Available online at www.sciencedirect.com

ScienceDirect

Computer Speech and Language 28 (2014) 1298–1316

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

Paraphrastic language models☆

X. Liu*, M.J.F. Gales, P.C. Woodland

Cambridge University, Engineering Department, Trumpington Street, Cambridge CB2 1PZ, England

Received 31 May 2013; received in revised form 28 March 2014; accepted 7 April 2014
Available online 30 April 2014

Abstract

Natural languages are known for their expressive richness. Many sentences can be used to represent the same underlying meaning. Only modelling the observed surface word sequence can result in poor context coverage and generalization, for example, when using *n*-gram language models (LMs). This paper proposes a novel form of language model, the paraphrastic LM, that addresses these issues. A phrase level paraphrase model statistically learned from standard text data with no semantic annotation is used to generate multiple paraphrase variants. LM probabilities are then estimated by maximizing their marginal probability. Multi-level language models estimated at both the word level and the phrase level are combined. An efficient weighted finite state transducer (WFST) based paraphrase generation approach is also presented. Significant error rate reductions of 0.5–0.6% absolute were obtained over the baseline *n*-gram LMs on two state-of-the-art recognition tasks for English conversational telephone speech and Mandarin Chinese broadcast speech using a paraphrastic multi-level LM modelling both word and phrase sequences. When it is further combined with word and phrase level feed-forward neural network LMs, a significant error rate reduction of 0.9% absolute (9% relative) and 0.5% absolute (5% relative) were obtained over the baseline *n*-gram and neural network LMs respectively.

Keywords: Language modelling; Paraphrase; Speech recognition

1. Introduction

Natural languages are known to have layered structures, a hidden and deeper structure that represents the meaning and core semantic relations within a sentence, and a surface form found in normal written texts or spoken language, as formulated in linguistic theories such as generative grammar Chomsky (1966), Jackendoff (1974). The mapping from the meaning to the observed surface form involves a natural language generation process. As multiple surface realizations can be used to convey identical or similar semantic information, this mapping is often one-to-many. These different surface realizations are paraphrastic to one another. They were created by using different syntactic, lexical and morphological rules in the generation process. Functionally these paraphrase variants represent different styles, dialects or other speaker specific characteristics. Due to their presence, only modelling the observed surface word sequence can result in poor context coverage, for example, when using standard *n*-gram language models (LMs).

☆ This paper has been recommended for acceptance by 'Riley Michael'.

* Corresponding author. Tel.: +44 1223 766512; fax: +44 1223 332662.
E-mail addresses: xl207@cam.ac.uk, xl207@eng.ac.uk (X. Liu), mjfg@eng.cam.ac.uk (M.J.F. Gales), pcw@eng.cam.ac.uk (P.C. Woodland).

http://dx.doi.org/10.1016/j.csl.2014.04.004
0885-2308/

One approach to handle this problem requires directly modelling paraphrase variants when constructing the LM. As alternative expressions of the same meaning are now considered, the resulting language model's context coverage and generalization performance is expected to improve. Along this line, the use of word level synonym features Cao et al. (2005), Hoberman and Rosenfeld (2002), Jelinek et al. (1990), Kneser and Peters (1997) has been investigated in early research for n-gram and class n-gram based Brown et al. (1992) language models. However, there are two issues associated with these existing approaches. First, the paraphrastic relationship between longer span syntactic structures, such as phrases, is largely ignored. A more general form of modelling that can also capture a higher level and longer span paraphrase mapping should be more effective. Second, previous research focused on using manually derived expert semantic labelling provided by resources such as WordNet Fellbaum (1998). As manual annotation is usually very expensive to produce, these methods cannot be applied to large corpora or languages without suitable WordNet-type resources. Hence, automatic, statistical paraphrase induction and extraction techniques are required.

In order to address these issues, this paper presents a novel form of language model, the paraphrastic language model (PLM). It provides a highly flexible and general form of paraphrase modelling that can be used at either the word, phrase or sentence level. The paraphrastic relationship between longer span syntactic structures can thus be effectively captured. A phrase level paraphrase model statistically learned from standard text data is used to generate multiple paraphrase variants for the training data. Language model probabilities are then estimated by maximizing the marginal probability of these variants. By linking language generation and modelling, paraphrastic LMs exploit an intuitive and interpretable parameter smoothing scheme to improve generalization performance. In order to leverage the complementary characteristics of paraphrastic LMs and feed-forward neural network LMs (NNLMs) Bengio et al. (2003), Kuo et al. (2012), Le et al. (2013), Park et al. (2010), Schwenk (2007), the combination between the two is also investigated.

This paper extends previous research summarized in Liu et al. (2012b, 2013c). A more complete study of using paraphrastic language models for speech recognition is presented. Various important aspects of this work, including the theory and implementation of the statistical paraphrase learning algorithm, the generation of paraphrase lattices and the construction of phrase and multi-level paraphrastic LMs, are covered in detail in this paper. These are further augmented by a full set of experimental results presented to demonstrate the advantages of paraphrastic LMs over existing modelling methods. This paper shows the applicability of paraphrastic LMs to multiple languages and genres, the scaling behaviour on varying amounts of training data, and their complementarity to other established language modelling techniques.

The rest of the paper is organized as follows. Paraphrastic language models are introduced in Section 2. A statistical n-gram phrase pair based paraphrase extraction scheme is presented in Section 3. Paraphrase lattice generation using a weighted finite state transducer (WFST) approach is described in Section 4. The estimation of paraphrastic LMs is presented in Section 5. The combination between paraphrastic LMs and feed-forward neural network LMs is proposed in Section 6. In Section 7 a range of paraphrastic LMs are evaluated on two state-of-the-art speech recognition tasks for English conversational telephone speech and Chinese broadcast speech respectively. Section 8 is the conclusion and possible future work.

2. Paraphrastic language models

As discussed above, in order to capture the paraphrase mapping between longer span syntactic structures, a more general form of modelling is required. To address this issue, the particular type of LMs proposed in this paper can flexibly model paraphrastic relationships at the word, phrase and sentence level. As LM probabilities are estimated in the paraphrased domain, they are referred to as *paraphrastic language models* (PLMs) in this paper. For a *surface word sequence* $\mathcal{W} = < w_1, w_2, \ldots, w_i, \ldots, w_L >$ of L words in the training data, for example, "*And I generally prefer*", rather than maximizing the surface word sequence log-probability $\ln P(\mathcal{W})$ as for conventional LMs, the marginal probability over its *paraphrase variant sequences*, $\{\mathcal{W}'\}$, such as "*And I just like*" or "*I mean I want*", is maximized Liu et al. (2012b, 2013c),

$$\mathcal{F}(\mathcal{W}) = \ln \left(\sum_{\boldsymbol{\psi}, \boldsymbol{\psi}', \mathcal{W}'} P(\mathcal{W}|\boldsymbol{\psi}) P(\boldsymbol{\psi}|\boldsymbol{\psi}') P(\boldsymbol{\psi}'|\mathcal{W}') P_{\mathsf{PLM}}(\mathcal{W}') \right) \tag{1}$$

where

- $P_{\text{PLM}}(\mathcal{W}')$ is the paraphrastic LM probability to be estimated;
- $P(\boldsymbol{\psi}'|\mathcal{W}')$ is a word to phrase segmentation model assigning the probability of a phrase level segmentation, $\boldsymbol{\psi}' =< v'_1, v'_2, \ldots, v'_K >$ of K phrases, given a paraphrase word sequence $\mathcal{W}' =< w'_1, w'_2, \ldots, w'_i, \ldots, w'_{L'} >$ of L' words in total.
- $P(\boldsymbol{\psi}|\boldsymbol{\psi}') = \prod_i P(v_i|v'_i)$ uses a phrase to phrase paraphrase model to compute probability of a phrase sequence $\boldsymbol{\psi} =< v_1, v_2, \ldots, v_i, \ldots, v_K >$ being paraphrastic to another one $\boldsymbol{\psi}' =< v'_1, v'_2, \ldots, v'_i, \ldots, v'_K >$;
- $P(\mathcal{W}|\boldsymbol{\psi})$ is a phrase to word segmentation model that converts a phrase sequence $\boldsymbol{\psi}$ to a word sequence $\mathcal{W}$, and by definition is a deterministic, one-to-one mapping, thus considered non-informative.

As multiple word to phrase segmentations are possible, ambiguity can occur. If there is no clear reason to favor one phrase segmentation over another, $P(\boldsymbol{\psi}'|\mathcal{W}')$ may be treated as non-informative. This is the approach adopted in this work. The input word vocabulary V_w associated with the word to phrase segmentation model, $P(\boldsymbol{\psi}'|\mathcal{W}')$, is a subset of the output phrase level vocabulary V_v. By definition, this allows single word phrases to be generated in addition to multi-word based ones.

