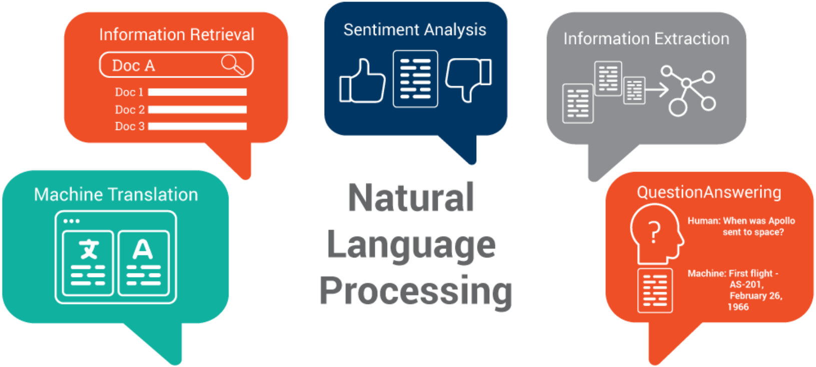
UNIVERSITY OF CAMBRIDGE

MLMI14: Spoken Language Processing and Generation

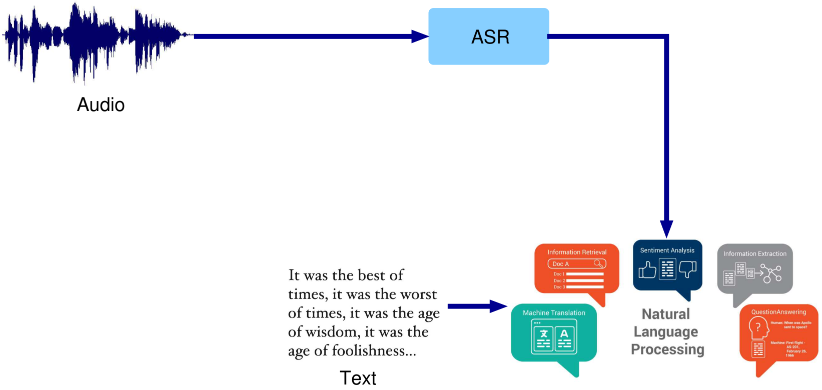
Mark Gales

Lent 2022

Natural Language Processing Applications



Simple Spoken Language Processing



- Convert the audio into text
  - treat as identical to written text

Example of Native English

ASR Output

okay carl uh do you exercise yeah actually um i belong to a gym down here gold's gym and uh i try to exercise five days a week um and now and then i'll i'll get it interrupted by work or just full of crazy hours you know

Example of Native English

ASR Output

okay carl uh do you exercise yeah actually um i belong to a gym down here gold's gym and uh i try to exercise five days a week um and now and then i'll i'll get it interrupted by work or just full of crazy hours you know

Meta-Data Extraction (MDE) Markup

Speaker1: / okay carl {F uh} do you exercise /
Speaker2: / {DM yeah actually} {F um} i belong to a gym down here /
/ gold's gym / / and {F uh} i try to exercise five days a week {F um} /
/ and now and then [REP i'll + i'll] get it interrupted by work or just full of crazy hours {DM you know } /

Example of Native English

ASR Output

okay carl uh do you exercise yeah actually um i belong to a gym down here gold's gym and uh i try to exercise five days a week um and now and then i'll i'll get it interrupted by work or just full of crazy hours you know

