# Speech Scenario Adaptation and Discourse Topic Recognition on Mobile Smart Terminal

Min Huang,Xuran Li,Silong Wu
School of Software Engineering
South China University of Technology
Guangzhou, China
minh@scut.edu.cn

Yinong Chen
School of Computing, Informatics, and Decision Systems Engineering
Arizona State University
Tempe, AZ 85287-8809

***Abstract*—Context identifying based on speech data is important to social services and city management. In a complex application environment, a speech recognition system needs to address two main problems: background noises and large vocabulary search latency. We use the adjustment acoustic model to deal with the scenario adaptation, and we use adjustment dictionary and language module to solve the discourse topic recognition. As a case study, we design and implement a continuous language speech recognition system on a mobile smart terminal. Experiments show that the scene adaptation effectively improves the accuracy rate of speech recognition, and the discourse topic recognition verifies the recognition effectiveness of our speech recognition system.**

***Keywords-speech recognition; scemario Adaptation; discourse tipic recogntion; acoustic model; mobile terminal***

## I. INTRODUCTION

The latest technological advancement in Internet of Things, Robot as a Service, cloud computing, and big data analysis are reshaping the world in the ways that connecting resources and virtual services to physical services [1][2]. Smart City has become a comprehensive example, which combines all the latest technologies in its construction. Smart city is based on the framework of digital city and its implementation in a real city by the ubiquitous sensor network [3]. In a smart city, a large number of residents have smart devices such as mobile phones, tablets, and computers. It is one of the goals of a smart city to use sensors in mobile smart devices to collect information about the users and their surroundings, and to provide convenient and intelligent services to residents, such as floating cellular data [4], traffic guide and social recommendation [2][5]. Researchers use Geolife GPS track and 20 thousand taxies' GPS track to find out interesting scenic spots and travel sequences [6]. Another example is using GPS track find the user's traffic pattern, such as metro, bus or car [7]. Speech data are one of those important data types that can provide necessary data to intelligent services. Speech processing technologies and speech recognition technologies have become mature. There are large amounts of speech data that can be collected consciously or unconsciously [8]. Then, they can be processed by big data analysis of cloud platform to obtain many meaningful results, such as context awareness or social recommendation [5][9]. Therefore, speech recognition technologies can combine physical machines with related intelligent services. Such combination plays a vitally important role in building a smart city.

In 1950s, AT&T Bell Labs invented the first speech recognition system, which can recognize 10 numbers in English [10]. In 1970s, Hidden Markov Model (HMM) is first applied in speech recognition and Vector Quantization (VQ) was developed. Today, speech recognition technology has been improved in many aspects. Subspace Gaussian mixture model (SGMM) is the extension of HMM, which can obtain more training and improvement, particularly in Chinese language recognition accuracy [11]. In language models, Recurrent Neural Network (RNN) [12][13] and other methods are exploring new modeling approach. With the development of deep learning (DL), deep neural network (DNN) is successfully applied in speech recognition and face recognition [14][15]. Compared with traditional GMM-HMM, DNN-HMM system brings the error rate down dramatically, as reported in an experiment in 309 hours in Switchboard mission and 2000 hours in Fisher mission, which is near one third [16]. The error rate is near one third of the GMM-HMM.

The speech recognition system on mobile terminals has to address many additional problems. First, the robustness of the system is usually not good enough [17]. The speech recognition system seriously depends on the environment, and it has little adaptability. Second, speech recognition model based on HMM is not usable when the speech signal has background noise. Not only the background noise, but also the oral speech voices will also influence the performance of the system. As an extension, speech recognition technology can be widely used in voice control, information search and other modern information services. Finally, compared with English, Chinese is much more difficult to apply in speech recognition. The main reason is that a Chinese sentence cannot be classified by words. Work in this paper is focus on solving background noise and storage problem.

Current solutions to oral speech voices and background noise are concentrated on HMM acoustic model self-adaptive algorithm. The training process to the acoustic model of the Chinese speech recognition based on semi-continuous hidden Markov model is as follows. In recent years, researchers proposed many ways to improve noise robustness, Normalized Regularized Minimum

978-1-4673-9795-7/15 $31.00 © 2015 IEEE
DOI 10.1109/CSCI.2015.68

Variance Distortionless Coefficients (NRMCC) and other based on neural network methods all can improve recognition accuracy in some ways [18][19].

