Using open source speech recognition software without an American accent.

The field

Julius: C, strange BSDish license, Japanese

HTK: (HMM (Hidden Markov Model) Toolkit)
C, non-commercial license

Kaldi: C++ library, Apache 2.0 (F.U. HTK)

CMU Sphinx Family: C, Java, BSD-ish

Smaller possibly dormant projects

Shout: C++, GPL

RWTH ASR: C++, non-commercial, German

Iatros: C, GPL3, Spanish

Carnegie Mellon University Sphinx

Sphinx 3, 4, and PocketSphinx share tools.

Sphinx development model

PhD driven:

1. Have a moderately good idea
2. Write code that “proves” your idea
3. Submit thesis
4. $$$*

* $$$ refers to a job at Nuance, Apple, Microsoft, Google, IBM,...

The “Coral Reef” development model.

What Americans do:

```
sudo aptitude install       \
  pocketsphinx-hmm-wsj1     \
  pocketsphinx-lm-wsj       \
  python-pocketsphinx       \
  # or gstreamer0.10-pocketsphinx
```

```
from pocketsphinx import Decoder

HMM = "/usr/share/pocketsphinx/model/hmm/wsj1/"
LM = "/usr/share/pocketsphinx/model/"\
     "lm/wsj/wlist5o.3e-7.vp.tg.lm.DMP"
DICT = "/usr/share/pocketsphinx/model/"\
     "lm/wsj/wlist5o.dic"

decoder = Decoder(hmm=HMM, lm=LM, dict=DICT)

fh = open("speech.wav")
fh.seek(44) # skip the WAV header
decoder.decode_raw(fh)
print decoder.get_hyp() # short for hypothesis
```

A good reference is David Huggins-Daines' live-coding PyCon2010 lightning talk.

Everything goes wrong and he still finishes in under 4 minutes.

The configuration lines:

```
HMM = ".../hmm/wsj1/"
LM = ".../lm/wlist5o.3e-7.vp.tg.lm.DMP"
DICT = ".../lm/wsj/wlist5o.dic"
```

HMM Acoustic model *(Hidden Markov Model)*
LM Language model
DICT Pronunciation dictionary

The *Acoustic model* matches sounds to phoneme probabilities.

It looks at the flow, not each sound in isolation.

The *Language model* rates the probability of a sequence of words.

✗ “Their is moor then won weight hoodoo it”
✓ “There is more than one way to do it”

The acoustic and language models talk to each other through the *dictionary*.

The dictionary maps phoneme sequences to words.

```
PERKY   P ER K IY
PERL    P ER L
PERLA   P ER L AH

WIND            W AY N D
WIND(2)         W IH N D
```

