Evaluation and collection of proper name pronunciations online

Ariadna Font Llitjós, Alan W Black

Language Technologies Institute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
aria@cs.cmu.edu

Abstract

Objective evaluation allows a model to be compared with other similar models. However, automatic pronunciation models should also be extensively evaluated by humans, since the ultimate goal of any pronunciation model is to produce an accurate pronunciation as judged by most people. This paper describes an initiative to evaluate and collect proper name pronunciations online, the development of the US Pronunciation of Proper Names Site (www.pronounce-names.org), and the results obtained so far. The internet, through our web-based interface, has already proven to be a very successful medium both in terms of number of evaluations and in terms of data collection. In 5 weeks, it has brought to our site 601 users, which have evaluated 477 names and corrected 281 pronunciations. The information gathered is useful to improve our pronunciation models, as well as to (automatically) correct the pronunciations in the CMU dictionary.

1. Introduction

One of the current goals in speech synthesis is to acquire high quality pronunciations for proper names. There are several factors that make proper names especially hard to pronounce. Names can be of very diverse etymological origin and can surface in another language without undergoing the slow process of assimilation to the phonologic system of the new language. Furthermore, the number of distinct names tends to be very large (Coker et al., 1990; Font Llitjós, 2001a).

Taking one step towards that goal, we built statistical pronunciation models specific for proper names in US American English that take into account language origin as well as language family information.

These models were built to test the hypothesis that an automatic pronunciation model can benefit from language origin information similarly to the way humans do when pronouncing proper names ('cc' in *Bocaccio* vs. *McCallum*).

Both automatic, objective evaluations as well as human, subjective evaluations are most valuable to test the quality of pronunciation models and speech synthesis. Whereas the first type of evaluation is easy to achieve, evaluations of the second kind have been much harder to conduct. And, even though there have been some efforts in the past to obtain feedback on how humans pronounce their names (an example of such efforts was carried out by Murray Spiegel [Black in personal communication]), there has been no large scale human evaluation of synthesized speech data so far.

Currently, we have a medium at hand to gather large amounts of human evaluation data and process such data in an automatic way: the World Wide Web. In 2002, 42% of the US households (a total of 44 million) have regular internet access (U.S. Department of Commerce, 2002). We can greatly benefit from this by building a web application which requires almost no expertise in speech synthesis or pronunciation, and has users rate the quality of audio files pronouncing their name.

In recent experiments, we have conducted a web-based evaluation to find out how humans rate our automatic pronunciation models.

This paper describes a web-based application (Font Llitjós, 2001b) that allows users to type in their names, and generates the phonetic transcription as well as an audio file according to (i) a pronunciation dictionary, (ii) a baseline, pronunciation model, which only takes into account the letters and their letter context, and (iii) a model incorporating language origin information.

Having listened to the audio files, the user is then asked to determine whether the pronunciations generated are correct, acceptable or unacceptable.

In a 5-week period, there have already been 477 evaluations, and 281 corrected pronunciations have been collected, which can be used to improve our models and correct the CMU dictionary of pronunciation.

In this early stage of the project, we are effectively getting humans to evaluate our pronunciation models. The question arises of how much noise there will be in the data collected through the web-based interface and how we can automatically detect it as such.

At later stages, our site will be able to present users with statistics on how people pronounce their names, and thus it will become a useful resource on proper names pronunciation.

Even though we believe in the *bona fide* of most users, such a site will unavoidably attract some misuse. An example of this is somebody entering /N AO1 R M AH0 JH IY1 N/ as the pronunciation of the name Marilyn Monroe, say. Therefore, this project also involves issues such as cleaning large amounts of data automatically. We have built a spurious pronunciation filter to detect such mismatches automatically and to be able to filter them out when adding user proposed pronunciations to our lexicon.

This paper is divided in two parts. The first part summarizes the automatic pronunciation models previously developed and reports objective evaluation results (sections 2 and 3). The second and main part of this paper describes the development of a web-based interface (section 4), reports subjective evaluation results on the pronunciation data generated by such models (section 5), described the data collection of corrected pronunciations from the users (section 6), and discusses interlabeler agreement measures and results (section 7).

2. Automatic Pronunciation Models for proper names

2.1. Goal

What we tried to model is the educated pronunciation of proper names in US American English. In the case of foreign proper names, we are interested in their Americanized pronunciation, not the original pronunciation of foreign words (which might be as puzzling to the American ear as a wrong pronunciation). For example, if we consider the proper name 'Van Gogh', what we want our system to output is not /F AE1 N G O K/ or /F AE1 N G O G/, which some people may claim is the correct way of pronouncing it, but rather the American pronunciation of it: /V AE1 N . G OW1/.

