Utilizing Phonetic Information for Textual NLP Tasks

Kenan Tang
The University of Chicago
kenantang@uchicago.edu

Abstract

The major tokenization methods in NLP split text into orthographic units by default. The phonetic information receives less attention when the NLP tasks do not directly involve audio, though such information is inseparable from the text when human uses language. Phonetic information is especially useful in analyses and tasks involving high orthographic but low phonetic distances. However, it is hard to choose among many datasets and grapheme-to-phoneme (G2P) systems. This survey provides an overview of such datasets and systems, emphasizing on how they differ in design, characteristics of target languages, and the need of downstream tasks.

1 Introduction

Word similarity plays an important role in NLP, as reflected by the rich history of orthographic tokenization and word embedding techniques (Mielke et al., 2021). However, when available, the text is usually considered as the only source of information, while the equally important phonetic information is ignored. Word association experiments have revealed that when given a word under conditions of stress (Clark, 1970) or distraction (Jung, 1975), people are more likely to respond with a phonetically rather than semantically related word. This example illustrates how phonetic information is weighed heavily by human language users. Also, phonetic similarity is important to concrete NLP tasks, including but not limited to named entity recognition or transliteration (§6.2.1) and text normalization (§6.1.2). Inspired by empirical observation or linguistic analyses (§5), utilizing phonetic information is useful for these tasks, especially when other methods fail to directly address linguistic traits or the scarcity of data.

This survey deals with the usage of phonetic information in various NLP tasks. In this survey, we will first explain the terminology (§2). In some scenarios, phonetic units provide more reasonable inter-unit distances than orthographic units (§3). In practice, however, there are multiple limits when one wants to obtain the information of phonetic units, either from text or audio. Thus, datasets and conversion systems from orthographic to phonetic units are proposed, but there are language-specific limits and drawbacks (§4). For high-resource languages, the resulting phonetic units can be directly applied to computational linguistic analysis (§5), often by utilizing the distance information. Moreover, even when the improved inter-unit distances are not directly used, the shared statistical strength of phonetic units can benefit NLP tasks (§6). Due to the high diversity across different languages, very different methods have been applied, and the combination between tasks and methods are not thoroughly searched. Therefore, we conclude by pointing out how some tasks may benefit from choosing alternative models (§7).

The selected works are obtained by keyword searches (phoneme, phone, articulatory features, etc.) on Google Scholar. The works are primarily from ACL Anthology and INTERSPEECH, excluding works that either (1) consider only audio instead of text or (2) mention phonetic information but do not use it as part of the model. The first criterion excludes most works in the large field of speech recognition and generation (such as Melnar and Liu (2006)) and Boulianne (2022), while the second criterion filters out irrelevant works returned from keyword searches (such as Mochihashi et al. (2009) who only use a transcript written in letters but not phonetic units). From the section of relevant works in these selected papers, some linguistics works are also selected in this survey when highly related.

It should be noted that the selection process is subjective and thus not exhaustive. This survey is unable to cover the full historical development of methods introduced in each section, but the readers

can refer to the section of relevant works in the selected works. Also, though without full rigor in the selection process, some general trends will be pointed out in this survey, such as the questionable but widely assumed independence of words (§4.1.1), the prevalence of phonemes instead of phones (§4.2.2), and the scarcity of both data and systems for low-resource languages (§4.3).

2 Phonemes and Phones

We begin the survey by defining phonemes and phones, which are crucial for any system that aims to output these units (§4.2). Both phonemes and phones are units to present pronunciation. Phoneme is usually regarded as the more perceptual units, while the phones are realizations of phonemes (Li et al., 2020a). The example below shows how the two representations can differ (Peters et al., 2017), with the standard way of showing phonemic representation in slashes and phonetic representation in brackets.

- pain: /peɪn/, [p^{h}eɪn]
- Spain: /speɪn/, [speɪn]

There are three key takeaways from the simple example here and many other examples shown in the selected works. First, the mapping between graphemes and phonemes can be many-to-many. For a letter in an English word, its pronunciation sometimes depends on the part-of-speech, which is determined by the sentence context. As an example from another language, a detailed rule set of German grapheme-phoneme correspondence provides a vivid illustration of the many-to-many mapping between graphemes and phonemes (Eisenberg and Fuhrhop, 2007). Even human could frequently err on these mappings, as further illustrated by a follow-up dataset of related spelling errors (Laarmann-Quante et al., 2016).

Second, even though the phonemes seem to contain all the pronunciation information of orthographic units and are thus widely applied to downstream tasks, a phoneme can in fact be realized in different phones (allophones). Phones are of finer granularity than phonemes, though they can be expressed in the same set of IPA symbols. To achieve this granularity, Moreno Bilbao et al. (1993) designs a system that performs grapheme-to-allophone conversion. Still, the choice of converting graphemes to phonemes instead of phones is much more common and somewhat natural, because of the parallel relationship between the graphemes and phonemes (Pulgram, 1951). Even for high resource languages such as English, phonetic representations are absent in the commonly used databases, such as Wiktionary[1]. Due to the somewhat ambiguous preference and data availability, it is necessary to distinguish between models that are trained on phonemes and those trained on phones.

Third, though only examples in English are shown here, the differences across languages deserve consideration. The need for language-specific treatments is widely supported by empirical evidence, especially those for the definition of word distance (§3) and the conversion from grapheme to phonemes (§4). While the motivation behind each language-specific improvement is briefly introduced in the survey, the readers can refer to the selected works for concrete examples of language differences.

For further reading, Hayes (2011) provides more examples, together with a detailed introduction of phonemes, phones, and articulatory features (§3.2).

3 Distance Metrics

Despite the sometimes ambiguous line between phonemes and phones, these two phonetic units do provide similarity information that are hard to obtain from orthographic units (Toutanova and Moore, 2002)[2]. Historically, different similarity metrics between phonetic units have been proposed, which aim to address different needs of downstream applications.

3.1 Without Features

Without a multi-dimensional feature being assigned to each phone, it is still natural to first define a binary similarity between pairs of phones. This basic definition of similarity already reveals the advantage of phonological representation over orthographical representation, resulting in improvements of performance. Intuitively, the binary similarity can intuitively be stored in a matrix, but the sparsity of this matrix also motivated alternative forms of storage and usage, giving rise to different names

[1] https://www.wiktionary.org/

[2] Interestingly, the examples used to illustrate the phonetic representation, such as the one in this survey from (Peters et al., 2017) and the one from (Li et al., 2020a), often do not show how phones better capture similarity than graphemes.

according to the context. As an example, the observation of commonly mistaken phone pairs can be summarized in the form of confusion sets (Wu et al., 2013). The presence of each pair is then a binary indicator of similarity. The improvement from the confusion sets could come from their incorporation into graph convolutional networks (Cheng et al., 2020).

Another way of defining similarity without features is to use hierarchical clusters of phonemes, which are either linguistically motivated (Dekel et al., 2004) or not (Salomon, 2001). The similarity between phonemes are characterized by the distance between them on a tree, all distances being integers which can be larger than 1.

Moreover, even when the edit distance between each pair of phones is set to one, the distance of graphemes, as calculated by phonemes from multiple languages, can be used as a somewhat continuous similarity metric. This is applied in the specific case where a single Chinese character has multiple pronunciations in Mandarin, Cantonese, Japanese, Korean, and Vietnamese (Hong et al., 2019). However, this method is language-specific and thus not universally applicable.

3.2 Articulatory Features

The previous section has shown that a continuous similarity metric is sometimes more desirable than a discrete one. Then, a natural way to define similarity is to first assign a feature vector to each phone, and then the similarity between each phone can be calculated by Manhattan distance or Euclidean distance.