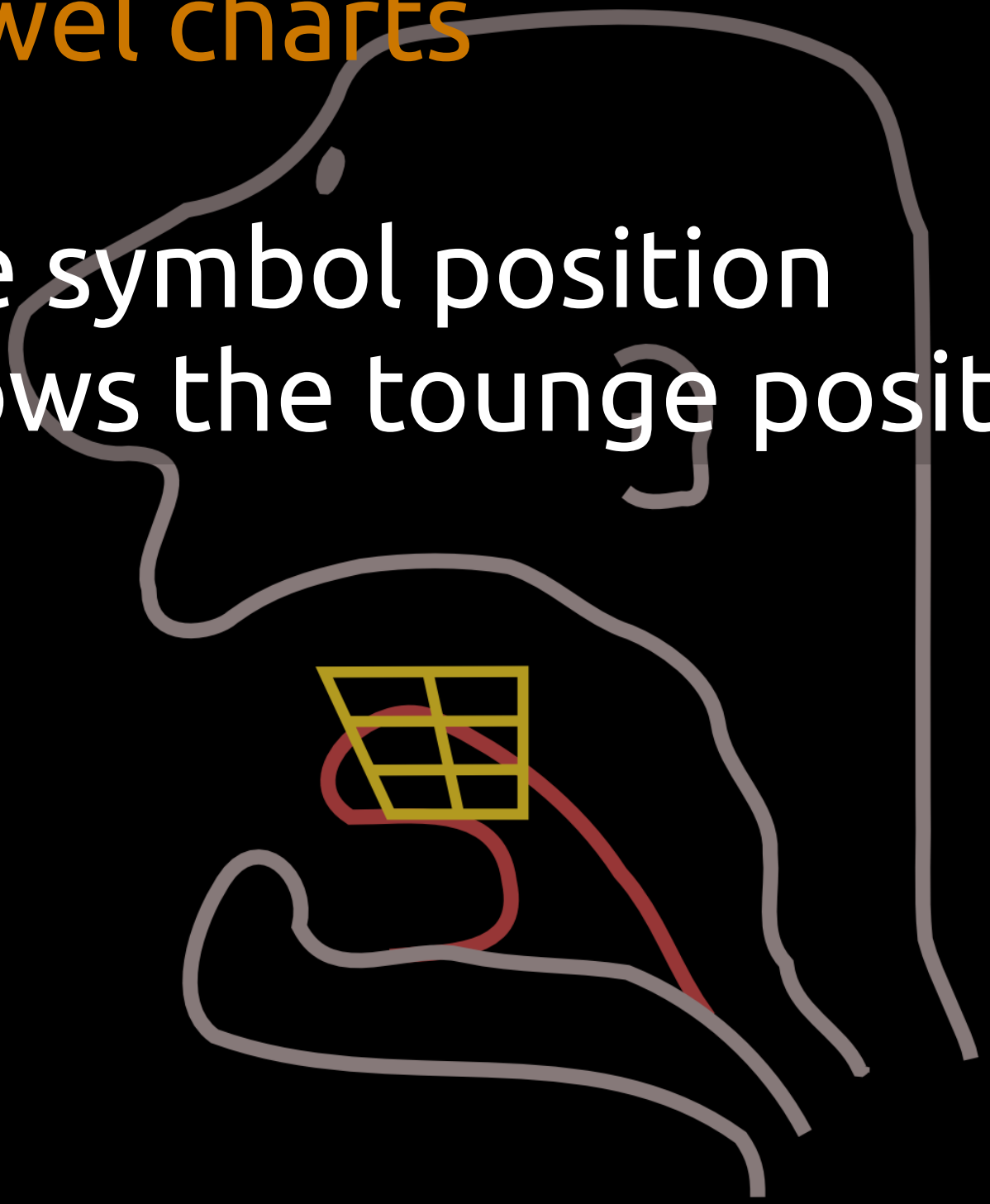
Extracts from the 130,000 words, 39 phoneme CMU dictionary of General American English.

The dictionary is a blunt instrument.

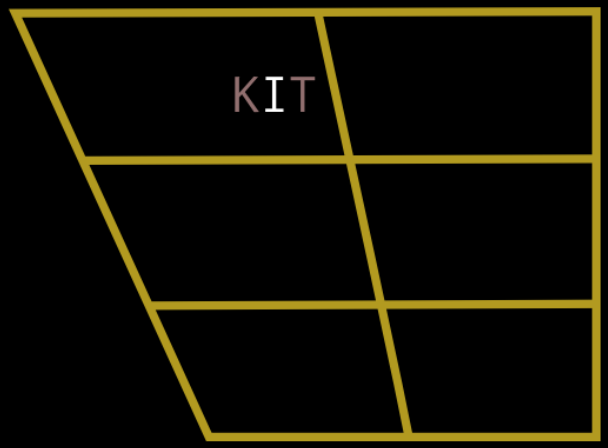
The sounds of words adapt to the context.

