russian native speakers use English. Research about Russian users of English reveals that four different error categories can be found: pronunciation, lexical, grammatical and syntactic errors [5]. On the one hand, pronunciation errors constitutes an appealing area to improve any ASR system, as it impacts the performance deeply. They can be somewhat fixed in the pronunciation dictionary and therefore help the system to understand alternating pronunciations of words. Despite of several distinct pronunciation errors, naming and analyzing them all would go beyond the scope of this project. Devoicing of plosive consonants like /d/ to /t/ (e.g. in /had/ to /hat/) constitutes an example of a recurrent error often introduced by Russian native speakers. Other errors concern vowels. As Russian only has five distinct vowels [6], pronouncing the correct vowel can lead to problems. Furthermore, the vowel length is not differentiated by Russian native speakers. Besides this common struggles, Russian native speakers also tend to have problems with sounds that do not exist in their native language, e.g. /ð/ or /θ/. More details can be found in [5].

On the other hand, assumptions regarding the language model can be made when looking at the syntactic errors. A common error seems to be to violate the fixed word order in English statements and in direct as well as in indirect questions.

Finally, next to syntactic errors we encounter grammatical errors, which might also play an important role regarding the language model: Russian native speakers often omit or misuse the articles in English [5]. Overall there are a few common errors visible that can be used to improve a system that has been only trained on native speaker speech.

In the following section the data as well as the building steps for the system are described. Firstly, Section 2 and 3 present the setup and the ASR baseline system, respectively. Thereafter, various model improvements that have been made are described in Section 4 and the overall results are displayed in Section 5. Finally, the findings are discussed in Section 6 and the final remarks are included in Section 7.

2. Setup

This section details the train, validation and test datasets as well as the steps necessary to build ASR models using the Kaldi toolkit. As mentioned in Section 1, the goal is to build a reliable ASR system that is able to recognize English speech of a Russian native speaker. As corpus, we were provided with a total of eight tutorial videos describing the steps necessary to build a simple digit-focused ASR system with Kaldi. This means that our little corpus is very domain-specific (computer science, linux, ASR-domain), therefore there are potentially many little tweaks that can be made to improve the overall performance of the system.

Overall, the corpus contains 4,796.51 seconds of speech, i.e. about 80 minutes or 1.33 hours of total data distributed in eight different `mp4` tutorial videos. For building the system, the corpus was split into train, validation and test datasets. Such splits are detailed in Table 1.

Table 1: *Corpus splits.*

	Train	Validation	Test
Seconds	1,737.75	949.85	2,108.91
Minutes	28.96	15.83	35.15

The train dataset accounts for the first two tutorials (largest ones), namely “installation” and “directories”, while the smaller development dataset accounts for the tutorials 3 and 4, “data” and “project”. The test dataset was built over the last four tutorials: “features”, “LM”, “mono”, and “tri”; as no transcrip-

tions were available. Annotations for the videos were provided in TSV files, which detail the timestamps of each utterance and, for the first four videos, their correspondent transcriptions. Transcriptions were produced with an automatic tool and then manually revised for the sake of efficiency and correctness. Unfortunately, Kaldi does not include a video decoder module for creating MFCC features. We used the open-source `ffmpeg` video converter tool to extract wavefiles that can be used by Kaldi from the tutorial videos. Such wavefiles consist of a single channel, a bit rate of 256k, and were codified with 16-bit precision.

With the help of a bash script, a folder for each dataset was set up containing the directory structure and all the necessary files to be used by the Kaldi recipes. It is worth noting that during the process we created 6 mock speakers for each dataset to prevent malfunctioning in the decoding step (while splitting each dataset), and that all the transcriptions in the corpus were capitalized. Additionally, to get the phonetic transcriptions of our lexicon, we used the LOGIOS Lexicon Tool [7] (based on the CMU Pronouncing Dictionary). LOGIOS is able to generate a pronunciation dictionary from a list of English words. Phones generated using this online tool correspond to the ARPABET phonetic transcription codes, which are also used in many other recipes, e.g. Librispeech [8].