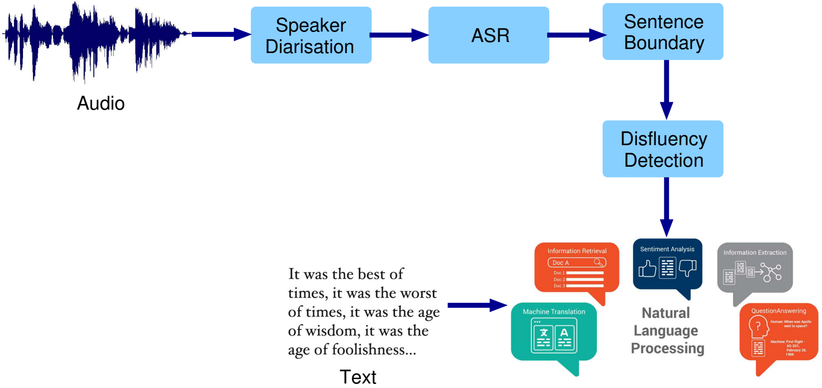
Meta-Data Extraction (MDE) Markup

Speaker1: / okay carl {F uh} do you exercise /
Speaker2: / {DM yeah actually} {F um} i belong to a gym down here /
/ gold's gym / / and {F uh} i try to exercise five days a week {F um} /
/ and now and then [REP i'll + i'll] get it interrupted by work or just full of crazy hours {DM you know } /

Written Text

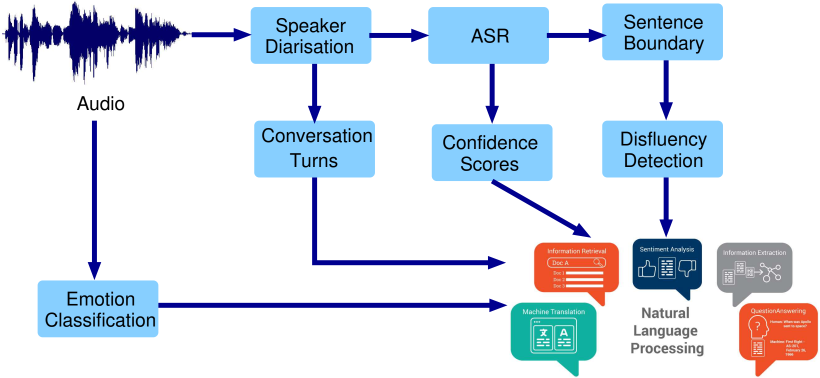
Speaker1: Okay Carl do you exercise?
Speaker2: I belong to a gym down here, Gold's Gym, and I try to exercise five days a week and now and then I'll get it interrupted by work or just full of crazy hours.

Spoken Language Processing Pipeline



- Convert the audio into form closer to written text
  - incorporate sentence boundary detection
  - incorporate disfluency detection removes:
    - repetitions, false-starts, hesitations, dialogue markers

Spoken Language Processing Pipeline



- Extract additional information from the acoustic signal, e.g.
  - link speaker turns, progress of interaction
  - confidence scores limit impact of low confidence words

Written vs Spoken Language Processing

- Advantages: written vs spoken
  - grammar is clearly defined for written text
  - no speech recognition errors
  - large(ish) data available (standard text sets)
  - no fillers, dialogue markers, false starts and repetitions
  - sentences normally clearly defined by punctuation
- Disadvantages: written vs spoken
  - spelling mistakes and out-of-vocabulary words
  - no audio information available, e.g. emotion

Spoken Language Generation Applications

skype™ Translate

Course Outline

- Spoken Language Processing - Introduction
- Keyword spotting
  - practical based on KWS - available from moodle
  - based on BABEL/MATERIAL projects
- Spoken language assessment and learning (2 lectures)
  - based on ALTA Institute research
- Meeting and spoken document summarisation
  - Spotify podcast challenge system and transformer attention

- Spoken Langage Generation - Introduction
- Deep learning for speech synthesis (2 lectures)

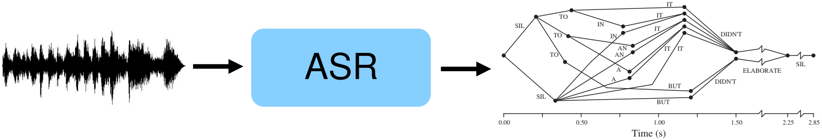
ASR Confidence Scores

ASR Confidence Scores [9]

- Useful to know whether ASR output is correct
  - confidence scores supply this information
  - three forms of error: substitutions, deletions and insertions

manual	AND	THESE	ARE		THE	FIMBLES
asr		THIS	ARE	TO	THE	FIMBLES
error	del	sub	—	ins	—	—
conf	—	0.4	0.8	0.3	0.9	0.9

