



# An ATC instruction processing-based trajectory prediction algorithm designing

Xuan Wang<sup>1</sup> · Yi Mao<sup>1</sup> · Xiaoyong Wu<sup>2</sup> · Qucheng Xu<sup>1</sup> · Weiyu Jiang<sup>1</sup> · Suwan Yin<sup>3</sup>

Received: 5 October 2020 / Accepted: 5 January 2021 / Published online: 30 January 2021  
© The Author(s), under exclusive licence to Springer-Verlag London Ltd. part of Springer Nature 2021

## Abstract

The radiotelephony communication is a voice communication mode between air traffic service unit and aircraft currently. The control instruction is a kind of unstructured data, so that the automatic systems cannot use understand its semantic. If control instruction is regarded as a sort of special “natural language,” methods such as syntax analysis and semantic analysis can be adopted to generate the structured instruction. The correct recognition of the language must be important for the control instruction. However, the control instruction in Chinese is different from the general use of Chinese language in form, resulting in prepositions becoming important for semantic analysis. This paper proposes a deep neural network-based Chinese language control construction algorithm for the trajectory prediction. In particular, analysis of semantic characteristics of control instruction is realized by using cognitive linguistics theory and construction grammar theory. The control instruction is then designed by the semantic ontology. Based on the deep neural networks by considering the word sequence of instruction as the inputs. The test results have demonstrated the effectiveness of the proposed algorithm with a developed entity extracting model. (The results are quantified using the BiLSTM-LAN-CRF in detail.)

**Keywords** ATC instructions · Restrictive factor of circumposition · Construction grammar theory · Semantic ontology · BiLSTM-LAN-CRF

## 1 Introduction

The radiotelephony communication is a voice communication mode between air traffic service unit and aircraft. The correct use of radiotelephony communication is crucial to the safe and efficient operation of aircraft. There are numerous cases of aviation unsafe incidents, even flight accidents, caused by irregular radiotelephony communication. Due to the differences in language, accent, semantic expression and modes of understanding among staff working in land and air, and factors such as work intensity, mental stress and emotions, misunderstandings in radiotelephony communication occur from time to time. In

actual ATC work, a tiny mistake may cause fatal accident. For example, the flight accident may occur when there are conflicts between two adjacent instructions. As such, structured processing of control instructions and generation of representations comprehensible to the system will help the system automatically judge whether there will be a potential conflict between two control instructions, which is of great significance to the safety of civil air transportation.

It is important to understand the ATC instructions, in [1], the ontology is proposed for ATC instructions understanding, and the instructions are replaced to the word sequence by ten corresponding class labels. These labels are proposed by Nguyen and Holone [2, 3], where positions (above, below, etc.) are also one class label. English ATC instructions always use position between verbs and place words, but in Chinese ATC instructions, the position may disappear sometimes. This means the semantic relation between verbs and place words depends on not only position, but also internal relevance. This paper uses construction grammar (CG) theory to explain this phenomenon and propose the method to analyze the semantic of ATC instructions.

✉ Yi Mao  
mao\_y@nuaa.edu.cn

<sup>1</sup> State Key Laboratory of Air Traffic Management System and Technology, Nanjing 210007, China

<sup>2</sup> JiangSu ATM Sub-BUREAU, Nanjing 210000, China

<sup>3</sup> Department of Civil and Environmental Engineering, Imperial College London, London SW7 2AZ, UK

Usually, structured instruction can be obtained through dependency parsing and semantic analysis. However, the control instruction cannot satisfy the dependency grammar (DG) theory strictly [4], which could reduce the accuracy of dependency parser. The reason is that some control terminology does not depend on any verbs, or other words, such as XX tower, XX approach. As shown in Fig. 1, in the Chinese construction “nan fang liu liu yao si, qing dao ta tai, di mian feng yao dong, 2 mi miao, pao dao yao guai, ke yi qi fei,” which is “CSN6614, Qingdao Tower, surface wind 10, 2 m/s, Runway 17, take-off” in English, the phrases “Qingdao Tower (qing dao ta tai)” and “surface wind 10, 2 m/s (di mian feng yao dong, 2 mi miao)” do not depend on the predicate “take-off (ke yi qi fei)” directly.

In addition, the restrictive ability of some words in control instruction would decrease the accuracy of parsing further. Therefore, it is difficult to use the dependency grammar theory to analyze the syntax of control instruction.

According to the construction grammar (CG) theory, the construction structure in the sentence will affect the semantic expression, that is, the construction is used to suppress the ambiguity of words [5]. In Chinese, the circumposition has the function of semantic restriction, which can be regarded as a construction structure to disambiguate [6]. Moreover, the verb–object structure should also be analyzed when multiple verbs appear in the control instruction simultaneously. Therefore, the syntax analysis of control instruction can be transformed into the analysis of construction structure.

The next step is semantic analysis. The essence of semantic analysis is to find the semantic relations between the entity and verb in control instruction. Semantic relation has different names in different grammar theories, for example, case in case grammar theory [7] and valence of verbs in valence grammar theory [8].

The algorithm of structural processing contains two steps: (1) extract entities and constructions; (2) analyze the relations between entities and verbs, and then generate a structural form. Entity and structure extraction of control instruction is similar to entity extraction task in natural language processing, both essence is to obtain the entities,

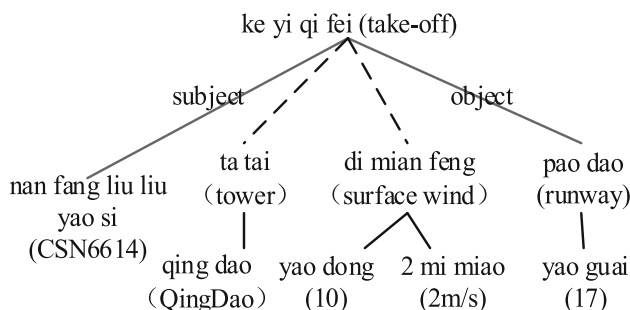


Fig. 1 Grammatical structure of control instruction example

and construction can be regarded as a kind of entity. Semantic analysis aims to find the “entity, relation, entity” tuple as shown in Fig. 2.

Entity extraction is a kind of sequence labeling task. Hidden Markov model (HMM) and conditional random field (CRF) [9] are two advanced statistical models for this task. However, they cannot capture the long-range dependencies information due to the limitation of Markov assumption.

Long short-term memory network (LSTM) can break through the limitation of Markov assumption, and it can catch the long-range dependencies information theoretically. Therefore, LSTM [10] and BiLSTM [11, 12] work better in the sequence labeling task. Zhang and Yang [13] proposed a lattice LSTM model for Chinese named entity recognition task. CNN can be used directly for sequence labeling [14] and is often applied to the character embedding [15]. Chiu et al. [16] use both BiLSTM and CharCNN for pos tagging.

The essence of LSTM is to build a unidirectional language model, while BiLSTM builds the forward language model:

$$p(x) = \prod_{t=1}^T p(x_t|x_{<t}) \tag{1}$$

and backward language model, respectively:

$$p(x) = \prod_{t=1}^T p(x_t|x_{>t}) \tag{2}$$

where  $p(x)$  denotes the probability of the given sentence  $x_t, t \in [1, T]$ .  $x_{<t}$  means the left context of  $x_t$ , while  $x_{>t}$  means the right context of  $x_t$ .