For this reason we restricted ourselves to the set of American English phonemes as defined in CMU dictionary (CMU Speech Group, 1998), but we allowed more letter to phone alignments than the one used for the whole CMU dictionary, which resulted in almost the double of phone combinations (Font Llitjós, 2001a).

2.2. Data and baseline model

The data used across all the models is a list of proper names from Bell Labs' directory listings (at least 20 years old), containing the 50,000 most frequent surnames and 6,000 names in the US, and their pronunciation as it appears in the CMU dictionary with stress.

We held out every tenth word in the 56,000-name list for testing and used the remaining 90% as training data.

Based on the techniques described by Black and colleagues (1998), we used decision trees (CART) to predict phones based on letters and their context.

2.3. Incorporating language origin information

To improve over the baseline model, we developed 4 automatic pronunciation models by adding language origin information to the CART. These models are sets of letter-to-sound (LTS) rules, which are used to pronounce out of vocabulary words, in our case, names that are not in the CMU dictionary.

The **25 language feature model** incorporates the language features extracted from a 25-fold language classifier (Catalan, English, French, German, etc.) and passes them on to the CART, together with the letter n-gram features, to build the pronunciation model.

The **5 Family language model** groups the 25 languages into 5 family languages and builds a 5-fold classifier instead to extract the language features to be added to the baseline CART.

The **2-pass algorithm** approach attempts to benefit from both the generalization from family languages as well as the 25 language specific letter language models (LLMs). The 2-pass algorithm first classifies the training data using the 5-family language models (which has 73% chance of assigning the right label), and then loads the language specific LLMs for the languages corresponding to that family to get the features that are going to be passed to the CART.

The **unsupervised language model** is a new approach we have begun to investigate, which consists of unsupervised clustering of proper names to derive *language* classes in a data-driven way. With this approach, no language classes need to be determined a priori, but rather they are inferred from the names and their pronunciation. The clustering method used takes into account letter trigrams as well as their aligned pronunciation at training time.

For more details about how these models were built and the motivation behind them, please see Font Llitjós (2001a). In the next section, we summarize their pronunciation word accuracy.

3. Objective Evaluation Results

For this task, objective evaluation is defined as pronunciation word accuracy, and it consists of comparing the pronunciations generated by our models with the pronunciations in the held out test data. The way this works is by taking all 5,600 test names (without their pronunciation) and running them through the 3 sets of LTS rules to obtain a pronunciation for each of them.

The evaluation is the result of strict comparison of such LTS generated pronunciations with the "correct" pronunciation, which is the one in the test data[1].

The results from the different pronunciation models described in section 2 above are summarized in Table 1.

Models	Letters NS[2]	Words NS	Words Stress
baseline	89.02%	58.97%	54.08%
25 languages	89.11%	60.23%	55.10%
5 families	89.20%	60.60%	55.02%
2-pass alg.	89.24%	60.76%	55.22%

Table 1: Pronunciation accuracy for the language (family) based models

Objective evaluation allows for fair comparison with other models. However, what we ultimately want to know is what pronunciation model people think is better on average.

Another reason for having a large sample of users evaluate the pronunciations produced by our models, is that our training data has a significant amount of noise.

Therefore, we decided to develop a web-base application, which allows us to have a subjective evaluation of our pronunciation models: the US Pronunciation of Proper Names Site.

4. US Pronunciation of Proper Names Site

www.pronounce-names.org

4.1. Goals

The main purposes of developing a web-based interface is to evaluate the pronunciation models described in section 3 and collect data to:

- improve our pronunciation models
- improve the CMU dictionary

The first step to improve our pronunciation models is to find out when they make mistakes. By doing error

[1] Even if the pronunciations in the test data are actually not correct, for the purposes of objective evaluation, we blindly take what is on the test data to be correct.

[2] NS stands for *no stress* ; i.e. it does not take stress into account when determining accuracy.

analysis, we will have a better understanding of the limitations of our models and if how to go about trying to improve them.

Noise user studies showed that the list of 56,000 names, which are part of the CMU dictionary, contained 15.36% unacceptable pronunciations (Font Llitjós, 2001a). Through our web-based evaluation, we can automatically find all the names for which the pronunciation given by the CMU dictionary was rated as unacceptable by an empirically determined number of users. Then, we can either correct the pronunciation by hand or use a filter to detect which pronunciations proposed by the user are trustworthy and thus, could be used as the correct pronunciation (see section 6 below).

We would like any educated native speaker of American English to be able to do this evaluation, thus the US Pronunciation of Proper Names Site (PPN-site) was designed for users with no expertise in speech synthesis or phonetics.

