# LANGUAGE MODEL ADAPTATION FOR TRANSCRIPTION OF BANKING PROTOCOLS

*I. Kraljevski, D. Hirschfeld*

*voice INTER connect GmbH, Ammonstraße 35, D-01067 Dresden, Germany*
*{kraljevski, hirschfeld}@voiceinterconnect.de*

**Abstract:** This paper presents an approach for adaptation of a LVCSR system on a specific domain - speech transcriptions for automated protocol generation during investment consultations. Because of the small amount of available domain-specific speech and textual data, it is not possible to create reliable statistical language model, therefore, word categories containing synonyms were used to train a word-class based model. To provide an appropriate domain-specific textual corpus for language model training, data augmentation was employed by creation of grammar rules and generation of large number of "artificial" sentences. Such language model could be used as standalone or could be merged with the general model. Recognition performance was compared across different language models: the domain-specific model, the general purpose model and as well as their weighted combinations. The results justified the proposed approach for domain-specific language modeling on banking protocols transcriptions.

## 1 Introduction

In this paper, a procedure for adaptation of a general purpose Large Vocabulary Continuous Recognition System (LVCRS) on a specific domain is presented. The target domain is transcription of speech for automated protocol generation during investment consultations at banking institutions. Typical for the domain, the conversation contains numerical data, dates, proper names of persons and locations and highly specialized terms like product names. The transcriptions are characterized by commonly used phrases and sentence parts that are frequently repeated. Because of the small amount of available domain-specific speech and texts, it is not possible to perform reliable statistical language modeling. On the other hand, general language models, trained from large textual corpora, cannot provide satisfactory performance due to the mismatch of the speech with the dictionary and the language model. The possible usage of Context-Free-Grammars (CFG) imposes a problem of complexity and error-prone design. Moreover, because of the privacy policies applied in the banking services, the needed amount of relevant domain-specific texts cannot be provided in the original form. Instead, examples of sentences with fake personal data, characteristic for transcription of banking protocols, were provided. However, the amount of data was not enough to model appropriate CF grammar, neither to perform statistical language model training.

One of the common approaches to overcome such issues, is to define word categories containing synonyms and to train word-class based language model. Again, this only makes sense if enough relevant domain-specific data exists and the word classes are easily distinguishable. To avoid this bottleneck, textual data augmentation was performed by creating grammars rules and generating large number of "artificial" sentences that correspond to the provided examples. The word-classes were defined by a list of domain-specific words, numbers (cardinal and ordinal), proper names, addresses, locations, services, assets etc. Those categories were considered in the text generation and represented with the corresponding word-class tag. Afterward, the generated sentences, together with domain-specific texts were used to train a n-gram language model. This model can be used as standalone or, can be merged with the general purpose language model.

In the paper, a procedure for domain-specific dictionary creation and adaptation, language model training and merging, and supervised learning of pronunciation variants is described. The procedure development was based on the speech data collected during a free trial phase and texts collected from various available sources. For the language models merging and adaptation, the approach which delivers best possible performance during the trial phase is proposed. The dictionary and the Grapheme-to-Phoneme (G2P) model were used in acoustic training, hence avoiding the acoustic model and dictionary mismatch. The models were trained on speech corpora collected for Command & Control applications in a Car, in Smart Home and for transcriptions in Tourism and Date negotiation domains. The best performing acoustic model was chosen and the recognition performance was compared across different language models: the domain-specific model, the general model and their weighted combinations. The experimental results justified the proposed approach for domain-specific language modeling and adaptation on banking protocols transcriptions. The outlook gives a the time and effort estimation for the start-up phase in similar tasks.

The paper is organized as follows: In the Section 2 the speech corpora used for acoustic modeling, the textual data augmentation method, lexical and language modeling are described. Section 3 presents the speech recognition system used for evaluation and in the Section 4 the results and discussion are elaborated. The paper is concluded with Section 5.

## 2 Domain-specific Language Modeling

On Figure 1, the flow diagram of the procedure for adaptation and merging of language models is presented.

