UNIVERSITY OF CAMBRIDGE

Low Resource Speech Processing

Mark J.F. Gales

February 2020

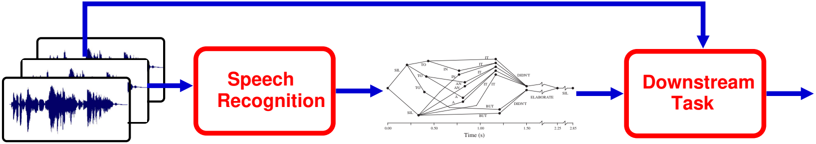
Introduction

- Over 7000 languages spoken around the world
  - over 90% used by less than 100,000 people
  - not viable to develop bespoke systems/collect data
- Restrict languages to those with a written form

Introduction

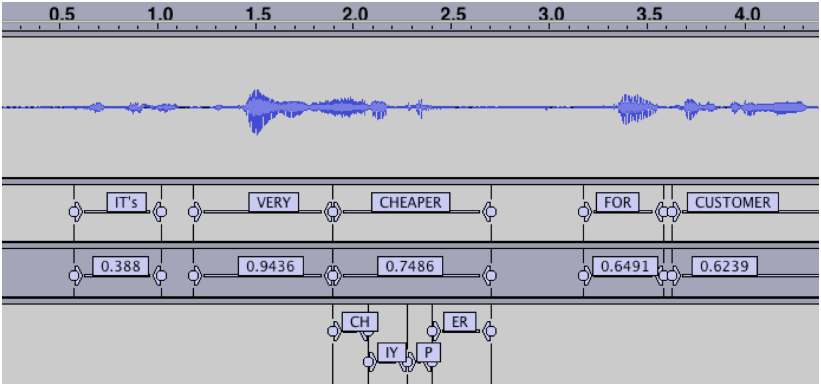
- Over 7000 languages spoken around the world
  - over 90% used by less than 100,000 people
  - not viable to develop bespoke systems/collect data
- Restrict languages to those with a written form

Spoken Language Processing Framework

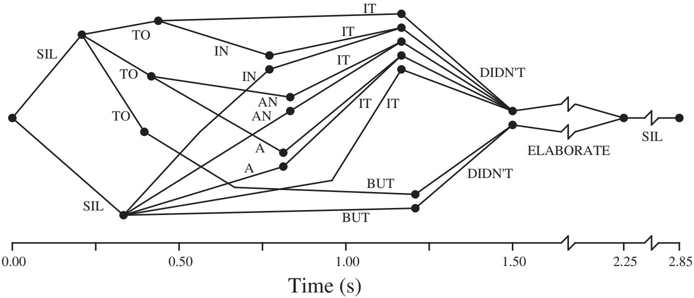


- First stage of speech processing usually speech recognition
  - yields (at least) word-sequences for downstream task
  - output may be significantly richer (lattices)
- Can be viewed as adding structure to the audio

Automatic Speech Recognition: 1-Best



Automatic Speech Recognition: Lattices



- A lattice, $\mathcal{L}$, comprises:
  - nodes (sometimes called state): associated with time stamps
  - arcs: have labels and scores (not shown)

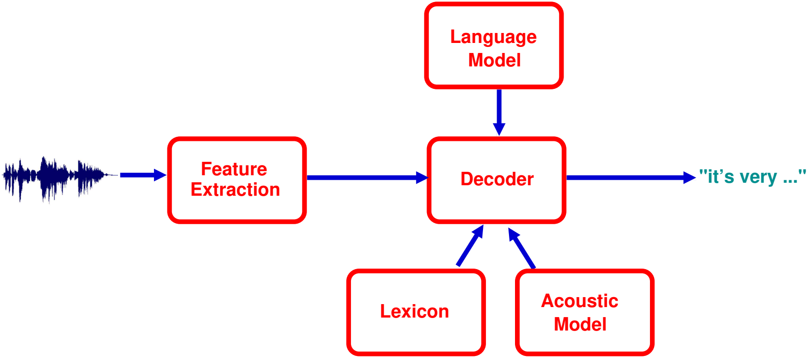
Low Resource Speech Processing

- Low-resource can refer to various elements:
  - acoustic model training data
  - audio transcriptions
  - lexicon (phonetic lexicon)
  - language model training data
  - language processing resources (parsers/PoS tagger)
  - downstream task training data
- Systems often have high error rates (at all stages)
  - need to mitigate impact of errors on downstream stages

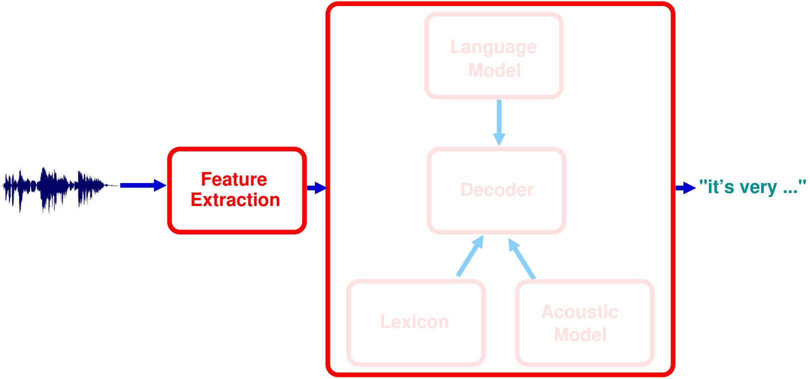
Low Resource Speech Processing

- Low-resource can refer to various elements:
  - acoustic model training data
  - audio transcriptions
  - lexicon (phonetic lexicon)
  - language model training data
  - language processing resources (parsers/PoS tagger)
  - downstream task training data
- Systems often have high error rates (at all stages)
  - need to mitigate impact of errors on downstream stages

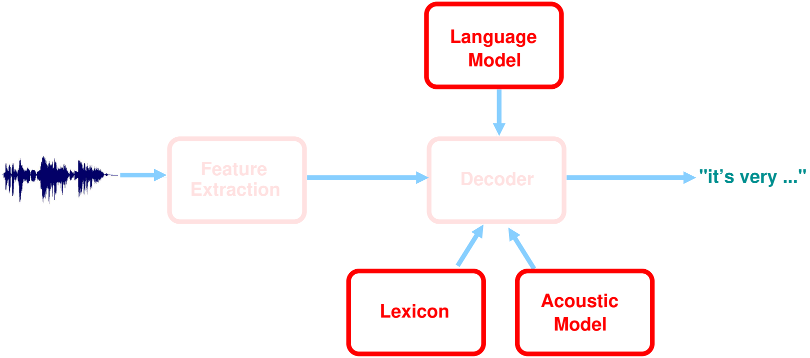
“Traditional” Speech Recognition Framework



“End-to-End” Speech Recognition Framework



Speech Recognition Components



"Traditional" Speech Recognition [6]

- For input $\boldsymbol{x}_{1:T}$ output the word sequence:

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}} \left\{ P(\boldsymbol{w}|\boldsymbol{x}_{1:T}) \right\} = \arg\max_{\boldsymbol{w}} \left\{ P(\boldsymbol{w}) p(\boldsymbol{x}_{1:T}|\boldsymbol{w}) \right\}$$

$$= \arg\max_{\boldsymbol{w}} \left\{ P(\boldsymbol{w}) \sum_{\boldsymbol{\theta}_{1:T} \in \boldsymbol{\Theta}_{\boldsymbol{w}}} p(\boldsymbol{x}_{1:T}, \boldsymbol{\theta}_{1:T}) \right\}$$

- The components are
  - language model: $P(\boldsymbol{w})$
  - lexicon: valid set of states for word sequence $\boldsymbol{w}$, $\boldsymbol{\Theta}_{\boldsymbol{w}}$
  - acoustic model: $p(\boldsymbol{x}_{1:T}, \boldsymbol{\theta}_{1:T})$