The language model was built using `mini_librispeech` recipes, and slightly tuned in the different baselines that are left to be discussed in the next section. Given the language model and the training data (MFCC features), four different ASR models are built. A monophone model, and three different triphone models: (1) a system accounting for delta/delta-delta features (TRI1), (2) a model with Linear Discriminant Analysis (LDA) and Maximum Likelihood Linear Transform (MLLT) feature transforms (TRI2), and finally (3) a model with Speaker Adaptive Training (SAT) on top of the LDA+MLLT transforms (TRI3B).

The performance of the ASR systems is measured in terms of word error rate (WER) on the validation dataset.

3. Baseline Model

In this section the baseline model used in this project is described. This model forms the basis for the following model improvements. The Kaldi toolkit offers different freely available recipes that can be used to build models. For this baseline system, the recipe from the `mini_librispeech` folder has been used.

The recipe relies on data from the `Librispeech` corpus, which contains 1000 hours of speech from audio books that are part of the LibriVox project [9]. The recipe uses a subset of the original corpus to train the decoding models and makes use of language model files, as well as pronunciations created on the complete corpus. The language model has been created with 14,500 public domain books as training data and the vocabulary file contains 200k words. The pronunciation dictionary holds pronunciations for the entire vocabulary.

Different decoding models were created as described in Section 2 and those were used to decode our data.

4. Model Improvements

This section provides insights on the improvements made to the baseline "mini_librispeech" ASR system. As our corpus is domain-specific, most of the changes are linguistically-oriented, targeting either the language model or the pronunciation dictionary. It is worth mentioning that the improvements have been incrementally implemented, so that any given model always takes benefits of the previous modifications.

4.1. Model 1

The first improvement on the baseline is motivated by its lamentable output. We noticed that it struggled to recognize most of the words and utterance structures. This came not as a surprise given that the baseline model used the language model of `librispeech`. Most of the lexicon of the tutorial videos, acronyms as well as sentence structures are thus not captured whatsoever in the training data.

For the first improvement we included the transcriptions of the training dataset to the language model, as well as its lexicon to the pronunciation dictionary. On top of that, we also coded a crawler to retrieve all the text present in the Kaldi tutorial webpage, which was added to the video transcriptions.

As far as the language model is concerned, we especially paid attention to the most used acronyms and domain-specific terms in the phonetic transcriptions and the crawled text, as they could be challenging for LOGIOS. It is worth noting that acronyms constitute a recurrent point of failure for the ASR system, as they appear endlessly throughout the tutorial videos. The analysis carried out on the pronunciation dictionary revealed that some terms and acronyms were incorrectly captured. Table 2 includes some misspelled terms along with their corrected pronunciations.

Table 2: *Some incorrect phonetic transcriptions.*

Term	LOGIOS	Corrected pronunciation
HMM	HH AH M	EY CH EH M EH M
ASR	AE Z R	EY EH S AA R
IMS	IH M S	AY EH M EH S
AWK	AO K	AA V EH K AA
CTRL	S IY T IY AA R L	K AH N T R OW L
EGS	EH G Z	IY JH IY EH S

4.2. Model 2

The second model aims to improve the pronunciation dictionary. The realization of the letter "r" in both Standard American English and Standard British English accents are either a postalveolar approximant (rhotic) /ɹ/ or an alveolar tap /ɾ/. However, the phonological system of standard Russian does not include this phoneme – the cyrillic letter "р" corresponds to the alveolar trill /r/. We have noticed though, that the speaker is able to mimic the rhotic r in most of the situations, but fails when the phoneme /ɹ/ is followed by a plosive consonant such as /p, b, k, g, t, d/, eliciting an alveolar trill instead. With this motivation in mind, we modified in the pronunciation dictionary all words containing the digraph pr, br, kr, gr, tr and dr accordingly to how they are actually pronounced by the speaker (Table 2).

We used the SAMPA symbol "`4`" for representing the IPA phoneme /ɾ/ (as ARPABET does not provide any alternative) whilst we continued using the ARPABET symbol "`R`" for the IPA phoneme /ɹ/ . After this modification, e.g. the term "CTRL" previously spelled as shown in Table 2, will be substituted by a new pronunciation entry: `K AH N T 4 OW L`.

