Culture Clubs

Processing Speech by Deriving and Exploiting Linguistic Subcultures

by

David Guy Brizan

Thesis Proposal for the degree of

Doctor of Philosophy

at the

Graduate Center of the City University of New York

Committee: Professor Andrew Rosenberg (Advisor and Chair)
Professor Liang Huang
Professor Rivka Levitan

May, 2015
Contents

1 Abstract 1

2 Introduction 2

  2.1 Thesis ......................................................... 4

  2.2 Expected Contributions ...................................... 4

  2.3 Possible Applications ...................................... 5

3 Proposed Work .................................................. 8

  3.1 Linguistic Subculture Detection ............................ 11

    3.1.1 Discourse Markers .................................. 18

  3.2 Evaluation with Speech and Language Processing Tasks ................. 19

    3.2.1 Automatic Speech Recognition ........................ 20

    3.2.2 Disfluency Detection .................................. 23

    3.2.3 Sentence Segmentation ................................. 27

    3.2.4 Dialogue Act Detection ................................. 31

4 Summary of Results - The ComParE 2015 PC Sub-Challenge ................. 37

  4.1 Background .................................................. 37

  4.2 Related Work .................................................. 38

  4.3 Method ......................................................... 39

  4.4 Results ......................................................... 41

  4.5 Discussion ................................................... 43

5 Timeline 45
1 Abstract

Spoken language understanding systems are error-prone for several reasons, including individual speech variability. This is manifested in many ways, among which are differences in pronunciation, lexical inventory, grammar and disfluencies. It would be ideal to create models for individuals, but this strategy is impractical and likely ineffective. There is, however, a lot of evidence pointing to stable language usage within subgroups of a language population.

I propose exploiting linguistic subcultures for speech and language processing tasks. A linguistic subculture is a language subgroup formed by people sharing an accent, dialect or demography. I describe an unsupervised method for deriving linguistic subcultures along with tests for determining the effectiveness of this technique using four tasks: automatic speech recognition, disfluency detection, sentence segmentation and dialogue act detection.
2 Introduction

Consumer-level spoken language understanding (SLU) systems such as Apple’s Siri [62], Google’s OK Google [51] and Microsoft’s Cortana [99] have become popular to the point of being ubiquitous. They can recognise the human voice, parse those into individual words and phrases, retrieve information using web search and act on commands. However, these systems are far from perfect at speech tasks. Huang et al. (2014) [53], for example, points to 40 years’ worth of innovation and research showing performance increases via declines in Word Error Rate (one measure of SLU system performance); however, the performance of systems still falls far short of human performance but still worse than human performance on speech recognition.

SLU processing is difficult not only because of the ambiguity inherent to language in general but also because of individual speaker idiosyncrasies: the speech signal is highly variable, subject to coarticulation and pronunciation differences between speakers [95] and subject to repeats, restarts and other speaker-generated disfluencies [14], both of which affect the recognition of individual words and generate errors in automatic speech recognition and SLU/SLP systems.

SLU systems are notoriously sensitive to regional accents or dialects [106], the “language characteristics of a particular population where the categorization is primarily regional” [19]. Examples include the difference between English spoken in the USA, the UK or India. While there does not appear to be consensus about the total number of dialects in the world, we may get a sense of the numbers from estimates of a few languages. A 2009 report by the United Nations counts 45 dialects of Russian in the Northern Caucasus and Siberia [75]. In a country of fewer than ten million, Sweden has six dialects [82]. And in the USA, the count of dialects ranges from three [60] to dozens, especially if cultural and bilingual dialects are included [112].

Although speech is highly variable and subject to individual differences, the evidence above points
to a number of linguistic subcultures: class-based similarities affecting acoustic realisation, language production, lexical choices, grammar and word meaning. The definition of a linguistic subculture is closely tied to the sociological concept of subculture: a group differentiated from a larger culture while maintaining some of its principles [48]. Formally, I define a linguistic subculture as group of people who use a language but can be differentiated from other speakers by the manner in which the group uses this language. This differentiation may manifest itself as shifts or inventions in grammar, prosodic, pronunciation or lexical usage when compared to the umbrella language.

An important aspect of the definition of linguistic subculture is that a speaker must remain relatively consistency within a (sufficiently large) window of time. Despite their uniform effect on speech production, I reject certain elements, such as the noise in an environment, as defining a linguistic subculture because of their temporary effect on the speaker and the speech produced.

In no particular order, these linguistic subcultures may be the result of: dialect (not only as a “regional accent” but also as an indicator of race or socioeconomic status) [103], age [86, 18], (which includes normal age-related hearing loss [15] and possibly leading to disfluencies), the demographics of the interlocutor [4], the speaker’s sexual orientation [63], gender, environment and speaking style and rate [87, 9]. Within a linguistic subculture, I hypothesize that the language signal will be sufficiently stable within a class to yield computational advantages.

In one recent automatic speech recognition system, Najafian et al. (2014) [73] divided the motley speech of the United Kingdom into a number of pre-determined geographically-based dialects. The models were tested by selecting the appropriate dialect based on accent ID, a phonotactic analysis. This confederated system showed a reduction (performance improvement) in word error rate of up to 60% relative (approximately 35% absolute) for speakers from Glasgow, Scotland and potential improvements for all groups, an average of 47% relative word error rate reduction on average. This
promising pilot study shows the power of employing one aspect of linguistic subculture to aid SLU systems.

2.1 Thesis

A speaker’s linguistic subculture has a number of realisations, including prosody, vocabulary and grammar; using unsupervised models built on other speakers in the same linguistic subculture leads to computational advantages in speech and language processing.

2.2 Expected Contributions

In defense of the thesis statement, I will offer the following contributions:

- **Determination of linguistic subcultures**: A system for clustering speakers into linguistic subcultures using unsupervised algorithms. Additionally, this system shall be tested for the efficacy of the clustering where the data is appropriately labelled.

- **Determination of Language Differentiation**: A system for determining the relative improvement of models built on data from a speaker’s linguistic subculture as opposed to models built on data from outside the speaker’s linguistic subculture. These models include the following:
  - Automatic Speech Recognition
  - Recognition of Disfluency
  - Sentence Segmentation
  - Recognition of Dialogue Acts
2.3 Possible Applications

This work may have a number of important applications. The implication to speech language understanding systems and speech research is clear, with obvious performance improvements to automatic speech recognition. Other tasks in speech processing include, but are not limited to: disfluency detection and processing, sentence segmentation and recognition of dialogue acts, such as agreement or sarcasm. I will tackle these topics in this work.

While the work in this proposal is focused on the problem of regional accents because of the corpora available, a closely related problem, that of second-language (L2) speakers speaking with foreign-language accents, also defines another set of linguistic subcultures. Typical SLU systems tend to be have lower accuracy when processing L2 speech because of the shift away from the majority of native language (L1) phonotactics [106], i.e. because of an foreign-language accent. L2 speakers are at least as problematic for SLU systems, and, because the production of SLU systems is not trivial and target one or a limited number of dialects of languages, these speakers find interacting with SLU systems difficult [53]. As shown by Figure [L1-DISTRO], the native language of the eleven most common languages in 2009 covers less than 40% of documented languages. We might infer that at least one of these “most common” languages (Javanese) is a “low resource” language because of its recent inclusion in the Spoken Language Technologies for Under-resourced languages (SLTU) conference [92]. Given this, most of the world’s speakers would access SLU systems using a second language and would benefit from the outcome of this work.

Many language analysis tasks could extend easily from this work. For example, this work could be used to automate the language analysis portion in the task of Language Analysis for the Determination of Origin (LADO). This process is used on behalf of political refugees seeking asylum in a friendly country. Upon migration, an asylum seeker’s language habits may be examined for
competence in languages and association to regional dialects or regional patterns. The output of this analysis phase is used to determine the speaker’s origin and therefore asylum status. While the analysis phase may be conducted in an automated fashion, the interpretation of results often requires a trained linguist who is familiar with the demographic pressures of the region in question [16].

This work could also facilitate the automation of the production of regional dictionaries for unique lexical entries, pronunciation or meaning. One way of generating these pronunciation dictionaries is shown by the International Dialects of English (IDEA) [71]. Although it was originally envisioned as a tool for actors, it is become valuable to customer service representatives and people in other fields. A dialogue system could be created from the speech signals of one linguistic subculture and could generate prompts to the human interlocutor using text-to-speech (TTS). Such a system could incorporate the unique phonotactics, lexicon and syntactic structure of a target group of people.

