Data-driven Design of a Sentence List for an Articulatory Speech Corpus

Jeffrey Berry[1], *Luciano Fadiga*[1,2]

[1]Department of Robotics, Brain and Cognitive Science, Italian Institute of Technology, Genoa, Italy
[2]Section of Human Physiology, University of Ferrara, Ferrara, Italy

jeffreyjames.berry@iit.it, luciano.fadiga@iit.it

Abstract

Articulatory data offers promising developments in our understanding of speech production and advances in speech technologies. However, it is more expensive and difficult to obtain than audio data, which means data collection must be carefully planned. This paper presents a method for designing an articulatory speech corpus comparable to the widely-used TIMIT corpus, for languages other than English, using Italian as a case study. This data-driven method searches freely-available online text corpora for a set of sentences that provide broad phonetic coverage, while still being small enough to be read in a single session, which is important given the often invasive nature of articulatory data collection. Sentences are first phonemically transcribed and scored based on negative log-likelihood of triphones, with sentences that have many rare triphones scoring higher. The search algorithm then finds sentences that have high scores, but also contain the most frequent triphones. Experiments show that the distribution of triphones in the automatically selected sentences is similar to that found in hand-constructed sentence sets for English, such as TIMIT, and offers broader phonetic coverage than selecting random sets of sentences.

Index Terms: Corpus design, Articulatory Data, Text Corpus, Information Theory

1. Introduction

This paper explores an information-based method for finding an optimal set of sentences to use as prompts for the collection of an articulatory speech corpus for use in training speech recognition systems. Although several well-known sentence lists exist for English, such as TIMIT [6] or the Harvard sentences [7], such widely-used hand-crafted lists often do not exist for other languages. The proposed method searches for the most informative sentences in large online text corpora such as Wikipedia, which is available in many languages. Since the objective of the resulting corpus is to train speech recognition systems, the information of a sentence is calculated in terms of triphones. Under this definition, more informative sentences are those which contain more previously unseen triphones.

Designing and collecting articulatory speech corpora for improving speech recognition may at first seem counter-intuitive, since at runtime, such data streams are not likely to be available. However there is a growing body of research that suggests that we rely on our knowledge of speech production during perception, and therefore articulatory data may play an important role in improving automatic speech recognition. The well-known McGurk effect [9] demonstrates that speech perception can involve data from multiple sources, i.e. visual information in addition to acoustic. Research on adding speech production knowledge to speech recognition systems has demonstrated that taking advantage of these different sources of information can lead to better performance (cf. [8] for discussion). Numerous behavioral and neurological studies have shown that the motor system is involved to some degree in speech *perception*, although the details are still under debate (cf. [5]). This suggests that other data sources that provide more complete information about the vocal tract configuration, such as electromagnetic articulography (EMA), ultrasound, or rtMRI [10], may be used to train better performing speech recognition systems.

Large-scale exploration of the possibilities is hampered by the lack of large articulatory speech corpora. This lack of corpora is likely due to the expense and difficulty of collecting articulatory data. For example, in the case of EMA, one of the most widely-used sources of articulatory data, the equipment is expensive and invasive. Coils often come unglued from the tongue and have to be re-applied, and speakers become fatigued fairly quickly. Similar practical constraints exist for most other methods of articulatory data acquisition, which has prevented the construction of corpora on the scale available for audio data. When dealing with articulatory data, stimulus prompts must therefore be carefully designed in order to make the best use of time and materials. The widespread availability of online text in many languages provides the data necessary to automatically construct phonetically-balanced sentences lists, which previously had to be constructed by hand.

2. Methodology

The aim of this paper is to find a set of sentences that gives broad phonetic coverage of the target language, with the constraint that the entire set of sentences should be able to be read in a few hours. This time constraint arises from the fact that when collecting articulatory data, subjects are often fatigued after 1–2 hours, especially in the case of EMA. Choosing a corpus size of about 100 sentences fits with this practical constraint, allowing time for attaching (and often reattaching) EMA coils to the articulators before the subject is fatigued. The methods presented here are flexible and can be applied for larger sentence sets as well, for example when collecting audio or less invasive articulatory data, or collecting data across multiple sessions, etc.

To find a set of sentences, a large amount of text data is required. Online sources such as Wikipedia present viable text corpora for this purpose in many languages. When using web text data, plain text must first be extracted from mark-up before beginning the search. For this paper, the PAISÀ Italian text corpus was used (`http://www.corpusitaliano.it`), which consists of plain text extracted from freely available sources from the Wikimedia Foundation, and licensed under Creative Commons [2]. The PAISÀ Italian corpus consists of 1.4 GB of UTF-8 encoded plain text, with over 225 million

words.