The articulatory features describe how phones are pronounced by different configurations of the speech organs. This is a thoroughly investigated branch of linguistics. Intuitively, a feature vector could be used to summarizes such features. For example, a binary entry in the feature vector can be used to indicate whether a consonant is labial or not. However, despite ample linguistic evidence, there is no universally accepted definition of the articulatory feature vectors in NLP. On one hand, the dimension of the vector greatly varies across systems. As an example, Mortensen et al. (2016) uses 21 dimensional vectors for both vowels and consonants, whereas Ahmed et al. (2022) uses 3 for vowels and 10 for consonants. On the other hand, the value of vector entries can either be binary (Mortensen et al., 2016) or multi-valued (Kondrak and Sherif, 2006; Scharf and Hyman, 2011). To achieve some consistency, a conversion from the latter to the former is possible, for example from the multi-valued place of articulation to binary contrast pairs (Dunbar et al., 2015).

These features have the advantage that they can be assigned to IPA symbols independent of languages, reducing the labor of constructing the confusion sets mentioned above, which are defined only within one language. However, it is also possible to define language-specific, linguistically motivated features. Sometimes more than one set of features is proposed for a single language (Vieregge et al., 1984; Hoppenbrouwers and Hoppenbrouwers, 1988). The articulatory features can even be generated for sign languages (Brentari et al., 1998; Keane, 2014), an example of how feature vectors can be extremely specific to a language, while enjoying the simple mathematical form for convenient calculation. The basis of these works is the fact that sign languages utilizes its own set of muscle movements just like spoken language, so a parallel can be drawn.

Despite the capacity for feature vectors to incorporate linguistic considerations, the result is not guaranteed to be positive. For certain languages, it has been shown that longer vectors does not necessarily outperform shorter vectors (Nerbonne and Heeringa, 1997). Other than in this computational linguistic study, an ablation study on feature length is generally lacking from the downstream NLP models (§6).

3.3 Other Features

The vector form of the articulatory features also enable the easy concatenation with different features. For example, acoustic features can also be assigned to phones (Mielke, 2005). They capture more or less similar aspects of phones like the articulatory features, but nevertheless involves audio, which is not the primary focus of this survey.

Other than articulatory or acoustic similarities, a third perceived similarity captures information independently from the previous two (Strange, 2007). The perceived similarity is relatively underexplored, because it requires laborious pairwise annotation. The annotation involves multiple human subjects listening to phone pairs, which has been considered infeasible for even 1,000 phone pairs between two languages. However, it has the potential of benefiting in the cases involving asym-

metric language pairs, such as the pair of a native language and a learned foreign language.

As an alternative to the perceptual distances between phones in limited pairs of languages, in multi-lingual settings, phonological distance (Melnar and Liu, 2006) or other language-level distances (Littell et al., 2017) can be used together with the phone-level distance metrics. These data-driven features more easily extends to hundreds of languages without laborious annotation.

Since articulatory features are defined for individual phones, without consideration for their places in a word, it is also desirable if the extended features can be aware of a phone's location in a word. Such features, termed pseudo-features, are used in transliteration (Tao et al. (2006), see §6.2.1).

3.4 Hierarchical Features

The features mentioned above are all expressed linearly as vectors. Historically, articulatory features have also been organized into other structures, most commonly a tree. However, such structures have usually been ignored in feature extraction systems, and there are two good reasons to do so. On one hand, attempts of using other structured subword features instead of linear ones have not been successful (Hong et al., 2019), despite the increased cost from annotation and computation (Pawlik and Augsten, 2015). On the other hand, perhaps more fundamentally, the feature trees of articulatory features always have the same topology, which reduces the distance between trees to that between vectors as calculated by various tree kernels (Collins and Duffy, 2001; Gärtner, 2003; Shin and Kuboyama, 2013; Gärtner et al., 2004).

Overall, different definitions of phonetic distances with linguistic background have been proposed, and the proposal of new definitions are mostly limited by the cost of annotation or the diminishing gain when complicating structures. Thus, in downstream applications, the use of any specific phonetic distance definition is often limited by the language-specific availability (§6), and the comparison between different definitions is seldom carried out, if at all.

4 Datasets and Conversion Systems

Before using phones or phonemes together with text, a grapheme-to-phoneme (G2P) conversion system is required. While rule-based conversion systems have been proposed, many datasets have facilitated the development of machine learning models.

4.1 Datasets

Despite their abundance, the datasets usually suffer from the blur boundary between phones and phonemes (§2). A further complication is whether the phones can be sufficiently well predicted from independent words, without a sentence context.

4.1.1 Word-Level

Dictionaries that map one grapheme string of a word to one phoneme string are available from crowd-sourced websites, such as Wikipedia and Omniglot[3]. At the same level of word, the SIGMORPHONE 2020 Grapheme-to-Phoneme Conversion challenge provides datasets for multiple languages (Kann et al., 2020), and the datasets were further classified into languages with high, medium, and low resources in 2021 (Ashby et al., 2021). These datasets originate from WikiPron (Lee et al., 2020), which is a word-level dictionary without sentence context. When considered as a one-to-one mapping, the G2P conversion is straightforward as long as a dictionary is available. The value of these datasets is more in the training of models that can deal with out-of-vocabulary words (§4.2.1) than in developing the contextual awareness.

However, the mapping between graphemes and phonemes does depend on the context, for example when the pronunciation changes with the part of speech. Thus, a dictionary is less useful in this view, and other datasets are constructed. Still, it is arguable that some downstream tasks do not require a sentence-level model, and thus not a sentence-level dataset. For example, Wibowo et al. (2021) discover that 95% of the Indonesian colloquial words used on Twitter can be normalized without context. Despite the relative ease of normalizing colloquial expressions, such expressions do negatively impact the outcome of translation systems (Fujii et al., 2020), which indicate the usefulness of word-level datasets, even in the absense of sentence-level ones. As another example, for tasks involving named entities (§6.2.1), though sentence-level models are used when available, word-level models can arguably perform as well, since the named entities seldom change their part of speech.

[3] https://omniglot.com/

4.1.2 Sentence-Level

For developing English grapheme-to-phoneme systems that are aware of context, the following sentence-level datasets are commonly used: CMUDict, Pronlex, and NetTalk. For more languages, datasets such as CMU Wilderness (Black, 2019) and VoxClamantis (Salesky et al., 2020) are available. The latter provides more extra alignment information than the former. While these datasets do not emphasize language-specific traits, other datasets aim to represent language-specific difficulties to G2P, such as polyphone disambiguation (Park and Lee, 2020). This language-specific challenge is highly dependent on the context. Another potential drawback of the CMU Wilderness dataset is developed on audios of the Bible, a genre that much differs from the user generated content in some downstream tasks (§6.1.2).

Overall, most datasets include phonemes instead of phones. As probably the only exception, AlloVera (Mortensen et al., 2020) is a large dataset for grapheme-to-phone conversion. Therefore, it is worth noting that the abbreviation G2P in most cases refers to the conversion to phonemes but not phones, if not always.

4.2 Models and Systems

With the above datasets, data-driven models of G2P conversion can be built. However, the models for low-resource languages or early ones for high-resource languages are mostly dictionary-based and rule-based.

4.2.1 Grapheme to Phoneme

Many G2P models have been developed for high-resource languages. For English, the most basic systems, such as the KTH test-to-speech system (Hunnicutt, 1980) or eSpeak[4], directly uses dictionaries and exception rules. Due to the presence of OOV words, models also have to processed grapheme on a subword level, which is not a trivial task, especially if further complicated by the consideration of sentence context (§4.1.2). When applied at letter-level, the HMM model (Taylor, 2005) performs poorly because of the strong dependence of a grapheme's pronunciation on its context. To address the low accuracy without requiring the context, models improved by taking different subword units into account, such as letter substrings (Jiampojamarn et al., 2008) or syllables (Bartlett et al., 2008).

From the orthogonal perspective of architectures, G2P systems haved utilized the CART tree (Black and Lenzo, 2001), maximum entropy models (Chen et al., 2003), LSTM models with attention (Toshniwal and Livescu, 2016), Bi-LSTM (Yao and Zweig, 2015; Park and Lee, 2020; Rao et al., 2015), CNN (Yolchuyeva et al., 2019), or Transformer (Yolchuyeva et al., 2020). However, a thorough search is not carried out for the combination of different subword units and model architectures.

