Transcription-less Call Routing using Unsupervised Language Model Adaptation

Nicolae Duta

Natural Language Understanding, Nuance Communications, Burlington, MA, USA
nicolae.duta@nuance.com

Abstract

A key challenge when building call routing applications is the need for an extensive set of in-domain data that is manually transcribed and labeled, a process which is both expensive and time consuming. In this paper we analyze a Language Model training approach based on unsupervised self-adaptation which does not require any manual transcriptions of the in-domain audio data. We investigate the usefulness of several sources of language data for building bootstrapped LMs as well as an utterance duration dependent adaptation scheme which balances the required computational resources. Results on deployed call routing applications show that the routing accuracy obtained using the self-adapted LM is within 1-5% absolute of the accuracy of the system trained on manual transcriptions irrespective of the original bootstrapped LMs.

Index Terms: language model adaptation, call routing

1. Introduction

Spoken language understanding systems have been deployed in numerous applications which require some sort of interaction between humans and machines [4],[5]. Most of the time, the interaction is controlled by the machine which asks users questions and then attempts to identify the intended meaning of their answers (expressed in natural language) and take actions such that to satisfy their requests. An important class of applications which currently employs Natural Language Understanding (NLU) technology is "call routing" whose goal is to automatically route a telephone query from a customer to the appropriate set of agents based on a brief spoken description of the problem [3]. Call routing systems reduce queue time and call duration, thus saving money and improving customer satisfaction by promptly connecting the customer to the right service representative in call centers.

To find the meaning of a human utterance in a call routing system, the caller's speech is first translated into a text string by an Automated Speech Recognition (ASR) system and the text is then fed into a NLU component called Router. The NLU task is modeled as a statistical classification problem: the text corresponding to an utterance is assigned to one or more of a set of predefined user intents (routes). Several classifiers (boosting [6], Maximum Entropy (ME) [11], Support Vector Machines (SVM) [8]) have been compared in the literature and shown to produce similar performance (1-2% differences in classification accuracy) [8]. The Router we employ in this study uses binary unigram features and a standard back-propagation neural network as a classifier.

Training a call routing application requires two sets of manual annotations of the customer spoken utterances. The first set of annotations, called transcriptions, denote the text corresponding to the spoken waveforms and is used to train the Language Model (LM) used in the ASR system as well as the Router. The second set of annotations (labels), concerns the customer's intent and is used to train the Router. We assume that during the pre-deployment stage the system can record the human operator actions and each action directly corresponds to one of the available system routes [11]. Therefore, very little or no manually labeled data is needed. However, language data is still needed to train the recognition LM.

As noted in [4], several approaches to enhance LM portability have been proposed:
(i) obtaining additional language training material,
(ii) interpolating domain-specific LMs with other LMs
(iii) improving probability estimation with limited in-domain data and
(iv) using unsupervised LM self-adaptation to in-domain audio

In this paper, we follow up on some of the ideas introduced in [11] on LM self-adaptation using cross-validation schemes and construct Language Models for several currently deployed call routing applications without using any manual transcriptions of the in-domain data. We report detailed results on some issues not yet fully investigated in the literature:

- The effectiveness of various language data sources that may be available for training the bootstrapped LM. Graphs of the Word Recognition Accuracy (WRA) and router accuracy (RA) corresponding to several adapted LMs are shown and discussed
- An utterance duration dependent procedure for automated transcription which balances the required computational resources and makes it easy to detect and analyze (in the research stage) the sources of recognition errors

Table 1. *Data used for language model bootstrapping.*

Description	Number (thousands) of utterances / words	Number of routes
Application 1 (telecommunications)	80 / 530 (training set) 10 / 80 (test set)	129
Application 2 (telecommunications)	100 /730 (training set) 10 / 72 (test set)	865
Application 3 (government)	36 / 257 (training set) 5.1 / 37 (test set)	140
Application 4 (utilities)	18 / 72 (training set) 2 / 8 (test set)	37
Application 5 (only LM data) telecommunications	703 / 4300	---
Fisher (US conversational)	1800 / 20500	---
U. of W. web data	5000 / 61000	---

2. Language Model Bootstrapping Data

Previous studies have discussed methods for reusing language data [4][6][11]. Several data sources have been found to be useful for replacing the current application's (also called in-domain) manually transcribed training set:

I. A small set (1-2 utterance samples for each route) of in-domain manual transcriptions

II. Automatic transcriptions of in-domain audio data

III. Manual transcriptions used for training other call routing applications (either in the same-sector / vertical or in a different sector)

