Situation Semantics Model of Text Semantic Understanding

Zhu Ping[1, a], Tan Yuanhua[2, b]

[1]Beijing Institute of KARAMAY HONGYOU Software CO., LTD., Beijing 100085, China

[2]Beijing Institute of KARAMAY HONGYOU Software CO., LTD., Beijing 100085, China

[a]email: zhuping2404@sohu.com, [b]email: tanyh66@petrochina.com.cn

Keywords: Situation; Knowledge Representation; Reason; Semantics Model; Common Sense

Abstract. This paper summarizes the theoretical basis of text semantic understanding, puts forward situation semantics model as framework to improve the case grammar, combined with several other semantic methods. It takes situation segmentation as a starting point, and text understanding as the goal, describes the common sense knowledge representation and software architecture design technology of situation semantics model. Finally, illustrates the application method and reasoning mechanism of situation semantics model with some examples.

Introduction

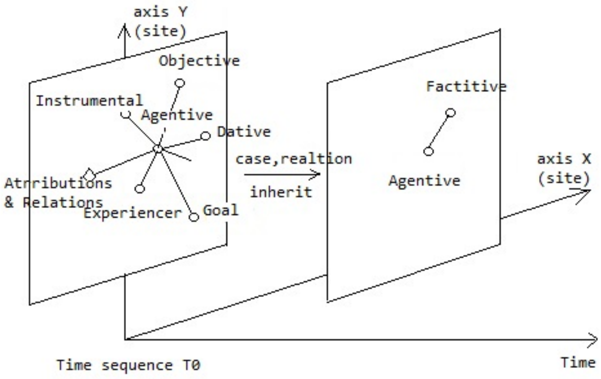
Existing studies of language information process are not semantic oriented, but syntax oriented [1]. This is one of the biggest shortcoming of existing semantic research, so it is difficult to really solve the language information processing problems encountered, including the Chinese word segmentation and the syntactic ambiguity problems encountered.

At present, the relevant theories of semantic problem mainly include: (1) the conceptual dependency theory, it proposed that all semantics can be described by a few number of primitive operations. (2) semantic field theory, it presents that the semantics are hierarchical, which can classify the semantics of the concept. (3) case grammar theory, it thought that the semantics behind the language follow into fixed case form, discussed the unchange semantic description based on syntactic changed description. (4) HowNet, it described some sememes, and the levels of taxonomic relations, represented the Chinese semantic relations behind the concept [2][3][4][5]. The Semantic Oriented Computing (SOC) has 3 meanings: (1) There are several meaning with a sentence (polysemous problem)? (2) How to determine the different meanings are similar (two sentences similarity problem)? (3) How to determine the meaning irrelevant sentences with similar truth values (two sentences related problem)? The three problems are the core problems of sentence semantic computing.

Starting from the analysis of paragraph semantics, this paper describes the situation semantics model with starting point of situation segmentation, it fuses the basic idea of the conceptual dependency theory, the case grammar theory and HowNet, puts the case grammar theory as the basic framework, combines common sense primitives and HowNet’s relation descriptions, uses the main idea of property inheritance of situation sequences, builds a variety of reasoning algorithms. The next section, we will discuss the situation semantics model. The third section, we will discuss the technical framework of realization. The fourth section, we will use the actual examples to illustrate the application of the situation semantics model. Finally, we will put forward the direction of further work.

Situation Semantics Model

The situation semantics model proposes that the semantic analysis is not depended on units such as sentences, but on units such as paragraph situation sequences. As showing the story with hand-painted cartoon, each situation is displayed by a picture, the pictures are arranged in a time-event sequence, continuous played like movie. The adjacent pictures has the inheritance relationship, i.e. not updated description content will remain unchanged, a new description of the content will replace the old location content. The whole story maintains an animation effect on the visual.



Each situation uses case grammar as the framework, Action case is added, for nested description of the son situation. Each situation uses common sense primitive description, including HowNet semantics, as a accessory to refine the semantics representation capability. It could be represented by Knowledge framework in detail as follow: Agentive, Action, Locative, Goal, Time, ect.; attribute relations between various cases; common sense primitives description represents synonyms, antonyms, common sense relations, predicate relations and knowledge state transition relations; truth value of situation (true/false) [6]; the Action case has tense & state relations, and can be nested represented sub-situations; and other various common sense primitives.

