뉴스에서 시멘틱 디코딩의 음성대화시스템을 위한 히든 벡터 상태 마코브모델의 상세설계

레콩탄*
*동국대학교 컴퓨터공학과
e-mail : thanhlct@gmail.com

A Detailed Design of Hidden Vector State Markov Model for Semantic Decoding of Spoken Dialogue System on News

Thanh Cong Le*
*Faculty of Computer Science, Dongguk University

Abstract

Nowadays, Spoken Dialogue System is rapidly growing by investing a lot from researches as well as organizations. One of concrete evidences is that the appearance of commercial systems such as Siri, SVoice, DARPA, CLASSiC, GSearch etc. Moreover, Spoken Dialogue System is widely believed to be the future direction of software development. In Spoken Dialogue System, users interact to software by using their own voice instead of use their hands, keyboard, and mouse. This paper continuously presents our development of the Spoken Dialogue System on News. Particularly, we propose detailed design such as semantic concepts, semantic frames, slots, and so on for applying Hidden Vector State Model into our Spoken Dialogue System for Spoken Language Understanding.

1. Introduction

Spoken Dialogue System (SDS) is a system which is able to communicate to users by using natural spoken language as a human. In traditional, we interact to computers by using our hands. With SDSs, our hands are free. You interact to a computer by your own voice. For example, if you would like to know the weather in Seoul on tomorrow, instead of you have to open a web browser and go to Google search to looking for the weather in Seoul, you just talk to a SDS "What is the weather in Seoul tomorrow?" or "Do you know the weather in Seoul tomorrow?" and even is "Do I need an umbrella at Seoul tomorrow?". Then the SDS will answer you like your own assistant. You could realize that SDS makes applications more flexible, faster, and friendly to users and so on. Moreover, you image that a smart SDSs which is integrated on mobile devices then SDS's value is increasing much more. In other words, SDS makes a new paradigm, the new paradigm for future of software development.

Spoken Dialogue on News [1] is the system which is able to automatic collect news from several famous websites such as BBC, Yahoo News, CNN and MSNBC etc. and is able to read a news or a list of news follow requirement of users. In doing so, users will not need to touch to a computer to know about the world by reading news. All users need to do talk to the system which news you wanted. Moreover, users are feeling comfortable, convenient. Specifically, means of the system to disenable human, old people.

In this paper, we continuously present the development of Spoken Dialogue System on News. In the [1], we figured out overall system architecture, along with models and methods will be applied into our system. In particular, we figure out explicit designs to implement Semantic Decoding by using Hidden Vector State Model.

The goal of the Semantic Decoding is to annotate each natural language input so as to allow these attribute-value pairs. A particularly simple form of annotation results from mapping each word w in an utterance W into a single discrete concept c. For example, the utterance "List flights from Boston to Denver" might be encoded as:

```
DUMMY(List) FLIGHT(flights) FROMLOC(from) CITY(Boston)
TOLOC(to) CITY(Denver)
```

Where DUMMY, FLIGHT, FROMLOC, CITY are semantic concept. These concepts help a SDS find out the user's desire or what is the meaning of the utterance. We will present one of best methods for Semantic Decoding in this paper. Under erroneous of Speech Recognition then, Semantic Decoding has an important task for making accurate conversation.

The structure of the paper is as follows. In the section 2 is some relate works on Spoken Language Understanding and Dialogue Manager. Then section 3 describes necessary designs for applying Hidden Vector State (HVS) model on our system, Spoken Dialogue System on News, to implement a Semantic Decoding of Spoken Language Understanding (SLU). Finally, the section 4 is the conclusion and future works.

2. Related Works

Semantic Decoding, traditionally, have been built using hand-crafted semantic grammar rules. Word pattern corresponding to semantic tokens are fill slot. In [4; 5], a dialogue act tagging has been developed to code various

levels of dialogue information about utterance. However, in this research, dialogue acts are analogues to user's intentions or information goals of an utterance. In [6, 7, 8], using grammar rules to fill slot different semantic frames in parallel. The frame with highest score yields the semantic representation. These approaches are called robust parsing.

