ON-LINE ADAPTATION OF SEMANTIC MODELS FOR SPOKEN LANGUAGE UNDERSTANDING

Ali Orkan Bayer and Giuseppe Riccardi

Signals and Interactive Systems Lab - University of Trento, Italy

{bayer, riccardi}@disi.unitn.it

ABSTRACT

Spoken language understanding (SLU) systems extract semantic information from speech signals, which is usually mapped onto concept sequences. The distribution of concepts in dialogues are usually sparse. Therefore, general models may fail to model the concept distribution for a dialogue and semantic models can benefit from adaptation. In this paper, we present an instance-based approach for on-line adaptation of semantic models. We show that we can improve the performance of an SLU system on an utterance, by retrieving relevant instances from the training data and using them for on-line adapting the semantic models. The instance-based adaptation scheme uses two different similarity metrics edit distance and n-gram match score on three different tokenizations; word-concept pairs, words, and concepts. We have achieved a significant improvement (6% relative) in the understanding performance by conducting re-scoring experiments on the n-best lists that the SLU outputs. We have also applied a two-level adaptation scheme, where adaptation is first applied to the automatic speech recognizer (ASR) and then to the SLU.

Index Terms— Spoken Language Understanding, Recurrent Neural Networks, On-line Adaptation

1. INTRODUCTION

Spoken language understanding (SLU) is the process of extracting semantic information from speech signals. Semantic information can be represented by conceptual constituents which are instantiated by words. SLU systems embody ASR modules and they must rely on the erroneous hypotheses that the ASR outputs. One of the approaches to build robust SLU systems is to use multiple ASR hypotheses rather than a single hypothesis for an utterance. Another component that SLU systems include is the SLU module which aligns word sequences with semantic representations. The reader may refer to [1] for a detailed explanation of SLU systems. Conditional random fields (CRFs) [2] have been successfully applied to SLU alignment. The performance of SLU systems is measured in terms of concept error rate (CER). In this paper, our SLU module consists of an *alignment model*, which is built by using CRFs, and a *scoring model* which assigns posterior probabilities to word-concept alignments. Having posterior probabilities for word-concept alignments enables us to re-score multiple hypotheses for a single utterance.

Neural network LMs (NNLMs), which have been introduced in [3], have gained popularity because of the improvements in computational power. NNLMs project the discrete word space onto a continuous space which results in better smoothing of probability distributions and in this way they do not suffer from data sparseness as much as conventional LMs do [4]. Recurrent neural networks (RNNs) which save the state of the network by using recurrent connections model a short-term memory. In terms of language modeling this short-term memory may be considered as the token history. RNNs are introduced to language modeling in [5], where significant reductions in word error rate and perplexity are reported. In [6] joint LMs that are based on RNNs (RNNLMs) are presented for the estimation of posterior probabilities of word-concept alignments.

LM adaptation has long been applied to ASR systems to improve their performance on a targeted domain. The process involves adapting a background LM by using domain specific data. A general review about statistical LM adaptation is given in [7]. Generally, LM adaptation is applied to conventional n-gram LMs. However, recently there have been studies that apply LM adaptation to neural networks (NNs). One of the approaches that has been applied to RNNLMs is to train the NN for one more iteration with the adaptation data [8]. Information retrieval approaches have been applied to LM adaptation that use *tf* and *idf* statistics for selecting the relevant documents [9, 10]. In [11] an instance-based on-line LM adaptation approach is presented for ASR. In this approach, for each ASR hypothesis relevant instances are selected from the training data and the background NNLM is adapted to the current utterance by using these instances.

