

# Selected Aspects of the Cambridge Arabic Speech-to-Text Systems

F. Diehl

10 March, 2011



Cambridge University Engineering Department

## Contributors

Frank Diehl

Phil Woodland

Mark Gales

Junho Park

Marcus Tomalin



# Contents

- Arabic LVCSR – what is the challenge?
- Arabic language processing within GALE
- MADA morphological decomposition
- Multilayer perceptron acoustic features
- Boosted MMI for acoustic model training
- Neural network language models



## Arabic LVCSR – what is the challenge?



## Arabic LVCSR – what is the challenge?

- Issues with Arabic STT:
  - Arabic is a morphologically complex language
    - ⇒ high Out-Of-Vocabulary (OOV) rates
    - ⇒ sparse LM training data
  - Modern Standard Arabic (MSA) is written without short vowels
    - ⇒ difficult dictionary generation



## Arabic LVCSR – what is the challenge?

- Romanisation / vowelisation

والاعتراف بالخطا فضيلة

wAlAetrAf      bAlxTA      fDylp

wAlAiEtirAfa      bAlxaTaA      faDiylapF



## Arabic LVCSR – what is the challenge?

- Morphological decomposition / diacritization

System	Example Sentence		
Word	<i>wktb</i>		<i>AlAyrAnY</i>
Morphemes	<i>w+</i>	<i>ktb</i>	<i>AlMyrAny</i>
Translation	<i>and</i>	<i>he-writes</i>	<i>the Iranian</i>



## Arabic LVCSR – what is the challenge?

- Proposed solutions:
  - System combination → complementary systems needed
  - Word-based and morpheme-based systems
  - PLP and PLP+MLP acoustic features
  - MPE and BMMI trained systems
  - Neural network LMs for lattice re-scoring



## Global Autonomous Language Exploitation (GALE)

- **DARPA** funded research program (Thanks to DARPA!)  
(<http://www.darpa.mil/ipto/programs/gale/gale.asp>)
- Goal:
  - ”...develop and apply computer software technologies to absorb, **analyse** and interpret **huge volumes of speech** and text in **multiple languages... delivering** pertinent, consolidated **information...in easy-to-understand forms to...monolingual English-speaking analysts...**”
- GALE consists of three major engines:
  - **Transcription**
  - Translation
  - Distillation



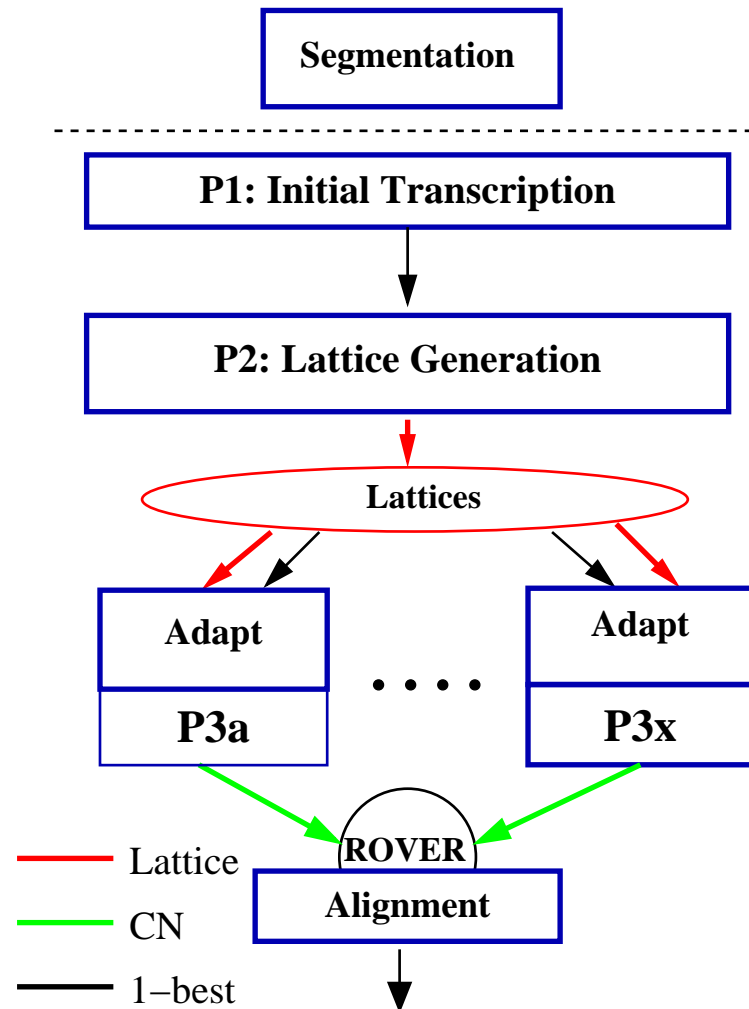
## CUED within the GALE Project

- Three competing groups: AGILE, ROSETTA, NIGHTINGALE
- The **AGILE** team: BBN, LIMSI, ..., and **CUED**
- The objective of AGILE consists in:
  - Transcription of Arabic/Chinese broadcast news/conversation data
  - Translation of the transcribed data to English
- **CUED** contributes to the **STT** and the **SMT** work packages
- Final STT result: **system combination** CUED, LIMSI, and BBN