For other high-resource languages, some improvements result from directly addressing language-specific errors from baseline systems, such as vowel errors (Lo and Nicolai, 2021), stress errors (Dou et al., 2009), or schwa deletion (Choudhury et al., 2004) and schwa epenthesis (Wasala et al., 2006). Motivated by the distinction between vowels and consonants, it is possible to process consonants and vowels in different steps of the pipeline (Mendonça and Aluísio, 2014)[5]. Similar to the case of English, some improvements also come from using language-specific units, such as the syllables in Bahasa Indonesia, which are richer than those in English and are accompanied by extra information of syllabification points (Suyanto, 2019).

After G2P models for multiple languages have become available, they are sometimes incorporated into a single systems for convenient usage. EpiTran is such a system (Mortensen et al., 2018). With the same idea of converting multiple languages to the same phonetic script, some other systems convert orthographic units to proxies of IPA phonemes, most commonly romanization. Two example systems that perform romanization for multiple languages are uroman (Hermjakob et al., 2018) and ScriptTranscriber (Qian et al., 2010). As another proxy to address the pronunciation change under code-mixing context (Çetinoğlu et al., 2016), the common label set (CLS) for Hindi-English code-mixing has been proposed (Ramani et al., 2013; Thomas et al., 2018). Sometimes, the romanized Unicode names of graphemes can even be utilized (Deri and Knight, 2016). The Unicode name captures similarity of phones across languages, but only applies to a small set of phones and genealogically related languages.

While these systems aim to support multiple languages, they usually do not incorporate the SOTA

[4] http://espeak.sourceforge.net/

[5] In fact, the distinction between consonants and vowels is a recurring theme (§3.2, 4.2.3, 6.1.1, 6.1.2), since it marks the most apparent distinction between phones.

models for part of the languages at the time they were built. Even for English, only the Phonetisaurus system is designed with the awareness of incorporating SOTA systems (Novak et al., 2016). Phonetisaurus also has the advantage of being able to produce a controllable number of candidates under its weighted finite-state transducer paradigm, which can be helpful for text normalization (§6.1.2). Thus, an ideal G2P system in the future should be able to dynamically incorporate the latest advances in each language, and also to incorporate for the same language different systems, whose performance differ depending on the downstream tasks. Currently, the systems closest to this ideal are dictionary-based and rule-based ones, for which the dictionaries and rule-sets can easily be replaced.

4.2.2 Phoneme to Phone

Since a phoneme can be realized by several different phones in a language, there is also motivation for providing phones as finer-grained units. Allosaurus is such a system (Li et al., 2020b).

This survey mainly deals with tasks in which at least one step involves text. In these cases, a known phone or phoneme inventory is always necessary. However, it is also possible to unsupervisedly learn the phonetic units from audio (Varadarajan et al., 2008). This methods results in an alternative inventory of subword units, the effectiveness of which in replacing phones in downstream tasks has not been tested. Still, these units have finer granularity than phonemes, because they correspond to realizations of the pronunciation rather than the perception.

4.2.3 Phoneme to Articulatory Features

After G2P, the next step in the pipeline is the conversion from phonemes to features. Many systems convert phonemes to articulatory features in a single language under certain tasks (§3.2), which are not necessarily applicable to different languages. For universal applicability, PanPhon is a system that converts phonemes to language-independent articulatory features (Mortensen et al., 2016). There are other multi-lingual systems that convert phonemes to articulatory features with different vector entries (Ahmed et al., 2022). However, comparisons of downstream task performance have not been made between these systems.

If allophones are ignored, the conversion from phoneme to articulatory features is a straightforward one-to-one mapping. If allophones are considered, a one-to-many mapping rule could intuitively convert a phoneme to multiple allophones, which should be the intermediate step of the pipeline from text to phonetic features. However, there are multiple alternatives that deal with phones, but without phonemes as the intermediate step. When the goal is to obtain phone and then articulatory features from audio, the two can be jointly extracted instead of in a pipeline (Zhu et al., 2021). Motivated by linguistic evidence (Markov, 2006), other systems directly try to recover phonetic features from a phone sequence, without knowing any feature categories beforehand. Such phoneme clustering systems are able to recover the distinction between consonants and vowels (Kim and Snyder, 2013b), sometimes also coronals (Hulden, 2017). The fact that the recovered phonetic features only corresponds well to limited categories reinforces the need for ablation studies on the length of articulatory feature vectors (§3.2).

As another alternative to phonetic features, the phoneme embeddings could be learned. With only the phoneme sequences, phoneme-level analogies to word embedding models such as word2vec have been used (Silfverberg et al., 2018). When audio is available, it is possible to learn either the embedding of individual phones (Synnaeve et al., 2014) or of the phone sequences that correspond to whole words (Hu et al., 2020). When text is available, weights from pre-trained RoBERTa can be utilized (Sundararaman et al., 2021). However, the dilemma to this approach is that phoneme embeddings can only be obtained for high-resource languages, for which the word embeddings already lead to strong performances and do not necessitate the usage of phoneme embeddings. In contrast, for the low resource languages that do need phoneme embeddings, such embeddings can only be constructed from linguistic knowledge due to the lack of data.

4.3 The Problem of Coverage

Most of the datasets and systems mentioned above better support high-resource languages (specifically English) than low-resource languages. The development of models on low-resource languages is especially difficult, as can be seen from the practice that when annotated data is scarce, researchers sometimes have to use a fraction of the high-resource language datasets to approximate a low-resource setting (Yadav et al., 2021; Salesky and Black, 2020). To increase the dataset size,

some methods can help speeding up the data annotation process (Dwyer and Kondrak, 2009; Kim and Snyder, 2013a). However, the adaptation of high-resource language models is indeed inevitable in some extreme scenarios when the resource cannot be easily expanded, such as reconstructing phonology for a dead language (Smith, 2007).

5 Computational Linguistic Analysis

Given a reliable G2P system, it is then possible to perform computational linguistic analysis, either within one language or across multiple languages. Due to that phones reflect similarity better than graphemes, the analyses are naturally centered around similarities defined under different contexts. Some of the similarities can be potentially utilized in downstream NLP tasks.

5.1 Monolingual

The similarity within a single language is usually defined between acoustic or textual word forms. The audio is only available for contemporary word forms, which limits its usage in both computational linguistics analysis and NLP applications (§6.1.2). Thus, sometimes the acoustic similarities are necessarily extracted from text.

5.1.1 Word-Level

Within a language, words with specific common structures can better reveal such structural similarity under phonetic representations. As an example, the analysis of reduplicative words can benefit from phonetic information (Le Hong et al., 2009). Though a reduplicative word consist of similar orthographic units, the edit distance between them can still be high that the reduplicative word cannot be distinguished from ordinary words.

However, the word-level analyses are not limited to similar word pairs. Given articulatory features, (Choudhury et al., 2007) tried to explain the evolution of languages with a Multi-Objective Genetic Algorithm, in which the objectives for optimization are defined using articulatory effort. Here, the pronunciations are actively evolving. This is a fundamentally different perspective from the pronunciation stability in different stages of evolution. The latter perspective is the key to pronunciation-aware models of historical text normalization (§6.1.2).

Also, a common limit in these two examples is that their models do not have a strong predictive power that enables either the detection of word variants (§6.1.2) or the production of them (§6.2.1). Therefore, they fall under the category of computational linguistic analysis but not NLP applications.

5.1.2 Phone(me)-Level

Phonetic information is necessary for analyzing acoustic-phonetic measurements (Salesky et al., 2020), such as formant frequencies for vowels and spectral shapes for sibilants (fricative consonants). Extracted from audio, these measurements contribute to a deeper understanding of articulatory features (§3.2) in different languages. However, the extraction is impossible in the common scenario when audio is only aligned with text instead of phones. A reliable G2P pipeline can bridge this gap.