Since the goal was to achieve broad phonetic coverage, the plain text was first phonemically transcribed. This was done automatically for PAISÀ using the Italian letter-to-sound engine for the Festival Speech Synthesis System available at `http://www2.pd.istc.cnr.it/FESTIVAL` [3]. The transcription used a set of 39 phonemes, including '#' to mark sentence boundaries. Stressed vowels were represented separately from non-stressed. In order to capture variation due to co-articulation, the transcriptions were converted to sequences of triphones, which were used as the basis for the search algorithm.

2.1. Scoring the sentences

In order to define an objective for the search algorithm to maximize, each transcribed sentence in the corpus was scored using the following heuristic, based on negative log-likelihood of triphones:

$$\text{Score} = \left(\frac{1}{T} \sum_t -\log(\frac{c_t}{N}) \right) \frac{u}{T}, \quad (1)$$

where T is the number of triphones in the sentence, c_t is the number of occurrences of triphone t in the entire corpus, N is the total number of triphones in the corpus, and u is the number of unique triphones in the sentence. The score consists of two basic parts. The first part $\frac{1}{T} \sum_t -\log(\frac{c_t}{N})$, which is the mean triphone negative log-likelihood, rewards sentences containing rare triphones, since the negative log-likelihood is larger for rare triphones. The second part $\frac{u}{T}$ rewards sentences with more distinct triphones, normalizing for sentence length. The highest scoring sentences therefore contain many different rare triphones, which are useful for generating a small set of sentences with broad phonetic coverage of the language.

Figure 1 shows the distribution of scores of the PAISÀ corpus using equation 1. From the figure, it appears that a small set

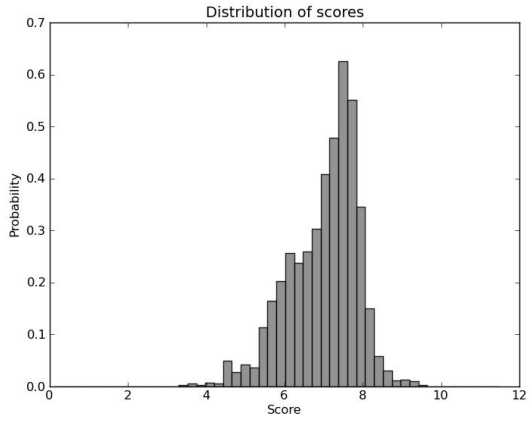

Figure 1: *Distribution of scores for the PAISÀ corpus resulting from equation 1.*

of sentences with scores above 9 contain a high-density of rare triphones. An examination the sentences determined that most of the highest scoring sentences were not Italian, but English. This illustrates that the heuristic in equation 1 is successful, since English sentences in an Italian corpus naturally contain many rare triphones, i.e. triphones that do not occur in Italian. This fact also brings to light one of the difficulties in working with text scraped from the web, which is that such data can be noisy and unreliable.

2.2. The search algorithm

The objective of the search is to find a set of k sentences that provide broad phonetic coverage of Italian. Simply taking the highest-scoring k sentences may not result in broad coverage, but would maximize the number of occurrences of rare triphones. Instead, the distribution of triphones in the selected sentences should reflect that of the target language, while ensuring that as many triphones as possible are represented. With this in mind, the search algorithm is designed to ensure that the most frequent triphones are covered, but uses high-scoring sentences to ensure representation of some of the rare triphones as well.

Listing 1 outlines the search algorithm. Each of the k sentences is selected by first finding the most frequent triphone that has not occurred in the previously selected sentences. Once that triphone is found, the highest scoring sentence containing that triphone is selected, and the process is repeated until all k sentences have been selected. If k is very large, it is possible that there may be no triphones left with 0 occurrences in the previously selected sentences. In that case, line 4 of the algorithm can be modified to search for the first triphone with 1 occurrence, and failing that, the first with 2 occurrences, and so on until a suitable triphone is found.