The forward language model only uses the left context to predict the target word, and the backward language model, on the other hand, uses the right context to predict the target word. Huang et al. [17] proposed the BiLSTM-CRF for sequence labeling. This model superimposed CRF layer on the basis of BiLSTM. The CRF can obtain the globally optimal sequence labeling through Viterbi decoding, that is, the entire sentence information is used to predict the sequence labeling. Furthermore, in regard to the sequence labeling tasks such as part-of-speech tagging, named entity recognition (NER), Huang et al. compared the performance of CRF, LSTM, BiLSTM and BiLSTM-CRF in these tasks and then found that BiLSTM-CRF on

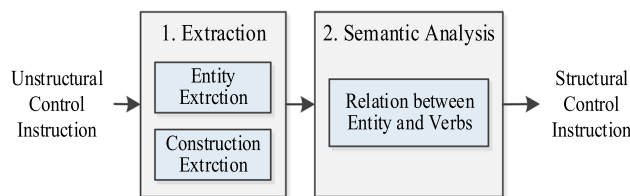


Fig. 2 The steps in algorithm of structural processing

the sequence labeling task is the best. Furthermore, Lample et al. [18] chose both character embedding and word embedding as input feature of BiLSTM-CRF. On this basis, Ma and Hovy [19] added CNN to the model and proposed the BiLSTM-CNNs-CRF model, which used CNN to process the character embedding.

Self-attention [20] gets better representational ability of sequence than LSTM by catching the long-range dependency information. Cao et al. [21] proposed to introduce self-attention mechanism into the neural network model to handle the Chinese named entity recognition. However, due to limitation of Markov assumption, CRF cannot catch the long-range dependency information for word representation, so it has limit when dealing with sequence tagging tasks for long sentences. Cui et al. [22] replaced CRF with label attention network (LAN) and proposed BiLSTM-LAN.

Semantic role labeling (SRL) is a basic NLP technology for event extraction. It aims to find the semantic role between arguments in sentence and event triggers. The semantic role contains agent, object, action, time, place and so on, which are similar to the case in case grammar theory.

For semantic role labeling, Chen et al. [23] proposed a CNN-based method to catch important information of sentence by dynamic multi-pooling. However, when multiple events take place in the sentence, the method performance will be degraded. Nguyen et al. [24] proposed an encoder–decoder model to extract semantic roles. On the basis of BiLSTM, He et al. [25] added the decoding algorithm with  $A^*$  constraint and purposes an end-to-end deep model. In addition, self-attention can also be used for semantic role labeling [26, 27].

Ontology model is also useful in semantic analysis, such as Semantic Web [28]. Ontology contains five basic modeling elements: class, relation, function, axiom and instance. Among them, class also names concept, and relation refers to interrelation between the concepts. Therefore, elements in class and relation can be used to describe semantic relation between predicate verb and entity.

This paper proposes a novel deep neural network-based algorithm for control instructions, which help automatic systems to predict the trajectory only from the control instruction. The innovations are as follows.

1. This paper analyzes the linguistic forms of control instruction by employing the theories of cognitive linguistics and construction grammar. The semantic features of control instruction are thus found, and the syntactic analysis is transformed into the extraction of constructions;
2. On the basis of that, this paper proposes a new deep neural network-based algorithm named BiLSTM-LAN-CRF to extract the entities of instruction;

3. The semantic relation between the entity and the verb could be represented by the semantic case. And the case grammar could be used to design the semantic ontology and conduct the semantic analysis. This process could be represented as “entity, semantic case, verb.”

## 2 Semantic analysis

### 2.1 Linguistic structure

A control instruction is of great significance for aviation safety, the rules of which are not the same as the Chinese language in daily usage. As a semi-artificial language, a control instruction has strict standards for radiotelephony communication, as set by the International Civil Aviation Organization (ICAO). An ATC controller uses concise, rigorous and unambiguous instructions to command an aircraft [29]. Thus, for understanding the semantics of control instructions, certain rules such as:

Adverse call letters + Own call letters + Content

or

Adverse call letters + Content

are helpful.

In the controller–pilot communication, the first part of the control instruction is always adverse call letters (air traffic controller/pilot), which can be regarded as the subject of the instruction. The control instruction also contains some terminology, such as surface wind and dew point which are often in the form:

Term + Number + Quantifier

It is difficult to generate the structural form only based on these rules because they cannot support the relation between entities and verbs.

The accuracy of the dependency parser is found to reduce when analyzing the Chinese ATC instructions. One important reason for this is that some words can be restrictive, which means that these words, such as a preposition, in a control instruction, can disappear sometimes. For example, in the Chinese instruction “dong fang san jiu ba si C tuo li,” which is “CES3984, vacate via C” in English, the preposition “via (jing you)” does not appear. The disappearance of this word also reduces the accuracy of the dependency parser. Furthermore, some verbs in the control instruction also have this ability of disappearance sometimes, and this makes parsing more difficult.

Although the accuracy of parsing would reduce because of the restrictive factor of circumposition, it also supports the theoretical basis of semantic analysis. In Chinese, a preposition does not always appear in a sentence when it could be expected to appear. This does not influence the whole semantic and, thus, the control instruction. However, a preposition is also retained in the instruction sometimes. For example, in the Chinese instruction “dong fangsan jiu ba si dao ting ji wei liang si,” which is “CES3984, to stand 24” in English, the preposition “to (dao)” is retained.

Consider a prepositional phrase as a circumposition construction. It exists in the control instruction because: (1) of the habit of the controllers; (2) it disambiguates the sentence. Based on the construction grammar theory, construction has the ability of disambiguation, and so does the circumposition. The circumposition construction in Chinese usually consists of three parts: preposition, content and postposition. The preposition can disappear conditionally. For example, the preposition “to (dao)” can disappear if it follows the verb, which describes the action of movement. Prepositions in a control instruction also satisfy this principle. For example, in the Chinese instruction “shang shen (dao) ba bai bao chi,” which is “climb to 800 and maintain” in English, the preposition “to (dao)” disappears. In this case, the difference from general Chinese language usage is that the preposition “to (dao)” can also disappear when it does not follow any verbs. For example, in the Chinese instruction “ma shang (dao) deng dai dian le,” which is “to the holding point at once” in English, the preposition “to (dao)” can also disappear and this indicates that the ability of preposition disappearance in the control instruction becomes stronger.

If a preposition exists in a control instruction, it indicates the presence of multiple semantic relations between the entity and the verb. Therefore, it is necessary to use the circumposition construction to disambiguate and determine the correct relation. The next step is to explain why the possibility of disappearance of a preposition in a control instruction is stronger.

The cognitive linguistic theory focuses on the link between the linguistic structure and human cognition [30], which uses “motivation” to explain this link. One definition of “motivation” is “non-arbitrary,” which means that the relationship between the linguistic structure and the semantic is not arbitrary [31]. Another definition is “interpretability” [32], which means that if and only if there is a particular connection,  $L$ , between  $A$  and  $B$  independently, and if  $L$  can explain the relation between  $A$  and  $B$ , it can be considered that  $A$  and  $B$  are in motivation.

Based on “motivation,” the special linguistic structure of a control instruction reveals its semantic feature. Therefore, finding out why preposition disappearance occur can help in choosing the method of semantic

analysis. In order to be unambiguous, the ATC controllers need to describe the semantics clearly. In the natural language, the word order and the circumposition construction influence the semantics of a sentence. Thus, both of these factors imply that the relation between a large number of entities and a verb in a control instruction is unique, and thus, the control instruction is always unambiguous. For example, the unique relation between the phrases “take-off” and “runway 17” calls the case “locative,” which generates the “runway 17, locative, take-off” tuple.

In a control instruction, verbs can also disappear sometimes. For example, in the Chinese instruction “dong fang san jiu ba si, liang si hao deng dai dian (deng dai) le,” which is “CES3984, wait at holding point 24” in English, the verb “wait (deng dai)” disappears. This also implies that the relation between some verbs and certain entities is unique and thus, if the verb disappears in the instruction, it can also be reasoned by certain entities while retaining the complete semantics of the sentence.

Furthermore, some terminology has a flexible order, which means that the positions of some terms or phrases are unrestricted. There are a lot of words in a control instruction that depend on verbs such as flight call, height, direction and runway. On the other hand, some terms such as surface wind do not depend on any verbs and their order is usually not fixed. A flexible word order implies that this terminology cannot influence the semantic relation between an entity and the verb in the sentence. However, it also indicates that the label of any word in a control instruction does not depend on the long-range label. This characteristic will influence the performance of deep neural networks used for entity extraction.