A linguistic subculture also indicates a speaker’s state, such as whether the speaker is intoxicated [12], assuming that intoxication is stable within the time of speech production being considered. It may also indicate whether a speaker suffers from speech-related pathologies, such as stuttering [14], or indeed any pathology which has a speech-related manifestation, such as depression [25, 26], autism spectrum disorders [24] or Parkinson’s Disease [54].

Finally, this work may be a tool for evidence-collection and analysis of sociological changes. Some evidence already exists to provide support for small language differentiation among young educated American women [114, 8]. While important and interesting, these analyses only creep toward the goal of providing answers to questions posed by in Putnam [1] about the collapse of American civic society. His book, *Bowling Alone*, provides evidence that participation in civic, political and social life by people in the U.S.A., strongly praised in de Tocqueville’s 1835 observations, has seen declines
lately as the population substitutes civic participation for membership in interest groups, such as the American Association of Retired Persons (AARP). According to the thesis of this book, new lifestyles which avoid civic engagement lead to fewer opportunities for communication with people of different ages, races, religions and other demographics which contribute to a person’s linguistic subculture.

Given careful data collection and labeling, the work I propose here could be used as a vehicle for determining whether continual engagement with people outside one’s linguistic subculture can be quantified and to determine whether the effect represents a generational societal change. In fact, taken to its logical conclusion, this work could be a linguistic tool used to analyse broad swaths of sociological change over geographic or generational distance as easily as it could determine the development of online micro-communities, albeit using a signal such as language models for clustering people into groups.
3 Proposed Work

My proposed work has two major foci: the unsupervised determination of linguistic subcultures and, once determined, the evaluation of the efficacy of linguistic subculture processing with certain speech and language processing tasks.

The work involves testing on a number of corpora of spoken English listed in Table 1. Each corpus provides a number of annotations related to linguistic subcultures including dialect, gender and age labels. These labels will be used as ground truth for the linguistic subculture detection work.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Speaker Labels</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accents of the British Isles (ABI) [31]</td>
<td>Speech samples from 14 regions of England, Ireland and Scotland composed of read numeric and alphabetic sequences (e.g. “BKUN”), words, sentences, short phrases and passages.</td>
<td>Speaker Region</td>
<td>Subculture Detection, ASR, Sentence Segmentation, Dialogue Act Detection</td>
</tr>
<tr>
<td>Fisher [20]</td>
<td>Spontaneous telephone conversations between strangers on a set topic.</td>
<td>Speaker Region, Age, Gender, Native Language, Education</td>
<td>Subculture Detection, ASR</td>
</tr>
<tr>
<td>IViE [42]</td>
<td>Speech from 9 regions of England and Ireland composed of read statements, and questions as well as semi-spontaneous and an interactive map task.</td>
<td>Speaker Region</td>
<td>Subculture Detection, ASR</td>
</tr>
</tbody>
</table>

Table 1: Corpora Descriptions, Labels and Tasks

The Accents of the British Isles (ABI) [31] and Intonational Variations in English (IViE) [42] cor-
pora describe several regional differences in Great Britain (England, Ireland, Scotland and Wales), where regional differences in the English language may be broader than in other English-speaking countries. While the focus of the ABI corpus is pronunciation differences, IViE was assembled with an eye toward prosodic regionalisms, such as the way questions are asked.

The ABI corpus contains read speech of 14 “accent groups” pre-defined by geographic location. There are ten speakers per gender for each region, approximately 70 hours’ total. Speakers were asked to perform a number of tasks, including:

- Reading 10 “catalogue codes”: 4 alphabetic characters (e.g. “BKUN”) intended to be spoken as a unit (e.g. “bee kay you enni”) and in ICAO alphabet (e.g. “bravo kilo uniform november”).

- Reading “equipment control” and “game control” commands as may be used in the control for a car or video game. Examples include “play disk two track three” or “insert waypoint.” These commands include phone numbers.

- Reading sequences of four numbers, such as PINs, intended to be spoken as a unit.

- Reading a number of “careful words” containing a sequences of an h-sound, a vowel and a d-sound, such as “heed.”

- Reading a number of sentences and short phrases, such as “an official deadline cannot be postponed.”

- Reading all or a portion of a passage (the “sailor diagnostic passage” from SCRIBE) designed to elicit the speakers’ regional accents. The speakers’ reading of this passage may be divided into many WAV files and accompanying transcripts.

All of the ABI corpus is transcribed and time indexed. Some of the transcription is missing where
it was not part of the the stimulus. (For example, Birmingham speaker 001 utters the word “no” after the first attempt at reading the “sailor diagnostic passage” to indicate a production error. This word is missing from the transcript.) In addition, the transcript has been marked for disfluencies; however, since, my investigation of the corpus finds several problems with the disfluency annotation, I have omitted this task from Table 1.

Subjects in the IViE corpus were prompted to recite stimuli from two categories: a re-telling of a short story about the Cinderella fairy tale and a number of simple declarative sentences and single-clause questions, such as “Are you growing limes or lemons?” Data collection took place in 9 cities throughout England and Ireland with the assumption that speakers from each city use the same regional accent. The speech samples show regional pronunciation differences as well as differences in intonation.

The English language portion of the Fisher corpus [20] was collected specifically for automatic speech recognition, from December 2002 to December 2003. The portion of the corpus which I will use contains over 11 thousand telephone conversations between strangers on topics set by the collection protocol. Each conversation lasts an average of 10 minutes. Speakers range in age, gender, level of education, place (mostly U.S. state but also non-U.S. countries) where raised and first language.

All the conversations have at least some portion transcribed. As this is the corpus with the most data and most extensive set of labels, this corpus may be useful in determining which demographic and socioeconomic labels can be derived given the methods outlined in this proposal.

Like Fisher, the Switchboard corpus [39] contains a few thousand conversations (2,438) between strangers on set topics, each lasting a few minutes. The participants are all native English speakers living in the U.S.A. at the time of data collection. Speakers are labeled with one of the dialect
regions mostly corresponding to Figure 1, which is taken from [21]. Unlike the Fisher corpus, the original purpose of the corpus was speaker identification, but it has since been used for wider SLU research, including all of the tasks outlined in this proposal. In fact, of the conversations contained in the corpus, 36 have been annotated for disfluencies, for sentence segmentation and for dialogue acts, among others.

At minimum, these four corpora will be used for the work outlined in the following subsections. I will consider incorporating additional corpora which are compatible to the tasks. Each of these corpora will be used as the input stream for the Linguistic Subculture Detection task. In addition, some may be used for other tasks as indicated in Table 1 as detailed below.

### 3.1 Linguistic Subculture Detection

My proposed system for determination of linguistic subcultures will be built on established work in the areas of dialect identification, sometimes conflated with accent identification, akin to language identification. Linguistic subculture may be informed largely by dialect, so I include an overview of this task below. However, other important signals, such as prosodic contour (voice intensity, fundamental frequency, speech rate and rhythm) [44] and language usage may also be important in
teasing apart subcultures within the same dialect. My approach to linguistic subculture detection will take an approach identical to dialect identification with the exception of including features important in determining finer-grained discriminations.

Language identification (LID) is the identification of a speaker’s language from an acoustic signal alone [118], which may be useful for domain adaptation: applying a language-specific lexicon, language model or grammar [69], for example. Dialect identification, which distinguishes dialects of the same language, is considered a more challenging problem because many artifacts which distinguish languages, such as phonotactics, lexicon, morphology and syntax may be shared among dialects [11], although Grabe [38] claims that the differences between dialects can be larger than that of two languages. Regardless, some combination of these distributional differences may be used to determine the linguistic subculture of the speaker. Biadsy (2009) [11], for example, lists several differences among Arabic dialects such as the Modern Standard Arabic consonant (/θ/) which is absent from other Arabic dialects. Likewise, Mousa (2014) contrasts the use of the interdental fricatives /θ/ and /ð/ between two L1 Jamaican English speakers living in England. One retains a Jamaican dialect for lack of interaction with the larger population; the other, with closer ties to the non-Jamaican population, has a pronunciation closer to the Received Pronunciation dialect (RP) of his environment.

State of the art systems have been used to distinguish between English dialects (American vs. Indian and Southern American vs. Non-Southern American) [12] among several Arabic dialects (Gulf, Levantine, Egyptian, Iraqi and Modern Standard Arabic) [11], between two Spanish dialects [117] and among three Chinese dialects [68].

Languages (and dialects) are different from one another in terms of their phonology (the sound inventory of the language), morphology (the words used), syntax (the classes and structures of
the words used) and prosody (the rhythm, timing and stress used in speaking) [118]. I will take advantage of the phonological and prosodic differences in speech by using one or both of two state of the art approaches, a PRLM (phone recognition to language model) approach and an i-vector approach.