2.1. Paraphrastic count smoothing

By differentiating Eq. (1) with respect to the paraphrastic LM log probabilities, it can be shown that the sufficient statistics for a maximum likelihood (ML) estimation of $P_{PLM}(\mathcal{W}')$ are accumulated for each paraphrase word sequence and weighted by its posterior probability. For a particular n-gram predicting word w_i following history h_i, the associated statistics $C(h_i, w_i)$ are

$$C(h_i, w_i) = \sum_{\boldsymbol{\psi}, \boldsymbol{\psi}', \mathcal{W}'} P(\mathcal{W}'|\boldsymbol{\psi}')P(\boldsymbol{\psi}'|\boldsymbol{\psi})P(\boldsymbol{\psi}|\mathcal{W})C_{\mathcal{W}'}(h_i, w_i) = \sum_{\mathcal{W}'} P(\mathcal{W}'|\mathcal{W})C_{\mathcal{W}'}(h_i, w_i) \tag{2}$$

where $C_{\mathcal{W}'}(h_i, w_i)$ is the count of subsequence $<h_i, w_i>$ occurring in paraphrase variant $\mathcal{W}'$. These sufficient statistics are then use to estimate the paraphrastic LM probabilities $\{P_{PLM}(\cdot|h_i)\}$ under a positive and sum-to-one constraint. By discounting and re-distributing statistics to alternative paraphrases of the same word sequence, paraphrastic LMs estimated using such statistics are expected to have a richer context coverage and improved generalization performance. This advantage can be exploited by various forms of LMs that do not explicitly capture the paraphrastic variability in natural languages. In this paper, the estimation of paraphrastic n-gram LMs is considered and will be described in detail in the following sections.

2.2. Interpolation with conventional LMs

At the same time as improving generalization and coverage, paraphrastic count smoothing can also increase modelling confusion compared to conventional LMs trained on just the surface word sequence. One approach to balance the specific, but poorer coverage word-based n-gram LMs with a more generic LM is to linearly interpolate the LM probabilities. This is commonly used with class-based LMs Brown et al. (1992), Niesler and Woodland (1996), Niesler et al. (1998) and is used in this paper with paraphrastic LMs. Let $P(\tilde{w}|\tilde{h})$ denote the interpolated LM probability for any in-vocabulary word $\tilde{w}$ following an arbitrary history $\tilde{h}$ is given by

$$P(\tilde{w}|\tilde{h}) = \lambda_{\text{NG}} P_{\text{NG}}(\tilde{w}|\tilde{h}) + \lambda_{\text{PLM}} P_{\text{PLM}}(\tilde{w}|\tilde{h}) \tag{3}$$

where λ_{NG} and λ_{PLM} are the interpolation weights assigned to the conventional LM distribution $P_{\text{NG}}(\cdot)$ and the paraphrastic LM $P_{\text{PLM}}(\cdot)$. These interpolation weights are positive under a sum-to-one constraint, and can be optimized on the perplexity of some held-out data.

The word level paraphrastic LMs investigated in this paper are in the same form as conventional back-off n-gram models. The only difference between the two lies in the way the sufficient statistics used in LM estimation are derived.

Instead of the normal n-gram counts computed directly from the surface word sequence, integer quantized paraphrastic word n-gram counts accumulated from explicitly generated paraphrase variants were used in this work to train word level paraphrastic LMs. These statistics were used by a conventional n-gram probability estimation and smoothing procedure, which makes no assumption over the underlying means used to derive the sufficient statistics. It also guarantees the resulting LM probabilities to be positive and sum-to-one, as well as those obtained after the linear interpolation in Eq. (3).

The same form of linear interpolation can also be used between phrase level conventional and paraphrastic LMs.[1] As multiple word to phrase segmentations are possible for the same word sequence in general, the exact phrase sequence level perplexity evaluation is non-trivial for both the conventional and paraphrastic phrase level LMs. Hence, in this paper only the perplexity performance of word level paraphrastic LMs are evaluated.

2.3. Phrase level and multi-level paraphrastic LMs

In order to increase the context span for paraphrastic LMs, a phrase level paraphrastic LM can also be trained. This can be obtained by optimizing a simplified form of the criterion given in Eq. (1), where the word to phrase segmentation model $P(\boldsymbol{\psi}'|\mathcal{W}')$ is dropped,

$$\mathcal{F}(\mathcal{W}) = \ln\left(\sum_{\boldsymbol{\psi},\boldsymbol{\psi}'} P(\mathcal{W}|\boldsymbol{\psi})P(\boldsymbol{\psi}|\boldsymbol{\psi}')P_{\mathsf{PLM}}(\boldsymbol{\psi}').\right) \tag{4}$$

Thus for a particular phrase level n-gram predicting phrase $\boldsymbol{\psi}_i$ following its history $\hat{h}_i$, the associated phrase level statistics $C(\hat{h}_i, \boldsymbol{\psi}_i)$ are accumulated as

$$C(\hat{h}_i, \boldsymbol{\psi}_i) = \sum_{\boldsymbol{\psi}'} P(\boldsymbol{\psi}'|\mathcal{W})C_{\boldsymbol{\psi}'}(\hat{h}_i, \boldsymbol{\psi}_i) \tag{5}$$

where $C(\hat{h}_i, \boldsymbol{\psi}_i)$ is the count of phrase level subsequence $<\hat{h}_i, \boldsymbol{\psi}_i>$ occurring in phrase level segmented paraphrase variant $\boldsymbol{\psi}'$. These are then used to estimated the phrase level paraphrastic n-gram LM probabilities $\{P_{PLM}(\cdot|\hat{h}_i)\}$ in this paper.

In order to incorporate richer linguistic constraints, it is possible to train and form a log-linear combination of LMs that model different units, for example, words and phrases. LMs built at word and phrase level are combined to yield a multi-level LM to further improve discrimination Liu et al. (2010, 2013a,b). This requires word level lattices to be first converted to phrase level lattices before the log-linear combination is performed. The log-linear interpolation weights determine the contribution from component LMs. These were empirically set as 0.6 and 0.4 for word and phrase level LMs, and kept fixed for all experiments of this paper.[2]

2.4. Outline of paraphrastic LM training procedure

As discussed above, paraphrastic LMs directly target expressive richness related variability in natural languages. A central part of this generative modelling framework uses a statistically trained phrase level generative model to explicitly produce multiple paraphrase variants for each training data sentence. This allows automatically discounted paraphrastic counts to be obtained to estimate LM probabilities. The overall general procedure of constructing a paraphrastic LM is summarized below.

[1] In this paper the interpolation weights assigned to the phrase level conventional and paraphrastic LMs are determined using their probabilities computed on phrase segmented held-out data based on the longest available phrase segmentation.

[2] In practice, this setting was found to give comparable performance to an equal log-linear weighting of word and phrase level LMs. The error rate was found insensitive to the setting of these weights when they are varied in the region from (0.3:0.7) to (0.7:0.3).

1: Estimation of the phrase to phrase paraphrase model $P(v|v')$ in Eqs. (1) and (4) for both word and phrase level paraphrastic LM training;
2: Construct the word to phrase segmentation model $P(\boldsymbol{\psi}'|\mathcal{W}')$ and the phrase to word segmentation model $P(\mathcal{W}|\boldsymbol{\psi})$ that produces or accepts the phrases allowed by the resulting phrase level paraphrase model $P(v|v')$;
3: **for** every sentence in the training data **do**
4: Generate paraphrase variants using the above word to phrase segmentation model $P(\boldsymbol{\psi}'|\mathcal{W}')$, the phrase level paraphrase model $P(v|v')$ and the phrase to word segmentation model $P(\mathcal{W}|\boldsymbol{\psi})$ (for word level paraphrase lattices only);
5: Accumulate paraphrastic n-gram counts at word level in Eq. (2), or phrase level in Eq. (5), over all generated paraphrase variants and weighted by their posteriors;
6: **end for**
7: Word or phrase level paraphrastic LM training using the above accumulated sufficient statistics.

In the following sections, each step of the above paraphrastic LM training procedure is described in detail.

3. Paraphrase phrase pair extraction

A phrase level paraphrase model is used in paraphrastic LMs, as discussed in Sections 1 and 2. In order to obtain sufficient phrase coverage, an appropriate technique to learn a large number of paraphrase phrase pairs is required. Since it is impractical to obtain expert semantic labelling at the phrase level, statistical paraphrase extraction schemes are needed.

3.1. Statistical paraphrase phrase pair learning

Statistical paraphrase induction methods can be categorized into two major types, depending on the nature of the data being used Androutsopoulos and Malakasiotis (2010), Madnani and Dorr (2010). The first category uses comparable or parallel text data. Coarse grained alignment Barzilay and Lee (2003), or statistical machine translation based extraction methods Brown et al. (1990) are used to learn the paraphrastic relationship among words and phrases. As these methods assume a partial or complete semantic overlap between sentences, highly specialized training material is required. Hence, it is expensive to obtain and use on a large scale. The second category of techniques perform paraphrase pair extraction using standard text data Lin and Pantel (2001), Pasca and Dienes (2005). These are motivated by *distributional similarity* theory Harris (1954), which postulates that phrase pairs often sharing the same left and right contexts are likely to be paraphrases of each other. As standard text data in large amounts can be used, wide phrase coverage can be obtained.

3.2. n-gram Paraphrase phrase pair extraction

In order to exploit the advantages of standard text data based statistical paraphrase induction techniques as discussed in Section 3.1, an n-gram based paraphrase induction algorithm given below is used in this paper to estimate the paraphrase model Liu et al. (2012b). When this distributional similarity based paraphrase learning algorithm is used, the minimum and maximum phrase length are set as $L_{\text{min}} = 1$ and $L_{\text{max}} = 4$, and the left and right context length set as $L_N = 3$. In practice these settings are found to produce a good balance between the coverage and quality of the extracted paraphrases.[3] Unless otherwise stated, these are thus kept fixed for all experiments in this paper.