4.2. Design and development of the PPN site

It is always a challenge to design a web-based interface that will allow accomplishing the task at hand and, at the same time, will be easy to use for a wide variety of users. Next, we list the different user profiles that need to be taken into account and discuss some of the design and development choices made.

4.2.1. Users profiles

The profile of the users of the PPN-site is mostly underspecified. They need have no expertise in speech synthesis or in phonetics to be able to evaluate our models. The interface tries to incorporate as much help and guidance as possible so that somebody that does not have any information about the site or about speech synthesis is able to accomplish the task successfully.

The assumption is that everyone is an "expert" on how to pronounce one's own name. However, there is one caveat. Two native American English speakers can pronounce a name very differently. For example, two people called *Irina* can pronounce it /IH R IY N AH/ and /AY R IY N AH/ respectively. So there will inevitably be some discrepancies even when we are just looking at native speakers data. In section 7 we will discuss interlabeler agreement for the task at hand.

There are mainly three different kinds of users, which we have to be prepared to deal with:

a) Users who want to help evaluate our pronunciation models.
b) Users who want to test the site (and maybe our models).
c) Users who want to know the pronunciation of an unfamiliar name.

Even though at this stage we would like all our users to be of type (a), we will inevitably get a significant amount of users of type (b) and (c) as well.

Users of type (c) will most likely not introduce any noise to our data. They will typically be non-native speakers of English, who will not really evaluate the different pronunciations, but rather just listen at the audio files and then exit the site.

Users of type (b) are the ones who pose the real problem to our automation process, since they will introduce noise, which is hard to detect. The behavior of such users ranges from trying to break the site to checking whether the same name, given different family language origins, has different pronunciations. In both cases, such users do not really care to give their faithful opinion of the quality of a pronunciation or its language origin.

4.2.2. Key decisions

In this section, we discuss some of the key decisions that needed to be made when designing the PPN-site. The first two, model selection and evaluation scores, clearly have a direct impact on the evaluation results.

Model selection

Generating the pronunciations using all 6 models (CMU dictionary, baseline, and the 4 models briefly described in section 2) is clearly not appropriate. It would put an unnecessary burden on the users, given that the first three models presented in section 2 have a very large amount of overlap, i.e. they predicted the same pronunciation for a large number of names in the test set.

On the other hand, informal studies showed that humans are not very good at identifying the language origin of a name, and when asked to classify 516 names from the test set, they could only tag 43% confidently (Font Llitjós, 2001a).

Asking users to guess the family language origin, namely to classify the name as either *Asian*, *Germanic*, *Romance*, *Slavic* or *Others* seems a much more reasonable task, than asking the user to classify the name as being *Catalan, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, French, German, Hebrew, Indian, Italian, Japanese, Korean, Malaysian, Norwegian, Polish, Portuguese, Serbian, Slovenian, Spanish, Swedish, Thai* or *Turkish.*

For all this reasons, we decided that in addition to the pronunciation from the CMU dictionary, and the one generated by the baseline, we would ask users to evaluate only one of the models mentioned in section 2, the 5 family languages model.

Evaluation Scores

There are many different ways we could have asked users to score the quality of pronunciations: GOOD or BAD; 1 2 3 4 5, where 1 is very good and 5 is very bad (or vice versa), etc.

We decided to ask users to assess whether a pronunciation was: **correct**, **acceptable**, and **unacceptable**; where 'correct' means like an educated native US American English speaker would pronounce it; 'acceptable' means that somebody could say it like that and it is understandable, and 'unacceptable' means that no one would pronounce it that way and that it is hard to understand what name is being meant.

The advantages of this scoring scheme are that (i) the score names are transparent as to what they mean and that (ii) it is always possible to collapse 'correct' and 'acceptable' into GOOD and have 'unacceptable' to be BAD, if a binary scheme is estimated to be preferable at a later point.

Phoneme set

In principle, there are many possible phoneme sets we could have used to encode the phonemic transcription of names. For simplicity, we decided to use the

DARPABET, which is the phoneme set used in the CMU dictionary (CMU Speech Group, 1998), as well as the one used for our pronunciation models. Another reason for making that choice is that we believe that making finer distinctions would most likely confuse users who do not have much knowledge of phonetics.

This phoneme set has 39 phonemes, not counting variations for lexical stress. Lexical stress is indicated by appending a 1 after the stressed vowel. For example, if we want to indicate that the fourth syllable in "evaluation" is stressed, we would write /IH V AE L Y UW **EY1** SH AH N/.

Since most users are probably not familiar with the phoneme set, at each point where the user might need to look up what each phoneme represents in our web-based application, we provide the user with the list of phonemes together with two or more written and audio examples.