Baseline Confidence Scores

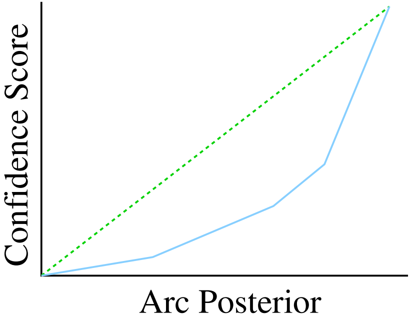


- Baseline confidence scores based on arc posteriors

$$\mathrm{p}(\boldsymbol{q}_{1:T}, \boldsymbol{x}_{1:T}) = \mathrm{p_a}(\boldsymbol{x}_{1:T}|\boldsymbol{q}_{1:T})^{\frac{1}{\gamma}} \mathrm{P_l}(w_{1:L}); \quad \mathrm{P}(a|\mathcal{L}) = \frac{\sum_{\boldsymbol{q}_{1:T} \in \mathcal{Q}_a} \mathrm{p}(\boldsymbol{q}_{1:T}, \boldsymbol{x}_{1:T})}{p(\boldsymbol{x}_{1:T})}$$

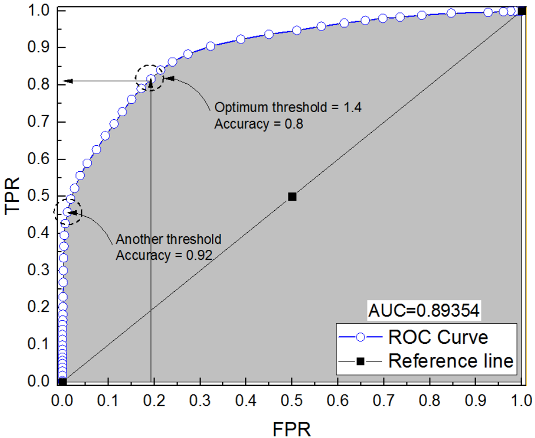
  - $\boldsymbol{q}_{1:T}$ T-length state sequence for word sequence $w_{1:L}$
  - $\mathcal{Q}_a$ set of state sequences that pass through arc a
  - γ is usually the LM scale factor
  - does not alter 1-best (compared to scaling LM)

Confidence Score Calibration [1]



- Confidence scores often over-estimated
    - simple piecewise-linear calibration approach

Assessment: Area Under Curve (Receiver Operating Curve)



- FPR False Positive Rate, TPR True Positive Rate
- Overall performance of system (no threshold selected)
  - rank-order based, not impacted by (monotonic) calibration

Assessment: Normalised Cross-Entropy

- The definition of the NCE score is

$$\texttt{NCE} = \frac{\mathcal{H}(\boldsymbol{c}) - \mathcal{H}(\boldsymbol{c}|\mathcal{M})}{\mathcal{H}(\boldsymbol{c})} \approx \frac{\mathcal{H}(\boldsymbol{c}) - \mathcal{H}(\boldsymbol{c}_{1:L}|\boldsymbol{\omega}_{1:L}, \mathcal{M})}{\mathcal{H}(\boldsymbol{c})}$$

- $\mathcal{H}(\boldsymbol{c})$:entropy of the class labels ($\texttt{correct}/\texttt{incorrect}$);

$$\mathcal{H}(\boldsymbol{c}) = -\overline{p}\log(\overline{p}) - (1-\overline{p})\log(1-\overline{p})$$

  - $\overline{p}$ estimated from error rate of hypothesis
- $\mathcal{H}(\boldsymbol{c}_{1:L}|\boldsymbol{\omega}_{1:L}, \mathcal{M})$ is the approx conditional entropy:
  - $\hat{\boldsymbol{c}}_{1:L}$ is the predicted confidence score

$$\mathcal{H}(\boldsymbol{c}_{1:L}|\boldsymbol{w}_{1:L}, \mathcal{M}) = -\frac{1}{L}\left(\sum_{i=1}^{L} c_i \log(\hat{c}_i) + (1-c_i)\log(1-\hat{c}_i)\right)$$