This paper proposes a speech recognition method, which uses speech scenario adaptation and discourse topic recognition. In order to prove the effectiveness, we design and implement a continuous Chinese speech recognition system on a mobile smart terminal based on PocketSphinx engine. The advantage of this method is that it adds the scenario adaptation model and discourse topic recognition mode. Scenario Adaptation means training the acoustic model in a specified scenario to improve recognition accuracy. And discourse topic recognition is that the authors propose hybrid modeling based on Ngram model. In this method, the language model consists of general language model and a predicted discourse topic Ngram model. When performing the speech recognition, special discourse topic words library's hit rate threshold is used to dynamically judge the user's context discourse topic, to load the related discourse topic Ngram model and general model of the general words library, and to process user's voice input.

The rest of paper is organized as follows. Section II analyzes the scenario feature and the purposes to use acoustic model adjustment to adapt different scenarios. Section III presents the system that uses adjustment dictionary and language model to solve discourse topic recognition. The system structure and performance evaluation of the proposed method is analyzed in section IV. Section V concludes the paper.

## II. SCENE ADAPTATION METHOD USING ACOUSTIC MODEL

Speech recognition system on mobile terminals has to adapt various application contexts. It should consider not only the background noise but also different voice input channel. This paper defines a general model and uses parameter adjustment in different scenarios. Adjusting HMM model parameters and pronunciation of the training model in a special scenario can achieve scene adaptation of the speech recognition system on mobile terminals. Our experiment data have proved the effectiveness of the scene adaptation.

PocketSphinx has default acoustic models *hub4wsj_scc_8k* (English) and *tdt_sc_8k* (Chinese). The adjustment model can make the adjusted data more suitable to it. For example, adjusting acoustic model will make user have better dictation experience, make speech recognition system adapt recording environment, audio transmission channel, oral speech voices and so on. We use the professional and clear broadcast data and telephone speech data to adjust model, and get the expected speech acoustic model. Compared with training a new acoustic model, this method is more robust. In other words, if the adaptive data size is small, it will also obtain a satisfactory result. For example, 5-minutes speech data are enough to improve dictation accuracy for special adjustment.

The steps to adjust the acoustic model are given in Figure 1, where *tdt_sc_8k* is used as the example.

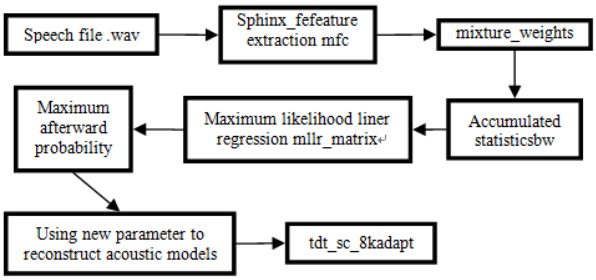


Figure 1. Training process of the Chinese acoustic model

The experiment selects scenarios that are typical in daily conversations. Scenarios should be in stationary noise environment with periodicity characteristics and avoid being in sudden nonstationary noise environment. The aim of the experiment is to prove the acoustic model adjustment effectiveness. Experiment scenarios are as follows:

- Quiet laboratory. Noise sources: opening a book, typing, sounds of machines
- Western restaurant with quiet music. Noise sources: music, chatting, sounds of tableware
- Classroom. Noise sources: talk from the teacher, chatting between students
- Outdoors. Noise sources: Engine of a car, sounds from a car horn

**Data preparation.** Pick up 500 sentences from daily conversations, which are about 1034 words. Then make dictionary *s500.dic* and language model *s500.lm*. Next, pick up 750 words or sentences from the training data set as testing data, record in four different scenarios, each is about 40 minutes, audio file sample rate is 16KHz, 16bit monaural recording, wav format. All recordings are in Mandarin.

**Recording adapt data.** Pick up 200 sentences from the data set. Record four adapt data set in four different scenarios: they have same content but different background noise. Time is about 4×10 minutes. Then obtain four acoustic models using previous acoustic model adjusting method. Finally, test before and after adjusting acoustic models in different scenarios, result is shown in Table I.

Experiments show that adjusting acoustic model can improve recognition accuracy of the speech recognition system in different scenarios, which is about 3%-9%.With the complexity of noise increasing and SNR decreasing, the number of words inserted, deleted or replaced has increased, and the recognition accuracy of the system decreased.