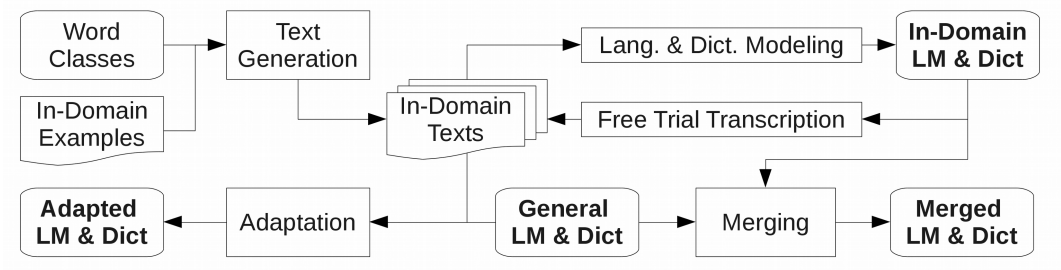


**Figure 1.** In-domain text generation, training, adaptation and merging of language models

Firstly, from the given domain-specific examples, the grammar rules with the word-class categories are defined. The generated texts were used for language and dictionary modeling to provide domain-specific recognition system. It was used in the free-trial phase to provide corrected transcriptions and to extend the content of the domain-specific textual corpus. The enriched text can be used again, to provide better language modeling or to adapt a general language model. The domain-specific model can be also merged with a general model. All three approaches were used in the experimental evaluation and the performance of the language models was observed.

### 2.1 Speech Corpora for Acoustic Modeling

The database used for acoustic modeling comprises of mixed speech corpora on German. The training set consists of 34558 utterances and the development set used for optimization of the training procedure, has 349 sentences. The total duration of the speech database is approx. 53 hours and 15 min. The words as found in the original transcriptions compose a vocabulary of 7845 unique words. The pronunciations were generated by automatized G2P procedure and included the training dictionary. The used G2P model was trained on a lexicon derived from the WebCelex database [1].

After each training session, the quality of the acoustic models (AM) was evaluated on a small test set of 96 utterances of 3 speakers, using a general purpose language model. Additional performance evaluations were conducted with a selection of trained acoustic models on a another different test set. This test set consists of 17043 utterances from 25 speakers. The utterances refer to different application domains, like: information retrieval, control of household devices, TV program selection/recording etc.

### 2.2 Textual Data Augmentation

The common problem in developing specific ASR solutions is the lack of appropriate domain-specific data, effectively rendering impossible reliable statistical language modeling. In order to accomplish the task for a given domain (transcriptions of banking protocols), it is necessary to collect information about the frequently used terms and their context in the sentences. This kind of information can be found in text material of the target domain, and it is used to build the transcription system's dictionary and the language model. The system's performance is highly dependent of the quantity and quality of the text material [2].

In applications like this, the texts are not available in the original form, only as a small collection of examples with fake personal and numerical data not suitable for language modeling. To overcome this problem, textual data augmentation should be employed, where the effective quantity of texts used to train language models is significantly increased. One way is, to create artificial data based on given examples, while another assumes that additional out-of domain texts are available and that can be used for language modeling.

In the first approach, the textual data can be produced by human experts in the way they believe it could be spoken as input to the domain-specific system. The problem is that, the developers could be easily biased by the provided examples and produce non-diverse sentences by repetition of similar phrase structures [3]. Automated procedures for sentence generation from context-free-rule-based templates also could be used for data augmentation [4]. The rules are expanded into alternative phrase instantiations to complete a sentence, and the templates could be designed to cover the patterns in statistical appropriate frequencies.

The second approach is useful only if word-class language modeling is employed. Namely, many n-grams are relatively domain independent, and the statistical relevance of the domain-specific data could be improved if they are included in the training. For example, the n-gram contexts where numerical data appears, like: currency, dates, days, months, years etc., could be included to the domain specific texts. Word-classes have to be precisely defined and the texts should be tagged with the appropriate class labels. Several well-known algorithms exist for this task, such as the contextual semantic tagger in [5].

To perform automated data augmentation, different context free grammar rules were designed to generate large amount of “artificial” sentences. Such grammars are powerful enough to describe most of the structure in spoken language and restrictive enough to have efficient parsers, yet they are inappropriate to be used for robust ASR since the grammar is almost always incomplete.

