Proceedings of the 18th Conference on Computer Science and Intelligence Systems pp. 1319–1324
DOI: 10.15439/2023F6058
ISSN 2300-5963 ACSIS, Vol. 35

Center for Artificial Intelligence Challenge on Conversational AI Correctness

Marek Kubis, Paweł Skórzewski
0000-0002-2016-2598
0000-0002-5056-2808
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
ul. Uniwersytetu Poznańskiego 4, 61-614 Poznań, Poland
Email: {marek.kubis, pawel.skorzewski}@amu.edu.pl

Marcin Sowański, Tomasz Ziętkiewicz
0000-0002-9360-1395
0000-0002-2594-4660
Samsung Research Poland
Plac Europejski 1, 00-844 Warsaw, Poland
Email: {m.sowanski, t.zietkiewicz}@samsung.com

***Abstract*—This paper describes a challenge on Conversational AI correctness with the goal to develop Natural Language Understanding models that are robust against speech recognition errors. The data for the competition consist of natural language utterances along with semantic frames that represent the commands targeted at a virtual assistant. The specification of the task is given along with the data preparation procedure and the evaluation rules. The baseline models for the task are discussed and the results of the competition are reported.**

I. Introduction

REGARDLESS of the near-human accuracy of Automatic Speech Recognition in general-purpose transcription tasks, speech recognition errors can significantly deteriorate the performance of a Natural Language Understanding model that follows the speech-to-text module in a virtual assistant. The problem is even more apparent when an ASR system from an external vendor is used as an integral part of a conversational system without any further adaptation. The goal of this competition is to develop Natural Language Understanding models that are robust to speech recognition errors.

The approach used to prepare data for the challenge is meant to promote models robust to various types of errors in the input, making it impossible to solve the task by simply learning a shallow mapping from incorrectly recognized words to the correct ones. It reflects real-world scenarios where the NLU system is presented with inputs that exhibit various disturbances due to changes in the ASR model, acoustic conditions, speaker variation, and other causes.

II. Related Work

The robustness of Natural Language Understanding models to various types of errors is a subject of several publications. Some authors proposed to use word confusion networks to improve models' robustness to ASR errors [1], [2], [3], [4]. Reference [5] developed a learning criterion that prefers NLU models that are robust to ASR errors by adding a loss term that measures the distance between the prediction distribution from transcriptions and ASR hypotheses. Reference [6] studied the performance of intent classification and slot labeling models with respect to several kinds of perturbations, such as substituting abbreviations and synonyms, changing casing and punctuation, paraphrasing, and introducing misspellings and morphological variants. Speech characteristics are among three aspects of robustness investigated by [7] in the assessment of task-oriented dialog systems. Reference [8] investigated data-efficient techniques that apply to a wide range of natural language understanding models used in large-scale production environments to make them robust against speech recognition errors, using domain classification as an example. The authors compared the effectiveness of several such techniques in terms of time-varying usage patterns and distribution of ASR errors.

Several benchmarks exist to evaluate NLU models regarding their robustness to ASR errors. RADDLE [9], a benchmark for evaluating the performance of dialog models, prefers models robust to language variations, speech errors, unseen entities, and out-of-domain utterances. ASR-GLUE [10] is a benchmark consisting of 6 different NLU tasks, for which the input data were recorded by six different speakers and at three different noise levels.

Mitigating the impact of ASR errors on downstream tasks was the subject of several contests. In [11], the authors proposed a challenge for improving the recognition rate of an ASR system on the basis of incorrect ASR hypotheses paired with reference texts. Post-edition of ASR output was also the objective of the shared task held by [12]. Speech-aware dialogue state tracking was the topic of a recent competition conducted by [13].

The data preparation procedure outlined in Section III involves combining a TTS model and an ASR system. Augmentation of speech corpora with the use of synthesized speech was investigated by [14] and [15]. Reference [13] uses synthesized inputs along with spoken utterances in their challenge.

Holding competitions as a method for finding promising solutions to scientific problems has a long history in computer science, particularly in natural language processing [16], [17].