Text-to-Speech System

For all our previous experiments, including the building of all our pronunciation models, we used Edinburgh University's Festival Speech Synthesis System (Black *et al*., 1998).

This TTS system is free and has been widely used for research as well as commercial systems.

We used a diphone voice instead of a unit selection voice, even though it does not sound as good, since it is easier for users to modify phones in a monotonic way.

4.2.3. Implementation details

The implementation of the PPN-site involved an iterative process with the following steps: (i) architectural design, (ii) navigation design (see data flow diagram in Figure 1), (iii) content design, (iv) interface design, (v) page generation and (vi) testing (mostly white-box testing).

The PPN-site has the same architecture as most web-based applications, but it also has a festival server, which is effectively equivalent to database lookups, with the difference that it actually generates the information dynamically. This poses some efficiency constraints; loading the appropriate functions at run time would be too slow, thus the functions that need to be called for this application are loaded *a priori* to the festival server.

On the client side, festival clients get initiated dynamically each time a user sends a query to the system, and are killed after calling the appropriate functions.

The web-based application consists of an initial HTML page and 7 CGI scripts which store the information given by the user and dynamically generate other HTML pages that allow the user to evaluate the pronunciation of the name they entered. For a simplified overview of the system, see the data flow diagram shown in Figure 1. This diagram shows the main steps involved, but omits error-checking steps as well as other minor implementation details.

The initial web page contains a query field, the box where the user types in their name in romanized from (more than one name and hyphenated names are also supported), asks the user to guess which is the family language origin for the name and gives the user some instructions and information about the PPN-site.

After filling in some classification questions, the user is presented with all the different pronunciations our 3 models generated for the name queried. It displays the phonemic transcription and the corresponding audio file, which the user can listen to as many times as necessary before rating each pronunciation.

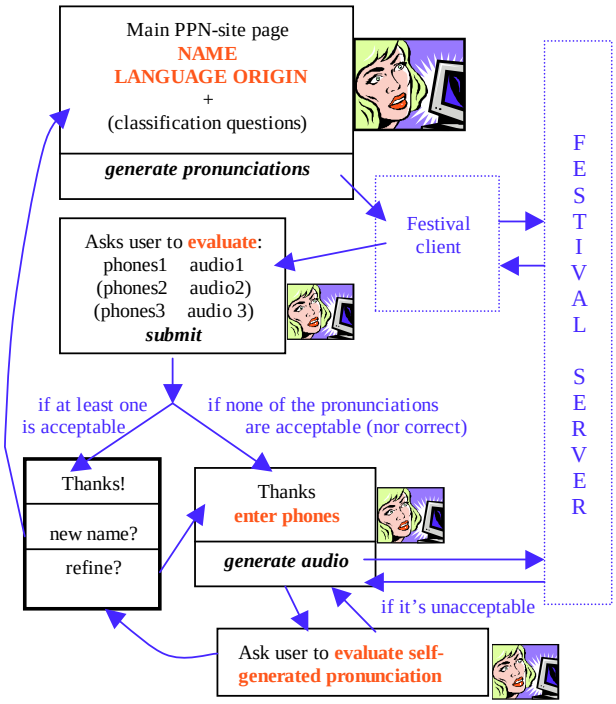

Figure 1: Simplified Data Flow Diagram

Once the user has scored the pronunciations, if at least one was acceptable, the evaluation is done and the user can either continue entering names or, if so wishes, s/he can refine the pronunciation that s/he just rated. On the other hand, if none of the pronunciations were acceptable, the user is asked to modify the phonemic transcription until it is correct.

In this case, the user can generate an audio file at any point from the phonemic transcription that s/he is trying to correct. Every time the user generated an audio file, s/he is asked to score it.

5. Subjective Evaluation Results

The main goal of the PPN-site is to allow us to find out what people think of our pronunciation models, both in comparative and absolute terms.

The web-based application described above is a good medium to massively evaluate our pronunciation models.

All the results presented in this section where gathered during a period of 5 weeks (February 19 to March 27) and the main bulk of evaluations occurred after publicizing the PPN-site in several relevant discussion lists[3].

To retrieve all the relevant information from user output directories and files, we wrote a post processing Perl script (about 1061 lines long), which extracted the information and presented it in a usable way.

[3] The PPN-site was widely announced 2 weeks before the end of the data collection, March 12.

5.1. General Statistics on the PPN-site

The general statistics about the queries sent to the PPN-site during that period are summarized in Table 2.