In this paper, we present an instance-based on-line adaptation scheme for SLU scoring models. Relevant instances are retrieved from the training data with respect to their similarity to the SLU hypothesis for that utterance. The background RNN scoring model is on-line adapted by using these instances. The n-best list for that utterance is re-scored by us-

This work was partially funded by Portdial FP7 project n. 296170

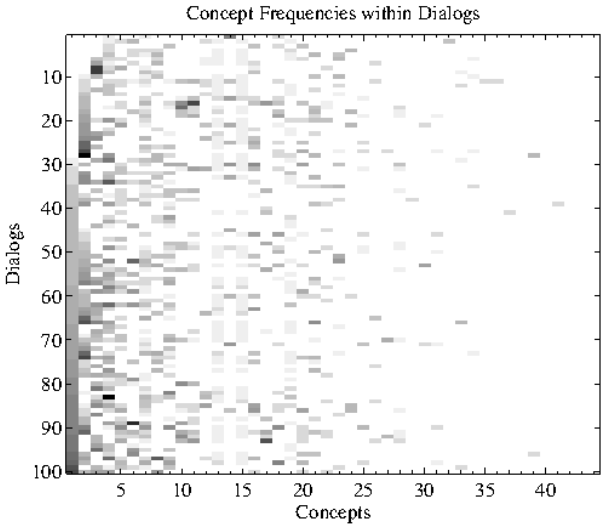

Fig. 1. The concept frequencies within dialogues for the Italian LUNA corpus. The dialogues are randomly selected from the training set. The darker areas show more frequent concepts, whereas the lighter areas show less frequent concepts. The concepts are rank ordered on the x-axis with respect to their frequencies in the training data.

ing the adapted scoring model. We have achieved significant improvements on CER. The next section, gives an overview of the on-line adaptation process for SLU. Section 3 presents the instance-based adaptation procedure on the SLU scoring model. Section 4 reports the experimental results on instance-based on-line adaptation.

2. ON-LINE ADAPTATION FOR SLU

SLU systems may benefit from adaptation as much as ASR systems. LM adaptation has been successfully applied to ASR systems for improving the performance of domain independent LMs on specific domains. In this paper, we present an on-line adaptation scheme for SLU scoring models. Figure 1 shows the distributions of concept frequencies within dialogues that are randomly selected from the training set of Italian LUNA corpus. It can be seen that except for a few concepts that occur very frequently, the general concept distribution is very sparse. Due to this sparsity a general model may fail to capture the distributions well, and adaptation of the model to the target dialogue may yield to improvements in the performance.

The SLU module we have used is composed of a CRF alignment model and a RNN scoring model as depicted in Figure 2. The scoring model estimates the posterior probabilities for word-concept alignments. Therefore, it is possible to re-rank multiple word-concept alignment hypotheses for a single utterance. On-line adaptation can be applied to the scoring model to improve the performance of re-ranking.

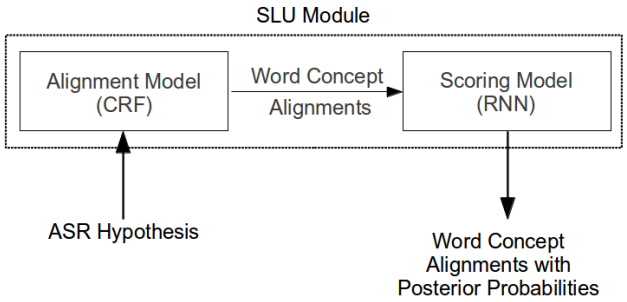

Fig. 2. The structure of the SLU module. It consists of a CRF alignment model, and an RNN scoring model. It takes an ASR hypothesis as input and outputs word-concept alignments with posterior probabilities.

2.1. RNN scoring model for SLU

Joint LMs can be used to assign posterior probabilities to word-concept alignments as given in [6]. In this paper we have constructed a scoring model by using RNNs. The model is similar to the class-based RNN structure that is given in [12], which is also available as a toolkit at http://www.fit.vutbr.cz/~imikolov/rnnlm/. This toolkit is modified to handle manual classes. In this way, we have clustered each word-concept pair semantically with respect to its concept label.