This research was partially funded by the "CAIMAC: Conversational AI Multilingual Augmentation and Compression" project, a cooperation between Adam Mickiewicz University and Samsung Electronics.

This contest is organized under the 1st Symposium on Challenges for Natural Language Processing (CNLPS), a part of the 18th Conference on Computer Science and Intelligence Systems (FedCSIS 2023). FedCSIS Conference Series hosted a wide range of data mining competitions through the years that covered topics such as identifying key risk factors for the Polish State Fire Service [18], network device workload prediction [19], and predicting the costs of forwarding contracts [20]. In the process of running our CNLPS challenge, we followed the best practices set out by the organizers of FedCSIS data mining competitions.

III. Data

The data for the task are derived from the Leyzer dataset [21]. The samples consist of user utterances and the semantic representation of the commands targeted at a virtual assistant (VA). A fraction of the utterances in the training set is contaminated with speech recognition errors; however, we left most of the utterances intact to make the task more challenging. The erroneous samples were obtained from user utterances using a TTS model followed by an ASR system.

A. Preparation of Base Text Corpus

We used the second version of the Leyzer corpus, which consists of more utterance variations when compared to the version described in the original paper. The second version of the corpus introduced two additional sub-intent differentiation levels called *naturalness level* (or simply *level*) and *verb pattern*. Although we have not implicitly used this information in this contest, it allowed us to create more variant corpus for the task. Leyzer consists of 20 domains across three languages: English, Spanish, and Polish, with 186 intents and a wide range of samples per intent. Domains can be grouped into several topics that can be found in the most popular VAs:

- **Communication** with Email, Facebook, Phone, Slack, and Twitter domains in that group, which all relate to communication and the transfer of ideas,
- **Internet** with Web Search and Wikipedia that groups domains related to the search for information on the web; therefore, these domains will have a lot of open-title queries,
- **Media and Entertainment** with Spotify, YouTube, and Instagram domains in that group, which relate to multimedia content with named entities connected to artists or titles,
- **Devices** with Air Conditioner and Speaker domains, which represent simple physical devices that can be controlled by voice,
- **Self-management** with Calendar and Contacts, which consist of actions that involve time planning and people,
- **Other**, uncategorized domains (Fitbit, Google Drive, News, Translate, Weather, Yelp) represent functions and language not shared by other categories. In this sense, the remaining domains can be understood as intentionally not matching the other domains.

Using scripts provided in the Leyzer repository, we generated the text corpus from JSGF grammars. The corpus was divided into `train`, `valid`, `test-A`, and `test-B` parts using the splitting script provided in the Leyzer repository. First, we differentiate `test-B` from the rest of the corpus. For `test-B`, a minimum of 1 test case and up to 20% of the total available sentences for each intent, level, and verb pattern were selected, and the remaining test cases were left in the development corpus. From the development part of the corpus, we further differentiate `test-A` using the same procedure as for `test-B`, which extracted a minimum of 1 and up to 20% of test cases for each intent, level, and verb pattern triplet. The remaining corpus was divided into `train` and `valid` subsets. The `valid` subset is 20% of randomly selected test cases without assuring that it contains at least 1 test case for each intent, level, and verb pattern triplet.

B. Augmenting Corpus with Back-transcription

Back-transcription is a technique that can be used to produce speech transcripts from text-only data. Textual data are fed to a TTS engine to produce a speech signal, which in turn is fed to an ASR system, producing an augmented text. Depending on the performance of both models and differences in text normalization performed on the input text, as well as inside these models, the resulting text can be identical to the input or may contain differences introduced in either processing stage. The technique has been used to develop post-processing [22] and error correction [23] models for ASR systems.

We use back-transcription to simulate a virtual assistant user's behavior. The user speaks to the system, and their speech is converted into text by an ASR model, which is subsequently processed by an NLU model (see Fig. 1). NLU text prompts from the Leyzer corpus are synthesized using a TTS engine. The resulting sound signal is used as input to an ASR model producing back a text with an augmented NLU prompt. The procedure is illustrated in Fig. 2. To perform Text-To-Speech synthesis, we used the FastSpeech 2[1] model [24] for English, the VITS model [25] for Polish, and Tacotron 2 [26] for Spanish, both from the Coqui TTS library [27]. Speech recognition was performed using the Whisper[2] model [28] for all three languages.