[3] Decreasing the setting of context length L_N weakens the constraint used in the paraphrase learning algorithm. This results in an increase in the number of phrase pairs extracted but also a deterioration in their quality. No performance improvement was observed by further increasing L_N, or the maximum phrase length L_{max}.

initialize phrase pair list $V = \{\}$;
initialize n-gram subsequence list $U = \{\}$;
for every sentence in training data **do**
  find and add all subsequences $< c_l, v, c_r >$ such that $L_{c_l} = L_N$, $L_{c_r} = L_N$ and $L_{\min} \leq L_v \leq L_{\max}$ into U.
end for
for every $< c_l, v, c_r >$ in U **do**
  for every other $< c'_l, v', c'_r >$ in U **do**
    if $c_l = c'_l, c_r = c'_r$ and $v \neq v'$ **then**
      if $< v \rightarrow v' >$ and $< v' \rightarrow v >$ not in V **then**
        add phrase pairs $< v \rightarrow v' >, < v' \rightarrow v >$ to V;
      end if
      increase co-occurrence counts $C(v \rightarrow v')$ and $C(v' \rightarrow v)$ both by 1;
    end if
  end for
end for
for every phrase pair $< v \rightarrow v' >$ in V **do**
  estimate paraphrase prob $p(v'|v) = \frac{C(v \rightarrow v')}{\sum_{\overline{v}} C(v \rightarrow \overline{v})}$
end for

The above algorithm can be extended to incorporate additional useful information. For example, it is possible to build domain or style dependent paraphrastic LMs via a directed paraphrasing by restricting the choice of target phrases being used. In common with other paraphrase induction methods, the above scheme can also produce phrase pairs that are non-paraphrastic, for example, producing antonyms. However, this is less of a concern for language modelling since the primary aim is to improve context coverage.

In general, it is possible to allow the extracted paraphrases to contain out-of-vocabulary (OOV) words that are not modelled by the conventional LM. This can improve the resulting paraphrastic LM's vocabulary coverage. In this paper, a common vocabulary is used for both the standard LMs and paraphrastic LMs. This requires the above n-gram based paraphrase learning algorithm to be modified so that all phrase pairs that contain OOV words are discarded in the accumulation of co-occurrence counts.

4. Paraphrase lattice generation

In order to train paraphrastic LMs, multiple paraphrase variants are required to compute the sufficient statistics given in Eq. (2). As all four components of the paraphrastic LM given in Eq. (1) can be efficiently represented by WFSTs Mohri (1997), WFST based paraphrase variant generation was used in this work, rather than designing special purpose decoding tools. For each training data sentence, the paraphrase word lattice $\mathcal{T}_{\mathcal{W}'}$ is generated using a sequence of WFST composition operations, before being projected onto the word sequence level and compressed via the determinization operation. This is given by

$$\mathcal{T}_{\mathcal{W}'} = \det(\pi_{\mathcal{W}'}(\mathcal{T}_{\mathcal{W}:\mathcal{W}} \circ \mathcal{T}_{\mathcal{W}:\boldsymbol{\psi}} \circ \mathcal{T}_{\boldsymbol{\psi}:\boldsymbol{\psi}'} \circ \mathcal{T}_{\boldsymbol{\psi}':\mathcal{W}'})) \tag{6}$$

where $\mathcal{T}_{\mathcal{W}:\mathcal{W}}$ is the transducer containing the original word sequence, $\mathcal{T}_{\mathcal{W}:\boldsymbol{\psi}}$ is the word to phrase segmentation transducer, $\mathcal{T}_{\boldsymbol{\psi}:\boldsymbol{\psi}'}$ the phrase to phrase paraphrase transducer and $\mathcal{T}_{\boldsymbol{\psi}':\mathcal{W}'}$ the phrase to word transducer. $\circ$, $\det(\cdot)$ and $\pi(\cdot)$ denote the WFST composition, determinization and projection operations.

An example of a word to phrase segmentation transducer is shown in Fig. 1 (a), which can generate seven phrases. These include, the sentence start "<s>" and end symbol "</s>", single word phrases "*and*", "*I*", "*generally*" and "*prefer*", as well as a two word phrase "*and_I*". Here "<e>" denotes the null symbol. When used for paraphrase lattice generation, in order to obtain a sufficient depth of the resulting lattices, all possible word to phrase segmentations are allowed in Eq. (6) to be further transformed to their associated phrase level paraphrases. The phrase to word transducer can be derived by taking the word to phrase transducer's inverse (swapping input and output symbols). As mentioned in Section 2, both the phrase to word, and word to phrase segmentation models are considered non-informative in this paper for paraphrastic LM training.

An example part of a phrase to phrase paraphrase model is shown in Fig. 1 (b), where an input phrase "*prefer*" is paraphrased into a total of 12 single word phrases including "*appreciate*", "*like*", "*want*", "*need*", "*love*" and "*wish*", as

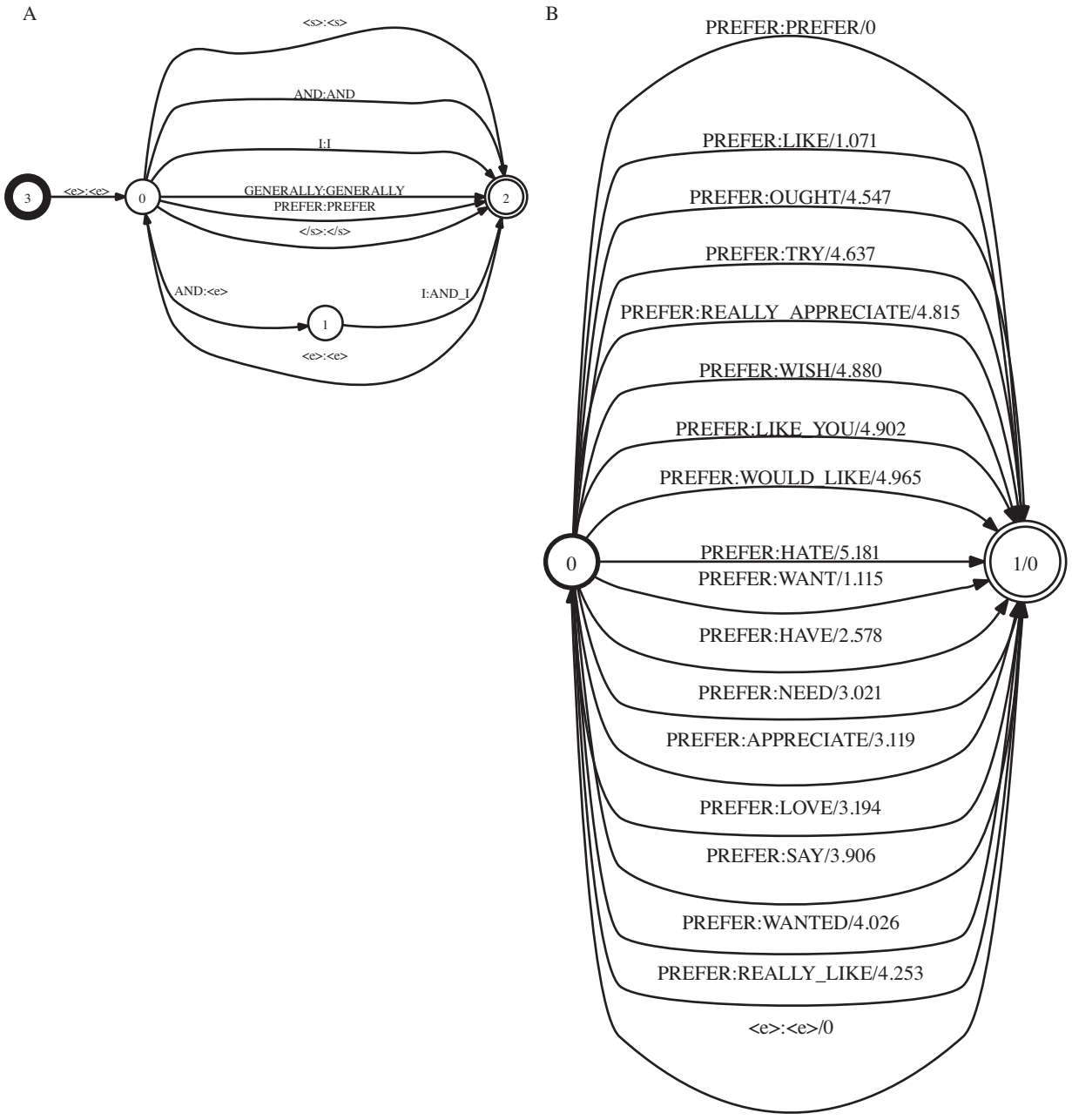

Fig. 1. Example WFST representation of (a) part of a word to phrase segmentation model that generates seven phrases including the sentence start "<s>" and end symbol "</s>", single word phrases "*and*", "*I*", "*generally*", "*prefer*" and a two word phrase "*and I*"; (b) part of a phrase to phrase paraphrase model containing arcs accepting an input phrase "*prefer*".

well as multi-word phrases such as "*really_like*" and "*really_appreciate*" and "*would_like*". When using the paraphrase phrase pair extraction method presented in Section 3, it is possible that some phrases may have no paraphrases available in the training data. In order to ensure the resulting paraphrase lattice is fully connected, self-reflexive arcs that map the input phrases to the same output are also included in the paraphrase transducer with zero cost. For the example shown in Fig. 1(b), this is represented by the top arc in the transducer that maps the input phrase "*prefer*" to itself. It should noted that including this self-reflexive arc will not lead to over counting in the paraphrastic counts accumulation in

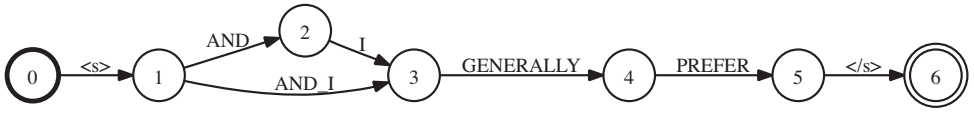

Fig. 2. Example of a phrase segmented lattice for a sentence "*And I generally prefer*" derived using the word to phrase segmentation WFST of Fig. 1(a).