The joint RNN has a node for every word-concept pair at the input layer, which takes the previous token as input. The input is fed to the network by using 1-of-n encoding. The output layer outputs probability distributions for each cluster and for each word-concept pair in that cluster. Therefore, at the output layer the posterior probabilities are factorized into class probabilities and membership probabilities as given in Equation 1, where (w_i, c_i) denotes the ith word-concept pair, h_i denotes the history for the ith pair, and cl_i denotes the semantic class that the ith pair is assigned to. The RNN structure is given in Figure 3.

$$P((w_i, c_i)|h_i) = P(cl_i|h_i)P((w_i, c_i)|cl_i, h_i) \qquad (1)$$

The background RNN scoring model is built by training the RNN on word-concept pairs with the whole training data. The training is performed by using back-propagation through time, which propagates the error through recurrent connections. Adaptation can be applied to this background model by further training the RNN with the adaptation data. In all the experiments, we have re-trained the background RNN for 5 iterations with the adaptation data.

3. INSTANCE-BASED ADAPTATION ON-LINE ADAPTATION

The main component of instance-based adaptation is to retrieve the most similar instances from the training data for

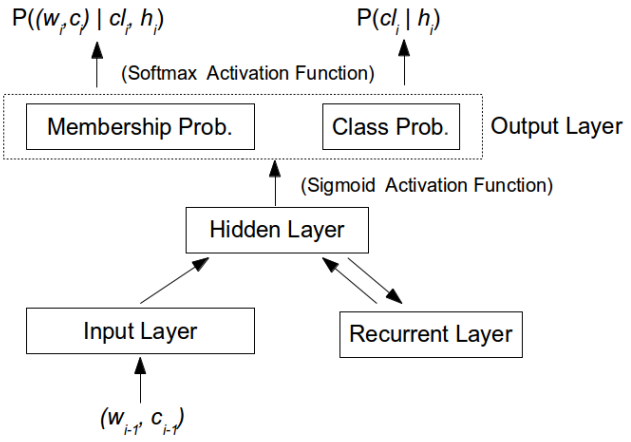

Fig. 3. RNN structure that is used in the scoring model. The input layer has as many nodes as the number of distinct word-concept (w_i, c_i) pairs. The output layer estimates probabilities for all the classes and word-concept pairs. The previous word-concept pair is fed to the input layer using 1-of-n encoding. (w_i, c_i) denotes the ith word-concept pair, cl_i denotes its class, and h_i denotes the history for that pair.

each test utterance. The retrieved instances are then used as adaptation data for the target test utterances. This section first presents the instance retrieval process in detail especially for SLU systems. Then, two different similarity metrics are proposed. Finally, we have provided the on-line adaptation architecture, and showed how it can be applied to a spoken language system.

3.1. Instance retrieval

Instance retrieval searches for the most similar instances from the training data for each hypothesis that the system produces. Therefore, it computes a similarity score between the system hypothesis and each training set instance. The errors that the system introduces in the hypothesis decrease the precision of the similarity scores when these scores are computed on the reference transcription. Thus, to increase the precision, the training data is passed through the SLU system and similarity scores are computed on the system hypotheses for the training data. However, the instances are retrieved from the corresponding reference transcription. In addition, since in general ASR is more precise on meaning bearing words, the words that map to *null* concepts are pruned before the similarity scores are computed. In this paper, for SLU systems, the comparison is performed over three different tokenizations; word-concept pairs, words, and concepts. This process is depicted in Figure 4.

3.2. Similarity metrics

We have used the same similarity metrics given in [6]. The first metric is the *edit distance* in which the hypothesis is

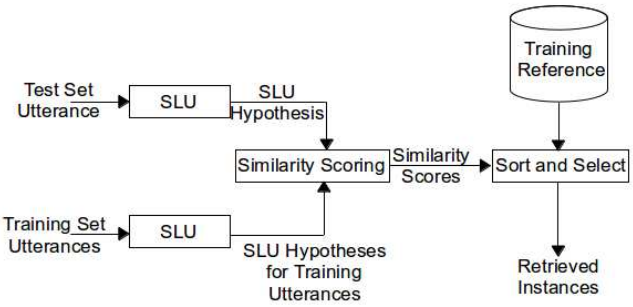

Fig. 4. Instance retrieval process for SLU systems. To compensate errors that SLU produces similarity scores are computed between the SLU hypothesis of the test utterance and the SLU hypotheses of the training set. However, the instances are retrieved from the reference transcription of the training data.

aligned with every utterance in the training data and the total number of errors (deletions, insertions, and substitutions) are computed for each alignment. Then, the instances are sorted in ascending order with respect to the number of errors and they are retrieved from the reference transcription of the training data.