The PRLM approach [11, 57, 118] is a pipeline which starts with individual (vowel and consonant) phone recognition (PR) from an acoustic signal. During training, the output of this recognition is used to build an n-gram (language) model (LM) which captures the phonotactic probability distributions of the language or dialect. During testing, the output is used to compare against existing models, the most likely of which becomes the hypothesis for the label of the language or dialect. PRLM has an effective variant in which multiple parallel phone recognition systems, each trained on different languages or dialects, are used in parallel, with the output hypotheses of those systems combined with a classifier for final prediction [11].

Chen et al. (2010) [19] claims that phone recognizers fail to capture acoustic differences across dialects, such as the retroflex /d/ common in Indian English. As a result, parallel PRLM, which employs PRLM in multiple languages. The “marry-merry-Mary merger” is an example: though most speakers of Standard American English hear these as homophones, some speakers in New England produce a distinct sound for each [29]. This distinction would be “inaudible” to a PRLM system built on Standard American English; however, other languages, and perhaps other dialects of English, may be sensitive to the distinction and therefore to the linguistic subculture from which the sounds are produced.

Chen et al. (2010) [19] show that relative pruning of the least common biphones improves the performance of a baseline PRLM system. In this work, biphone pruning rates between 10% and 37% were varied to determine an optimal equal error rate (EER). An 18% rate of pruning of biphones
yielded optimal results, and rates as low as 29% were found to prune so aggressively as to give results indistinguishable from a monophone system.

Similarly, albeit on keystroke data, work described in [40] shows that pruning of events occurring fewer than an absolute number of times is also effective as measure by EER. In this system, the largest improvement was generated by including events occurring only greater than 7 or 8 times.

I will experiment with these two strategies (relative pruning and absolute pruning) in my system. For each, I will use and different pruning thresholds and report results of each in aligning the derived clusters against the labels in the corpora.

The second approach will use i-vectors. An i-vector is a low-dimensional representation of speech which aims to capture the speaker and channel variability given one or more utterances [27]. It works by creating a “Universal Background Model” (UBM), $m$, representing the total variability of all speaker utterances in a supervector – a set of stacked mean vectors from a Gaussian Mixture Model (GMM) [97]. Each speaker utterance, $M$, can be described by the UBM, by a matrix of features, $T$ and a vector representing the identity of the speaker – the i-vector – $w$, all described in by (1).

\[
M = m + Tw
\]  

(1)

The work for i-vectors was suggested as a solution to the problem of speaker recognition, specifically the problem of identifying whether an utterance was generated by a particular speaker known to a system. It has since been used for other tasks, such as speaker diarization [97], identification of a speaker’s cognitive state [107] and dialect (accent) identification [73].

For linguistic subcultures, the term $w$ in (1) may be the cultural identity, such as may be derived from a set of regional speech samples. Previous work assumes a priori knowledge of identity.
Once i-vectors have been computed, a normalization step may be performed to compensate for channel noise. Two compensation steps are suggested by [27]: within-class covariance normalization (WCCN) or linear discriminant analysis (LDA). Finally, similarity (or differences) between utterances may be calculated with cosine distances as given by (2) [98]. Where the distance falls within a confidence for a given speaker, x, the utterance is determined to have been produced by speaker x.

$$\text{cosine score}(w_1, w_2) = \frac{(w_1)^t(w_2)}{||w_1|| \cdot ||w_2||}$$  \hspace{1cm} (2)

Shum et al., 2011 [97] describe a multi-step approach of post-processing i-vectors to derive the gender of the speakers. In this approach, Principal Components Analysis (PCA)-based projection is applied as a proportion of i-vector dimension. K-means clustering is applied to the output of this step based on cosine distance. (In this work, K was set only to 2 for the purpose of bifurcating gender. Additional refinements were applied to generate better segmentation for speaker diarization; these are omitted here.)

Inspired by these state of the art dialect identification systems, I will build two linguistic subculture detection systems. The first system will be the implementation of a phonotactic approach, using a custom feature extractor based on the phone sequences generated by phnrec [69] in four languages (Czech, Hungarian, English and Russian) and the phone sequences generated by Kaldi [83]. These sequences will be placed in a system of monophones, diphones and triphones. The set of features will be pruned using the relative pruning and absolute pruning strategies, as described above, ultimately generating a point in multi-dimensional space to describe the utterances of the speakers. Distances in the distributions between utterances in this space will be used as a measure of similarity (or distance).
Varying different parameters to produce an exploration of the search space – number of clusters and neighbourhood size, for example – I will cluster similar speakers using the algorithms k-means, expectation maximization (EM), density-based spatial clustering of applications with noise (DBSCAN) [10] and spectral clustering [76] as implemented in Weka [45], in scikit-learn [81]. Clustering refers to the division of objects in a data set such that similar objects are grouped together and dissimilar objects are placed in separate groups [10]. Clustering algorithms fall into three categories: those using hierarchical agglomerative algorithms, those using hierarchical divisive algorithms [89] and those using partitioning or density-based algorithms [58].

Hierarchical agglomerative algorithms, such as k-means or expectation-maximization (EM), build clusters from the “bottom up,” starting with the assumption that each object belongs to its own cluster. Clusters are joined according to distance until some convergence criterion is reached. Hierarchical divisive algorithms take the opposite tack: all objects belong to a single cluster, and that cluster is divided repeatedly until convergence. Partitioning algorithms, such as DBSCAN and ordering points to identify the clustering structure (OPTICS), identify dense areas of objects for connectivity, sometimes distinguishing between the central areas of these regions and the border areas. These algorithms have the advantage of not requiring an a priori number of clusters but also assume that density changes are equivalent to cluster borders.

In all clustering algorithms, the distance measure for features is important. In Table 2, I list the distance measures which I will use in my linguistic subculture detection systems.

<table>
<thead>
<tr>
<th>Name</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euclidean distance</td>
<td>( d = \sqrt{\sum_i (a_i - b_i)^2} )</td>
</tr>
<tr>
<td>Squared Euclidean</td>
<td>( d = \sum_i (a_i - b_i)^2 )</td>
</tr>
<tr>
<td>Manhattan distance</td>
<td>( d = \sum_i</td>
</tr>
<tr>
<td>Maximum distance</td>
<td>( d = \max_i</td>
</tr>
</tbody>
</table>

Table 2: Distance metrics for hierarchical clustering
Clustering algorithms, especially the density-based ones, have been shown to be intractable or to yield poor results on high-dimensional data because the data appear to be sparse. Dimensionality reduction through DFT and PCA has been shown to help. More recent approaches – for example, subspace clustering, in which only some features are used for clustering – have also been proven helpful [2, 56].

The second system will be built using an i-vector approach. For this, I will use Alize [13], a popular implementation of the i-vector approach. As with the phonotactic approach, the output of this defines a point in a multi-dimensional space representing similarity or differences among speakers. I-vectors have customarily been built over MFCC inputs, with post-processing analysis performed using cosine distance as described in (2). Clustering approaches [98] may also be useful in establishing linguistic subcultures if prosodic contour features and n-gram features are used as inputs to an i-vector system. I may therefore add these additional features such as n-grams. These additional features may be useful in fine-grained distinctions, such as generational differences between people from the same region.

In addition to the implementation of these two systems, I will evaluate the utility of combining the outputs of these two systems to form a unified linguistic subculture feature extractor. In each system, I will place special emphasis on the Fisher corpus because of its rich set of speaker labels. I expect the clustering approaches to derive groups based on some or all of these speaker labels. While these labels may represent a subset, superset or a set orthogonal to the linguistic subcultures of the speakers, these labels also represent reportable metrics of the system, so it will be interesting to determine which may be computed using this technique and, therefore, which may be helpful when speaker labels are unavailable.
3.1.1 Discourse Markers

Discourse markers, also known as cue phrases, are a potentially interesting signal which we can derive from this clustering. They are words or expressions with relatively little meaning but useful in signaling discourse planning, hedging or backchanneling. In English, words and phrases such as actually, basically, I mean, well may fall into this category [49, 67, 84]. I hypothesize that discourse markers will have different distributions depending on linguistic subculture, so we may see interesting results from generating a ranking of cue phrases by linguistic subculture.

One example of this difference comes from Wharry’s study of language in African American churches [111]. Having investigated various religious discourse markers (for example, Amen, Hallelujah, Glory) for frequency, the study finds that male preachers tend to be more faithful to one expression while their female counterparts exhibit more variety. Therefore, in the absence of other signals, this difference may be used to differentiate gender. Anecdotally, I find young, male African Americans and Latinos in New York utter “you feel me?” and “know what I mean?” at rates higher than the rest of the English-speaking population.