IV. Transcriptions of spoken conversational data (e.g. the Fisher corpus [7])

V. Conversational–like web-crawled text data [1].

Depending on the application and language spoken, various subsets of the five data sources mentioned above may be available. We have only focused on English applications where large amounts of high quality, spoken conversational and conversational-like text data have been collected. We also used data from four call routing applications currently deployed by Nuance Communications: two large (over 100 routes) applications in the telecommunication sector, one large application in the government sector and one medium scale application in the utilities sector. Each application has its own training and test sets fully transcribed and route sets which differ (see Table 1) both quantitatively and qualitatively (different number of slots corresponding to pieces of information to be extracted from the spoken utterance). Each application's audio training set was automatically transcribed as described in Section 3 and the automatic transcriptions were used to train application specific LMs and Routers which were subsequently tested on the application's test set. The web-crawled text data, besides its size, has a noticeable advantage: it comes from multiple English speaking countries. This was helpful in enhancing the lexicon for Application 2 which is deployed in Canada and uses some words not encountered in the US-collected Fisher corpus.

In order to assess the effectiveness of various data sources, we investigate the following bootstrapped LMs (see also Table 2):

1. Human conversational (Fisher data): LM_1.
2. Spoken conversational enhanced with conversational-like text data: LM_2.
3. Same sector application: LM_3 trained on Application 5 was tested on Applications 1 and 2. No same sector-data was available for Applications 3 and 4.
4. Different sector application: LM_4 trained on Application 3 was tested on Applications 1, 2 and 4 while LM_3 was tested on Application 3.
5. Leave-one-out: LM_5 trained on all available data excluding current application.

Table 2. *Bootstrapped LMs used in the first recognition pass of the LM adaptation process.*

LM	Description	Lexicon (K words)	2-grams (K)	3-grams (K)
LM_1	Human conversational	36	1300	500
LM_2	Spoken conversational + conversational-like text data	82	2800	--
LM_3	Same sector application	4	83	308
LM_4	Different sector application	4	33	75
LM_5	Leave-one-out	83	2800	--

All language models were trained using the weighted count interpolation method and Good-Turing discounting [2]. Due to our LM representation as Finite State Machines (FSM) we had to limit the LM size to less than 3 million n-grams, therefore LM_2 and LM_5 are heavily pruned bigram LMs.

The N-gram coverage of the bootstrapped LMs was measured on each application's training set and is shown in the second column of Table 3. The first number represents the Out-of-Vocabulary (OOV) rate, the last number is the percentage of {2-3}-grams covered by each LM while the center number is the percentage of words covered at 1-gram level. The Fisher data has a small OOV rate of 0.2-0.8%. That rate is further reduced to 0-0.3% when the web text data is added. Call routing data from the same-sector application has a 1-1.5% OOV rate while call routing data from a different-sector application has a larger 2-6% rate. The {2-3}-gram coverage is high: 85-95% for all LMs except for the different-sector LM where coverage is only 65-75%. Therefore we expect few recognition errors to be due to insufficient LM coverage.

3. Automated transcription of the in-domain audio

In-domain (application specific) audio is usually collected during the pre-deployment stage of an application or during post-deployment tuning procedures. Several studies proposed methods to automatically transcribe the in-domain audio using unsupervised self-adaptation [9][11]. One typically starts by building a bootstrapped LM followed by several iterations of in-domain audio recognition and LM adaptation. We used the five bootstrapped LMs described in Section 2.

Table 3. *Language model coverage, word accuracy and router accuracy corresponding to the bootstrapped and adapted LMs during the LM self-adaptation process.*