Similar to the explanation of word evolution in the previous subsection, articulatory features can be used to explain the formation of vowels (De Boer, 2000). Again, the prediction power in phone-level analysis is even weaker, because new phones in a language is much less likely to emerge than is words in a small time scale.

5.2 Multilingual

For multiple languages, the similarity is either defined at a language level or a word level. For both levels, the word pairs involved are both semantically and phonetically close, though orthographically distant. While semantic similarity is necessary for the purpose of analysis here, other interesting phenomena involve word pairs that are both semantically and orthographically distant, but phonetically close (§6.1.3).

5.2.1 Language-Level

The phonetic similarity can be easily used to derive the similarity between languages, including consistency of cross-lingual phonological features (Johny et al., 2019), surprisal-duration trade-off (Pimentel et al., 2021), and acoustic similarity (Wu et al., 2021a). However, metrics in these analyses are defined over sentences, a definition that requires sentence-level datasets (§4.1.2) and makes pronunciation dictionaries useless (§4.1.1).

The compared objects can also be dialects instead of languages (Nerbonne and Heeringa, 1997; Prokić and Van de Cruys, 2010), in which case phones have to be used instead of phonemes. This kind of analysis necessitates the development of grapheme-to-allophone conversion systems (§4.2.2).

As introduced in §3.3, these language-level analyses result in features that can be used together with the articulatory features in downstream tasks.

5.2.2 Word-Level

The phonetic units can be used to analyze loan words (Wu et al., 2021b), which intuitively share similar pronunciations and the same meaning. In this work, phonetic information is not incorporated into the model, but are qualitatively considered as one criterion for dataset construction. The time scale of this work is also inevitably long, because the development of loan words could take centuries.

6 NLP Tasks

Most of the computational linguistic analyses focus on the history of languages, which may be less interesting because they can hardly be used for making predictions in a similar time scale of centuries. However, there are scenarios where language evolve much more rapidly, so that the models under a similar evolutionary perspective can be useful, and the phonetic information becomes indispensable. As pointed out in §5.1.1, the key is the stability of phonetic representation of word forms. The phonetic information can also be helpful for tasks unrelated to language evolution.

6.1 Monolingual

Different from the weak predictive power of CL analysis, the monolingual application of phonetic information generates concrete outputs, including word segmentation, normalized text, and detected text segments of humor.

6.1.1 Phonotactic Segmentation

Phonotactics is the study of possibility of a certain phoneme sequence appearing in a word. Similar to language models, a common choice for phonotactic modelling is the n-gram phone model. Though the number of phones is much smaller than the number of words in usual dictionaries, the n is usually larger (at least 3) than in language models because of the need to capture common phoneme sequences. Therefore, the back-off mechanism is usually necessary.

Phonotactics can help the segmentation of a continuous sequence of orthographic units into words, when dictionaries are not available. As its foundation, phonotactic segmentation requires a G2P system that does not utilize known dictionaries (§4.2.1). The independence from dictionaries makes this segmentation method especially helpful for low-resource languages, especially those without a orthographic representation, but OOV words from high-resource languages also benefit from it.

The early works in phonotactic segmentation do not utilize any articulatory features. As an example, Fleck (2008) tests phonotactic segmentation on English and Arabic. Common prefixes and suffixes are used to bootstrap the naive n-gram model. This idea of utilizing units longer than a single phoneme is similar to the attempt of using longer subword units in the G2P conversion (§4.2.1), but disables the usage of articulatory features that are defined on individual phones.

The articulatory features can indeed be helpful. Using articulatory features, Shcherbakov et al. (2016) have shown the good performance of phonotactic segmentation on not only low-resource languages, but also English. Since phonotactic methods is more sensitive to the distinction between vowels and consonants than other articulatory features, the vowels and consonants are abstracted by their category in an abstracted n-gram model. However, the segmentation based on such features can be highly language-independent, since certain types of diphones may mark word boundaries in one language but not another (Daland and Zuraw, 2013).

The task of phonotactic segmentation is interesting mainly for two reasons. On one hand, phonotactic segmentation lies between NLP and CL. Though phonotactic segmentation can theoretically be used to split audio for downstream task, it is also important for modeling child language acquisition (Brent and Cartwright, 1996). On the other hand, phonotactic segmentation lies between monolingual and multilingual applications. Though the phonotactic segmentation models are usually tested on one language, one model has to be valid cross-linguistically, so that it could be a candidate for understanding human language acquisition (Loukatou et al., 2019).

6.1.2 Normalization

Text normalization tasks include the normalization of user-generated content (UGC), spelling correction, and historical text canonicalization (Jurish, 2010). In these tasks, it is always assumed that the input and output text share the same pronunciation. The three tasks are not mutually exclusive, despite

being named differently depending on the context.

Phonetic issues in UGC can be categorized into 4 categories, namely fusions, omissions, equivalents, and onomatopoeias (De Clercq et al., 2013). The UGC normalization can be performed by using context-independent phonetic signatures extracted from individual words (Jahjah et al., 2016). Here, the word signature refers to the IPA representation of English words, after the simplification of removing vowels and duplicate consonants. This method effectively deals with the latter 3 of the 4 categories mentioned above. This work does not utilize a formal subword-level G2P system, probably because of the fuzzy nature of unnormalized text. In another example of fuzziness, as many as 10 phoneme strings could be used for a single OOV word (Arshi Saloot et al., 2015), suggesting the high tolerance of text normalization to errors from G2P systems. The high tolerance is further illustrated, when Jahjah et al. (2016) explicitly use an English dictionary with more pronunciation entries than words, but they only pair one pronunciation with one word, violating the dependence of pronunciation on the context sentence (§4.1.1). With the high tolerance, a letter-level G2P model is still required to deal with all the OOV words. When only word-level but not letter-level G2P models are available, the normalization of OOV words can only rely on grapheme edit distance (Alegria et al., 2015).

The spelling correction can be performed with the help of phonemes from the metaphone phoneme transformation rules (Philips, 1990) as by `jazzy`[6]. In early works that start to be rule-independent, spelling correction is done by the noisy channel model (Toutanova and Moore, 2002), following an n-gram G2P module (Fisher, 1999). The noisy channel could also be applied using the units of syllables (Xu et al., 2015).

As models advanced, spelling correction has been improved by incorporating phone information as a loss in large-scale pre-trained models (Zhang et al., 2021). Due to the high resource nature, the embedding of word pronunciation are represented by embeddings jointly trained with word embeddings, instead of using individual phones or linguistically motivated articulatory features. Also, instead of concatenation, the word embedding and the phone embeddings are mixed by a linear combination weighted by the predicted spelling error probability. A similar mixing strategy is adopted by Liu et al. (2019), who alternatively get the word pronunciation embedding by averaging phone embeddings, and then linearly combine the pronunciation embedding with the text embedding weighted by a fixed hyperparameter. However, the usage of large pre-trained models for high resource languages again falls into the dilemma pointed out in §4.2.3.

For historical text, the motivation of canonicalization is mainly the convenience of searching within the text using modern spellings, which is very different from the historical spellings. As an example, Robertson and Willett (1993) summarize the different forms of alternative spellings in historical English. Because different spellings share similar pronunciations, phonetic information are often used for normalization. If the historical word forms are assumed to share the same pronunciation with modern word forms, a historical word can be converted into phoneme strings by a G2P system and then be compared with pronunciations of similar modern words. Then, one of the similar modern words will hopefully be the corresponding word (Jurish, 2010). If a P2G system is available, the phonemic representation of a historical word can also be directly transformed into modern words. The rules of phonetic evolution can be incorporated into this process (Porta et al., 2013). However, some results show that a purely grapheme-to-grapheme model using the Phonetisaurus framework would suffice, without explicitly involving phonemes as an intermediate step (Etxeberria et al., 2016a,b).