Three dozen conversations in the Switchboard corpus have labels for discourse markers. Once linguistic subcultures have been determined by the above clustering method, I will take the additional step of producing and reporting a list of discourse markers ordered by frequency for each linguistic subculture.

Since it is possible that the frequency of discourse markers may be a signal for predicting linguistic subculture, this work may be very useful if it can be extended to other corpora.
3.2 Evaluation with Speech and Language Processing Tasks

Four tasks will be used to determine the effectiveness of linguistic subculture clustering and processing: Automatic Speech Recognition, Disfluency Detection, Sentence Segmentation and Dialogue Acts. An overview of each of these along with the specific tasks to be performed, appears in the rest of this section.

The corpus will be segmented into training and test data as described in the details below. All four tasks will be evaluated using two model architectures. In a “unified” model architecture, I will use all the training data for the task. Once a model has been built, I will use the testing data to evaluate the performance of the model.

![Figure 2: Confederated Model Architecture](image)

In the second model architecture, the “confederated” version, I will create a number of models – one for each linguistic subculture – to describe the training data. I will create a Linguistic Subculture Selector to determine the closest match of each speaker in the test corpus. This selector will pass the appropriate model to the Task Evaluator, which will generate a report of performance. This is
shown in Figure 2.

3.2.1 Automatic Speech Recognition

Automatic speech recognition (ASR) is the act of translating spoken words to text [108]. This area is also known as speech recognition (SR) or as speech to text (STT). There are many applications for automatic speech recognition including data entry [80], financial transactions and information retrieval [88]. More importantly, of course, it is a critical step to many SLU tasks. Automatic speech recognition has a few variants based chiefly on the expected input of the speaker, including: speaking mode (isolated words vs. continuous speech), speaking style (broadcast speech, read speech, meeting speech, conversational and spontaneous speech and others) [77] and vocabulary size (small through large).

My work on the corpora listed in Table 1 will largely be considered LVCSR (large vocabulary, continuous speech recognition). In other work, two toolkits have been used for this task: HTK [113] and Kaldi [83]. Both are implementations of a Hidden Markov Model approach. These systems are built from three components: a Feature Extractor, an Acoustic Modeler and a Decoder, as shown in Figure 3. The Feature Extractor converts an acoustic signal into a vector of features representing the signal. Popular representations are Critical Band Energies (CRBE), and Mel-frequency Cepstrum Coefficients (MFCCs) [105]. The Acoustic Modeler converts this representation into a sequence of phone hypotheses. Finally, the Decoder takes the phone sequences, along with the Language Model and Pronunciation Model derived during training, to produce word hypotheses.

Initially, the Feature Extractor uses the input acoustic signal to produce a sequence of observation vectors, $O$, for each time slide in the acoustic signal. The sequence of observation vectors is defined
Figure 3: Example ASR pipeline

by (3), where \(O_t\) the vector \(O\) at time \(t\).

\[ O = o_1, o_2, ..., o_T \]  \hspace{1cm} (3)

For recognition of isolated words, the HMM system defines the probability of some previously seen
word, \(w\), given the sequence of observation vectors detected at time \(t\) as the word with the highest
probability given the observation, as shown in (4).

\[ \arg\max_{t} \{ P(w_i|O) \} \]  \hspace{1cm} (4)

Using Bayes’ Rule, (4) may be re-written as (5), which has the advantage of being computable from
previously-seen data. Specifically, \(P(w_i)\) may be derived from the language model: the frequency
of words seen during training; \(P(O)\) may be derived from the acoustic model, and \(P(O|w_i)\) may be
calculated from an estimation of Markov model parameters for a given observation, \(O\).

\[ P(w_i|O) = \frac{P(O|w_i)P(w_i)}{P(O)} \]  \hspace{1cm} (5)

The recognition of words in a continuous speech system involves further assumptions due to, for
example, lack of discrete word boundaries, coarticulation effects or noise. HTK, Kaldi and similar
HMM ASR systems assume that the transition from one pronunciation state to another is hidden.
The hidden portion of the HMM model, which is to be estimated by the system, is therefore the transition state sequence. With an initial estimate of the model parameters before training using the Baum-Welch formula, these model parameters are calculated during training using the Viterbi algorithm to find the most likely transition sequences and therefore the most likely word matching a given signal.

The performance of ASR systems have generally been compared with Word Error Rate (WER), with lower numbers indicating higher performance. Table 3 samples a few recent systems, the feature set used, models and their performance. The results shown in this table should be viewed in context of the data being considered: whether the environment has noise, the number of simultaneous speakers and a number of other considerations detailed in the sources.

<table>
<thead>
<tr>
<th>Source</th>
<th>Feature Set</th>
<th>Model(s)</th>
<th>Highest Performance</th>
<th>Lowest Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Du [30]</td>
<td>LMFB</td>
<td>DNN</td>
<td>16.7% WER</td>
<td>25.6% WER</td>
</tr>
<tr>
<td>Geiger [36]</td>
<td>MFCC</td>
<td>GMM / DNN</td>
<td>22.2% WER</td>
<td>66.0% WER</td>
</tr>
<tr>
<td>Najafian [74]</td>
<td>MFCC</td>
<td>HMM / GMM</td>
<td>4.0% WER</td>
<td>60.0% WER</td>
</tr>
<tr>
<td>Tuske-MFCC [105]</td>
<td>MFCC</td>
<td>DNN</td>
<td>19.4% WER</td>
<td>25.3% WER</td>
</tr>
<tr>
<td>Tuske-Time [105]</td>
<td>Time Signal</td>
<td>DNN</td>
<td>29.4% WER</td>
<td>36.8% WER</td>
</tr>
</tbody>
</table>

Table 3: Performance of recent ASR systems

Clearly, these results show a large variation in performance. However, my intention is not to build the best-performing ASR system but to build one which is reasonably good enough to test my hypothesis. This system will allow the application of different pronunciation models and language models built on speakers from the same linguistic subculture. The goal is the comparison of two ASR systems, different only in the training data used.

For my work, I will use either HTK or Kaldi as my ASR toolkit. All four corpora in Table 1 are annotated with data necessary for English language ASR. I will build a system to unify these into a large input stream. A portion of the input stream will be reserved for testing; the remainder will form the training portion. I will take care to ensure that no speaker appears in both the training and
test portions simultaneously and, where possible, that each linguistic subculture contains training data from multiple speakers.

The system I build will have two variants: a “unified” model and a “confederated” model. In the unified model variant, the system will use the entire training data for performing the ASR task. The confederated model, however, is a set of models, each representing a unique linguistic subculture, each derived as illustrated by Figure 2 and as discussed in the Linguistic Subculture Detection section above. In this variant, ASR will be performed in two phases: first, the speaker will be assigned to one or more linguistic subcultures as indicated by the features required to form the subculture clusters. Once assigned, ASR will be performed with only the subset of training data used to form the cluster. Where possible, I will test with a variable number of clusters.

Where applicable, I will report the out-of-vocabulary (OOV) rate as a possibly interesting measure. It is possible that certain lexical entries may exist in a particular linguistic subculture and not in others, so the OOV rate may provide insight into the degree to which a linguistic subculture model matches the speaker. I hypothesize this effect would be more apparent when two or more people of the same linguistic subculture engage each other. Ostensibly, the corpora I propose to use do not show this effect, so I will compare the effectiveness of these system variants using WER.

3.2.2 Disfluency Detection

Although the language of spoken utterances may have more relaxed grammars than that of written language, some grammar still defines them [23, 67]. Fluent speech is an uninterrupted sequence of words that follow the rules of syntax; a disfluency, therefore, is any sequence of words which violates these [14]. Disfluencies are a heterogeneous set of phenomena composed of filled and unfilled pauses, repairs, repetitions and revisions. They are caused by delays in speech, speech
planning issues, corrected errors, attempts to “hold the floor” and requests for help with language production from the interlocutor. The unified set of disfluencies represents between 6% and 10% of all words in spontaneous utterances [33] and about one-third of all utterances [79]. If speech processing systems are to interact with human speakers the way other people do, these systems must handle this common phenomenon gracefully.

Filler disfluencies are silent or vocal pauses within speech. Filler words include items otherwise outside the lexicon (for example, ah, eh, uh, um) or may appear as cue phrases or discourse markers (for example, actually, basically, I mean, well and others) [67, 84]. This phenomenon is discussed in detail in another section (Dialogue Act Detection). Work by Ostendorf and Hahn (2013) [79] considers filled pauses “trivial” to detect, perhaps because they are simply added to a lexicon.