Rmutt generator was used to produce sentences with word-class tags by generating random strings from context-sensitive grammars [6]. Rmutt is a Turing-complete interpreted language and takes as input set of user-defined grammatical rules, each of which represents a set of choices that can be made at a particular level of grammatical description. Rmutt then makes these choices randomly, resulting in text which conforms to the grammar but is otherwise unpredictable. Rules can be weighted to control the frequency with which they are used in production.

## 2.3 Lexical modeling

### 2.3.1 Dictionary

The base dictionary was created using the WebCelex database retrieved from MPI [1]. The original dictionary was corrected in order to remove erroneous entries, garbage and pure foreign phonemes and words. They will confuse the G2P training, and very often the resulting pronunciation is erroneous. For example, all “ta-Z” like in “Etage”, “Plantage”, “Sabotage” are eliminated, because they will confuse the training for “*tage”. Corrections were done also on orthography, syllable boundaries and on pronunciations. This was done to improve the acoustic modeling and the recognition, but also to correct errors in the transcription of some individual words. All corrections were done automatically, even the corrections of individual words, in such way that it is easily repeatable and extendable.

### 2.3.2 Grapheme-to-Phoneme Transformation

G2P conversion was employed to create new and to extend the existing dictionary. Application specific dictionaries were derived in the training process of the language models. Many of the observed words were not part of the base dictionary and their pronunciations had to be created. The G2P conversion consists of: sequence alignment, to align the grapheme and phoneme sequence pairs in a training dictionary, training, to produce a model able to generate new pronunciations for unseen words, and decoding, to find the most likely pronunciation for the given model. The “Phonetisaurus”, a WFST-based open source toolkit for G2P conversion was used to train a model and to provide pronunciations [7].

The G2P training is fully data-driven, and it was performed over a training set with more than 320k words from the corrected WebCelex dictionary. The evaluation was done on a test set with 3136 words and 30062 tokens, not included in the training. They were randomly chosen from the base dictionary, and their pronunciations taken as reference. The token error rate (TER%) was calculated by the equation TER = ((S+I+D)/T) % and it was 0.11%, while the sequence (word) error was 0.96%.

| | | | |
|---|---|---|---|
| **(T)otal tokens in reference:** | 30062 | **(S)equences:** | 3136 |
| **(M)atches** | 30039 | **(C)orrect sequences:** | 3106 |
| **(S)ubstitutions** | 19 | **(E)rror sequences:** | 30 |
| **(I)nsertions** | 9 | **% Sequence ER (E/S)** | 0.96 |
| **(D)eletions** | 4 | **% Sequence Acc (1.0-E/S)** | 99.04 |
| **% Correct (M/T)** | 99.92 | | |
| **% Token ER ((S+I+D)/T)** | 0.11 | | |
| **% Accuracy 1.0-ER** | 99.89 | | |

**Table 1** - Detailed evaluation results of the G2P conversion

From the presented results it could be seen, that the applied modifications in dictionary introduced highly consistent pronunciation patterns, which is important also for the acoustic modeling. The results analysis showed that substitutions are mostly between similar phonemes like: /a/ confused with /a:/, /e:/ confused with /E/, /u:/ confused with /U/, the combination /ts/ confused with the pair /t/ and /s/ etc.

## 2.4 Language modeling

### 2.4.1 Domain-specific Language Modeling

The Rmutt tool was used to create 10 Million sentences employing grammar of 100 lines (processing time of 2 min). The corresponding dictionaries were extracted from the text and

the word-class files needed for LM definition were generated. Such textual corpus is already tagged with the class labels and it was used to create various domain-specific language models. They differ regarding the number and content of the words classes:

- "BankingV10", 118 word classes describing ordinal and credit card numbers, locations and regions, proper names, assets, banking services and products, as well as common used phrases, like: *"Herr Weiß möchte zu Beginn übernächster Woche Investitionen für € 9340 monatlich tätigen."*, *"Und Herrn Franke 's jährliche Ausgaben betragen € 44 Tausend."*, *"Er möchte vor dem 26. 10. 2023 kein weiteres Kapital investieren."*
- "BankingV14", same classes as in BankingV10, including punctuation as special words, in order to provide syntactically correct transcriptions.