Algorithm 1 The search algorithm

input
Int k {number of sentences to find}
Dictionary T {key: triphone, value: list of sentence IDs}
Dictionary S {key: sentence ID, value: list of triphones}
Dictionary $Scores$ {key: sentence ID, value: score}
List $Rank$ {Triphones sorted by frequency, descending}
end input

1: Initialize Dictionary $Count$ {key: triphone, value: count}
2: Initialize List $Selected$ {list of selected sentences}
3: **while** length of $Selected < k$ **do**
4: Find first triphone t in $Rank$ with value 0 in $Count$
5: Get sentence IDs $slist$ from $T(t)$
6: Sort $slist$ in descending order using $Scores$
7: Initialize Bool $found \leftarrow$ **false**
8: Initialize Int $index \leftarrow 0$
9: **while not** $found$ **do**
10: **if** $slist(index)$ **not** in $Selected$ **then**
11: add $slist(index)$ to $Selected$
12: **for** triphone t in $S(slist(index))$ **do**
13: increment $Count(t)$
14: **end for**
15: $found \leftarrow$ **true**
16: **else**
17: increment $index$
18: **end if**
19: **end while**
20: **end while**
21: **return** $Selected$

As mentioned above, many of the highest scoring sentences in the PAISÀ corpus are in English rather than Italian. To deal with that problem, we added a second conditional after line 10, which checks whether the sentence is Italian. This was accomplished by querying the Google Translate API,

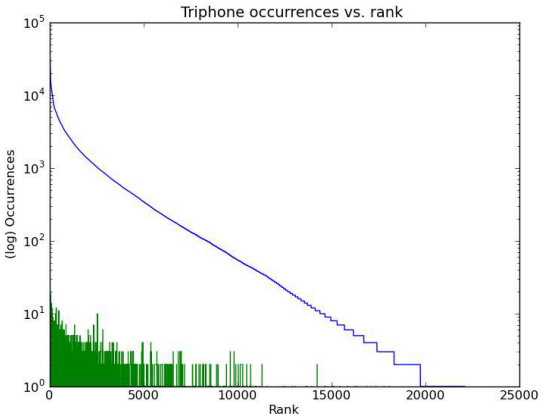

Figure 2: *The distribution of triphones for the 100 sentences chosen from the PAISÀ corpus. The blue line represents triphone counts sorted by frequency (rank) for the entire corpus. The green lines show counts for the corresponding triphones in the selected sentences.*

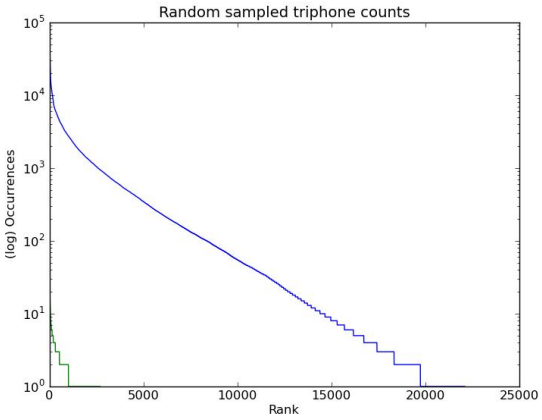

Figure 3: *The distribution of triphones for 100 randomly chosen Italian sentences. The blue line, as in Figure 2 represents the ranked triphone counts. The green line shows the triphone counts for the chosen sentences, which do not provide as broad phonetic coverage.*

which provides a language detection function (see `https://developers.google.com/translate/`). An implementation of the search algorithm, together with supporting code is available at `https://github.com/jjberry/database-construction`.

The time complexity of the search algorithm is much better than the intractable factorial-time brute-force solution $O(\binom{n}{k})$, although the search algorithm does not guarantee the optimal set of k sentences. The input dictionaries T, S, and $Scores$ can each be computed in $O(n)$, and the sorted list $Rank$ in $O(m \log m)$, where $m \ll n$ is the number of triphone types. The slowest part of the search algorithm is the sort in line 6, which in the worst case is $O(n \log n)$, and is repeated k times. In practice the length of the list of candidate sentences $slist$ is much less than n which makes the sort much faster. In the authors' Python implementation, the largest bottleneck occurs when querying the Google Translate API, due to bandwidth issues, which requires the script to sleep on each iteration of line 10.

3. Experiments

We conducted several experiments to evaluate the effectiveness of the proposed scoring heuristic and search algorithm. The first experiment was performed using the PAISÀ Italian corpus, with $k = 100$, i.e. selecting a set of 100 sentences. Each of the sentences was verified as Italian using the Google Translate API. Figure 2 shows the results of the experiment. The figure is arranged as a rank vs. occurrences plot for triphones. The continuous blue line shows the expected Zipf's law distribution for triphones in the PAISÀ corpus. The green lines at the bottom of the figure show the number of occurrences for triphones in the selected 100 sentences. The results show that the most frequent triphones are well covered, and show good coverage of many less-frequent triphones as well.