Eqs. (2) and (5), as both are computed using properly normalized paraphrase sequence posterior probabilities derived from a lattice forward-backward pass.

It is also possible to construct standard, non-paraphrastic phrase level LMs using the phrase segmentation transducer shown in Fig. 1(a). First, a phrase level lattice $\mathcal{T}_{\boldsymbol{\psi}}$ containing all possible segmentations for the original word sequence $\mathcal{W}$ is derived by the following WFST operations,

$$\mathcal{T}_{\boldsymbol{\psi}} = \omega_{\doteq 1}(\det(\pi_{\boldsymbol{\psi}}(\mathcal{T}_{\mathcal{W}:\mathcal{W}} \circ \mathcal{T}_{\mathcal{W}:\boldsymbol{\psi}}))) \tag{7}$$

where $\omega_{\doteq 1}(\cdot)$ denotes the operation of setting all the WFST arc weights to 1. For an example sentence "*And I generally prefer*", the resulting phrase level segmented WFST lattice is shown in Fig. 2. In order to obtain a maximum phrase level context span and linguistic constraints, the shortest path in $\mathcal{T}_{\boldsymbol{\psi}}$, for the same example shown in Fig. 2, "*And_I generally prefer*", which contains the longest available phrase segmentation, is used to train a conventional, non-paraphrastic phrase sequence level LM.

In order to improve the efficiency in paraphrase lattice generation, a relative beam of 5.0 based WFST pruning was applied. Combined with the pruning, a bi-gram LM was used to further improve the paraphrasing efficiency and deweight the statistics accumulated from very unlikely paraphrase sequences.[4] The WFST based paraphrase generation approach in Eq. (6) is thus modified as

$$\mathcal{T}_{\mathcal{W}'} = \det(\pi_{\mathcal{W}'}(\mathcal{T}_{\mathcal{W}:\mathcal{W}} \circ \mathcal{T}_{\mathcal{W}:\boldsymbol{\psi}} \circ \mathcal{T}_{\boldsymbol{\psi}:\boldsymbol{\psi}'} \circ \mathcal{T}_{\boldsymbol{\psi}':\mathcal{W}'}) \circ \mathcal{G}_{\mathcal{W}'}) \tag{8}$$

where $\mathcal{G}_{\mathcal{W}'}$ is the acceptor representing the bi-gram LM estimated on the surface word sequence. In practice this was found to provide a good balance between improving the paraphrase generation quality and retaining sufficient depth in the lattices.

As discussed in Section 2, phrase level paraphrastic lattices are used to accumulate the sufficient statistics in Eq. (5) to train phrase level paraphrastic LMs. These phrase level paraphrase lattices $\mathcal{T}_{\boldsymbol{\psi}'}$ can be generated using a sequence of WFST operations similar to those above used for word level paraphrase lattice generation given in Eq. (8), except that the phrase to word segmentation transducer $\mathcal{T}_{\boldsymbol{\psi}':\mathcal{W}'}$ is now dropped, and a fixed phrase level bi-gram LM acceptor $\mathcal{G}_{\boldsymbol{\psi}'}$, trained on phrase segmented texts derived using the WFST operation in Eq. (7) associated with the longest available phrase segmentation, is used instead.

Using the above WFST based decoding approach for an example sentence "*And I generally prefer*", and a paraphrase model trained on 545 million words of conversational English data, an example paraphrase lattice after pruning is shown in Fig. 3.

Inside the lattice, the following paraphrase variants are among those generated: "*And I just like*", "*I mean I want*", "*I guess I prefer*", "*You know I need*", "*And I appreciate*", "*I probably have*", "'*Cause I like*", "*Well I need*" and "*So I like*". As the *n*-gram based paraphrase extraction method presented in Section 3 can also produce phrase pairs that are non-paraphrastic, antonyms such "*hate*" for word "*prefer*", previously shown in the 9th arc from the top in the phrase level paraphrase transducer of Fig. 1(b). Hence, word sequences such as "*And you know I hate*" were also found in paraphrase lattice before pruning. Using the same above paraphrase model, the lattice density measured on the word level paraphrase lattices generated for the 20 M word Fisher data is 5.1 arcs on average for every word in the surface word sequence.

In order to improve phrase coverage, expert semantic labelling provided by resources such as WordNet Fellbaum (1998) can be used. The expert semantic labelling by WordNet, including synonyms, hypernyms, hyponyms and pertainyms, were used to extract manually derived paraphrases, for example, "*choose*" and "*favor*" for word "*prefer*", in addition to those automatically learnt and shown in Fig. 1(b). The semantic similarity based HowNet Dong and Dong

[4] Initial experiments show when no such bi-gram LMs were used in paraphrase lattice generation, a small performance degradation was found in the resulting paraphrastic LM.

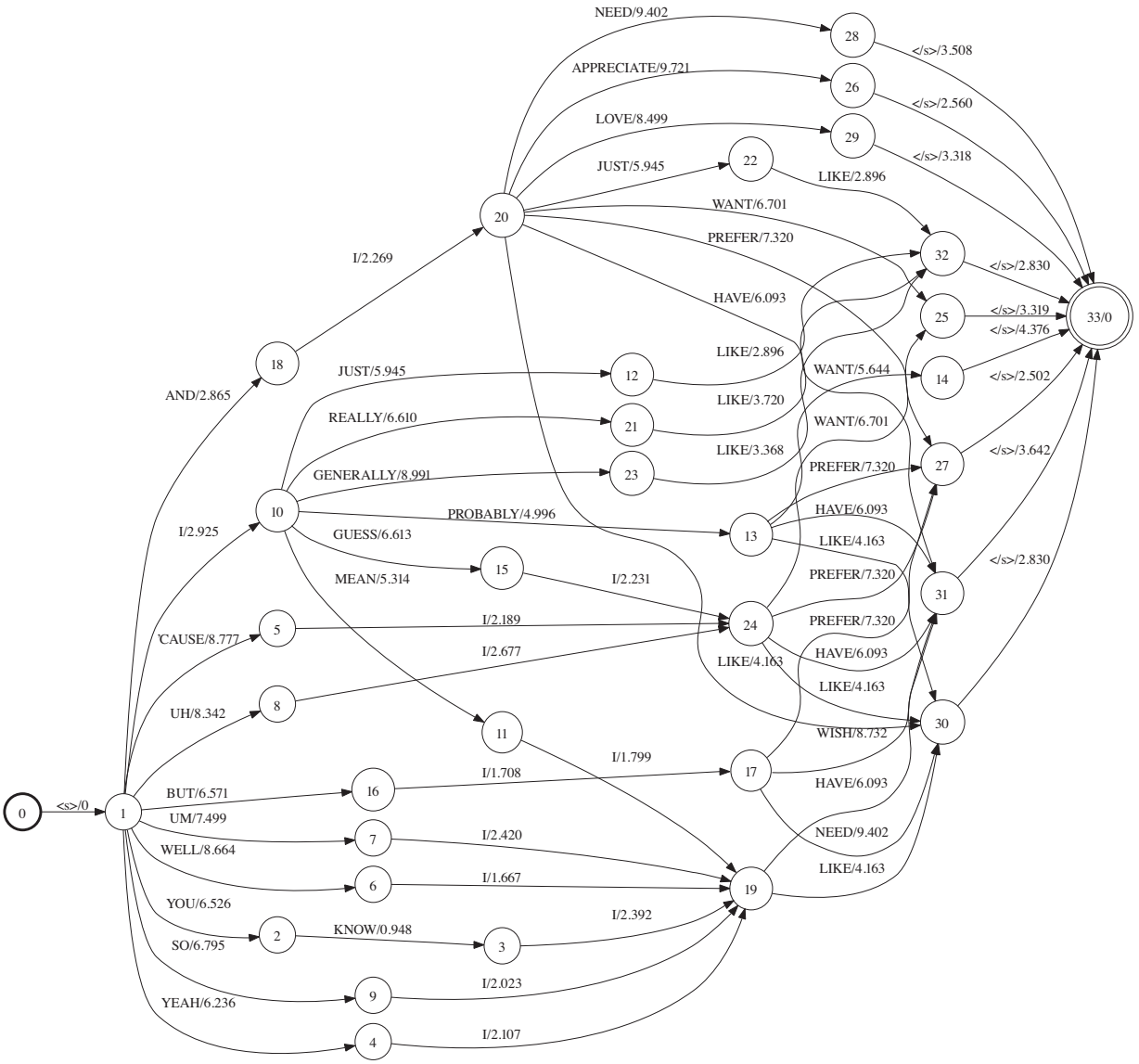

Fig. 3. A paraphrase lattice for sentence "*And I generally prefer*".

(2006) for the Chinese language was also used in this paper to generate expert paraphrases. Multi-character words that share the same semantic class labelling and part-of-speech tagging are considered as paraphrases to each other. As for both expert resources these paraphrase phrase pairs are not statistically derived, the resulting paraphrase model are treated as non-informative, and all the arcs in the paraphrase transducer, for example, shown in Fig. 1(b), carry zero cost. In order to reduce the statistics contribution from very unlikely paraphrase sequences, the bi-gram LM introduced in Eq. (8) is again used in paraphrase lattice generation. Due to their different nature, statistically learned and expert derived paraphrase pairs were used to generate separate sets of lattices, and paraphrastic LMs. These models are then used in the interpolation with standard LMs in Eq. (3). When combined with conventional LMs using Eq. (3), their interpolation weights are optimized on the perplexity of some held-out data, as discussed in Section 2.