Vowel charts

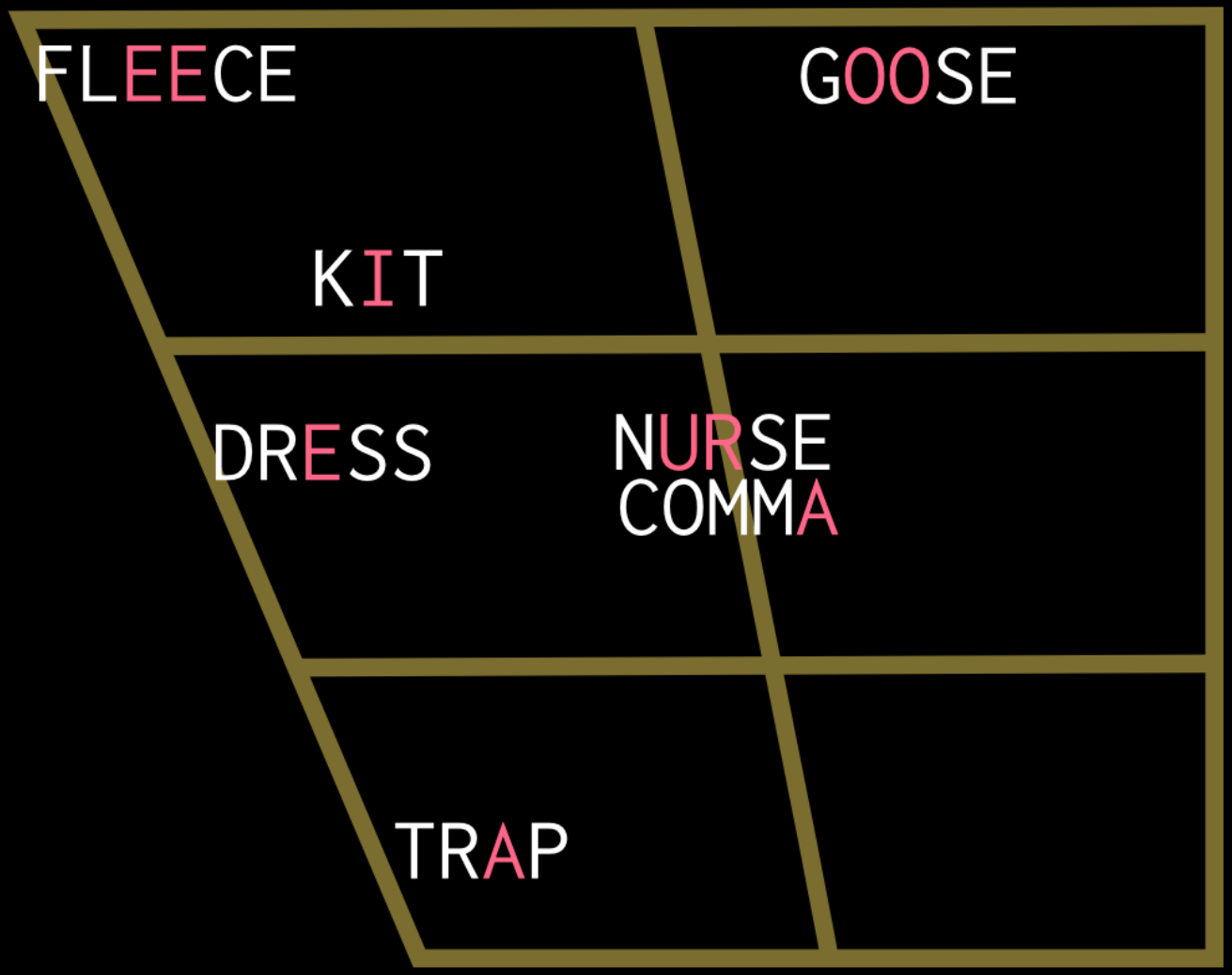
The symbol position shows the tounge position