In summary, the relation between an entity and a verb is always unique in control instructions, and a circumposition construction can be applied to disambiguate multiple relations. Based on this point, the entities and constructions can be extracted first, followed by an analysis of the semantic relation between the entity and the verb as the second step.

## 2.2 Structural form

The design of the structural form is based on the case grammar theory. This theory focuses on the semantics of a sentence, in particular, on the relation between linguistic signs and objects. It assumes that a sentence consists of modality and proposition, where modality includes the tense and voice of the sentence, while proposition refers to the relation between a verb and other words in the sentence. In this work, a semantic relation is defined as a case. A case is a kind of a fixed deep semantic relation between nouns (entities) and verbs.

In Chinese, there are three levels in the case framework [33]. The top level consists of a “role” and a “scene,” and in the second level, the “role” includes a “subject,” “object,” “adjacent” and “copula,” while the “scene” includes a “dependent,” “environment” and “reason.” There are 22 cases in the third level, which belong to the seven factors in the second level. One basic principle of case grammar is that although some sentences have different surface syntactic structures, their case framework is unique if they have the same predicate verb and also the same cases.

However, not all the cases are important in control instruction. In practice, the most important part of a control instruction is the words which describe the trajectory of the aircraft. Therefore, a part of the cases which describe the trajectory can be used to design the structural form of the control instruction.

As shown in Fig. 3, ten cases are chosen to describe the semantic relation in a control instruction. In addition, the environment class includes six cases: range, time, locative, direction, source and goal, which are used to describe the trajectory of an aircraft.

The structural form is defined as an “entity, case, verb” tuple, where the case denotes the relation between the entity and the verb. Therefore, the control instruction can be represented by one or more tuples.

In Chinese control instruction, different verbs have different types and numbers of cases. The semantic ontology can support these semantic relations and generate structural instruction.

### 2.3 Semantic ontology

The semantic ontology is built to support the relations between an entity and a verb. There are two kinds of

entities, those that can describe the trajectory of an aircraft, such as flight call, runway, taxiway, holding point and height, which are related to verbs directly. The other kind is the entities that are not related to verbs directly and include some terms such as surface wind and temperature. A preposition is a kind of an important functional word, which can be defined in an independent class.

The class of prepositions contains: “from (cong),” “to (dao),” “toward (xiang),” “along (yan)” and so on. Some other prepositions, such as “to (zhi)” and “to (wang)” are as the same as “to (dao).”

There are two types of verbs in a control instruction: control verbs and auxiliary verbs. The control verbs consist of landing, taxiing, leaving, take-off and so on, which describe the action of the aircraft. The auxiliary verbs consist of contact, receive and so on, which are used only for communication and not for describing the action of the aircraft.

There are five classes in semantic ontology: control entities, other entities, the preposition, control verbs and auxiliary verbs. The semantic relation contains ten different cases as given in Table 1. The semantic ontology can support the cases between an entity and a verb of a control instruction and can then generate the structural form of the Chinese control instruction as post in Table 2.

As shown in Fig. 4, the inputs of the semantic ontology are the label of the entity, circumposition construction and verbs, where the label of the entity is an element in the class of entities. The output is tuples that can constitute the structural instruction. However, the input must be obtained by extracting the entity, verb and construction from the target control instruction. A description on how to build a deep neural network to extract the entity and construction has been given in the next section.

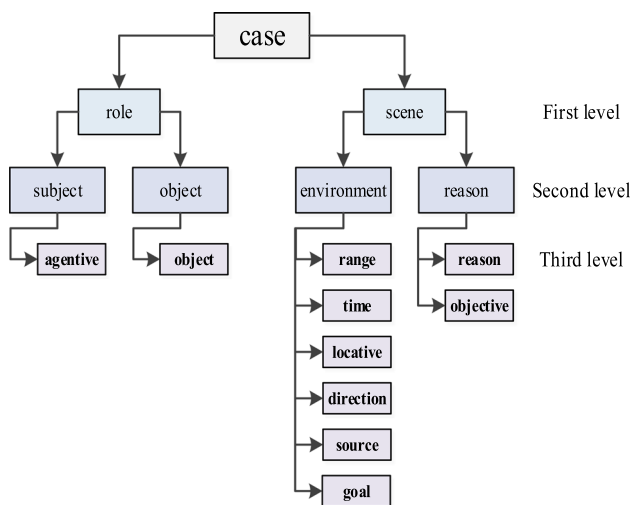


Fig. 3 Cases for designing the structural instruction

Table 1 Definition of cases for control instructions

Case	Definition
Agentive	The initiator of actions
Object	The state of actions
Range	The limits of space and states after aircraft acting
Time	When do actions operate
Locative	Where do actions operate
Direction	The general line that aircraft moves after acting
Source	Location when aircraft start actions
Target	Location when aircraft finish actions
Reason	The conditions of aircraft’s actions
Objective	The target of aircraft’s actions

**Table 2** The preposition frameworks control instructions

Prep	Definition
from (cong)	Describe the case of “source,” it includes space source, time source, range source, direction source. The framework is defined as: from + location/time/direction
to (dao)	Describe the case of “goal,” it includes space goal, time goal, range goal, direction goal. The framework is defined as: to + location/direction/ time
toward (xiang)	Describe the case of “direction.” The framework is defined as: toward + direction
along (yan)	Describe the case of “locative.” The framework is defined as: along + location

### 3 Entity extraction

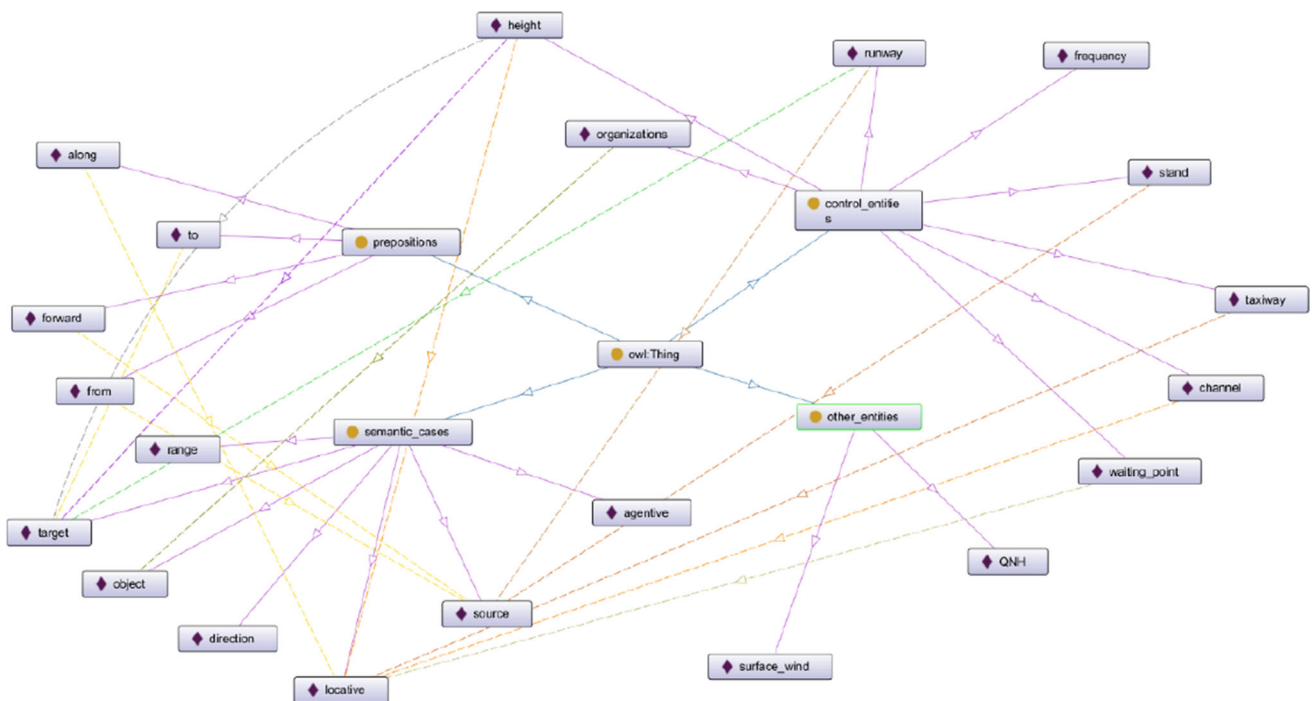
Entity extraction is also called named entity recognition (NER) task. It aims to capture the entities from the tag sequence of given sentence. In control instruction, tags contain flight call, runway, taxiway, holding point, height, action and so on, and the sentence is tagged in “BIO” format, where “B-” and “I-” tag indicate beginning and intermediate position of entities, and “O” indicates that word does not belong to any entity. For example, the control instruction of CDG471, Runway 17, take-off, goodbye is tagged as: B-FLY I-FLY I-FLY I-FLY I-FLY I-FLY I-FLY B-RW I-RW I-RW B-ACT O. In tag sequence, FLY denotes flight call, RW denotes the runway and ACT denotes verbs. Since prep words are important for semantic of ATC instructions, it is necessary to extract