Appl. 1	Bootstrapped LM coverage	Adapted LM coverage	Pass I WRA%	Pass II WRA%	Test WRA%	Test RA%
Baseline		0.4 3.6 96	73.0	73.0	75.0	77.6
LM_1	0.5 8.5 91	1.1 7.2 91.7	55.1	66.5	68.3	74.4
LM_2	0.3 5.2 94.4	0.8 6.9 82.3	52.4	65.5	63.8	76.8
LM_3	1.2 11.6 87.2	1.0 6.2 92.8	73.1	72.4	69.5	77.4
LM_4	3.3 19.8 76.9	0.8 5.9 93.3	63.5	70.5	68.3	76.8
LM_5	0.3 2.5 97.2	0.7 5.8 93.5	61.9	70.7	67.4	75.6
Appl. 2						
Baseline		0.5 5.5 94	76.3	76.3	71.9	77.7
LM_1	0.8 12 87.2	1.7 8.3 90	42.7	65.2	62.1	75.2
LM_2	0.3 7.5 92.2	1.3 9.6 89.2	46.7	55.8	66.1	74.4
LM_3	1.3 10.4 88.3	1.5 6.1 92.3	65.1	66.5	70.2	75.3
LM_4	5.8 24.6 69.7	7.2 11.3 81.5	69.7	68.0	57.9	73.2
LM_5	0.2 3.2 96.6	1.0 6.9 92.1	61.5	61.0	64.3	75.2
Appl. 3						
Baseline		0.7 6.6 92.7	74.1	74.1	71.2	79.2
LM_1	0.5 10 89.5	1.5 8.6 89.9	42.3	63.0	58.2	76.4
LM_2	0.3 6.1 93.6	1.4 9.8 88.8	47.7	57.6	53.5	75.9
LM_3	4.3 22.7 73	5.5 11.2 83.3	58.1	58.6	52.6	73.7
LM_5	0.3 5.0 94.7	1.2 8.7 90.1	55.7	61.5	64.2	75.4
Appl. 4						
Baseline		0.9 8.0 91.1	81.3	81.3	75.7	88.9
LM_1	0.2 11 88.8	1.8 10.3 87.9	45.5	72.1	65.9	87.3
LM_2	0 7.2 92.8	1.9 11.5 86.6	52.2	62.0	59.7	85.6
LM_4	2.1 17 70.9	0.8 6.5 92.7	64.3	73.4	69.0	86.1
LM_5	0 11.6 88.4	2.0 9.6 88.3	64.9	72.4	68.4	86.9

The detailed unsupervised transcription procedure is shown in Figure 1 and consists of the following steps:

3.1. Sorting the in-domain utterances by duration

The in-domain utterances are sorted by duration and divided into multiple batches of roughly equal total duration. That places all utterances with identical (or close) transcriptions in the same batch or in adjacent batches. These batches can be decoded in parallel and require similar processing times.

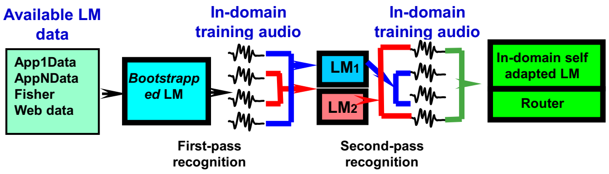

Figure 1: *The unsupervised self-adaptation procedure for automated transcription of the in-domain audio.*

3.2. First-pass recognition using a bootstrapped LM

The in-domain utterances are first recognized using a LM bootstrapped from the available language data. Figure 2 plots the Word Recognition Accuracy (WRA = 1 – Word Error Rate) for each batch and different LMs[1] (including the baseline LM trained on manual transcriptions).

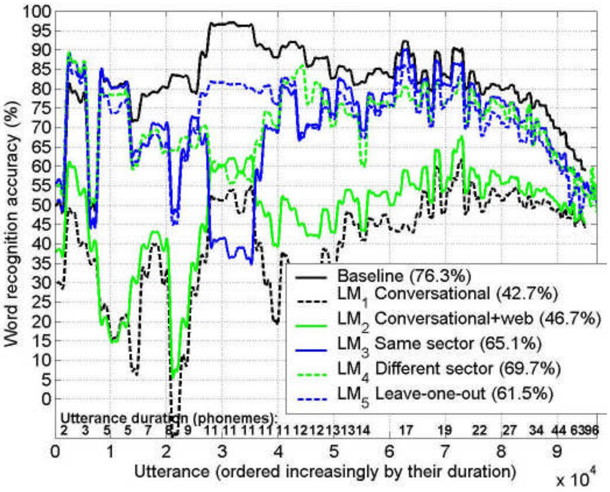

Figure 2: *First-pass word recognition accuracy as a function of utterance duration for multiple bootstrapped LMs.*

Plotting the recognition accuracy on small batches as a function of utterance duration[2] has several advantages:

(i) It makes easier to detect and analyze the causes of recognition errors especially the effect of the OOV words (for example, the largest drop in accuracy with respect to the baseline LM in Figure 2 occurs on utterances about 8-phoneme long which correspond to the (duplicate) instances of the OOV word "simpatico")

(ii) It makes easier to design processing methods which depend on the utterance length (for example it is easy to notice in Figure 4 that the router accuracy is very low on long utterances so it makes sense to reject them right away)

(iii) One can readily inspect the duration distribution and amount of duplication among the training utterances (call routing applications typically have a large number of utterances which correspond to the same text string).

