Results of the Second SIGMORPHON Shared Task on Multilingual Grapheme-to-Phoneme Conversion

Lucas F.E. Ashby[*], **Travis M. Bartley**[*], **Simon Clematide**[†], **Luca Del Signore**[*], **Cameron Gibson**[*], **Kyle Gorman**[*], **Yeonju Lee-Sikka**[*], **Peter Makarov**[†], **Aidan Malanoski**[*], **Sean Miller**[*], **Omar Ortiz**[*], **Reuben Raff**[*], **Arundhati Sengupta**[*], **Bora Seo**[*], **Yulia Spektor**[*], **Winnie Yan**[*]
[*]Graduate Program in Linguistics, Graduate Center, City University of New York
[†]Department of Computational Linguistics, University of Zurich

Abstract

Grapheme-to-phoneme conversion is an important component in many speech technologies, but until recently there were no multilingual benchmarks for this task. The second iteration of the SIGMORPHON shared task on multilingual grapheme-to-phoneme conversion features many improvements from the previous year's task (Gorman et al. 2020), including additional languages, a stronger baseline, three subtasks varying the amount of available resources, extensive quality assurance procedures, and automated error analyses. Four teams submitted a total of thirteen systems, at best achieving relative reductions of word error rate of 11% in the high-resource subtask and 4% in the low-resource subtask.

1 Introduction

Many speech technologies demand mappings between written words and their pronunciations. In open-vocabulary systems—as well as certain resource-constrained embedded systems—it is insufficient to simply list all possible pronunciations; these mappings must generalize to rare or unseen words as well. Therefore, the mapping must be expressed as a mapping from a sequence of orthographic characters—*graphemes*— to a sequence of sounds—*phones* or *phonemes*.[1]

The earliest work on *grapheme-to-phoneme conversion* (G2P), as this task is known, used ordered rewrite rules. However, such systems are often brittle and the linguistic expertise needed to build, test, and maintain rule-based systems is often in short supply. Furthermore, rule-based systems are outperformed by modern neural sequence-to-sequence models (e.g., Rao et al. 2015, Yao and Zweig 2015, van Esch et al. 2016).

With the possible exception of van Esch et al. (2016), who evaluate against a proprietary database of 20 languages and dialects, virtually all of the prior published research on grapheme-to-phoneme conversion evaluates only on English, for which several free and low-cost pronunciation dictionaries are available. The 2020 SIGMORPHON Shared Task on Multilingual Grapheme-to-Phoneme Conversion (Gorman et al. 2020) represented a first attempt to construct a multilingual benchmark for grapheme-to-phoneme conversion. The 2020 shared task targeted fifteen languages and received 23 submissions from nine teams. The second iteration of this shared task attempts to further refine this benchmark by introducing additional languages, a much stronger baseline model, new quality assurance procedures for the data, and automated error analysis techniques. Furthermore, in response to suggestions from participants in the 2020 shared task, the task has been divided into high-, medium-, and low-resource subtasks.

2 Data

As in the previous year's shared task, all data was drawn from WikiPron (Lee et al. 2020), a massively multilingual pronunciation database extracted from the online dictionary Wiktionary. Depending on the language and script, Wiktionary pronunciations are either manually entered by human volunteers working from language-specific pronunciation guidelines and/or generated from the graphemic form via language-specific server-side scripting. WikiPron scrapes these pronunciatons from Wiktionary, optionally applying case-folding to the graphemic form, removing any stress and syllable boundaries, and segmenting the pronunciation—encoded in the Interna-

[1]We note that referring to elements of transcriptions as *phonemes* implies an ontological commitment which may or may not be justified; see Lee et al. 2020 (fn. 4) for discussion. Therefore, we use the term *phone* to refer to symbols used to transcribe pronunciations.

Proceedings of the Seventeenth SIGMORPHON Workshop on Computational Research

in Phonetics, Phonology, and Morphology,pages 115–125

tional Phonetic Alphabet—using the Python library `segments` (Moran and Cysouw 2018). In all, 21 WikiPron languages were selected for the three subtasks, including seven new languages and fourteen of the fifteen languages used in the 2020 shared task.[2]

In several cases, multiple scripts or dialects are available for a given language. For instance, WikiPron has both Latin and Cyrillic entries for Serbo-Croatian, and three different dialects of Vietnamese. In such case, the largest data set of the available scripts and/or dialects is chosen. Furthermore, WikiPron distinguishes between "broad" transcriptions delimited by forward slash (`/`) and "narrow" transcriptions delimited by square brackets (`[` and `]`).[3] Once again, the larger of the two data sets is the one used for this task.