## III. DISCOURSE TOPIC RECOGNITION METHOD

Most people use a few areas of knowledge in daily life conversation. The vocabulary range is centralized in these areas, and the discourse topics and vocabulary in most other areas of knowledge are rarely used. In this paper, we propose a hybrid Ngram language model based on statistical language models. During Ngram model training period, we not only train the single general Ngram model, but also train predicted discourse topic Ngram model for many discourse topics.

TABLE I. COMPARISON OF BEFORE AND AFTER ADJUSTMENT OF ACOUSTIC MODELS UNDER DIFFERENT SCENARIOS

| Scenario | *Adjust* | *Accuracy (%)* | *Error Rate (%)* | *Insert* | *Delete* | *Replace* |
|---|---|---|---|---|---|---|
| Laboratory | N | 84.35% | 15.65 | 17 | 11 | 93 |
| Laboratory | Y | 89.15% | 10.85 | 15 | 10 | 81 |
| Western Restaurant | N | 83.55% | 16.45 | 24 | 15 | 99 |
| Western Restaurant | Y | 86.95% | 13.05 | 20 | 11 | 89 |
| Classroom | N | 82.45% | 17.55 | 30 | 21 | 118 |
| Classroom | Y | 86.05% | 13.95 | 27 | 14 | 101 |
| Outdoors | N | 72.75% | 27.25 | 49 | 31 | 133 |
| Outdoors | Y | 79.85% | 20.15 | 40 | 25 | 127 |

In the actual use, the speech recognition system determines the user's current dynamic scenario discourse topic by special discourse topic words library's hit rate threshold. The system dynamically loads the appropriate discourse topic Ngram model and combines with generic Ngram model in order to process and recognize the user's voice input. This method can significantly improve the speech recognition system's processing ability to user's personalized discourse topic in various context discourse topics. Figure 2 is the training and applying process of hybrid Ngram model.

PocketSphinx has a default Chinese dictionary and Chinese language model. The vocabulary size reaches more than 6000. The experiment shows that if the speech recognition system use this default Chinese dictionary and Chinese language model, the mobile smart terminal will run the application slowly and the recognition accuracy will not be good either. In order to meet the requirement of personalized input as well as performance, the system needs a general dictionary and language model with the vocabulary size is less than 1000 words, and a topic dictionary and language model for different discourse topic. The dictionary and language model can be generated in several ways. For example, CMUCLMTK and SRILM both have the tool to build statistical language models. If the vocabulary size is small (less than 200), CMU Internet service is a more convenient method to build a dictionary and language model.

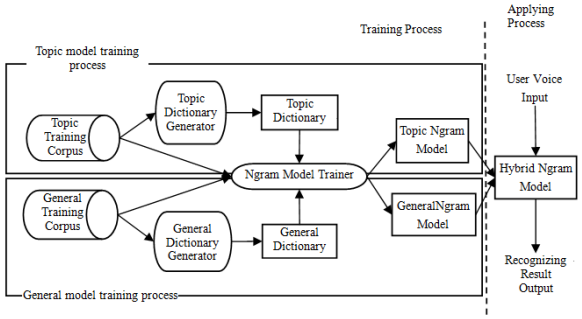


Figure 1. Training and applying process of the hybrid Ngram model

After getting the new discourse topic dictionary and the language model, the new dictionary needs to be integrated into the general dictionary, and the new topic language model needs to be integrated into the general language model. The integration of dictionary only needs to add and insert them correctly by the words arrangement characteristics of the dic (Chinese phoneme combination arrangement). The integration of language model needs interpolation, such as SEILM has many language model integration commands.

Using the discource topic recognition model, speech recognition can support more verticals better, such as map, music, game and so on. Template designing and training corpus in specific areas can give customized services to specific users or merchants. Disource topic selecting mostly depends on words library division of different knowledge fields. Words library division and corpus selecting need large amount of data. The experiments in this paper build three kinds of typical disource topic words libraries: Champion's League football, Java vocabulary, and Guangzhou Metro, as shown in Table II.

Using these discourse topics to make discourse topic dictionaries and train language models, the system can obtain *.dic* file and *.lm* file:

*eur_fball.dic* —— *eur_fball.lm*

*java.dic* —— *java.lm*

*metro.dic* —— *metro.lm*

Generally speaking, the vocabulary size of a special discourse topic is about 50 - 200, and the size of a general vocabulary is about 500 - 2000.