For comparison, we conducted a second experiment choosing 100 sentences at random. Just as with the first experiment, each of the 100 random sentences was validated as Italian using the Google Translate API. Figure 3 shows the resulting distribution of triphones. Although choosing random sentences adequately covers the most frequent triphones, it is clear from the results that broad coverage has not been obtained.

In order to provide a point of comparison to hand-constructed sentence lists, the results of the experiments on the PAISÀ Italian corpus were compared to the TIMIT and Harvard sentences for English, which are phonetically balanced. To estimate the triphone counts for English, we used the Brown corpus [4] downloaded from the Natural Language Toolkit (NLTK) website `http://nltk.googlecode.com/svn/trunk/nltk_data/index.xml`[1]. The Brown corpus was transcribed with the English version of Festival as described in section 2. The TIMIT corpus sample, available via NLTK, and the Harvard sentence list were similarly transcribed and used as selected sentence sets. The TIMIT selection contained 160 sentences, and the Harvard set contained 720 sentences.

Figure 4 shows the distribution of triphones in the Brown/TIMIT sentences, with the blue line representing the triphone counts in the Brown corpus sorted by frequency, and the green lines representing the triphone counts in the TIMIT selection. Similarly, Figure 5 shows the distrubtion for the Brown/Harvard sentences. The results show a distribution of triphones for the Harvard sentences that looks similar to the one for PAISÀ shown in Figure 2, with broad phonetic coverage. It is interesting to note that the TIMIT sentences contain large counts of several rare triphones, denoted by the spikes in the tail of Figure 4. The Harvard sentences on the other hand, appear to be better balanced in the sense that they provide broad coverage, while not over-representing the more rare triphones.

Examining the distributions of scores from equation 1 shows some interesting differences between hand-constructed sentence sets and the set selected using the proposed method, as seen in Figure 6. The Harvard and TIMIT sentences show significantly higher scores than randomly selected English sentences (shown by the 'Rand EN' label in the figure), and are much higher than the scores for the Italian sentences. Interestingly, the randomly selected English sentences have similar scores to the Italian sentences, which suggests that naturally-occurring sentences have similar phonetic densities across lan-

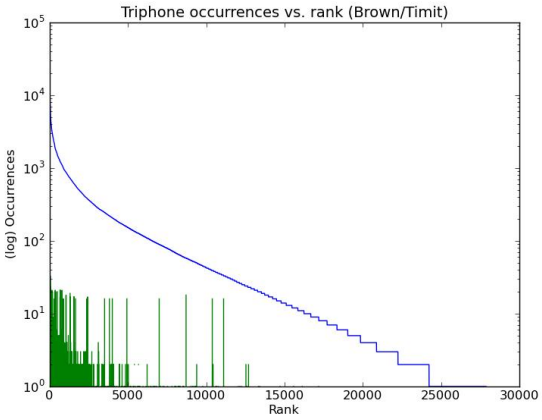

Figure 4: *The triphone distribution of the Brown corpus in blue, and the TIMIT selection in green.*

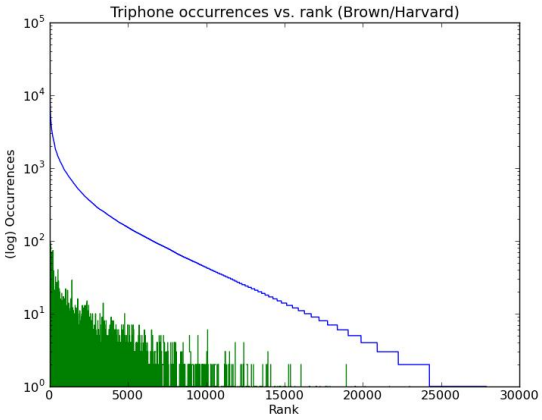

Figure 5: *The triphone distribution of the Harvard sentences.*

guages. Since the sentences selected by the proposed method are taken from natural sentences, it is not surprising to see scores that are similar to randomly selected sentences. A t-test showed that the mean score of the selected set was significantly higher than the mean of the random sentences ($t = 8.10$, $p < 0.001$), which is expected since the search algorithm chooses higher-scoring sentences.