3 Quality assurance

During the previous year's shared task we became aware of several consistency issues with the shared task data. This lead us to develop quality assurance procedures for WikiPron and the "upstream" Wiktionary data. For a few languages, we worked with Wiktionary editors who automatically enforced upstream consistency via "bots", i.e., scripts which automatically edit Wiktionary entries. We also improved WikiPron's routines for extracting pronunciation data from Wiktionary. In some cases (e.g., Vietnamese), this required the creation of language-specific extraction routines.

In early versions of WikiPron, users had limited means to separate out entries for languages written in multiple scripts. We therefore added an automated script detection system which ensures that entries for the many languages written with multiple scripts—including shared task languages Maltese, Japanese, and Serbo-Croatian—are sorted according to script.

We noticed that the WikiPron data includes many hyper-foreign pronunciations with non-native phones. For example, the English data includes a broad pronunciation of *Bach* (the surname of a family of composers) as /bɑːx/ with a velar fricative /x/, a segment which is common in German but absent in modern English. Furthermore, unexpected phones may represent simple human error. Therefore, we wished to exclude pronunciations which include any non-native segments. This was accomplished by creating phonelists which enumerate native phones for a given language. Separate phonelists may be provided for broad and narrow transcriptions of the same language. During data ingestion, if a pronunciation contains any segment not present on the phonelist, the entry was discarded. Phonelist filtration was used for all languages in the medium- and low-resource subtasks, described below.

4 Task definition

In this task, participants were provided with a collection of words and their pronunciations, and then scored on their ability to predict the pronunciation of a set of unseen words.

4.1 Subtasks

In the previous year's shared task, each language's data consisted of 4,500 examples, sampled from WikiPron, split randomly into 80% training examples, 10% development examples, and 10% test examples. As part of their system development, two teams in the 2020 shared task (Hauer et al. 2020, Yu et al. 2020) down-sampled these data to simulate a lower-resource setting, and one participant expressed concern whether the methods used in the shared task would generalize effectively to high-resource scenarios like the large English data sets traditionally used to evaluate grapheme-to-phoneme systems. This motivated a division of the data into three subtasks, varying the amount of data provided, as described below.[4]

High-resource subtask The first subtask consists of a roughly 41,000-word sample of Mainstream American English (`eng_us`). Participating teams were permitted to use any and all external resources to develop their systems except for Wiktionary or WikiPron. It was anticipated participants would exploit other freely available American English pronunciation dictionaries.

Medium-resource subtask The second subtask represents a medium-resource task. For each of the ten target languages, a sample of 10,000 words was used. Teams participating in this subtask were

[2]The fifteenth language, Lithuanian, was omitted due to unresolved quality assurance issues.

[3]Sorting by script, dialect, and broad vs. narrow transcription is performed automatically during data ingestion.

[4]Languages were sorted into medium- vs. low-resource subtasks according to data availability. For example, Icelandic was placed in the low-resource shared task simply because it has less than 10,000 pronunciations available.

permitted to use UniMorph paradigms (Kirov et al. 2018) to lemmatize or to look up morphological features, but were not permitted to use any other external resources. The languages for this subtask are listed and exemplified in Table 1.

Low-resource subtask The third subtask is designed to simulate a low-resource setting and consists of 1,000 words from ten languages. Teams were were not permitted to use any external resources for this subtask. The languages for this subtask are shown in Table 2.

4.2 Data preparation

The procedures for sampling and splitting the data are similar to those used in the previous year's shared task; see Gorman et al. 2020, §3. For each of the three subtasks, the data for each language are first randomly downsampled according to their frequencies in the Wortschatz (Goldhahn et al. 2012) norms. Words containing less than two Unicode characters or less than two phone segments are excluded, as are words with multiple pronunciations. The resulting data are randomly split into 80% training data, 10% development data, and 10% test data. As in the previous year's shared task, these splits are constrained so that inflectional variants of any given lemma—according to the UniMorph (Kirov et al. 2018) paradigms—can occur in at most one of the three shards. Training and development data was made available at the start of the task. The test words were also made available at the start of the task; test pronunciations were withheld until the end of the task. Some additional processing is required for certain languages, as described below.

English The Wiktionary American English pronunciations exhibit a large number of inconsistencies. These pronunciations were validated by automatically comparing them with entries in the CALLHOME American English Lexicon (Kingsbury et al. 1997), which provides broad ARPAbet transcriptions of Mainstream American English. Furthermore, a script was used to standardize use of vowel length and enforce consistent use of tie bars with affricates (e.g., /tʃ/ → /t͡ʃ/). However, we note that Gautam et al. (2021:§2.1) report several residual quality issues with this data.

Bulgarian Bulgarian Wiktionary transcriptions make inconsistent use of tie bars on affricates; for example, ц is transcribed as both /ts, t͡s/. Furthermore, the broad transcriptions sometimes contain allophones of the consonants /t, d, l/ (Ternes and Vladimirova-Buhtz 1990); e.g., л is transcribed as both /l, ɫ/. A script was used to enforce a consistent broad transcription.