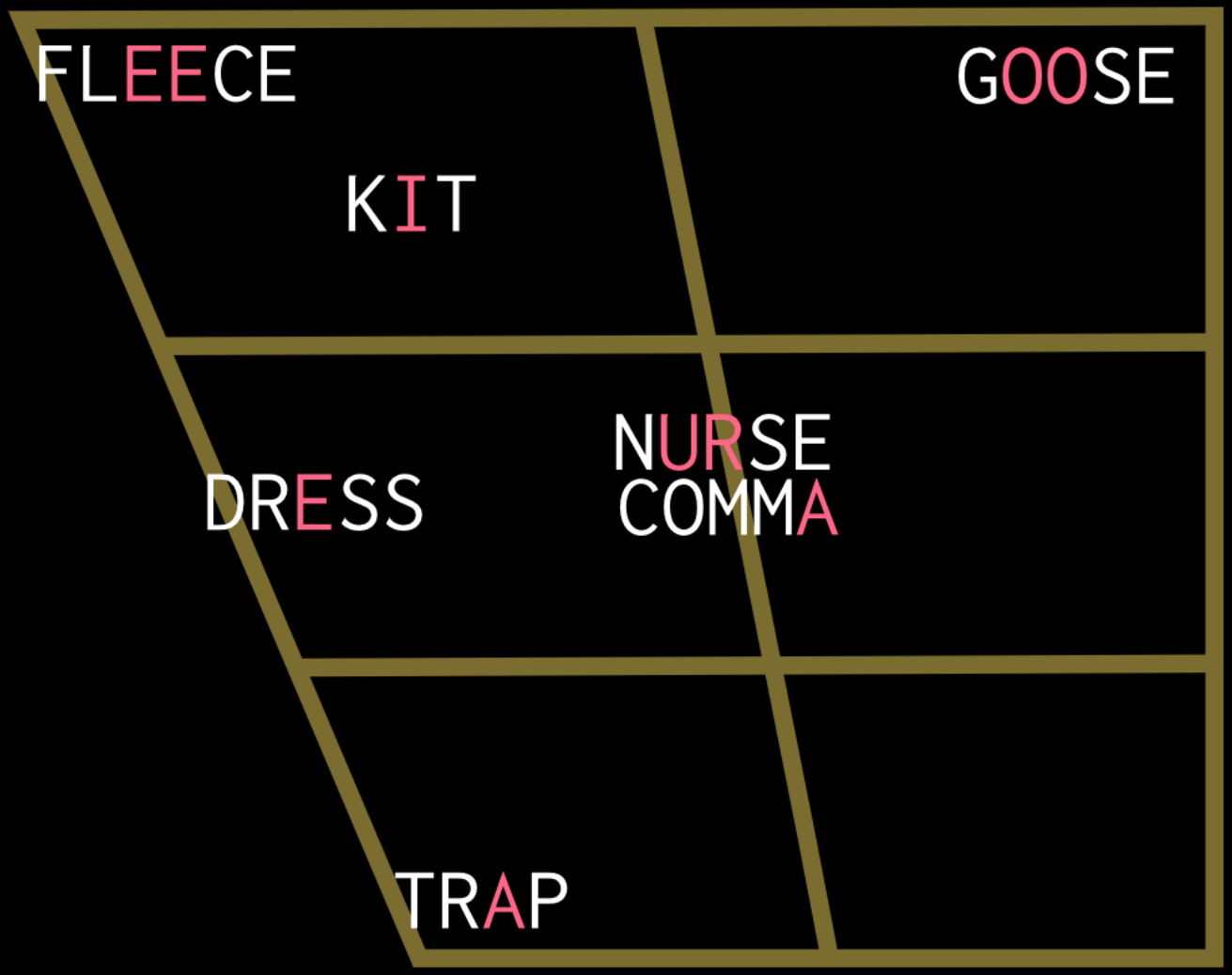
KIT represents a *lexical set*
A bit like IPA's phonemic /ɪ/
(but *not* IPA phonetic [ɪ])
fish and *chips* use the KIT vowel