C. CAICCAIC Dataset

The training data are located in the `train` directory of the contest's repository[3]. The `train` directory contains two files:

- `in.tsv` with four columns:
    1) sample identifier: *306*,
    2) language code: *en-US*,
    3) data split type: *train*,
    4) utterance: *adjust the temperature to 82 degrees fahrenheit on my reception room thermostat.*
- `expected.tsv` with three columns representing:

[1] https://huggingface.co/facebook/fastspeech2-en-ljspeech

[2] https://huggingface.co/openai/whisper-large

[3] https://github.com/kubapok/cnlps-caiccaic

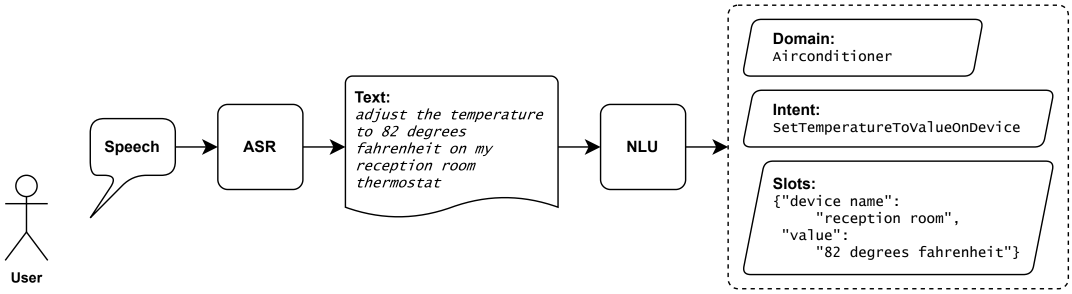
User
Speech
ASR
Text:
adjust the temperature to 82 degrees fahrenheit on my reception room thermostat
NLU
Domain:
Airconditioner
Intent:
SetTemperatureToValueOnDevice
Slots:
{"device name": "reception room", "value": "82 degrees fahrenheit"}

Fig. 1. Spoken Language Understanding.

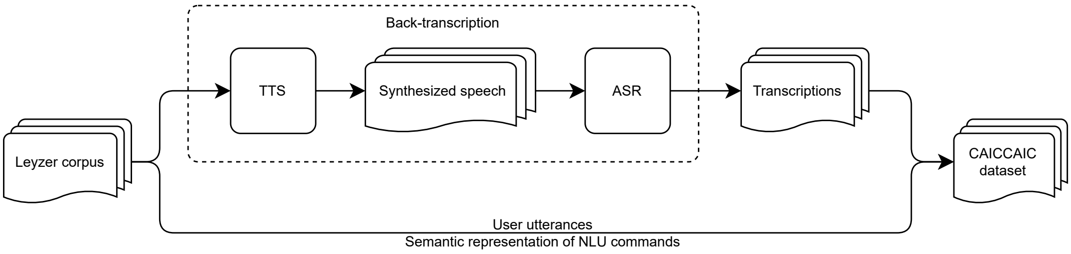

Fig. 2. Dataset preparation pipeline.

1) domain label: *Airconditioner*,
2) intent label: *SetTemperatureToValueOnDevice*,
3) slot values:
`{"device_name": "reception room", "value": "82 degrees fahrenheit"}`

For experimentation, we provide the validation dataset in the `dev-A` directory of the contest's repository. It was created using the same pipeline as the `train` dataset. The test data are located in `test-A` and `test-B` directories and contain only input values, while expected values hidden for contestants are used by the evaluation platform to score submissions.