# CUED System Overview

- Decoding Strategy



- Multi-pass combination framework

- P1 Used to generate **initial** hypothesis
  - \* Word-based or morpheme-based branch
- P1 Hypothesis used for rapid **adaptation**
  - \* LSLR, diagonal variance transforms
- P2 lattices generated for re-scoring
  - \* **Apply complex LMs**
- P3 Adaptation
  - \* **Acoustic model re-scoring**
  - \* 1-best CMLLR
  - \* Lattice-based MLLR
- CN decoding / ROVER combination



## Overview Standard System

- **Broadcast News/Conversational transcription tasks**
  - Approx. 1850 hours of acoustic training data
- **Front-end parametrisation**
  - Basic front-end: PLP-features + energy (up to third differential)
  - 39-dimensional feature stream after HLDA
- **Acoustic modelling**
  - Graphemic / phonetic modelling (explicit short vowel modelling)
  - 9k tied states, 324k diagonal Gaussian mixtures
  - MPE trained gender dependent models
- **Language modelling**
  - 1.2G tokens within 22 sources
  - 350k word list
  - Interpolation of 22 component n-gram LMs
  - 3-gram LM for lattice generation, 4-gram LM for lattice re-scoring



## MADA Morphological Decomposition



## MADA Morphological Decomposition

- Arabic is a highly inflected agglutinative language
  - Many lexical variants per root word → high OOV rates  
(e.g. 59k English dict: ~ 0.5% OOV, 350k Arabic dict: ~ 1.0% OOV)
  - Sparse language model training material
- MADA (Habash'06) morphological decomposition alleviates these problems by
  - Splitting prefixes from stems:  $IAIREb \rightarrow I+ AIREb$
  - Stem normalisation:  $AlAyrAnY \rightarrow AlMyrAnY$
- MADA 'D2 DIAC' decomposition scheme used:
  - Separates five proclitics ( $l, b, k, w, f$ ) from associated word roots
  - Provides stem normalisation
  - Provides vowelisation (diacritization)



## MADA Morphological Decomposition

- Stem normalisation makes the morpheme-to-word conversion a non-trivial task:
  - No simple prefix-stem gluing possible → morpheme-to-word mapping
  - MADA is context sensitive → a unigram mapping is suboptimal
- Morpheme-to-word mapping viewed as Machine Translation (MT) task:
  - Translate the morpheme sequence ( $S$ ) to the word sequences ( $T$ )
  - Linear alignment ( $T, S$ ) between source and target token sequences exists

⇒ N-gram SMT approach

$$p(T, S) = \prod_{k=1}^K p((t, s)_k | (t, s)_{k-1}, \dots, (t, s)_{k-N+1})$$

- Decoding: likelihood maximisation of  $p(T|S)$  w.r.t..  $T = (t_1, t_2, \dots, t_K)$ ,



## MADA Morphological Decomposition

- Dictionary generation:
  - Arabic is commonly written **without diacritic markers** indicating short vowels
  - Many phonetic word forms for one graphemic word form

Graphemic Form	Vowelised Form	Meaning
<i>ktb</i>	<i>kataba</i> <i>kutiba</i> <i>kitab</i> <i>kutubun</i> <i>kutubu</i>	he writes it is written/fated/destined book (indefinite) books (indefinite) books (definite)

- Use of graphemic and phonetic systems in combination
  - ⇒ Graphemic system: good coverage and robustness
  - ⇒ Phonetic system: better accuracy but some words not available



## MADA Morphological Decomposition

- OOV rates for the word-based and MADA-based word lists

Wordlist		Testset		
Type	Size	dev07	eval07	dev08
Word-pho	260k	2.68	3.39	2.03
Word-gra	350k	1.19	1.26	1.14
MADA-pho	171k	1.40	1.33	1.38
MADA-gra	331k	0.79	0.54	0.63

- ⇒ MADA: OOV rate reduction by nearly a factor of 2
- ⇒ Improved coverage of graphemic vs phonetic systems



## MADA Morphological Decomposition

- Graphemic / phonetic / word-based / morpheme-based modelling contrast

System	Configuration	Testset		
	mada	dev07	eval07	dev08
<b>G</b>	-	13.2	14.1	14.9
<b>G</b> <sub>Ma</sub>	✓	12.6	13.6	14.1
<b>V</b>	-	11.6	13.2	14.2
<b>V</b> <sub>Ma</sub>	✓	11.0	12.4	13.5
<b>ROVER</b>	<b>V</b> <sub>Ma</sub> ⊕ <b>G</b> <sub>Ma</sub>	11.0	11.9	12.9
<b>ROVER</b>	<b>V</b> <sub>Ma</sub> ⊕ <b>V</b>	10.6	11.9	13.0
<b>ROVER</b>	<b>V</b> <sub>Ma</sub> ⊕ <b>V</b> ⊕ <b>G</b> <sub>Ma</sub>	10.4	11.5	12.4

- ⇒ MADA provides WER reductions of typically 0.5% absolute
- ⇒ Graphemic / phonetic combination: WER reductions of 0.0-0.6% absolute
- ⇒ Morpheme / word combination: WER reductions of 0.4-0.5% absolute
- ⇒ 3-way combination: WER reductions of 0.6-0.9% absolute