The second metric is the *n-gram match score*, which computes the similarity by considering n-grams. To compute the similarity, each system hypothesis is aligned with the hypotheses of the training data. The score is computed by using Equation 2, where n refers to the number of words in the system hypothesis, ug, bg, and ng refer to matching uni-gram count, matching bi-gram count and matching n-gram count respectively, and ins refers to the number of insertions. The instances are sorted in descending order and instances are retrieved from the reference transcription of the training data.

$$score = (\frac{ug}{n} + \frac{bg}{n-1} + ... + \frac{ng}{1} - \frac{ins}{n})/n \qquad (2)$$

3.3. Instance-based on-line adaptation scheme

The instance-based on-line adaptation procedure can be both applied at the ASR output or at the SLU output. In this approach the system hypothesis (ASR or SLU hypothesis) is used to retrieve the similar instances from the training data and a background model is adapted by using these instances. The ASR can be improved by adapting a word-based background LM, on the other hand, SLU can be improved by adapting a joint scoring model. The general flow of instance-based on-line adaptation is as follows. The first step is to retrieve the relevant instances from the training data. Then, the background model is adapted by using these instances. The n-best hypotheses of the system are re-scored by combining the posterior probabilities of adapted model with acoustic scores. The general flow is depicted in Figure 5.

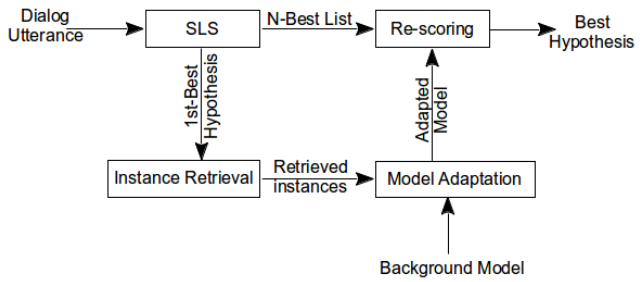

Fig. 5. The general flow of instance-based on-line adaptation scheme. The relevant instances are selected by using the spoken language system (SLS) hypothesis. The background model is adapted by using the retrieved instances. N-best SLS hypotheses are re-scored by using the adapted model.

4. EXPERIMENTAL WORK

The Human-Machine (HM) part of the Italian LUNA conversation corpus [13] is used for testing the performance of instance-based on-line adaptation for SLU tasks. The LUNA corpus is collected by a customer care and technical support center for software and hardware. The HM part is collected with a Wizard of Oz approach. The corpus is split into training, development, and test sets, which include 3171, 387, and 634 utterances respectively. The training set has a vocabulary size of 2399 words and 44 concepts, the out-of-vocabulary rate of the test set is 3.68% on words.

We show the performance of instance-based on-line adaptation by performing re-scoring experiments. N-best SLU hypotheses are re-scored with the adapted RNN scoring model. The first set of experiments (Table 2 and 3) shows the possible upper bounds for the proposed approach. Therefore they are performed by using the reference transcription (Table 2) of the test set utterances and the oracle hypotheses (Table 3) of the SLU system. The actual performance of the instance-based on-line adaptation is presented with second set of experiments (Table 4), which uses the SLU hypothesis of the system. Finally, we have combined ASR adaptation with SLU adaptation (Table 5). In this approach, first the ASR hypothesis is improved by applying on-line LM adaptation and the SLU system uses this hypothesis for the adaptation process.

4.1. Baseline system

The baseline system has two modules; an ASR and a SLU module. The ASR uses hybrid ANN/HMM acoustic models that are adapted to the Italian LUNA corpus. A tri-gram conventional LM with Kneser-Ney smoothing is used as the LM. The ASR uses finite state transducer decoding and outputs lattices. 100-best list is compiled by using those lattices.