Maltese In the Latin-script Maltese data, Wiktionary has multiple transcriptions of digraph *għ*, which in the contemporary language indicates lengthening of an adjacent vowel, except word-finally where it is read as [ħ] (Hoberman 2007:278f.). Rather than excluding multiple pronunciations, a script was used to eliminate pronunciations which contain archaic readings of this digraph, e.g., as pharyngealization or as [ɣ].

Welsh WikiPron's transcriptions of the Southern dialect of Welsh include the effects of variable processes of monophthongization and deletion (Hannahs 2013:18–25). Once again, rather than excluding multiple pronunciations, a script was used to select the "longer" pronunciation—naturally, the pronunciation without variable monophthongization or deletion—of Welsh words with multiple pronunciations.

5 Evaluation

The primary metric for this task was word error rate (WER), the percentage of words for which the hypothesized transcription sequence is not identical to the gold reference transcription. As the medium- and low-resource subtasks involve multiple languages, macro-averaged WER was used for system ranking. Participants were provided with two evaluation scripts: one which computes WER for a single language, and one which also computes macro-averaged WER across two or more languages. The 2020 shared task also reported another metric, phone error rate (PER), but this was found to be highly correlated with WER and therefore has been omitted here.

6 Baseline

The 2020 shared task included three baselines: a WFST-based pair n-gram model, a bidirectional LSTM encoder-decoder network, and a transformer. All models were tuned to minimize per-language development-set WER using a limited-budget grid search. Best results overall were obtained by the bidirectional LSTM. Despite the extensive GPU resources required to execute a

Armenian (Eastern)	arm_e	համադրություն	h ɑ m ɑ d ə ɾ u tʰ j u n
Bulgarian	bul	обосноваият	o̯ b o̯ s n o v a n i j ə t̪
Dutch	dut	konijn	k o: n ɛ i̯ n
French	fre	joindre	ʒ w ɛ̃ d ʁ
Georgian	geo	მოუქნელად	m ɔ u kʰ n ɛ l ɑ d
Serbo-Croatian (Latin)	hbs_latn	opadati	o p ă: d a t i
Hungarian	hun	lobog	l o b o g
Japanese (Hiragana)	jpn_hira	ぜんたいしゅぎ	d͡z ɕ n t a̠ i ɛ ɨʲ gʲ i
Korean	kor	쇠가마우지	sʰ w ɐ̞ g a̠ m a̠ u d͡ʑ i
Vietnamese (Hanoi)	vie_hanoi	ngừng bắn	ŋ ɨ ŋ ˨˩ ʔ ɓ a n ˧˥

Table 1: The ten languages in the medium-resource subtask with language codes and example training data pairs.

Adyghe	ady	кIэшIыхьан	t͡ʃʼ a ʃʼ ə ħ a: n
Greek	gre	λέγεται	l e ɟ e t e
Icelandic	ice	maður	m a: ð ʏ r
Italian	ita	marito	m a r i t o
Khmer	khm	ប្រហារ	p r ɑ h a:
Latvian	lav	mīksts	m iː k s t s
Maltese (Latin)	mlt_latn	minna	m ɪ n n a
Romanian	rum	ierburi	j e r b u rʲ
Slovenian	slv	oprostite	ɔ p r ɔ s t íː t ɛ
Welsh (Southwest)	wel_sw	gorff	g ɔ r f

Table 2: The ten languages in the low-resource subtask with language codes and example training data pairs.

per-language grid search, the best baseline was handily outperformed by nearly all submissions. This led us to seek a simpler, stronger, and less computationally-demanding baseline for this year's shared task.

The baseline for the 2021 shared task is a neural transducer system using an imitation learning paradigm (Makarov and Clematide 2018). A variant of this system (Makarov and Clematide 2020) was the second-best system in the 2020 shared task.[5] Alignments are computed using ten iterations of expectation maximization, and the imitation learning policy is trained for up to sixty epochs (with a patience of twelve) using the Adadelta optimizer. A beam of size of four is used for prediction. Final predictions are produced by a majority-vote ten-component ensemble. Internal processing is performed using the decomposed Unicode normalization form (NFD), but predictions are converted back to the composed form (NFC). An implementation of the baseline was provided during the task and participating teams were encouraged to adapt it for their submissions.

[5]The baseline was implemented using the DyNet neural network toolkit (Neubig et al. 2017). In contrast to the previous year's baseline, the imitation learning system does not require a GPU for efficient training; it runs effectively on CPU and can exploit multiple CPU cores if present. Training, ensembling, and evaluation for all three subtasks took roughly 72 hours of wall-clock time on a commodity desktop PC.