IV. Baseline Models

We use XLM-RoBERTa Base [29] as a baseline model for intent detection and slot-filling. The XLM-RoBERTa model, also known as XLM-R, is a transformer-based multilingual masked language model that employs a multilingual masked language model (MLM) objective using only monolingual data. During training, streams of text from each language are sampled, and the model is trained to predict the masked tokens in the input. Subword tokenization is applied directly to raw text data using SentencePiece [30] with a unigram language model. The model does not use language embeddings, which allows it to handle code-switching better. It uses a large vocabulary size of 250K with a full softmax.

XLM-R was pre-trained on 2.5 TB of filtered CommonCrawl data containing 100 languages. This large-scale training led to significant performance gains for various cross-lingual transfer tasks. The model significantly outperforms multilingual BERT (mBERT) on various cross-lingual benchmarks.

Our baseline models were trained independently on the entire training set and optimized on the evaluation set. All baseline models have 12 layers, 768 hidden units, and 12 attention heads, totaling 270M parameters, and a size of 1.1 GB.

We use the `leyzer-fedcsis`[4] dataset from the Hugging Face Model Hub in the baseline training process. Each language-specific portion is processed individually, retaining only the `utterance` and `intent` columns. The processed datasets are then merged and split into training, validation, and testing sets. The model is defined for a sequence classification task using the `AutoModelForSequenceClassification` class, with the number of labels corresponding to the unique intents in the training dataset. Training hyperparameters were set to a learning rate of 2×10^{-5}, a training batch size of 16, a weight decay of 0.01, and 10 training epochs. Evaluations are performed after each epoch.

Finally, performance metrics such as accuracy and F_1 score are computed to assess the model's effectiveness in its classification task. The final epoch checkpoint evaluation results on the test set are presented in Table II in the "official baseline" row. All baseline intents models achieved results above 90% accuracy, with Spanish, Polish, and all-language models achieving above 95%. We analyzed misclassification errors and found that most of them could be resolved if a model resisted token distortion and could separate syntactically similar classes.

The error analysis of the intent recognition models for English, Spanish, and Polish languages reveals similarities and differences across the models. The *Spotify* domain tends to be the most problematic for all three languages, suggesting that these models may struggle with understanding and predicting

[4] https://huggingface.co/datasets/cartesinus/leyzer-fedcsis

TABLE I
UTTERANCE LENGTH DISTRIBUTION IN THE DATASET.

Locale	Split	Utterances	Mean Length	Length StdDev	Min	50%	Max
en-US	test	3344	9.951	4.322	1	9	33
	train	13022	9.345	3.718	1	9	33
	valid	3633	9.281	3.799	1	9	30
es-ES	test	3520	13.214	6.110	1	12	36
	train	15043	13.369	6.022	1	12	39
	valid	3546	13.152	5.948	1	12	39
pl-PL	test	3494	8.927	3.059	1	9	22
	train	12753	8.972	3.028	1	9	26
	valid	3498	9.018	3.053	1	9	23

TABLE II
EVALUATION RESULTS.

#	submission	description	pl-PL EMA	es-ES EMA	en-US EMA	Slot WRR	Intent accuracy	Domain accuracy	EMA
1	8850	mbart-large-50	**0.799**	**0.884**	0.569	**0.872**	0.916	0.963	**0.754**
2	8774	flan-t5-large	0.649	0.787	0.628	0.805	0.922	0.969	0.689
3	8347	*official baseline*	0.767	0.595	**0.686**	0.752	**0.945**	**0.980**	0.682
4	8812	flan-t5-large+context	0.648	0.794	0.548	0.770	0.898	0.955	0.665
5	8687	flan-t5-large	0.550	0.716	0.435	0.738	0.822	0.931	0.569
6	8846	flan-t5	0.495	0.503	0.479	0.692	0.898	0.958	0.493
7	8853	transformer t5	0.516	0.389	0.481	0.626	0.866	0.949	0.461
8	8869	dfd	0.469	0.457	0.411	0.627	0.675	0.959	0.446
8	8856	flan-t5-base	0.463	0.475	0.389	0.624	0.849	0.945	0.443
10	8847	all done	0.344	0.368	0.278	0.451	0.582	0.926	0.331

intents related to music streaming or the specific language used in this domain. *Slack* and *Console* domains also prove problematic for the English and Polish models, while for the Spanish model, the recognition of the *Airconditioner* and *Email* domains was the most challenging. Regarding specific intents, the English model has the most trouble with *ConsoleEdit* and *AddAlbumToPlaylist*, the Spanish model struggles with *PlayAlbumOfTypeByArtist* and *TurnOn*, and the Polish model with *SetPurposeOnChannel* and *PlayAlbumOfTypeByArtist*. These intents may be harder to recognize due to their semantic complexity, similarity to other intents, or underrepresentation in training data.