4. Discussion

Results in section 3 suggest that the proposed method is successful in finding a set of phonetically balanced sentences. Although hand-constructed sentences are more phonetically dense, the proposed method is still able to achieve comparable phonetic coverage using natural sentences. As mentioned in the methodology section, the highest scoring sentences in the PAISÀ corpus are actually not Italian sentences, but rather English, which required the use of the Google Translate API to verify that selected sentences were Italian. Although the results of the API language detection query were accurate, many of the Italian high scoring sentences contained non-Italian words, most often foreign names. Additionally, high scoring sentences were likely to contain spelling and grammatical errors. Since

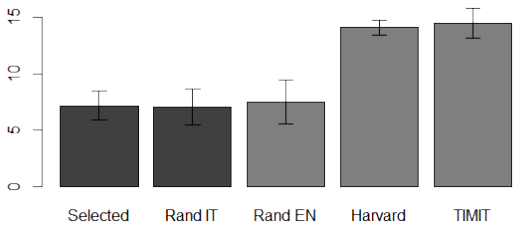

Figure 6: *Means and standard deviations of scores for the various sentence sets. Hand constructed sentence sets are much more phonetically dense than natural sentences.*

the automatic transcription using the Italian version of Festival makes use almost exclusively of letter-to-sound rules, rather than dictionary look-ups as is the case with the English version, foreign names were often transcribed in ways that native Italian speakers would not pronounce them. Similarly, typographical errors were not recognized as such by the transcriber. These transcription errors lead to high scores for these sentences, since they contained very rare triphones, i.e. triphones that don't actually exist in Italian. This presents obvious problems, since subjects will likely auto-correct the typographical and grammatical errors and pronounce foreign words correctly when reading the sentences, rather than reading the erroneous triphones that the sentence was selected for. To resolve this problem, sentences must be proofread by native speakers, which means many of the highest-scoring sentences have to be discarded. Alternatively, proofread text corpora can be used in place of collections like PAISÀ, although the availability of such materials may be an obstacle.

These problems relating to noisy data present a weakness to the proposed method for finding a set of sentences, although the weakness lies primarily in the data itself together with the naive transcription method rather than the scoring heuristic and the search algorithm. The simple letter-to-sound rule transcription system can likely be improved and trained to detect foreign words and typographical errors, which would improve the robustness to noise and usability of the system. For languages whose orthographies require transcription dictionaries rather than letter-to-sound rules, out-of-dictionary errors can act as a filter for these types of problems. On the other hand, the results point to another possible use of the scoring heuristic to find problematic sentences, since unusually high-scoring sentences are likely to have foreign words and errors in them.

The method presented here for finding a set of sentences for broad phonetic coverage of the language presents an automated solution for an issue that must be resolved before collecting data for a speech corpus. Covering as many phonetic contexts as possible in a relatively small set of sentences is especially relevant when creating speech corpora with articulatory data, where practical constraints require careful planning. Such corpora, such as the MOCHA-TIMIT data set [11], have shown promising results for the integration of articulatory data in speech recognition systems for English [8]. The proposed method will help facilitate the design of similar data sets for other languages.

5. References

[1] Bird, S., Klein, E., Loper, E., "Natural language processing with Python: Analyzing text with the Natural Language Toolkit", O'Reilly Media, 2009.

[2] Brunello, M., " PAISÀ - A Creative Commons corpus", presented at NLP group meeting, University of Leeds, Leeds, January 20th 2011.

[3] Cosi, P., Tesser, F., Grette, R., and Avesani, C., "Festival speaks Italian!", Proc. Eurospeech 2001, Aalborg, Denmark, September 3–7, 2001, pp. 509–512.

[4] Francis, W., Kucera, H., "Brown corpus manual", Brown University, 1964, rev. 1979, Online: `http://icame.uib.no/brown/bcm.html`, accessed on 18 Mar 2013.

[5] Galantucci, B., Fowler, C., Turvey, M., "The motor theory of speech perception reviewed", Psychonomic Bulletin and Review 13(3): 361–377, 2006.

[6] Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallet, D., Dahlgren, N., "The DARPA-TIMIT acoustic-phonetic continuous speech corpus CDROM", NIST, 1986.

[7] IEEE Subcommittee, "IEEE recommended practice for speech quality measurements", IEEE Trans. on Audio and Electroacoustics 17(3): 225–246, 1969.

[8] King, S., Frankel, J., Livescu, K., McDermott, E., Richmond, K., Wester, M., "Speech production knowledge in automatic speech recognition", J. Acoust. Soc. Am. 121(2): 723–742, 2007.

[9] McGurk, H., MacDonald, J., "Hearing lips and seeing voices", Nature 264(5588): 746–748, 1976.

[10] Narayanan, S., Nayak, K., Lee, S., Sethy, A., Byrd, D., "An approach to real-time magnetic resonance imaging for speech production", J. Acoust. Soc. Am. 115(5): 1771–1776, 2004.

[11] Wrench, A., "A multi-channel/multi-speaker articulatory database for continuous speech recognition research", Phonus, Institute of Phonetics, 2000.