Talk Overview

- Language Model
- Lexicon
- Acoustic Model
- Downstream Speech Procesing Tasks
    - key-word and phrase spotting
    - cross-language information retrieval

Language Model

Language Modelling

- Component of many speech/language applications

"it's very cheaper for customer"

Given a sequence, what is the next word

- Statistical approaches have dominated for many years:

$$P(w_i|w_1, \ldots, w_{i-1}) = P(w_i|\mathbf{w}_{1:i-1})$$

- Sometimes need the probability of the words sequence

$$P(\mathbf{w}_{1:L}) = P(w_1)\prod_{i=2}^{L} P(w_i|\mathbf{w}_{1:i-1})$$

Language Modelling

- Component of many speech/language applications

"it's very cheaper for customer"

Given a sequence, what is the next word

- Statistical approaches have dominated for many years:

$$P(w_i|w_1, \ldots, w_{i-1}) = P(w_i|\boldsymbol{w}_{1:i-1})$$

- Sometimes need the probability of the words sequence

$$P(\boldsymbol{w}_{1:L}) = P(w_1)\prod_{i=2}^{L} P(w_i|\boldsymbol{w}_{1:i-1})$$

- Component of many speech/language applications

"it's very cheaper for customer"

Given a sequence, what is the next word

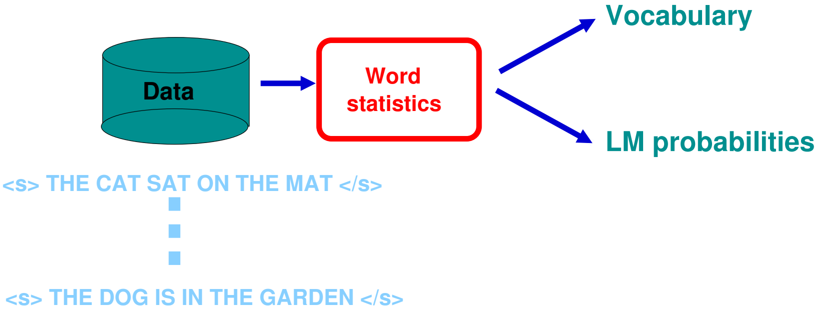
- Statistical approaches have dominated for many years:

$$P(w_i|w_1,\ldots,w_{i-1}) = P(w_i|\boldsymbol{w}_{1:i-1})$$

- Sometimes need the probability of the words sequence

$$P(\boldsymbol{w}_{1:L}) = P(w_1)\prod_{i=2}^{L} P(w_i|\boldsymbol{w}_{1:i-1})$$

Language Model Training Data

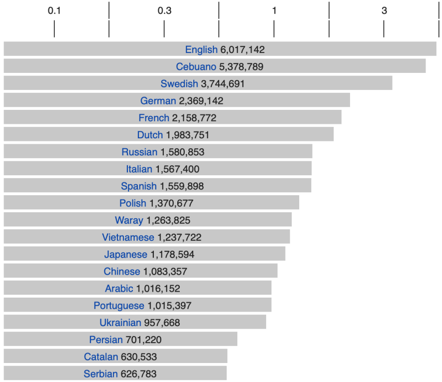


- Text data essential for current ASR systems
  - determines the possible vocabulary for the systems
    impacts Out Of Vocabulary (OOV) rate
  - quantity of data determines accuracy (and order) of LMs

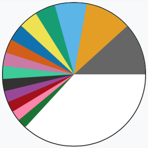
The World Wide Web

Web Training Data: Wikipedia

Logarithmic graph of the 20 largest language editions of Wikipedia
(as of 19 February 2020)[147]
(millions of articles)



The unit for the numbers in bars is articles.

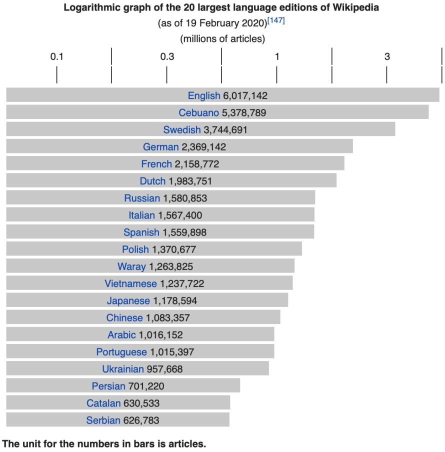


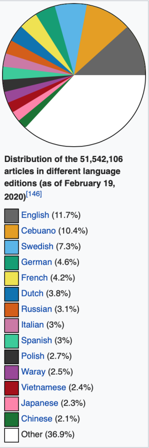
Distribution of the 51,542,106 articles in different language editions (as of February 19, 2020)[146]

- English (11.7%)
- Cebuano (10.4%)
- Swedish (7.3%)
- German (4.6%)
- French (4.2%)
- Dutch (3.8%)
- Russian (3.1%)
- Italian (3%)
- Spanish (3%)
- Polish (2.7%)
- Waray (2.5%)
- Vietnamese (2.4%)
- Japanese (2.3%)
- Chinese (2.1%)
- Other (36.9%)

Can we make use of web-data for language model training?

Web Training Data: Wikipedia





Can we make use of web-data for language model training?

Nature of Web Text Data [7, 1, 10]

- Most text on the data “written” not speech transcripts
  - significant mismatch with conversational form
  - closer match to broadcast news
  - Wikipedia not a perfect match!
- A number of issues need to be considered
  - sources of data to use
  - ensure match to target language (language identification)
  - select data that matches target domain
  - tidying data
- Build language model source components/interpolate

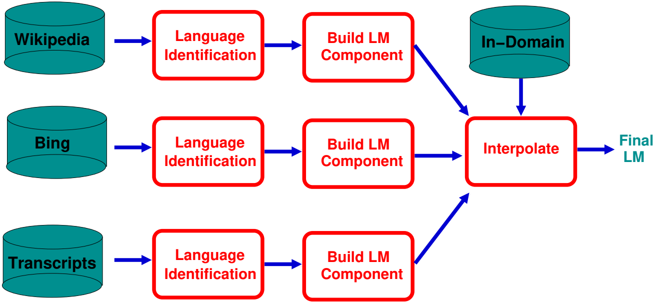
Nature of Web Text Data [7, 1, 10]

- Most text on the data "written" not speech transcripts
  - significant mismatch with conversational form
  - closer match to broadcast news
  - Wikipedia not a perfect match!
- A number of issues need to be considered
  - sources of data to use
  - ensure match to target language (language identification)
  - select data that matches target domain
  - tidying data
- Build language model source components/interpolate

Nature of Web Text Data [7, 1, 10]

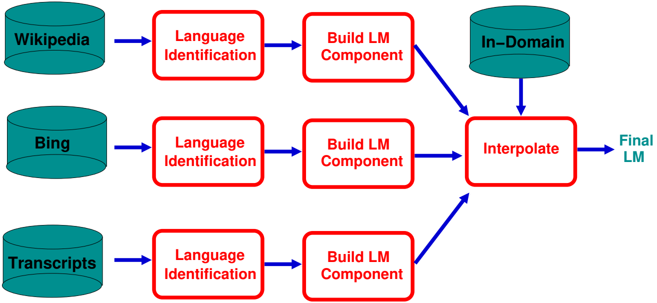
- Most text on the data “written” not speech transcripts
  - significant mismatch with conversational form
  - closer match to broadcast news
  - Wikipedia not a perfect match!
- A number of issues need to be considered
  - sources of data to use
  - ensure match to target language (language identification)
  - select data that matches target domain
  - tidying data
- Build language model source components/interpolate

Language Model Interpolation [13]



- Using limited held-out data to compute weights
  - weights will indicate how source matches domain
    also influenced by data quantity
- Can use untranscribed audio data ...

Language Model Interpolation [13]



- Using limited held-out data to compute weights
  - weights will indicate how source matches domain also influenced by data quantity
- Can use untranscribed audio data ...