7 Submissions

Below we provide brief descriptions of submissions to the shared task; more detailed descriptions—as well as various exploratory analyses and post-submission experiments—can be found in the system papers later in this volume.

AZ Hammond (2021) produced a single submission to the low-resource subtask. The model is inspired by the previous year's bidirectional LSTM baseline but also employs several data augmentation strategies. First, much of the development data is used for training rather than for validation. Secondly, new training examples are generated using substrings of other training examples. Finally, the AZ model is trained simultaneously on all languages, a method used in some of the previous year's shared task submissions (e.g., Peters and Martins 2020, Vesik et al. 2020).

CLUZH Clematide and Makarov (2021) produced four submissions to the medium-resource subtask and three to the low-resource subtask. All seven submissions are variations on the imitation learning baseline model (section 6). They experiment with processing individual IPA Unicode characters instead of entire IPA "segments" (e.g., CLUZH-1, CLUZH-5, and CLUZH-6), and larger ensembles (e.g., CLUZH-3). They also experiment with input dropout, mogrifier LSTMs, and adaptive batch sizes, among other features.

Dialpad Gautam et al. (2021) produced three systems to the high-resource subtask. The Dialpad-1 submission is a large ensemble of seven different sequence models. Dialpad-2 is a smaller ensemble of three models. Dialpad-3 is a single transformer model implemented as part of CMU Sphinx. Gautam et al. also experiment with subword modeling techniques.

UBC Lo and Nicolai (2021) submitted two systems for the low-resource subtask, both variations on the baseline model. The UBC-1 submission hypothesizes that, as previously reported by van Esch et al. (2016), inserting explicit syllable boundaries into the phone sequences enhances grapheme-to-phoneme performance. They generate syllable boundaries using an automated onset maximization heuristic. The UBC-2 submission takes a different approach: it assigns additional language-specific penalties for mis-predicted vowels and diacritic characters such as the length mark /:/.

8 Results

Multiple submissions to the high- and low-resource subtasks outperformed the baseline; however, no submission to the medium-resource subtask exceeded the baseline. The best results for each language are shown in Table 3.

8.1 Subtasks

High-resource subtask The Dialpad team submitted three systems for the high-resource subtask, all of which outperformed the baseline. Results for this subtask are shown in Table 4. The best submission overall, Dialpad-1, a seven-component ensemble, achieved an impressive 4.5% absolute (11% relative) reduction in WER over the baseline.

Medium-resource subtask The CLUZH team submitted four systems for the medium-resource subtask. All of of these systems are variants of the baseline model. The results are shown in Table 5; note that the individual language results are expressed as three-digit percentages since there are 1,000 test examples each. While several of the CLUZH systems outperform the baseline on individual languages, including Armenian, French, Hungarian, Japanese, Korean, and Vietnamese, the baseline achieves the best macro-accuracy.

Low-resource subtask Three teams—AZ, CLUZH, and UBC—submitted a total of six systems to the low-resource subtask. Results for this subtask are shown in Table 6; note that the results are expressed as two-digit percentages since there are 100 test examples for each language. Three submissions outperformed the baseline. The best-performing submission was UBC-2, an adaptation of the baseline which assigns higher penalties for mis-predicted vowels and diacritic characters. It achieved a 1.0% absolute (4% relative) reduction in WER over the baseline.

8.2 Error analysis

Error analysis can help identify strengths and weaknesses of existing models, suggesting future improvements and guiding the construction of ensemble models. Prior experience using gold crowd-sourced data extracted from Wiktionary suggests that a non-trivial portion of errors made by top systems are due to errors in the gold data itself. For example, Gorman et al. (2019) report that a substantial portion of the prediction errors made by the top two systems in the 2017 CoNLL–SIGMORPHON Shared Task on Morphological Reinflection (Cotterell et al. 2017) are due to *target errors*, i.e., errors in the gold data. Therefore we conducted an automatic error analysis for four target languages. It was hoped that this analysis would also help identify (and quantify) target errors in the test data.

