Combining Statistical and Syntactical Systems for Spoken Language Understanding with Graphical Models

S. Schwärzler, J. Geiger, J. Schenk, M. Al-Hames, B. Hörnler, G. Ruske, G. Rigoll

Institute for Human-Machine Communication
Technische Universität München
80290 Munich, Germany
{sts,gej,joa,alh,hbe,rus,ri}@mmk.ei.tum.de

Abstract

There are two basic approaches for semantic processing in spoken language understanding: a rule based approach and a statistic approach. In this paper we combine both of them in a novel way by using statistical and syntactical dynamic bayesian networks (DBNs) together with Graphical Models (GMs) for spoken language understanding (SLU). GMs merge in a complex, mathematical way probability with graph theory. This results in four different setups which raise in their complexity. Comparing our results to a baseline system we achieve a F1-measure of 93.7% in word classes and 95.7% in concepts for our best setup in the ATIS-Task. This outperforms the baseline system relatively by 3.7% in word classes and by 8.2% in concepts. The expermiments were performend with the graphical model toolkit (GMTK).

Index Terms: natural language understanding, machine learning, graphical models

1. Introduction

Semantic processing is one of the key elements in spoken dialog systems. It analyzes the users query and produces a representation of its semantic content that allows the dialog manager to take context-sensitive decisions about the dialog follow-up [1, 2, 3, 4, 5]. In [6] we introduced the hierarchical decoding in order to gain information about the meaning of the spoken sentences. There each sentence is decoded by identifying so-called concepts, like "origin", "destination" or "time", and word classes belonging to each concept, like "cities", "fromloc" or "toloc". A concept depends on the words, however only specific words may belong to a concept. As a result a city name is not only identified as being "city" but also as the concept in which the word occurs. The best word concept hypothesis is found in a maximum likelihood (ML) manner by the well known Viterbi-algorithm. However, in [6] the grammatical rules have to be explicitly defined. In this work we follow the approach of hierarchical decoding, with the advantage that the grammatical bindings are modeled by Graphical Models (GMs). Their parameters are learned automatically providing a hierarchically annotated training set. This paper is organized as follows. In Sec. 2 we introduce language understanding in general, in Sec. 3 GMs are explained in full detail and how they can be adopted for language understanding, in Sec. 4 different language models are introduced. In Sec. 5 the models we used for our various systems are explained, these are evaluated in Sec. 6 on a state-of-the-art corpus. Finally, we conclude in Sec. 7.

2. Language Understanding

Semantic processing is defined as an automatic mapping between words W output by the automatic speech recognition to a sequence of word classes (labels) L and concepts C needed to perform understanding [7]. An example for the air travel information system (ATIS) task can be seen in Fig 1, therefore concepts are semantic and labels are word classes. All abbreviations used in the ATIS task are described in Sec. 6. In previous work [6] we

Sentence:	Flight	four	sixteen	departs	Dallas	at	9	:	10	A.M.	Correct	?	
Concepts:	FN	FN	FN	OR	OR	TD	TD	TD	TD	TD	DU		DU
Labeling :	IN	NU	NW2	FR	CI	AD	N10	IT	N60	IT	DU		DU

Fig. 1. Classification into word classes and concepts.

use extended-context-free grammars (ECFG) to build a weighted transition network. The ECFGs, which build the structure of the language, are constructed from an expert manually. In this paper we aim to learn automatically the semantic structure of the language from a corpus and to use in addition grammar rules (similar to ECFGs) for parts of the spoken language. Therefore a graphical representation, which allows rapid modeling, EM-training and decoding with the well known Viterbi-algorithm, is most suitable.

3. Graphical Models and Spoken Language Understanding

In this section we present Graphical Models, give a common notation, and explain, how they can be adopted to language understanding.

3.1. Dynamic Bayesian Network Model

Graphical Models (GMs) [8] are a combination of probability and graph theory, providing a visual graphical language and efficient algorithms for probability calculations and decision making. A Bayesian network (BN) is one type of GM where the graphs are directed and acyclic. The joint probability distribution (called directed factorization property [9]) over $\vec{X}$ is factorized as $p(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} p(x_i|parents(x_i))$. Dynamic Bayesian Networks (DBNs) are a generalization of BNs, they are used to describe time series: One BN represents one time slice. Additionally dashed edges describe the dependencies between subsequent time slices. For a given observation O with length T the DBN is unrolled: the time slices are repeated (T-2) times and connected through their inter-edges. They have been used for language understanding in [10]. In contrast, we use "Switching Parents" in order to integrate syntactical rules to the statistical approach.

3.2. Switching Parents

Normally a variable has only one set of parents. In Figure 2 variable S selects one concept out of $\{C_1, \ldots, C_N\}$, therefor the concept is the only parent of the labels L.