Low Resource Web Scraping [10]

- Sources can be split into two broad classes:
- General search strategies: use Bing/Google to search web
  - extract search terms from limited available data
  - generates large quantities of data
  - language filtering becomes important (Mandarin/Cantonese, Kazakh/Russian)
- Directed Searches: use known language sources
  - examples: Wikipedia, Blogs, News Forums, Twitter, TED talks

Low Resource Web Scraping [10]

- Sources can be split into two broad classes:
- General search strategies: use Bing/Google to search web
  - extract search terms from limited available data
  - generates large quantities of data
  - language filtering becomes important (Mandarin/Cantonese, Kazakh/Russian)
- Directed Searches: use known language sources
  - examples: Wikipedia, Blogs, News Forums, Twitter, TED talks

Document Filtering

- Filtering approaches aim to match target domain
  - build language model using limited available data
  - filter documents using perplexity and OOV rates
- Perplexity: average number of following words
  - using the in-domain language model
  - compute perplexity of the document word sequence $\boldsymbol{w}_{1:L}$

$$\mathrm{PPL}(\boldsymbol{w}_{1:L}) = \exp\left(-\frac{1}{L}\sum_{i=1}^{L}\log(P(w_i|\boldsymbol{w}_{1:i-1}))\right)$$

- OOV rate: percentage of words missing from LM vocabulary
  - simply computed for the document $\boldsymbol{w}_{1:L}$

Web Data Statistics

Language	LM	Data (K) words	Data (K) vocab	FLP Weight	OOV (%) ASR	OOV (%) KWS
Pashto	FLP	535	14.4	—	1.96	11.38
	Web	104624	376.3	0.981	0.68	3.05
Amharic	FLP	388	35.0	—	9.80	15.42
	Web	13911	223.6	0.976	5.67	9.16
Dholuo	FLP	467	17.5	—	3.26	12.17
	Web	1217	18.8	0.998	3.01	10.73

- FLP is the (matched) in-domain CTS data
- Quantity of web-data available highly dependent on language
  - interpolation weight ("match") web data: range ≈ 0.1 to 0.001
  - large impact on OOV rates

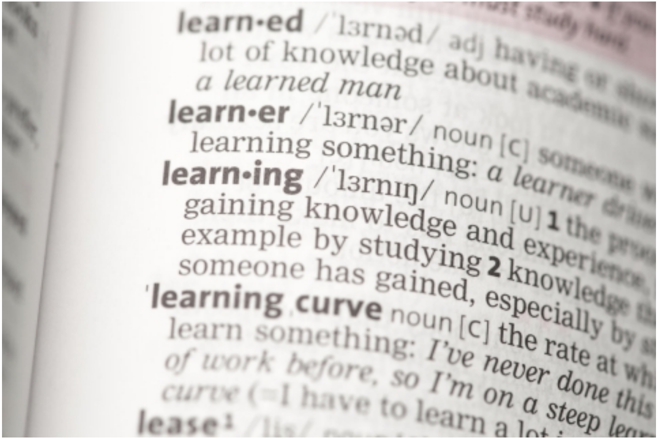
Impact of Web Data: Lithuanian

Data	LM	OOV (%)	WER (%)
NB	FLP	7.7	37.1
	+Web	4.6	34.3
WB	FLP	23.4	52.3
	+Web	4.7	25.9

- Evaluated on two types of data
  - NB: narrow-band data from conversational telephone speech
  - WB: wide-band data from news/topical speech data
- Significant gains on WB, small gains on NB

Lexicon

Pronunciation Lexicon



Phonetic Lexicons

- Most speech recognition systems use a phonetic lexicon:

A	ax
A	ey
A.	ey
A.'S	ey z
AAH	aa

- Each phone has attributes used for decision tree questions

ax	Vowel V-Back Back Short Medium Unrounded
ey	Vowel Short Dipthong Front-Start Fronting Medium Unrounded
z	Fricative Central Lenis Coronal Anterior Continuent Strident

- Initial phonetic lexicon generated manually
    - add terms using grapheme-to-phoneme (G2P) systems

Phonetic Lexicons

- Most speech recognition systems use a phonetic lexicon:

A	ax
A	ey
A.	ey
A.'S	ey z
AAH	aa

- Each phone has attributes used for decision tree questions

ax	Vowel V-Back Back Short Medium Unrounded
ey	Vowel Short Dipthong Front-Start Fronting Medium Unrounded
z	Fricative Central Lenis Coronal Anterior Continuent Strident

- Initial phonetic lexicon generated manually
  - add terms using grapheme-to-phoneme (G2P) systems

Graphemic Lexicons [8]

- As well as manual cost other issues with phonetic lexicons
  - inconsistencies depending on the phonetician
  - sometimes transcriptions generated for particular speaker
- An alternative is to generate a graphemic lexicon

A	a^I
A.	a^I;B
A.'S	a^I;BA s^F
AAH	a^I a^M h^F

  - deterministic process - no manual/G2P system required
- CUED system additional markers added (phonetic possible)
  - A - apostrophe following the letter
  - B - abbreviation (A., B. etc)
  - position - I (initial), M (middle), F (final)

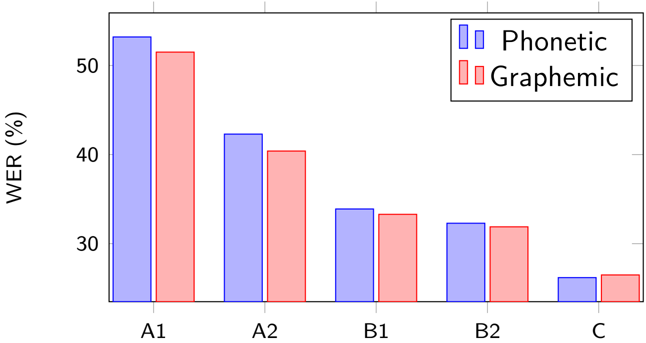
Graphemic Lexicons [8]

- As well as manual cost other issues with phonetic lexicons
  - inconsistencies depending on the phonetician
  - sometimes transcriptions generated for particular speaker
- An alternative is to generate a graphemic lexicon

A	a^I
A.	a^I;B
A.'S	a^I;BA s^F
AAH	a^I a^M h^F

  - deterministic process - no manual/G2P system required
- CUED system additional markers added (phonetic possible)
  - A - apostrophe following the letter
  - B - abbreviation (A., B. etc)
  - position - I (initial), M (middle), F (final)

Performance on English - Non-Native Learners [9]



- For "beginners" graphemic systems outperform phonetic
  - as ability improves ASR performance improves
  - graphemic systems can be useful for (even) English!

Writing Systems

Writing Systems

- English/European languages Latin script is used

What about general languages world-wide?

- There are a range of writing schemes used:
  - Pictographic - graphemes represent concepts
  - Logographic - graphemes represent words of morphemes
  - Syllabries - graphemes represent syllables
  - Segmental - form examined on the Babel project
- Segmental writing systems can be further partitioned as
  - alphabet - consonants and vowels both written
  - abugida - vowels marked as diacritics on consonants
  - abjad - only the consonants are written

Writing Systems

- English/European languages Latin script is used

What about general languages world-wide?