Knowledge framework	Category	Content
Situation ID		N
Agentive	Case description	fox
	Attributes	<fox,hungry>, <fox,a>
Action	Case description	went
	Tense & state relation	<went,past,done>
	Attributes	
Locative	Case description	vineyard
	Attributes	<vineyard,a>
Goal	Case description	food
	Attributes	
Time	Case description	one day
	Attributes	<one day, very hot>
Objective	Case description	
	Attributes	
…	…	…
Situation truth value	true/false	true
Next Situation ID		N+1

Common sense primitive description	Synonym	[hungry, so hungry, very hungry], [hot, very hot]
	Antonym	[hungry, full]
	Common sense relation	<summer, heat>, <grapes, mature, summer >
	Common sense state transition relation	<<eat, now, done>, food, hungry, full>, <<eat, past, done>, food, hungry, full>
	Common sense predicate relation	have<vineyards, grape>, belongs<grape, food>
	Common sense primitive	the inference rules of the various relations, such as common sense reasoning relations: < reach, now, done >, < eat, now, done> >; situation reasoning rules, such as the rules of Agentive inherited in different situation; situation truth value, for example, tell lies.
	Concept: (part of speech, < features, the operations that can be performed, rules>)	thing({noun}, <{quantity}, {all, optional, each, queuing}, {}>)

Each element of the above description is a concept, each concept is composed by (part of speech, < features, the operations that can be performed on concept and features, rules>), contained in the Common sense primitive description framework. For example:

thing ({noun}, <{quantity}, {all, optional, each, queuing}, {}>)

set ({noun},<{element, element quantity},{partition(fore, back; fore, middle, back)}，{the element quantity = add fore part and back part son element quantities, the element quantity = add fore part, middle part, and back part element quantities}>)

element ({noun}，<{quantity}，{}，{}>)

number ({noun}, <{quantity(1); value}, {read; write}, {}>, <{quantity(2); average value; total; difference; product; quotient; remainder}, {add; minus; plus; divide; arranging; queuing}, {average value = total - divided by - quantity(2); total=average value - plus - quantity(2); quantity(2) = total - divided by - average value}>, <{quantity(N); average value; total; product}, {arranging; queuing}, {average value = total - divided by - quantity(N); total=average value - plus - quantity(N); quantity(N)=total - divided by - average value}>)

row ({noun}, <{quantity(1); fore; back; middle; order; neighbor; quantity}, {partition (fore, back; fore, middle, back)}, {the element quantity = add fore part and back part son element quantities, the element quantity = add fore part, middle part, and back part element quantities, the difference of order value of neighbor elements is 1}>, <{quantity(N); fore; middle; after; order; NO.}, {partition(fore, back; fore, middle, back)}, {the element quantity = add fore part and back part son element quantities, the element quantity = add fore part, middle part, and back part element quantities, the difference of order value of neighbor elements is 1}>)

When problem solving, according to the specific semantic environment description, the knowledge framework content is instantiated. Such as situation{One very hot day, a hungry fox went into a vineyard looking for food.}, see above table of instantiated description; the common sense attributes are also instantiated this way, such as situation{5 numbers are written in a row}， instantiated description as follow: The concept of number constricted by five, forming a new concept of running, namely a row of five numbers. It has attribute 5 and attribute row. Can be represented as number({noun},<{quantity(5), row(quantity(1))},{},{}>). And the concept row has self attributes and the attributes inherited from concept set. Hence, the running new concept has the attributes of concept thing and concept number, and also has self attributes of concept row and attributes of concept set.