In the other hand, other systems view semantic parsing as a pattern recognition problem and adopt statistical approaches to derive semantic representations. In this section, we mentioned to several major models.

Finite-state tagger (FST) treats the generation of a Spoken sentences as HMM-like process whose hidden states correspond to semantic concepts and output correspond to individual words in the utterance. One of the most notable FST models is CHRONUS [9, 10, 11, 12] developed by AT&T. In the system, words and phrases are classified into categories in order to reduce effective size of the lexicon [9].

Stochastic Context-Free Grammar models are non-lexicalized such as BBN's Hidden Understanding [thesis 13, 14], hierarchical HMMs [15, 16], immediate-based parsing models [17] and structured language model [18, 19]. These models overcome the drawbacks of FST that are the flat left-right structure does not allow any hierarchical grouping of the concepts. The lack of hierarchical group weakens the prediction ability, specially is the long distance dependences. Moreover, Stochastic Context-Free Grammar models with hierarchical group facilitate to create preterminal concepts, thus, avoid fragmenting the training data. For example, we use a concept CITY instead of using two concepts TOPLACE and FROMPLACE. Thus, The city "Boston" will included in only one concept CITY instead of two concepts FROMPLACE, TOPLACE.

We also have lexicalized models [17, 20, 21, 22], history based models [20, 23, 24]. Specially, in recent, researchers investigate to use Reinforcement Learning for Spoken Language Understanding [27, 28].

3. Semantic Decoding with Hidden Vector State Model

Semantic Decoding maps the word string recognized onto a sequence of predefined semantic labels or concepts. In particular, each of word or phrase is tagged by specific concept as preterminal. For examples, the phrase "to go" is assigned the concept TRAVEL, the work "Boston", "Denver" tagged the concept CITY. In HVS, these concepts are built hierarchical structures as in Figure 1.

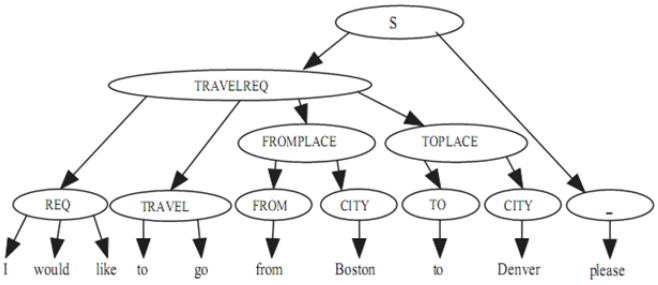

FIGURE 1 - A Representation of Semantic Decoding by a hierarchical model as HVS [3]

As figured out in [2], to build a Semantic Decoding use HVS in specified domain, we need to list a set domain specific lexical classes and abstract annotation for each utterance. In our system, the domain is interaction to news. Particularly, user asks for list of all news or list of news follow an interested subject. Then, user asks for reading specified news. Based on expectation, we propose domain specific lexical classes and abstract annotation for each utterance as followings.

With domain specific lexical class, we have:

- Magazines (Yahoo News, MSNBC, BBC etc.)
- Genres (Music, Entertainments, Word News, Sports, etc.)
- Locations (England, US, Asia, etc.)
- Organization (Microsoft, Apple, Google, Samsung, Stanford University etc.)
- People (Barack Obama, Steve Jobs, Bill Gates, Westlife, Wonder Girls etc.)
- Products (Surface, iPhone, iPad, Galaxy S III, etc.)
- Events (Word cup, Olympic, etc.)
- Numbers (first, second, third, fourth, etc.)
- … …

With abstract annotation for each utterance, we have:

- LISTNEW(all)
- LISTNEWS(on(MAGAZINE))
- LISTNEWS(of(GENRE))
- LISTNEWS(at(LOCATION))
- LISTNEWS(of(ORGANIZATION))
- LISTNEWS(of(PERSON))
- LISTNEWS(of(PRODUCT))
- LISTNEWS(of(EVENT)
- LISTNEWS(on(MAGAZIE) about (GENRE))
- LISTNEWS(on(MAGAZIE) about(LOCATION))
- LISTNEWS(on(MAGAZIE) about(ORGANIZATION))
- LISTNEWS(on(MAGAZIE) about(PERSON))
- LISTNEWS(on(MAGAZIE) about(PRODUCT))
- LISTNEWS(on(MAGAZIE) about(EVENT))
- LISTNEWS(on(MAGAZINE) of(GENRE) at(LOCATION) of(ORGANIZATION) of(PERSON) of(PRODUCT) of(EVENT))
- STOP()
- READNEWS(order(NUMBER))
- READNEWS(about(LOACTION))
- READNEWS(about(ORGANIZATION))
- READNEWS(about(PERSON))
- READNEWS(about(PRODUCT))
- READNEWS(about(EVENT))
- … …

Note that if users mention a news's title instead of the order of this news, the system will use a simple matching the title to a list of titles which the SDS has read to find out the order of desired news.

On other hands, semantic-aware or discourse is indispensable in a good SDS. For examples, a user talk that "Please read the Microsoft news", then our SDS can understand if in the list shown to users has only one news about Microsoft, or even several Microsoft news existed, Concretely, it is an unobvious. However, the SDS should ask user again like as "Which news do you like in three news,

title 1, title 2, and title3".

When build a system from scratch, a dialogue designer can take each possible abstract annotation and give examples of corresponding natural language forms. For examples, with abstract annotation LISTNEWS(on(MAGAZINE)), we have:

- List news on M.
- Could you please tell me news from M.
- I wanna to know news on M
- Etc.

Where M is an instance of MAGAZINE such as BBC, MSNBC and so on. If we have available corpus of representative user utterances, then each utterance can easily be tagged with the appropriate abstract annotation.

In HVS, next step will be preprocessing with expanding abstract annotation to flattened concept sequence. This preprocessing based on the sequence of vector state used for mapping a pattern utterance. Each of flattened concepts is concatenation of concepts in a vector state represents for a word}. For examples, with the abstract annotation LISTNEWS(on(MAGAZINE(BBC))) combined with a specified utterance and its parse tree and its vector state as Figure 2, we will have flattened concepts as followings:

- LISTNEWSON+LISTNEWS
  LISTNEWSON+ON
  LISTNEWSON+ON+MAGZINE(BBC)

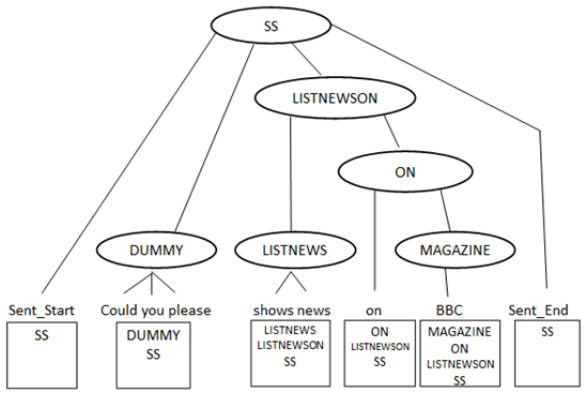

FIGURE 2 – A parse tree and its vector state of the abstract annotation LISTNEWS(on(MAGAZINE(BBC)))

In natural speech, user probably uses many irrelevant input words since the non-fluency, full sentence and grammar and so on. Therefore, a DUMMY tag is allowed everywhere in preterminal position to presents these words. In order to accommodate this, the above vector state sequence is finally expanded to:

- LISTNEWSON+LISTNEWS
  LISTNEWSON+LISTNEWS+DUMMY
  LISTNEWSON+ON
  LISTNEWSON+ON+DUMMY
  LISTNEWSON+ON+MAGZINE+BBC
  LISTNEWSON+ON+MAGZINE(BBC)+DUMMY

With the expansion, the model only tag DUMMY at preterminal. Thus, if the model occurs consecutive irrelevant word inputs, the model will stay in the same vector until observe a relevant word input.