4.3. Model 3

Similarly to the modifications done in the second model, the third model aims to improve the pronunciation dictionary even

Table 3: *Phone substitutions carried out in the pronunciation dictionary.*

Old pronunciation	New pronunciation
B R	B 4
P R	P 4
K R	K 4
G R	G 4
T R	T 4
D R	D 4

further. It is known that the vowel inventory of English is quite complex as compared to other mainstream languages, e.g. Japanese, Chinese or Spanish. Non-native speakers of such languages often struggle to hear and reproduce the differences between short and long vowels present in the English language, such as /i/ vs. /ɪ/ (for instance in words like "fit" and "feet", respectively), and /u/ vs. /ʊ/ (e.g. in words like "food" and "foot", respectively). On top of this, rounded and unrounded open vowels such as /ɒ/ vs. /ɑ/ represent further complications for foreign speakers.

Analyzing the corpus carefully, we noticed that the video presenter was not making distinctions between neither short and long nor between rounded and unrounded vowels, most of the times producing the long and unrounded versions instead. We modified the pronunciation dictionary to account for this phenomena as shown in Table 4, mapping the phones /ʊ, ɪ, æ, ʌ/ to /u, i, ɑ, ɑ/, respectively.

Table 4: *Vowel substitutions done in the pronunciation dictionary.*

Old pronunciation	New pronunciation
UH	UW
IH	IY
AE	AA
AH	AA

4.4. Model 4

The fourth model carries the improvements made in Model 1 and 2, but not in Model 3, and likewise, its objective is to take the pronunciation dictionary to the next level. The modification proposed for this model accounts for the devoicing phenomenon that occurs in many languages, e.g. German and Russian. More in detail, in the Russian language, all voiced consonants at the end of words are pronounced with their voiceless counterpart. If we extrapolate the devoicing to the English language, we are assuming that the voiced plosives /d, b, g/ at the end of words would be uttered respectively as /t, p, k/ by a Russian speaker.

As the described devoicing was found repeatedly in the corpus, we modified the pronunciation dictionary once more as shown in Table 5.

Table 5: *Modification of the pronunciation dictionary to account for devoicing.*

Old pronunciation	New pronunciation
B$	P$
D$	T$
G$	k$

Note that the symbol "$" indicates that substitutions are carried out when symbols appear at the end of words, not in the middle of them.

4.5. Model 5

With the aim of improving the language model, the fifth system extends it by including publications of the video presenter. By this means, the language model will have more data regarding the ASR domain and ultimately will also capture the way the presenter expresses himself in English.

We gathered the original LaTeX manuscripts of works [10, 11, 12, 13] and after preprocessing it, by removing both LaTeX commands and metadata before feeding it to the language model, a total of 435 new sentences were added to the language model.

4.6. Model 6

In order to improve the language model even further, the sixth and last model makes use of the Russian Error-Annotated Learner English Corpus (REALEC) [14]. It consists of hundreds of error and grammatically-annotated essays and thousands of sentences written in English by Russian native speakers. The main objective of including this corpus into the language model is to capture the most common grammatical errors that Russian native speakers make when expressing themselves in English, e.g. not including definite articles in front of nouns. The use of this corpus implies the addition of 369,464 non-domain-specific sentences to the language model.

5. Results

This section presents the gained results from testing the various systems on the validation set as well as the result of using our best model to decode the testing data. Overall the `tri1` and `tri3b` models proved to perform the best among the different decoding models. An overview of the results for the validation data across the different models can be seen in 7.

The baseline model scores with a WER of 79.06% for the `tri3b` model unsurprisingly bad compared to the improved systems because of the lack of domain specific vocabulary as well as speaker adapted pronunciations. As can be seen in 7, the results for all the other models generally show WERs that are in the range of 39% to 48%.

Model 1 demonstrates the incorporation of the transcriptions into the language model, as well as added phrases from the kaldi website. Furthermore, some adjustments were made regarding the pronunciation dictionary. With those changes we were able to achieve much lower WER scores for all decoding models. The best score for this configuration was achieved with using the `tri1` decoding model and lies at 41.39%.