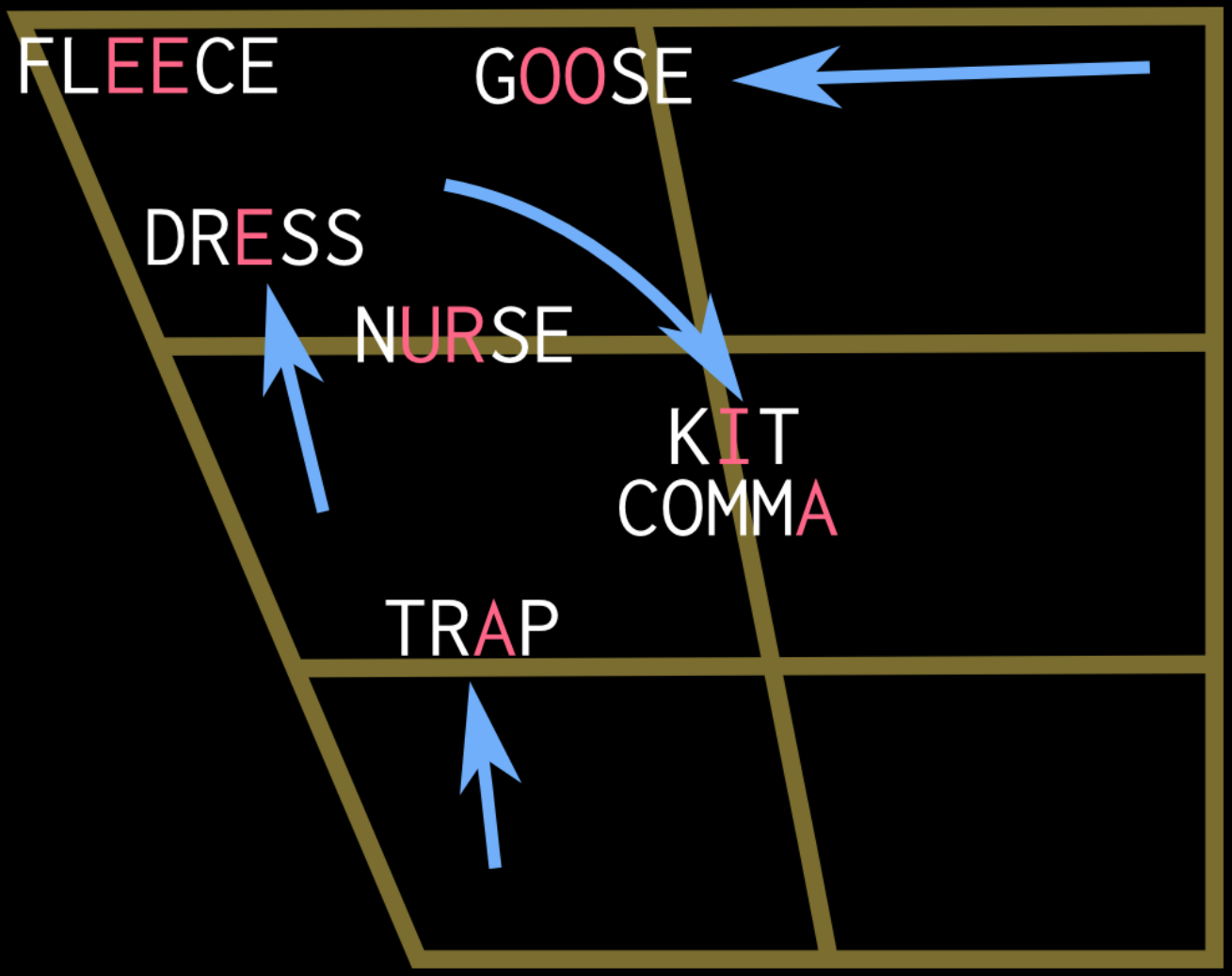
A selection of General American vowels



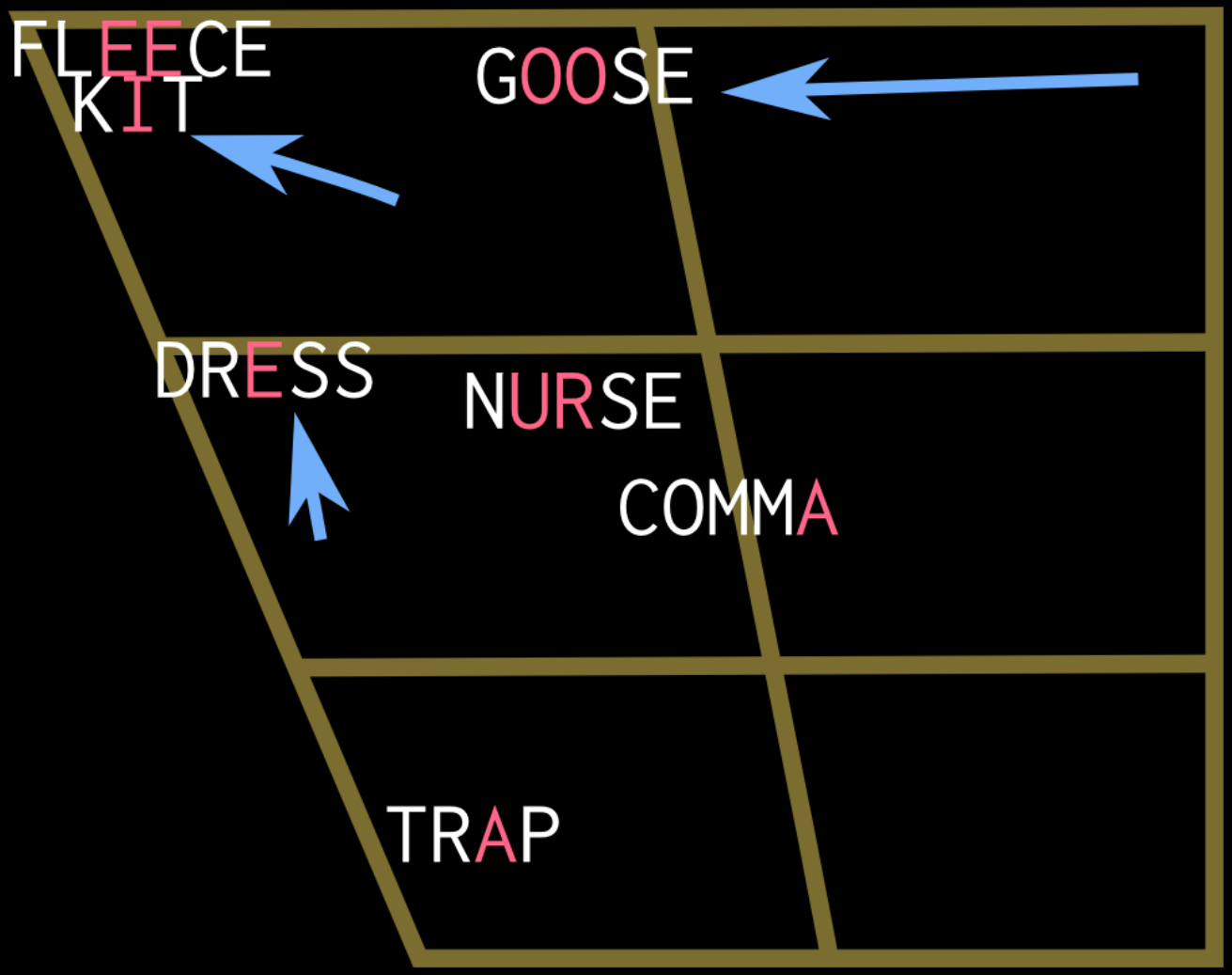
The same vowels in RP (high class British)



New Zealand English



Australian English



General American English

Father and *bother* rhyme
(but *farther* doesn't).