## Multilayer Perceptron Acoustic Features



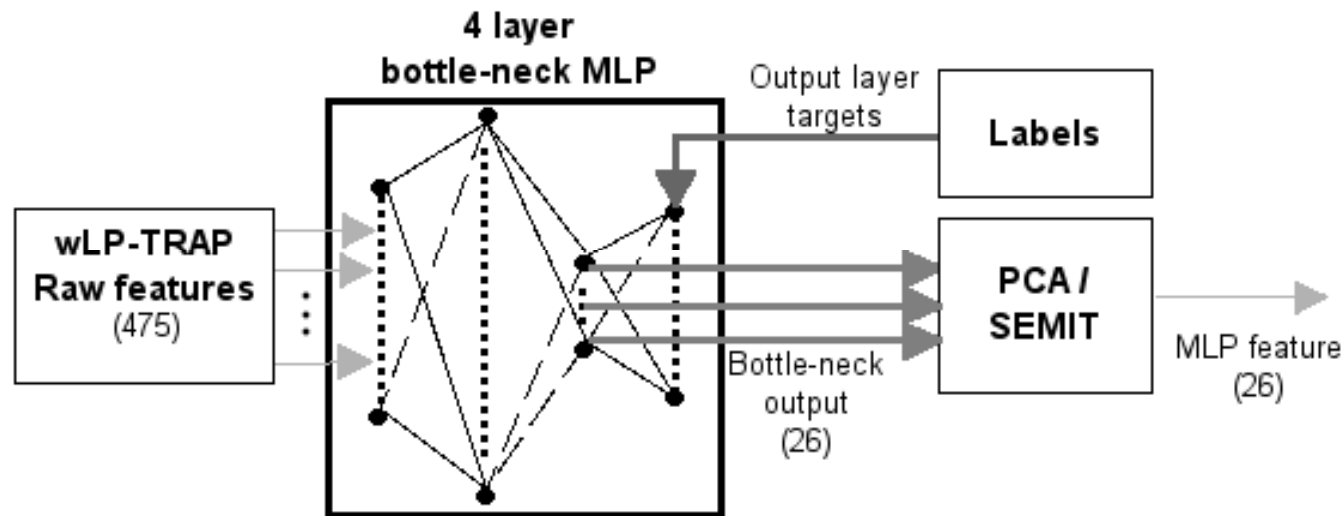
## Multilayer Perceptron Acoustic Features

- MLP features:
  - Obtained from an MLP trained to classify phonetic or sub-phonetic units
  - Provide explicit phonetic knowledge to the feature stream
  - Constitute an additional source of diversity



## Multilayer Perceptron Acoustic Features

- Neural network layout



- Bottle-neck design
  - No dimensionality reduction needed
  - No domain transform  $[0, 1] \rightarrow ] - \infty, \infty[$  needed



## Multilayer Perceptron Acoustic Features

- Configuration
  - 475x3500x26x39 nodes
  - Input: wLP-TRAP features, long timespan features (1s)
  - Targets: 39 phonemes, hard targets, inclusive short vowels
  - Training material: 1350 hours
  - Training criterion: cross-entropy minimisation
- Test accuracy
  - 65.6% phone accuracy