Then, the system obtains a hybrid dictionary and a hybrid Ngram language model with SRILM tools. There are general dictionary *topic_g*, general+Champions League football dictionary *topic_ge,* general+java dictionary *topic_gj*, general+Guangzhou Metro dictionary *topic_gjm*, general+Champions League football+java dictionary *topic_gej*, general+java+Guangzhou Metro dictionary *topic_gjm*, general+Champions League football+ Guangzhou Metro dictionary *topic_gem*, general+Champions League football+java +Guangzhou Metro dictionary *topic_gejm*.

TABLE II. DISCOURSE TOPIC, WORDS EXAMPLES AND CORPUS

| Scenario | *Words Example* | *Corpus* |
|---|---|---|
| Champions League football | Arsenal, Manchester United, Barcelona, Real Madrid, Messi, Xavi, Casey, corner kick, shoot, save, flank pass, the three-nil…… | League game live, sports news |
| Java | Constructor, initialization, the object, call, member variables, overloading, polymorphism, encapsulation, multidimensional arrays, commissioning, abstract classes, interfaces..... | Java teaching video，Related books |
| Guangzhou Metro | Metro Line 4, Higher Education Mega Center South, Wanshengwei Station, Tiyu Xilu Station, Wu Shan Station, transfer, the next station, terminal, crush, get off and then on, YangChengTong-Card | Metro radio information, Metro name |

In order to prove the hybrid dictionary and the language model is effective, it is necessary to use the speech recognition system to do the general dictionary recognition and special discource topic dictionary recognition. For example, for the general+Guangzhou metro dictionary *tipic_gm*, we will select words or sentences in the general dictionary to do the test, then select words or sentences in the special topic dictionary to do the test. Finally, use words from different dictionaries and put them into sentences to do the test. The experiment uses random sampling, and the amount of samples is about 20% of the dictionary size. Because the testing result of seven hybrid dictionaries is similar, the table shows the average value, as is shown in Table III.

It is apparently that if the system uses the hybrid words library, the recognition accuracy is very high, which is above 80%. The result proves that the language model adjustment method based on linear interpolation algorithm is effective. However, the accuracy can be improved in the future, especially when the source is combination of different words libraries. Because in the statistical level, the special discourse topic words library is a negligible portion of the total dictionary, there may have data sparseness problem. The simple linear interpolation algorithm could not solve this problem, stricter semantic constraints or identify caching mechanism is considerable.

## IV. Case Study: Android Speech Recognition System

This section presents the implementation and testing of a continuous Chinese speech recognition system on Android.

### A. System Structure

The structure is shown in Figure 3. Components of the system is elaborated as following.

*1) Front-end processing model in client-side:* A/D converter converts the analog voice signal into digital voice signals, generates audio signal files, and transfers it into the proper format the speech recognition system needed, such as wav file. Voice input is processed before speech recognition. It needs to separate speech and silence signal via endpoint detection. Adding a window frame and pre-emphasis can obtain stable signal.

TABLE III. Recognition Accurancy of the System

| Source | *Recognition Accuracy* |
|---|---|
| General dic | 87.50% |
| Special topic dic | 83.00% |
| Combination of different dics | 81.50% |

Then the system does Mel cepstrum feature extraction, and finally uses 51-dimensional feature parameter vector to characterize the speech signal for one frame.