prep words, so prep words are also defined as the entities for training the model.

This section introduces three popular neural networks for NER task: BiLSTM-Softmax, BiLSTM-CRF [14] and BiLSTM-LAN [19]. Moreover, it will introduce a new model named BiLSTM-LAN-CRF for NER task of the control instructions.

#### 3.1 BiLSTM-Softmax

Recurrent neural network (RNN) as shown in Fig. 5, can process the sequence modeling tasks such as language modeling, speech recognition, and named entity recognition. RNN can utilize the historical information in task. However, it cannot catch the long-range dependency information due to the defects of gradient vanish and



**Fig. 4** The semantic ontology (based on Protégé tool)

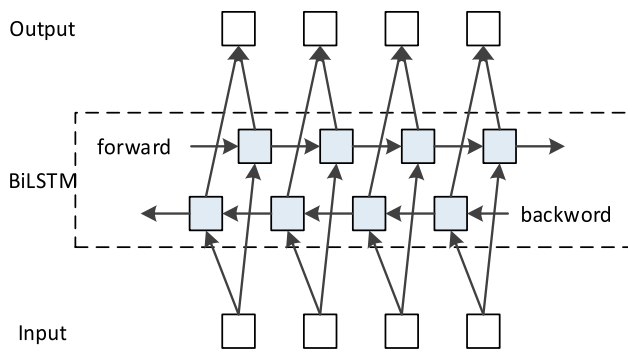


Fig. 5 The BiLSTM-Softmax model

gradient explosion, which limits its performance for the long sentence.

Long short-term memory (LSTM) can solve this problem by using memory cell. The bidirectional LSTM (BiLSTM) contains LSTMs in two different directions. It concatenates the hidden state of forward LSTM and backward LSTM to obtain the hidden state  $h = [\vec{h}, \overleftarrow{h}]$ .

Input  $h$  to the output layer, the output layer applies softmax function to normalize the hidden state, and then output the label sequence.

This model can be described in the mathematical form. Assume input  $x = x_1, x_2, \dots, x_T$ ; RNN calculates output  $y = y_1, y_2, \dots, y_T$  by:

$$\begin{aligned} h_t &= f(Ux_t + Wh_{t-1}) \\ y_t &= g(Vx_t) \end{aligned} \tag{3}$$

where  $U$ ,  $W$  and  $V$  are weight matrices, and  $h_t$  denotes hidden state in position  $t$ , which is computed by current input  $x_t$  and previous hidden state  $h_{t-1}$ .  $f(z)$  and  $g(z)$  are sigmoid function and softmax function:

$$\begin{aligned} f(z) &= \frac{1}{1 + e^{-z}} \\ g(z) &= \frac{e^{z_i}}{\sum_k e^{z_k}} \end{aligned} \tag{4}$$

As shown in Fig. 6, LSTM utilizes the memory cell in hidden layers, and every memory cell contains one or more cells, and three gates: forget gate, input gate and output gate. These gates help LSTM to remember more historical information. The hidden state  $h_t$  can be calculated by:

$$\begin{aligned} f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \\ i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\ c_t &= f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ h_t &= o_t \tanh(c_t) \end{aligned} \tag{5}$$

where  $\sigma()$  is the activation function, and  $f_t$ ,  $i_t$  and  $o_t$  are outputs of forget gate, input gate and output gate in

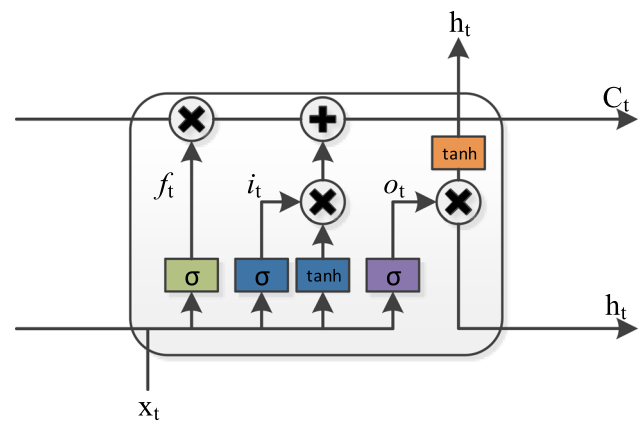


Fig. 6 The memory cell of LSTM

position  $t$ , and  $c_t$  is the state of cell. The hidden state  $h_t$  can be calculated by  $o_t$  and  $c_t$ .  $W_{xf}, W_{xi}, W_{xo}$  and  $W_{xc}$  are weight matrices, and  $\tanh(z) = 2f(2z) - 1$  is also activation function where  $f(z)$  is the sigmoid function.

### 3.2 BiLSTM-CRF

BiLSTM-Softmax infers the tags only from the current hidden state of BiLSTM, instead of the context label information. However, the tags in many sentences usually depends on its context labels, so the performance of BiLSTM-Softmax is limited. As shown in Fig. 7, CRF can utilize the whole sentence information to infer the optimal output sequence due to Viterbi decoding. Thus, BiLSTM-CRF can obtain the better performance.

The definition of CRF is as follows: Assume that when input sequence  $x = x_1, x_2, \dots, x_T$  is given, the output sequence  $y = y_1, y_2, \dots, y_T$  is predicted based on the conditional probability distribution  $p(y|x)$ :

$$p(y|x) = p(y_t|x_t, y_1, \dots, y_{t-1}, y_{t+1}, \dots, y_T), \quad t = 1, 2, \dots, T \tag{6}$$

with Markov assumption:

$$p(y_t|x_t, y_1, \dots, y_{t-1}, y_{t+1}, \dots, y_T) = p(y_t|x_t, y_{t-1}, y_{t+1}), \quad t = 1, 2, \dots, T \tag{7}$$

The output in each position depends only on information of its previous and next position explicitly due to Markov assumption. Therefore, it cannot be used for word representation. However, CRF can use Viterbi decoding to predict the output sequence, with the whole sequence information when it is used in the output layer, as shown in Fig. 8.