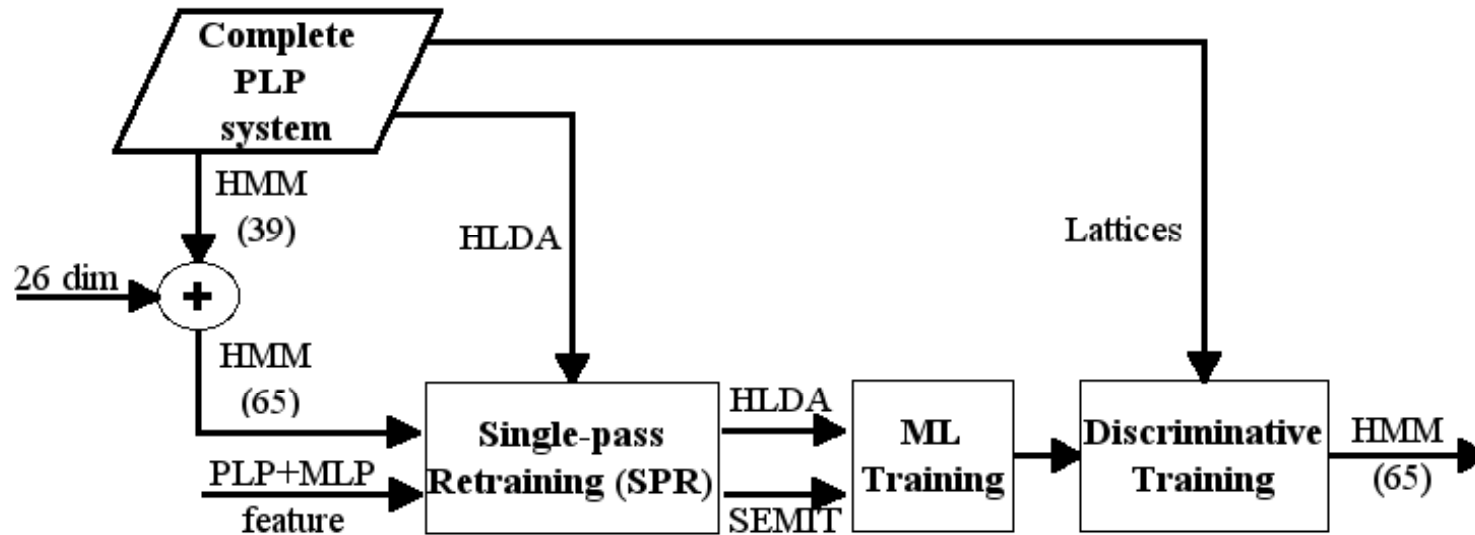


## Multilayer Perceptron Acoustic Features

- **TANDEM connectionist approach**
  - Concatenate the 39 dim PLP-features with the 26 dim MLP-features
- **Rapid system build avoids the lattice re-generation for MPE/BMMI training**
  - Use the decision tree, HLDA transform, and lattices from the PLP system.
  - Extend the PLP to a PLP+MLP model by Single-Pass-Retraining
  - Estimate a semi-tied transform for the MLP features (decorrelation)
  - Apply the HLDA transform to the PLP-features and the semi-tied transform to the MLP-features (block diagonal transform)
  - Refine the models using Baum-Welch training
  - Discriminative training with the original lattices from the PLP system



## Multilayer Perceptron Acoustic Features



## Multilayer Perceptron Acoustic Features

- Evaluation of the 'fast' system build
  - Performed on a reduced training set: 172 hours of data
  - 'unadapted' decoding using graphemic models
  - Nomenclature:
    - \* 'full' : complete lattice rebuild
    - \* 'semi-fast': PLP-based lattices are re-phonemarked
    - \* 'fast' : PLP-based lattices are used directly

System			WER	
Training	Front End	Lattices	dev07	dev08
<b>MPE</b>	PLP	full	20.5	22.8
	PLP+MLP	fast	17.7	20.0
		semi-fast	17.7	19.9
		full	17.7	19.9



## Multilayer Perceptron Acoustic Features

- System evaluation results
  - Compare PLP-features with PLP+MLP-features

System	Configuration				Testset		
	plp+mlp	wrd	mada	bmmi	dev07	eval07	dev08
$\mathbf{V}$	-	✓	-	-	11.4	12.9	14.0
$\mathbf{V}_{Ta}$	✓	✓	-	-	11.3	12.4	13.7
<b>ROVER</b>	$\mathbf{V} \oplus \mathbf{V}_{Ta}$				11.1	12.2	13.4
$\mathbf{V}_{Ma}$	-	-	✓	-	11.0	12.4	13.5
$\mathbf{V}_{TaMa}$	✓	-	✓	-	10.8	11.7	12.8
<b>ROVER</b>	$\mathbf{V}_{Ma} \oplus \mathbf{V}_{TaMa}$				10.4	11.7	12.7

⇒ PLP+MLP-features outperform PLP-features by 0.1-0.7% in abs. WER  
 ⇒ Combination gains reach 0.0-0.4% in abs. WER



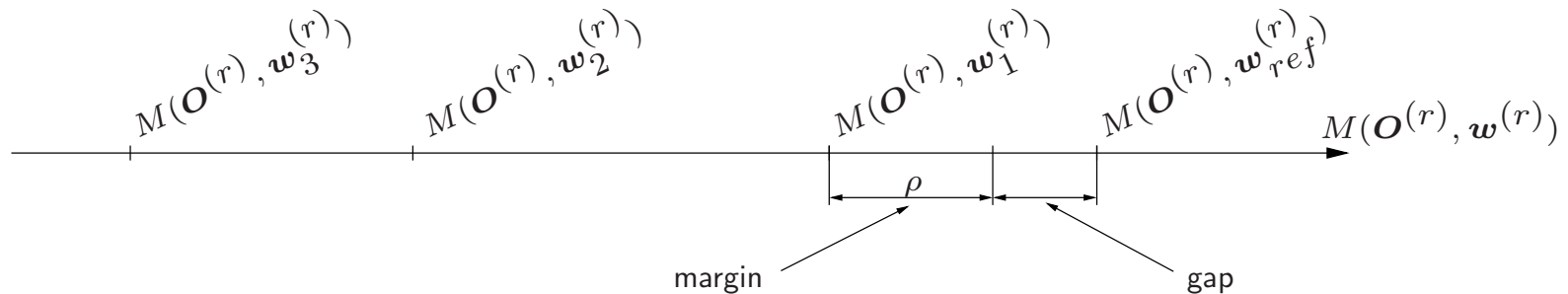
# Boosted MMI for Acoustic Model Training



## Boosted MMI for Acoustic Model Training

- Boosted MMI, a large margin approach (Povey '08)
  - **Margin size constraint** between the reference and competing transcriptions

$$M(\mathbf{O}^{(r)}, \mathbf{w}_{ref}^{(r)}) - \max_{\mathbf{w} \neq \mathbf{w}_{ref}^{(r)}} \{M(\mathbf{O}^{(r)}, \mathbf{w}) + \rho\} \geq 0$$



- **Train** on all samples which **violate the constraint** ('negative gap')



