HAL open science

Adapting Language Models When Training on Privacy-Transformed Data

Mehmet Ali Tugtekin Turan, Dietrich Klakow, Emmanuel Vincent, Denis Jouvet

To cite this version:

Mehmet Ali Tugtekin Turan, Dietrich Klakow, Emmanuel Vincent, Denis Jouvet. Adapting Language Models When Training on Privacy-Transformed Data. LREC 2022 - 13th Language Resources and Evaluation Conference, Aug 2021, Brno, Czech Republic. hal-03189354v1

HAL Id: hal-03189354

https://inria.hal.science/hal-03189354v1

Submitted on 3 Apr 2021 (v1), last revised 8 May 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adapting Language Models When Training on Privacy-Transformed Data

M. A. Tuğtekin Turan[1], *Dietrich Klakow*[2], *Emmanuel Vincent*[1], *and Denis Jouvet*[1]

[1]Université de Lorraine, CNRS, Inria, Loria, F-54000, Nancy, France
[2]Spoken Language Systems Group, Saarland University, Germany

{tugtekin.turan | emmanuel.vincent | denis.jouvet}@inria.fr,
dietrich.klakow@lsv.uni-saarland.de

Abstract

In recent years, voice-controlled personal assistants have revolutionized the interaction with smart devices and mobile applications. These dialogue tools are then used by system providers to improve and retrain the language models (LMs). Each spoken message reveals personal information, hence, it is necessary to remove the private data from the input utterances. However, this may harm the LM training because privacy-transformed data is unlikely to match the test distribution. This paper aims to fill the gap by focusing on the adaptation of LM initially trained on privacy-transformed utterances. Our data sanitization process relies on named-entity recognition. We propose an LM adaptation strategy over the private data with minimum losses. Class-based modeling is an effective approach to overcome data sparsity in the context of n-gram model training. On the other hand, neural LMs can handle longer contexts which can yield better predictions. Our methodology combines the predictive power of class-based models and the generalization capability of neural models together. With privacy transformation, we have a relative 11% word error rate (WER) increase compared to an LM trained on the clean data. Despite the privacy-preserving, we can still achieve comparable accuracy. Empirical evaluations attain a relative WER improvement of 8% over the initial model.

Index Terms: language model adaptation, privacy-preserving learning, speech recognition, class-based language modeling.

1. Introduction

Spoken dialogue systems aim to identify user's intents expressed in natural language to satisfy those requests. In the first step of any system, the input utterance is recognized with an automatic speech recognizer (ASR). However, the state-of-the-art data-driven ASR technology utilizes large amounts of speech that is collected without any privacy concerns [1]. With the growing public awareness such as the European Union's General Data Protection Regulation (GDPR), storing the user data raises serious privacy concerns [2]. It might even contain critical information such as passwords, credit card numbers, or even health status. Therefore, a spoken message which contains sensitive information about the user characteristics should not be centralized in a single place.

Hiding private information creates a bottleneck in building an accurate ASR system. Using high-quality labeled data has extreme importance for any machine learning task. However, preserving privacy while sharing data is important since such data may contain confidential information. This paper uses a data sanitization approach over the named-entity recognition to transform personal information. Entities such as person, location, and organization names are first transformed with random tokens. The challenge in this sanitization is ensuring that the performance of the dialogue system trained using the sanitized data should be as good as the ones before the transformation. In this work, we employ a mixture of word- and neural-based LMs alongside with the class-based model to represent the private data better, where each named-entity category corresponds to one class [3].

The simplest form of privacy-transformation is to modify the values of named-entities or replacing them with generic tokens. If they are not already marked during transcription or labeling, one can utilize automatic entity extraction methods, which are well-studied in the computational linguistics area [4]. Our approach consists of labeling the named-entities in the given utterance and then hiding them by using a word-by-word replacement type of data sanitization [5]. In this work, we train our LMs over the privacy-transformed text where we replace the named-entities with random values from the same entity category. Partly inspired by successful research in the field of ASR, various forms of class-based LMs have been shown to improve the recognition quality when used in combination with standard word-level LMs [6]. Following this idea, our work investigates an interpolation between different LM types.