- Standard criterion used by NIST
  - note: NCE can go negative (due to approximation)

Additional Features for Confidence Estimation

- To improve performance extract additional features
  - confusion networks: extract CNs/word posteriors
  - acoustic stability: generate N-hypotheses
    - using a set of N language model scale-factors

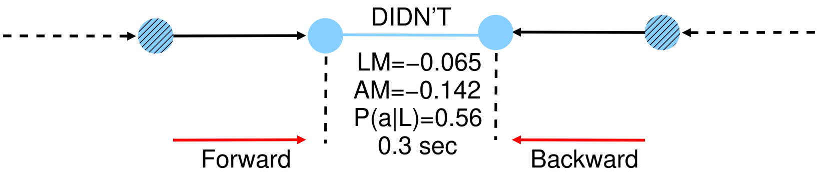
$$\mathcal{C}_{\text{as}}(w_i) = \frac{1}{N}\sum_{n=1}^{N}\delta(w_i, \texttt{Align}_i(\boldsymbol{w}_{1:L}, \boldsymbol{w}^{(n)}_{1:L^{(n)}}))$$

    - $\texttt{Align}_i()$ is the Levenschtein alignment of two sequences
    - correct words appear in same position for multiple LM factors
  - in 1-best: does the word occur at that position in the 1-best
  - hypothesis density: high lattice density - low confidence
  - language model scores: probability/back-off
  - acoustic model scores
  - word/phone durations

Combining Features and Sequence Confidence [8]

- Need to combine multiple features in probabilistic framework
- Errors tend to occur in bursts
  - language model links word-prediction to surrounding words
  - OOV phrases
  - interfering background noise
  - unseen accent, '"goats"
- Useful to predict complete confidence sequence $\hat{\boldsymbol{c}}_{1:L}$
- Standard classification problem
  - alternative model - Conditional Random Field
  - could use deep-learning ...
- Possible labels (states q_i): `correct`/`incorrect`
  - assume T length sequences, both word and confidence scores
  - general features $\boldsymbol{x}_t$ (use start state q_0)

Neural Network Based Confidence Scores



- Use more general sequence model
  - for 1-best $w_{1:L} = w_1, \ldots, w_L$
  - use information associated with each arc $\mathrm{a}_{1:L}$

$$P(w_i|\mathrm{a}_{1:L}) = \mathcal{F}(\mathrm{a}_i, \overrightarrow{\mathrm{a}}_{1:i-1}, \overleftarrow{\mathrm{a}}_{i+1:L})$$

RNN-Based Confidence Scores [2, 7]

- Simple approach use recurrent neural networks

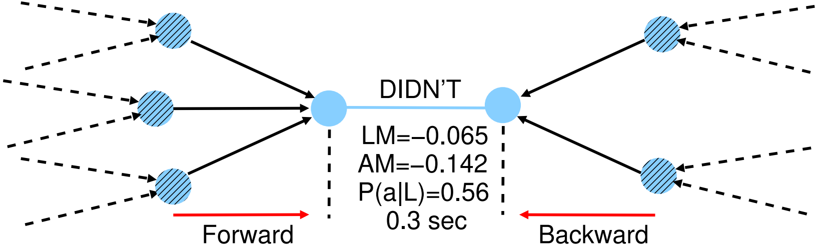
$$\overrightarrow{\boldsymbol{h}}_i = \mathcal{F}(\overrightarrow{\boldsymbol{h}}_{i-1}, \mathrm{a}_i); \quad \overleftarrow{\boldsymbol{h}}_i = \mathcal{F}(\overleftarrow{\boldsymbol{h}}_{i+1}, \mathrm{a}_i);$$

$$P(w_i|\mathrm{a}_{1:L}) = \mathcal{F}(\overrightarrow{\boldsymbol{h}}_i, \overleftarrow{\boldsymbol{h}}_i)$$

- Evaluation: Georgian Conversation Telephone Speech
  - ASR Performance: 38% Word Error Rate
    (not impacted by monotonic calibration)
  - RNN-based on: posteriors, word ID and durations

System	NCE	AUC
Arc posteriors	-0.1978	0.7112
+ calibration	0.2755	0.7112
+ RNN	0.2911	0.7194