The alignment model we have used is based on CRFs. CRFs model the conditional probability of the concept sequence given the word sequence. The first type of features is the orthographic feature. This feature considers the first or last i letters of the word, where i changes between 1 and 5. Another type of feature is the bi-gram feature. For this type we have taken the word bi-grams that consist of previous word and current word, current word and next word, previous word and next word. In addition to these features we have used binary features which label numerical expressions. We have also consider the value of the previous concept when predicting the current one. All these features are independent of each other in the window of [-1, +1]. The CER for the alignment model on the reference transcription of the test set is 21.5%. The scoring model is a joint RNN model which is trained with all the training data on word-concept pairs. This model is used as a background model for instance-based on-line adaptation. The performance of this model with the performance of the baseline system is given in Table 1.

Table 1. Baseline performance. Oracle CER is given for the 100-best list. The baseline performance of the background RNN scoring model is given for 100-best list re-scoring.

	CER
1-best	26.7%
Oracle on 100-best list	18.3%
100-best re-scored with the RNN model	26.1%

4.2. Upper bounds for instance-based on-line adaptation

This section presents the upper bounds by using both the reference transcription of the test set and the oracle hypothesis of SLU system. The instance retrieval process differs at the similarity computation for these experiments. The utterances do not include any errors when the reference transcription is used or they are at minimum when the oracle hypothesis is used. Therefore, similarity scores are computed with the reference transcription of the training data rather than the SLU hypotheses. Table 2 gives the performance of the system with the reference transcription and Table 3 presents the performance for the oracle hypothesis.

As can be seen from the results a significant improvement can be achieved when instance-based on-line adaptation is applied to the SLU model. In general we can see that the performance is best when similarity is computed at *concept* tokens. Additionally, using the oracle hypothesis yields better performance than using the reference transcription. We can obtain 10.8% relative (2.9% absolute) improvement on CER with respect to the baseline when oracle hypothesis is used and similarity is computed at the *concept* level with *n-gram match* score.

4.3. Performance of instance-based on-line adaptation

In this section we present actual performance of the instance-based on-line adaptation on the SLU model. Therefore, instance retrieval is performed by using the SLU hypothesis of

Table 2. CER upper bounds when using the reference transcriptions as input to instance retrieval. "Ins." refers to the number of instances that are retrieved; 3, 9, 16, 31, and 158 corresponds to 0.1%, 0.3%, 0.5%, 1.0%, and 5.0% of the number of training utterances. "wc pr." refers to word-concept pairs. "conc." refers to concept tokenization. "ng match" refers to the n-gram match score.

Ins.	Edit dist.			ng match		
	wc pr.	words	conc.	wc pr.	words	conc.
1	25.2%	25.7%	25.0%	24.9%	25.6%	25.4%
3	24.8%	**24.8%**	25.1%	25.3%	25.3%	24.8%
9	24.8%	25.1%	**24.4%**	**25.2%**	25.8%	**24.4%**
16	24.9%	25.6%	**24.4%**	**25.2%**	25.4%	24.7%
31	24.9%	25.1%	24.7%	25.3%	**25.2%**	24.7%
158	**24.5%**	25.6%	25.2%	25.7%	26.1%	25.3%

Table 3. CER upper bounds when using the oracle hypotheses as input to instance retrieval.

Ins.	Edit dist.			ng match		
	wc pr.	words	conc.	wc pr.	words	conc.
1	25.3%	25.3%	25.1%	24.8%	25.1%	25.6%
3	**24.6%**	25.1%	25.2%	24.7%	**24.7%**	25.1%
9	**24.6%**	25.3%	**24.1%**	**24.6%**	24.9%	24.7%
16	24.7%	25.1%	24.2%	**24.6%**	25.0%	24.2%
31	24.8%	**24.7%**	24.3%	24.8%	24.9%	**23.8%**
158	24.8%	25.1%	24.8%	25.4%	25.7%	25.3%

the system. As we have mentioned, to compensate for the errors that SLU hypothesis possesses we have used the SLU hypotheses of the training data when computing the similarity scores. The performance of this approach is given in Table 4.