Class-based LMs have proved their success for training on a small dataset for fast LM adaptation [7]. By grouping words with similar distributional behavior into equivalent classes, class-based LMs have fewer parameters to train and can make predictions based on longer histories [8]. This makes them particularly attractive in situations where word-based n-gram coverage is low due to a shortage of training data. Moreover, it has been found that neural- and word-based contributions are complementary and interpolation between these models usually leads to the best results [9]. In this paper, we also integrate the long short-term memory (LSTM) based LMs, which have the ability to model longer temporal dependencies than n-grams and traditional neural-based LMs [10].

Language models (LMs) inside the ASR task are typically trained on a text corpora from similar domains with the target test dataset. These types of corpora, sometimes, are unlikely to match the test distributions, which results in lower performance for spoken test utterances [11]. Adaptation attempts to adjust the parameters of any LM so that it will perform well on target domain data. Therefore, the LM is commonly adapted using a smaller held-out in-domain dataset that matches with the test distribution [12]. In particular, we focus on the cross-domain LM adaptation paradigm, that is, to adapt an LM trained on one domain (here the anonymized background domain) to a different domain (e.g. adaptation domain), for which only a small amount of non-anonymized data is available.

In ASR, word context is heavily influenced by the domain, which is mostly characterized by the conversational topic and speaking style as well. Generally, interpolation between several

LMs provides implicit modeling of the domain. However, it has been found that the adaptation of LMs to small amounts of matched in-domain text data can yield a decrease in both perplexity (PPL) and word error rate (WER) [13]. This work, therefore, investigates adaptation strategies for the initially trained LMs. Specifically for word- and class-based LMs, we study fast marginal adaptation (FMA) by combining adapted uni-grams with tri-grams trained on a background corpus [14]. For the LSTM-based scheme, we implement a "pre-train and fine-tune" methodology as an adaptation strategy [15].

In this paper, our main contribution is to present a framework where anonymous (privacy-transformed) data can be used inside the LM training. In doing so, we selected methodologies that are suitable for on-the-fly running with mobile devices using as few resources as possible. Applying the class-based idea, we were able to represent anonymous data better via named-entities. Thanks to linear interpolation over the adapted LMs, it is possible to recover some performance we lost because of data anonymization.

2. Methodology

Our methodology consists of a two-stage adaptation scheme. After getting an anonymized data, the first part performs a generic LM training with several models. Then, the next stage introduces the LM adaptation using a small amount of in-domain clean (non-anonymized) data. Figure 1 illustrates a general overview of the proposed adaptation approach. The following subsections present the major components of our methodology.

2.1. Privacy-Transformation

Since the LM adaptation depends critically on the quality of background data, we first review how to gather effective anonymization for a clean input. Following the named-entity recognition of the CoNLL's shared task [16], we consider anonymization that spans utterances annotated with one of these four labels: persons (*PER*), organizations (*ORG*), locations (*LOC*), and miscellaneous names such as date or time (*MISC*). We do not identify demographic attributes like gender, age, and ethnicity of individuals, or any other potentially private information.

At the final stage, the overall anonymization is performed by the same-type transformation strategy (word-by-word) [5]. In this alternative representation, even in the cases when identification of private information fails to detect a relevant word occurrence, an attacker cannot easily distinguish whether these words are the result of an actual transformation or not.

Once we achieve privacy-transformed text, individual LMs are trained on top of anonymized data. To train class-based language models, we first find named-entities in a given text and replace them with their category tags like `LOC` or `MISC`. This yields a more realistic scenario in real-world applications. In other words, named-entities are presented to the class-based LM in an unsupervised manner.

2.2. Modeling Word Classes

In this paper, we employ a class-based strategy into our fast LM adaptation framework. Some words are similar to other words in their meaning or syntactic function. In the context of text sanitization described above, an anonymized group of words under a particular entity-tag can be considered close to each other compared to their inter-class relations. If we can successfully assign words to classes, it may be possible to make more reasonable predictions for histories that have private information by assuming that they are similar to other histories that we have seen.