As described in [3], the next step is parameter initialization. The HVS model contains three main components. They are the stack shift operation, the push of a new preterminal semantic tag, and the selection of a new word. Thus each vector state is associated with three sets of probabilities: a vector stack shift probabilities, a vector of semantic tag probabilities and a vector of output probabilities. Three probabilities described respectively in equations followings:

$$P(n_t|W_1^{t-1}, \mathbf{C}_1^{t-1}) \approx P(n_t|\mathbf{c}_{t-1})$$
$$P(c_t[1]|W_1^{t-1}, \mathbf{C}_1^{t-1}, n_t) \approx P(c_t[1]|c_t[2..D_t])$$
$$P(w_t|W_1^{t-1}, \mathbf{C}_1^{t}) \approx P(w_t|\mathbf{c}_t)$$

Three sets of probabilities are determined by the maximum vector state stack depth, the number of preterminal semantic tags, and the vocabulary size, respectively. Hence, the total information required to define an HVS model is:

- Number of distinct vector states
- Number of preterminal semantic tags
- Maximum vector state stack depth.
- Vocabulary size
- For each vector state:
  - State name
  - Stack shift operation proability vector
  - Tag transition probability vectors (push of a new preterminal tag)
  - Output probability vector?

We should use a prototype definition which uses a format similar to that described for HMM definitions in the HTK Toolkit [25] or use a itself defines syntax based on standard XML. Thus, we have an initial HVS for our SDS as described in Table 1 followings.

```
~o
<NumStates>1000
<NumTermTags> 50
<MaxStackDepth> 4
<VocabSize> 611
<BeginHVS>
        <State> 1
                <StaeName> SS
                <StackOP> 0.200*5
                <TagTrans> 0.015
                <Output>
                0.000*255  0.000*255 1.000  0.000*130
        <State> 2
                <StaeName> SS + DUMMY
                <StackOP> 0.200*5
                <TagTrans> 0.015
                <Output>
                0.000*255  0.000*255  0.000*101
        <State> 3
                <StaeName> SS + LISTNEWSON
                <StackOP> 0.200*5
                <TagTrans> 0.015
                <Output>
                0.000*255  0.000*255  0.000*101
<EndHVS>
```

TABLE 1 – Prototype Definition of Initial HVS for SDS on News

The next step will be parameter re-estimation, all of initial parameters are then iteratively refined using the Expectation-Maximisation (EM) [26]. EM-based re-estimation formulation as in [3] figured out that are:

$$\hat{P}(n|\mathbf{c}') = \frac{\sum_t P(n_t = n, \mathbf{c}_{t-1} = \mathbf{c}'|W, \lambda)}{\sum_t P(\mathbf{c}_{t-1} = \mathbf{c}'|W, \lambda)},$$

$$\hat{P}(c[1]|c[2..D]) = \frac{\sum_t P(\mathbf{c}_t = \mathbf{c}|W, \lambda)}{\sum_t P(c_t[2..D] = c[2..D]|W, \lambda)},$$

$$\hat{P}(w|\mathbf{c}) = \frac{\sum_t P(\mathbf{c}_t = \mathbf{c}|W, \lambda)\delta(w_t = w)}{\sum_t P(\mathbf{c}_t = \mathbf{c}|W, \lambda)},$$

Where $\delta(w_t=w)$ is one iff the word at time is w, otherwise it is zero. For references more, readers suggested refer to [3].

With HVS, we do not need an annotated data to training model, as well as do not need to list specified explicit pattern or utterances.

The output of Semantic Decoding, sequence of semantic concepts, works as inputs for Dialog Act Detection to find out dialog act carried by the utterance.

4. Conclusion and Future Works

In this paper, we show detailed design for applying Hidden Vector State Model to Semantic Decoding of our Spoken Dialogue System on News. We present concretely semantic concepts, as well as transferring abstract annotation to flattened concept sequence, along with prototype definition of a Hidden Vector State Model. We also use EM to estimate the model parameters from observed data. A good Semantic Decoding reduces much more errors of Speech Recognition.

We will continuously finish the Spoken Dialogue System by applying TANs and Partially Observable Markov Decision Process for Dialog Act Detection and Dialogue Manager.

References

[1] Thanh Cong Le. “Spoken Dialogue System on News”. Submitted to 39th Conference of Korea Institute of Information Scientist and Engineers (KIISE) , 2012.