The results show that when the instance-based on-line adaptation is applied to the SLU model, it gives significant improvements on CER. When these results are compared to the upper bounds we can see that there is still a huge possible improvement. In addition to that, when similarity computation scores over *word* tokens give the worst performance with the reference transcription and the oracle hypothesis, they perform the best when actual SLU hypothesis is used for instance retrieval. Also *concept* tokens perform the worst which is not the case with the upper bounds. This is most likely due to the fact that the SLU hypothesis has more errors on *concept* tokens when compared to *word* tokens. We have obtained 6.0% relative (1.6% absolute) performance improvement on CER with respect to the baseline system.

4.4. Two-level application of adaptation

It is also possible to apply the instance-based on-line adaptation at two different levels aiming at first improving the ASR hypothesis and then the SLU hypothesis. The motivation behind this is to see if we can benefit from ASR and SLU adaptation schemes in combination. Therefore the following pipeline is used. The utterances are first fed into ASR, the ASR n-best list is re-scored by applying instance-based on-line adaptation to a NNLM background model as given in [6]. The improved ASR hypothesis and the n-best list is fed into the SLU model and concept representations are extracted. As the final step, on-line adaptation is applied to the SLU model by using the SLU hypothesis that is obtained with the improved ASR and the n-best list is re-scored by using the adapted SLU model.

Table 4. CER on-line adaptation performances. The instances are retrieved by using the SLU hypothesis of the system for each utterance. These results must be compared to Table 1.

Ins.	Edit dist.			ng match		
	wc pr.	words	conc.	wc pr.	words	conc.
1	26.0%	26.1%	26.0%	25.7%	25.9%	26.1%
3	26.3%	25.8%	26.6%	25.4%	**25.2%**	26.2%
9	**25.3%**	25.2%	**25.7%**	**25.3%**	25.3%	25.9%
16	**25.3%**	**25.1%**	25.9%	25.6%	25.8%	26.2%
31	25.5%	25.2%	26.1%	25.7%	25.6%	25.7%
158	**25.3%**	26.2%	26.0%	25.8%	26.1%	**25.6%**

Table 5. The performance (CER) of the full on-line adaptation pipeline, where first the ASR hypothesis is improved by using instance-based on-line LM adaptation. Then the improved ASR hypothesis is used with the SLU model to apply the on-line adaptation process to the SLU model.

Ins.	Edit dist.			ng match		
	wc pr.	words	conc.	wc pr.	words	conc.
1	26.0%	25.8%	25.7%	25.8%	26.0%	25.8%
3	26.1%	26.2%	25.4%	25.6%	25.8%	**25.3%**
9	**25.2%**	**25.7%**	**25.3%**	**25.5%**	26.0%	**25.3%**
16	**25.2%**	**25.7%**	**25.3%**	25.9%	**25.7%**	25.8%
31	25.9%	26.1%	25.5%	25.7%	**25.7%**	25.6%
158	25.7%	**25.7%**	26.1%	26.2%	26.3%	26.2%

The performance of the full pipeline is given in Table 5. When compared to Table 4, where adaptation is only applied to the SLU model, we cannot see a significant improvement. Also, we can see a drop in the performance on *word* tokens. On the contrary, the performance of *concept* tokens are slightly improved, probably due to the fact that with the improved ASR hypothesis the errors on *concept* tokens have decreased.

4.5. Statistical significance of the results

This sections shows that achieved improvements on CER by using instance-based on-line adaptation for the SLU model are statistically significant with respect to the baseline system. We compare the performance of the baseline

system with the best performing on-line adaptation system (Table 4) with the two similarity metrics on *word* tokens. The *bootstrap-t* confidence intervals are calculated by using bootstrap method that is given in [14]. In addition, p-values are calculated by using the randomization method given in [15] which is implemented in the toolkit that is available at http://www.nlpado.de/~sebastian/software/sigf.shtml. As can be seen from Table 6 the improvements on CER are statistically significant since p-values are smaller than 0.05.