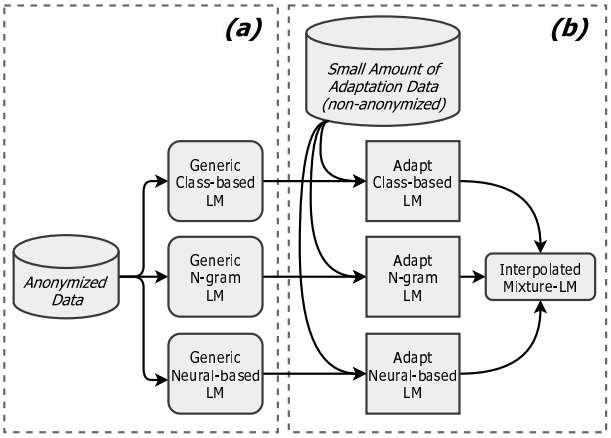

Figure 1: *Block diagram of the proposed adaptation scheme. (a) corresponds to initial generic LMs using privacy-transformed text data; (b) corresponds to the adaptation of target domain using a limited amount of the clean (non-anonymized) text data.*

Given the anonymized input text, the class-related probabilities can be estimated by the maximum likelihood principle which simply counts the number of occurrences of the word divided by the number of all words in that word class. Moreover, conditional class probabilities can be calculated similarly. The only difference is that the word sequences used for training must be converted to class sequences.

Specifically, we propose a general way of incorporating class-based LMs with anonymized word-to-class mapping into the finite-state transducer (FST) framework. The FST composition allows handling the class-based LM in the first decoding pass. A class-based LM can be represented by a composition of two FSTs, namely class-map and n-gram of the class sequence. In our implementation, we can replace the word-based LM by cascading class pair mapping and n-gram LM based on word-classes instead of words.

Similarly in [17], we first train an LM transducer with class entries mapped to non-terminal string identifiers like `<LOC>` or `<PER>`. Then, sub-language models for each word-class are trained. At the final stage, we insert these sub-language model WFSTs together to acquire the final LM transducer.

2.3. Adaptation of Language Models

After training the individual LMs at Stage (a) depicted in Figure 1, we apply two different adaptation strategies at Stage (b). For word- and class-based LMs, we focus on using uni-gram LMs to adapt tri-grams. This approach combines uni-gram and tri-gram information inspired by the fast marginal adaptation (FMA) idea proposed in [18]. As an initial distribution, it uses the tri-gram trained on the background corpus. The desired tri-gram has to satisfy that its marginal is the uni-gram trained on the adaptation data. For a given word, w, with a history, h, the adapted LM, $P_A(w|h)$, satisfies the following constraint,

$$P_A(w|h) = \frac{1}{Z(h)} \left(\frac{P_A(w)}{P_B(w)}\right)^{\beta} P_B(w|h) \qquad (1)$$

where $P_B(w)$ denotes the background LM and β is a weighting parameter. An efficient way of calculating the normalization factor, $Z(h)$, is explained in [18]. The general idea of this adaptation scheme can be summarized as follows. The first scaling factor, $\frac{P_A(w)}{P_B(w)}$, captures certain words up or down that are more/less frequent in the adaptation data than in the background corpus. If the ratio is one nothing changes that also adds robustness to the overall method.

For the neural-based LSTM modeling, we adopt a fine-tuning idea where we first train a background LM on the entire training set. Then, we use this converged model to initialize the adaptation stage. The ultimate adaptation is performed by fine-tuning the cluster soft-max layer. In other words, some layers are frozen and their weight matrices are not updated during back-propagation. Only the weights of unfrozen layers get fine-tuned.

2.4. N-gram Approximation of Neural Language Models

This paper uses an interpolated mixture LM for final decoding. We use Kaldi[1] speech recognition toolkit for all our experiments. The decoding can be easily done for n-gram originated word- and class-based LMs because of their ARPA-style format which is a default usage inside Kaldi. For LSTM-based neural LM, it is possible to use it for lattice scoring (in a second processing pass). However, we also employ an n-gram approximation to use LSTM-LM into the first-pass decoding to avoid introducing delay.

In [19], several approximation techniques are presented. In the reported experiments, we used an updated version of the probability-based conversion technique which provided better performance for our experiments [20]. For every word, w_i, of the anonymized training corpus, and associated history, h, corresponding to uni-grams, bi-grams, tri-grams, we compute the neural-based LM probability. Then, these values are averaged (if multiple occurrences exist) and normalized to obtain an approximated probability distribution. By doing so, we produce an n-gram ARPA LM, which is later used in the first-pass.