Repeats and revisions, collectively known as edit disfluencies, are requests by the speaker to change to an already-delivered utterance. They generally have the form shown in (6). Edit disfluencies contain three parts: the reparandum, which is the originally-delivered utterance to be changed; the interruption point, shown as an asterisk, marks the end of the original delivery and the start of the repair; finally, the repeat or revision, which may also be marked by an optional edit term or interregnum and the repair or correction being the utterance to replace the reparandum [67]. Two examples are shown in Figure 4.

\[
< \text{reparandum} > \ast < \text{editterm} > \text{repair} \quad (6)
\]

Complex disfluencies involve multiple stacked filled or unfilled pauses, repairs, repeats or revisions [79].

Previous work suggests that each type of disfluency has a unique distribution depending on the
state or intention of the speaker [46]. For example, when the speaker knowingly addresses a system
(as opposed to another human), the rate of disfluencies drops to one-quarter the normal rate [14].
While their research was based on the differentiation of speakers by their overall disfluency rates,
Honal and Schultz [52] conducted an experiment on speaker-dependent disfluency detection, with
encouraging results. Bortfeld et al. [14] suggest that disfluencies are uttered at different rates
depending on the age and gender of the speaker, with males and older subjects uttering more fillers
and repetitions. Because others find demographic differences in disfluency production across these
linguistic subcultures, I expect to see similar differences across some linguistic subcultures. And
because Honal and Schultz, a similar system, has produced a computational advantage in disfluency
processing, I expect to see similar results. Note that I do not intend to address clinical disfluency
(i.e. stuttering) as part of this work despite the fact that it may also belie a linguistic subculture.

I will use the Switchboard corpus for disfluency experiments. Three dozen conversations in the
corpus have been annotated with the following disfluency events: filled pauses, explicit editing
terms, revisions, discourse markers. The same annotation for these conversations contain a number
of non-disfluency events, specifically: coordinating conjunctions, asides and slash-units: continuers
and (partial) phrases. I will ignore these.

The Switchboard annotation allows for complex disfluencies; however, I will follow [84] in assigning
one of the following labels to each token in the corpus:

- BE: Beginning of a multi-word reparandum
• IE: A token in a multi-word reparandum

• EE: End of a multi-word reparandum

• SE: A single-word reparandum

• O: Any token outside a reparandum

I will use a conditional random field (CRF) for predicting a label for each token in the portion of the Switchboard corpus which contains labels for disfluency. A CRF is a statistical machine learning approach in which the context of the prediction, specifically the labels for previous tokens, may become features in the classification [61]. Each token in the input will be given one label from the set above. Table 4 contains a number of features from recent systems which my system will use in making this classification.

<table>
<thead>
<tr>
<th>Feature Class</th>
<th>Description / Specifics</th>
<th>Source(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean-up</td>
<td>Removal of partial words</td>
<td>[84]</td>
</tr>
<tr>
<td>n-gram</td>
<td>Unigram within window of 5 tokens</td>
<td>[79] [84]</td>
</tr>
<tr>
<td></td>
<td>Bigrams within window of 1 token</td>
<td>[79] [84]</td>
</tr>
<tr>
<td></td>
<td>Trigrams within window of 1 token</td>
<td>[79] [84]</td>
</tr>
<tr>
<td>Part-of-speech</td>
<td>POS of tokens within window of 5 tokens</td>
<td>[79] [84]</td>
</tr>
<tr>
<td></td>
<td>POS of tokens within window of 1 tokens</td>
<td>[79] [84]</td>
</tr>
<tr>
<td>Logic</td>
<td>1-hot: adjacent unigrams are identical within 3-token window?</td>
<td>[79] [84]</td>
</tr>
<tr>
<td></td>
<td>1-hot: adjacent unigram POS are identical within 3-token window?</td>
<td>[79] [84]</td>
</tr>
<tr>
<td>Prosodic</td>
<td>Jitter–perturbation in the pitch period</td>
<td>[66]</td>
</tr>
<tr>
<td></td>
<td>Spectral Tilt–overall slope of spectrum</td>
<td>[66]</td>
</tr>
<tr>
<td></td>
<td>Open Quotient–the ratio of time in which the vocal folds are open</td>
<td>[66]</td>
</tr>
</tbody>
</table>

Table 4: Features to be used in Disfluency Models

I will use F-measure as the performance metric for disfluency systems. The F-measure is defined by two calculations: (1) Precision, $P$–the ratio of correctly-predicted disfluency events to the total number of tokens in the corpus; (2) Recall, $R$–the ratio of correctly-predicted disfluency events to the total number of disfluency events. Combining precision and recall, I will use F-measure, $F$, defined as $F = \frac{2PR}{P+R}$ [84, 79].
Three dozen conversations in the Switchboard corpus contain disfluency annotations. The speakers in these conversations come from potentially different linguistic subcultures. Although all have Standard American English as their first language, the speakers have different genders, a 39 year range in birth years (1930 to 1969) and were tagged as being from six distinct U.S. regions (North Midland, Northern, New York City, Southern, South Midland, Western) as well as a catch-all "Mixed" region. Still, because of the relatively small number of conversations, I will use a leave one speaker out cross validation method for this task: the speech of one speaker will become the test data, while the other speakers’ data will be used to form the training data. I will iterate over speakers so that each speaker becomes a test subject exactly once.

I will create two variants of this disfluency system. One variant, the “unified” one, will use all training data in creating a single disfluency model. The second, “confederated,” variant will contain a disfluency model for each linguistic subculture. See Figure 2 for the architecture of this. In this variant, a disfluency model will be selected for the test speaker in the same way linguistic subcultures are determined. Only the selected model will be used in this variant. I will report the performance of the variants using the F-measure calculation as described above and compare the relative effectiveness of the two approaches.

3.2.3 Sentence Segmentation

The output of ASR is an unannotated sequence of words. This output contains no meta-data information, such as who is speaking, where speech is disfluent (cf. previous section), whether the speaker is asking a question or any other indications to help the reader understand the speakers’ intentions. An example of an unannotated transcript, along with its annotated counterpart, appears in Figure 5.
One important element of meta-data missing from ASR output is sentence boundary information [67], the focus of this section.

For dialogue act detection and other semantic processing tasks such as question answering and machine translation, as shown in [115] and automatic summarization, as shown in [72], sentence segmentation—detection of boundaries between sentences in a stream of words—has been found to be helpful. Where the output of an ASR system is meant for human readers, full punctuation restoration—including commas, question marks and dashes—along with appropriate capitalization and proper formatting of dates and numbers is very useful [43]. This is easily seen in Figure 5, from the television show “Mad About You.”

Sentence segmentation has been found to be a challenging problem in English text as well as other languages, partially because multi-token lexical items (“ice cream”), contractions (“we’ll”), multi-word expressions (“kick the bucket”) and other language phenomena make the word segmentation non-trivial [5].

While prosodic cues may be useful in this task, Grabe’s chapter in [38] shows how these cues may worsen the performance on a punctuation restoration task unless the system considers the
speakers’ linguistic subcultures. In one example, the Southern British English indicate declarative sentences with a falling intonation and common questions with a rising intonation. Speakers in Belfast use a rising intonation for both. Of course, this possible source of confusion may also point to stability within a linguistic subculture, indicating that a possible computational advantage using the approach I propose.

Sentence segmentation may be treated as a classification task in which, at each word boundary, a decision is made about whether the boundary requires a particular punctuation symbol or not, in which case a space will be maintained at the boundary [115]. Like any classification problem, many features have been proven useful in making this decision. A list of the features I will use, inspired by recent systems, appears in Table 5. Note that I omit certain features, such as discourse markers, where they are not reliably available or easily computable from all the source corpora in this input stream.

<table>
<thead>
<tr>
<th>Feature Class</th>
<th>Description / Specifics</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word Sequence</td>
<td>n-gram features</td>
<td>[43]</td>
</tr>
<tr>
<td>Part of Speech</td>
<td>POS language model data generated from a POS tagger</td>
<td>[67]</td>
</tr>
<tr>
<td>Pause</td>
<td>Pause duration between words</td>
<td>[115]</td>
</tr>
<tr>
<td>Prosodic</td>
<td>Pitch, pitch slope, energy</td>
<td>[70]</td>
</tr>
<tr>
<td>Sentence Length</td>
<td>Number of tokens in current utterance</td>
<td>[115]</td>
</tr>
</tbody>
</table>

Table 5: Features to be used in Sentence Segmentation Models

Noting that the majority (70%-90%) of punctuation is composed of commas and sentence endings, Gravano et al. (2009) [43] find that n-gram features have a mild positive effect until n=4 or n=5. However, low values of “n” decrease performance for questions since there is often a larger distance between a “wh-”marker and the location of the question mark; in the case of questions, n=6 or greater is recommended. Their system correctly classifies over 30% F-measure for the most common phenomena, with approximately log-linear gains with respect to the volume of training data. Liu and Shriberg (2007) [65] indicate roughly the same performance when training with prosodic features.
for conversational telephone speech and slightly better performance for a broadcast news corpus.