With Model 2 we were able to decrease the WER across all decoding models further. The changes to the pronunciation dictionary were powerful enough to decrease the WER for the before mentioned decoding model further to 40.21%.

We tried to make us of our knowledge about the vowels used by Russian native speakers in Model 3. The made changes did not improve the model outcome from Model 2, but showed slightly worse results. When using the `tri1` decoding model, the best achieved result for this improvement model is a WER score of 40.46%.

The next model, Model 4, is build upon Model 1 and Model 2, leaving out the changes made in Model 3 because those were not helping to enhance the system. Model 4 is the system which provided us with the overall best result of 39.22% WER for both

Table 6: *The output of one sentence throughout the different decoding models. RF stands for reference, BL is the baseline model, M1 to M6 are Model 1 to Model 6 respectively. Correctly recognized words are highlighted.*

Model	Sentence
RF	and in order to check if what are possibilities you would need to go to this bin directories and see if there is anything what you can use
BL	checked youth **what** art the citizens **you** amidst a gold has **this** been glutted dissonance sooty **there**'s sniffing with **you can use**
M1	**check if** <unk>watter facilities **you would need to go to this** pineda is **and see if there is** nothing **what you can use**
M2	**check if** <unk>**what are** as lizay's **you need to go to** the is being nycteris **and** c freezing **what you can use**
M3	checking <unk>**what** artlessly dc **would need to go to this** p **directories and see if** receiving **what you can use**
M4	**check if** <unk>**what are** less resists **you need to go to this** being **directories and see if there is** nothing **what you can use**
M5	**check** give <unk>wonder us analysis **you need to go to** the speed **directories and see if there is** nothing **what you can use**
M6	**check** youth **want** our personalities **you need to go to** the cinema are represented **see** reciving **what you can use**

the `tri1` and the `tri3b` decoding system. We tried to achieve even better results by incorporating more language model material in Model 5 and 6, but both models were not able to go beyond the scores set by Model 4. The best score for Model 5 was achieved using either the `tri1` or `tri3b` decoding model and lies at 42.26%. Regarding the scores for the before mentioned decoding models, Model 5 scores worst among all improvement models. As can be seen in 7, Model 6 scores badly for the `mono` and also the `tri2b` decoding models. The other two scores are comparable to Model 5.

Table 7: *%WER for decoding the validation data, shown for all decoding models. BL corresponds to baseline model, M1 to M6 are showing the results for the models 1 to 6 respectively.*

System	BL	M1	M2	M3	M4	M5	M6
`mono`	87.79	45.11	43.56	44.61	43.68	43.68	48.51
`tri1`	83.89	41.39	40.21	40.46	**39.22**	42.26	41.57
`tri2b`	83.83	44.11	43.56	44.42	43.74	42.38	45.60
`tri3b`	79.06	42.13	41.08	41.14	**39.22**	42.26	42.57

For the final test of our best model on the test set, we had to choose one decoding model. Both the `tri1` and the `tri3b` model showed the same best result for Model 4. Due to the fact that the first one performed better in general, we choose that model for the final decoding. The result on the test data, 33.28% WER, was better than the best WER score we achieved up to this point.

Table 6 depicts the decoded version of one sentence taken from the validation dataset decoded models. For the example always the result using the `tri1` model is chosen because that model has the best WER. As can be seen in the table some parts of the sentence seem to be easier to understand than other parts. In general it becomes more visible that the recognition gets better respective to the results stated before.

6. Discussion

In this section the results are discussed and possible explanations for the results are explored. As of Model 1 the ASR system had access to domain specific words and phrases such that the results from all the improved models naturally differ from the baseline model. Nevertheless it is interesting to take a closer look on the different results and used methods. In order to achieve lower WERs, we focused mainly on two components of the speech recognition system: the language model and the pronunciation dictionary.