Table 6. The comparison of the baseline system with the instance-based on-line SLU model adaptation. 90% confidence intervals using 10^4 bootstrap replications are given in brackets. Also p-values for the comparison between the baseline and the two approaches are given. The results show that the improvements are significant.

	CER	p-value
Baseline	26.7% [24.2 - 29.2]	NA
Edit dist. best	25.1% [22.7 - 27.5]	0.01
n-gram match best	25.2% [22.8 - 27.7]	0.03

5. CONCLUSION

In this paper, we have presented an instance-based on-line adaptation scheme that is aimed at improving the performance of SLU systems. The main idea behind instance-based on-line adaptation is to select relevant instances from the training data by using the hypothesis that the system outputs for each utterance. These instances are then used to adapt the model that will be used for re-scoring. We have achieved significant improvements on CER for SLU by using *word* tokens with the *edit distance* and the *n-gram match score* metrics. However, there is still a huge possibility of improvement as the upper bounds show. The two-level adaptation scheme, in which first the ASR hypothesis is improved by performing LM adaptation and then the SLU model is adapted by using this hypothesis, does not bring any additional improvements.

6. REFERENCES

[1] R. De Mori, F. Bechet, D. Hakkani-Tur, M. McTear, G. Riccardi, and G. Tur, "Spoken language understanding," *Signal Processing Magazine, IEEE*, vol. 25, no. 3, pp. 50–58, 2008.

[2] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of ICML*. 2001, pp. 282–289, Morgan Kaufmann.

[3] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2000.

[4] H. Schwenk, "Continuous space language models," *Computer Speech Language*, vol. 21, no. 3, pp. 492–518, 2007.

[5] T. Mikolov, M. Karafiat, L. Burget, J. Cernock, and S. Khudanpur, "Recurrent neural network based language model," in *Proceedings of Interspeech*. 2010, pp. 1045–1048, ISCA.

[6] A. O. Bayer and G. Riccardi, "Joint language models for automatic speech recognition and understanding.," in *Proceedings of Spoken Language Technology Workshop (SLT)*. 2012, IEEE.

[7] J. R. Bellegarda, "Statistical language model adaptation: review and perspectives," *Speech Communication*, vol. 42, pp. 93–108, 2004.

[8] S. Kombrink, T. Mikolov, M. Karafiat, and L. Burget, "Recurrent neural network based language modeling in meeting recognition," in *Proceedings of Interspeech*. 2011, pp. 2877–2880, ISCA.

[9] A. Aamodt and E. Plaza, "Case-based reasoning; foundational issues, methodological variations, and system approaches," *AI Communications*, vol. 7, no. 1, pp. 39–59, 1994.

[10] M. Eck, S. Vogel, and A. Waibel, "Language model adaptation for statistical machine translation based on information retrieval," in *Proceedings of LREC 2004*, 2004.

[11] A. O. Bayer and G. Riccardi, "Instance-based on-line language model adaptation," in *Proceedings of Interspeech*. 2013, ISCA.

[12] T. Mikolov, S. Kombrink, L. Burget, J. Cernock, and S. Khudanpur, "Extensions of recurrent neural network language model.," in *Proceedings of ICASSP*. 2011, pp. 5528–5531, IEEE.

[13] M. Dinarelli, S. Quarteroni, S. Tonelli, A. Moschitti, and G. Riccardi, "Annotating spoken dialogs: from speech segments to dialog acts and frame semantics," in *Proceedings of SRSL 2009 Workshop of EACL*, Athens, Greece, 2009.

[14] M. Bisani and H. Ney, "Bootstrap estimates for confidence intervals in asr performance evaluation," in *Proceedings of ICASSP*, 2004.

[15] A. Yeh, "More accurate tests for the statistical significance of result differences," in *Proceedings of the 18th conference on Computational linguistics - Volume 2*, Stroudsburg, PA, USA, 2000, COLING '00, pp. 947–953, Association for Computational Linguistics.