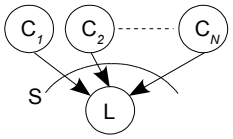

Fig. 2. Switching parents in Graphical Models.

In this paper we successfully use the switching parents to integrate the grammar based rules approach described in [6]. Furthermore, in state-of-the-art corpora not all combinations , e. g.in "time", "weekday", "date", and "numeric symbols" are available. We implement these concepts successfully with switching parents.

4. Language Understanding Models

The GM used in this work to model the semantic interpretation of spoken language consists of three nodes in each time slice. Thus the problem is modeled with three variables for each word of a sentence. We introduce four different setups to model different relations among these variables. Thus while the nodes of the models remain the same in all three GMs, the arcs between them are different. Therefore each GM describes a different factorization of the problem. These setups will be compared to a baseline system, in which the maximum likelihood of the concepts and labels of a word is decoded. In order to incorporate the concepts and labels we begin with the simplest model, setup 1 (Fig. 3). Note, that this setup and the following GMs represent factorial Hidden Markov Models (HMMs). In setup 1 we modeled the dependence on the concepts. Having observed that concepts do not only depend on the previous concepts but also on the previous labels (e. g. *from*—fromloc—origin, *Dallas*—city—origin) we extended this model to setup 2. Setup 3 has been further improved to include the dependence on the previously observed word. Finally, in setup 4 grammar rules were included by the GM concept "switching parents".

5. Experiments

In this section we describe the joint probabilities of the four setups. We assume a corpus with N_W different words W, words with the same meaning are grouped into N_L word classes (labels) L. One can combine more than one word class into (N_C different) concepts C. In the following the nodes of the models and the underlying probabilities are explained, whereby $\vec{x}, \vec{y}, \vec{z}$ describe the observables

$$P(C = \vec{x}) = \sum_{i=1}^{N_C} c_i \cdot \delta(\vec{x} - \vec{\mu_i}) \quad \text{with} \sum_{i=1}^{N_C} c_i = 1,$$

$$P(L = \vec{y}) = \sum_{j=1}^{N_L} l_j \cdot \delta(\vec{y} - \vec{\nu_j}) \quad \text{with} \sum_{j=1}^{N_L} l_j = 1,$$

$$P(W = \vec{z}) = \sum_{j=1}^{N_W} w_j \cdot \delta(\vec{z} - \vec{\omega_j}) \quad \text{with} \sum_{j=1}^{N_W} w_j = 1.$$

5.1. Setup 1

A schematic drawing of the GM is shown in Fig. 3. Each column represents one time slice. The top node C_t models the underlying concept of the current word. This concept is not observed and therefore modeled as hidden. In the first time slice $t = 1$ the concept does not depend on any other variable and is given by the probability $p(c_1) = 1$, which is the initial concept distribution. In all following time slices the concept is only dependent on the concept from previous time slices: $p(c_t|c_{t-1})$. Thus, in this GM the sequence of concepts is represented by a first order markov chain. The node for each searched label L_t is in all time slices only conditioned by the concept C_t of the current word W_t and is not observed and therefore hidden. As there is no interaction between the labels among subsequent time slices, this model represents a label which is drawn independently of the previous label but dependent on the current concept C_t given by the probability $p(L_t|C_t)$. Finally, the words of the sequence are known and modeled as observed nodes W_t. Any current word depends on both the label and the concept of the current time slice. Again subsequent words have no direct interaction but are connected through the concept markov chain. That is, with known concept C_t subsequent words W_t and W_{t-1} are independent. In this model the probability of a word is thus expressed as $p(W_t|L_t, C_t)$, which is similar to a factorized HMM. Altogether the GM in setup 1 factorizes the joint probability of the sequence of words, labels, and concepts as