In BiLSTM-CRF, consider  $P$  to be the matrix of scores output by BiLSTM layer, and then define the score to be:

$$s(x, y) = \sum_{i=0}^T A_{y_i, y_{i+1}} + \sum_{i=1}^T P_{i, y_i} \tag{8}$$

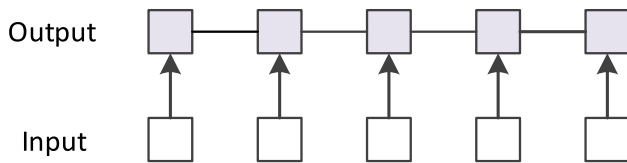


Fig. 7 The CRF model

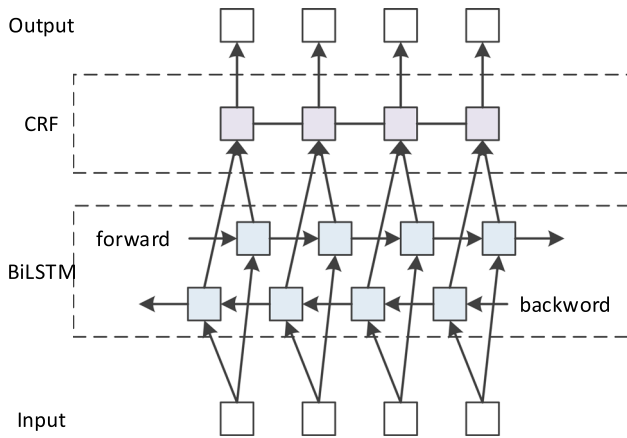


Fig. 8 The BiLSTM-CRF model

where  $A$  is a matrix of transition scores and  $A_{y_i, y_{i+1}}$  indicates the score of a transition from  $y_i$  to  $y_{i+1}$ .  $P_{i, y_i}$  denotes the score of output  $y_i$  of the representations of word  $x_i$  after BiLSTM layer.

Then, calculate all the possible tag sequences yielding a probability of  $y$  by softmax function:

$$P(y|x) = \frac{e^{s(x,y)}}{\sum_{\tilde{y} \in Y_x} e^{s(x,\tilde{y})}} \tag{9}$$

where  $Y_x$  denotes all possible output sequences for  $x$ . During training, it needs to maximize the log probability of the correct output sequence:

$$\log(p(y|x)) = s(x, y) - \log\left(\sum_{\tilde{y} \in Y_x} e^{s(x,\tilde{y})}\right) \tag{10}$$

While decoding, the output sequence is obtained by maximizing the score:

$$y^* = \arg \max_{\tilde{y} \in Y_x} s(x, \tilde{y}) \tag{11}$$

where  $y^*$  denotes the optimal sequence, which is computed by Viterbi decoding.

### 3.3 BiLSTM-LAN

There are some drawbacks of BiLSTM-CRF. Due to Markov assumption, this model cannot catch the long-range dependency information explicitly for word representation. CRF can be also computationally expensive

when larger number of labels exist in data by using Viterbi decoding. In BiLSTM-LAN, the label attention network (LAN) can catch the long-range dependency between the label sequence and input sequence. It can be utilized both in inference layer to output tag sequence, and in word representation layer to encode the input sequence as shown in Fig. 9.

In LAN layer, the self-attention mechanism uses the multi-head attention to capture multiple possible of the potential label distributions in parallel. The multi-head attention is based on scaled dot product attention.

The scaled dot product attention can output the weighted representations of input sequence. Assume query as  $Q$ , key as  $K$ , and value as  $V$ , and the expression is:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{12}$$

where  $\sqrt{d}$  is the scaled factor  $softmax(QK^T/\sqrt{d})$  is the attention weight matrix.

As shown in Fig. 10, Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions:

$$Multi-head(Q, K, V) = Concat(head_1, head_2, \dots, head_{num})W^O \tag{13}$$

where  $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$ , num is the number of heads, and  $W_i^Q \in R^{d \times \frac{d}{h}}$ ,  $W_i^K \in R^{d \times \frac{d}{h}}$ ,  $W_i^V \in R^{d \times \frac{d}{h}}$ ,  $W_i^O \in R^{d \times d}$  are parameter matrices.

BiLSTM-LAN is in encoder–decoder structure. In encoder, BiLSTM-LAN can represent the words of input sequence as word representation layer. However, in decoder, LAN is used to inference the output as inference layer.

Assume the input sequence  $x = x_1, x_2, \dots, x_T$ , obtain the representation  $H^B \in R^{n \times d}$  by BiLSTM, then define

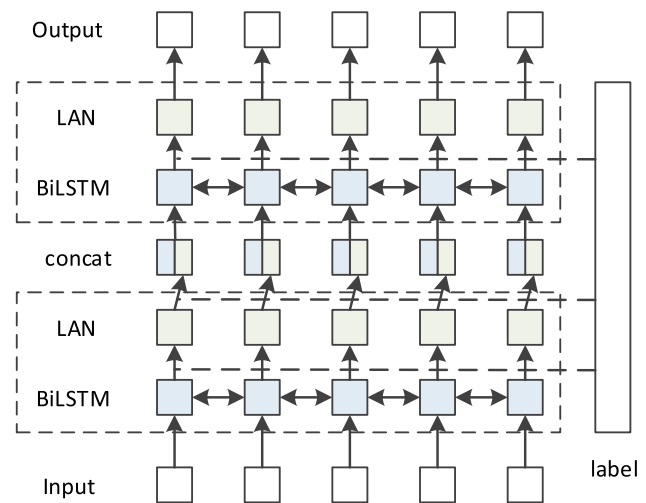


Fig. 9 The BiLSTM-LAN model



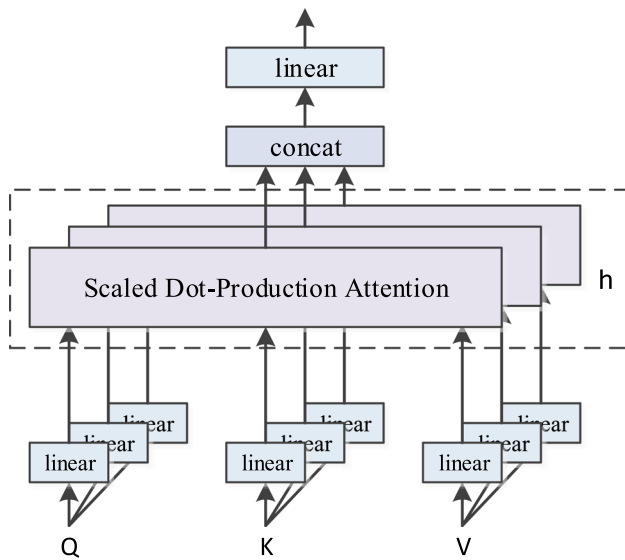


Fig. 10 The multi-head attention

$Q = H^B, K = V = x^l$  as the input of multi-head attention, where  $x^l \in R^{|L| \times d}$  is the label embedding and  $|L|$  is the number of labels. The output of multi-head attention is  $H^L = Multi-head(Q, K, V)$ . The encoder outputs  $H = [H^B, H^L]$ , then input  $H$  to the decoder and output the tag sequence:

$$y_i^j = \arg \max_j (y_i^1, y_i^2, \dots, y_i^{|L|}) \tag{14}$$

where  $i = 1, 2, \dots, T, j = 1, 2, \dots, |L|$ , and  $y_i^j$  denotes the tag of  $i$ th input word,  $i$  denotes the position of sequence, and  $j$  denotes the number of tags.

### 3.4 BiLSTM-LAN-CRF

BiLSTM-LAN-CRF is a kind of model for NER task of control instruction. It is built in encoder–decoder structure, with BiLSTM-LAN to be the encoder and BiLSTM-CRF to be the decoder. In encoder, the BiLSTM-LAN can catch the long-range dependency information explicitly to represent the input sequence, and in decoder, the BiLSTM-CRF can output the global optimal tag sequence.

As same as BiLSTM-LAN, consider the input sequence  $x = x_1, x_2, \dots, x_T$ ; the output of the encoder  $H = [H^B, H^L]$  can be regarded as the representation of  $x$  by concatenating the hidden state  $H^B$  of BiLSTM and output  $H^L$  of the multi-head attention.