In their experiments, Liu et al. (2006) [67] also note that interpolation of features across the classes indicated in Table 5 was more effective than pooling features. Specifically, their system merged Word Sequence and Prosodic features first. This merged set was then combined with Part-of-Speech features in the final system.

There have been several proposals for evaluating system performance on this task. (See Liu and Shriberg (2007) [65] for an in-depth discussion of the merits of various metrics.) But because it appears to be sensitive to subtle changes, I will use the NIST error rate, $E$, for comparisons among my experiments. This metric may also be used with a window for in case punctuation within a span of its true location is useful to the reader—for example, if the correct punctuation symbol appears within $n$ spaces of its reference location. The NIST error rate is defined in (7) by the confusion matrix’ $fp$ (false positive), $fn$ (false negative), $tp$ (true positive) and $tn$ (true negative) values. This error rate has the unfortunate possibility of exceeding 100%.

$$E = \frac{fn + fp}{tp + fn}$$  \hspace{1cm} (7)

As with the rest of this proposal, my hypothesis is that punctuation restoration is a task which can see computational advantages from models trained on others in the same linguistic subculture. In order to test this hypothesis, I will implement a sentence segmentation system for this task. This system will have a variable Word Sequence and POS Sequence length as tunable parameters. In addition, where possible I will follow Liu and Shriberg by interpolating features certain features.

Two corpora—Switchboard and ABI (the “sailor diagnostic passage” portion)—contain labels for punctuation restoration. I will therefore divide the corpora into two portions, train and test, so that
no speaker appears in both portions simultaneously. I will also be diligent in balancing with respect
to the sources so that there are roughly equal numbers of speakers from each linguistic subculture
found in the initial step. Using the features in Table 5, I will build a system to predict punctuation
labels in the set \{., ?, \}. Note that the DASH symbol is never used as an annotation for these
corpora, and, because the capitalization is not consistent within the corpora, I will ignore this label
in calculating system performance.

I will build two variants of the systems above. The first will be a “unified” model with all the
training data. The second, as illustrated by Figure 2, will be a “confederated” model. In this second
model, the speakers in the training set will be clustered according to acoustic signal. The speaker
under test will be assigned to one or more of the clusters most closely representing the speaker’s
linguistic subculture. The sentence segmentation system will be trained on speakers from these
closest linguistic subcultures. I will compare results from the system as trained with the global
model and the clustered model using the NIST error rate as defined above.

3.2.4 Dialogue Act Detection

A dialogue act is a speech act for a spoken dialogue system; in other words, it is used to change the
state of a system by informing, commanding, requesting or promising some action [55], for example.
Typical dialogue acts are statements, questions and backchannels [116]. The analysis of dialogue
acts is rooted in the linguistic field of pragmatics, specifically the work of John Austin and John
Searle, and can be used to shine a light on the distinction between a bona fide yes-no question
(“Were you at KC’s bar on the night of July 15?”) from a request for an action (“Is there any salt?”
as a proxy for having a listener pass the salt to the speaker) [104]. In its practical application, this
task identifies the function or goal of an utterance in the context of a longer exchange for a system’s
response to the speech of its users [91, 28].

In performing the task of dialogue act labeling, a system assigns one or more labels or tags from a finite set to a grouping of words, perhaps each sentence-like unit (SU), in an utterance stream. For a dialogue system, these tags correspond to the system’s capabilities [4]. The act of attaching tags has the effect of not only distinguishing statements from questions but also, depending on choice of labels, determining what the users’ expectations of responses from the system should be. As such, it is an important step toward speech understanding and is therefore a useful tool in commercial spoken dialogue systems.

One set of studied examples comes from the LUNA corpus [28], which consists of hardware/software troubleshooting dialogues in Italian. The system has labels for 8 dialogue acts such as greet, offer or apology which correspond to actions the system may perform. In this example, a person performing a greet dialogue act will illicit a welcoming message from the system. (This corpus is omitted from the corpora in Table 1 because the “system” is really a human impostor.)

There are many tag sets for dialogue acts. One such set, Dialog Act markup in Several Layers (DAMSL), described by Core and Allen [22] and adopted by others (for example, [104]), aims to define a task-independent set of a few dozen utterance-level tags. The DAMSL tags occurring at a rate of 5% or greater when applied to the spontaneous utterances in the Switchboard corpus are listed in Table 6, along with an example for each.

Some dialogue systems have shown distributional differences among linguistic subcultures. Shriberg et al. [96] show an insignificant performance boost when using gender as a feature in building a Dialogue Act system against the Switchboard corpus. While their conclusion ultimately rejects any gender-based difference in the task of predicting dialogue acts, subsequent work [35] found Australian English speakers exhibit a quantifiable high rise in the middle of statements (“Australian
questioning intonation” or “uptalk”), which is less prevalent among non-Australian speakers, clearly pointing to prosodic differences between linguistic subcultures in the way questions are asked; the same work finds speaker-specific effects and distributional differences depending on the speaker’s role in a conversation. Other work finds similar phenomena when contrasting speakers from Belfast with speakers from other parts of the United Kingdom [38].

The environment and medium of the language producer also has an effect on the dialogue acts produced. For example, Carpenter and Fujioka [17] find, unsurprisingly, that people in online chats use punctuation less, commit more typos and misspellings, write acronyms and abbreviations more and write emotional indicators and messages which make little sense out of context. In addition, they find the distribution of dialogue acts is different when compared to Switchboard. It is therefore reasonable to conclude that, in addition to the effects we see from the linguistic subcultures of the interlocutors, the medium has some effect on the language.

Dinarelli et al. [28] provide a relatively simple overview of the labelling system which I propose to use in my implementation. The corpus is divided into a series of turns based on the users’ utterances. This is especially important in multi-party interactions (meetings, for example) or any interaction...
in which it is difficult to distinguish speakers’ starting or ending utterances and less important in the corpora on which I will run experiments. Nevertheless, there are useful techniques from this research. For example, a speaker may have more than one utterance per turn, but it may also be useful to overlap turn boundaries across utterances for cases in which speaker turns include restarts, long pauses or environmental or interlocutor interruptions. Once turns have been extracted, the system segments the input into hypothesised utterances [96], removing any disfluencies, non-verbal content or other parts of the conversation deemed irrelevant [6]. Each pruned utterance then receives one or more hypotheses for the dialogue act based on features extracted [34].

One assumption in the task of dialogue act labeling is that a dialogue, \( U \) is composed of \( n \) sequential (albeit possibly overlapping) utterances, \( u_1, u_2, \ldots, u_n \). In determining the dialogue act of an utterance, \( u_j \), some commercial spoken dialogue systems may employ models which take previous utterances (\( u_i \) such that \( i < j \)), labels and other features into account but not data, labels or other features uttered after \( u_j \). In other words, they choose not to find a globally optimal solution. Systems built in this way, which are required to assign a label to speech without future knowledge may be seen as more credible since they can generate real-time responses to speech. Bangalore et al. [6] refer to the set of current and historical features as “static” features; others as “dynamic” features. Despite the possible loss of credibility, there are analyses–Germesin et al. [37], for example–which use dynamic features.

I will conduct two phases of Dialogue Act Detection experiments. The first phase of experiments will distinguish questions from statements on the ABI corpus. The second phase will predict DAMSL labels on the Switchboard corpus.

For the first phase of experiments, I will use the recited “sailor passage” section of the ABI corpus. This portion of the corpus contains a number of questions and statements. Although the questions
Table 7: Features to be used in Dialogue Act Detection task

and statements are not explicitly marked in the transcripts of the speech, they can be inferred from the stimulus, so I will combine the stimulus and transcripts in order to form the dialogue act labels.

In this experiment, I will assign the DAMSL label of \textit{qy} (yes-no-question) to all questions in this passage and \textit{sd} (statement-non-opinion) to all statements. I will retain a portion of the corpus for training, and the remaining will be used for testing. Where possible, I will ensure that there are representatives from each derived linguistic subculture in the training and testing portion. No speaker will appear in training and testing simultaneously.