All models are available on the Hugging Face platform with details of how each model was trained and how to execute them:

- intent: en-US[5], es-ES[6], pl-PL[7], and all[8] that was trained and evaluated on all three languages together
- slot: en-US[9], es-ES[10], pl-PL[11]

V. Evaluation

The solutions for the task were submitted via the Gonito platform [31] challenge available at https://gonito.csi.wmi.amu.edu.pl/challenge/cnlps-caiccaic. For `in.tsv` file located in `test-A` directory, the participants were expected to provide `out.tsv` file in the same directory containing the predictions. The format of `out.tsv` was the same as the format of `train/expected.tsv`. Participants were allowed to use any publicly available data and models. Manual labeling was forbidden. A maximum of five submissions per day were allowed.

The submissions were scored using *Exact Match Accuracy* (EMA), i.e., the percentage of utterance-level predictions in which domain, intent, and all the slots are correct. Besides EMA scores, we also report the following auxiliary metrics:

- *domain accuracy*, i.e., the percentage of utterances with correct domain prediction;
- *intent accuracy*, i.e., the percentage of utterances with the correct intent prediction;
- *slot word recognition rate*, i.e., word recognition rate (WER) calculated on slot annotations, which is the percentage of correctly annotated slot values.

All scores were calculated using the GEval [32] library, which was also made available to participants for offline use.

VI. Results

We received 28 submissions from 9 teams. Table II presents the final ranking with cumulative metrics for all languages[12]. Notably, most submissions are based on pre-trained Transformer models [33] adapted to the task, with the Flan-T5 model [34] being the preferred choice. However, the winning

[5] https://huggingface.co/cartesinus/fedcsis-intent_baseline-xlm_r-en
[6] https://huggingface.co/cartesinus/fedcsis-intent_baseline-xlm_r-es
[7] https://huggingface.co/cartesinus/fedcsis-intent_baseline-xlm_r-pl
[8] https://huggingface.co/cartesinus/fedcsis-intent_baseline-xlm_r-all
[9] https://huggingface.co/cartesinus/fedcsis-slot_baseline-xlm_r-en
[10] https://huggingface.co/cartesinus/fedcsis-slot_baseline-xlm_r-es
[11] https://huggingface.co/cartesinus/fedcsis-slot_baseline-xlm_r-pl

[12] Detailed results can be found at https://gonito.csi.wmi.amu.edu.pl/challenge/cnlps-caiccaic/allentries.

TABLE III
MOST PROBLEMATIC FEATURES FOR THE WINNING MODEL COMPARED WITH THE BASELINE MODEL.

metric	**feature**	**count**	**metric δ**
Intent acc.	in<4>:hundred	357	-0.423
Intent acc.	in<4>:eight	224	-0.442
Intent acc.	in<4>:six	206	-0.408
Intent acc.	in<4>:images	417	-0.290
Intent acc.	out:FindImages[..][13]	31	-1.000
Intent acc.	in<2>:en-US	3344	-0.090
Intent acc.	in<4>:being	55	-0.582
Intent acc.	in<4>:small	48	-0.604
Intent acc.	out:	2013	-0.103

TABLE IV
MOST PROBLEMATIC FEATURES FOR THE BASELINE MODEL COMPARED WITH THE WINNING MODEL.