3. Experimental Setup

We evaluate our proposed LM adaptation scheme using the Augmented Multiparty Interaction (AMI) corpus containing multi-hour meeting recordings. These meetings were recorded as part of the AMI/AMIDA projects[2] co-directed by the University of Edinburgh and Idiap Research Institute. The AMI Meeting Corpus is a collection of data captured in specially instrumented meeting rooms, which record the multimodal signals (audio and video) for each participant. We partition the AMI data into training, adaptation, and test sets ensuring that no speaker appears in more than one set. Also, we only use speech data recorded with individual headset microphones. The following table presents some statistics of the data where the average utterance length is 7.5 words.

In this split, the adaptation set represents around 12% of the training data. Other splits are also investigated in the experimental evaluations to measure the impact of larger adaptation sets (representing 15% and 20% of the training data). Note that, size of the test set is the same in all cases.

For the named-entities, we use annotated tags given by AMI named-entity instructions which mainly follow the hierarchical structure of the NIST task definition [21]. To identify the named-entities in the anonymized text, we utilized an open-source software library called spaCy[3], which also comes with pre-trained pipelines. Eventually, we obtained $2,167$ unique entity tags including 226 for `LOC`, 489 for `PER`, 515 for `MISC`, and 937 for `ORG`.

Table 1: *Some statistics of the training, adaptation and test sets, from the AMI corpus. The overall duration is* 100 *hours, yet the unique vocabulary is limited because of its dialogue nature.*

Set	Dur. (min.)	Utterance	Number of Words	
			Unique	Occurence
Train	4,880	108,221	11,882	802,604
Test	580	13,059	4,145	94,914
Adaptation	531	12,612	3,913	89,635
All	*5,991*	*133,892*	*13,079*	*987,153*

During the experimental evaluation, we use word error rate (WER) and perplexity (PPL) as objective metrics. For all results presented in this paper, matched pairs sentence-segment word error (MAPSSWE) based statistical significance test was performed at a significance level of $\alpha = 0.05$. The MAPSSWE is essentially a parametric t-test for estimating the mean difference of normal distributions with unknown variances [22]. The "sc_stats" tool from NIST[4] is used to compute the MAPSSWE test.

For our experiments, we employ an acoustic modeling (AM) which is based on time-delay neural network (TDNN) architecture inside Kaldi's chain models. The TDNN-based AM operates on 40-dimensional Mel-frequency cepstral coefficient (MFCC) features extracted from frames of 25ms length and 10ms stride, and is similar to the model specified in [23]. The speed-perturbation technique of [24] is also used with a 3-fold augmentation where copies of training data are created according to factors of 0.9, 1.0, and 1.1.

4. Results and Discussion

4.1. Baseline Performance

We first compare the generic LMs trained at Stage (a) of Figure 1 without any adaptation or interpolation methods. These individual LMs are the following: (M1) 3-gram word-based LM in first-pass decoding; (M2) 3-gram class-based LM again in first-pass decoding; (M3) a 3-gram approximation of the LSTM-based model in the first-pass decoding; and finally, (M4) a re-scoring of the word lattice hypotheses with the LSTM-LM (the lattices result from first-pass decoding using a 3-gram word-based LM).

Table 2 shows baseline results where all these evaluations are performed over the clean test data (non-transformed). The first part of this table presents the performance of generic LMs trained over the anonymized input text, whereas the last two column presents the LM results trained on the original (non-transformed) training data (i.e., before applying the anonymization process). For anonymized training data, best

[1] Kaldi ASR Toolkit: `https://www.kaldi-asr.org`
[2] AMI Project Consortium: `http://www.amiproject.org`
[3] spaCy Library: `https://github.com/explosion/spaCy`
[4] NIST Toolkit: `https://github.com/usnistgov/SCTK`

Table 2: *Target WER and PPL performances using the individual LMs trained using either privacy-transformed (anonymized) or original (non-transformed) data.*