5. Estimation of paraphrastic LMs

After paraphrase lattices are generated using the WFST based decoding algorithm discussed in Section 4, the n-gram statistics given in Eqs. (2) and (5) required for paraphrastic LM estimation are accumulated from these word or phrase

level paraphrase lattices via a forward-backward pass. These sufficient statistics are then use to estimate back-off word or phrase level n-gram Katz (1987) probabilities. In order to improve generalization to contexts that cannot be found in either the training data or the associated paraphrases, appropriate parameter smoothing and discounting methods, for example, modified KN smoothing Chen and Goodman (1999), are required.

As the standard modified KN smoothing algorithm was originally derived only for integer based n-gram counts, it cannot be directly used on the fractional paraphrastic LM counts. To address this issue, it is possible to extend the modified KN smoothing method itself to handle fractional statistics along the line of the work proposed in Tam and Schultz (2008).[5] In this paper, a simplified approach was used to handle the same issue. Word or phrase level fractional n-gram counts were quantized into integers before n-gram estimation. First, a minimum float count cut-off of 0.001 was applied to discard all rare paraphrastic counts below such threshold. All the retained fractional counts were then increased by 1 before rounding to the nearest integer. The resulting integer quantized statistics were used in n-gram estimation and modified KN smoothing with the SRILM toolkit Stolcke (2002). Independent of the nature of the underlying means used to derive the integer based n-gram sufficient statistics, being computed directly from the surface word sequence as in conventional LMs, or from paraphrase lattices as in this work, this well established n-gram probability estimation and smoothing procedure guarantees the resulting LM probabilities to be under the positive and sum-to-one constraints. In practice this approach was found to outperform alternative smoothing methods such as Witten-Bell smoothing in terms of both perplexity and error rate. The resulting paraphrastic n-gram LMs were then linearly combined with the conventional LM using the form of interpolation given in Eq. (3).

6. Combining paraphrastic LMs with neural network LMs

In order to handle the data sparsity problem, language modelling techniques based on a continuous vector space representation of word sequences, such as neural network LMs (NNLM), can be used. Depending on the network architecture being used, these can be categorised into feed-forward n-gram based NNLMs Bengio et al. (2003), Kuo et al. (2012), Le et al. (2013), Park et al. (2010), Schwenk (2007), and recurrent NNLMs Mikolov et al. (2010), Sundermeyer et al. (2012). In this paper, the combination between paraphrastic LMs and feed-forward n-gram based NNLMs is considered Liu et al. (2013c).

Both paraphrastic LMs and feed-forward NNLMs can improve LM generalization. However, there are major differences between them that can also be exploited as complementary characteristics. First, paraphrastic LMs can be trained using large amounts of training data. In contrast, to reduce computational cost, feed-forward NNLMs are normally trained using only a small in-domain data set, for example, audio transcripts, and optionally a re-sampled subset of out-of-domain data Schwenk (2007) available in large quantities. Secondly, paraphrastic LMs re-distribute sufficient statistics to variable length paraphrase variants of the same sentence. The resulting sequence level smoothing of LM probabilities is different to the n-gram level smoothing used by NNLMs. Finally, the paraphrastic LMs considered in this paper are based on n-gram models. Despite being more efficient than NNLMs in probability computation, their generalization ability remains limited for unseen contexts that cannot be found in either the training data or the associated paraphrases. Hence, in order to leverage the strengths of both models, the combination between paraphrastic LMs and NNLMs is investigated in this paper. The particular form of combination considered in this paper is a linear interpolation between the paraphrastic LM, the feed-forward NNLM and the conventional n-gram LM. The interpolated LM probabilities given in Eq. (3) are therefore modified as,

$$P(\tilde{w}|\tilde{h}) = \lambda_{\mathsf{NG}} P_{\mathsf{NG}}(\tilde{w}|\tilde{h}) + \lambda_{\mathsf{PLM}} P_{\mathsf{PLM}}(\tilde{w}|\tilde{h}) + \lambda_{\mathsf{NN}} P_{\mathsf{NN}}(\tilde{w}|\tilde{h}) \quad (9)$$

where λ_{NN} is the interpolation weight assigned to the neural network LM. In the same fashion as in Eq. (3), component LM interpolation weights can be optimized on held-out data.

For the multi-level paraphrastic LMs discussed in Section 2, the above interpolation needs to be performed at both the word and phrase level prior to the log-linear combination between the word and phrase level LMs. In addition to a word level neural network LM, a neural network LM constructed using phrase level segmented training data is also

[5] A fractional Kneser–Ney smoothing scheme was investigated in Tam and Schultz (2008) for correlated bi-gram latent semantic analysis (LSA) models. In the experimental results presented in that work, a small error rate reduction of 0.1% absolute was reported over interger counts based Witten–Bell smoothing, where the baseline bi-gram LSA model using gave an error rate of 23.8%

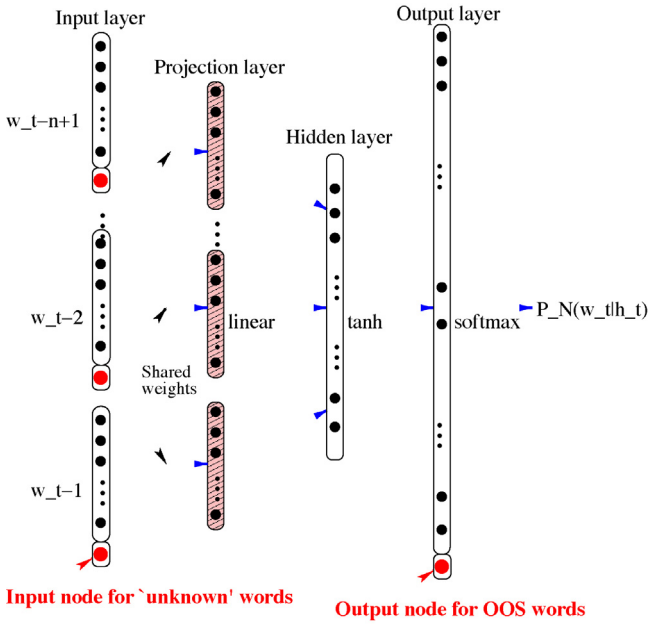

Fig. 4. Architecture of a 4-gram NNLM with an OOS output node.

required. The phrase level segmented data can be obtained using the WFST operations given in Eq. (7) and 1-best search in the resulting phrase lattices as described earlier in Section 4.

To reduce computational cost, conventional feed-forward NNLMs only model the probabilities of a small and more frequently occurring subset of the complete vocabulary, commonly referred to as the *shortlist* Schwenk (2007). The output layer normally only contains nodes for in-shortlist words. A similar approach may also be used at the input layer when a large vocabulary is used. Two issues arise when using this conventional NNLM architecture. First, NNLM parameters are trained only using the statistics of in-shortlist words thus introduces an undue bias to them. Secondly, as there is no explicit modelling of probabilities of *out-of-shortlist* (OOS) words in the output layer, statistics associated with them are also discarded in NNLM training. To handle these issues, alternative network architectures that model a full vocabulary at the output layer can be used Park et al. (2010), Le et al. (2013). In this paper, an NNLM architecture with an additional output node explicitly modelling the probability mass of OOS words Park et al. (2010) is used. This ensures that all training data are used in NNLM training, and the probabilities of in-shortlist words are smoothed by the OOS probability mass, thus obtaining a more robust parameter estimation. The architecture of a 4-gram feed-forward NNLM of this form is illustrated in Fig. 4.

7. Experiments and results

In this section, the performance of various paraphrastic language models are evaluated using two HTK-based large vocabulary speech recognition tasks. The first was developed for English conversational telephone speech (CTS) used in the 2004 DARPA EARS evaluation, while the second system for Mandarin Chinese broadcast speech was used in the 2011 DARPA GALE evaluation. A series of experiments were conducted on these two tasks. These were designed to investigate the following topics:

- the performance of the paraphrastic LM presented in Section 2 when being used to improve word level n-gram LMs;
- the performance of the multi-level paraphrastic LM proposed in Section 2 to incorporate additional phrase level linguistic constraints;
- the scalability of paraphrastic LM training using large or small amounts of data;
- the generalization of paraphrastic LMs to different languages and tasks;
- the combination between paraphrastic LMs and neural network LMs Schwenk (2007).

Table 1
Text size, paraphrase extraction method and the number of phrase pairs extracted from different data sources.

Source	Size	Extraction	# phrase pairs
WordNet	–	Expert	480 k
Fisher	20 M	Automatic	90 k
UWWeb	525 M	Automatic	2.9 M

For all results presented in this paper, a matched pairs sentence-segment word error (MAPSSWE) based statistical significance test was performed at a significance level $\alpha = 0.05$. All n-gram LMs considered in the experiments were built using modified KN smoothing Chen and Goodman (1999).

7.1. Experiments on English conversational telephone speech

The 2004 CU English CTS LVCSR system Evermann et al. (2005) was trained on approximately 2000 h of Fisher conversational speech released by the LDC. A 59 k recognition word list was used in decoding. The system uses a multi-pass recognition framework. The initial lattice generation used gender dependent cross-word triphone acoustic models. These acoustic models include conversation side level normalization of PLP Woodland et al. (1996) features; HLDA Kumar (1997), Liu et al. (2003) projection; HMM parameter estimation using MPE Povey and Woodland (2002); and unsupervised MLLR Leggetter and Woodland (1995) speaker adaptation. An interpolated 3-gram word level baseline LM was used. The resulting lattices are then used in rescoring experiments to evaluate performance of various LMs. A detailed description of the baseline system can be found in Evermann et al. (2005). The 3 h *dev04* data, which includes 72 Fisher conversation sides, was used as a test set.