The semantic frameworks of situation (including the Knowledge framework and Common sense primitive description) can precisely describe the semantics of language, but also could describe the refinement and generalization of the semantic of the situation semantic frameworks. For example, if we tell the story of “the fox couldn't reach the grapes and complained that they were sour”, every sentence can be accurately described with a semantic framework, then, a deeper semantic can be summarized as idiom{sour grapes}, the semantic frameworks of above fox story should match the idiom to some extend. The method is that we previously set manually semantic frame content of the idiom. For example, idiom {sour grapes} can be described with semantic frameworks as follow:

Knowledge framework	Category	Content
Situation ID		0001
…	…	…
Action	Case description	reach
	Tense & state relation	<reach,now,undo>, <reach,past,undo>
	Attributes	
…	…	…
Objective	Case description	grapes
	Attributes	
…	…	…
Situation truth value	true/false	true
Next Situation ID		0002

Knowledge framework	Category	Content
Situation ID		0002
…	…	…

Action	Case description	complain
	Tense & state relation	<complain,now,done>, <complain,past,done>
	Attributes	
…	…	…
Objective	Case description	grapes
	Attributes	<grape,sour>
…	…	…
Situation truth value	true/false	false
Next Situation ID		END

You can also expand the literal meaning, describes as {can't get something, say something bad}, as follows:

Knowledge framework	Category	Content
Situation ID		0001
…	…	…
Action	Case description	get
	Tense & state relation	<get,now,undo>, <get,past,undo>
	Attributes	
…	…	…
Objective	Case description	X
	Attributes	
…	…	…
Situation truth value	true/false	true
Next Situation ID		0002

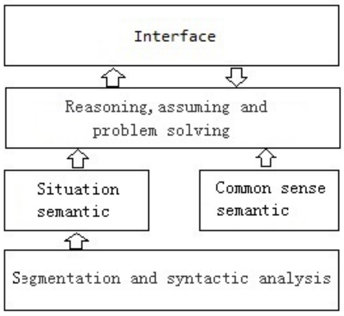
Knowledge framework	Category	Content
Situation ID		0002
…	…	…
Action	Case description	say
	Tense & state relation	<say,now,done>, <say,past,done>
	Attributes	
…	…	…
Objective	Case description	X
	Attributes	<X,bad>
…	…	…
Situation truth value	true/false	false
Next Situation ID		END

Attention: common sense Synonym [bad,sour] and common sense reasoning relations <<reach, now, done>, <eat, now, done>>, <<reach, past, done>, <eat, now, done>>.

According to the situation descriptions of the two idioms and the semantic frameworks of the above story, calculate the semantic distance, we know that they have the related meaning.

Technical Architecture

The technical architecture of semantic understanding is divided into 5 parts: namely, segmentation and syntactic analysis; situation semantic representation; common sense semantic representation; reasoning, assuming and problem solving; human-computer interaction interface.



First, the ambiguity problem should be resolved in segmentation and syntactic analysis, you can first solve a part in common sense range; in addition, the polysemy problem can resolve in semantic analysis. For example: {installing bulbs workers}, syntactic analysis has the choice of 2 ambiguity: ((V+N)(+N) or V+(N+N)), then we can conclude that the former is correct, based on common sense "the object of installing should be the equipment". Another example: {perching gun nearby} has 2 meanings, one is the static state description; the other is the dynamic description of action. The choice should be carried out in the situation context. In word segmentation and syntactic analysis stage, we can mark two kinds of meaning, and solve the problem by computing degree of situation semantic fluency.

Semantic annotation is the foundation of automatic semantic understanding [7]. The purpose is manual tagging and segmentation in first, and then, judging case attributes with syntactic and common sense. For example, {One very hot day, a hungry fox went into a vineyard looking for food.}, the core word {day} of adverbial{One very hot day,} can be annotated as Time case{day} and Time case attributes <day, one> and <day, very hot>; subject {a hungry fox} can be annotated as Agentive {fox} and attributes <fox, hungry> and <fox, a> (Attention: fox has the animal attribute, hence fox is Agentive); The first predicate part {went into a vineyard} can be annotated as Action case {went into} and Tense & state relation <went, past, done>, the object {vineyard} of Action {went into} can be annotated as Locative case {vineyard} with common sense and attribute < vineyards, a>; the second predicate part {looking for food} can be annotated as Goal case {food}. When a certain amount of situation semantic annotations are accumulated, can develop automatic situation semantic representation and transforming algorithm, automatic represent semantic knowledge in text.