metric	**feature**	**count**	**metric δ**
EMA	out:subject	707	-0.808
EMA	exp:subject	711	-0.802
EMA	exp:SendEmail[..][14]	766	-0.756
EMA	out:SendEmail[..][15]	771	-0.752
EMA	out:message	835	-0.677
EMA	exp:message	837	-0.671
EMA	in<4>:un	982	-0.609
EMA	exp:to	1020	-0.593
EMA	in<4>:email	748	-0.686
EMA	out:to	885	-0.567

solution [35] used the mBART model [36] as its basis to train a joint, text-to-text model of domain, intent, and slots. This model achieved an Exact Match Accuracy of 0.754 across all the samples, with top results attained for Polish and Spanish NLU commands (0.799 and 0.884 EMA, respectively). It demonstrated outstanding performance in slot recognition with a slot WRR of 0.872 (0.067 better than the second-best solution). Although the winning solution performed well overall, it was within the accuracy of XLM-RoBERTa baseline models regarding domain and intent accuracy. This observation is intriguing and could be a valuable starting point for future research on developing joint models for domains, intents, and slots.

To gain more insight into the differences between the winning model and the baseline, we performed the analysis using the Geval tool [32]. Geval's "most worsening feature" function was used to analyze cases for which one of the models is problematic while the other behaves correctly. The function calculates the difference in a chosen metric between two models being compared, on cases containing a specific feature. The results are reported for cases for which the difference is statistically significant. Table III shows the features that had the most negative impact on the winning results compared to the baseline submission. It appears that numbers in their written form in English input are problematic for the mBART model. Also, it is not surprising to see that English inputs, in general, are easier for the baseline solution compared to the winning one, considering the overall results presented in Table II. Additionally, the mBART model has problems with one of the image-finding intents, which is consistent with the problematic word "images" in input sentences. Conversely, Table IV presents features that were problematic for the winning submission while being easier for the baseline model. The most problematic features are connected with the *Email* domain. It looks as baseline model has problems with identifying all kinds of slots of commands used for sending emails. These observations should prompt the authors of the winning submission and anyone else who wants to improve on these results to take a closer look into the specific causes of these particular types of errors and work towards addressing them.

REFERENCES

[1] D. Hakkani-Tür, F. Béchet, G. Riccardi, and G. Tur, "Beyond ASR 1-best: Using word confusion networks in spoken language understanding," *Computer Speech & Language*, vol. 20, no. 4, pp. 495–514, 2006. doi: https://doi.org/10.1016/j.csl.2005.07.005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0885230805000495

[2] F. Ladhak, A. Gandhe, M. Dreyer, L. Mathias, A. Rastrow, and B. Hoffmeister, "LATTICE RNN: Recurrent neural networks over lattices," in *Interspeech 2016*, 2016. [Online]. Available: https://www.amazon.science/publications/lattice-rnn-recurrent-neural-networks-over-lattices

[3] G. Tur, A. Deoras, and D. Hakkani-Tür, "Semantic parsing using word confusion networks with conditional random fields," in *Annual Conference of the International Speech Communication Association (Interspeech)*, Sep. 2013.

[4] X. Yang and J. Liu, "Using word confusion networks for slot filling in spoken language understanding," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[5] W. Ruan, Y. Nechaev, L. Chen, C. Su, and I. Kiss, "Towards an ASR error robust spoken language understanding system," in *Interspeech 2020*, 2020. [Online]. Available: https://www.amazon.science/publications/towards-an-asr-error-robust-spoken-language-understanding-system

[6] S. Sengupta, J. Krone, and S. Mansour, "On the robustness of intent classification and slot labeling in goal-oriented dialog systems to real-world noise," in *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*. Online: Association for Computational Linguistics, Nov. 2021. doi: 10.18653/v1/2021.nlp4convai-1.7 pp. 68–79. [Online]. Available: https://aclanthology.org/2021.nlp4convai-1.7

[7] J. Liu, R. Takanobu, J. Wen, D. Wan, H. Li, W. Nie, C. Li, W. Peng, and M. Huang, "Robustness testing of language understanding in task-oriented dialog," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, Aug. 2021. doi: 10.18653/v1/2021.acl-long.192 pp. 2467–2480. [Online]. Available: https://aclanthology.org/2021.acl-long.192