The baseline LM was trained using a total of 1.0 billion words from 8 difference text sources. The two text sources with the highest interpolation weights, the LDC Fisher acoustic transcriptions Cieri et al. (2004), *Fisher*, of 20 million words (0.6), and the University Washington conversational web data Bulyko et al. (2003), *UWWeb* of 525 million words (0.2), were used to build various language models. These LMs are then used for lattice rescoring and word error rate (WER) performance evaluation. Information on the corpus size, the paraphrase extraction schemes used and the number of phrase pairs extracted from the these two text sources, as well as the phrase pairs extracted from WordNet, are given in Table 1. Using the automatic n-gram paraphrase extraction scheme presented in Section 3, a total of 90 k and 2.9 M phrase pairs were extracted from the *Fisher* and *UWWeb* data respectively. The expert semantic labelling by WordNet, including synonyms, hypernyms, hyponyms and pertainyms, was used to generate 480 k paraphrase phrase pairs.

7.1.1. Experiments on CTS Fisher data

The WER performance of various LMs trained using the *Fisher* data only are shown in Table 2 for *dev04*. The first three baseline LMs are non-paraphrastic. The word level 4-gram baseline LM "w4g" gave a WER of 17.6%. When further interpolated with a class based LM of 1000 automatically derived word clustersKneser and Ney (1993), the "w4g+clslm" model reduces the error rate by 0.2% absolute. The third baseline LM in Table 2 is a multi-level LM, "w4g ∘ p4g", which incorporates phrase level linguistic constraints by the log-linear combination of the word, and phrase level back-off 4-gram LMs (taking the most recent context of three phrases to predict the current phrase). It was

Table 2
WER performance of LMs trained using the *Fisher* data only for *dev04*. "w4g" denotes word level 4-gram LM, "w4g+clslm" a word level 4-g LM interpolated with a class LM with 1000 classes, and "w4g ∘ p4g" a multi-level LM log-linearly combining word and phrase level 4-gram LMs.

LM	Paraphrastic	dev04
w4g	×	17.6
w4g+clslm	×	17.4
w4g ∘ p4g	×	17.5
w4g	√	17.2
w4g ∘ p4g	√	17.0

Table 3
Perplexity of LMs trained using the *Fisher* data only for *dev04*. Naming convention same as Table 2.

LM	Paraphrastic	dev04
w4g	×	56.0
w4g+clslm	×	54.0
w4g	√	54.9

Table 4
WER performance of LMs trained using the *Fisher* data only with different paraphrase learning methods for *dev04*. "w4g" denotes word level 4-g LM.

LM	Paraphrastic	Extraction	dev04
w4g	√	Automatic+Expert	17.2
		Automatic	17.3

constructed by adding a total of 16 k distinct multi-word phrases found in the *Fisher* data generated paraphrase phrase table to the baseline 59 k word list, and trained on the phrase level text data obtained using a longest available word to phrase segmentation. This is similar to the method used in Padmanabhan et al. (1998). As discussed in Section 2, word level lattices need to be first converted to phrase level lattices when using the multi-level LM. This was implemented using a WFST composition between the word level lattice with the phrase level segmentation transducer shown in Fig. 1(a). After the log-linear combination between word and phrase level LMs is performed, the resulting phrase level lattices are converted back to word level again via a WFST composition with the phrase to word transducer, to obtain the 1-best word level hypothesis for WER evaluation. By adding additional phrase level features, this multi-level LM gives a small improvement of 0.1% absolute over the word level 4-gram baseline LM.

In contrast, the comparable word level paraphrastic 4-gram LM, shown in the 4th line of Table 2, using the paraphrase phrase pairs extracted from the *Fisher* training data itself and WordNet, as given in Table 1, outperformed the word level baseline 4-gram LM, and the class LM baseline, by 0.4% and 0.2% absolute respectively. As discussed in Section 2, paraphrastic LMs re-distribute sufficient statistics to alternative variants of the same surface word sequence. As expected, an improvement in n-gram coverage was also found using the word level paraphrastic 4-gram LM. For example, the 3-gram hit rate on the test data reference transcription was increased by 12.4% relative over the baseline word level 4-gram LM. At the same time the total number of n-grams was increased by more than a factor of four from 17.5 M in baseline word level 4-gram LM to 81 M after interpolation with the comparable word level paraphrastic 4-gram LM. Applying entropy based pruning to the resulting interpolated LM and reducing its number n-grams to approximately the same as the baseline non-paraphrastic 4-gram LM, a competitive WER of 17.3% was obtained. This suggests paraphrastic LMs can also be used to improve LM performance for tasks with constrained computing resources.

It is interesting that the rank order in terms of perplexity is slightly different to that for WER between the class LM and paraphrastic 4-gram LM (see the 2nd and 3rd lines of Table 3). This may be due to two reasons. First, a stronger constraint is used by paraphrastic LMs during parameter estimation. As discussed in Section 2, paraphrastic LMs re-distribute statistics only to the paraphrase variants of the original sentence, and optionally non-paraphrastic sequences that share a similar syntactic or semantic structure. Hence, the resulting LM is focused on modelling what is considered to be acceptable by the paraphrase model for the particular language being considered. In contrast, such a constraint is not explicitly exploited by class LMs. Many sentences, including very unlikely ones, can share the same underlying class sequence representation. Such an increase in modelling confusion can lead to a loss in discrimination and increase in error rate. Furthermore, the correlation between perplexity and error rate is known to be fairly weak for current speech recognition systems.

It was also found that adding the paraphrases extracted from WordNet gave only a marginal improvement over using only those automatically learned from the *Fisher* data. For example, a comparable paraphrastic word level LM derived using only the 90 k phrase pairs obtained from *Fisher* data gave a very similar WER performance of 17.3%, as is shown in the last line of Table 4, where the standard paraphrastic LM constructed using both automatically and

Table 5
WER performance of LMs trained using the *Fisher* data and its three subsets for *dev04*. Naming convention same as Table 2.

LM	Paraphrastic	dev04			
		1 M	5 M	10 M	20 M
w4g	×	20.3	18.7	18.0	17.6
w4g ∘ p4g	√	19.8	18.2	17.6	17.0

Table 6
WER performance of LMs trained using *Fisher* and *UWWeb* data on *dev04*. Naming convention same as Table 2.

LM	Paraphrastic	dev04
w4g		16.7
w4g+clslm	×	16.5
w4g ∘ p4g		16.5
w4g	√	16.4
w4g ∘ p4g		16.2

expert derived paraphrases, previously shown in the 4th line of Table 2, is again shown in the 1st line of Table 4 for a contrast.

The marginal difference in error rate between the two paraphrastic LMs is expected as the *Fisher* corpus provides the in-domain data for the CTS task, while the expert paraphrases of WordNet are more task independent. It is also found that a word level paraphrastic 4-gram LM derived using a restricted paraphrase model that only allow word to word paraphrases, gave an error rate comparable to the class LM baseline (previously shown in the 2nd line of Table 2).

When using the paraphrastic multi-level LM, shown in the last line of Table 2, a significant WER reduction of 0.5% was obtained over the baseline non-paraphrastic multi-level LM shown in the 3rd line of Table 2. The overall improvement over the word level 4-gram baseline LM is 0.6% absolute, which is also statistically significant.

7.1.2. Experiments on smaller and larger amounts of training data

Consistent performance improvements using paraphrastic multi-level LMs were also obtained over the baseline 4-gram LM when both were trained on reduced amounts of training data, where randomly selected *Fisher* data subsets of 10 M, 5 M and eventually 1 M words, were used in the experiments. For each training data subset, both the paraphrase models and the resulting paraphrastic LM were re-estimated on the corresponding reduced amount of data.[6] These are shown in Table 5, where the WER performance on the 20 M word full *Fisher* set (also previously shown in the 1st and 5th lines of Table 2), are again shown in the last column. These results confirm the good scalability of paraphrastic LMs when trained using small amounts of data.

The same trend can also be found in a set of experiments conducted on a larger LM training set where the 525 M word *UWWeb* data is also used in LM training as a second source via a linear interpolation with the *Fisher* data trained LM, as are shown in Table 6. As expected, adding this data source significantly improved the performance of three non-paraphrastic baseline LMs by 0.9–1.0% absolute, compared with the results shown in the first three lines of Table 2. The word level paraphrastic 4-gram LM, as is shown in the 4th line of Table 6, using the paraphrase phrase pairs extracted from all three data sources given in Table 1, outperformed the word level baseline LM by 0.3% absolute. Consistent with the results presented in Table 4, the same performance was obtained if only using statistically learnt paraphrases in PLM training. In a similar manner to the experiments previously conducted using the *Fisher* data only in Table 2, the 3-gram hit rate on the test data reference transcription was also increased by 11.3% relative over the baseline word level 4-gram LM. When using the paraphrastic multi-level LM, as is shown in the last line of Table 6, an overall significant WER reduction of 0.5% absolute was obtained over the word level 4-gram baseline LM. It also

[6] In order to obtain a good balance between the richness of automatically derived paraphrases and the robustness of paraphrase model estimation, for the two smaller subsets of 1 M and 5 M words, a decreased left and right context length $L_N = 2$ and a maximum phrase length $L_{\max} = 2$ were used during paraphrase extraction.

Table 7
Text size, paraphrase extraction method and the number of phrase pairs extracted from different Chinese text data sources.