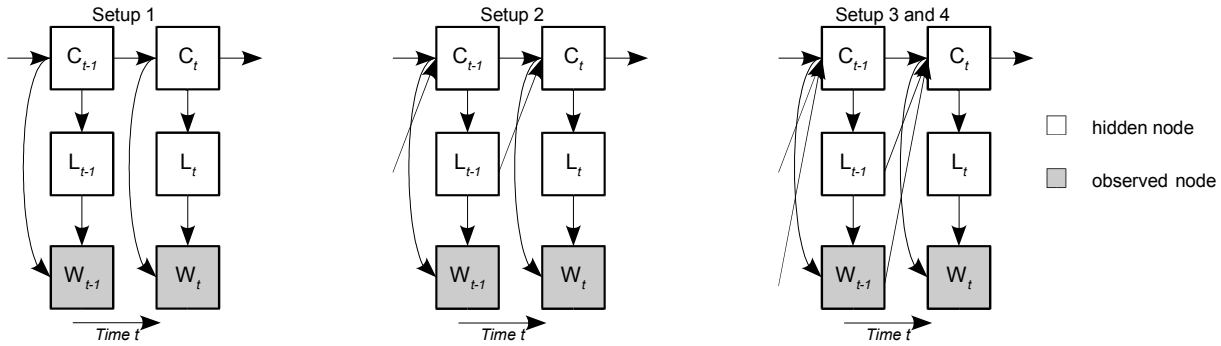

Fig. 3. *Used Graphical Model setup in GMTK*

$$p(V_1) = p(C_1) \cdot p(L_1|C_1) \cdot p(W_1|C_1, L_1) \cdot \cdot \prod_{t=2}^{T} p(C_t|C_{t-1}) \cdot p(L_t|C_t) \cdot p(W_t|C_t, L_t).$$

5.2. Setup 2

In addition to setup 1, setup 2 contains also the dependencies on the previous label L_{t-1}. For example, both labels "toloc" and "city" belong to the concept "destination". With the knowledge of the previous label, e. g. "toloc", the determination of the next label in the "destination" concept, e. g. "city" is much more robust. In this specific case the label "city" is determined to belong to the concept "destination". The top node C_t models the underlying concept of the current word. In contrast to setup 1, after the first time slice the concept distribution is conditioned by the label and concept of the previous time slice: $p(C_t|C_{t-1}, L_{t-1})$. In this GM the concept C_t benefits from the knowledge of the previous labels C_{t-1} and L_{t-1}. The further nodes, conditions, and the corresponding probabilities are the same as described in setup 1. Overall the GM in setup 2 factorizes the joint probability of the sequence of words, labels, and concepts as

$$p(V_2) = p(C_1) \cdot p(L_1|C_1) \cdot p(W_1|C_1, L_1) \cdot \cdot \prod_{t=2}^{T} p(C_t|C_{t-1}, L_{t-1}) \cdot p(L_t|C_t) \cdot p(W_t|C_t, L_t).$$

5.3. Setup 3

As an extension to setup 2, in this setup not only the previous label but also previous word contribute to the determination of the concept. For example, the words in a flight number "four", "sixty_five" and "hundred" are categorized into certain labels ("number word 1","number word 2","number 10","number 60", "number rest"). With the knowledge of the previous words and labels the determination of the next label in the concept e. g. "flight number" is improved compared to setup 2. In this GM the concept C_t benefits from the knowledge of the previous labels C_{t-1}, L_{t-1} and W_{t-1}, which is expressed by the transition probability $p(C_t|C_{t-1}, L_{t-1}, W_{t-1})$. Altogether the GM in setup 3 factorizes the joint probability of the sequence of words, labels, and concepts as

$$p(V_3) = p(C_1) \cdot p(L_1|C_1) \cdot p(W_1|C_1, L_1) \cdot \cdot \prod_{t=2}^{T} p(C_t|C_{t-1}, L_{t-1}, W_{t-1}) \cdot p(L_t|C_t) \cdot p(W_t|C_t, L_t).$$

5.4. Setup 4

This setup is graphical not distinguishable from setup 3, therefore the same factorization of the composite probability is used. However, the problem of non existing words in the corpus, e. g. certain dates, times, and numbers was solved by using grammar rules (switching parents) in this setup. For example, the city in concept "origin" is frequently derived from the word class "fromloc". Hence the following equation was implemented with switching parents.

$$p(C_t = \text{"origin"}|L_{t-1} = \text{"fromloc"}) = 1.$$

5.5. Parameters and Classifications

Each semantic meaning (M) in natural spoken language utterances can now be described by the GM setups $x = \{1, 2, 3, 4\}$ with the parameters

$$M_x = \{W_x, L_x, C_x\}.$$

The model parameters M_x are learned for each of the N_C concepts classes and N_L word classes with the EM-algorithm during the training phase. During the classification of an unknown utterance with given words W the model parameters M_x can be estimated for each model $P(L_x, C_x|W)$ with the highest likelihood

$$P_x = \underset{L_x, C_x}{\operatorname{argmax}} P(C_x, L_x|W_T) = \underset{L,C}{\operatorname{argmax}} \frac{P(V_x)}{P(W_T)} \quad \text{with } P(W_T) = \prod_{t=1}^{T} p(w_t).$$

Applying the Viterbi-algorithm to each natural spoken utterance in the different setups, leads to a different semantic segmentation of them.