[8] Y. Nechaev, W. Ruan, and I. Kiss, "Towards NLU model robustness to ASR errors at scale," in *KDD 2021 Workshop on Data-Efficient Machine Learning*, 2021. [Online]. Available: https://www.amazon.science/publications/towards-nlu-model-robustness-to-asr-errors-at-scale

[9] B. Peng, C. Li, Z. Zhang, C. Zhu, J. Li, and J. Gao, "RADDLE: An evaluation benchmark and analysis platform for robust task-oriented dialog systems," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, Aug. 2021. doi: 10.18653/v1/2021.acl-long.341 pp. 4418–4429. [Online]. Available: https://aclanthology.org/2021.acl-long.341

[10] L. Feng, J. Yu, D. Cai, S. Liu, H. Zheng, and Y. Wang, "ASR-GLUE: A new multi-task benchmark for ASR-robust natural language understanding," *CoRR*, vol. abs/2108.13048, 2021. doi: 10.48550/arXiv.2108.13048. [Online]. Available: https://arxiv.org/abs/2108.13048

[11] M. Kubis, Z. Vetulani, M. Wypych, and T. Ziętkiewicz, "Open challenge for correcting errors of speech recognition systems," in *Human Language Technology. Challenges for Computer Science and Linguistics*, Z. Vetulani, P. Paroubek, and M. Kubis, Eds. Cham: Springer International Publishing, 2022. doi: 10.1007/978-3-031-05328-3_21. ISBN 978-3-031-05328-3 pp. 322–337.

[12] D. Koržinek, “Results of the PolEval 2020 Shared Task 1: Post-editing and Rescoring of Automatic Speech Recognition Results,” in *Proceedings of the PolEval 2020 Workshop*, 2020, pp. 9–14.

[13] H. Soltau, I. Shafran, M. Wang, A. Rastogi, J. Zhao, Y. Jia, W. Han, Y. Cao, and A. Miranda, “Speech Aware Dialog System Technology Challenge (DSTC11),” 2022. [Online]. Available: https://arxiv.org/abs/2212.08704

[14] T. Hayashi, S. Watanabe, Y. Zhang, T. Toda, T. Hori, R. Astudillo, and K. Takeda, “Back-translation-style data augmentation for end-to-end ASR,” in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018. doi: 10.1109/SLT.2018.8639619 pp. 426–433.

[15] A. Laptev, R. Korostik, A. Svischev, A. Andrusenko, I. Medennikov, and S. Rybin, “You do not need more data: Improving end-to-end speech recognition by text-to-speech data augmentation,” in *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2020. doi: 10.1109/CISP-BMEI51763.2020.9263564 pp. 439–444.

[16] R. Grishman and B. Sundheim, “Message Understanding Conference-6: A brief history,” in *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996. [Online]. Available: https://aclanthology.org/C96-1079

[17] A. Kilgarriff and M. Palmer, “Introduction to the special issue on SENSEVAL,” *Comput. Humanit.*, vol. 34, no. 1-2, pp. 1–13, 2000. doi: 10.1023/A:1002619001915. [Online]. Available: https://doi.org/10.1023/A:1002619001915

[18] A. Janusz, A. Krasuski, S. Stawicki, M. Rosiak, D. Ślęzak, and H. S. Nguyen, “Key risk factors for Polish state fire service: a data mining competition at knowledge pit,” in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. P. M. Ganzha, L. Maciaszek, Ed., vol. 2. IEEE, 2014. doi: 10.15439/2014F507 pp. 345–354. [Online]. Available: http://dx.doi.org/10.15439/2014F507

[19] A. Janusz, M. Przyborowski, P. Biczyk, and D. Ślęzak, “Network device workload prediction: A data mining challenge at knowledge pit,” in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, S. Agarwal, D. N. Barrell, and V. K. Solanki, Eds., vol. 21. IEEE, 2020. doi: 10.15439/2020KM159 pp. 77–80. [Online]. Available: http://dx.doi.org/10.15439/2020KM159