6.1.3 Humor Detection

Some UGC use figurative language to achieve rhetorical effects, which is better detected in their original forms than normalized. Among the different uses of figurative language (Abulaish et al., 2020), humor most apparently requires phonetic information for its detection. The current humor detection systems uses phonetic information by detecting alliteration and rhyming (Yang et al., 2015; Mihalcea and Strapparava, 2005). These two rhetoric devices are used only either at the start or the end of each word. While not formally analyzed, G2P systems presumably err at different rates in the two locations. Thus, the application in humor detection suggests a finer-grained metric than word error rate (WER) and phoneme error rate (PER). The metric should be aware of the po-

[6] `http://jazzy.sourceforge.net`

sition of phonemes in a word. Evaluated under this metric, a good G2P system in the humor detection pipeline should ideally perform well also on the middle of words, so other underexplored rhetoric devices, such as assonance, can be investigated.

It is worth noting that sound-based humor is exactly an example of minimizing the phonetic distance while maximizing the semantic distance, in order to achieve a startling effect (Attardo, 2010). Other applications discussed in this survey usually involve semantically close pairs, which do not necessitate the use of phonetic distance, if the orthographic distance already performs well.

6.2 Multilingual

The similarity between multiple languages took centuries to arise. Still, a subset of the similarity phenomena is the result of rapid language evolution. In these scenarios, OOV words are commonly produced, which are hard to handle by dictionary-based G2P systems (§4.2.1). To handle OOV words, the NLP systems for these tasks necessarily use subword textual units as their input.

6.2.1 Named Entities

The tasks involving named entities are inherently bilingual, if not involving more than 2 languages due to multiple origins of the named entities (Waxmonsky and Reddy, 2012). The named entities are either recognized in existing text (NER), or generated by transliteration.

On the recognition side, the use of phonetic information mainly helps NER in a single low-resource language (Mortensen et al., 2016), or in the transfer between orthographically distant languages (Bharadwaj et al., 2016). Despite its intuitive relevance to the task, phonetic information is not widely reported to improve NER performance.

On the generation side, transliteration almost always involves a P2G step. Sometimes after the P2G conversion, the granularity is made coarser by converting phonemes to generalized initials and finals (GIFs) (Virga and Khudanpur, 2003), reflecting the large differences between languages (§2). Though the transliteration task could be completed at the word level, the model could be made somewhat aware of the sentence context, if the model-generated transliteration candidates are ranked or corrected by their web-mined alternatives (Jiang et al., 2007; Yang et al., 2008). This approach largely differs from constructing a sentence-level dataset for training context-aware models (§4.1.2). In rare exceptions, a transliteration model does not use phoneme as an intermediate step, but still uses phonotactic knowledge to post-process the output (Ekbal et al., 2006).

It is worth noting that the phonetic similarity metric that helps transliteration may be different from those used in other applications. As an example, to improve models that fully rely on grapheme information (Li et al., 2004), Pervouchine et al. (2009) used phonetic information to aid grapheme alignment. Instead of using a similarity matrix derived from pre-defined feature vectors, their method results in a data-driven similarity matrix between phonemes (§3.3) under the specific setting of transliteration. Tao et al. (2006) discovered that articulatory features alone are not able to determine whether a consonant would be deleted during transliteration. Some consonants are deleted because they are not allowed to appear as codas in the target languages. Such information is included as pseudo-features that are used together with the articulatory features. This phenomenon can be seen as another evidence to support the necessity of finer-grained evaluation metrics for G2P accuracy (§6.1.3).

6.2.2 Intent Classification

While the use of phonetic information is not intuitively necessary here, intent classification for low resource languages can also benefit from it, just like in the case of NER. It might be expected that due to data scarcity, the linguistically motivated phonetic features are preferable (§4.3). However, when the dataset size is reasonably large, either training phone embeddings from scratch in an LSTM (Gupta et al., 2021) or using pre-trained phone embeddings (Yadav et al., 2021) outperforms linguistic embeddings from PanPhon (Mortensen et al., 2016).

In the work of Yadav et al. (2021), the pre-trained phone embeddings come from an ASR system Allosaurus (Li et al., 2020a). The phoneme embeddings pre-trained by text normalization models (§6.1.2) can in principle be used here, and it is interesting to see if any performance gain will result from a different pre-training procedure. Also, though the PanPhon linguistic embedding (Mortensen et al., 2016) is reported to perform worse with reasonable amount of data, it is unclear whether this conclusion applies to alternative definitions of the articulatory feature set, especially language-specific ones (§3.2).

6.2.3 Potential of Replacing Text

Though this survey is primarily concerned with how phonetic information can help purely textual tasks, phones together with their features can even completely replace orthographical information, such as in directly translating source speech to target speech without intermediate text (Salesky and Black, 2020). This paradigm is very different from the one in normalization tasks (§6.1.2), where the phonetic information are considered more of a latent representation of variable spellings. However, here the power of phonetic information is more strongly suggested.

7 Conclusion

In conclusion, phonetic information has been shown to be helpful for many textual tasks. To achieve such improvements, a G2P conversion system is necessary, but the existing systems are not ideal for multiple reasons. Thus, there are three mutually beneficial directions for future improvements:

- The quality of datasets should be improved. Since word-level datasets are already available for most languages, constructing sentence-level datasets for more languages should be the major goal. Language-specific traits should be reflected in these datasets. Also, the datasets should ideally cover both formal and informal text. If only the formal text is available, they should at least cover multiple genres.
- The G2P models should be trained and evaluated on sentence-level datasets. A qualitative error analysis should be presented for individual languages, so that improvements could be made based on such observations.
- Due to the diversity of models, the combination between G2P models and downstream models (and definitions of articulatory features, if applicable) should be thoroughly searched. In NLP tasks, the search criterion can be the accuracy. If the comparison is more subtle for linguistic analysis, at least results for multiple combinations should be presented.

Broadly speaking, the problems addressed by these directions are not limited to the usage of phonetic information. The separation of downstream tasks and tokenization methods is also evident for textual tokenization (Mielke et al., 2021). Adopting better practices from such analogous fields is also potentially beneficial, if not vital.

Acknowledgements

This survey paper is a class project of TTIC 31210. The author thanks Professor Kartik Goyal for the helpful suggestions.

References

Muhammad Abulaish, Ashraf Kamal, and Mohammed J Zaki. 2020. A survey of figurative language and its computational detection in online social networks. *ACM Transactions on the Web (TWEB)*, 14(1):1–52.

Tafseer Ahmed, Muhammad Suffian Nizami, and Muhammad Yaseen Khan. 2022. Discovering Lexical Similarity Through Articulatory Feature-based Phonetic Edit Distance. *IEEE Access*, 10:1533–1544.

Iñaki Alegria, Nora Aranberri, Pere R Comas, Víctor Fresno, Pablo Gamallo, Lluis Padró, Iñaki San Vicente, Jordi Turmo, and Arkaitz Zubiaga. 2015. Tweetnorm: a benchmark for lexical normalization of spanish tweets. *Language resources and evaluation*, 49(4):883–905.

Mohammad Arshi Saloot, Norisma Idris, Liyana Shuib, Ram Gopal Raj, and AiTi Aw. 2015. Toward tweets normalization using maximum entropy. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 19–27, Beijing, China. Association for Computational Linguistics.

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, and Winnie Yan. 2021. Results of the second SIGMORPHON shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 115–125, Online. Association for Computational Linguistics.

Salvatore Attardo. 2010. *Linguistic theories of humor*, volume 1. Walter de Gruyter.

Susan Bartlett, Grzegorz Kondrak, and Colin Cherry. 2008. Automatic syllabification with structured SVMs for letter-to-phoneme conversion. In *Proceedings of ACL-08: HLT*, pages 568–576, Columbus, Ohio. Association for Computational Linguistics.

Akash Bharadwaj, David Mortensen, Chris Dyer, and Jaime Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource

transfer settings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1462–1472, Austin, Texas. Association for Computational Linguistics.