For the language model, we added four different sources to feed the system with domain specific vocabulary and speaker specific sentence structures. In Model 1 we added the transcriptions of the training dataset, as well as a dataset created from the kaldi website. In Model 5 we added publications of the speaker and in Model 6 we added texts from a learner corpora. The first two changes seemed to have a huge effect on the system, the remaining changes, however, could not be shown to be have a positive effect on the WER. The reason for that behavior might be that in the beginning adding domain specific information was urgently needed to recognize reasonable words. The later added sources have added more vocabulary, however, those words did not come from the same domain and were hence not useful for the task at hand. At this point it is also important to note that the lastly added sources did not cover spoken language, but were on the contrary both written formats. Hence it might be the case that transcripts of spoken texts could reveal a different result. In the scope of this project it was not possible to find or generate such a corpus to use for this purpose.

In Model 2 to 4, and also in Model 1 the focus was on the adaptation of the pronunciation dictionary. Model 1 showed a huge general improvement in comparison to the baseline model, but it is not completely clear how big the impact of the changed pronunciations for the acronyms in the first model is. But Model 2 and Model 4 can prove that adapting the pronunciations helps the model to recognize the speaker to a great extent. The only exception seems to be Model 3, where the adaptations to the pronunciations did not help but rather slightly raised the WER. This result is surprising since the adaptation of the vowels followed the insights we gained from researching about difficulties that Russian native speaker face when learning English [5]. As described in 1 and 4.3 we assumed a certain behavior and tried to cover that with a new mapping of vowels. A possible explanation for the outcome might be that the speaker not only uses the reduced set of vowels but does use more vowels throughout the entire speech. That means that some vowels might be recognized well with our mapping but that the model is missing out on vowels that are actually correctly pronounced.

7. Conclusion

In this work we adapted a Kaldi-based ASR system for recognizing English spoken by a Russian native speaker. On top of our baseline, trained with the `mini_librispeech` corpus, we added three modifications on the pronunciation dictionary to attend common pronunciation deficiencies of Russian speakers, and we extended the language model to capture domain-specific lexicon as well as grammatical/syntactical errors. The results show high accuracy improvements when accounting for the pronunciation phenomena and the domain-specific lexicon, whereas further language model extensions seem to overcomplicate the model, reporting lower WERs.

8. References

[1] B. Juang and L. Rabiner, "Automatic speech recognition - a brief history of the technology development," *Elsevier Encyclopedia of*

Language and Linguistics, Second Edition, 2005.

[2] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Vesel, "The kaldi speech recognition toolkit," *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 2011.

[3] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, A. Ragni, V. Valtchev, P. Woodland, and C. Zhang, *The HTK Book (version 3.5a)*, 12 2015.

[4] D. Rybach, C. Gollan, G. Heigold, B. Hoffmeister, J. Lööf, R. Schlüter, and H. Ney, "The rwth aachen university open source speech recognition system," in *INTERSPEECH*, 2009.

[5] O. Bondarenko, "Does russian english exist?" *American Journal of Educational Research*, vol. 2, no. 9, pp. 832–839, 2014. [Online]. Available: http://pubs.sciepub.com/

[6] L. G. Jones, "The vowels of english and russian: An acoustic comparison," *¡i¿WORD¡/i¿*, vol. 9, no. 4, pp. 354–361, 1953. [Online]. Available: https://doi.org/10.1080/00437956.1953.11659480

[7] K. . Lee, H. . Hon, and R. Reddy, "An overview of the sphinx speech recognition system," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 1, pp. 35–45, 1990.

[8] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[9] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.

[10] P. Denisov, N. T. Vu, and M. F. Font, "Unsupervised domain adaptation by adversarial learning for robust speech recognition," 2018.

[11] P. Denisov and N. T. Vu, "End-to-end multi-speaker speech recognition using speaker embeddings and transfer learning," 2019.

[12] ——, "Ims-speech: A speech to text tool," 2019.

[13] ——, "Pretrained semantic speech embeddings for end-to-end spoken language understanding via cross-modal teacher-student learning," 2020.

[14] E. Kuzmenko and A. Kutuzov, "Russian error-annotated learner English corpus: a tool for computer-assisted language learning," in *Proceedings of the third workshop on NLP for computer-assisted language learning*. Uppsala, Sweden: LiU Electronic Press, Nov. 2014, pp. 87–97. [Online]. Available: https://www.aclweb.org/anthology/W14-3507