The WRAs are shown in the fourth column of Table 3 (as well as in Figure 2) and compared to the two-way cross recognition baseline (each half-training set is recognized using a LM trained on the manual transcription of the other half; as in Section 3.3). The WRAs corresponding to the bootstrapped LMs are 10-35% (absolute) lower than the baselines and the smallest accuracy drop is obtained when the LM is trained on call routing transcriptions from different applications. Figure 2 also shows that the call routing LMs make fewer (and different) errors than the conversational LMs on short (1-4 words) utterances since short utterances are very common and better modeled in call routing applications.

3.3. Cross-adaptation of the language model

The utterance batches are next divided into two subsets A_1 and A_2. Subset A_1 contains the shortest and the longest 20% of the utterances, while subset A_2 contains the remaining mid-length utterances. In this way, one can minimize the transcription overlap between A_1 and A_2 while each subset can still retain a good representation of the full utterance set. For each subset A_i build a language model LM_i out of the text recognized on A_i at step 3.2 (see Figure 1).

3.4. Second-pass recognition using self-adapted LMs

A second-pass recognition of the in-domain audio is performed using the 2-way cross adapted LMs computed at step 3.3. That is, subset A_1 is recognized using LM_2 and A_2 is recognized using LM_1 in order to avoid feeding the recognition errors at the previous step back into the adaptation process [11].

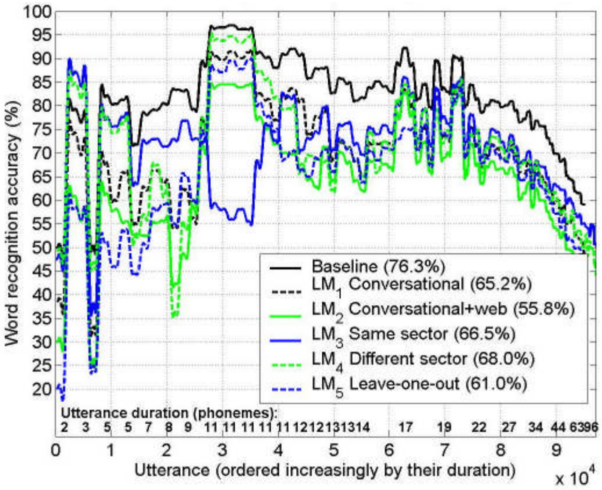

Figure 3: *Second-pass word recognition accuracy as a function of utterance duration for multiple adapted LMs*

Figure 3 and Table 3 show the WRAs for the five adapted LMs. The accuracy differences corresponding to the adapted LMs are much reduced with respect to those corresponding to the bootstrapped LMs and most WRAs are less than 15% (absolute) lower than the baseline. The adapted LMs corresponding to bootstrapped LMs based on call routing data are still 5-10% more accurate than the conversational-based LMs.

[1] We found it less informative to show the accuracy graphs averaged over the four applications since the shape of each graph is dependent on application specific patterns (size and distribution of the training set, OOVs, etc). Therefore we only show the accuracy graphs for Application 2 (largest application) and note that the graphs corresponding to the remaining three applications are very similar in nature.

[2] For a less cluttered display, the actual accuracy numbers were smoothed using polynomial spline-functions

3.5. Final LM adaptation

A final adapted LM as well as a Router is computed out of the text recognized at step 3.4. The N-gram coverage of the adapted LMs was measured on each application's test set and is shown in the third column of Table 3. The OOV rate is higher by at least a factor of two with respect to the baseline LM and by a factor of three with respect to the bootstrapped LMs which suggests that no instance of some words was correctly recognized during the adaptation process although those words were covered by the bootstrapped LMs at least at the 1-gram level (for example, the word "attendant" was recognized as "I tend and"). Therefore, the final lexicon may have to be combined with the lexicons used in related call routing applications in order to insure a better word coverage. The lexicon enhancement appears to benefit most applications with smaller training sets (Applications 3 and 4 showed a 0.5% gain in RA while there was no gain for Applications 1 and 2).

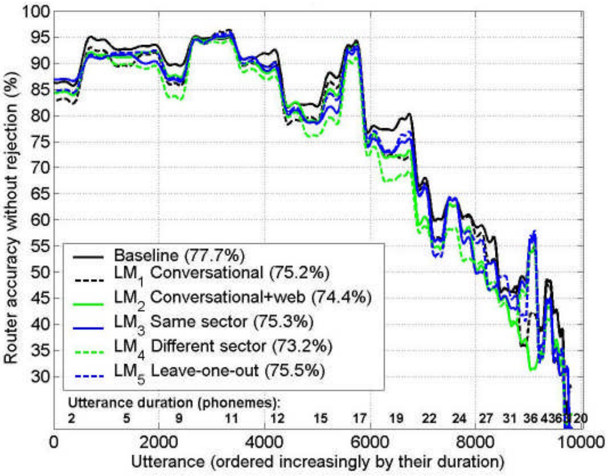

Figure 4: *Router accuracy as a function of utterance duration for multiple adapted LMs.*