Alan W. Black. 2019. CMU wilderness multilingual speech dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*, pages 5971–5975. IEEE.

Alan W Black and Kevin A Lenzo. 2001. Flite: a small fast run-time synthesis engine. In *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*.

Gilles Boulianne. 2022. Phoneme transcription of endangered languages: an evaluation of recent ASR architectures in the single speaker scenario. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2301–2308, Dublin, Ireland. Association for Computational Linguistics.

Michael R Brent and Timothy A Cartwright. 1996. Distributional regularity and phonotactic constraints are useful for segmentation. *Cognition*, 61(1-2):93–125.

Diane Brentari et al. 1998. *A prosodic model of sign language phonology*. Mit Press.

Özlem Çetinoğlu, Sarah Schulz, and Ngoc Thang Vu. 2016. Challenges of computational processing of code-switching. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 1–11, Austin, Texas. Association for Computational Linguistics.

Stanley F Chen et al. 2003. Conditional and joint models for grapheme-to-phoneme conversion. In *INTERSPEECH*. Citeseer.

Xingyi Cheng, Weidi Xu, Kunlong Chen, Shaohua Jiang, Feng Wang, Taifeng Wang, Wei Chu, and Yuan Qi. 2020. SpellGCN: Incorporating phonological and visual similarities into language models for Chinese spelling check. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 871–881, Online. Association for Computational Linguistics.

Monojit Choudhury, Anupam Basu, and Sudeshna Sarkar. 2004. A diachronic approach for schwa deletion in Indo Aryan languages. In *Proceedings of the 7th Meeting of the ACL Special Interest Group in Computational Phonology: Current Themes in Computational Phonology and Morphology*, pages 20–26, Barcelona, Spain. Association for Computational Linguistics.

Monojit Choudhury, Vaibhav Jalan, Sudeshna Sarkar, and Anupam Basu. 2007. Evolution, optimization, and language change: The case of Bengali verb inflections. In *Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology*, pages 65–74, Prague, Czech Republic. Association for Computational Linguistics.

Herbert H Clark. 1970. Word associations and linguistic theory. *New horizons in linguistics*, 1:271–286.

Michael Collins and Nigel Duffy. 2001. Convolution kernels for natural language. In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 625–632. MIT Press.

Robert Daland and Kie Zuraw. 2013. Does Korean defeat phonotactic word segmentation? In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 873–877, Sofia, Bulgaria. Association for Computational Linguistics.

Bart De Boer. 2000. Self-organization in vowel systems. *Journal of phonetics*, 28(4):441–465.

Orphée De Clercq, Sarah Schulz, Bart Desmet, Els Lefever, and Véronique Hoste. 2013. Normalization of Dutch user-generated content. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 179–188, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Ofer Dekel, Joseph Keshet, and Yoram Singer. 2004. An online algorithm for hierarchical phoneme classification. In *International workshop on machine learning for multimodal interaction*, pages 146–158. Springer.

Aliya Deri and Kevin Knight. 2016. Grapheme-to-phoneme models for (almost) any language. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 399–408, Berlin, Germany. Association for Computational Linguistics.

Qing Dou, Shane Bergsma, Sittichai Jiampojamarn, and Grzegorz Kondrak. 2009. A ranking approach to stress prediction for letter-to-phoneme conversion. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 118–126, Suntec, Singapore. Association for Computational Linguistics.

Ewan Dunbar, Gabriel Synnaeve, and Emmanuel Dupoux. 2015. Quantitative methods for comparing featural representations. In *ICPhS*.

Kenneth Dwyer and Grzegorz Kondrak. 2009. Reducing the annotation effort for letter-to-phoneme conversion. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 127–135, Suntec, Singapore. Association for Computational Linguistics.

Peter Eisenberg and Nanna Fuhrhop. 2007. Schulorthographie und graphematik. *Zeitschrift für Sprachwissenschaft*, 26(spec):15–41.

Asif Ekbal, Sudip Kumar Naskar, and Sivaji Bandyopadhyay. 2006. A modified joint source-channel model for transliteration. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 191–198, Sydney, Australia. Association for Computational Linguistics.

Izaskun Etxeberria, Iñaki Alegria, Larraitz Uria, and Mans Hulden. 2016a. Combining phonology and morphology for the normalization of historical texts. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 100–105, Berlin, Germany. Association for Computational Linguistics.

Izaskun Etxeberria, Iñaki Alegria, Larraitz Uria, and Mans Hulden. 2016b. Evaluating the noisy channel model for the normalization of historical texts: Basque, Spanish and Slovene. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1064–1069, Portorož, Slovenia. European Language Resources Association (ELRA).

William M Fisher. 1999. A statistical text-to-phone function using ngrams and rules. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, volume 2, pages 649–652. IEEE.

Margaret M. Fleck. 2008. Lexicalized phonotactic word segmentation. In *Proceedings of ACL-08: HLT*, pages 130–138, Columbus, Ohio. Association for Computational Linguistics.

Ryo Fujii, Masato Mita, Kaori Abe, Kazuaki Hanawa, Makoto Morishita, Jun Suzuki, and Kentaro Inui. 2020. PheMT: A phenomenon-wise dataset for machine translation robustness on user-generated contents. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5929–5943, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Thomas Gärtner. 2003. A survey of kernels for structured data. *ACM SIGKDD explorations newsletter*, 5(1):49–58.

Thomas Gärtner, John W Lloyd, and Peter A Flach. 2004. Kernels and distances for structured data. *Machine Learning*, 57(3):205–232.

Akshat Gupta, Xinjian Li, Sai Krishna Rallabandi, and Alan W. Black. 2021. Acoustics Based Intent Recognition Using Discovered Phonetic Units for Low Resource Languages. *arXiv:2011.03646 [cs]*.

Bruce Hayes. 2011. *Introductory phonology*. John Wiley & Sons.

Ulf Hermjakob, Jonathan May, and Kevin Knight. 2018. Out-of-the-box universal Romanization tool uroman. In *Proceedings of ACL 2018, System Demonstrations*, pages 13–18, Melbourne, Australia. Association for Computational Linguistics.

Yuzhong Hong, Xianguo Yu, Neng He, Nan Liu, and Junhui Liu. 2019. FASPell: A fast, adaptable, simple, powerful Chinese spell checker based on DAE-decoder paradigm. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 160–169, Hong Kong, China. Association for Computational Linguistics.

C Hoppenbrouwers and G Hoppenbrouwers. 1988. De featurefrequentiemethode en de classificatie van nederlandse dialecten. tabu. *Bulletin voor Taalwetenschap, Jaargang*, 18.

Yushi Hu, Shane Settle, and Karen Livescu. 2020. Multilingual jointly trained acoustic and written word embeddings. *Proc. Interspeech 2020*, pages 1052–1056.

Mans Hulden. 2017. A phoneme clustering algorithm based on the obligatory contour principle. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 290–300, Vancouver, Canada. Association for Computational Linguistics.

S Hunnicutt. 1980. Grapheme-to-phoneme rules: A review. *Speech Transmission Laboratory, Royal Institute of Technology, Stockholm, Sweden, QPSR*, (2-3):38–60.

Vincent Jahjah, Richard Khoury, and Luc Lamontagne. 2016. Word normalization using phonetic signatures. In *Canadian Conference on Artificial Intelligence*, pages 180–185. Springer.

Sittichai Jiampojamarn, Colin Cherry, and Grzegorz Kondrak. 2008. Joint processing and discriminative training for letter-to-phoneme conversion. In *Proceedings of ACL-08: HLT*, pages 905–913, Columbus, Ohio. Association for Computational Linguistics.

Long Jiang, Ming Zhou, Lee-Feng Chien, and Cheng Niu. 2007. Named entity translation with web mining and transliteration. In *IJCAI*, volume 7, pages 1629–1634.

Cibu Johny, Alexander Gutkin, and Martin Jansche. 2019. Cross-Lingual Consistency of Phonological Features: An Empirical Study. In *Interspeech 2019*, pages 1741–1745. ISCA.