## Boosted MMI for Acoustic Model Training

- Boosted MMI, a large margin approach
  - Chose **margin** which **scales linearly** with an **error measure** on the transcription

$$\rho = \alpha \mathcal{D}(\mathbf{w}, \mathbf{w}_{ref}^{(r)})$$

- With the **'hinge' function**  $[f(x)]^- = \max(0, -f(x))$  and the **softmax inequality**  $\max_x g(x) \leq \log \sum_x e^{g(x)}$  the objective function is defined as:

$$\begin{aligned} \mathcal{F}(\boldsymbol{\lambda}) &= \frac{1}{R} \sum_{r=1}^R \left[ M(\mathbf{O}^{(r)}, \mathbf{w}_{ref}^{(r)}; \boldsymbol{\lambda}) - \max_{\mathbf{w}} \left\{ M(\mathbf{O}^{(r)}, \mathbf{w}; \boldsymbol{\lambda}) + \alpha \mathcal{D}(\mathbf{w}, \mathbf{w}_{ref}^{(r)}) \right\} \right]^- \\ &\leq \frac{1}{R} \sum_{r=1}^R \left[ M(\mathbf{O}^{(r)}, \mathbf{w}_{ref}^{(r)}; \boldsymbol{\lambda}) - \log \sum_{\mathbf{w}} e^{\left\{ M(\mathbf{O}^{(r)}, \mathbf{w}; \boldsymbol{\lambda}) + \alpha \mathcal{D}(\mathbf{w}, \mathbf{w}_{ref}^{(r)}) \right\}} \right]^- \end{aligned}$$

- $\mathcal{F}(\boldsymbol{\lambda})$  needs to be maximised w.r.t.  $\lambda$



## Boosted MMI for Acoustic Model Training

- Evaluate the softmax approximation first:

$$\log \sum_{\mathbf{w}} e^{\{M(\mathbf{O}^{(r)}, \mathbf{w}) + \alpha \mathcal{D}(\mathbf{w}, \mathbf{w}_{ref}^{(r)})\}} \geq M(\mathbf{O}^{(r)}, \mathbf{w}_{ref}^{(r)})$$

- The ‘hinge’ function argument is always negative  $\rightarrow$  sum over **all data**  $R$

- Evaluate the ‘hinge’ function first:

- The summation reduces to the **data violating the margin constraint**
- $R$  is given by

$$\mathcal{R} = \left\{ r \mid \max_{\mathbf{w}} \mathcal{S}(\mathbf{w}, \mathbf{w}_{ref}^{(r)}) > \mathcal{S}(\mathbf{w}_{ref}^{(r)}, \mathbf{w}_{ref}^{(r)}) \right\}$$

where

$$\mathcal{S}(\mathbf{w}, \mathbf{w}_{ref}^{(r)}) = M(\mathbf{O}^{(r)}, \mathbf{w}) + \alpha \mathcal{D}(\mathbf{w}, \mathbf{w}_{ref}^{(r)})$$



## Boosted MMI for Acoustic Model Training

- Boosted MMI

- With  $M(\mathbf{O}^{(r)}, \mathbf{w}) = \log(p(\mathbf{O}^{(r)} | \mathbf{w}; \boldsymbol{\lambda})^\kappa P(\mathbf{w}))$
- And  $\mathcal{D}(\mathbf{w}, \mathbf{w}_{ref}^{(r)}) = -\mathcal{A}(\mathbf{w}', \mathbf{w}_{ref}^{(r)})$  (the **negative phone accuracy**) one gets

$$\mathcal{F}_{\text{BMMI}}(\boldsymbol{\lambda}) = \sum_{r=1}^R \log \frac{p(\mathbf{O}^{(r)} | \mathbf{w}_{ref}^{(r)}; \boldsymbol{\lambda})^\kappa P(\mathbf{w}_{ref}^{(r)})}{\sum_{\mathbf{w}'} p(\mathbf{O}^{(r)} | \mathbf{w}'; \boldsymbol{\lambda})^\kappa P(\mathbf{w}') e^{-\alpha \mathcal{A}(\mathbf{w}', \mathbf{w}_{ref}^{(r)})}}$$

- **Penalises good** hypothesises  $\rightarrow$  **more weight** for **worse** hypothesises
- Works similar as acoustic scale  $\kappa$  which lifts weak acoustic likelihoods



## Boosted MMI for Acoustic Model Training

- Implementation issues
  - Use **approximate phone accuracy** (sum of phone-arc specific phone accuracies)
  - **Lattice implementation** possible ( subtract the phone-arc specific phone accuracy from the phone-arc specific acoustic likelihood)
- Runtime issues
  - Tuning of **boosting factor**  $\alpha$  needed, best performance for  $\alpha = 2.0$
  - **Smoothing** of the statistics by a **dynamic ML prior**
  - The **reduced training set** always resulted in a **performance loss** → not used



## Boosted MMI for Acoustic Model Training

- Comparison to MPE

- BMMI:

$$\mathcal{F}_{\text{BMMI}}(\boldsymbol{\lambda}) = \sum_{r=1}^R \log \frac{p(\mathbf{O}^{(r)} | \mathbf{w}_{ref}^{(r)}; \boldsymbol{\lambda})^\kappa P(\mathbf{w}_{ref}^{(r)})}{\sum_{\mathbf{w}'} p(\mathbf{O}^{(r)} | \mathbf{w}'; \boldsymbol{\lambda})^\kappa P(\mathbf{w}') e^{-\alpha \mathcal{A}(\mathbf{w}', \mathbf{w}_{ref}^{(r)})}}$$

- MPE:

$$\mathcal{F}_{\text{MPE}}(\boldsymbol{\lambda}) = \sum_{r=1}^R \frac{\sum_{\mathbf{w}} p(\mathbf{O}^{(r)} | \mathbf{w}; \boldsymbol{\lambda})^\kappa P(\mathbf{w}) \mathcal{A}(\mathbf{w}, \mathbf{w}_{ref}^{(r)})}{\sum_{\mathbf{w}'} p(\mathbf{O}^{(r)} | \mathbf{w}'; \boldsymbol{\lambda})^\kappa P(\mathbf{w}')}$$

- The **phone accuracy** plays a central role in both objective function



## Boosted MMI for Acoustic Model Training

- System evaluation results

- Compare BMMI with MPE
- Are both methods complementary (both use the phone accuracy) ?