The language modeling was performed using SRILM toolkit and the procedure was split in two stages to reduce the computational and memory requirements [10]. Firstly, n-gram counts were estimated on the text corpora and later the same counts were used to create various LMs. For the domain-specific LMs, the threshold of $10^{-5}$ is chosen to prune the n-gram probabilities, if their removal causes (in the training set) perplexity of the model to increase by less than the threshold relative. On large corpora with more diversity, a much smaller value should be chosen, otherwise there will be very few tri-grams (and even bi-grams) included.

### 2.4.2 General Language Modeling

The freely available databases from the web corpus of the WaCky-Initative [9], were used:

- "deWaC", a 1.7 billion word corpus constructed from the Web limiting the crawl to the .de domain and using medium-frequency words from the Süddeutsche Zeitung corpus and basic German vocabulary lists as seeds, and
- "sdewac", a 0.88 billion word corpus derived from deWaC, duplicate sentences and some noise have been removed. The corpus is in Unicode format.

A general purpose language model (noted in the text as "freespeech") was created using the "deWaC" corpora, with a vocabulary restriction of 50000 most frequent words as found in the base dictionary HaGenLex [10]. For the G2P conversion the same model was used as described before. Another general model (named as "sdewac") was created on the "sdewac" corpus, following the same procedure, the used vocabulary has 86000 unique words, of which 63000 most frequent words from the WebCelex, and the others from application specific word-lists (e.g. banking, home automation, information retrieval, locations etc). Good-Turing discounting and pruning threshold of $10^{-8}$ were used for the both language models.

### 2.4.3 Model Adaptation and Interpolation

Adaptation of a general language models ("freespeech" and "sdewac") on unseen content was conducted by a creating of small model from existing domain-specific text and merging the models together. Another approach is to merge the class based model with a general language model which could cover the phrases unseen in the domain specific texts, preserving the advantages and the flexibility of the class based approach. In the experiments, the before mentioned SRILM toolkit was used for interpolation of different n-gram models, with a weighting factor controlling the influence of the main model (values between 0 and 1).

## 3 Speech Recognition System

The recognition framework used for acoustic modeling and recognition is Sphinx/pocketsphinx [11]. Acoustic model training and performance evaluation was conducted using Sphinx training tools. Customized procedure for model training and testing

was established. The critical parameters, as the word recognition performance (WER), real time factor (xRT) and acoustic model size were analyzed. The standard procedure for acoustic modeling was used with additional modifications:

- Forced-Alignment (FA) was used to properly align the transcriptions to the utterances prior to training, resulting in more consistent data and better acoustic modeling.
- Linear Discriminative Analysis (LDA) combined with Maximum Likelihood Linear Transformation (MLLT), as feature-space transformations were used to provide improvements in WER (observed, up to 25% relative), the decoding speed and memory footprint of the acoustic model.

A number of acoustic models were trained with different training configurations and their quality was estimated on the test set described in Section 2.1.

| AM Designation | Type | Senones | Gaussians | WER | xRT |
|---|---|---|---|---|---|
| cont_1000_4_FA_CLX | continuous | 1000 | 4 | 17.33 | 0.03 |
| cont_1000_8_FA_CLX | continuous | 1000 | 8 | 16.57 | 0.06 |
| cont_1000_16_FA_CLX | continuous | 1000 | 16 | 15.62 | 0.09 |
| **cont_1000_32_FA_CLX** | **continuous** | **1000** | **32** | **15.43** | **0.14** |
| cont_1000_64_FA_CLX | continuous | 1000 | 64 | 16.95 | 0.24 |
| semi_1000_512_FA_CLX | semi-continuous | 1000 | 512 | 16.57 | 0.02 |
| **semi_4000_512_FA_CLX** | **semi-continuous** | **4000** | **512** | **15.24** | **0.02** |

**Table 2** - Acoustic model (AM) evaluation

The best performing continuous AM was chosen, since it provides better speaker adaptation and more robust performance than semi-continuous models. The used language model is “freespeech” with its corresponding dictionary.