The WRA on each application's test set are shown in the sixth column of Table 3 and are generally 5-15% lower than the baseline. Figure 4 (and last column in Table 3) shows the router accuracy (without rejection) measured on the applications' test sets. The RA differences among various adapted LMs are quite small and there is only 2-5% absolute loss compared to the baseline where both the LM and the Router are trained on manual transcriptions. This fact indicates that WRA is not well correlated and therefore not a good predictor for the RA (see also [10]). The best router performance is attained when starting with a bootstrapped LM computed using data from a same-sector application if such data is available. When that is not available, the conversational LM can be used as a starting point. Notice that the much lower OOV rate of the text-enhanced conversational LM does not translate into better router accuracy, mostly due to the increase in lexicon size.

Performing more adaptation iterations only slightly increases the router accuracy (0.3-0.5%) therefore, depending on the computational resources which are available, one may decide to stop after a single iteration. However, performing a single recognition pass through the audio generates a relatively large loss in RA although the loss in WRA may be quite small. For example, on App. 1, a single recognition pass using LM_3 decreases the WRA from 69.5% to 68% but the RA degrades a lot more from 77.4% to 71.9%.

4. Conclusions

In this paper we demonstrated that it is possible to train Language Models for call routing applications with no manual transcriptions of the in-domain data. We investigated the usefulness of several sources of language data for building bootstrapped LMs and showed that the routing accuracy obtained using self-adapted LMs is within 1-5% absolute of the accuracy of the system trained on manual transcriptions irrespective of the original bootstrapped LMs. However, if in-domain audio is not available, the routing accuracy loss is higher (6-12% absolute, see also [6]) and bootstrapped LMs trained on manual transcriptions from the same industry applications are more effective than bootstrapped LMs trained on general conversational data.

5. Acknowledgements

The author would like to thank Real Tremblay, Mark Fanty and Matthieu Hebert for valuable discussions and comments during this project.

6. References

[1] I. Bulyko, M. Ostendorf, M. Siu, T. Ng, A. Stolcke and O. Cetin, "Web resources for language modeling in conversational speech recognition," *ACM Trans. on Speech and Language Processing*, vol. 5(1), 2007, Data available at: https://ssli.ee.washington.edu/projects/ears/ WebData/web_data_collection.html.

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

[3] S. Cox, "Discriminative techniques in call routing", in *Proc. ICASSP Conf.*, Hong Kong, Vol. I, pp. 620-623, 2003.

[4] Y. Gao, L. Gu and H-K. J. Kuo, "Portability challenges in developing interactive dialogue systems", in *Proc. ICASSP Conf.*, Philadelphia, PA, Vol. V, pp. 1017-1020, 2005.

[5] N. Gupta, G. Tur, D. Hakkani-Tür, S. Bangalore, G. Riccardi and M. Rahim, "The AT&T Spoken Language Understanding System", *IEEE Transactions on Speech and Audio Processing*, vol. 14(1), 2005, pp. 213-221.

[6] D. Hakkani-Tür, G. Tur, M. Rahim and G. Riccardi, "Unsupervised and Active Learning in Automatic Speech Recognition for Call Classification", in *Proc. ICASSP Conf.*, Montreal, Canada, Vol. I, pp. 429-432, 2004.

[7] Linguistic Data Consortium (LDC), "Fisher English Training Speech", Philadelphia, 2004 Available: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004T19.

[8] R. Sarikaya, H-K. J. Kuo, V. Goel and Y. Gao, "Exploiting Unlabeled Data Using Multiple Classifiers for Improved Natural Language Call-Routing", in *Proc. Interspeech Conf.*, Lisbon, 2005, pp. 433-436.

[9] G. Tur, and A. Stolcke, "Unsupervised Language Model Adaptation for Meeting Recognition", in *Proc. ICASSP Conf.*, Honolulu HI, Vol. IV, pp. 173-176, 2007.

[10] Y.Y.Wang, A. Acero, and C. Chelba, "Is Word Error Rate a Good Indicator for Spoken Language Understanding Accuracy?", in *Proc. ASRU Conf.,* St. Thomas, USA, pp. 577-582, 2003.

[11] Y. Wang, J. Lee and A. Acero, "Speech Utterance Classification Model Training Without Manual Transcriptions", in *Proc. ICASSP Conf.*, Toulouse, France, Vol. I, pp. 553-556, 2006.