Source	Size	Extraction	# phrase pairs
HowNet	–	Expert	3.7 M
BN+BC	20 M	Automatic	80 k
GigaXin	680 M	Automatic	35.9 M
GALEWeb	800 M	Automatic	1.8 M

outperformed a word level 4-gram baseline LM trained using twice the amount of data, 1.0 billion words, with 6 more text sources in addition to *Fisher* and *UWWeb*, by 0.2%.

The results in both Tables 2, 5 and 6confirm the first three advantages of paraphrastic LMs, as discussed in the beginning of Section 7. First, paraphrastic LMs are effective in improving the generalization performance of word level *n*-gram LMs. Second, multi-level paraphrastic LMs can exploit additional useful linguistic constraints at phrase level. Finally, it is possible to induce paraphrase phrase pairs and train paraphrastic LMs on small or large amounts of data.

7.2. Experiments on mandarin Chinese broadcast speech

The 2011 CU Mandarin Chinese LVCSR system Liu et al. (2012a) was then used to examine the generalization ability of the paraphrastic LM to different languages and tasks. The system was trained on 1960 h of Mandarin Chinese broadcast speech data released by the LDC for the DARPA GALE program.[7] A 63 k recognition word list was used in decoding. The system uses the same multi-pass recognition framework as described in Section 7.1. In the initial lattice generation stage, MLLR Leggetter and Woodland (1995) speaker adapted gender dependent cross-word triphone MPE Povey and Woodland (2002) acoustic models with HLDA Kumar (1997), Liu et al. (2003) projected PLP Woodland et al. (1996) features augmented with pitch features, and an interpolated 3-gram word level baseline LM were used. A detailed description of the baseline system can be found in Liu et al. (2012a). A 3 h test set of Chinese speech used in the GALE program, *dev09s*, of mixed broadcast news (BN) and conversation (BC) genres was used.

The baseline LM was trained using a total of 5.9 billion characters from 28 difference text sources. These account for 4 billion words after a longest first based character to word segmentation as applied. The four text sources with the highest interpolation weights, the acoustic transcriptions, *BN* (0.13) and *BC* (0.31), of 20 million words in total, the LDC GigaWord Xinhua News data, *GigaXin* (0.16), of 680 million words, and the GALE web data *GALEWeb* of 800 million words (0.09), were used to build various language models. These LMs are then used for lattice rescoring and character error rate (CER) evaluation. The 4 billion word full set trained 4-gram LM gave an error rate of 10.3% on *dev09s*, while a comparable 4-gram LM trained using only the above four text sources produced a competitive CER score of 10.4% and was used as a baseline in the following experiments, as is shown in the 1st line of Table 8. Using this baseline system the BN and BC genre specific performance on *dev09s* are 5.4% and 15.2% respectively.

Information on the corpus size, the paraphrase extraction schemes used and the number of phrase pairs extracted from the these four text sources, as well as those from HowNet Dong and Dong (2006), an expert semantic database for the Chinese language, are given in Table 7. Using the automatic *n*-gram paraphrase extraction scheme presented in Section 3, a total of 80 k, 35.9 M and 1.8 M phrase pairs were extracted from the *BN+BC*, *GigaXin* and *GALEWeb* data respectively. The expert semantic labelling by HowNet was used to generate 3.7 M paraphrase phrase pairs.

The word level paraphrastic 4-gram LM, as is shown in the 4th line of Table 8, outperformed the word level baseline LM "w4g" (shown in the 1st line of Table 8) by 0.3% absolute. This is consistent with the trends previously shown in Tables 2 and 6. The two multi-level LMs in Table 8 both used a total of 503 k distinct multi-word phrases found in the *BN*, *BC* and *GigaXin* data generated paraphrase phrase table that were added to the baseline 63 k word list. The baseline non-paraphrastic multi-level LM, shown in 3rd line of Table 8, was trained on the phrase level text data obtained using a longest available word to phrase segmentation, in the same fashion as described in Section 7.1 for the English CTS system. The paraphrastic multi-level LM, shown in the last line of Table 8, outperformed its comparable

[7] The purpose of the GALE program is to make Arabic and Chinese broadcasts, newswire, and web logs accessible to monolingual English speakers. Hence, the GALE program has sponsored annual competitive evaluations of machine translation systems in which speech recognition is a necessary front-end for broadcast material. Details of much of the research performed under the GALE program can be found in Olive et al. (2011).

Table 8
WER performance of LMs trained using *BN*, *BC*, *GigaXin* and *GALEWeb* data on *dev09s*. Naming convention same as Table 2.

LM	Paraphrastic	dev09s
w4g	×	10.4
w4g+clslm		10.3
w4g ∘ p4g		10.1
w4g	√	10.1
w4g ∘ p4g		9.9

Table 9
WER performance of LMs trained using *BN*, *BC*, *GigaXin* and *GALEWeb* data on *dev09s*. "w4g" denotes word level 4-gram LM, "w4g+nn$_w$" a word level 4-gram LM interpolated with a word level 4-gram feed-forward NNLM, and "(w4g+nn$_w$) ∘ (p4g + nn_p)" a multi-level LM log-linearly combining word and phrase level LMs, after a linear interpolation of 4-gram LMs with corresponding NNLMs at both word and phrase level before a log-linear combination.

LM	Paraphrastic	
	n-gram LM	dev09s
w4g	×	10.4
w4g+nn$_w$		10.0
(w4g+nn$_w$) ∘ (p4g + nn_p)		9.8
w4g	√	10.1
w4g+nn$_w$		9.7
(w4g+nn$_w$) ∘ (p4g + nn_p)		9.5

non-paraphrastic baseline, shown in the 3rd line of Table 8, by 0.2% absolute. Using this paraphrastic multi-level LM, an overall significant CER reduction of 0.5% absolute was obtained over the word level 4-gram baseline LM. It also significantly outperformed the word level 4-gram baseline LM, which was trained using four times the amount of data (4 billion words) with 24 more text sources, by 0.4% absolute. These results, together with those previously shown in Table 6 for English conversational telephone speech, confirm that paraphrastic LMs improve performance for different language and domains, and once again demonstrate their scalability when large amounts of training data is used.

7.3. Experiments on combining paraphrastic LMs with neural network LMs

So far in this paper, paraphrastic LMs have been used to successfully improve the performance of standard n-gram LMs. As discussed in Section 6, it is also interesting to investigate the combination between paraphrastic LMs and state-of-the-art language modelling techniques, such as neural network LMs Schwenk (2007). A set of experiments were first conducted on the Mandarin Chinese broadcast task described previously in Section 7.2. A total of four LMs shown in Table 8, including the baseline 4-gram word level LM, its paraphrastic counterpart, and the two multi-level LMs, were combined with various feed-forward NNLMs using the method presented in Section 6. A word level 4-gram feed-forward NNLM with an OOS output layer node Park et al. (2010) was trained using the 20 million words of the *BN*+*BC* acoustic transcriptions only. The size of the NNLM input and output vocabularies are 45 k and 20 k words respectively. A phrase level 4-gram NNLM was also trained using the same data, and the same phrase segmentation used by the multi-level LMs of Table 8. Its input and output vocabularies contain 100 k and 20 k most frequent phrases. For both the word and phrase level NNLMs, a total of 600 projection layer nodes (200 nodes per input word) and 400 hidden layer nodes were used. The performance of the baseline and the paraphrastic 4-gram word level LMs without any interpolation with NNLMs, are shown in the 1st and 4th lines of Table 9 (also previously shown in the 1st and 4th lines of Table 8).

The results in Table 9 show that the improvements from paraphrastic LMs and neural network LMs are largely additive. For example, the word level paraphrastic 4-gram LM outperformed the baseline 4-gram LM "w4g" by 0.3% absolute. The same improvement was retained when both LMs were further combined with the word level neural network LM, "w4g+nn$_w$", as are shown in the 2nd and the 5th line of Table 9 respectively.

Table 10
WER performance of LMs trained using *Fisher* and *UWWeb* data on *dev04*. "w4g" denotes a word level 4-gram LM, "w4g+nn$_w$" a word level 4-gram LM interpolated with a word level feed-forward NNLM.

LM	Paraphrastic	
	n-gram LM	dev04
w4g	×	16.7
w4g+nn$_w$	×	16.3
w4g	√	16.4
w4g+nn$_w$	√	16.1

Similarly on the English conversational telephone speech task presented in Section 7.1, combining the baseline word level 4-gram LM (the 1st line in Table 6 and again, the 1st line in Table 10), and the word level 4-gram paraphrastic LM (the 4th line in Table 6 and again, the 3rd line in Table 10), with a word level 4-gram feed-forward NNLM of a similar architecture (single OOS output node, 38 k input and 20 k output vocabularies, 600 projection layer nodes and 400 hidden layer nodes) trained using the *Fisher* transcription only, comparable WER reductions of 0.4% and 0.3% absolute were obtained, as are shown in the 2nd and 4th lines of Table 10. These results confirm the complementarity between paraphrastic LMs and neural network LMs as discussed in Section 6.

Over the three baseline non-paraphrastic LMs with increasing modelling complexity (first three lines of Table 9) on the GALE Mandarin task, a consistent CER reduction of 0.3% absolute was obtained when using the comparable paraphrastic LMs (last three lines of the Table 9). Also in line with the results shown in Tables 2, 6 and 8, further improvements were obtained using multi-level LMs. The best performance was obtained using the paraphrastic multi-level LM shown in the bottom line of Table 9, which used a three-way interpolation between the baseline LM, paraphrastic LM and neural network LM at both the word and phrase level before a log-linear combination was performed.[8] Using this LM, total error rate reductions of 0.9% absolute (9% relative) and 0.5% absolute (5% relative) were obtained over the baseline 4-gram word level LM "w4g" and the neural network LM "w4g+nn$_w$" respectively, both being statistically significant. The genre specific CER reductions over the baseline 4-gram word level LM "w4g" are 0.5% absolute (9% relative) for BN and 1.2% absolute (8% relative) for BC.