CG Jung. 1975. Psicologia, linguaggio e associazione verbale. *Roma, Newton Compton*.

Bryan Jurish. 2010. Comparing canonicalizations of historical German text. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 72–77, Uppsala, Sweden. Association for Computational Linguistics.

Katharina Kann, Arya D. McCarthy, Garrett Nicolai, and Mans Hulden. 2020. The SIGMORPHON 2020 shared task on unsupervised morphological paradigm

completion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 51–62, Online. Association for Computational Linguistics.

Jonathan Keane. 2014. *Towards an articulatory model of handshape: What fingerspelling tells us about the phonetics and phonology of handshape in American Sign Language*. Ph.D. thesis, The University of Chicago.

Young-Bum Kim and Benjamin Snyder. 2013a. Optimal data set selection: An application to grapheme-to-phoneme conversion. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1196–1205, Atlanta, Georgia. Association for Computational Linguistics.

Young-Bum Kim and Benjamin Snyder. 2013b. Unsupervised consonant-vowel prediction over hundreds of languages. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1527–1536, Sofia, Bulgaria. Association for Computational Linguistics.

Grzegorz Kondrak and Tarek Sherif. 2006. Evaluation of several phonetic similarity algorithms on the task of cognate identification. In *Proceedings of the Workshop on Linguistic Distances*, pages 43–50, Sydney, Australia. Association for Computational Linguistics.

Ronja Laarmann-Quante, Lukas Knichel, Stefanie Dipper, and Carina Betken. 2016. Annotating spelling errors in German texts produced by primary school children. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 32–42, Berlin, Germany. Association for Computational Linguistics.

Phuong Le Hong, Thi Minh Huyen Nguyen, and Azim Roussanaly. 2009. Finite-state description of Vietnamese reduplication. In *Proceedings of the 7th Workshop on Asian Language Resources (ALR7)*, pages 63–69, Suntec, Singapore. Association for Computational Linguistics.

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 modeling with WikiPron. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4223–4228, Marseille, France. European Language Resources Association.

Haizhou Li, Min Zhang, and Jian Su. 2004. A joint source-channel model for machine transliteration. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 159–166, Barcelona, Spain.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R. Mortensen, Graham Neubig, Alan W. Black, and Florian Metze. 2020a. Universal phone recognition with a multilingual allophone system. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 8249–8253. IEEE.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R. Mortensen, Graham Neubig, Alan W. Black, and Florian Metze. 2020b. Universal phone recognition with a multilingual allophone system. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pages 8249–8253. IEEE.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14, Valencia, Spain. Association for Computational Linguistics.

Hairong Liu, Mingbo Ma, Liang Huang, Hao Xiong, and Zhongjun He. 2019. Robust neural machine translation with joint textual and phonetic embedding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3044–3049, Florence, Italy. Association for Computational Linguistics.

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*, pages 131–140, Online. Association for Computational Linguistics.

Georgia R. Loukatou, Steven Moran, Damian Blasi, Sabine Stoll, and Alejandrina Cristia. 2019. Is word segmentation child's play in all languages? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3931–3937, Florence, Italy. Association for Computational Linguistics.

Andreĭ Andreevich Markov. 2006. An example of statistical investigation of the text eugene onegin concerning the connection of samples in chains. *Science in Context*, 19(4):591–600.

Lynette Melnar and Chen Liu. 2006. A combined phonetic-phonological approach to estimating cross-language phoneme similarity in an ASR environment. In *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology and Morphology at HLT-NAACL 2006*, pages 1–10, New York City, USA. Association for Computational Linguistics.

Gustavo Mendonça and Sandra M Aluísio. 2014. Using a hybrid approach to build a pronunciation dictionary

for brazilian portuguese. In *INTERSPEECH*, pages 1278–1282.

Jeff Mielke. 2005. Modeling distinctive feature emergence. In *Proceedings of the West Coast Conference on Formal Linguistics*, volume 24, pages 281–289. Citeseer.

Sabrina J. Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y. Lee, Benoît Sagot, and Samson Tan. 2021. Between words and characters: A brief history of open-vocabulary modeling and tokenization in NLP. *CoRR*, abs/2112.10508.

Rada Mihalcea and Carlo Strapparava. 2005. Making computers laugh: Investigations in automatic humor recognition. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 531–538, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Daichi Mochihashi, Takeshi Yamada, and Naonori Ueda. 2009. Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 100–108, Suntec, Singapore. Association for Computational Linguistics.

M Asunción Moreno Bilbao, D Poig, Antonio Bonafonte Cávez, Eduardo Lleida, Joaquim Llisterri, José Bernardo Mariño Acebal, and Climent Nadeu Camprubí. 1993. Albayzin speech database: Design of the phonetic corpus. In *EUROSPEECH 1993: 3rd European Conference on Speech Communication and Technology: Berlin, Germany: September 22-25, 1993*, pages 175–178. . EUROSPEECH.

David R. Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. Epitran: Precision G2P for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

David R. Mortensen, Xinjian Li, Patrick Littell, Alexis Michaud, Shruti Rijhwani, Antonios Anastasopoulos, Alan W Black, Florian Metze, and Graham Neubig. 2020. AlloVera: A multilingual allophone database. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5329–5336, Marseille, France. European Language Resources Association.

David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. PanPhon: A resource for mapping IPA segments to articulatory feature vectors. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3475–3484, Osaka, Japan. The COLING 2016 Organizing Committee.

John Nerbonne and Wilbert Heeringa. 1997. Measuring dialect distance phonetically. In *Computational Phonology: Third Meeting of the ACL Special Interest Group in Computational Phonology*.

Josef Robert Novak, Nobuaki Minematsu, and Keikichi Hirose. 2016. Phonetisaurus: Exploring grapheme-to-phoneme conversion with joint n-gram models in the wfst framework. *Natural Language Engineering*, 22(6):907–938.

Kyubyong Park and Seanie Lee. 2020. g2pm: A neural grapheme-to-phoneme conversion package for mandarin chinese based on a new open benchmark dataset. *arXiv preprint arXiv:2004.03136*.

Mateusz Pawlik and Nikolaus Augsten. 2015. Efficient computation of the tree edit distance. *ACM Transactions on Database Systems (TODS)*, 40(1):1–40.

Vladimir Pervouchine, Haizhou Li, and Bo Lin. 2009. Transliteration alignment. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 136–144, Suntec, Singapore. Association for Computational Linguistics.

Ben Peters, Jon Dehdari, and Josef van Genabith. 2017. Massively multilingual neural grapheme-to-phoneme conversion. In *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages 19–26, Copenhagen, Denmark. Association for Computational Linguistics.

Lawrence Philips. 1990. Hanging on the metaphone. *Computer Language*, 7(12):39–43.

Tiago Pimentel, Clara Meister, Elizabeth Salesky, Simone Teufel, Damián Blasi, and Ryan Cotterell. 2021. A surprisal–duration trade-off across and within the world's languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 949–962, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jordi Porta, José-Luis Sancho, and Javier Gómez. 2013. Edit transducers for spelling variation in old spanish. In *Proc. of the workshop on computational historical linguistics at NODALIDA 2013. NEALT Proc. Series*, volume 18, pages 70–79.

Jelena Prokić and Tim Van de Cruys. 2010. Exploring dialect phonetic variation using PARAFAC. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 46–53, Uppsala, Sweden. Association for Computational Linguistics.

Ernst Pulgram. 1951. Phoneme and grapheme: A parallel. *Word*, 7(1):15–20.

Ting Qian, Kristy Hollingshead, Su-youn Yoon, Kyoung-young Kim, and Richard Sproat. 2010. A

python toolkit for universal transliteration. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

B Ramani, S Lilly Christina, G Anushiya Rachel, V Sherlin Solomi, Mahesh Kumar Nandwana, Anusha Prakash, S Aswin Shanmugam, Raghava Krishnan, S Kishore Prahalad, K Samudravijaya, et al. 2013. A common attribute based unified hts framework for speech synthesis in indian languages. In *Eighth ISCA Workshop on Speech Synthesis*.