Two forms of error analysis were employed here. First, after Makarov and Clematide (2020), the most frequent error types in each language are shown in Table 7. From this table it is clear that many errors can be attributed either to the ambiguity of a language's writing system. For example, in both Serbo-Croatian and Slovenian the most common errors involve the confusion or omission of suprasegmental information such as pitch accent and vowel length, neither of which are represented in the orthography. Likewise, in French and Italian the most frequent errors confuse vowel sounds

	Baseline WER	Best submission(s)	WER
eng_us	41.91	Dialpad-1	**37.43**
arm_e	7.0	CLUZH-7	**6.4**
bul	**18.3**	CLUZH-6	18.8
dut	**14.7**	CLUZH-7	**14.7**
fre	8.5	CLUZH-4, CLUZH-5, CLUHZ-6	**7.5**
geo	**0.0**	CLUZH-4, CLUHZ-5, CLUZH-6, CLUZH-7	**0.0**
hbs_latn	**32.1**	CLUZH-7	35.3
hun	1.8	CLUZH-6, CLUZH-7	**1.0**
jpn_hira	5.2	CLUZH-7	**5.0**
kor	16.3	CLUZH-4	**16.2**
vie_hanoi	2.5	CLUZH-5, CLUZH-7	**2.0**
ady	**22**	CLUZH-2, CLUZH-3, UBC-2	**22**
gre	21	CLUZH-1, CLUZH-3	**20**
ice	12	CLUZH-1, CLUZH-3	**10**
ita	**19**	UBC-1	20
khm	34	UBC-2	**28**
lav	55	CLUZH-2, CLUZH-3, UBC-2	**49**
mlt_latn	19	CLUZH-1	**12**
rum	**10**	UBC-2	**10**
slv	49	UBC-2	**47**
wel_sw	**10**	CLUZH-1	**10**

Table 3: Baseline WER, and the best submission(s) and their WER, for each language.

	Baseline	Dialpad-1	Dialpad-2	Dialpad-3
eng_us	41.94	**37.43**	41.72	41.58

Table 4: Results for the high-resource (US English) subtask.

represented by the same graphemes.

Many errors may also be attributable to problems with the target data. For example, the two most frequent errors for English are predicting [ɪ] instead of [ə], and predicting [ɑ] instead of [ɔ]. Impressionistically, the former is due in part to inconsistent transcription of the *-ed* and *-es* suffixes, whereas the latter may reflect inconsistent transcription of the low back merger.

The second error analysis technique used here is an adaptation of a quality assurance technique proposed by Jansche (2014). For each language targeted by the error analysis, a finite-state covering grammar is constructed by manually listing all pairs of permissible grapheme-phone mappings for that language. Let C be the set of all such g, p pairs. Then, the covering grammar γ is the rational relation given by the closure over C, thus $\gamma = C^*$. Covering grammars were constructed for three medium-resource languages and four of the low-resource languages. A fragment of the Bulgarian covering grammar, showing readings of the characters б, ф, and ю, is presented in Table 8.[6]

Let $\mathcal{G}$ be the graphemic form of a word and let $\mathcal{P}$ and $\hat{\mathcal{P}}$ be the corresponding gold and hypothesis pronunciations for that word. For error analysis we are naturally interested in cases where $\mathcal{P} \neq \hat{\mathcal{P}}$, i.e., those cases where the gold and hypothesis pronunciations do not match, since these are exactly the cases which contribute to word error rate. Then, $P = \pi_o(\mathcal{G} \circ \gamma)$ is a finite-state lattice representing the set of all "possible" pronunciations of $\mathcal{G}$ admitted by the covering grammar.

When $\mathcal{P} \neq \hat{\mathcal{P}}$ but $\mathcal{P} \in P$—that is, when

[6]Error analysis software was implemented using the Pynini finite-state toolkit (Gorman 2016). See Gorman and Sproat 2021, ch. 3, for definitions of the various finite-state operations used here.

	Baseline	CLUZH-4	CLUZH-5	CLUZH-6	CLUZH-7
arm_e	7.0	7.1	6.6	6.6	**6.4**
bul	**18.3**	20.1	19.2	18.8	19.7
dut	**14.7**	15.0	14.9	15.6	**14.7**
fre	8.5	**7.5**	**7.5**	**7.5**	7.6
geo	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
hbs_latn	**32.1**	38.4	35.6	37.0	35.3
hun	1.8	1.5	1.2	**1.0**	**1.0**
jpn_hira	5.2	5.9	5.3	5.5	**5.0**
kor	16.3	**16.2**	16.9	17.2	16.3
vie_hanoi	2.5	2.3	**2.0**	2.1	**2.0**
Macro-average	**10.6**	11.4	10.9	11.1	10.8

Table 5: Results for the medium-resource subtask.