Model	Anonymized Data		Original Data	
	WER [%]	**PPL**	**WER [%]**	**PPL**
[M1]	32.3	121	28.8	82
[M2]	**30.2**	103	29.3	74
[M3]	32.9	137	29.1	88
[M4]	30.5	103	**27.6**	73

results are obtained using either the class-based LM, *M2*, in the first pass decoding, or through re-scoring with the LSTM-LM in *M4*. These experiments help us to understand the impact of the anonymization process. We see a large degradation between the models trained on original and anonymized data due to the privacy-transformation. For example, estimating the 3-gram word-based model, *M1*, on anonymized data leads to around 11% relative WER degradation, compared to training it on original data. The best results are obtained when a second-pass is applied for re-scoring hypotheses, *M4*, but this increases the computational requirements and induces an extra delay before getting the ASR output.

4.2. Effect of the Adaptation Data Size

Using a limited amount of non-anonymized data, we observe the benefit of adapting LMs initially trained on a large set of anonymized data. Both WER and PPL results have improvements for each type of modeling compared to the baseline models in Table 2. Note that the performance is increased when the amount of additional non-anonymized data gets larger in the case of 15% and 20% data size. In all cases, the class-based model, *M2*, outperforms all other.

Table 3: *WER and PPL performances when adapting the LMs with various amounts of adaptation data (non-anonymized).*

Model	Size: 12%		Size: 15%		Size: 20%	
	WER	**PPL**	**WER**	**PPL**	**WER**	**PPL**
[M1]	31.5	109	31.2	98	31.0	93
[M2]	**29.9**	94	**29.8**	91	**29.7**	86
[M3]	30.8	101	30.6	94	30.3	90
[M4]	30.1	95	29.9	91	29.8	85

For the smallest size considered here, the performance of our proposed LM adaptation schemes still gives improved performance. In this context, it is important to conclude that the adaptation performs useful even when less amount the additional data exists. Indeed, it is not feasible to obtain a large adaptation set for more practical implementations.

4.3. Interpolation of the Adapted LMs

At the final stage, our methodology proposes a mixture LM for a final decoding. Note that we utilize the default adaptation split in Table 1 (corresponding to 12% of the training size) for our interpolation experiments. Thus, a linear interpolation to the previously adapted LMs is employed with the best weight combinations for 3-gram class- and word-based LMs ($\lambda_w = 0.3$ and $\lambda_c = 0.7$). Table 4 presents the results of this experiment on the first line. For both first- and second-pass decoding schemes, we utilize $\lambda_n = 0.4$ for the LSTM-LM interpolation,

$$[(1-\lambda_n) * (\lambda_w * P_A^w + \lambda_c * P_A^c)] + \lambda_n * P_A^n \qquad (2)$$

where P_A^w, P_A^c, and P_A^n denote the adapted word-, class-, and neural-based LMs.

Table 4: *The WER and PPL performances after the linear interpolation experiments of the adapted LMs.*

Model	Description	WER	PPL
[M1 + M2]	word- and class-based	29.7	95
[M1 + M2 + M3]	+LSTM (3-gram app.)	29.6	92
[M1 + M2 + M4]	+LSTM (2nd-pass dec.)	**29.4**	86

We obtain the best results with the interpolation of neural-based LMs at the final stage. However, it should be noted that combining word- and class-based LMs in the first place also achieves modest results. We believe that in larger and more challenging datasets, the contribution of neural-LMs will be greater.

5. Conclusions

This paper proposes LM adaptation schemes over the privacy-transformed text under an ASR task. To train an anonymized text, we first apply data sanitization techniques using the named-entities. Our models are evaluated for the AMI dialogue corpus, where we partitioned the data by distinct training, test, and adaptation splits. We present an LM adaptation method using a class-based formulation by modifying WFSTs over the sanitized data. The word classes are automatically determined by a named-entity recognizer and linearly interpolated with the word- and neural-based LMs together to achieve the best results. We also investigate the adaptation of our LMs over a small piece of non-transformed adaptation data. Our methods prove that the adaptation is still effective with fewer amounts of data. Eventually, we show that, by hiding task-dependent named-entities, we can preserve the privacy of the speakers, and still achieve comparable ASR performance with the ones before the privacy-transformation. As a future direction, the same ideas are also extendable to a more challenging text with a large set of entity modeling. One can evaluate the proposed LM adaptation strategy to hide other private information (e.g. gender), or other topics that span diverse domains such as finance, healthcare, and politics.