System	Configuration				Testset		
	plp+mlp	wrd	mada	bmmi	dev07	eval07	dev08
$\mathbf{V}$	-	✓	-	-	11.4	12.9	14.0
$\mathbf{V}_{Bm}$	-	✓	-	✓	11.2	12.6	13.5
<b>ROVER</b>	$\mathbf{V} \oplus \mathbf{V}_{Bm}$				11.0	12.3	13.4
$\mathbf{V}_{TaMa}$	✓	-	✓	-	10.8	11.7	12.8
$\mathbf{V}_{TaMaBm}$	✓	-	✓	✓	10.7	11.6	12.3
<b>ROVER</b>	$\mathbf{V}_{TaMa} \oplus \mathbf{V}_{TaMaBm}$				10.5	11.5	12.3

⇒ BMMI outperforms MPE by 0.1%-0.5% in absolute WER

⇒ Combination gains reach 0.1%-0.3% in absolute WER



# Neural Network Language Models



## Neural Network Language Model

- Problems of standard N-gram LMs:

$$P(w) = \prod_{k=1}^T P(w_k | w_{k-1}, \dots, w_{k-(N-1)})$$

- **Unstructured model**  $\Rightarrow$  direct estimation (counting) of N-gram probabilities
  - **No interpolation** of unseen model parameters  $\Rightarrow$  back-off strategies
  - **Difficult LM adaptation**, many parameters little data
- **Neural network LMs offer:**
    - **A structured modelling** approach with a continuous feature space
    - **Parameter interpolation** possible  $\Rightarrow$  no back-off strategy needed
    - **Copes better** with the **data sparsity** problem of standard N-gram LMs
    - Provides a way for **LM adaptation**



## Neural Network Language Model

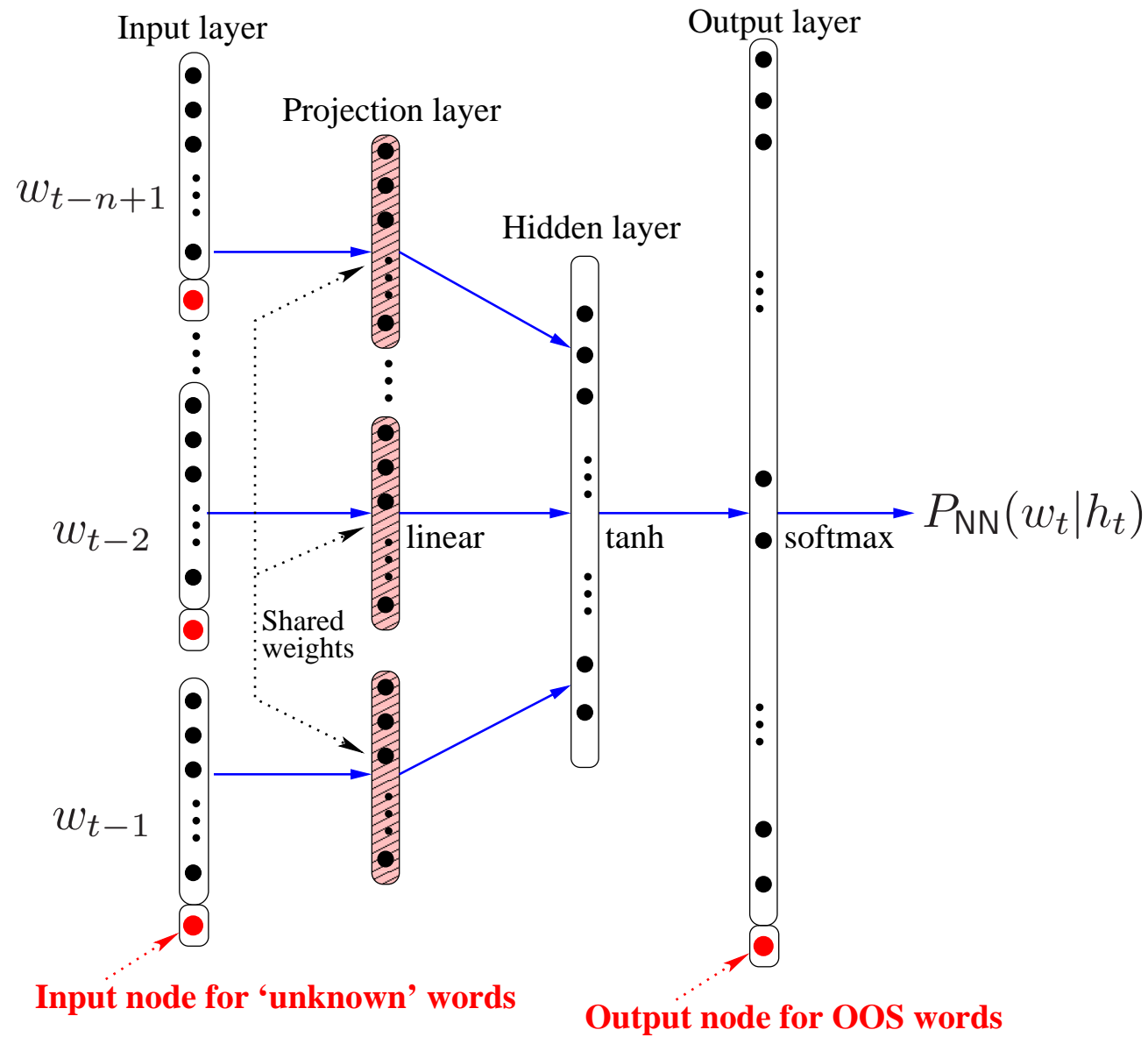
- General description of NNLM
  - Projects a set of contexts  $h_t = w_{t-1} \dots w_{t-n+1}$  onto a continuous space
  - Calculates the LM probability for each word given a history,  $P(w_t = i | h_t)$ 
    - \* Provides a full context span probability distribution for all words
    - \* No back-off strategy to lower order  $n$ -gram LMs needed