General Stats	Native speakers	non-native speakers
# queries	683	627
# diff IP address	357	244
# diff names	576	482
# evaluations	292	185
# user correction	117	164

Table 2: General Statistics comparing native and non-native speakers

The number of IP addresses is probably the best approximation to the number of different users that sent queries to the system, the total adds up to 601 users.

During 5 weeks, users sent 1310 queries. On average, each native speaker sent 1.9 queries and each non-native speaker sent 2.57 queries to the PPN-site.

The number of names that were repeated is 107 for native speakers and 145 for non-native speakers.

It is interesting to note, that there were more non-native speakers that evaluated the pronunciations and corrected the phonemic transcription than native speakers, 164 vs 117. This is an unexpected result. Such non-native speakers clearly do not adjust to the user profile (c) provided in section 4.2.1. However, this does not represent a source of noise, since we can easily separate phonemic transcriptions entered by non-native speakers from those entered by native speakers.

For this paper, we did not do any formal evaluation of the family language origin accuracy. But informal inspection showed us that some users do not know what the family origin of names is. In order to draw any conclusions about this, we will need to conduct a formal evaluation, which is left as future work.

In interpreting the results in this section, it is worth noting that *not* all names are in the CMU dictionary, whereas the baseline and the 5 family model will predict a pronunciation for every name.

Another important point to keep in mind when looking at the results is that most of the names that were queried to the PPN-site are fairly uncommon and, therefore, are hard to pronounce. See Figure 2 for a small sample picked at random from the log.

(…) Ho Hai Thuy, Maeve, Vlissides, Leanne, Eissfeldt, Kecia, Sanjay, Jolene, Lengkeek, Langcake, Recchia, Siobhan, Zeme, Banga, Zbyslaw, Shervin, Higginbotham, Cazel, Kunigunde, Waltho, Gytha, Swoger, Iseli, Ruczynski, Skrenta, Tolles (…)

Figure 2: names picked from the PPN-site log

5.2. Comparing native and non-native speakers

Even though, for our purposes we are mostly interested in the native speaker results, it is informative to compare the native speakers with non-native speakers.

An interesting question to try to answer with this data is what percentage of the pronunciations each model alone would get the best score. Some pronunciations were the same for some of the models, which meant that more than one model, got the "best" score on several occasions (i.e. percentages do not add up to 100). Results are shown in Table 3.

Since our ultimate goal is trying to improve over the baseline by building a better set of LTS rules, which would be applied when having to pronounce an out of vocabulary word, even a more interesting question is: what is the proportion of better pronunciations generated by using the CMU dictionary with only one of the two LTS rules?

Models	native speakers	non-native speakers
CMU	68.84%	56.52%
baseline	79.79%	84.78%
5 family	58.90%	65.21%
CMU+base	95.89%	95.65%
CMU+5fam	92.12%	88.41%

Table 3: percentage of pronunciations that each model would get better than any other model if used by itself. The last 2 rows indicate the proportion of times we would get a better pronunciation if using the CMU dictionary with only one of the LTS rules.

The answer is that the CMU dictionary together with the baseline model get better pronunciations a bit more often, but not significantly so.

The fact that CMU+base and CMU+fam5 are much closer than the baseline and the 5 family model indicates that the overlap of pronunciations between the CMU dictionary and the baseline is larger than the CMU dictionary and the 5 family model.

Even though the 5 family model has higher word accuracy than the baseline according to our objective evaluation, the results of subjective evaluation show that most of the time users prefer the pronunciations produced by the baseline model.

These are the opposite results we were expecting to see, since the objective evaluation results described in section 3 support the fact that the model incorporating family language futures is slightly better than the baseline model.

It is also interesting to note that the non-native users thought that the model incorporating the 5 family features was actually a bit better than the native users. This might be because non-native users actually appreciate more a model that tries to mimic the "original" pronunciation, whereas native speakers prefer the more "Americanized" pronunciation.

5.3. Different classification parameters

In the rest of the section, we concentrate on users that are native speakers of US American English. We compare the results according to different classification parameters.

There are many interesting observations that can be made by looking at the results in Table 4. First, it is important to note that not all the classes are of the same size and that some are actually quite small (the class of people with an associate degree has only 19 users).

The few users holding an associate degree clearly deviate from the general trend and actually prefer the 5 family model over the other two models (89.47% of the time it was considered to do better than the other two).

The 47 users with a PhD, on the other hand, clearly preferred the CMU dictionary pronunciations to the ones produced by any of the LTS rules. This is also the class who thought the 5 family model was worse (39.29%). Hence, according to these results, the model that we thought would be mostly appreciated by educated users, turned out to be the one they disliked the most.