Common sense is the necessary premise to solve the semantic problem. In addition to the seven basic forms of common sense primitive description in the second section, do not refuse new common sense forms. There is transition relationship between the situation semantic representation and common sense semantic representation. After the text is represented by the situation semantics frameworks, the run-time knowledge can also be learned to become the new common sense. We can look at the semantics knowledge framework, the relationship between the concepts, and the relationship between concepts and their attributes as common sense, and look at the attribute value of the concept and the case attribute filling as run-time concepts instantiated. That is to say, look at the semantics knowledge framework, the relationship between the concepts, and the relationship between concept and their attributes as the classes in OOP program realization, and look at the attribute value of the concept and the case attribute filling as objects instantiated. Generally, we think that common sense include common sense relation, common sense state transition relation, common sense predicate relation, common sense primitive and the concept description. The common sense representation can make up the situation semantic representation in problem solving, so that the solution can be carried out in the premise that can be explained. The situation knowledge also can be preserved as common sense in some canonical forms, can work as background for problem solving in the future.

Reasoning, assuming, and problem solving consist of two aspects:

(1) The inference types of existing knowledge have deduction, induction and analogy ect., the method of traditional heuristic problem solving is able to solve the problems. For example, the

above story of {sour grapes}, finally judging if it is right situation or not:{Time case {Finally}, Agentive {fox}, Action case {eat}, Objective {grapes}, Action attribute <eat, past, done>}. You can retrieval and reason the situations and common sense, and know that is false.

(2) The abstract description of certain human common intuition ability, such as the intuition processing mechanism of natural numbers, order, enumeration, the difference and partition of set must be solved by non monotonic reasoning. Including forecast, assumptions, and sensory intuition training achievements that cannot be described by accurate knowledge. For example, for {In 99 consecutive natural number, the maximum number is 25.5 times the minimum number, then what is the average number of these 99 natural number? }, the method for solving this problem is: there is no background knowledge and situation representation can directly reason result, as long as people enumerated sequence from the initial number of 1, a few steps you can try out the answer. And for such as the concept: the number quantity of the fore-part of the queue, the number quantity of the back-part of the queue, must use human observation obtained from partitioning, and verify the legality (the sum of the fore-part and the back-part is equal to the the total quantity of the sequence), these are only be processed in the modular of reasoning, assuming and problem solving.

The interface of human-computer interaction is divided into two aspects: input and output. For input, can manually annotate situation semantics and common sense into the system, also the develop automatic annotation tools in a certain amount of manual annotations. Input can accept a single semantic record, can also accept the text scanning process. For the output, you can divide the intelligent level, the low level of output can only solve the specific problem, such as the fourth grade primary school mathematics application problem, or reading comprehension problem. The high level Intelligent output can randomly response on human natural language problems.

Application Examples

In this section we will take two specific examples to illustrate the application of the situation semantics model:

Example 1: {At last, quite tired out, he said: “Who wants grapes like those, anyway? Anyone can see they are as sour as green lemons.” He tried to comfort himself by saying so.}

Semantics representation: Time case {at last}, Action case {tired out}, Action case attribute <tired out, quite>, Agentive {he}, Action case {said} nests {Agentive {who: question|no one|everyone}, Action case {want}, Goal {grapes like those}, Agentive {Anyone}, Action case {can see} nests {reference {they}, Synonym sour<they, [green] lemon>}}, <lemon, sour> and <lemon, green, sour>}}, Agentive {he}, Action case {tried to} nests {Action case {comfort}, Objective {himself} and Path {saying so}};

Attention: the Action case {said} nested situation truth value assigns false.

Problem: {The grapes are as sour as green lemons or not?}

Semantics representation: Synonym sour<grapes, [green] lemons>

Reasoning Algorithm:

(1) Agentive {fox} replaces reference, such as {he};

(2) Travel situations according Time case sequence, search Synonym sour<grapes, [green] lemons>, which should replaces synonyms.

(3) if it is not founded, exit with UNKNOWN, otherwise get the situation truth value, if true, would end with YES, otherwise exit with NO.

The process of solving the above example is as follows:

(1) Agentive {fox} replaces reference Agentive {he}, Goal case {grapes} replaces reference {they};

(2) Found the Synonym sour<they, [green] lemon>, reasoning {Synonym sour<grapes, [green] lemons>} is true;

(3) Search situation truth value is false;

(4) reason the answer NO.