Lattice Neural Network Based Confidence Scores [3, 4]

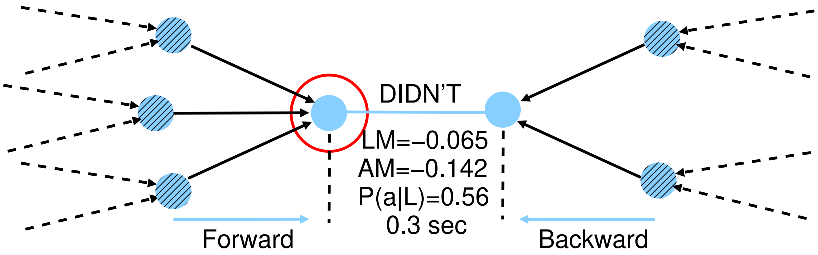


- Make use of complete lattice $\mathcal{L}$

$$P(w_i|\mathcal{L}) = \mathcal{F}(\mathtt{a}_i, \overrightarrow{\mathcal{Q}}_{\mathtt{a}_i}, \overleftarrow{\mathcal{Q}}_{\mathtt{a}_i})$$

  - $\overrightarrow{\mathcal{Q}}_{\mathtt{a}_i}$ set of arcs in forward direction to $\mathtt{a}_i$
  - $\overleftarrow{\mathcal{Q}}_{\mathtt{a}_i}$ set of arcs in backward direction to $\mathtt{a}_i$

Lattice Neural Network Based Confidence Scores

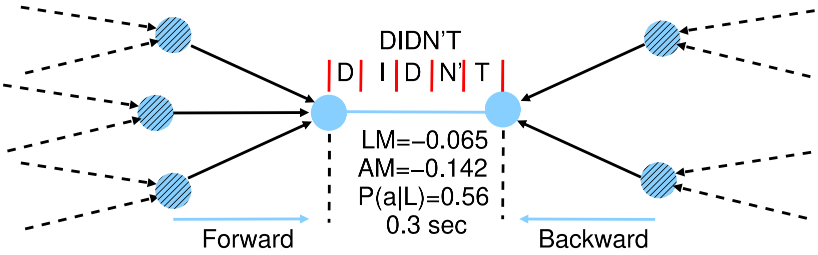


- Use attention to merge arcs - $\overrightarrow{\mathcal{N}}_{\mathrm{a}_i}$ "forward" neighbours of a_i

$$\overrightarrow{\boldsymbol{h}}_i = \mathtt{attention}\left(\{\overrightarrow{\boldsymbol{h}}_j\}_{j \in \overrightarrow{\mathcal{N}}_{\mathrm{a}_i}}, \mathrm{a}_i\right); \quad \overleftarrow{\boldsymbol{h}}_i = \mathtt{attention}\left(\{\overleftarrow{\boldsymbol{h}}_j\}_{j \in \overleftarrow{\mathcal{N}}_{\mathrm{a}_i}}, \mathrm{a}_i\right);$$

$$P(w_i|\mathrm{a}_{1:L}) = \mathcal{F}(\overrightarrow{\boldsymbol{h}}_i, \overleftarrow{\boldsymbol{h}}_i)$$

Grapheme Features



- Add grapheme ID and duration information

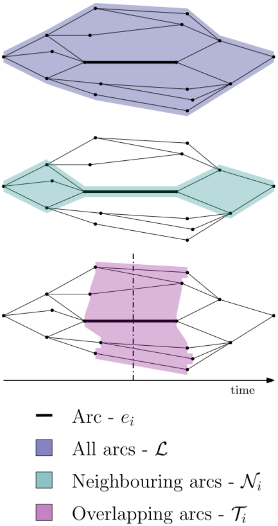
$$\boldsymbol{g}_i = \texttt{self} - \texttt{attention}(\{\boldsymbol{g}_i^{(1)}, \ldots, \boldsymbol{g}_i^{(N)}\}); \quad \boldsymbol{g}_i^{(j)} = \begin{bmatrix} \texttt{id}_i^{(j)} \\ d_i^{(j)} \end{bmatrix}$$