6. Results

In order to show the performance of our systems we use the ATIS-0 corpus for evaluation [11]. In this database there are 840 naturally spoken user utterances, e. g. "Please, find the cheapest flight from Atlanta to Dallas on Thursday." To reduce the complexity we connected abbreviations, and complex city names with an underscore character. We also replaced punctuation marks by labels. The database contains a class of 9185 words W. The cognition of semantic meanings demands for every word a categorization into a word class and a concept class. Thereof 4284 were classified as not relevant and became the concept "dummy". The remaining 4901 words became one of the eleven non-dummy concepts, e g. "origin", "destination", "price", "flight number", and "airline". Every word is assigned to one word class. In this paper, we distinguish between 26 word classes. 25 word classes contain thereby relevant information and in addition the word class "dummy" contains all semantic irrelevant words. Every word in the ATIS corpus is hierarchically labeled by a word class and a concept. Each setup was evaluated by a 10-fold cross-validation with 90% training and 10% test sentences. Altogether there are 10 test-cases, whereby each sentence was selected randomly. The two-by-two contingency classify non-dummy and dummy words to relevant concepts/word classes, where the counts are for the well-known F1 measurement method.

	F-1 measure			recognition rate		
	C	*L*	Ø	*C*	*L*	Ø
Setup 1 [%]	90.8	93.5	92.2	90.7	93.2	92.0
Setup 2 [%]	93.2	94.0	93.6	93.0	93.8	93.4
Setup 3 [%]	93.7	94.3	94.0	93.5	94.1	93.8
Setup 4 [%]	94.0	95.1	94.6	93.7	94.8	94.3

Table 1. 10-fold cross-validation of all setups

The four setups in the GM were compared to a baseline system, which extracts the maximum likelihoods of word classes or concepts by given words $W_{L|C}$. This results in a baseline recognition rate of 91.7% for word classes and 86.6% for concepts. Tab. 2 shows that the simplest setup 1 as well as setup 4 outperfom the baseline system.

	Max.	Setup 1	Setup 4
Concept [%]	86.6	90.8	93.7
Label [%]	91.7	93.5	95.1

Table 2. F1-measure setup 1, 4 compared to a baseline.

7. Conclusions

In this paper four different setups of GMs for natural language understanding were presented. In all setups the probabilities were estimated by the language model in the ATIS corpus. The concepts and the word classes of the GMs were compared to the annotated labels and concepts in the ATIS corpus with maximum likelihood. The GMs show a significantly higher recognition performance than the baseline approach. Compared to maximum likelihoods the best setup has a relative error reduction of 3.7% in word classes and 8.2% in concepts. With the setup 4 we reach our bests results and outperform the dbn-based multi-level stochastic spoken language understanding system[10]. In the future we plan to enter recognition rates from our one-stage decoder [12] and thus superiorly utilize the potential of graphical models.

8. References

[1] E. Levin and R. Pieraccini, "Concept-based spontaneous speech understanding system," in *Proc. of ESCA Eurospeech*, 1995.

[2] R. Schwartz et al, "Hidden understanding models for statistical sentence understanding," in *Proc. of IEEE ICASSP*, 1997.

[3] C. Raymond et al, "On the use of finite state transducers for semantic interpretation," in *Speech Communication*, 2006, vol. 48, pp. 3–4288–304.

[4] F. Pla et al, "Language understanding using two-level stochastic models with pos and semantic units," in *LNCS series*, 2001, vol. 2166, pp. 403–409.

[5] Y. He and S. Young, "Semantic Processing Using the Hidden Vector State Model," *Computer Speech and Language*, vol. 19, no. 4, pp. 85–106, 2005.

[6] S. Schwärzler, J. Schenk, F. Wallhoff, and G. Ruske, "Natural Language Understanding By Combining Statistical Methods And Extended Context-free Grammars," in *Inproc. of 30th DAGM Symposium*, Gerhard Rigoll, Ed., Heidelberg, Germany, 2008, LNCS 5096, pp. 254 – 263, Springer.

[7] C. Raymond and G. Riccardi, "Generative and discriminative algorithms for spoken language understanding," in *Proc. of the ISCA Interspeech*, Antwerp, Belgium, 2007.

[8] J.A. Bitmes and C. Bartels, *Graphical Model Architectures for Speech Recognition*, vol. 22 of *5*, IEEE Signal Processing Society, NY, USA, Sept. 2005.

[9] S. L. Lauritzen, "Graphical models," New York, Oxford, 1996.

[10] Fabrice Lefvre, "A DBN-Based multi-level stochastic spoken language understanding system," in *Proceeding of the IEEE Spoken Language Technology Workshop*, Palm Beach, Aruba, 12 2006, IEEE.

[11] C. T. Hemphill and G. R. Doddington J. J. Godfrey, "The atis spoken language systems pilot corpus," http://www.ldc.upenn.edu/Catalog/docs/LDC93S4B/corpus.html.

[12] M. Thomae, T. Fabian, R. Lieb, and G. Ruske, "A One-Stage Decoder for Interpretation of Natural Speech," in *Proc. NLP-KE'03*, Beijing, China, 2003.