- There are a range of writing schemes used:
  - Pictographic - graphemes represent concepts
  - Logographic - graphemes represent words of morphemes
  - Syllabries - graphemes represent syllables
  - Segmental - form examined on the Babel project
- Segmental writing systems can be further partitioned as
  - alphabet - consonants and vowels both written
  - abugida - vowels marked as diacritics on consonants
  - abjad - only the consonants are written

Example Writing Schemes

Language	System	Script	Graphemes
Pashto	Abjad	Arabic	47
Tagalog	Alphabet	Latin	53†
Tamil	Abugida	Tamil	48
Zulu	Alphabet	Latin	52†
Kazakh	Alphabet	Cyrillic/Latin	126†
Telugu	Abugida	Telugu	60
Amharic	Abugida	Ethiopic	247
Mongolian	Alphabet	Cyrillic	66†

- Count excludes apostrophe, hyphen, punctuation ...
  - includes capitals for Latin/Cyrillic scripts

Graphemic System Attributes [4, 15]

- Often no attributes associated with graphemes
  - limits decision tree questions to grapheme
  - no attributes such as voiced/unvoiced
  - how to handle very rare graphemes?
- Interesting to examine additional attributes
  - bottom-up clustering of observed graphemes
  - make use of attributes of the unicode coding
- Diacritics not always marked on found data
  - can yield mismatch in vocabulary and pronunciation

Graphemic System Attributes [4, 15]

- Often no attributes associated with graphemes
  - limits decision tree questions to grapheme
  - no attributes such as voiced/unvoiced
  - how to handle very rare graphemes?
- Interesting to examine additional attributes
  - bottom-up clustering of observed graphemes
  - make use of attributes of the unicode coding
- Diacritics not always marked on found data
  - can yield mismatch in vocabulary and pronunciation

Kazakh Lexicon

- Mixture of Cyrillic and Latin script
  - use unicode descriptors to map between forms

i	G6;D2D3D6	LATIN SMALL LETTER I
I	G6;D8D3D6	LATIN CAPITAL LETTER I
и	G6;D1D2D3	CYRILLIC SMALL LETTER I
ѝ	G6;D1D2D3D4	CYRILLIC SMALL LETTER I WITH GRAVE
й	G6;D1D2D3D5	CYRILLIC SMALL LETTER SHORT I

where the following attributes are defined

D1	CYRILLIC	D2	SMALL	D3	LETTER	D4	WITH GRAVE
D5	SHORT	D6	LATIN	D8	CAPITAL		

- Able to relate accented letters to root grapheme
  - also delete diacritics from actual graphemes

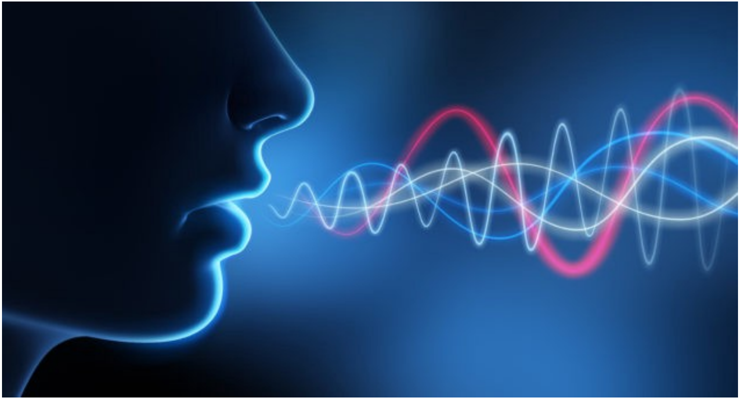
Phonetic vs Graphemic Performance

Language	Script	WER (%)		
		Phon	Grph	Comb
Tok Pisin	Latin	40.6	41.1	39.4
Kazakh	Cyrillic/Latin	53.5	52.7	51.5
Telugu	Telugu	69.1	69.5	67.5

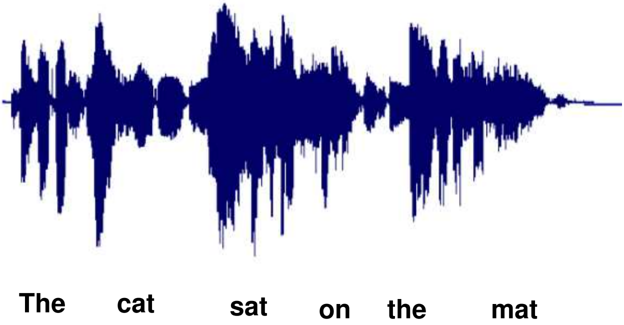
- Comparable performance of graphemic/phonetic systems
  - graphemic/phonetic systems are complementary to one another
- Similar trend observed over many other languages

Acoustic Model

Speech Data



Acoustic Model Training



- Acoustic models training with supervised training
  - pairs: (parametrised) waveform & orthographic transcription

Handling Limited Training Data

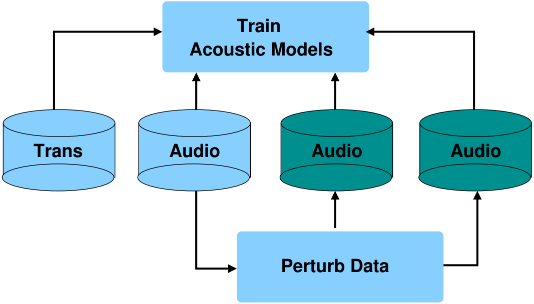
- Increased training data yields performance gains
  - but collecting data may be expensive (depending on language)
  - manually transcribing data expensive (alt. crowd-sourcing)
- Interested in approaches that increase data quantity
  - without incurring significant costs
- Approaches discussed here
  - data perturbation (artificially generate data)
  - multi-language acoustic models
  - semi-supervised training (use untranscribed data)

Handling Limited Training Data

- Increased training data yields performance gains
  - but collecting data may be expensive (depending on language)
  - manually transcribing data expensive (alt. crowd-sourcing)
- Interested in approaches that increase data quantity
  - without incurring significant costs

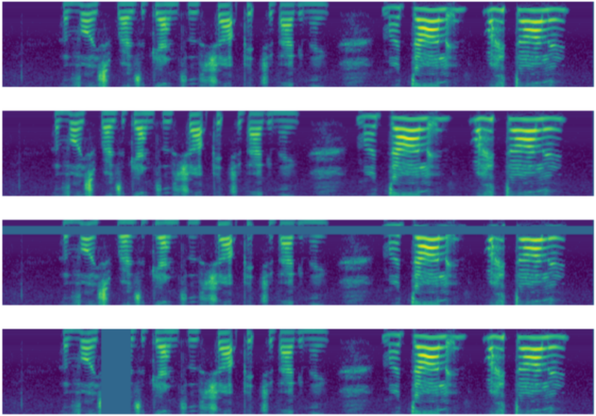
- Approaches discussed here
  - data perturbation (artificially generate data)
  - multi-language acoustic models
  - semi-supervised training (use untranscribed data)

Data Perturbation [14]



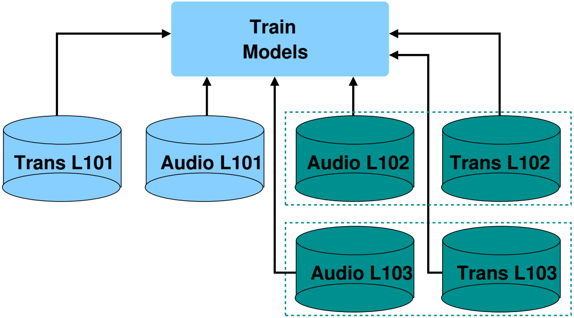
- Perturb data with speaker perturbation
  - synthesise data at a range of VTLN warp factors
  - also possible to use speed and noise perturbation
- Transcription is known!