Classification parameters	CMU	baseline	5 families
E: other[4] (22)	70%	77.08%	58.75%
E: high school (52)	63.46%	92.31%	59.61%
E: associate (19)	52.63%	84.21%	**89.47%**
E: bachelors (85)	64.71%	75.29%	55.29%
E: masters (67)	77.61%	83.58%	65.67%
E: PhD (47)	**78.72%**	63.83%	**39.29%**
SS: daily (47)	59.57%	82.98%	63.83%
SS: weekly (55)	72.72%	74.54%	45.45%
SS: 1/month (88)	64.77%	82.95%	62.50%
SS: <1 /month (102)	**75.25%**	78.22%	60.40%
No other lang. (142)	69.72%	82.39%	60.56%
Other lang. (150)	68%	77.33%	57.33%

Table 4: Percentage of times a model was better than the others for native speakers according to their level of education (E), their familiarity with speech synthesis (SS) and whether they speak a language other than English.

Users highly exposed to speech synthesis (daily) seem to have a higher acceptance for LTS rules, and users who listen to speech synthesis less than once a month (could be never before), liked the CMU dictionary pronunciations more than the average.

Finally, a somewhat counter intuitive result is that users who did not know any language other than English rated slightly higher the LTS rules incorporating language origin information than the users who know (an)other language(s). This data seems to contradict the results from Table 3, that people familiar with other languages (non-natives) seem to prefer the more "foreign" pronunciation, possibly because they are more used to it.

6. Data collection

The phonemic transcriptions entered by native speakers as corrections to the ones generated by our models are retrieved from the database and are used to make a lexicon, which contains all the pronunciations proposed by users.

Even though there were 117 user corrections (see Table 2 above), there were only 61 different names the pronunciation of which was corrected. Users can correct a phonemic transcription as many times as they whish, and they are asked to rate each proposed correction.

Most users entered one or two phonemic transcription for a name, and some users corrected their transcription up to 6 times, resulting in 7 phonemic transcriptions for a name. In such cases, we are only interested in the transcription ranked higher by the user, the one s/he thinks is better. See Figure 3 for some examples of entries in the user proposed lexicon.

[4] In practice, education: other includes things such as PhD candidate, which should have been marked as bachelor degree or masters.

We implemented a spurious pronunciation filter that aligns the written names with the proposed phones to determine when the pronunciation is way off. We use this filter to detect which pronunciations proposed by native users are trustworthy and thus, could be used to correct the CMU dictionary entry.

```
("boutcher" (b aw1 ch er ))
("cipriano villa" (s iy p r iy aa n ow v iy ah ))
("dietz" (d iy1 t s ))
("jaroslav vrchlicky" (y aa1 r ow s l ah f v er1 hh l ih t s k iy )
("meineke" (m ay1 n ih k ))
("nazelrod" (n ey z ih l r aa1 d ))
("pershing" (p er1 zh ih ng ))
```

Figure 3: a sample of lexicon entries created from user proposed pronunciations

6.1. Improving the CMU dictionary

User studies have shown that the CMU dictionary has a considerable amount of noise (15.36%), however it is not possible for a human to go through all the names and hand correct the pronunciations that are wrong.

We can use the evaluation data to automatically detect the CMU dictionary entries that should be looked at, and possibly corrected. Another, somewhat more risky, use of the data collected through the PPN-site, is to automatically apply the spurious pronunciation filter to detect which pronunciations proposed by Native users are trustworthy and thus, could be used as the correct pronunciation.

In order to make such process completely automatic, there would have to be a number of people greater than an empirically determined threshold, who suggested the same phonemic transcription correction for a specific name.

7. Interlabeler Agreement

The results in section 6 do not gives us any insight on how much people agree on the quality of the pronunciations.

Even though we have tried to isolate native speakers of US American English, there is still plenty of regional variation left.

What a person from New Orleans judges as the correct pronunciation of a name might not coincide with what a person from New York thinks is correct. Thus, there is the need to determine whether different users agree on the scoring of the pronunciation, and if so, how much.

Even after collecting a significant amount of data, we did not have enough different people evaluating the same name to be able to measure agreement on the data from the general PPN-site. Thus, we set up controlled user studies, which ask users to rate only 10 names, the same 10 names for all users participating in the user studies (Scorcese, Traugott, Aileen, Dombey, Nietzsche, Hishaam, Muhammad, Satidchoke, Nguyen, Eratosthenes).

To determine what 10 names we were going to use for the user studies, we picked the last 50 names from the general PPN-site log, and after synthesizing all of them according to all the models, we selected the ones that had more than one different pronunciation and which pronunciations were significantly different.