Caught sounds like *cot*, not *court*.

Marry and *merry* are indistinguishable.

Pants rhymes with *Glance*, *Hurry* with *Furry*.

Panda and *pander* sound different
(American English is rhotic).

New Zealand English

bear and *beer* are ~~becoming~~ indistinguishable

The vowel in *fish* is the vowel in *the*.

Otherwise it makes similar distinctions to RP, but not sounds.

How do we get a New Zealand English model?

For the acoustic model there are two methods:

1. Adapt an existing one

2. Create a new one

Existing language models *might* be OK.

Adapting an acoustic model

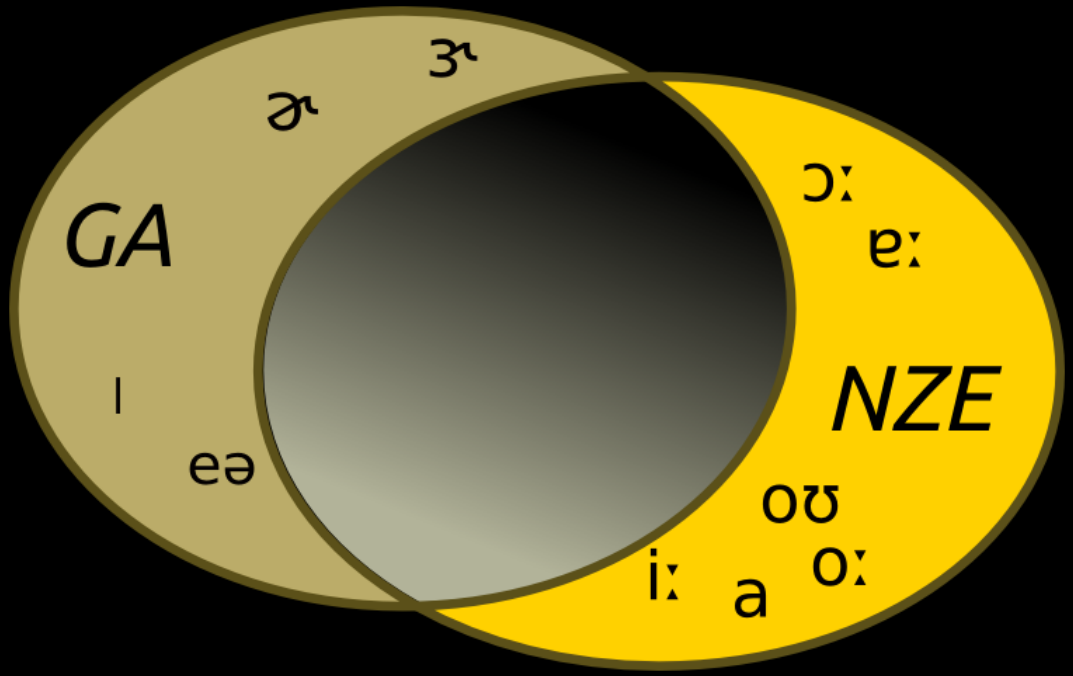
Read http://cmusphinx.sf.net/wiki/tutorialadapt

Requires a “small” corpus of transcribed speech.

The *phoneme set* is fixed.

The CMU dictionary uses 39 phonemes.

New Zealand English uses about 44
... but not all of CMU's 39.



An adaptation using the CMU dictionary

- can learn mergers (at a cost)
- can't make new distinctions

Effectively down to 35 or so phonemes

Useless duplication:

```
BEAR    B EH R #B EH
BEER    B IH R #B EH
```

wrongly conjoined:

```
FATHER  F AA DH ER
BOTHER  B AA DH ER # B OH DH AX
```

Just wrong:

```
TUNE    T UW N # T Y UW N
```

Squeezing NZE into the CMU phoneset

A new dictionary could remap vowels:

```
F AA DH IH
B AO DH IH  # ?? AO is the PAW vowel
            # IH is the KIT vowel.
```

There is no good way to do it.