## 4 Results and Discussion

Numerous recognition experiments were conducted in order to observe the performance of the domain-specific LMs, general purpose LMs and their merged versions. Three test sets were employed in the experiments. One is the test set used for AM quality evaluation noted as TEST3. The other two are sets with domain-specific content of same speech utterances, but with slightly different transcriptions. The difference is that, in the first - TEST1, numerical values are represented as digits, while in the second TEST2, they are represented as words. This was done to investigate the influence of the general LMs where the digits are seldom represented in the model and the dictionary. The domain-specific speech (TEST1 and TEST2) was recorded by 20 different speakers (13 male and 7 female) among them, 2 non-native German speakers. The performed experiments were:

- Merging general model “sdewac” with the domain-specific models Banking V10 and Banking V14 (Table 4).
- Adaptation of the general purpose language models (“freespeech” and “sdewac”) with domain-specific text different than the transcriptions in TEST1 and TEST2 (Table 5);

The percent word-error-rate (WER) and perplexity (PPL) analysis, per group of experiments and across the test sets are presented in the following tables. Correctness (Corr) was calculated as the percent of correctly recognized words (excluding deletions and substitutions), while the WER takes into account also the insertions as an error. For the language model evaluation, the transcriptions of all 3 test sets were processed accordingly to the used model type. That means, in the case of a class based model, the texts were parsed and tagged with the corresponding class labels. OOVs is a percent out-of-vocabulary words, tokens that appear in the test transcriptions but not in the language model (dictionary).

Logarithm of probability (logprob) is calculated by excluding the unknown tokens, and the perplexity is defined as:

$$PPL = 10 \wedge (-logprob / (words - OOVs + sentences))$$

In all cases, the best performing acoustic model was used (see Table 2.), 32 continuous Gaussian densities and with 1000 senones. From Table 3, it is clear that the general purpose models are not suitable for transcription of banking protocols (TEST1 and TEST2), both "freespeech" and "sdewac" impose high WER, mostly because of the LM mismatch and partially because of large number of OOV.

| | TEST1 | | | | TEST2 | | | | TEST3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | %Corr | %WER | PPL | %OOV | %Corr | %WER | PPL | %OOV | %Corr | %WER | PPL | %OOV |
| Banking V10 | 72.7 | 32.8 | 326.6 | 19.0 | 47.4 | 55.3 | 451.7 | 26.8 | 7.6 | 97.5 | 4178.8 | 50.3 |
| Banking V14 | 72.8 | 40.7 | 176.1 | 22.3 | 46.1 | 61.5 | 350.8 | 29.3 | 6.1 | 100.9 | 10067 | 45.1 |
| sdewac | 58.9 | 49.9 | 2169.8 | 0.0 | 49.6 | 54.5 | 1051.5 | 0.4 | 86.5 | 14.7 | 620.2 | 0.0 |
| freespeech | 50.9 | 82.2 | 359.5 | 22.8 | 63.4 | 48.3 | 244.7 | 10.4 | 85.7 | 15.4 | 156.1 | 0.6 |

**Table 3** - Baseline language models performance (domain-specific and general)

The domain-specific models performed better, still the WER is quite high. The reason is high OOV due to missing words in some classes as well as the nature of the models them self – they were produced on artificial data. On the other side, the BankingV10 and V14 are completely useless for other purposes than the transcription of banking protocols.

| sdewac/V10 | TEST1 | | | | TEST2 | | | | TEST3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| weights | %Corr | %WER | PPL | %OOV | %Corr | %WER | PPL | %OOV | %Corr | %WER | PPL | %OOV |
| 0.25 | 77.1 | 27.8 | 618.3 | 0.0 | 52.6 | 49.92 | 531.9 | 0.42 | 64.6 | 37.3 | 1021.5 | 0.0 |
| 0.50 | 76.6 | 27.8 | 531.8 | 0.0 | 53.1 | 49.22 | 417.7 | 0.42 | 75.1 | 26.9 | 589.4 | 0.0 |
| 0.75 | 75.8 | 28.3 | 614.2 | 0.0 | 53.0 | 49.14 | 437.6 | 0.42 | 81.0 | 21.5 | 465.6 | 0.0 |

**Table 4** - Merged (Banking V10 with sdewac using different weights)

After merging general model „sdewac" with "BankingV10", in the best case (using 0.5 interpolation weight), the relative WER improvements evaluated on TEST1, were: 44.3% against „sdewac" and 15.0% against "BankingV10" (Table 4). Smaller improvements are observable on TEST2, while the WER increased on TEST3 which was expected. In order to adapt the "sdewac" LM on a domain-specific content, text which is not included in the training and testing was used to create small language model which was merged with "sdewac" with different weights.