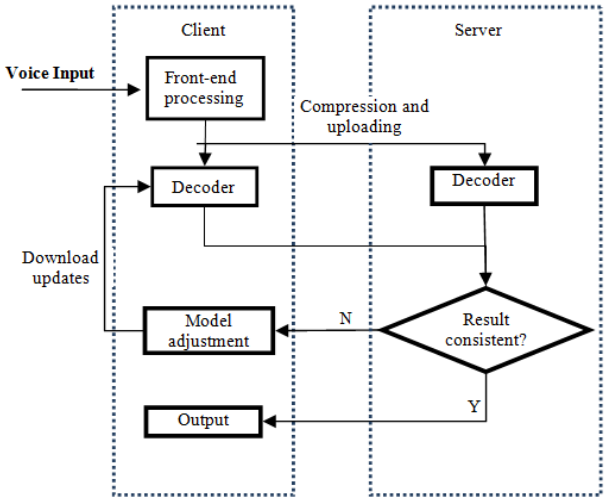


Figure 2. The structure of speech recognition system

*2) Decoder module:* Decoder consists of acoustic model, dictionary, language model, and speech decoding algorithm. Both of client and server have decoder, but they are not the same. The decoder in client requires light computational load, small storage, and fast recognition speed. Thus, we need to simplify the acoustic model, dictionary, language model, and even the speech decoding algorithm, which may lead lower recognition accuracy. On the server side, the decoder uses distributed computing and storage, which can provide high-speed data processing with large volume of data. As a result, the acoustic model, dictionary, language model and speech decoding algorithm in the decoder of server can be much robuster. Furthermore, the client isn't able to train these models, except for adjusting them. All the trainings of models are done in the server.

*3) Output model:* The result is given in text format. In off-line state, the system directly shows the search results. In online state, the client will decode and send the result to the server and compares it with decoding result of the server. If two results are not consistent, the system will do model adjustment. In online state, when the system selects the output, decoding result of the server shall prevail, whatever the decoding result of the client is. The server will record users' information, which is used to train personalized model.

*4) Model adjustment:* Model adjustment has two aspects, Scene adaptation and discourse topic recognition.

### B. Influence of Vocabulary to Accuracy and Speed

Definition of recognition speed: Assume the audio file's playtime is $p$. If the decoding time is $d$, then the recognition speed is $d/p \times RT$. For example, if the play time is two hours, and the decoding time is 6 hours, the speed is $3 \times RT$. In this part, the experiments will test the relationship between the vocabulary size and recognition accuracy and speed. In the experiment, the language models use unigram, bi-gram and tri-gram. Dictionary and language models are divided into

eight levels according to the vocabulary size, which is shown in Table IV.

The experiments we done are on the mobile phones with Android 4.0 OS, dual-core 1.5GHz and 1GB memory. Tests are divided into eight levels. We record sentences of 1 second, 5 seconds, and 20 seconds and 1minute, so the size of the sentences voice data is about 10×8×4, selecting from the training sets. The vocabulary size of the words voice data is about 10%-40% of the dictionary. Using the recognition speed API provided by PocketSphinx and the recognition accuracy testing tools provided by Sphinx, authors did the test to the speech recognition system. The result is shown in Table V.

TABLE IV. DICTIONARY AND LANGUAGE MODELS ON DIFFERENT LEVELS

| **Dictionary** | ***Language Model*** | ***Sentences*** | ***Vocabulary*** | ***Storage Space/KB*** |
|---|---|---|---|---|
| s10.dic | s10.lm | 10 | 15 | 2.1 |
| s20.dic | s20.lm | 20 | 45 | 5.5 |
| s50.dic | s50.lm | 50 | 90 | 10.7 |
| s100.dic | s100.lm | 100 | 160 | 20.3 |
| s200.dic | s200.lm | 200 | 366 | 48.1 |
| s500.dic | s500.lm | 500 | 1034 | 146.6 |
| s1000.dic | s1000.lm | 1000 | 2259 | 376.8 |
| s2000.dic | s2000.lm | 2000 | 3471 | 671.6 |

TABLE V. INFLUENCE OF VOCABULARY TO RECOGNITION ACCURACY AND SPEED

| **Vocabulary (Dictionary)** | ***Recognition*** | | ***Recognition Speed (average value)/xRT*** |
|---|---|---|---|
| | ***Sentence*** | ***Word*** | |
| 15 (s10.dic) | 98.00% | 95.25% | 1.10-1.43(1.2) |
| 45 (s20.dic) | 97.75% | 90.75% | 1.12-1.45(1.2) |
| 90 (s50.dic) | 97.00% | 83.75% | 1.13-1.46(1.3) |
| 160 (s100.dic) | 94.75% | 76.50% | 1.15-1.49(1.3) |
| 366 (s200.dic) | 90.25% | 67.50% | 1.18-1.54(1.3) |
| 1034 (s500.dic) | 86.25% | 55.00% | 1.28-1.90(1.4) |
| 2259 (s1000.dic) | 80.75% | 38.25% | 1.34-2.18(1.6) |

The result in Table V shows that with the increase of the vocabulary size, the recognition accuracy becomes lower and lower, which is the same as the recognition speed. We believe that the recognition rate should be at least 80% to meet the requirement of speech recognition function. Thus, when the vocabulary size is less than 2000, the performance of sentence recognition is satisfying. However, when it goes to word recognition, the system could have misjudgment to homonym or polysemy. The reason is that only the unigram works when the system recognizes a single word,

Furthermore, the recognition speed is acceptable when the vocabulary size is less than 2000 (in 2 seconds). The recognition speed will fluctuate within a certain range: the speed is slow when recognizing short sentences or words, and it can be fast when recognize long sentences. The reason is that a long sentence can become constrained by multiple syntaxes, which can shorten searching time. The storage space is another factor to influence the recognition speed.