	Baseline	AZ	CLUZH-1	CLUZH-2	CLUZH-3	UBC-1	UBC-2
ady	**22**	30	24	**22**	**22**	25	**22**
gre	21	23	**20**	22	**20**	22	22
ice	12	22	**10**	12	**10**	13	11
ita	**19**	25	23	24	21	20	22
khm	34	42	32	33	32	31	**28**
lav	55	53	53	**49**	**49**	58	**49**
mlt_latn	19	19	**12**	16	14	19	18
rum	**10**	13	13	13	12	14	**10**
slv	49	90	50	59	55	56	**47**
wel_sw	**10**	40	**10**	13	12	13	12
Macro-average	25.1	35.7	24.7	26.3	24.7	27.1	**24.1**

Table 6: Results for the low-resource subtask.

the gold pronunciation is one of the possible pronunciations—we refer to such errors as *model deficiencies*, since this condition suggests that the system in question has failed to guess one of several possible pronunciations of the current word. In many cases this reflects genuine ambiguities in the orthography itself. For example, in Italian, *e* is used to write both the phonemes /e, ɛ/ and *o* is similarly read as /o, ɔ/ (Rogers and d'Arcangeli 2004). There are few if any orthographic clues to which mid-vowel phoneme is intended, and all submissions incorrectly predicted that the *o* in *nome* 'name' is read as [ɔ] rather than [o]. Similar issues arise in Icelandic and French. The preceding examples both represent global ambiguities, but model deficiencies may also occur when the system has failed to disambiguate a local ambiguity. One example of this can be found in French: the verbal third-person plural suffix *-ent* is silent whereas the non-suffixal word-final *ent* is normally read as [ɑ̃]. Morphological information was not provided to the covering grammar, but it could easily be exploited by grapheme-to-phoneme models.

Another condition of interest is when $\mathcal{P} \neq \hat{\mathcal{P}}$ but $\mathcal{P} \notin P$. We refer to such errors as *coverage deficiencies*, since they arise when the gold pronunciation is not one permitted by the covering grammar. While coverage deficiencies may result from actual deficiencies in the covering grammar itself, they more often arise when a word does not follow the normal orthographic principles of its language. For instance, Italian has borrowed the English loanword *weekend* [wikɛnd] 'id.' but has not yet adapted it to Italian orthographic principles. Finally, coverage deficiencies may indicate target errors, inconsistencies in the gold data itself. For example, in the Italian data, the tie bars used to indi-

eng_us	ɪ	ə	113	ɑ	ɔ	112	_	ʊ●	96	_	ɪ●	85	ɪ	i	76
arm_e	_	ə●	16	ə●	_	10	tʰ	d	6	d	tʰ	6	j●	_	3
bul	ɛ●	d͡	32	a	ə	31	ə	ɤ	30	_	◌̩	27	ə	a	25
dut	ə	eː	10	_	ː	10	ə	ɛ	9	eː	ə	8	z	s	8
fre	a	ɑ	6	_	●s	5	ɔ	o	5	e	ɛ●ʁ	3	_	●t	3
geo															
hbs_latn	_	ː	85	ː	_	76	_	◌̌	55	◌̌	◌̂	53	◌̌	_	52
hun	_	ː	6	h	ɦ	3	ʃ	s	2	ː	_	2			
jpn_hira	_	◌̥	20	_	◌̊	11	_	d͡	4	ː	●ɯ̟ᵝ	3	h	ɯ̟ᵝ	3
kor	_	ː	73	ː	_	28	ʌ̹	əː	23	ʰ	◌̈	9	əː	ʌ̹	6
vie_hanoi	_	w●	3	_	ɬ	3	_	w●ŋ͡m●	2	◌͡ɛ	●ɪ	2	_	ʔ●	2
ady	’	_	3	ː	_	3	ʃ	ʂ	3	ə●	_	2	a	ə	2
gre	ɾ	r	8	r	ɾ	3	i	j̟	3	m●	_	2	ɣ	g	2
ice	ː	_	2	◌̥	_	2	_	ː	2						
ita	o	ɔ	6	e	ɛ	5	j	i	3	◌͡	●	2	ɔ	o	2
khm	aː	i●ə	3	_	ʰ	3	_	●ɑː	2	ĕ	ɔ	2	ɑ	a	2
lav	◌̄	◌̂	11	_	◌̂	10	◌̀	_	9	◌̄	_	7	_	◌̀	4
mlt_latn	_	ː	5	_	ɪ●	2	ɐ	a	2	b	p	2	a	ɐ	2
rum	◌͡	●	2												
slv	◌́	◌̀	7	◌̀ː	_	6	◌́ː	_	6	_	◌́ː	5	ɛ	éː	4
wel_sw	ɪ	iː	3	ɪ	i̯	2	_	ɛ●	2						

Table 7: The five most frequent error types, represented by the hypothesis string, gold string, and count, for each language; ● indicates whitespace and _ the empty string.

…	
б	b
б	bʲ
б	p
…	
ф	f
ф	fʲ
…	
ю	ju
ю	u
…	

Table 8: Fragment of a covering grammar for Bulgarian; the left column contains graphemes and corresponding phones are given in the right column.

cate affricates are not always present, and many apparent errors are the result of gold pronunciations which omit a tie bar.