However, the decoder is instead of BiLSTM-CRF; this is because the control instruction is a kind of short text. In addition, the flexible terminology orders in control instruction indicate that the long-range dependency between tag and input sentence is not strong. Therefore,

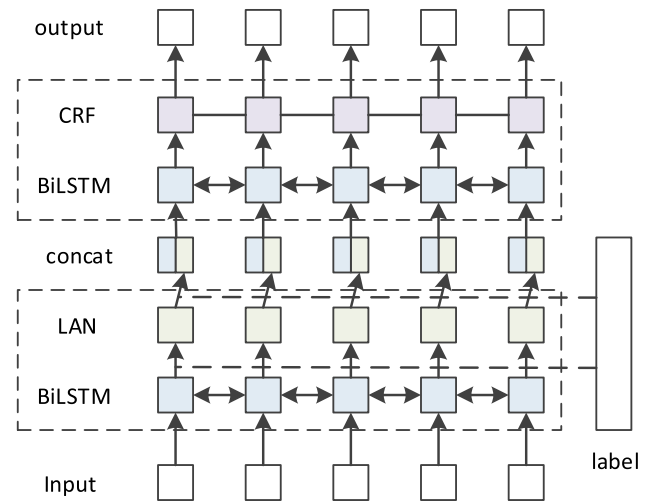


Fig. 11 The BiLSTM-LAN-CRF model

LAN loses its advantage for control instructions. However, BiLSTM-CRF can utilize the less parameters to obtain the similar performance in decoder.

Compared with original BiLSTM-CRF, after encoder, our model, as shown in Fig. 11, gets the better representation of input sequence to BiLSTM-CRF, which increases the performance of task. Compared with BiLSTM-LAN, our model can obtain the similar performance by using fewer parameters.

## 4 Algorithm

### 4.1 Verb-object construction

There are two steps for generating the structural instructions: (i) applying a deep neural network to extract the entities and verbs, (ii) using semantic ontology to obtain the semantic relation between the entity and the verb. A circumposition construction can be used to disambiguate multiple relations between a given entity and a verb.

The control instruction will become complex if it contains more than one verb. It is difficult to find out which verb is related to which entity by extracting the entity alone. However, it is necessary to extract the verb–object construction, since it can support the correct relation between an entity and a verb. A verb–object construction can disambiguate the surface relation between an entity and a verb. The constructions which need to be extracted also contain a verb-object construction.

## 4.2 Algorithm of structural processing

In summary, the new algorithm of structural processing of Chinese ATC instructions consists of the following steps:

---

Input: control instruction in unstructural form

Output: control instruction in structural form

1. Generate the word embedding;
  2. Use deep neural networks to extract the entities and verbs;
  3. Judge the number of verbs, if there is only one verb, get entity–verb pairs, then go to step 5, otherwise, continue;
  4. Extract the verb–object constructions, get entity–verb pairs based on the constructions;
  5. If there is any preposition in text, extract the prep constructions;
  6. Put entity–verb pairs, and prep constructions into the semantic ontology, output the structural form
- 

Firstly, the input instruction converts to word sequence after embedding, and then deep neural network is used to extract the entities, verbs and constructions. After this way, it extracts the constructions to disambiguate the surface and semantic relations between the entities and the verbs, where the verb-object construction can ensure the surface relation and the prepositional construction can ensure the semantic relation. Finally, it uses the semantic ontology to find the semantic case between verb and entity words and generate the structural instruction in triple form.

## 5 Empirical results

This section describes the empirical study carried out for testing our new algorithm of structural processing for control instruction. In the first step of extraction, results obtained from three models, namely BiLSTM-Softmax, BiLSTM-CRF and BiLSTM-LAN, have been compared with those obtained from our model, BiLSTM-LAN-CRF. The process of construction extraction is the same as entity extraction by these neural networks. Therefore, in the next subsection, a description of only the procedure of entity extraction, for a control instruction within the extraction step of the new algorithm, will be given. The computation was done using one NVIDIA GeForce GTX 1060 GPU to train the models.

### 5.1 Data preparation

The experimental data consist of 5000 control instructions, which include phrases corresponding to take-off, departure, approach and landing. A control instruction is a kind of a

short text message, where the longest instruction contains 41 words, while the shortest one contains 6 words among the set of 5000 instructions. An average instruction contains 15 words as shown in Fig. 12.

The maximum length of a sentence was defined to be 35. From the control instructions, 4500 instructions were chosen as training data and 500 instructions were selected as the testing data.

These data instructions contain 10 types of entities corresponding to an aircraft movement: flight call, frequency, action, location, runway/taxiway/ channel, holding point, weather, height, time and other. These were used as the labels to tag the data by “BIO” format.

### 5.2 Comparison

This subsection describes the results of the performance of BiLSTM-Softmax, BiLSTM-CRF and BiLSTM-LAN models. In LAN, the number of heads was set to 1, 2, 4, 8 separately, and the parameter  $d$  was set to 512. BiLSTM-LAN model was observed to give the best performance when the number of heads was 2. Before inputting the model, the instruction needs to be converted to word embedding by using Word2Vec algorithm.

Set 2 consisted of hidden layers in the BiLSTM model, with each hidden layer containing 256 neurons. In addition, the following parameter settings were used: The batch size was 50, epochs = 10, dropout = 0.5, cross-entropy loss function was chosen as the loss function, and an SGD optimizer was used with parameters  $\beta = 0.9$ , and learning rate,  $lr = 0.01$ . The error, plotted in Fig. 13, was defined as the percentage of wrong tags and is taken as the performance parameter for all the models tested in this work.

As can be seen from Fig. 13, all the models achieve optimal results after training. Among them, BiLSTM-Softmax model (blue curve) is seen to have the highest error of about 15%. However, BiLSTM-CRF (green curve) and BiLSTM-LAN (red curve) models show a smaller error

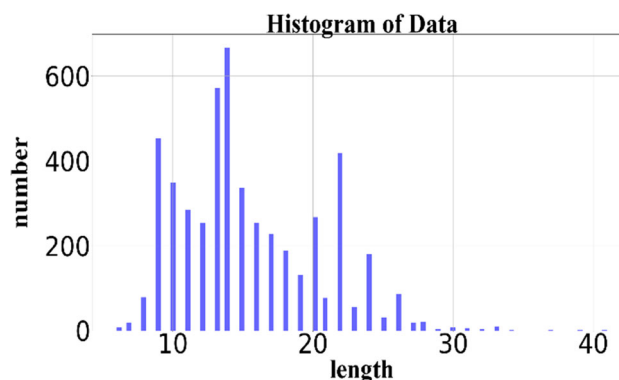


Fig. 12 The histogram of data

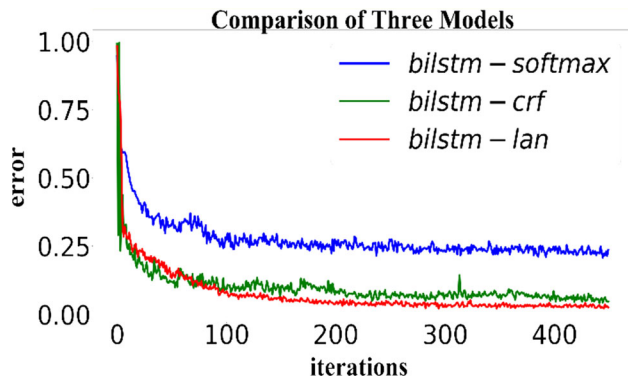


Fig. 13 Plot showing the performance of BiLSTM-Softmax (blue curve), BiLSTM-CRF (green curve) and BiLSTM-LAN algorithms (color figure online)

of about 5 and 2%, respectively. Their performance is similar due to the short length of the control instruction.

As is given in Table 3, LSTM is used to represent the input sequence in BiLSTM-Softmax and BiLSTM-CRF models. The BiLSTM-LAN model uses LAN for representation, which has a better representational ability. Therefore, the performance of BiLSTM-LAN model is observed to be the best. Furthermore, CRF can output the optimal label sequence in the output layer, and thus, BiLSTM-CRF works better than BiLSTM-Softmax model.