### *C. Tests to The Occupancy of CPU and The Memory*

Testing the occupancy of CPU and memory is an important part to know if an Android application meets the performance requirement.TOP command is commonly used as a performance analysis tool in Linux,which can show the real-time system resource usage status of each process.

The experiment chooses s50.dic, s50.lm, trained by 50 sentences, and s1000.dic, s1000.lm, trained by 1000 sentences. TOP command, clock timing and Android app start at the same time. In order to effectively analyze the occupancy of CPU and the memory at all stages, the experiment use "standby 5seconds after start—recognizing 2seconds—standby 10seconds—recognizing 20seconds—standby 20seconds" to record the test results. TOP command uses *adb shell top –d 0.1 |grep* speech recognition, which means the duration between sample points is equal or less than 0.1 second. The results are shown in Figure 4 and Figure 5.

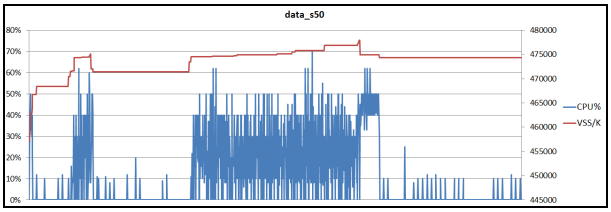


Figure 3. Occupancy of CPU and memory when the vocabulary size is s50

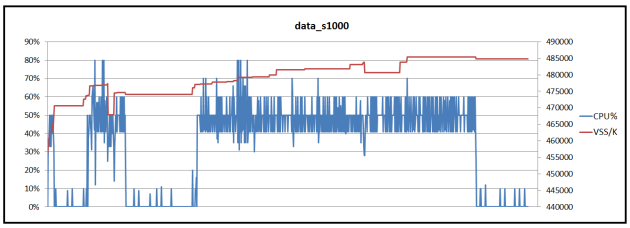


Figure 4. Occupancy of CPU and memory when the vocabulary size is s1000

In the first second, the speech recognition app starts initialization, occupying about 50% of CPU. Then the app stands by. Memory usage no longer increases, and CPU occupancy rate starts reducing. The speech recognition system has specialized resource to clean up the voice input signal resources at regular intervals, which can both deal with signal processing lag and avoid speech data occupying a large amount of memory. In the recognizing period, CPU occupancy rate and memory usage rise sharply. Then in standing by for a period of 10 seconds, CPU occupancy rate becomes 0 and memory usage reduces. In recognizing the

20 seconds period, because of the long recognizing time, CPU occupancy rate keeps high and memory usage increases gradually. Finally, in standing by 20 seconds period, the app releases resource.

Note, in recognizing 20 seconds period, data_s50 only uses little time to process and release resources, and thus CPU occupancy rate curves a number of peaks and valleys. It only costs 3 seconds to process speech data. Data_s1000 keeps high CPU occupancy rate during processing period and the recognition speed is slower. In addition, there are significant differences between the local processing resource releasing and overall resource application.

## V. CONCLUSIONS

Speech recognition technology is an active research direction in the field of human-computer interaction. It can combine the physical devices and the services provided by the city. In this research, we designed and implemented a continuous Chinese speech recognition system on an Android mobile smart terminal. Based on the acoustic model training process, we used the maximum likelihood linear regression (MLLR) algorithm and maximum a posteriori probability (MAP) algorithm to adjust acoustic models for different scenarios. Based on the analysis of discourse topic features and Ngram language model training process, we developed a linear interpolation algorithm to fuse the language model.