## Neural Network Language Model

- Architecture of network with OOS node
  - Fully-connected multi-layer perceptron (MLP) structure
  - Use shortlists for context and target vocabulary
    - ⇒ Reduce computational load
  - The inputs to the network ( $N - 1 * 100k$ )
    - ⇒ Indices of the  $n - 1$  history words in the input vocabulary  $V_{in}$
  - Between input and output layers ( $100 \times 400$ )
    - ⇒ Two hidden layers for projection and probability estimation
  - The outputs of the network ( $20k$ )
    - ⇒ Posterior probabilities of all words following a given history
- 15M words of acoustic transcriptions for training





## Neural Network Language Model

- Interpolation with N-gram LM
  - Smaller dictionary coverage than N-gram LM  $\Rightarrow$  LM interpolation

$$P(w_t|h_t) = \lambda P_{\text{NG}}(w_t|h_t) + (1 - \lambda) \tilde{P}_{\text{NN}}(w_t|h_t)$$

- $\lambda$  is the interpolation weight assigned to N-gram distribution  $P_{\text{NG}}(\cdot)$
- Tuning of the interpolation weight  $\lambda$ 
  - $\Rightarrow$   $\lambda$  was tuned based on WER results
  - $\Rightarrow$  Best performance was obtained for  $\lambda = 0.5$



## Neural Network Language Model

- Probability normalisation

$$\tilde{P}_{\text{NN}}(w_t|h_t) = \begin{cases} P_{\text{NN}}(w_t|h_t) & w_t \in V_{\text{sl}} \\ \beta_{\text{S}}(w_t|h_t)P_{\text{NN}}(w_{\text{oos}}|h_t) & \text{otherwise} \end{cases}$$

$$\beta_{\text{S}}(w_t|h_t) = \frac{P_{\text{NG}}(w_t|h_t)}{\sum_{\tilde{w}_t \notin V_{\text{sl}}} P_{\text{NG}}(\tilde{w}_t|h_t)}$$

⇒ Very expensive in decoding time

- Approximated normalization

$$\tilde{P}_{\text{NN}}(w_t|h_t) = \begin{cases} P_{\text{NN}}(w_t|h_t) & w_t \in V_{\text{sl}} \\ P_{\text{NG}}(w_t|h_t) & \text{otherwise} \end{cases}$$