NZE pronouncing dictionaries

There are none.

British English pronouncing dictionaries

- CELEX — proprietary
- COALD, beep, Oxford 710 — non-commercial
- UNISYN — non-commercial, multi-accent

Generating a dictionary via grapheme-to-phoneme systems

Espeak, Festival, FreeTTS read English

Espeak specialises in British English

Espeak speaks IPA

```
$ espeak -q --ipa 'father bother'
 fˈɑːðə bˈɒðə
```

Espeak is rule based, so it
is never stuck for an answer:

```
$ espeak -q --ipa 'Wellington Perlmongers'
 wˈɛlɪŋtən pˈɜːlmʌŋgəz
(“pˈɜːlmʌŋəz” might be better)
```

Transcribed speech

Sphinx likes to be trained with short (5 to 30 second) snippets of speech.

There is a control file a bit like this:

```
COLENSO SAID THAT HE COULDN'T ++UM++ PERSUADE (DGI038-auto-series-2-033)
OUR HEARTS AND OUR BACKS TO HOIST ANCHOR (LunaTick-20080319-ill-mfc-a0411)
SAVINGS IN THAT YEAR WERE NAUGHT POINT TWO (questions-20120802-wav-0007)
```

and audio file with matching names.

Accurate transcribing is slow

Slower than recording.

Cutting up long transcriptions is slower than recording but faster than transcribing.

Sources of transcribed speech

- Voxforge
- Wellington Corpus
- Canterbury Corpus
- Hansard
- TV and radio

http://voxforge.org aims
to create GPL speech models.

It has about an hour of NZE
from about 12 volunteers
reading prompts

Wellington Corpus of Spoken English

A million spoken words (half natural conversation)

Recorded and transcribed in the 1990s by VUW linguistics students

Imprecise timing information

On-site access only.

Canterbury Corpus

About 150 hours of natural conversation

Transcribed in 2000s by Canterbury students with software help

Good timing information

On-site access only.

Both Wellington and Canterbury linguistics people are keen to help

Canterbury have resources, namely Robert Fromont

VUW are skint.

Hansard

OK, but prone to acronym expansion, missed interjections, and correction.

MP: “to the minister: which assets identified by CERA...”
Hansard: “to the Minister for Canterbury Earthquake Recovery: Which of the assets identified by the Canterbury Earthquake Recovery Authority”

MP: “twelve hundred”
Hansard: “1200”
espeak: “one thousand two hundred”

Radio

The news on the RNZ website is *almost* the same as the news on the radio.

TV

Scripts and teletext subtitles.

Adaptation recap

- Requires a bit of speech
- Forces NZE into a foreign phoneme set
- Is quick enough to play around with

Creating an acoustic model

A general model requires at least 50 hours of speech

The dictionary and phoneme set can be tailored to suit

http://cmusphinx.sf.net/wiki/tutorialam

The instructions are long and complicated

but that is sort of moot

because I don't have 50 hours of speech

It seems the training software is:

- Good at uncovering transcription errors
- Less good at providing error messages

Tidying up the data is a huge job

Progress so far

I am focusing on adapting existing models

Related ephemera at
http://github.com/douglasbagnall/nze-vox
(Makefile, python and shell scripts, wiki pages).

My aim is automated testing of
dictionaries and other parameters

Robert Fromont is working on a model from scratch, using:

- the Canterbury Corpus
- the non-free CELEX dictionary

The problems seem to be with the data.

It *should* eventually be simple to replace CELEX with a free dictionary.

Why I am interested

I work as an artist.

I am making an artwork that eavesdrops and produces video relevant to what people are talking about.

I want it to work.