Because most of the names were uncommon, this resulted in an unusual distribution disfavoring the CMU dictionary, and boosting the scores for the 5 family model. However, recall that the purpose of the user studies is to determine interlabeler agreement (ILA), also known as coder or rater agreement, not to evaluate the models.

There are different ways to measure ILA, the appropriateness of which is determined by the task. In the remaining of the section, we describe a couple of ILA measures and present the results.

7.1. Kappa coefficient

There seems to be some consensus that kappa statistic is an appropriate measure to evaluate ILA.

The kappa statistic tells us how different the results are from random and whether or not the data produced by coding is too noisy to use for other purposes for which it was collected (Carletta, 95).

More formally, the kappa coefficient (k) measures pairwise agreement among a set of coders making category judgments, correcting for expected chance agreement:

$$k = \frac{P(A) - P(E)}{1 - P(E)}$$

where P(A) is the proportion of times that the coders agree and P(E) is the proportion of times that we would expect them to agree by chance.

Carletta (95) only gives examples of P(E) for the case when there are only two coders, however in our case, we have 10 coders. When the number of coders is larger than 2, the P(E) has to take the number of coders into account as well as the number of categories. The following formula for P(E) has the appropriate behavior:

$$P(E) = \frac{num\_cat}{num\_cat^{num\_coders}}$$

k= 0 means no agreement other than the one expected by chance, k = 1 mean total agreement.

Content analysis researchers generally consider $k > 0.8$ as good reliability, with $0.67 < k < 0.8$ allowing for tentative conclusions to be drawn.

Tasks for which kappa has been most successfully applied have nominal categories and there are a relatively small number of coders.

The task at hand has a gradable scale of values, from more acceptable to less acceptable. The important information, is whether all or most users agree as to which model was ranked higher, rather than giving them the same exact labels.

For example, it is very possible that a user has high tolerance and thinks that everything is more or less acceptable, so s/he is going to rank the best pronunciation as correct and the second best as acceptable. On the other hand, there will be other users who are less tolerant and will think all pronunciations are wrong, but still will rank the best one as being more acceptable.

The important thing for us to know is whether the users agreed on which model was ranked higher for every one of the 10 names. We tried to capture this with the "soft kappa" measure, where P(A) is the proportion of times coders agreed on a model deserving the highest score for a pronunciation.

Alternatively, we can reduce the classification problem to a binary one by collapsing 'correct' and 'acceptable' into GOOD and have 'unacceptable' to be BAD. As expected, the kappa score goes up, and in fact it falls within the range of what has traditionally been considered reliable kappa scores.

As we can see from Table 5, there is enough agreement between what users consider to be acceptable and unacceptable in this task, and thus we can draw conclusions from it.

Kappa statistics	native speakers		non-native sp.	
	3 cat	binary	3 cat	Binary
hard kappa	0.497	**0.749**	0.408	**0.636**
soft kappa	0.538	n/a	0.467	n/a

Table 5: kappa statistics comparing the 3 category user evaluations with the reduced, binary case.

In spite of its popularity, kappa's value is at least controversial. It has been shown that kappa may be low even though there are high levels of agreement and individual ratings are accurate (Uebersax, 2000). For this reason, we discuss some alternative cumulative measures next.

7.2. Alternative cumulative measures

The motivation behind our user studies was to make sure that there is agreement on which model people find more accurate.

For this purpose, we also calculated model cumulative scores and cumulative scores weighted by user idiosyncrasies.

Score values 'correct', 'acceptable' and 'unacceptable' have 3, 2 and 1 as internal values respectively; the higher the score, the more acceptable it is.

By model cumulative score we mean the mere sum of all the scores that users gave to that model. The model that gets the higher score is the one preferred.

Similarly, we can get user cumulative scores by adding up all the scores that particular user gave to all the pronunciations. The difference between the user that has the highest score and the user that has the lowest score is called the user score range.

Once we have user cumulative scores, we want to determine how much each user should affect the final cumulative score (every user should have the same amount of *correctness* s/he is allowed to assign). First, we calculate the mean score over all users. To get the user weight we divide the score given by the user by the user cumulative score times the mean score. This effectively compensates for user idiosyncrasies, if a user tends to score all pronunciations as correct, his or her weight will be lower than the average of users.

The weighted cumulative score is the cumulative score as described above but taking user weights into account.

Table 6 illustrates that the weighted cumulative score, which is corrected for user idiosyncrasies, is almost the same as the cumulative score. The ranking of models is maintained, which means that there is reasonable user agreement.