## Neural Network Language Model

- System evaluation results

LM	WER		
	dev07	eval07	dev08
NG	11.0	12.4	13.5
NG + NN.OOS	10.9	12.3	13.1

– Single branch phonetic system

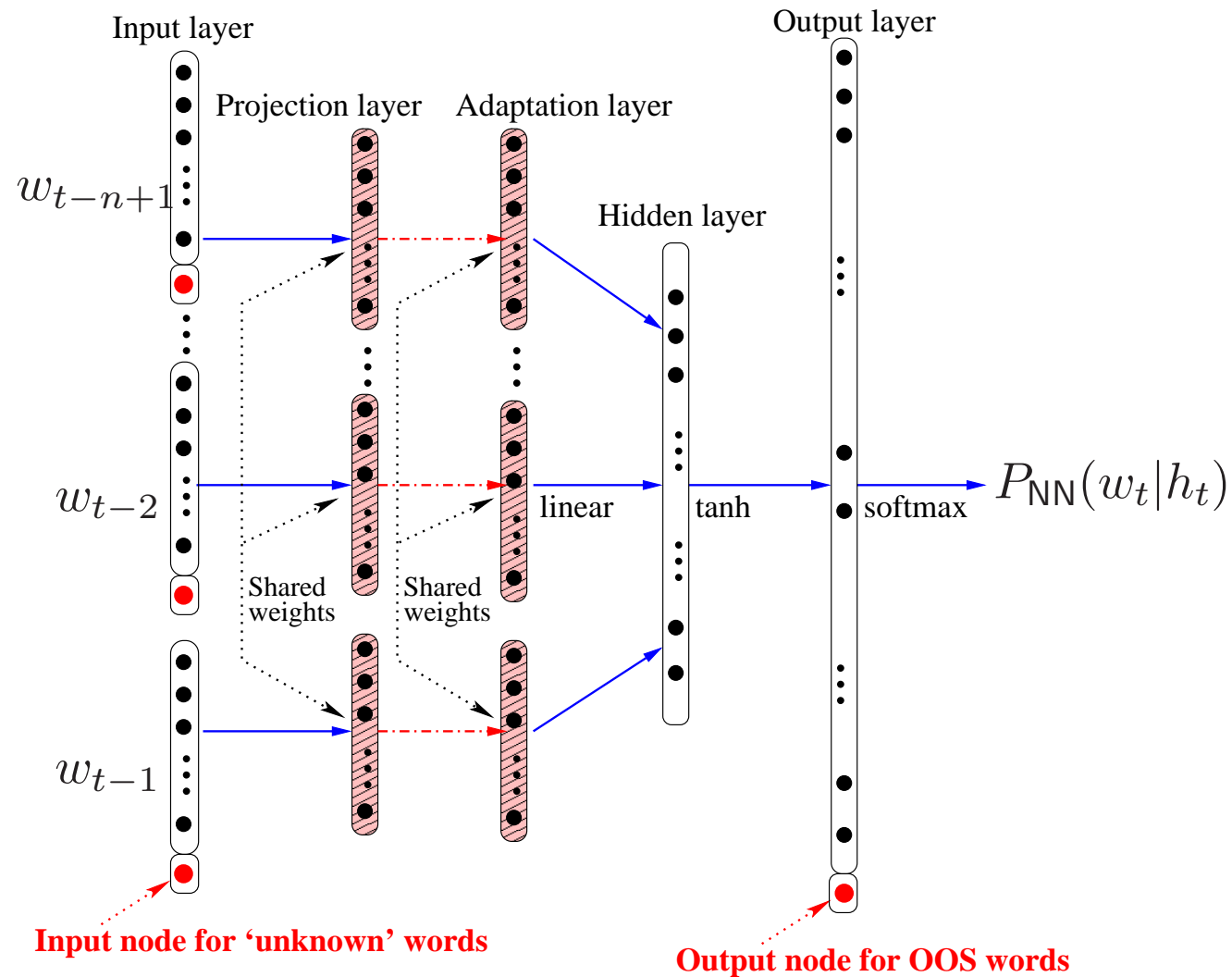
⇒ WER reductions of 0.1%-0.4% absolute over the N-gram LM



## Neural Network Language Model

- LM adaptation to a particular domain or task
  - Improve robustness to varying styles or tasks
  - Data sparsity issue
    - ⇒ Impractical adaptation of N-gram prob. (limited amounts of data)
  - Continuous space representation in NNLM
    - ⇒ Continuous feature space allows parameter interpolation
    - ⇒ Stronger generalisation ability
    - ⇒ Cascading an additional adaptation layer to the projection layer
  - Estimation of the adaptation layer weights
    - ⇒ A linear transform of the projection layer with bias
    - ⇒ Supervision: 1-best result from unadapted NNLM/ $n$ -gram
    - ⇒ CV with 20% of the data





## Neural Network Language Model

- System evaluation results

LM	WER		
	dev07	eval07	dev08
NG	11.0	12.4	13.5
NG + NN.OOS	10.9	12.3	13.1
NG + NN.OOS.adapt	10.9	12.2	13.1

– Single branch phonetic system

⇒ Small improvements by NNLM adaptation



## System Combination

- 4-way ROVER results

System	Configuration				Testset		
	plp+mlp	wrd	mada	bmmi	dev07	eval07	dev08
$\mathbf{G}_{Ta}$	✓	✓	-	-	12.6	13.4	14.4
$\mathbf{G}_{MaBm}$	-	-	✓	✓	12.5	13.5	14.9
$\mathbf{G}_{Ma}$	-	-	✓	-	12.8	13.6	14.3
$\mathbf{V}_{Ma}$	-	-	✓	-	11.0	12.4	13.5
$\mathbf{V}_{Bm}$	-	✓	-	✓	11.2	12.6	13.5
$\mathbf{V}_{TaMaBm}$	✓	-	✓	✓	10.7	11.6	12.3
<b>ROVER</b>	$\mathbf{G}_{Ta} \oplus \mathbf{G}_{Ma} \oplus \mathbf{V}_{Bm} \oplus \mathbf{V}_{TaMaBm}$				9.9	10.8	11.7
<b>ROVER</b>	$\mathbf{G}_{Ta} \oplus \mathbf{G}_{MaBm} \oplus \mathbf{V}_{Bm} \oplus \mathbf{V}_{TaMaBm}$				9.9	10.9	11.7
<b>ROVER</b>	$\mathbf{G}_{Ta} \oplus \mathbf{G}_{Ma} \oplus \mathbf{V}_{Ma} \oplus \mathbf{V}_{TaMaBm}$				10.1	11.0	12.0

⇒ Best ROVER results demand complementary system



## Conclusions

- MADA morphological decomposition
  - Significant reduction of the OOV problem
  - Graphemic / phonetic model combination alleviates the dictionary generation problem
  - Morpheme / word combination is very effective
- BMMI
  - Tends to outperform MPE
  - Combines well with MPE



## Conclusions

- MLP-features
  - PLP+MLP-features outperform PLP-features
  - Additional gains by PLP+MLP-PLP-feature combination
- NN LMs
  - NN LMs alleviate the data sparsity problem
  - NN LMs provide a method for LM adaptation
- ROVER combination
  - Combining complementary systems is more effective than combining the best systems



Thank you for your attention