Kanishka Rao, Fuchun Peng, Hasim 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, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*, pages 4225–4229. IEEE.

Alexander M Robertson and Peter Willett. 1993. A comparison of spelling-correction methods for the identification of word forms in historical text databases. *Literary and linguistic computing*, 8(3):143–152.

Elizabeth Salesky and Alan W Black. 2020. Phone features improve speech translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2388–2397, Online. Association for Computational Linguistics.

Elizabeth Salesky, Eleanor Chodroff, Tiago Pimentel, Matthew Wiesner, Ryan Cotterell, Alan W Black, and Jason Eisner. 2020. A corpus for large-scale phonetic typology. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4526–4546, Online. Association for Computational Linguistics.

Jesper Salomon. 2001. Support vector machines for phoneme classification. *Master of Science, School of Artificial Intelligence, Division of Informatics, University of Edinburgh*.

Peter M Scharf and Malcolm D Hyman. 2011. Linguistic issues in encoding sanskrit. *The Sanskrit Library*.

Andrei Shcherbakov, Ekaterina Vylomova, and Nick Thieberger. 2016. Phonotactic modeling of extremely low resource languages. In *Proceedings of the Australasian Language Technology Association Workshop 2016*, pages 84–93, Melbourne, Australia.

Kilho Shin and Tetsuji Kuboyama. 2013. A comprehensive study of tree kernels. In *JSAI International Symposium on Artificial Intelligence*, pages 337–351. Springer.

Miikka P. Silfverberg, Lingshuang Mao, and Mans Hulden. 2018. Sound analogies with phoneme embeddings. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2018*, pages 136–144.

Eric Smith. 2007. Phonological reconstruction of a dead language using the gradual learning algorithm. In *Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology*, pages 57–64, Prague, Czech Republic. Association for Computational Linguistics.

Winifred Strange. 2007. Cross-language phonetic similarity of vowels. *Language experience in second language speech learning: In honor of James Emil Flege*, 17:35.

Mukuntha Narayanan Sundararaman, Ayush Kumar, and Jithendra Vepa. 2021. Phoneme-bert: Joint language modelling of phoneme sequence and asr transcript. *arXiv preprint arXiv:2102.00804*.

Suyanto Suyanto. 2019. Incorporating syllabification points into a model of grapheme-to-phoneme conversion. *International Journal of Speech Technology*, 22(2):459–470.

Gabriel Synnaeve, Thomas Schatz, and Emmanuel Dupoux. 2014. Phonetics embedding learning with side information. In *2014 IEEE Spoken Language Technology Workshop (SLT)*, pages 106–111. IEEE.

Tao Tao, Su-Youn Yoon, Andrew Fister, Richard Sproat, and ChengXiang Zhai. 2006. Unsupervised named entity transliteration using temporal and phonetic correlation. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 250–257, Sydney, Australia. Association for Computational Linguistics.

Paul Taylor. 2005. Hidden markov models for grapheme to phoneme conversion. In *Ninth European Conference on Speech Communication and Technology*. Citeseer.

Anju Leela Thomas, Anusha Prakash, Arun Baby, and Hema A Murthy. 2018. Code-switching in indic speech synthesisers. In *INTERSPEECH*, pages 1948–1952.

Shubham Toshniwal and Karen Livescu. 2016. Jointly learning to align and convert graphemes to phonemes with neural attention models. *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 76–82.

Kristina Toutanova and Robert Moore. 2002. Pronunciation modeling for improved spelling correction. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 144–151, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Balakrishnan Varadarajan, Sanjeev Khudanpur, and Emmanuel Dupoux. 2008. Unsupervised learning of acoustic sub-word units. In *Proceedings of ACL-08: HLT, Short Papers*, pages 165–168, Columbus, Ohio. Association for Computational Linguistics.

Wilhelm H Vieregge, ANTONIUS CM Rietveld, and Carel IE Jansen. 1984. A distinctive feature based system for the evaluation of segmental transcription

in dutch. In *Proceedings of the Tenth International Congress of Phonetic Sciences*, pages 654–659. De Gruyter Mouton.

Paola Virga and Sanjeev Khudanpur. 2003. Transliteration of proper names in cross-lingual information retrieval. In *Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition*, pages 57–64, Sapporo, Japan. Association for Computational Linguistics.

Asanka Wasala, Ruvan Weerasinghe, and Kumudu Gamage. 2006. Sinhala grapheme-to-phoneme conversion and rules for schwa epenthesis. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 890–897, Sydney, Australia. Association for Computational Linguistics.

Sonjia Waxmonsky and Sravana Reddy. 2012. G2P conversion of proper names using word origin information. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 367–371, Montréal, Canada. Association for Computational Linguistics.

Haryo Akbarianto Wibowo, Made Nindyatama Nityasya, Afra Feyza Akyürek, Suci Fitriany, Alham Fikri Aji, Radityo Eko Prasojo, and Derry Tanti Wijaya. 2021. IndoCollex: A testbed for morphological transformation of Indonesian word colloquialism. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3170–3183, Online. Association for Computational Linguistics.

Peter Wu, Jiatong Shi, Yifan Zhong, Shinji Watanabe, and Alan W. Black. 2021a. Cross-lingual Transfer for Speech Processing using Acoustic Language Similarity. *arXiv:2111.01326 [cs, eess]*.

Shih-Hung Wu, Chao-Lin Liu, and Lung-Hao Lee. 2013. Chinese spelling check evaluation at SIGHAN bakeoff 2013. In *Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing*, pages 35–42, Nagoya, Japan. Asian Federation of Natural Language Processing.

Winston Wu, Kevin Duh, and David Yarowsky. 2021b. Sequence models for computational etymology of borrowings. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4032–4037, Online. Association for Computational Linguistics.

Ke Xu, Yunqing Xia, and Chin-Hui Lee. 2015. Tweet normalization with syllables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 920–928, Beijing, China. Association for Computational Linguistics.

Hemant Yadav, Akshat Gupta, Sai Krishna Rallabandi, Alan W. Black, and Rajiv Ratn Shah. 2021. Intent Classification Using Pre-Trained Embeddings For Low Resource Languages. *arXiv:2110.09264 [cs, eess]*.

Diyi Yang, Alon Lavie, Chris Dyer, and Eduard Hovy. 2015. Humor recognition and humor anchor extraction. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2367–2376, Lisbon, Portugal. Association for Computational Linguistics.

Fan Yang, Jun Zhao, Bo Zou, Kang Liu, and Feifan Liu. 2008. Chinese-English backward transliteration assisted with mining monolingual web pages. In *Proceedings of ACL-08: HLT*, pages 541–549, Columbus, Ohio. Association for Computational Linguistics.

Kaisheng Yao and Geoffrey Zweig. 2015. Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. *arXiv preprint arXiv:1506.00196*.

Sevinj Yolchuyeva, Géza Németh, and Bálint Gyires-Tóth. 2019. Grapheme-to-phoneme conversion with convolutional neural networks. *Applied Sciences*, 9(6):1143.

Sevinj Yolchuyeva, Géza Németh, and Bálint Gyires-Tóth. 2020. Transformer based grapheme-to-phoneme conversion. *arXiv preprint arXiv:2004.06338*.

Ruiqing Zhang, Chao Pang, Chuanqiang Zhang, Shuohuan Wang, Zhongjun He, Yu Sun, Hua Wu, and Haifeng Wang. 2021. Correcting Chinese spelling errors with phonetic pre-training. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2250–2261, Online. Association for Computational Linguistics.

Chengrui Zhu, Keyu An, Huahuan Zheng, and Zhijian Ou. 2021. Multilingual and crosslingual speech recognition using phonological-vector based phone embeddings. *arXiv:2107.05038 [cs, eess]*.