SpecAugment [12]



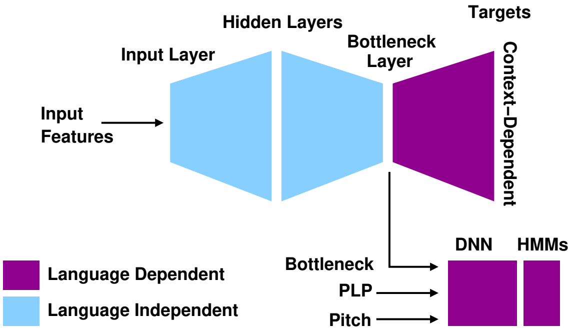
- Motivated by computer vision occlusion
  - mask regions of time/frequency in the training data

Multi-Language Framework [3]



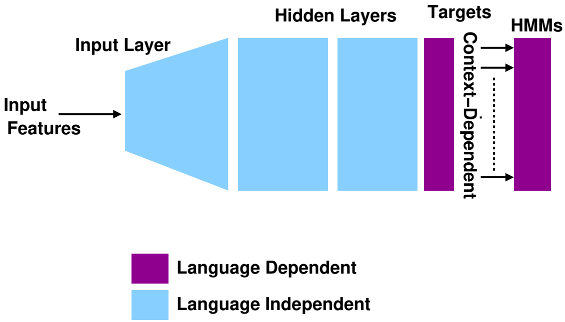
- Data from non-target language used to train model:
  - train complete acoustic model (see later)
  - train DNN to extract multi-language features

Multi-Language Bottleneck Features



- Generate features from multiple languages
  - aim to make feature extractor language independent
  - all layers other than output layer shared over all languages
  - output-layer language-specific - "hat-swapping"

Multi-Language Acoustic Models



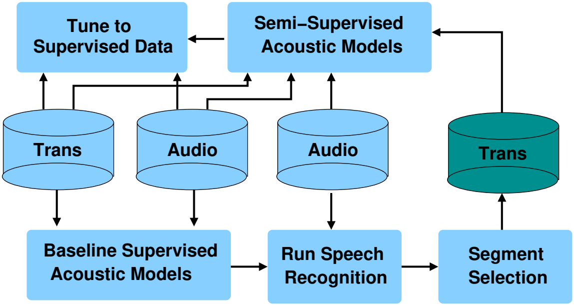
- Shared layer of networks over multiple acoustic models
  - output-layer language-specific - "hat-swapping"
  - can "fine-tune" parameters to target language

Multilingual System Performance
TDNN-LSTM

System	Language (WER %)		
	Bulgarian	Lithuanian	Tagalog
—	39.3	41.1	41.1
ML-Feature	35.2	38.1	39.5
ML-Model	37.2	39.3	39.6

- Multi-Language models based on 20+ languages
  - performance gains for all set-ups using multi-lingual data
- Contrast of features and models
  - additional hyper-parameter tuning needed for ML-Model

Semi-Supervised Training: Framework



- Segment level selection of data to use
  - use confidence scores in data selection

Semi-Supervised Training: Criterion/Regularisation

- Split data according to training criterion
  1. train network using all data using cross-entropy criterion
  2. train network using transcribed data and sequence training

- Use unsupervised trained network as a prior
  1. train network using all data ($\mathcal{M}_{\tt prior}$)
  2. train network using transcribed data using $\mathcal{M}_{\tt prior}$ as prior
- For CE training this yields

$$\mathcal{F}(\mathcal{M}) = \sum_{i=1}^{T}\sum_{k=1}^{K} t_{ik}\log(P(\omega_k|\boldsymbol{x}_i,\mathcal{M})) + \alpha\sum_{i=1}^{T}\sum_{k=1}^{K} P(\omega_k|\boldsymbol{x}_i,\mathcal{M}_{\tt prior})\log(P(\omega_k|\boldsymbol{x}_i,\mathcal{M}))$$

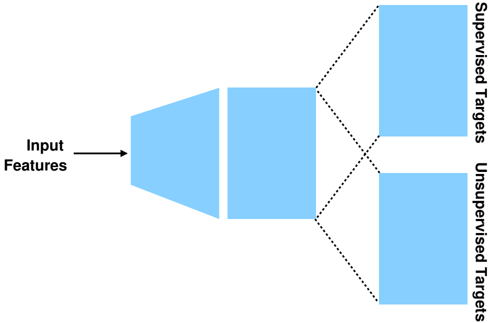
Semi-Supervised Training: Criterion/Regularisation

- Split data according to training criterion
  1. train network using all data using cross-entropy criterion
  2. train network using transcribed data and sequence training

- Use unsupervised trained network as a prior
  1. train network using all data ($\mathcal{M}_{\tt prior}$)
  2. train network using transcribed data using $\mathcal{M}_{\tt prior}$ as prior
- For CE training this yields

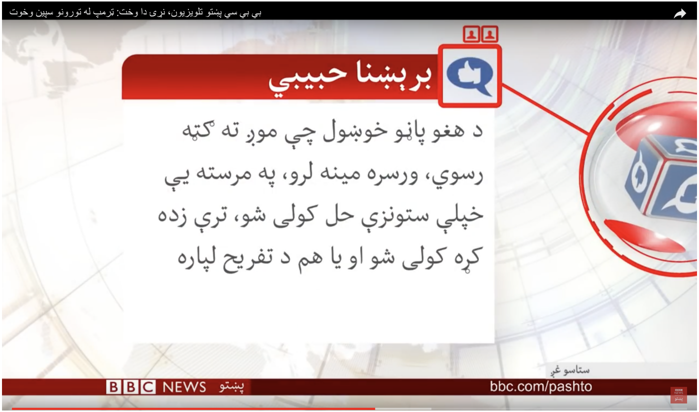
$$
\begin{aligned}
\mathcal{F}(\mathcal{M}) = & \sum_{i=1}^{T}\sum_{k=1}^{K} t_{ik}\log(P(\omega_k|\boldsymbol{x}_i,\mathcal{M})) \\
& +\alpha\sum_{i=1}^{T}\sum_{k=1}^{K} P(\omega_k|\boldsymbol{x}_i,\mathcal{M}_{\tt prior})\log(P(\omega_k|\boldsymbol{x}_i,\mathcal{M}))
\end{aligned}
$$

Semi-Supervised Learning: Multi-Task Criterion



- Have two separate output layers:
    - targets associated with transcribed training data
    - targets associated with untranscribed training data
- The training utterance transcription determines output layer
    - simple form of "hat-swapping" (change output layer)

Example Data Source: BBC Pashto



Semi-Supervised Porting

- Possible mismatch between transcribed/evaluation data
  - transcribed data: narrow-band conversational telephone speech
  - evaluation data: wide-band broadcast and podcast speech

- Train acoustic and language models on available data
  1. collect text web-data for target domain
  2. down-sample evaluation data to narrow-band - recognise data
  3. select data for model training - use wide-band parameters
  4. train model - no use of transcribed data

Semi-Supervised Porting

- Possible mismatch between transcribed/evaluation data
  - transcribed data: narrow-band conversational telephone speech
  - evaluation data: wide-band broadcast and podcast speech

- Train acoustic and language models on available data
  1. collect text web-data for target domain
  2. down-sample evaluation data to narrow-band - recognise data
  3. select data for model training - use wide-band parameters
  4. train model - no use of transcribed data

Semi-Supervised Porting: Narrow-to-Wide Band
TDNN-F

System	Language (WER %)					
	Bulgarian		Tagalog		Somali	
	NB	WB	NB	WB	NB	WB
ML-Feature	34.3	23.6	39.2	36.0	52.7	59.0
ML-Model	35.7	23.0	39.2	37.0	52.2	53.7
Comb	32.9	21.4	37.3	34.5	50.0	53.7

- Multi-Language models based on 20+ languages
  - performance gains for all set-ups using multi-lingual data
  - additional hyper-parameter tuning needed for ML-Model
- Down-sample WB data to allow NB models to be used

Semi-Supervised Porting: Narrow-to-Wide Band

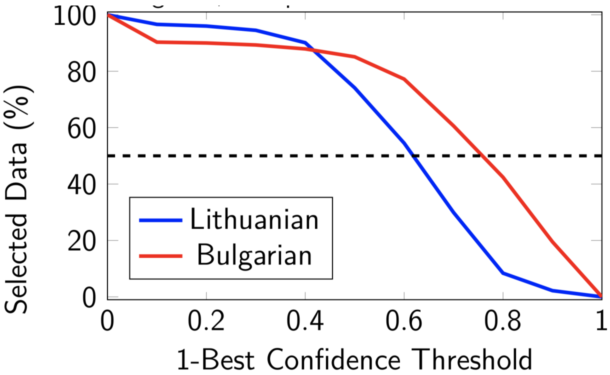
TDNN-F

System	Bulgarian NB	Bulgarian WB	Tagalog NB	Tagalog WB	Somali NB	Somali WB
ML-Feature	34.3	23.6	39.2	36.0	52.7	59.0
ML-Model	35.7	23.0	39.2	37.0	52.2	53.7
Comb	32.9	21.4	37.3	34.5	50.0	53.7