  - bi-directional encoding of grapheme info also useful

Attention-Based Confidence Scores [6]

- Attention mechanism over all the arcs
  - include distance between arcs
- Arc, a_i, of interest as query
  - arc as key
- Uses multi-head attention

- Options for sets of arcs to include
  - can make use of multiple lattices

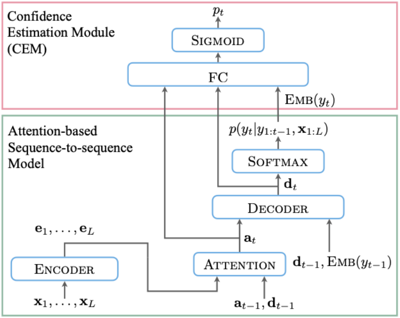


Confidence Score Performance

- Evaluation on Georgian converstaional telephone speech
  - RNN-based on: posteriors, word ID and durations
  - latticeRNN acts on confusion networks

Context	System	NCE	AUC
1-best	Decision Tree	0.2755	0.7112
	RNN	0.2911	0.7194
	Attention	0.2949	0.7209
CN	Lattice-RNN	0.2934	0.7185
	Attention	0.3001	0.7312
5 CNs	Attention	0.3035	0.7340

End-to-End Confidence Scores [5]



- Previous approaches rely on "rich" lattices (many arcs/paths)
  - can be challenging for sequence-to-sequence ASR models
- Alternative: use 1-best output and additional classifier with
  - attention ($\mathbf{a}_t$), decoder-state ($\mathbf{d}_t$), word embedding ($\mathtt{EMB}(y_t)$)

Course Outline

Course Outline

- Spoken Language Processing - Introduction
- Keyword spotting
  - practical based on KWS - available from moodle
  - based on BABEL/MATERIAL projects
- Spoken language assessment and learning (2 lectures)
  - based on ALTA Institute research
- Meeting and spoken document summarisation
  - Spotify podcast challenge system and transformer attention

- Spoken Langage Generation - Introduction
- Deep learning for speech synthesis (2 lectures)

[1] G. Evermann and P. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *Proc. of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2000.

[2] K. Kalgaonkar, C. Liu, Y. Gong, and K. Yao, "Estimating confidence scores on ASR results using recurrent neural networks," in *ICASSP*, 2015.

[3] F. Ladhak, A. Gandhe, M. Dreyer, L. Mathias, A. Rastrow, and B. Hoffmeister, "LatticeRNN: Recurrent neural networks over lattices," in *Interspeech*, 2016, pp. 695–699.

[4] Q. Li, P. Ness, A. Ragni, and M. J. F. Gales, "Bi-directional lattice recurrent neural networks for confidence estimation," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*, 2019, pp. 6755–6759. [Online]. Available: https://doi.org/10.1109/ICASSP.2019.8683488

[5] Q. Li, D. Qiu, Y. Zhang, B. Li, Y. He, P. C. Woodland, L. Cao, and T. Strohman, "Confidence estimation for attention-based sequence-to-sequence models for speech recognition," in *ICASSP*, 2015.

[6] A. Ragni, M. Gales, O. Rose, K. Knill, A. Kastanaos, Q. Li, and P. Ness, "Increasing context for estimating confidence scores in automatic speech recognition," *submitted IEEE Transactions Audio, Speech and Language Processing*, 2002.

[7] A. Ragni, Q. Li, M. J. F. Gales, and Y. Wang, "Confidence estimation and deletion prediction using bidirectional recurrent neural networks," in *2018 IEEE Spoken Language Technology Workshop, SLT 2018, Athens, Greece, December 18-21, 2018*, 2018, pp. 204–211. [Online]. Available: https://doi.org/10.1109/SLT.2018.8639678

[8] M. S. Seigel and P. C. Woodland, "Combining information sources for confidence estimation with CRF models," in *INTERSPEECH 2011, 12th Annual Conference of the International Speech Communication Association, Florence, Italy, August 27-31, 2011*, 2011, pp. 905–908.

[9] F. Wessel, R. Schlüter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary continuous speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.