[20] A. Janusz, A. Jamiołkowski, and M. Okulewicz, “Predicting the costs of forwarding contracts: Analysis of data mining competition results,” in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. Maciaszek, M. Paprzycki, and D. Ślęzak, Eds., vol. 30. IEEE, 2022. doi: 10.15439/2022F303 p. 399–402. [Online]. Available: http://dx.doi.org/10.15439/2022F303

[21] M. Sowański and A. Janicki, “Leyzer: A dataset for multilingual virtual assistants,” in *Text, Speech, and Dialogue*, P. Sojka, I. Kopeček, K. Pala, and A. Horák, Eds. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-58323-1_51. ISBN 978-3-030-58323-1 pp. 477–486.

[22] C. Park, J. Seo, S. Lee, C. Lee, H. Moon, S. Eo, and H. Lim, “BTS: Back TranScription for speech-to-text post-processor using text-to-speech-to-text,” in *Proceedings of the 8th Workshop on Asian Translation (WAT2021)*. Online: Association for Computational Linguistics, Aug. 2021. doi: 10.18653/v1/2021.wat-1.10 pp. 106–116. [Online]. Available: https://aclanthology.org/2021.wat-1.10

[23] J. Guo, T. N. Sainath, and R. J. Weiss, “A spelling correction model for end-to-end speech recognition,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019. doi: 10.1109/ICASSP.2019.8683745 pp. 5651–5655.

[24] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, “FastSpeech 2: Fast and high-quality end-to-end text to speech,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=piLPYqxtWuA

[25] J. Kim, J. Kong, and J. Son, “Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech,” in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul. 2021, pp. 5530–5540. [Online]. Available: https://proceedings.mlr.press/v139/kim21f.html

[26] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, “Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018. doi: 10.1109/ICASSP.2018.8461368 pp. 4779–4783.

[27] G. Eren and The Coqui TTS Team, “Coqui TTS,” Jan. 2021. [Online]. Available: https://github.com/coqui-ai/TTS

[28] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 23–29 Jul 2023, pp. 28 492–28 518. [Online]. Available: https://proceedings.mlr.press/v202/radford23a.html

[29] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, “Unsupervised cross-lingual representation learning at scale,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020. doi: 10.18653/v1/2020.acl-main.747 pp. 8440–8451. [Online]. Available: https://aclanthology.org/2020.acl-main.747

[30] T. Kudo and J. Richardson, “SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018. doi: 10.18653/v1/D18-2012 pp. 66–71. [Online]. Available: https://aclanthology.org/D18-2012

[31] F. Graliński, R. Jaworski, Ł. Borchmann, and P. Wierzchoń, “Gonito.net - Open Platform for Research Competition, Cooperation and Reproducibility,” in *Branco, António and Nicoletta Calzolari and Khalid Choukri (eds.), Proceedings of the 4REAL Workshop: Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language*, 2016, pp. 13–20. [Online]. Available: http://4real.di.fc.ul.pt/wp-content/uploads/2016/04/4REALWorkshopProceedings.pdf

[32] F. Graliński, A. Wróblewska, T. Stanisławek, K. Grabowski, and T. Górecki, “GEval: Tool for debugging NLP datasets and models,” in *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Florence, Italy: Association for Computational Linguistics, Aug. 2019. doi: 10.18653/v1/W19-4826 pp. 254–262. [Online]. Available: https://www.aclweb.org/anthology/W19-4826

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS’17. Red Hook, NY, USA: Curran Associates Inc., 2017. ISBN 9781510860964 p. 6000–6010.

[34] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, “Scaling instruction-finetuned language models,” 2022.

[35] S. Jadczak and R. Jaworski, “Boosting conversational AI correctness by accounting for ASR errors using a sequence to sequence model,” in *Proceedings of the 18th Conference on Computer Science and Intelligence Systems*, 2023.

[36] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, “Multilingual denoising pre-training for neural machine translation,” *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 726–742, 2020. doi: 10.1162/tacl_a_00343. [Online]. Available: https://aclanthology.org/2020.tacl-1.47