Language (WER %)

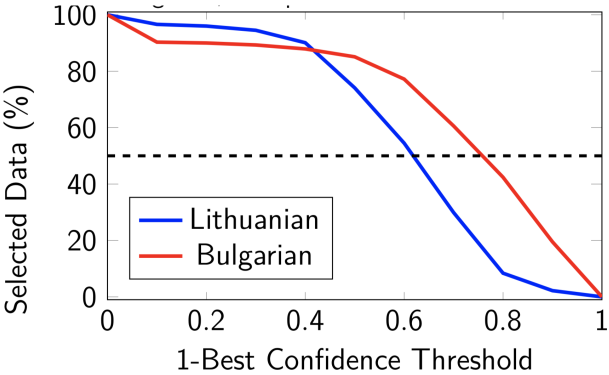
- Multi-Lingual models based on 20+ languages
  - performance gains for all set-ups using multi-lingual data
  - additional hyper-parameter tuning needed for ML-Model
- Down-sample WB data to allow NB models to be used

Confidence-Based Data Selection



- Select data with the highest confidence score
  - compute average confidence score for each utterance
  - automatically does language verification per utterance
- Alternative approach is to use lattices during training

Confidence-Based Data Selection



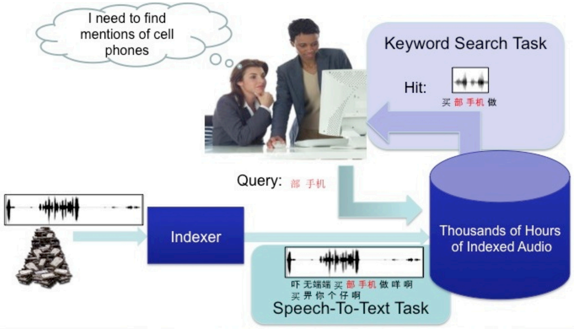
- Select data with the highest confidence score
  - compute average confidence score for each utterance
  - automatically does language verification per utterance
- Alternative approach is to use lattices during training

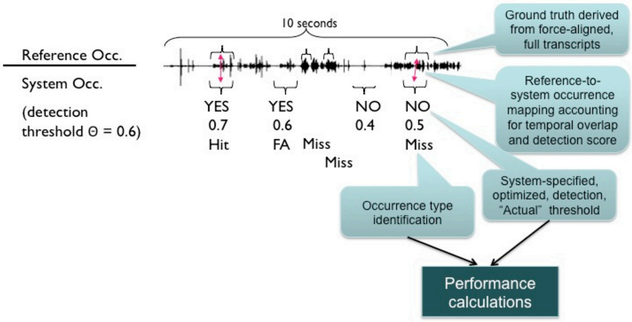
Language	YT (hrs)		WER %	
	Avl	Sel	NB	YT
Bulgarian	2382	1444	23.6	17.8
Lithuanian	805	439	25.9	20.6

- Use ML-features to transcribe WB YouTube (YT) data
  - select 50% of data using confidence scores
  - train model only on WB data

Downstream Processing

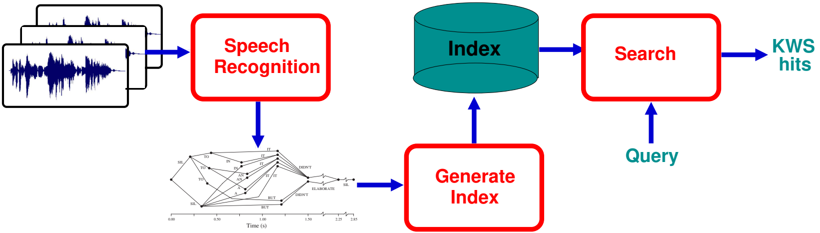
Task: Key Word (Phrase) Spotting [5]





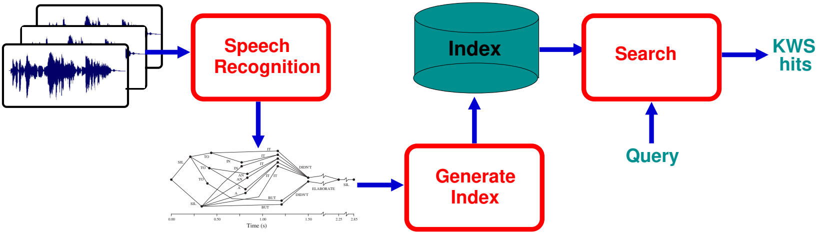
- Term Weighted Value (TWV) - official metric ($\beta = 999.9$)
  - $TWV(\theta) = 1 - [P_{Miss}(\theta) + \beta P_{FA}(\theta)]$

Key Word (Phrase) Spotting: Framework

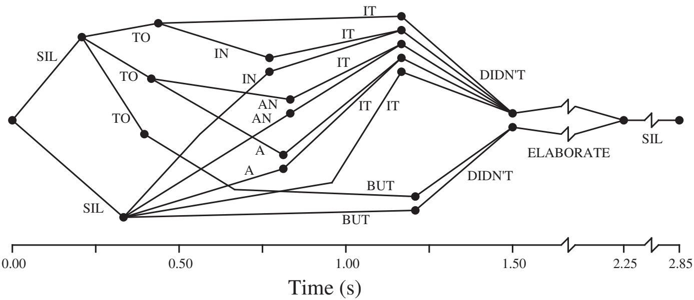


- Key problems are:
  - ASR systems with very limited training data available
  - ASR systems for highly diverse languages
  - KWS systems with high out-of-vocabulary query terms
  - KWS for low accuracy ASR systems

Key Word (Phrase) Spotting: Framework



- Key problems are:
  - ASR systems with very limited training data available
  - ASR systems for highly diverse languages
  - KWS systems with high out-of-vocabulary query terms
  - KWS for low accuracy ASR systems



- A lattice, $\mathcal{L}$, comprises:
  - nodes (sometimes called state): associated with time stamps
  - arcs: have labels and scores (not shown)

Word Search Strategy

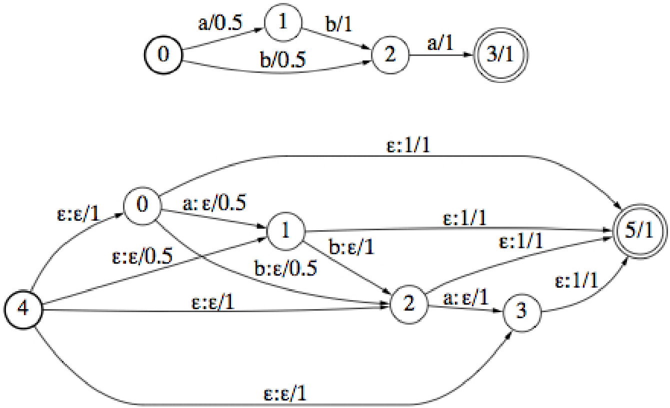
- Initially just consider detecting whether a word, $\tilde{w}$, occurs
  1. retrieve all arcs, a, in the index for which $a \in \mathcal{I}(\tilde{w})$ (grouped according to time-stamp information as well)
  2. compute the posterior for that arc in the lattice $P(a|\mathcal{L}(a))$
  3. construct the probability for word $\tilde{w}$ in lattice $\mathcal{L}$

$$P(\tilde{w}|\mathcal{L}) = \sum_{a \in \mathcal{I}(\tilde{w}):\mathcal{L}(a)=\mathcal{L}} P(a|\mathcal{L}(a))$$