| sdewac/"in-dom" | TEST1 | | | | TEST2 | | | | TEST3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| weight | %Corr | %WER | PPL | %OOV | %Corr | %WER | PPL | %OOV | %Corr | %WER | PPL | %OOV |
| 0.1 | 68.3 | 37.4 | 471.8 | 0.0 | 57.0 | 45.5 | 401.0 | 0.4 | 82.1 | 19.8 | 202.7 | 0.0 |
| 0.5 | 70.1 | 35.6 | 242.8 | 0.0 | 56.9 | 45.7 | 286.0 | 0.4 | 77.5 | 24.8 | 260.0 | 0.0 |
| 0.9 | 66.6 | 40.4 | 271.3 | 0.0 | 54.3 | 48.9 | 471.2 | 0.4 | 59.0 | 43.1 | 861.7 | 0.0 |

**Table 5** - Adaptation with, unseen domain-specific text, different weights

In Table 5 it could be seen that relative WER improvements against "sdewac" LM (weight 0.5), are: on TEST1 - 28.5%, on TEST2 - 16.1%, and, as expected, relative increase of WER for TEST3 - 68.7%. Degradation of the WER performance on TEST3 is irrelevant since the main objective was to adapt the general purpose model for the domain-specific sentences.

## 5 Conclusions

In this paper, a procedure for dictionary and language model creation and adaptation for a domain-specific application is described. The target domain is speech transcription for automated protocol generation during investment consultations. The bootstrapping for domain specific adaptation is based on speech and data collected during a free trial phase and the available similar texts. Due to small amount of domain-specific texts, data augmentation was done by generating large amount of artificial sentences. Those texts were used to train class-based statistical language models (as standalone or merged). The recognition performance was evaluated across different language models: the domain-specific model, the general purpose model and their weighted combinations. From the results it can be concluded that, data augmentation can be used successfully to create domain-specific word-class based language models. Merging class-based with general models and adaptation with a small amount of relevant texts improves the WER and PPL in both cases. The results justified the proposed approach for domain-specific language modeling on banking protocols transcriptions.

**Acknowledgments:** The authors are grateful to Ralf Kompe for his contribution on the development of this study. This research was supported by Zentrales Innovationsprogramm Mittelstand – ZIM, KF2403504WD2.

## References

[1] Max Planck Institute for Psycholinguistics. (2001). WebCelex [database]. Retrieved from http://celex.mpi.nl/

[2] Neves, Luís, et al., "Domain adaptation of a Broadcast News transcription system for the Portuguese Parliament", *Computational Processing of the Portuguese Language*. Springer Berlin Heidelberg, 2008. 163-171.

[3] Liu, Sean, Stephanie Seneff, and James Glass, "A collective data generation method for speech language models." *Spoken Language Technology Workshop (SLT), 2010 IEEE*.

[4] Chung, Grace, Stephanie Seneff, and Chao Wang. "Automatic induction of language model data for a spoken dialogue system.", *6th SIGdial Workshop on Discourse and Dialogue*. 2005.

[5] Cucchiarelli, Alessandro, Danilo Luzi, and Paola Velardi. "Semantic tagging of unknown proper nouns." *Natural Language Engineering* 5.02 (1999): 171-185.

[6] http://sourceforge.net/projects/rmutt/

[7] Novak, J. R., Minematsu, N., & Hirose, K. (2012, July). "WFST-based Grapheme-to-Phoneme Conversion: Open Source Tools for Alignment, Model-Building and Decoding.", In *10th International Workshop on Finite State Methods and Natural Language Processing* (p. 45)

[8] Stolcke, Andreas. "SRILM-an extensible language modeling toolkit." *Interspeech*, 2002.

[9] Baroni, Marco, et al. "The WaCky wide web: a collection of very large linguistically processed web-crawled corpora." *Lang. resources and evaluation*, 43.3 (2009): 209-226.

[10] Sven Hartrumpf, Hermann Helbig, and Rainer Osswald: "The Semantically Based Computer Lexicon HaGenLex - Structure and Technological Environment". In: *Traitement automatique des langues*, 44(2), 2003, pp. 81-105.

[11] Huggins-Daines, David, et al. "Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices." *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. IEEE International Conference on*. Vol. 1. IEEE, 2006.