Taking a machine learning approach and using the features listed in Table 7, I will predict the labels for each turn. Since the passage is the same for all speakers, I will also conduct a version of this experiment without using n-gram features to avoid overfitting.

The second phase of experiments will be conducted on the portion of the Switchboard corpus marked for dialogue acts. Three dozen conversations in the corpus have been annotated with DAMSL labels such as the ones appearing in Table 6. Because of the relatively small size of this
corpus, I will use a Leave One Out cross validation method in this second experiment: the speech of one speaker will be the entirety of the test data, and all other available speech will represent the training data.

I will segment the relevant portion of the Switchboard corpus into a series of utterances as described by the Switchboard corpus and take a machine learning approach to predict these labels for each utterance. The features to be extracted from each utterance appear in Table 7 along with systems which have used these features.

The dialogue act detection system I build will have two variants: a “unified” variant and a “confederated” one. In the unified variant, the system will use the entire training data for performing the ASR task. Each model in the confederated variant will represent a unique linguistic subculture, as discussed in the Linguistic Subculture Detection section and illustrated by Figure 2. I will conduct one experiment for each of these variants and for each of the phases defined above, for a total of four experiments. Speakers being tested will be assigned to one or more linguistic subcultures with the same features used to form the clusters. Once assigned, dialogue act tagging will be performed with only the subset of training data used to form the cluster.

I will compare the effectiveness of these system variants using error rate, following Bangalore et al. [6], with lower scores indicating higher performance.
4 Summary of Results - The ComParE 2015 PC Sub-Challenge

This section describes a submission to the Parkinson’s Condition sub-challenge of the Computational Paralinguistics Challenge (ComParE) 2015. The work on this project does not directly contribute to the topics in this proposal because its focus is on one linguistic subculture; however the work provides some details and of the tools and methods I intend on using.

The ComParE 2015 PC Sub-Challenge was performed jointly with computer science PhD students Guozhen An, Hernisa Kacorri, Min Ma, Ali Raza Syed, with linguistics PhD students Michelle Morales and Rachel Rakov along with Professor Andrew Rosenberg. We submitted a paper to the Interspeech 2015 conference as a result of this effort.

4.1 Background

Parkinson’s disease (PD) is a progressive, neurodegenerative disease affecting millions of people globally: about 1% of people over the age of 65 [78], with 12 to 15 cases per 100,000 people in Europe and the United States. These rates increase with age and change with gender, ethnicity and possibly with socioeconomic factors [41]. Most people affected by PD present difficulties exhibit so-called “TRAP” sensorimotor symptoms: tremors while at rest, rigidity, akinesia (impairment of voluntary motor control), and postural instability [54], i.e. lack of balance. Some of these TRAP symptoms may lead to speech problems or speech dysfunction. In addition to these symptoms, the disease has also been shown to be responsible for a number of neurological and psychological symptoms such as changes in sleep patterns, loss of emotional well being, cognition, visual and spatial deficits, as well as changes in perception [7].

As participants in the Parkinson’s Condition sub-challenge, we were provided with a corpus of 2,562 examples of speech drawn from 100 Spanish L1 speakers from Colombia: 50 control subjects
with no known PD diagnosis and 50 individuals with PD. Each speech sample appears in its own file. Additionally, we were provided with a number of baseline acoustic features drawn from Mel-frequency cepstral coefficient (MFCCs) and “functionals” (their deltas and double-deltas). 1,470 of the speech samples comprised the training set, 630 the development set and 462 the test set.

Each speech sample is also given a label on the Unified Parkinson’s Disease Rating Scale (UPDRS). The UPDRS was introduced in 1987 by Fahn and Elton and still enjoys popularity as an instrument to quantify the longitudinal effects of the disease or therapies on patients [59]. In associating a speech signal to the UPDRS, Orozco-Arroye et al. [78] focus on 5 vowels of Spanish-speaking subjects, looking at a number of noise measures as well as periodicity and stability features such as jitter. Their results support that variability of pitch is a good cue in characterizing vowels uttered by people with PD.

### 4.2 Related Work

Dysarthria, speech disturbances caused by the neurological portion of the disease, has been the focus of most research due to large percentage of PD patients it affects, as high as 90% [110]. I survey a body of research which points to a linguistic subculture of PD spanning language and dialect. This subculture is identifiable mostly by features derived from voice quality and vowel realizations in those with a PD diagnosis.

Scott and Caird [94], for example, find that the main features of speech disorder of PD patients are reduced intensity of voice, a tendency toward overall increased pitch, monotony of speech, and an abnormal rate of speaking. Skodda et al. [100] build on this by focusing on the stability and duration measures applied to syllables /pa/ and /ba/. Operating under the loss of control hypothesis, Walsh and Smith (2011) [110] find higher rates of disfluency for longer or more syntactically complicated
utterances in mild-moderate PD subjects, with implications for patients with more severe symptoms.

Ho et al. [50] look at similar voice qualities predictive of PD, such as harshness, reduced volume, disturbed intonation, fluency inappropriate pausing, syllable repetition and imprecise articulation resulting from lack of motor control. The study classifies speech impairment in 200 patients with PD into five levels of overall impairment severity and describes the corresponding type (voice, articulation, fluency). Most (73.5%) of the subjects in the study demonstrate a gradual deterioration of speech features, almost always involving voice first, before progressing to the prominent voice and articulation pattern, with the latter being the most severely affected.

In more recent work, Skodda et al. (2011) [102] and Skodda et al. (2012) [101] analyze the loss of articulation capacity by using different measures based on 2 formants (F1 and F2) of the voice spectrum: VAI (vowel articulation index) and tVSA (triangular vowel space area).

While much of the research is focused on the articulatory difficulties exhibited by PD patients, Lieberman et al. [64] find significant correlation between an increase in syntactic errors on a language assessment task and voice onset time (VOT) errors on a single-word production task. Errors in comprehension of syntax are measured by the Rhode Island Test of Sentence Comprehension (RITLS).

Our hypothesis is that the PD linguistic subculture may be identifiable by the techniques I put forward in this proposal. The detail of how we test this hypothesis appears in the subsequent section.

4.3 Method

The corpus was divided into three sets (training, development and test) by the organizers. We attempted to recreate the baseline experiments. After this, we augmented the baseline with our
own features. Two of the feature sets (phonotactic and i-vector) are to be re-purposed for this proposal; hence their appearance in this section.

We often find it useful to recreate a baseline result for a challenge so that we are assured that our changes are reproducible and consistent. This experiment, however, used a version of software (OpenSMILE [32], version 2.1) which was unavailable to us because it was in pre-release. Additionally, the configuration for OpenSMILE used by the challengers was not made available. Instead, we used a similar configuration by the same challengers for detecting speech pathology, a similar task, from 2012, three years earlier [93]. We performed four-fold cross-validation on the training set and established this as our new baseline.

To this baseline we added the following features. Each set is described and added to the baseline from above.

- **Syllable-level Features**: We detected the pseudosyllable regions based on the Villing envelope based approach [109] as implemented in AuToBI [90], and derived a number of features for each utterance, including total number of syllables and total duration of syllable regions.

- **Low-level Descriptor (LLD) Features**: We added a number of features from a similar challenge.

- **Formant Features**: Using PRAAT, we extracted F1 and F2 features for each utterance as a proxy for the movement of speech articulators since they are reportedly affected by PD.

- **Phonotactic Features**: I extracted the monophones, biphones and triphones in Czech, English, Hungarian and Russian for each speech sample.

- **Select i-vector Features**: We extracted 10 i-vectors and performed feature selection to
whittle this number down to 5 new features.

- **Stacked LDA Transform Features**: We extracted a higher-dimensional i-vector (200) followed by Probabilistic Latent Discriminant Analysis (PLDA) post-processing and Local Fisher Discriminant Analysis (LFDA) post-processing steps.