[2] Steve Young, “The Statistical Approach to Design of Spoken Dialogue Systems”, Cambridge Technical Report, 2002.

[3] Yulan He, Steve Young. “Semantic Processing Using the HiddenVector State Model”. Computer Speech and Language, Elsevier, 2005.

[4] J. Carletta, A. Isard, J.C. Kowtko, G. Doherty-Sneddon, A.H.Anderson. “The Reliability of a Dialogue Structure Coding Scheme”. Computational Linguistics, 1997.

[5] J. Allen, M. Core. “Draft of DAMSL: Dialogue Act Markup in Several Layers”. 1997

[6] W. Ward, S. Issar. “Recent Improvements in the CMU Spoken Language Understanding System”. Proc. Of the ARPA Human Language Technology Workshop. 1996.

[7] S. Seneff. “Robust Parsing for Spoken Language Systems”. IEEE, 1992.

[8] J.Dowding, R. Moore, F. Andry, D. Moran. “Interleaving Syntax and Semantics in an Efficient Bottom-Up Parse”. ACL, 1994.

[9] R. Pieraccini and E. Levin, “Stochastic Representation of Semantic Structure for Speech Understanding”. Prof. of Eurospeech, 1991.

[10] R. Pieraccini, E. Levin, C. Lee. "Stochastic representation of Conceptual Structure in the ATIS Task". The DARPA Speech and Natural Language Workshop, 1991.

[11] R. Pieraccini, E. Tzoukermann, Z. Gorelov, E. Levin, C. Lee, J. Gauvain. “Progress Report on the CHRONUS System: ATIS Benchmark Results”. The DARPA Speech and Natural Language Workshop, 1992.

[12] E. Levin, R. Pierarccini. “CHRONUS, the next Generation”. The DARPA Speech and Natural Language Workshop, 1995.

[13] S. Miller, M. Bates, R. Bobrow, R. Ingria, G. Makhoul, R. Schwartz. “Recent Progress in Hidden Understanding Models”. The DARPA Speech and Natural Language Workshop, 1995.

[14] R. Schwartz, S. Miller, D. Stallard, J. Makhoul. “Language Understanding Using Hidden Understanding Models. Intl. Conf. on Spoken Language Procesing. 1996.

[15] S. Fine, Y. Singer, N. Tishby. “The Hidden Markov model: Analysis and Application”. Machine Learning. 1998.

[16] K.P. Murphy, M.Paskin. “Linear Time Inference in Hierarchical HMMs”. Proc. of Neural Information Processing Systems. 2001.

[17] E. Charniak. “Immediate-Head Parsing for Language Models”. Proc. of the 39th Annual Meeting of the Association for Computational Linguistics, 2001.

[18] C. Chelba, F. Jelinek. “Structured Language Modeling”. Computer Speech and Language, 2000.

[19] H. Erdogan, R. Sarikaya, Y. Gao, M. Picheny. “Semantic Structured Language Models”. Proc. of Conf. on Spoken Language Processing, 2002.

[20] F. Jelinek, L. Lafferty, D. Magerman, R. Mercer, A. Ratnaparkhi, S. Roukos. “Decision Tree Parsing Using a Hidden Derivation Model”. Proc. of the 1994 Human Language Technology Workshop, 1994.

[21] D. Magerman, “Statistical Decision-Tree Models for Parsing”. Proc. of the 33rd Annual Meeting of the Association for Computational Linguistics, 2001.

[22] E. Charniak. “Statistical Parsing with a Context-Free Grammar and Word Statistics”. Prof. of the Fourteenth National Conference on Artificial Intelligence, AAAI, 1997.

[23] C. Chelba, P. Xu. “Richer Syntactic Dependencies for Structured Language Modeling”. Proc. of the Automatic Speech Recognition and Understanding Workshop, 2001.

[24] C. Chelba, “Portability of Syntactic Structure for Language Modeling”. Proc. of the IEEE Intl. Conf. on Acoustics, Speech and Signal Processing, 2001.

[25] Young, S., Evermann, G., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P., The HTK Book. Cambridge University Engineering Department, 2004.