Example 2: {5 numbers are written in a row, the average value of 3 fore numbers is 15, average value of 2 back numbers is 10.}

Semantics representation:

number({noun},<{quantity(5),row(quantity(1))},{},{}>)

number({noun},<{quantity(3),average(15),row(quantity(1)),fore},{},{}>)

number({noun},<{quantity(2),average(10),row(quantity(1)),back},{},{}>)

Problem:{What's the average of these 5 numbers?}

Semantics representation: number({noun},<{quantity(5), average(?)},{},{}>)

Reasoning Algorithm:

(1) Search feature calculation rules, see whether the conditions are satisfied; if meet, is solved;

(2) Search all calculation rules of solution conditions, reason results;

(3) Search all features calculation rules of solving, if need calculation conditions, question all calculation conditions;

(4) Check if all calculation conditions are satisfied, if satisfied, success;

(5) If can't reason new result with features calculation rules, exit(There is no answer for this problem);

(6) Jump to step 2, loop.

The process of solving the above example is as follows:

(1) The average of fore 3 numbers is 15, semantics represent as follows :

number({noun},<{quantity(3), average(15), total(45), row(quantity(1)), fore},{},{}>)

(2) The average of back 2 numbers is 10, semantics represent as follows :

number({noun},<{quantity(3), average(10), total(20), row(quantity(1)), back},{},{}>)

(3)What's the average of these 5 numbers? Reasoning as follows:

number({noun},<{quantity(5), average(?), total(??), row(quantity(1))},{},{}>)

(4) number({noun},<{quantity(5), average(?), total(??), row(quantity(1))}, {}, {}>):: total(??) = number ({noun},<{quantity(3), average(15), total(45), row(quantity(1)), fore},{},{}>):: total(45) + number({noun},<{quantity(3), average(10), total(20), row(quantity(1)), back},{},{}>):: total(20); Attention: Judge the legitimacy.

(5) number({noun},<{quantity(5), average(?), total(65), row(quantity(1))},{},{}>)

(6) number({noun},<{quantity(5), average(13), total(65), row(quantity(1))},{},{}>)

Conclusion

This paper mainly discusses the semantic knowledge representation and reasoning problems under the situation condition. It is worth noting that the method in this paper is textual level solutions. For the 3 core semantic computing problems that this paper mentioned at the beginning: A polysemous problem solves by analyzing the syntactic and lexical represented by text context and choosing with degree of semantic fluency; Two sentences similarity problem solves by reasoning synonym in common sense primitive description, may reason by degree of confidence; Sentences related problem solves by Situation truth value and related situation attributes. This paper focuses on suitable strategy of knowledge representation at present, reasoning carried out with simpler form, and expressing more common sense knowledge. The next step, will focus on reasoning decision strategies about hyper-heuristic and semantic knowledge representation transformation towards common sense knowledge. Expect that plenty of semantic annotations will bring many new research directions.

Acknowledgement

Project supported by national 863 plan (2013AA01A607), the Xinjiang Uygur Autonomous Region major projects (201130110-3) and the Karamay Technology Bureau (SK2012-050).

References

[1] Qi Xuan, Ma Hongmei, Chen Huowang. Research of Chinese Semantic Analysis [J], Computer Science and Engineering, 2001 23 (3) 89-92.

[2] Qiu Zhenqiangyao. Chinese Semantic Research [C], Central South University press.

[3] Zhu Qian, Han Fei, Cheng Xianyi. Sentence Semantic 3D Representation Model [J], Journal of Intelligent Systems, 2009 4 (2) 43-46.

[4] Hou Yonghua. Contextual and Semantic Understanding [J], Neijiang Science and Technology, 2009 (3) 47-47.

[5] Yin Yan. Understanding Model Based on Semantic Image Schema [D], Master Degree Thesis of Nanjing Agricultural University, 2013.6.

[6] Sun Bo, He Kekang. The Situation Theory of Chinese Understanding Based on the Model [J], Computer Research and Development, 1992 (4) 1-19.

[7] Bai Xue. Semantic Role Labeling Based on Structure Learning [D], Master Degree Thesis of Beijing University of Posts and Telecommunications, 2010.1.