We tested feature sets in three ways:

1. Four-fold cross validation on the training set

2. Train-and-test using the $\text{train}=\text{training set}$ and $\text{test}=\text{development set}$

3. Train-and-test using the $\text{train}=\text{training set}$ and $\text{test}=\text{test set}$

### 4.4 Results

For all experiments, we report results as a Spearman correlation coefficient. For the first phase of experiments, four-fold cross validation on the training set, the results appear in Table 8. The table has three sections: the baseline, the effect of adding a single feature set to the baseline and the effect of adding multiple feature sets to the baseline.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Spearman</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (BF)</td>
<td>0.6093</td>
<td>–</td>
</tr>
<tr>
<td>Syllable + BF</td>
<td>0.6104</td>
<td>+0.0011</td>
</tr>
<tr>
<td>LLD + BF</td>
<td>0.6586</td>
<td>+0.0493</td>
</tr>
<tr>
<td>F1 Formant + BF</td>
<td>0.5042</td>
<td>-0.1051</td>
</tr>
<tr>
<td>F2 Formant + BF</td>
<td>0.5040</td>
<td>-0.1053</td>
</tr>
<tr>
<td>Phonotactic + BF</td>
<td>0.6739</td>
<td>+0.0646</td>
</tr>
<tr>
<td>i-vector + BF</td>
<td>0.6104</td>
<td>+0.0011</td>
</tr>
<tr>
<td>Stacked LDA + BF</td>
<td>0.6369</td>
<td>+0.0300</td>
</tr>
<tr>
<td>Syllable, LLD + BF</td>
<td>0.6594</td>
<td>+0.0418</td>
</tr>
<tr>
<td>Select i-vector, Syllable, LLD + BF</td>
<td>0.6607</td>
<td>+0.0501</td>
</tr>
<tr>
<td>Stacked LDA, Select i-vector, Syllable, LLD + BF</td>
<td>0.6937</td>
<td>+0.0844</td>
</tr>
<tr>
<td>All features + BF</td>
<td>0.7088</td>
<td>+0.0995</td>
</tr>
</tbody>
</table>

Table 8: *Training Set: PC Sub-Challenge Cross-Validated Training Results*
While the results show that the two formant features (F1 Formant and F2 Formant) have a detrimental effect on the baseline, all other feature sets have a positive effect. The phonotactic feature set makes the largest single positive contribution, with the LLD Features and Stacked LDA features contributing with the same order of magnitude. We excluded the two formant feature sets from further consideration.

However, the results for the train-and-test using the development set showed something different. These results appear in Table 9.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Spearman</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (BF)</td>
<td>0.5076</td>
<td></td>
</tr>
<tr>
<td>Syllable + BF</td>
<td>0.5083</td>
<td>+0.0007</td>
</tr>
<tr>
<td>LLD + BF</td>
<td>0.5139</td>
<td>+0.0063</td>
</tr>
<tr>
<td>Phonotactic + BF</td>
<td>0.4808</td>
<td>-0.0268</td>
</tr>
<tr>
<td>i-vector + BF</td>
<td>0.5086</td>
<td>+0.0010</td>
</tr>
<tr>
<td>Stacked LDA + BF</td>
<td>0.5207</td>
<td>+0.0131</td>
</tr>
<tr>
<td>Syllable, LLD + BF</td>
<td>0.5140</td>
<td>+0.0064</td>
</tr>
<tr>
<td>i-vector, Syllable, LLD + BF</td>
<td>0.5150</td>
<td>+0.0074</td>
</tr>
<tr>
<td>Stacked LDA + i-vector, Syllable, LLD + BF</td>
<td>0.5258</td>
<td>+0.0182</td>
</tr>
<tr>
<td>All features + BF</td>
<td>0.5051</td>
<td>-0.0025</td>
</tr>
</tbody>
</table>

Table 9: Training Set: PC Sub-Challenge Development Set Results

With the exception of the phonotactic feature set, the results are more or less consistent with the four-fold cross-validation experiments. The LLD features yielded an increase in system performance, but the effect was slightly weaker. We attributed the surprising decrease from the phonotactic features to the differences in the experimental methodology. The four-fold cross validation experiments were conducted so that it was possible to overlap speakers across folds. This was not the case in this second set of experiments. We therefore inferred that the phonotactic feature set was able to identify individual speakers.

We had a limited number of trials for the final set of experiments, so we looked largely at combinations of systems to determine whether the above results held. Upon inspection of Table 10, we found they largely did.
As part of a limited-time challenge, this effort largely becomes one a speed-accuracy trade-off. This led us to place too large a burden on our machine learning tools. For example, while Skodda et al. (2011) [100] report diagnostic success when focusing on a limited number of syllables, we found that some of the speech samples did not contain the syllables in question. (In one example, the speech sample *Mi casa tiene tres cuartos* does not contain any instances of the /pa/ or /ba/ syllables.) Since we were not able to extract certain signals consistently, we made a decision to extract all syllables to see whether there was a correspondence between PD and the observed syllables. Our overgeneralized approach was not successful for the Formant features, so it may have been useful to cull these to the narrower sets shown useful by the work of others.

In the same vein, the two i-vector approaches were only useful after they were manipulated with our two post-processing methods: feature selection and Stacked LDA Transforms. Not described in the submission was the addition of 10 and 20 raw i-vector features. These produced no increase in system performance, sometimes having a detrimental effect.

Clearly, the phonotactic set of features carried a signal useful in one set of experiments. Equally clear is how the feature set did not generalise to the set of all speakers. Anecdotal evidence pointed to the possibility of adjacent repetition of vowel sounds as an indicator of severity in PD. However, we extracted all phones, not limiting ourselves to vowel sounds. It is possible that the phonotactic feature set would have benefited from culling or post-processing, similar to what was done for the
i-vector sets.

In relating this pilot to the general case of linguistic subcultures, I find that more careful processing of phonotactic and i-vector features is required to generalize the feature set to generate a useful signal on which my proposed approaches may be performed. Specifically, I believe culling phonotactic features, numbering as high as in the hundreds of thousands, to a smaller set of exemplars would not only have more descriptive power but would also create a more tractable computation. I also believe that i-vector features should be made subject to several types of post-processing as suggested by Shum et al. [97], for example.
5 Timeline

<table>
<thead>
<tr>
<th>Date</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 2015</td>
<td>Build PRLM-based Linguistic Subculture system</td>
</tr>
<tr>
<td></td>
<td>Build i-vector-based Linguistic Subculture system</td>
</tr>
<tr>
<td></td>
<td>Test Linguistic Subculture systems on Fisher corpus</td>
</tr>
<tr>
<td>August, 2015</td>
<td>Determine and report discourse markers for each Linguistic Subculture</td>
</tr>
<tr>
<td>September 2015</td>
<td>Build Automatic Speech Recognition (ASR) system</td>
</tr>
<tr>
<td></td>
<td>Create subculture models for ASR system</td>
</tr>
<tr>
<td></td>
<td>Create global model for ASR system</td>
</tr>
<tr>
<td></td>
<td>Test ASR system on ABI, IViE, Fisher and Switchboard</td>
</tr>
<tr>
<td>October 2015</td>
<td>Assemble Linguistic Subculture and ASR results</td>
</tr>
<tr>
<td></td>
<td>Write paper and seek publication venue, if appropriate</td>
</tr>
<tr>
<td>November 2015</td>
<td>Build disfluency system</td>
</tr>
<tr>
<td></td>
<td>Train disfluency system on clustered model</td>
</tr>
<tr>
<td></td>
<td>Train disfluency system on global model</td>
</tr>
<tr>
<td></td>
<td>Test disfluency systems on Switchboard corpus</td>
</tr>
<tr>
<td>December 2015</td>
<td>Build sentence segmentation system</td>
</tr>
<tr>
<td></td>
<td>Train sentence segmentation system on clustered model</td>
</tr>
<tr>
<td></td>
<td>Train sentence segmentation system on global model</td>
</tr>
<tr>
<td></td>
<td>Test sentence segmentation on ABI and Switchboard corpora</td>
</tr>
<tr>
<td>January 2016</td>
<td>Assemble disfluency and sentence segmentation results</td>
</tr>
<tr>
<td></td>
<td>Write paper on results; seek publication venue if appropriate</td>
</tr>
<tr>
<td>February 2016</td>
<td>Build dialogue act system</td>
</tr>
<tr>
<td></td>
<td>Train dialogue act system on clustered model</td>
</tr>
<tr>
<td></td>
<td>Train dialogue act system on global model</td>
</tr>
<tr>
<td></td>
<td>Test dialogue act system on ABI corpus</td>
</tr>
<tr>
<td></td>
<td>Test dialogue act system on Switchboard corpus</td>
</tr>
<tr>
<td>March 2016</td>
<td>Assemble dialogue act results</td>
</tr>
<tr>
<td></td>
<td>Write paper on results; seek publication venue if appropriate</td>
</tr>
<tr>
<td>April 2016</td>
<td>Complete written thesis</td>
</tr>
<tr>
<td>May 2016</td>
<td>Defend thesis</td>
</tr>
</tbody>
</table>

Table 11: Schedule of Tasks

Table 11 describes the timeline to complete the remainder of this thesis. I will keep several
germane conferences and journals in mind, publishing as much as possible during this time.
References


[71] Paul Meier. IDEA International Dialects of English Archive.


[113] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying A. Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, and others. The HTK book (for HTK version 3.4). 2006.