WER and model deficiency rate (MDR) is shown for select systems and three languages from the medium-resource subtask in Table 9, and Table 10 shows similar statistics for four low-resource languages. Note that by construction, one can obtain the coverage deficiency rate simply by subtracting MDR from WER. By comparing WER and MDR one can see the overwhelming majority of errors in these seven languages are model deficiencies, most naturally arising from genuine ambiguities in orthography rather than target errors (i.e., data inconsistencies).

To facilitate ensemble construction and further error analysis, we release all submissions’ test set predictions to the research community.[7]

9 Discussion

We once again see an enormous difference in language difficulty. One of the languages with the highest amount of data, English, also has one of the highest WERs. In contrast, the baseline and all four submissions to the medium-resource subtask achieve perfect performance on Georgian. This is a substantial change from the previous year’s shared task: with a sample roughly half the size of this year’s task, the best system (Yu et al. 2020) obtained a WER of 24.89 on Georgian (Gorman et al.

[7] https://drive.google.com/drive/folders/1Fer7UfHBnt5k-WFHsVXQO8ac3BvREAyC

	Baseline		CLUZH-5	
	WER	MDR	WER	MDR
bul	18.3	17.6	19.2	19.0
fre	8.5	7.5	7.5	6.8
jpn_hira	5.2	4.4	5.3	4.5

Table 9: WER and model deficiency rate (MDR) for three languages from the medium-resource subtask.

	Baseline		AZ		CLUZH-1		UBC-2	
	WER	MDR	WER	MDR	WER	MDR	WER	MDR
ady	22	22	30	23	24	21	22	22
gre	21	18	23	19	20	17	22	21
ice	12	9	22	17	10	7	11	5
ita	19	15	25	19	23	16	22	19

Table 10: WER and model deficiency rate (MDR) for four languages from the low-resource subtask.

2020:47). This enormous improvement likely reflects quality assurance work on this language,[8] but we did not anticipate reaching ceiling performance. Insofar as the above quality assurance and error analysis techniques prove effective and generalizable, we may soon be able to ask what makes a language hard to pronounce (cf. Gorman et al. 2020:45f.).

As mentioned above, the data here are a mixture of broad and narrow transcriptions. At first glance, this might explain some of the variation in language difficulty; for example, it is easy to imagine that the additional details in narrow transcriptions make them more difficult to predict. However, for many languages, only one of the two levels of transcription is available at scale, and other languages, divergence between broad and narrow transcriptions is impressionistically quite minor. However, this impression ought to be quantified.

While we responded to community demand for lower- and higher-resource subtasks, only one team submitted to the high- and medium-resource subtasks, respectively. It was surprising that none of the medium-resource submissions were able to consistently outperform the baseline model across the ten target languages. Clearly, this year's baseline is much stronger than the previous year's.

Participants in the high- and medium-resource subtasks were permitted to make use of lemmas and morphological tags from UniMorph as additional features. However, no team made use of resources. Some prior work (e.g., Demberg et al. 2007) has found morphological tags highly useful, and error analysis (§8.2) suggests this information would make an impact in French.

There is a large performance gap between the medium-resource and low-resource subtasks. For instance, the baseline achieves a WER of 10.6 in the medium-resource scenario and a WER of 25.1 in the low-resource scenario. It seems that current models are unable to reach peak performance with the 800 training examples provided in the low-resource subtask. Further work is needed to develop more efficient models and data augmentation strategies for low-resource scenarios. In our opinion, this scenario is the most important one for speech technology, since speech resources—including pronunciation data—are scarce for the vast majority of the world's written languages.

10 Conclusions

The second iteration of the shared task on multilingual grapheme-to-phoneme conversion features many improvements on the previous year's task, most of all data quality. Four teams submitted thirteen systems, achieving substantial reductions in both absolute and relative error over the baseline in two of three subtasks. We hope the code and data, released under permissive licenses,[9] will be used to benchmark grapheme-to-phoneme conversion and sequence-to-sequence modeling techniques more generally.

[8] https://github.com/CUNY-CL/wikipron/issues/138

[9] https://github.com/sigmorphon/2021-task1/

Acknowledgements

We thank the Wiktionary contributors, particularly Aryaman Arora, without whom this shared task would be impossible. We also thank contributors to WikiPron, especially Sasha Gutkin, Jackson Lee, and the participants of Hacktoberfest 2020. Finally, thank you to Andrew Kirby for last-minute copy editing assistance.

References

Simon Clematide and Peter Makarov. 2021. CLUZH at SIGMORPHON 2021 Shared Task on Multilingual Grapheme-to-Phoneme Conversion: variations on a baseline. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017. CoNLL–SIGMORPHON 2017 shared task: universal morphological reinflection in 52 languages. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 1–30.