Alternative measures	CMU		Baseline		5 family	
	3c	bin	3c	bin	3c	bin
cumulative score	20.63 %	19.21 %	36.99 %	38.92 %	42.28 %	41.87 %
user score range	17 [38-55]	6 [38-44]	17 [38-55]	6 [38-44]	17 [38-55]	6 [38-44]
weight. cum.	20.92 %	19.28 %	36.86 %	38.81 %	42.21 %	41.89 %

Table 6: model cumulative scores, user score range and user weighted cumulative scores comparing the 3 category user evaluations with the reduced, binary case.

5 family model scores are higher than both the baseline and the CMU dictionary scores for this data. This is due to the skewed choice of names for the users studies. Recall that this scores are not valid to evaluate the models, but rather the interlabeler agreement.

8. Conclusions

Using the internet as the medium to collect subjective evaluations has proven to be successful in terms of number of evaluations and data collection. The web makes our evaluation widely accessible and thus it allows us to collect large amounts of data, which would have otherwise been impossible to gather in such a short time.

Nevertheless, the web-base approach to evaluation has some drawbacks. It is impossible to control the quality of the evaluations, that is, to automatically separate the noise from the serious evaluations.

Traditional evaluations are conducted in a very controlled environment, where the evaluators make sure that the users understand the instructions before proceeding to do the evaluation.

Uncontrolled, online evaluation, on the other hand, involves a significant amount of risk. Even though there are instructions on the first page of the PPN-site, there is no guarantee that users read them. This increases the chances of each user understanding 'correct, 'acceptable' and 'unacceptable' in a different way.

Another related issue is that what in principle was a feature of this approach, i.e. being widely accessible, turns out to be not so benign in the end. Even though we are after educated users of US American English, there is no guarantee that the users who claim to be native speakers are actually native speakers of US American English. It is easy to misread the question as "native English speakers", in which case, we get evaluations from Australians, Scottish, British, etc. classified as native US English speakers.

Therefore, the uncontrolled nature of online evaluations forces us to take the subjective evaluation results with a grain of salt.

There is a significant amount of system engineering involved in setting up a working system such as the one described in this paper. Furthermore, when hundreds of people query the PPN-site, issues such as storage, efficient retrieval as well as exploitation of large amounts of data arise.

Finally, we would have liked our subjective evaluation to support the objective evaluation results, and show how the LTS rules that take family language origin into account are better than the baseline LTS rules. However, the opposite turned out to be true, and we are forced to face the fact that, according to the data collected from the PPN-site, most users think the baseline LTS rules produce better pronunciations.

9. Future Work

Once we have collected enough data, we will be able to present users with statistics about what previous users thought the best pronunciation for a queried name was. So far, we have already had many more queries than evaluations (1310 queries, 477 evaluations). This suggests that there is a demand for such a site, which given a name, tells you its pronunciation.

We are planning to use the data collected through the PPN-site to correct the CMU dictionary pronunciations that were rated as unacceptable by several users, and to explore automatic, but safe, ways of doing this.

Once the proper name pronunciation database is mature enough, we will make it available to the research community.

10. References

Black, A. W, Taylor, P. and Caley, R. 1998. *The {F}estival Speech Synthesis System*. http://festvox.org/festival

Black, Alan W, Lenzo, Kevin and Pagel, Vincent. 1998. *Issues in Building General Letter to Sound Rules*. 3rd ESCA Speech Synthesis Workshop, pp. 77-80, Jenolan Caves, Australia.

Carletta, Jean. 1995. *Assessing agreement on classification tasks: the kappa statistic*. ACL 95.

CMU Speech group. 1998. The Carnegie Mellon Pronouncing Dictionary: http://www.speech.cs.cmu.edu/cgi-bin/cmudict

Coker C.H., Church, K.W. and Liberman, M.Y. 1990. *Morphology and Rhyming: Two Powerful Alternatives to Letter-to-Sound Rules for Speech Synthesis*. ESCA Speech Synthesis, Autran, France.

Font Llitjós, Ariadna and Black Alan W. 2001. *Knowledge of Language Origin Improves Pronunciation Accuracy of Proper Names*. Eurospeech 2001, Aalborg, Denmark (vol. 3, pp.1919-1922).

Font Llitjós, Ariadna. 2001a. *Improving Pronunciation Accuracy of Proper Names with Language Origin Classes*. Masters Thesis (Technical Report: CMU-LTI-01-169).

Font Llitjós, Ariadna. 2001b. Pronunciation of Proper Names site: www.pronounce-names.org

Uebersax, John. 2000. *Kappa Coefficients*. http://ourworld.compuserve.com/homepages/jsuebersax/kappa.htm

U.S. Department of Commerce. 2002. *A Nation Online: How Americans Are Expanding Their Use Of The Internet.* Economics and Statistics Administration. Telecommunications and Information Administration. February 2002.