8. Conclusion

This paper investigated using paraphrastic language models for speech recognition. Phrase level paraphrase models statistically learned from standard text with no semantic annotation were used to generate multiple paraphrase variants. Language model probabilities are then estimated in the paraphrase domain. Phrase level linguistic constraints were further incorporated using a multi-level LM framework. The combination between paraphrastic LMs and neural network LMs was also investigated. Significant error rate reductions of 0.5–0.6% absolute were obtained on two state-of-the-art large vocabulary speech recognition tasks. Experimental results suggest:

- paraphrastic LMs can be used to improve performance of *n*-gram LMs;
- multi-level paraphrastic LM can give further performance improvements by incorporating additional phrase level linguistic constraints;
- it is possible to estimate paraphrastic LMs using large or small amounts of data;
- paraphrastic LMs can be applied to multiple languages and tasks;
- performance improvements from paraphrastic LMs can be retained when combined with neural network LMs.

Future research will focus on improving paraphrase extraction and directed paraphrasing for task and style adaptation.

[8] A slight performance degradation of 0.1% was found using "(w4g+nn$_w$) ○ p4g", a multi-level LM that used a linear interpolation between 4-gram LMs and an NNLM at the word level only.

Acknowledgments

The research leading to these results was supported by EPSRC grant EP/I031022/1 (Natural Speech Technology) and DARPA under the Broad Operational Language Translation (BOLT) program.

References

Androutsopoulos, I., Malakasiotis, P., 2010. A survey of paraphrasing and textual entailment methods. J. Artif. Intell. Res. 38, 135–187.
Barzilay, R., Lee, L., 2003. Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In: Proc. of HLT-NAACL 2003, Edmonton, pp. 16–23.
Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C., 2003. A neural probabilistic language model. J. Mach. Learn. Res. 3 (2), 1137–1155.
Brown, P.F., et al., 1990. A statistical approach to machine translation. Comput. Linguist. 16 (2), 79–85.
Brown, P.F., et al., 1992. Class-based *n*-gram models of natural language. Comput. Linguist. 18 (4), 467–470.
Bulyko, I., Ostendorf, M., Stolcke, A., 2003. Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures. In: Proc. HLT, Edmonton, Canada.
Cao, G., Nie, J.-Y., Bai, J., 2005. Integrating word relationships into language models. In: Proc. ACM SIGIR2005, Salvador, Brazil, pp. 298–305.
Chen, S.F., Goodman, J.T., 1999. An empirical study of smoothing techniques for language modeling. Comput. Speech Lang. 13 (4), 359–394.
Cieri, C., Miller, D., Walker, K., 2004. The Fisher corpus: a resource for the next generations of speech-to-text. In: Proc. LREC, pp. 69–71.
Chomsky, N., 1966. Topics in the Theory of Generative Grammar, vol. 56. Walter de Gruyter, The Hague, The Neatherlands.
Dong, Z., Dong, Q., 2006. HowNet and the Computation of Meaning. World Scientific, Charlottesville, Virginia, pp. 1–316, ISBN: 978-981-256-491-7.
Evermann, G., Chan, H.Y., Gales, M.J.F., Jia, B., Mrva, D., Woodland, P.C., Yu, K., 2005. Training LVCSR systems on thousands of hours of data. In: Proc. ICASSP2005, vol. 1, Philadelphia, PA, pp. 209–212.
Fellbaum, C., 1998. WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA.
Harris, Z., 1954. Distributional structure. Word 10 (23), 146–162.
Hoberman, R., Rosenfeld, R., (2002). Using WordNet to supplement corpus statistics (online document). Available from: http://www.cs.cmu.edu/~roseh/Papers/wordnet.pdf, 2002.
Jackendoff, R., 1974. Semantic Interpretation in Generative Grammar. MIT Press, Cambridge, MA, ISBN: 0262100134.
Jelinek, F., Mercer, R., Roukos, S., 1990. Classifying words for improved statistical language models. In: Proc. IEEE ICASSP1990. vol. 1, Albuquerque, New Mexico, pp. 621–624.
Katz, S.M., 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. IEEE Trans Acoust. Speech Signal Process. 35, 400–401.
Kneser, R., Ney, H., 1993. Improved clustering techniques for class based statistical language modeling. In: Proc. EuroSpeech93', Berlin.
Kneser, R., Peters, J., 1997. Semantic clustering for adaptive language modeling. In: Proc. ICASSP1997. vol. 2, Munich, pp. 779–782.
Kumar, N., 1997. Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition. John Hopkins University, Baltimore (Ph.D. thesis).
Kuo, H.-K., Arisoy, E., Emami, A., Vozila, P., 2012. Large scale hierarchical neural network language models. In: Proc. ISCA Interspeech2012, Portland, Oregon.
Leggetter, C.J., Woodland, P.C., 1995. Maximum likelihood linear regression for speaker adaptation of continuous density HMMs. Comput. Speech Lang. 9, 171–186.
Le, H.-S., Oparin, I., Allauzen, A., Gauvain, J.-L., Yvon, F., 2013. Structured output layer neural network language models for speech recognition. IEEE Trans. Audio Speech Lang. Process. 21 (1), 197–206.
Lin, D., Pantel, P., 2001. DIRT – discovery of inference rules from text. In: Proc. ACM SIGKDD2001, San Francisco, CA, pp. 323–328.
Liu, X., Gales, M.J.F., Woodland, P.C., 2003. Automatic complexity control for HLDA systems. In: Proc. IEEE ICASSP2003. vol. 1, Hong Kong, China, pp. 132–135.
Liu, X., Gales, M.J.F., Hieronymus, J.L., Woodland, P.C., 2010. Language model combination and adaptation using weighted finite state transducers. In: Proc. ICASSP, Dallas, TX, pp. 5390–5393.
Liu, X., Gales, M.J.F., Woodland, P.C., 2013a. Use of contexts in language model interpolation and adaptation. Comput. Speech Lang. 27 (1), 301–321.
Liu, X., Hieronymus, J.L., Gales, M.J.F., Woodland, P.C., 2013b. Syllable language models for Mandarin speech recognition: exploiting character sequence models. J. Acoust. Soc. Am. 133 (1), 519–528.
Liu, X., Gales, M.J.F., Woodland, P.C., 2012a. Language model cross adaptation for LVCSR system combination. Comput. Speech Lang. 27 (4), 928–942, June 2013.
Liu, X., Gales, M.J.F., Woodland, P.C., 2012b. Paraphrastic language models. In: Proc. ISCA Interspeech2012, Portland, Oregon.
Liu, X., Gales, M.J.F., Woodland, P.C., 2013c. Paraphrastic language models and combination with neural network language models. In: Proc. IEEE ICASSP2013, Vancouver, Canada.
Madnani, N., Dorr, B., 2010. Generating phrasal and sentential paraphrases: a survey of data-driven methods. Comput. Linguist. 36 (3), 2010.
Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., Khudanpur, S., 2010. Recurrent neural network based language model. In: Proc. ISCA Interspeech, Makuhari, Japan, pp. 1045–1048.
Mohri, M., 1997. Finite-state transducers in language and speech processing. Comput. Linguist. 23 (2), 269–311.

Niesler, T.R., Woodland, P.C., 1996. Combination of word-based and category-based language models. In: Proc. ICSLP96', Philadelphia, pp. 220–223.

Niesler, T.R., Whittaker, E.W.D., Woodland, P.C., 1998. Comparison of part-of-speech and automatically derived category-based language models for speech recognition. In: Proc. IEEE ICASSP1998. vol. 1, Seattle, WA, pp. 177–180.

Olive, J., Caitlin, C., McCary, J., 2011. Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation. Springer.

Padmanabhan, M., et al., 1998. Speech recognition performance on a voicemail transcription task. In: Proc. IEEE ICASSP1998, Seattle, WA, pp. 913–916.

Park, J., Liu, X., Gales, M.J.F., Woodland, P.C., 2010. Improved neural network based language modelling and adaptation. In: Proc. ISCA Interspeech2010, Makuhari.

Pasca, M., Dienes, P., 2005. Aligning needles in a haystack: paraphrase acquisition across the web. In: Proc. IJCNLP2005, Jeju Island, pp. 119–130.

Povey, D., Woodland, P.C., 2002. Minimum phone error and I-smoothing for improved discriminative training. In: Proc. IEEE ICASSP2002, Orlando.

Schwenk, H., 2007. Continuous space language models. Comput. Speech Lang. 21, 492–518.

Stolcke, A., 2002. SRILM – An Extensible Language Modeling Toolkit. In: Proc. Proc. ICSLP'02, Denver.

Sundermeyer, M., Schlueter, R., Ney, H., 2012. LSTM neural networks for language modeling. In: Proc. ISCA Interspeech2012, Portland, Oregon.

Tam, Y., Schultz, T., 2008. Correlated bi-gram LSA for unsupervised language model adaptation. In: Proc. NIPS2008, Vancouver, p. 2008.

Woodland, P.C., Gales, M.J.F., Pye, D., Young, S.J., 1996. The development of the 1996 HTK broadcast news transcription system. In: Proc. DARPA Speech Recognition Workshop, NY, US. Arden House, pp. 73–78.