Vera Demberg, Helmut Schmid, and Gregor Möhler. 2007. Phonological constraints and morphological preprocessing for grapheme-to-phoneme conversion. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 96–103.

Daan van Esch, Mason Chua, and Kanishka Rao. 2016. Predicting pronunciations with syllabification and stress with recurrent neural networks. In *INTERSPEECH 2016: 17th Annual Conference of the International Speech Communication Association*, pages 2841–2845.

Vasundhara Gautam, Wang Yau Li, Zafarullah Mahmood, Frederic Mailhot, Shreekantha Nadig, Riqiang Wang, and Nathan Zhang. 2021. Avengers, ensemble! Benefits of ensembling in grapheme-to-phoneme prediction. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*.

Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at the Leipzig Corpora Collection: from 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation*, pages 759–765.

Kyle Gorman. 2016. Pynini: a Python library for weighted finite-state grammar compilation. In *Proceedings of the SIGFSM Workshop on Statistical NLP and Weighted Automata*, pages 75–80.

Kyle Gorman, Lucas F.E. Ashby, Aaron Goyzueta, Shijie Wu, and Daniel You. 2020. The SIGMORPHON 2020 shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 40–50.

Kyle Gorman, Arya D. McCarthy, Ryan Cotterell, Ekaterina Vylomova, Miikka Silfverberg, and Magdalena Markowska. 2019. Weird inflects but OK: making sense of morphological generation errors. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, pages 140–151.

Kyle Gorman and Richard Sproat. 2021. *Finite-State Text Processing*. Morgan & Claypool.

Michael Hammond. 2021. Data augmentation for low-resource grapheme-to-phoneme mapping. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*.

S. J. Hannahs. 2013. *The Phonology of Welsh*. Oxford University Press.

Bradley Hauer, Amir Ahmad Habibi, Yixing Luan, Arnob Mallik, and Grzegorz Kondrak. 2020. Low-resource G2P and P2G conversion with synthetic training data. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 117–122.

Robert Hoberman. 2007. Maltese morphology. In Alan S. Kaye, editor, *Morphologies of Asia and Africa*, volume 1, pages 257–282. Eisenbrauns.

Martin Jansche. 2014. Computer-aided quality assurance of an Icelandic pronunciation dictionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 2111–2114.

Paul Kingsbury, Stephanie Strassel, Cynthia McLemore, and Robert MacIntyre. 1997. CALLHOME American English Lexicon (PRONLEX). LDC97L20.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Arya D. McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. UniMorph 2.0: universal morphology. In *Proceedings of the 11th Language Resources and Evaluation Conference*, pages 1868–1873.

Jackson L. Lee, Lucas F.E. Ashby, M. Elizabeth Garza, Yeonju Lee-Sikka, Sean Miller, Alan Wong, Arya D. McCarthy, and Kyle Gorman. 2020. Massively multilingual pronunciation mining with WikiPron. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4216–4221.

Roger Yu-Hsiang Lo and Garrett Nicolai. 2021. Linguistic knowledge in multilingual grapheme-to-phoneme conversion. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*.

Peter Makarov and Simon Clematide. 2018. Imitation learning for neural morphological string transduction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2877–2882.

Peter Makarov and Simon Clematide. 2020. CLUZH at SIGMORPHON 2020 shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 171–176.

Steven Moran and Michael Cysouw. 2018. *The Unicode Cookbook for Linguists: Managing Writing Systems using Orthography Profiles*. Language Science Press.

Graham Neubig, Chris Dyer, Yoav Goldberg, Austin Matthews, Waleed Ammar, Antonios Anastasopoulos, and Pengcheng Yin. 2017. DyNet: the dynamic neural network toolkit. arXiv:1701.03980.

Ben Peters and André F.T. Martins. 2020. One-size-fits-all multilingual models. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 63–69.

Kanishka Rao, Fuchun Peng, Haşim Sak, and Françoise Beaufays. 2015. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4225–4229.

Derek Rogers and Luciana d'Arcangeli. 2004. Italian. *Journal of the International Phonetic Association*, 34(1):117–121.

Elmar Ternes and Tatjana Vladimirova-Buhtz. 1990. Bulgarian. *Journal of the International Phonetic Association*, 20(1):45–47.

Kaili Vesik, Muhammad Abdul-Mageed, and Miikka Silfverberg. 2020. One model to pronounce them all: multilingual grapheme-to-phoneme conversion with a transformer ensemble. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 146–152.

Kaisheng Yao and Geoffrey Zweig. 2015. Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. In *INTERSPEECH 2015: 16th Annual Conference of the International Speech Communication Association*, pages 3330–3334.

Xiang Yu, Ngoc Thang Vu, and Jonas Kuhn. 2020. Ensemble self-training for low-resource languages: grapheme-to-phoneme conversion and morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 70–78.