In our future research, efforts are being made in data collection and data exploitation. Unconscious data collection can be useful. Voice sensors and GPS can catch more available data. These data can be sent to cloud server by GSM or WIFI. However, data uploading should have no influence to other applications. Data collection requires internet bandwidth and power. An effective incentive measure and crypto security measure is necessary. The value of data can be recognized in behavior analysis and context recognition. An intelligent service in the smart city can receive large amounts of data continuously. Currently, we only save data instead of processing them. The server can starts processing them when the mobile terminal start the application, match the related database, recognize activities, select activities that the user may be interested, display them on the map.

## ACKNOWLEDGMENT

This work is supported in part by grants from two Guangdong province science and technology planning projects of China (No.2014A040401018 and No. 2013B040404009).

The authors declare that there is no conflict of interests regarding the publication of this article.

## REFERENCES

[1] Chen, Yinong, Hualiang Hu, “Internet of intelligent things and robot as a service”, Simulation Modelling Practice and Theory Volume 34, May 2013, Pages 159–171.

[2] Yinong Chen, W.T. Tsai, Service-Oriented Computing and Web Software Integration, 5th edition, Kendall Hunt Publishing, 2015.

[3] Deren Li, Jie Shan, Zhengfeng Shao, “Geomatics for smart cities-concept,” Key Techniques and Applications, vol.16, No.3, 2013, pp. 13-24.

[4] WikiPedia, “Flating car data.” http://en.wikipedia.org/wiki/Floating_car_data

[5] Ruzhi Xu, Shuaiqiang Wang, Xuwei Zheng, Yinong Chen, "Distributed collaborative filtering with singular ratings for large scale recommendation, Journal of Systems and Software, Volume 95, September 2014, pp. 231-241.

[6] Y. Zheng, L. Zhang, X. Xie, “Mining interesting locations and travel sequences from GPS trajectories,” Proceedings of the 18th international conference on World Wide Web, Spain, pp. 791-800.

[7] Y. Zheng, Y. Chen, Q. Li, “Understanding transportation modes based on GPS data for web applicaition,” ACM Transaction on the Web(TWEB), New York, 2010, vol.4, No.1, pp. 1.

[8] I Stojmenovic, Keynote I, “Mobile cloud and crowd computing and sensing,” 2012 IEEE 18th Intenational Conference(ICPADS): Parallel and Distributed Systems, Singapore, 2012, pp. xxix.

[9] S. Krishnaswamy, Gama. J, Gaber. M. M, “Mobile data stream mining: From algorithms to applications,” IEEE 13th International Conference: Mobile Data Management, India, 2012, pp. 360-363.

[10] H. K, Davis, R. Biddulph, S. Balashe, “Automatic recognition of spoken digits,” J.Acoust.Soc.Am, 1952, vol. 24, No.6, pp. 637-642.

[11] Yunpeng Xiao, Weibin Zhu, “Subspace Gaussian mixture models for Chinese speech recogtion,” Beijing Jiaotong University, 2013, pp. 1567-1541.

[12] Yongzhe Shi, Weiqiang Zhang, Jia Liu, “RNN language model with word clustering and class-based output layer,” EURADIP Journal on Audio, Speech and Music Processing, 2013, pp. 22.

[13] Andrew Ng, Jquan Ngiam, Chuan Yu Foo, “unsapervised feature learning and deep learning,” Deeplearning.stanford.edu/wiki/index.php

[14] G. E. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, et al. “Deep neural networks for acoustic modeling in speech recognition,” IEEE Signal Processing Magazine, 2012, vol.29, pp. 82-97.

[15] Tara Sainath, Brain Kingsbury, Vikas Sindhwani, Ebru Arisoy, and Bhuvana Ramabhamran, “Low-rank matrix factorization for deep neural network training with high-dimensional output targets,” IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP), Canada, 2013, pp.6655-6659.

[16] F. Seide, G. Li, and D. Yu, “Conversational speech transcription using context-dependent deep neural networks,” Interspeech, 2011, pp. 437-440.

[17] B. Kotnik, Z. A. Kacic, “Niose robust feature extraction algorithm using joint wavelet packet sub-band decompositon and AE modeling of speech signals,” Signal Processing, 2007, voi. 87, No.6, pp. 1202-1223.

[18] M. J.Alam, P. Kenny, D. O’Shaughnessy, “Regularized MVDR spectrum estimation-based robust feature extractors for speech recognition,” Interspeech, 2013, pp. 891-895.

[19] D. Yu, M. L.Seltzer, “Improved bottleneck features using pretrained deep neural networks,” Interspeech, 2011, pp. 237-240.