6. Acknowledgements

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program under the grant number 825081 (COMPRISE)[5]. Experiments were carried out using the Grid5000[6] testbed, supported by a scientific interest group hosted at CNRS, INRIA, RENATER and several universities as well as other organizations.

[5] https://www.compriseh2020.eu
[6] https://www.grid5000.fr

7. References

[1] M. Tang, D. Hakkani-Tür, and G. Tür, "Preserving privacy in spoken language databases," in *International Workshop on Privacy and Security Issues in Data Mining*, 2004.

[2] A. Nautsch, C. Jasserand, E. Kindt, M. Todisco, I. Trancoso, and N. Evans, "The GDPR & speech data: Reflections of legal and technology communities, first steps towards a common understanding," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2019.

[3] P. F. Brown, V. J. Della Pietra, P. V. Desouza, J. C. Lai, and R. L. Mercer, "Class-based n-gram models of natural language," *Computational linguistics*, vol. 18, no. 4, pp. 467–480, 1992.

[4] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[5] D. I. Adelani, A. Davody, T. Kleinbauer, and D. Klakow, "Privacy guarantees for de-identifying text transformations," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2020.

[6] W. Naptali, M. Tsuchiya, and S. Nakagawa, "Topic-dependent class-based n-gram language model," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 5, pp. 1513–1525, 2012.

[7] C. Samuelsson and W. Reichl, "A class-based language model for large-vocabulary speech recognition extracted from part-of-speech statistics," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1999.

[8] A. Axelrod, Y. Vyas, M. Martindale, and M. Carpuat, "Class-based n-gram language difference models for data selection," in *International Workshop on Spoken Language Translation*, 2015.

[9] S. R. Gangireddy, P. Swietojanski, P. Bell, and S. Renals, "Unsupervised adaptation of recurrent neural network language models." in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2016.

[10] W. De Mulder, S. Bethard, and M.-F. Moens, "A survey on the application of recurrent neural networks to statistical language modeling," *Computer Speech & Language*, vol. 30, no. 1, pp. 61–98, 2015.

[11] J. Gao, H. Suzuki, and W. Yuan, "An empirical study on language model adaptation," *ACM Transactions on Asian Language Information Processing*, vol. 5, no. 3, pp. 209–227, 2006.

[12] I. McGraw, R. Prabhavalkar, R. Alvarez, M. G. Arenas, K. Rao, D. Rybach, O. Alsharif, H. Sak, A. Gruenstein, F. Beaufays *et al.*, "Personalized speech recognition on mobile devices," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016.

[13] X. Chen, T. Tan, X. Liu, P. Lanchantin, M. Wan, M. J. Gales, and P. C. Woodland, "Recurrent neural network language model adaptation for multi-genre broadcast speech recognition," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2015.

[14] D. Klakow, "Language model adaptation for tiny adaptation corpora," in *International Conference on Spoken Language Processing*, 2006.

[15] M. Ma, M. Nirschl, F. Biadsy, and S. Kumar, "Approaches for neural-network language model adaptation." in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017.

[16] E. T. K. Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in *Proceedings of Natural Language Learning at HLT-NAACL*, 2003.

[17] A. Horndasch, C. Kaufhold, and E. Nöth, "How to add word classes to the kaldi speech recognition toolkit," in *International Conference on Text, Speech, and Dialogue*. Springer, 2016.

[18] R. Kneser, J. Peters, and D. Klakow, "Language model adaptation using dynamic marginals," in *European Conference on Speech Communication and Technology*, 1997.

[19] H. Adel, K. Kirchhoff, N. T. Vu, D. Telaar, and T. Schultz, "Comparing approaches to convert recurrent neural networks into backoff language models for efficient decoding," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2014.

[20] M. Singh, Y. Oualil, and D. Klakow, "Approximated and domain-adapted lstm language models for first-pass decoding in speech recognition," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017.

[21] N. Chinchor, E. Brown, L. Ferro, and P. Robinson, "Named entity recognition task definition," *Mitre and SAIC*, 1999.

[22] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1989.

[23] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2015.

[24] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2015.