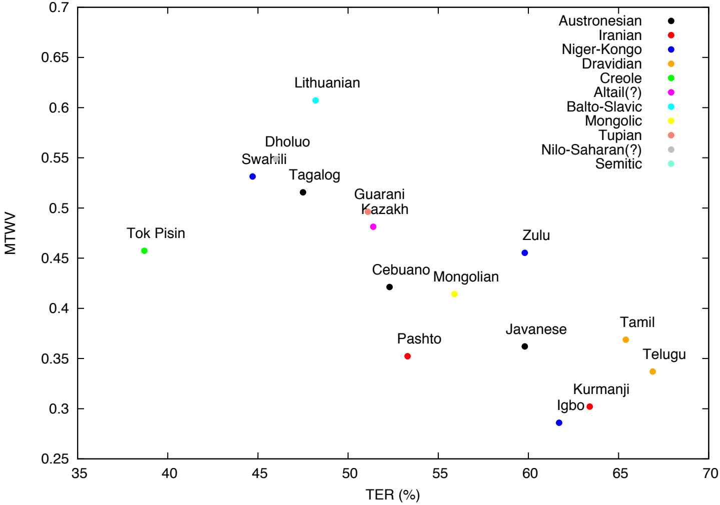
  4. define a threshold of $P(\tilde{w}|\mathcal{L})$ for existence of word in utterance

- Yields count for a particular word for a lattice.
  - how to obtain the posterior efficiently and handle phrases

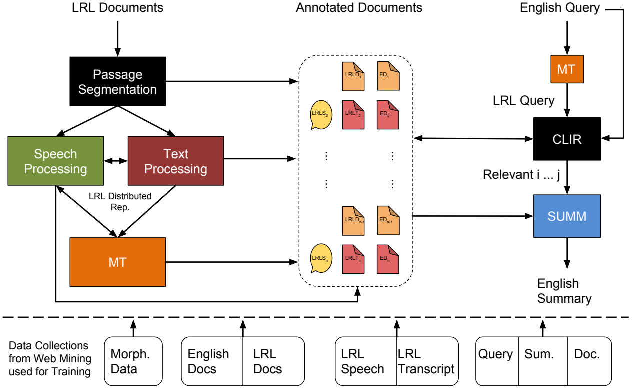
WFST Index Implementation [2]



Highly Diverse Languages - ASR/KWS Performance

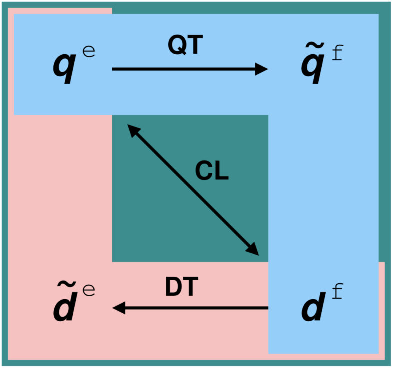


Task: Cross Language Information Retrieval [11]



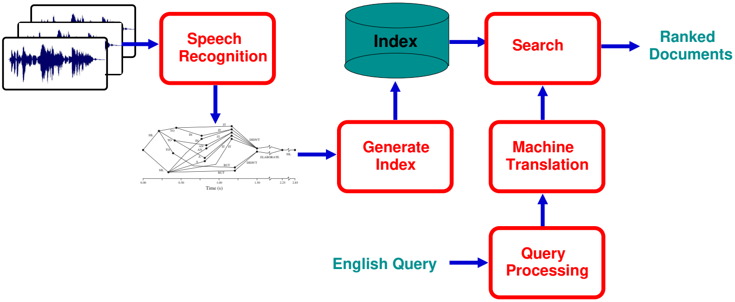
- Find documents in source language relevant to English query

CLIR: Search Options



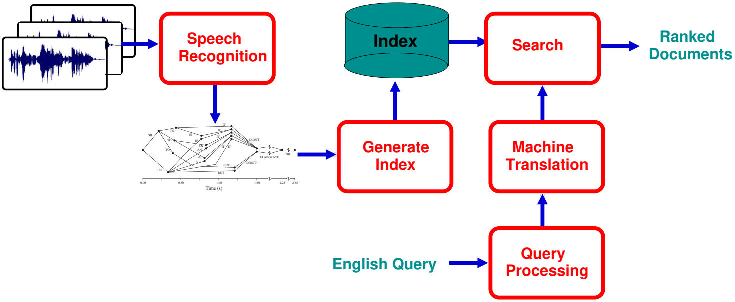
Queries:	English ($\mathbf{q}^e$)	translate (QT)	Source ($\tilde{\mathbf{q}}^f$)
Document:	Source ($\mathbf{d}^f$)	translate (DT)	English ($\tilde{\mathbf{d}}^e$)

CLIR: framework



- Only consider search in source language
- Additional challenge
  - limited machine translation data
  - need to generalise beyond word/phrase occurrences

CLIR: framework



- Only consider search in source language
- Additional challenge
  - limited machine translation data
  - need to generalise beyond word/phrase occurrences

Generative CLIR

- Compute probability generating query $\mathbf{q}^{\mathrm{e}}$ from document $\mathbf{d}^{\mathrm{f}}$

$$P(\mathbf{q}^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}}) = \prod_{w^{\mathrm{e}} \in \mathbf{q}^{\mathrm{e}}} \left[(1-\alpha)P(w^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}}) + \alpha P(w^{\mathrm{e}}|\mathbf{g}^{\mathrm{e}})\right]$$

  - $\mathbf{g}^{\mathrm{e}}$ general English model - used for smoothing
  - α tunable model (usually small 0.1)

- Need to find $P(w^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}})$ from spoken document
  - from ASR $\mathbf{d}^{\mathrm{f}} \rightarrow \mathcal{L}^{\mathrm{f}}$

$$P(w^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}}) = \sum_{w^{\mathrm{f}} \in \mathcal{L}^{\mathrm{f}}} P(w^{\mathrm{e}}|w^{\mathrm{f}})P(w^{\mathrm{f}}|\mathcal{L}^{\mathrm{f}})$$

  - $P(w^{\mathrm{e}}|w^{\mathrm{f}})$ word-level translation table - requires limited data
  - $P(w^{\mathrm{f}}|\mathcal{L}^{\mathrm{f}})$ similar to word-level KWS

Generative CLIR

- Compute probability generating query $\mathbf{q}^{\mathrm{e}}$ from document $\mathbf{d}^{\mathrm{f}}$

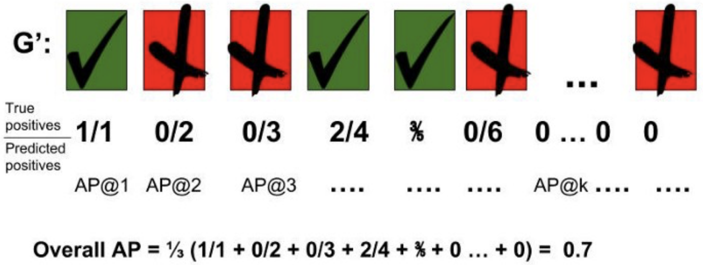
$$P(\mathbf{q}^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}}) = \prod_{w^{\mathrm{e}} \in \mathbf{q}^{\mathrm{e}}} \left[(1-\alpha)P(w^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}}) + \alpha P(w^{\mathrm{e}}|\mathbf{g}^{\mathrm{e}}) \right]$$

  - $\mathbf{g}^{\mathrm{e}}$ general English model - used for smoothing
  - α tunable model (usually small 0.1)

- Need to find $P(w^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}})$ from spoken document
  - from ASR $\mathbf{d}^{\mathrm{f}} \rightarrow \mathcal{L}^{\mathrm{f}}$

$$P(w^{\mathrm{e}}|\mathbf{d}^{\mathrm{f}}) = \sum_{w^{\mathrm{f}} \in \mathcal{L}^{\mathrm{f}}} P(w^{\mathrm{e}}|w^{\mathrm{f}}) P(w^{\mathrm{f}}|\mathcal{L}^{\mathrm{f}})$$