The label of each word in a control instruction does not depend on any long-range labels due to the flexible word order of some terms. Therefore, CRF does not perform worse than LAN in the output layer for the control instruction. Thus, the most important reason for the better performance of BiLSTM-LAN model is because of its encoder–decoder structure, which enables it to represent the input better.

### 5.3 Performance of BiLSTM-LAN-CRF

This subsection describes the experimental results of our model. For the experiment, the LAN parameters were set as follows:  $d = 256$ ,  $num = 1$ , and 2 hidden layers of BiLSTM were used with each hidden layer containing 64 neurons. As shown in Fig. 14, the batch size was chosen to be 50, epochs = 10, dropout = 0.5, the cross entropy loss function was used as the loss function, and the Adam

Table 3 The testing error of three models

Model	Testing error (%)
BiLSTM-Softmax	15.46
BiLSTM-CRF	5.57
BiLSTM-LAN	2.43

optimizer with parameters  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ ,  $\epsilon = 10^{-8}$ , and learning rate,  $lr = 0.01$  was used.

As can be seen from Fig. 15, the test error of BiLSTM-LAN-CRF (2.82%) is as similar as that of BiLSTM-LAN (2.43%), with fewer parameters. As can be seen from Fig. 16, BiLSTM-LAN-CRF obtains the lowest test error with the same number of parameters.

The experiment also shows that the test error is similar with different numbers of heads used in our model. The testing error is computed by using the testing data as post in Table 4.

As can be seen from the table, the lowest testing error is 2.86% when the head number is defined as 2. This indicates that our model performs better than the BiLSTM-CRF model. This is because our model also incorporates the encoder–decoder structure and has a better ability to representation. Moreover, both the lowest test error of our model and BiLSTM-LAN model are similar due to the flexible word order of the control instruction, which reduces the long-range dependency of the label sequence. However, our model uses fewer parameters than BiLSTM-LAN model.

### 5.4 Performance of new algorithm

A description of the experiment carried out using our algorithm is presented in this subsection. The algorithm uses BiLSTM-LAN-CRF to extract the entities, verbs and constructions first, followed by the use semantic ontology, built using the Protégé tool (see Fig. 4), to support the relations. Classes in the semantic ontology contain the entities, prepositions, cases and others. The class of entities includes 10 elements, the class of prepositions includes four prepositions, and the class of cases includes 10 cases. The relation contains control verbs and auxiliary verbs. The control verbs are important for aircraft movement and includes about 20 verbs.

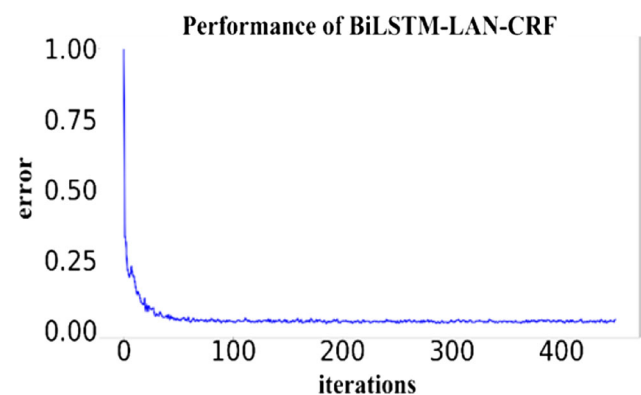
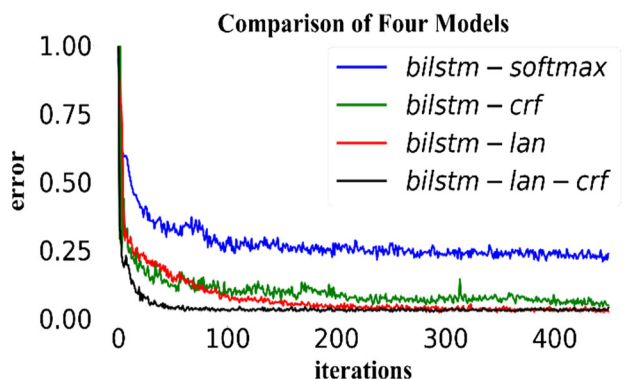
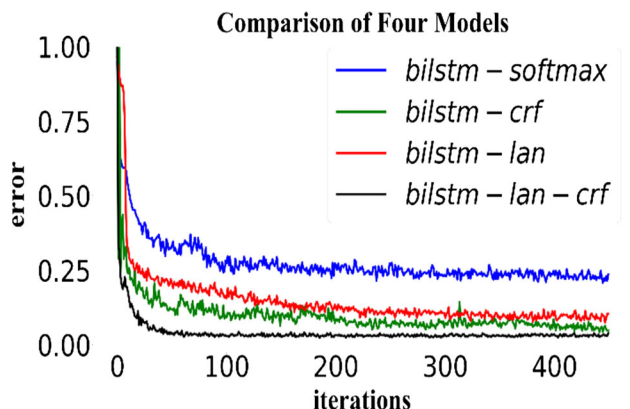


Fig. 14 The performance of BiLSTM-LAN-CRF model



**Fig. 15** Plot showing a comparison of BiLSTM-Softmax (blue curve), BiLSTM-CRF (green curve), BiLSTM-LAN (red curve) and BiLSTM-LAN-CRF (black curve) algorithms with the best settings (color figure online)



**Fig. 16** Plot showing a comparison of BiLSTM-Softmax (blue curve), BiLSTM-CRF (green curve), BiLSTM-LAN (red curve) and BiLSTM-LAN-CRF (black curve) algorithms with the same settings (color figure online)

**Table 4** The testing error of our models with different numbers of heads

Num of heads	Testing error (%)
1	3.47
2	2.65
4	5.24
8	4.991

The Chinese instruction “ji xiang yao liang si liang di mian jing feng pao dao yao guai ke yi qi fei,” which is “DKH1242, static wind, Runway 17, take-off” in English, is used as the input of the algorithm and the BiLSTM-LAN-CRF model is used to process it. As a result, the following label sequence is generated: B-FLY I-LY I-FLY

**Table 5** The result of entity extraction

Label	Flight call	Weather	Runway	Action
Entity	DKH1242	Static wind	17	Take-off

**Table 6** The structural form of control instruction (e.g., 1)

Action	Entity	Case
Take-off	DKH1242	Agentive
	Static wind	–
	Runway 17	Locative

**Table 7** The structural form of control instruction (e.g., 2)

Actions	Entity	Case
Climb, maintain	DKH1116	Agentive
	QNH900	Target

I-FLY I-FLY I-FLY B-WHE I-WHE I-WHE I-WHE B-RW I-RW I-RW I-RW B-ACT I-ACT I-ACT I-ACT. Therefore, as post in Table 5, the entities with their labels are obtained.

The verb “take-off” is denoted as the central word, and the following pairs containing the verb and the label of entities are obtained: “flight call, take-off,” “weather, take-off,” “runway, take-off.” These pairs are then given as an input to the semantic ontology to find the relation between them. Among them, the relation between flight call and take-off is agentive, while the relation between runway and take-off is locative. There is no case to describe the relation between weather and take-off, and therefore, the structural form is generated as post in Table 6.

It is necessary to extract the prepositional construction and verb–object construction if any prepositions or a large number of verbs are present in the control instruction. For example, in the Chinese instruction “ji xiang yao yao yao liu shang shen dao xiu zheng hai ya jiu bai bao chi” which is “DKH1116, climb to QNH 900 and maintain” in English, there are two verbs “climb (shang shen),” “maintain (bao chi)” and the preposition “to (dao).” After extracting the constructions, the following label sequence is obtained: O O O O O B-VOC I-VOC I-VOC I-VOC I-VOC I-VOC I-VOC I-VOC I-VOC I-VOC I-VOC I-VOC, where VOC denotes the verb–object construction, and label sequence: O O O O O O O O B-PREP I-PREP I-PREP I-PREP I-PREP I-PREP

I- PREP O O, where PREP denotes the circumposition construction. In the verb–object construction, “climb to QNH 900 and maintain,” both the verbs “climb” and “maintain” make up the verb–object construction with “QNH 900.” It is clear that “QNH 900” is the target of the verbs based on the circumposition construction “to QNH 900” Thus, the structural form from the semantic ontology is generated as shown in Table 7.