  - $P(w^{\mathrm{e}}|w^{\mathrm{f}})$ word-level translation table - requires limited data
  - $P(w^{\mathrm{f}}|\mathcal{L}^{\mathrm{f}})$ similar to word-level KWS



- Use mean average precision to assess system performance
  - standard information retrieval metric
  - only assesses the ranking of the documents retrieved

CLIR Performance

Language	ASR System	WER % NB	WER % WB	mAP 1-Best	mAP Lat
Swahili	CUED	36.0	31.5	0.2058	0.2088
Bulgarian	CUED	32.6	18.9	0.7366	0.7413
Lithuanian	CUED1	41.8	24.4	0.6466	0.7049
	CUED2	37.4	21.4	0.6666	0.7477
	CUED3	35.8	20.6	0.6948	0.7440

- Consistent gains using lattices over 1-best
  - lattice search less sensitive to ASR accuracy
  - but need to control lattice size

Conclusions

Conclusions

- "Plug and Play" systems built for diverse languages
  - graphemic lexicons worked well for all languages
- Multi-language acoustic models important
  - either bottleneck features, or "complete" models
- Data augmentation approaches important
  - semi-supervised training can handle acoustic mismatch
- Use "rich" output from ASR system (lattices)
  - improves downstream application performance

Conclusions

- "Plug and Play" systems built for diverse languages
  - graphemic lexicons worked well for all languages
- Multi-language acoustic models important
  - either bottleneck features, or "complete" models
- Data augmentation approaches important
  - semi-supervised training can handle acoustic mismatch
- Use "rich" output from ASR system (lattices)
  - improves downstream application performance

Conclusions

- "Plug and Play" systems built for diverse languages
  - graphemic lexicons worked well for all languages
- Multi-language acoustic models important
  - either bottleneck features, or "complete" models
- Data augmentation approaches important
  - semi-supervised training can handle acoustic mismatch
- Use "rich" output from ASR system (lattices)
  - improves downstream application performance

Conclusions

- “Plug and Play” systems built for diverse languages
  - graphemic lexicons worked well for all languages
- Multi-language acoustic models important
  - either bottleneck features, or “complete” models
- Data augmentation approaches important
  - semi-supervised training can handle acoustic mismatch
- Use “rich” output from ASR system (lattices)
  - improves downstream application performance

Thank-you!

Acknowledgements

This work was supported in part by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD/ARL) contract number W911NF-12-C-0012. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.
This work made use of data provided by IARPA [1].
The authors would like to thank the contributions of the members of the CUED Babel team during the project, and all the members of the LORELEI team, in particular the IBM and RWTH Aachen Babel teams.

[1] The following data was used in the FLP configuration: IARPA-babel106-v0.2f, IARPA-babel202b-v1.0d, IARPA-babel204b-v1.1b, IARPA-babel205b-v1.0a, IARPA-babel206b-v0.1d, IARPA-babel207b-v1.0a, IARPA-babel301b-v1.0b, IARPA-babel302b-v1.0a, IARPA-babel303b-v1.0a, IARPA-babel304b-v1.0b, IARPA-babel104b-v0.4bY, IARPA-babel306b-v2.0c, IARPA- babel401b-v2.0b, IARPA-babel402b-v1.0b, IARPA-babel403b-v1.0b, IARPA-babel404b-v1.0a.

[1] I. Bulyko, M. Ostendorf, and A. Stolcke, “Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures,” in *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, HLT-NAACL 2003, Edmonton, Canada, May 27 - June 1, 2003*, M. A. Hearst and M. Ostendorf, Eds. The Association for Computational Linguistics, 2003. [Online]. Available: https://www.aclweb.org/anthology/N03-2003/

[2] D. Can and M. Saraclar, “Lattice indexing for spoken term detection,” *IEEE Trans. Audio, Speech & Language Processing*, vol. 19, no. 8, pp. 2338–2347, 2011. [Online]. Available: https://doi.org/10.1109/TASL.2011.2134087

[3] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nußbaum-Thom, M. Picheny, Z. Tüske, P. Golik, R. Schlüter, H. Ney, M. J. F. Gales, K. M. Knill, A. Ragni, H. Wang, and P. C. Woodland, “Multilingual representations for low resource speech recognition and keyword search,” in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2015, Scottsdale, AZ, USA, December 13-17, 2015*. IEEE, 2015, pp. 259–266. [Online]. Available: https://doi.org/10.1109/ASRU.2015.7404803

[4] M. J. F. Gales, K. M. Knill, and A. Ragni, “Unicode-based graphemic systems for limited resource languages,” in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*. IEEE, 2015, pp. 5186–5190. [Online]. Available: https://doi.org/10.1109/ICASSP.2015.7178960

[5] ——, “Low-resource speech recognition and keyword-spotting,” in *Speech and Computer - 19th International Conference, SPECOM 2017, Hatfield, UK, September 12-16, 2017, Proceedings*, ser. Lecture Notes in Computer Science, A. Karpov, R. Potapova, and I. Mporas, Eds., vol. 10458. Springer, 2017, pp. 3–19. [Online]. Available: https://doi.org/10.1007/978-3-319-66429-3_1

[6] M. Gales and S. Young, “The application of hidden Markov models in speech recognition,” *Foundations and Trends in Signal Processing*, vol. 1, no. 3, 2007.

[7] R. Iyer, M. Ostendorf, and H. Gish, “Using out-of-domain data to improve in-domain language models,” *IEEE Signal Processing Letters*, vol. 4, no. 8, pp. 221–223, Aug 1997.

[8] S. Kanthak and H. Ney, “Context-dependent acoustic modeling using graphemes for large vocabulary speech recognition,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2002, May 13-17 2002, Orlando, Florida, USA*. IEEE, 2002, pp. 845–848. [Online]. Available: https://doi.org/10.1109/ICASSP.2002.5743871

[9] K. M. Knill, M. J. F. Gales, K. Kyriakopoulos, A. Ragni, and Y. Wang, “Use of graphemic lexicons for spoken language assessment,” in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, F. Lacerda, Ed. ISCA, 2017, pp. 2774–2778. [Online]. Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0978.html

[10] G. Mendels, E. Cooper, and J. Hirschberg, “Babler - data collection from the web to support speech recognition and keyword search,” in *Proceedings of the 10th Web as Corpus Workshop*. Berlin: Association for Computational Linguistics, Aug. 2016, pp. 72–81. [Online]. Available: https://www.aclweb.org/anthology/W16-2609

[11] D. W. Oard, “Transcending the tower of babel: Supporting access to multilingual information with cross-language information retrieval,” in *Emergent Information Technologies and Enabling Policies for Counter-Terrorism*, R. L. Popp and J. Yen, Eds. Wiley-IEEE Press, 2006.

[12] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “Specaugment: A simple data augmentation method for automatic speech recognition,” *CoRR*, vol. abs/1904.08779, 2019. [Online]. Available: http://arxiv.org/abs/1904.08779

[13] A. Ragni and M. J. F. Gales, “Automatic speech recognition system development in the "wild",” in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed. ISCA, 2018, pp. 2217–2221. [Online]. Available: https://doi.org/10.21437/Interspeech.2018-1085

[14] A. Ragni, K. M. Knill, S. P. Rath, and M. J. F. Gales, “Data augmentation for low resource languages,” in *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, H. Li, H. M. Meng, B. Ma, E. Chng, and L. Xie, Eds. ISCA, 2014, pp. 810–814. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2014/i14_0810.html

[15] Y. Wang, X. Chen, M. J. F. Gales, A. Ragni, and J. H. M. Wong, “Phonetic and graphemic systems for multi-genre broadcast transcription,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*. IEEE, 2018, pp. 5899–5903. [Online]. Available: https://doi.org/10.1109/ICASSP.2018.8462353