The new algorithm can process unstructured control instruction and generate the structural form based on the “entity, case, verb” tuple.

## 6 Conclusions

This paper describes a new algorithm of structural processing for Chinese ATC instructions, which can generate structural instruction for the systems. This algorithm can be used in an automated system for many applications such as predicting the trajectory of the aircraft and conflict detection. The algorithm consists of two steps: (i) entity extraction and construction extraction, and (ii) semantic analysis by semantic ontology. The following are the key points of this work:

1. On using a dependency parser to analyze the control instruction, the accuracy of the results is observed to decrease. This is because of the flexible word order of a sentence and the ability of preposition disappearance. Some terms of the control instruction do not depend on any verbs, and the instruction thus has a flexible order. Moreover, the circumposition construction can disambiguate the relation between an entity and a verb based on the construction grammar theory. In addition, the stronger ability of preposition disappearance in a control instruction indicates that the relation between the entity and the verb is usually unique.
2. Based on the linguistic structure of the control instruction, the semantic ontology is built and the structural form is designed by the case grammar theory. The semantic ontology can support the correct semantic relation (defined as case) between an entity and a verb, which can generate the structural instruction.
3. The entity extraction and construction extraction are used instead of parsing. The constructions include a preposition and a verb–object, and the construction extraction is as similar as an entity extraction. BiLSTM-Softmax, BiLSTM-CRF and BiLSTM-LAN models have been used for this task. This work also proposes a new model named BiLSTM-LAN-CRF, which can obtain a better performance of entity extraction for control instruction as compared to the other three models.
4. Based on the above points, a new algorithm of structural processing for Chinese ATC instructions has been proposed, which can be used for predicting the trajectory only from the control instructions.

The algorithm of structural processing can be used to convert non-structural ATC instructions. The key information would become uncertainty if there are some dialects or errors in the ATC communications. The proposed algorithm cannot process ATC instruction with errors. So next work needs to improve robustness of this algorithm. On the other hand, there are some other factors that impact trajectory prediction, such as weather, emergency and the wrong operations by controller or pilot; it is necessary to consider more information from other data in order to predict more accuracy trajectory. So next work also needs to improve performance of algorithm by incorporating more other information.

**Acknowledgements** This research is supported by the National Key Research and Development Project (No. 2020YFB2104204) and the China Postdoctoral Science Foundation (No. 2020M681750)

## Compliance with ethical standards

**Conflict of interest** The author declares that they have no conflict of interest.

## References

1. Helmke H, Slotty M, Poiger M, et al (2018) Ontology for transcription of ATC speech commands of SESAR 2020 solution PJ.16–04. In: 37th digital avionics systems conference (DASC)
2. Nguyen VN, Holone H (2016) N-best list re-ranking using syntactic score: a solution for improving speech recognition accuracy in Air Traffic Control. In: 16th int. conf. on control, automation and systems (ICCAS 2016), pp 1309–1314
3. Nguyen VN, Holone H (2016) N-best list re-ranking using syntactic relatedness and syntactic score: an approach for improving speech recognition accuracy in Air Traffic Control. In: 16th int. conf. on control, automation and systems (ICCAS 2016), pp 1315–1319
4. Wang X, Wang G, Cheng XQ (2019) A new structural template design of control instruction for semantic analysis. In: Proceedings of CICTP, pp 2935–2945
5. Goldberg AE (1985) Construction: a construction grammar approach to argument structure. The University of Chicago Press, Chicago
6. Wang SQ (2016) Research of prepositions in contemporary Chinese (in Chinese). Nanjing University Press, Nanjing
7. Fillmore CJ (1968) The case for case. In: Bach E, Harms RT (eds) Universals in linguistic theory. Holt, Rinehart, New York
8. Yuan YL (2010) Research on Chinese valency theory (in Chinese). The Commercial Press, Beijing
9. Lafferty J, McCallum A, Pereira F (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of ICML, pp 282–289

10. Hammerton J (2003) Named entity recognition with long short-term memory. In: HLT-NAACL 2003, vol 4, pp 172–175
11. Kiperwasser E, Goldberg Y (2016) Simple and accurate dependency parsing using bidirectional LSTM feature representations. *TACL* 4:313–327
12. Dozat T, Manning C (2017) Deep biaffine attention for neural dependency parsing. In: *ICLR*
13. Zhang Y, Yang J (2018) Chinese NER using lattice LSTM. In: *ACL*, pp 1554–1564
14. Collobert R, Weston J, Bottou L, Karlen M et al (2011) Natural language processing (almost) from scratch. *J Mach Learn Res* 12:2493–2537
15. Santos C, Guimaraes V (2015) Boosting named entity recognition with neural character embeddings. In: *ACL*, pp 25–33
16. Chiu J, Nichols E (2016) Named entity recognition with bidirectional LSTM-CNNs. In: *ACL*, pp 357–370
17. Huang Z, Xu W, Yu K (2015) Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint [arXiv:1508.01991](https://arxiv.org/abs/1508.01991)*
18. Lample G, Ballesteros M, Subramanian S, Kawakami K, Dyer C (2016) Natural architectures for named entity recognition. In: *ACL*, pp 260–270
19. Ma X, Hovy E (2016) End-to-end sequence labeling via vi-directional LSTM-CNNs-CRF. In: *ACL*, pp 1064–1074
20. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, et al (2017) Attention is all you need. In: *Advances in natural information processing system*, pp 6000–6010
21. Cao P, Chen Y, Liu K, Liu S (2018) Adversarial transfer learning for Chinese named entity recognition with self-attention mechanism. In: *Conf. on empirical methods in natural language processing*, pp 182–192
22. Cui L, Zhang Y (2019) Hierarchically-refined label attention network for sequence labeling. In: *ACL*, pp 4115–4128
23. Chen Y, Xu L, Liu K, Zeng D, Zhao J (2015) Event extraction via dynamic multi-pooling convolutional neural networks. *ACL* 1:167–176
24. Nguyen T, Cho K, Grishman R (2016) Joint event extraction via recurrent neural networks. In: *Conf. on the North American Chapter of ACL: human language technologies*, pp 300–309
25. He L, Lee K, Lewis M, Zettlemoyer L (2017) Deep semantic role labeling: what works and what’s next. In: *Proceedings of the 55th annual meeting of the association for computational linguistics*, pp 473–483
26. Tan Z, Wang M, Xie J, Chen Y, Shi X (2017) Deep semantic role labeling with self-attention. *arXiv preprint [arXiv:1712.01586](https://arxiv.org/abs/1712.01586)*
27. Strubell E, Verga P, Andor D, Weiss D, McCallum A (2018) Linguistically-informed self-attention for semantic role labeling. In: *ACL*, pp 5027–5038
28. Guha RV, McCool R, Miller E (2003) Semantic search. In: *Proceeding of the 12th international World Wide Web conference*, pp 700–709
29. Wang WL, Zhao Q (2016) *Radiotelephony communications for air traffic controllers (in Chinese)*. Tsinghua University Press, Beijing
30. Newman J (2004) Motivating the uses of basic verbs: linguistic and extralinguistic considerations. In: Radden G, Panther K-U (eds) *Studies in linguistic motivation*. Mouton de Gruyter, Berlin, pp 193–218
31. Hiraga MK (1994) Diagrams and metaphors: iconic aspects in language. *J Pragmat* 22:5–21
32. Lakoff G (1987) *Woman, fire and dangerous things: what categories reveal about the world*. The University of Chicago Press, Chicago
33. Lin GX, Wang LL, Sun DJ (1994) *Dictionary of verbs in contemporary Chinese (in Chinese)*. Beijing Language Institute Press, Beijing

**Publisher’s Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.