International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

RESEARCH ARTICLE

A Review of NLP Based Upon Semantic Analysis

Parneet Kaur[*]
Student of Master Technology
Department of CSE
Desh Bhagat University
Mandi Gobindgarh, Punjab, India
parneetbenipal@gmail.com

Vandana Pushe[*]
Assistant Professor
Department of CSE
Desh Bhagat University
Mandi Gobindgarh, Punjab, India
vandanapushe@gmail.com

Rupinderdeep Kaur*
Assistant Professor
Department of CSE
Thapar University
Patiala, Punjab, India
rupinderdeep@thapar.edu

Abstract— The ability to understand natural-language instructions is critical to building intelligent agents that interact with humans. We present a system that learns to transform natural-language navigation instructions into executable formal plans. Given no prior linguistic knowledge, the system learns by simply observing how humans follow navigation instructions. The system is evaluated in three complex virtual indoor environments with numerous objects and landmarks. A previously collected realistic corpus of complex English navigation instructions for these environments is used for training and testing data.

Keywords— Semantic Parser, Formal Language, Ontology, Inference Mechanism, Distributional Semantics

I. INTRODUCTION

Semantic Parsing is probably best defined as the task of representing the meaning of a natural language sentence in some formal knowledge representation language that supports automated inference. A semantic parser is best defined as having three parts, a formal language, an ontology, and an inference mechanism. Both the formal language (e.g. first-order logic) and the ontology define the formal knowledge representation. The formal language uses predicate symbols from the ontology, and the ontology provides them with meanings by defining the relations between them.1. A formal expression by itself without an ontology is insufficient for semantic interpretation; we call it uninterpreted logical form. An uninterpreted logical form is not enough as a knowledge representation because the predicate symbols do not have meaning in themselves, they get this meaning from the ontology. Inference is what takes a problem represented in the formal knowledge representation and the ontology and performs the target task (e.g. textual entailment, question answering, etc.). Prior work in standard semantic parsing uses a pre-defined set of predicates in a fixed ontology. However, it is difficult to construct formal ontologies of properties and relations that have broad coverage, and very difficult to do semantic parsing based on such an ontology. Consequently, current semantic parsers are mostly restricted to fairly limited domains, such as querying a specific database (Kwiatkowski et al., 2013; Berant et al., 2013). Paper proposed a semantic parser that is not restricted to a predefined ontology. Instead, we use distributional semantics to generate the needed part of an on-the-fly ontology. Distributional semantics is a statistical technique that represents the meaning of words and phrases as distributions over context words (Turney and Pantel, 2010; Landauer and Dumais, 1997). Distributional information can be used to predict semantic relations like synonymy and hyponymy between words and phrases of interest (Lenci and Benotto, 2012; Kotlerman et al., 2010). The collection of predicted semantic relations is the "on-the-fly ontology" our semantic parser uses. A distributional semantics is relatively easy to build from a large corpus of raw text, and provides the wide coverage that formal ontologies lack. The formal language we would like to use in the semantic parser is first-order logic. However, distributional information is graded in nature, so the on-the-fly ontology and its predicted semantic relations are also graded. This means, that standard first-order logic is insufficient because it is binary by nature. Probabilistic logic solves this problem because it accepts weighted first order logic formulas.
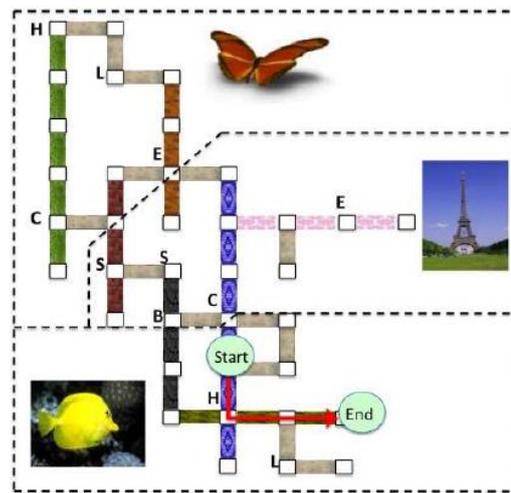
Fig. 1 This is an example of a route in our virtual world. The world consists of interconnecting hallways with varying floor tiles and paintings on the wall (butterfly, fish, or Eiffel Tower.) Letters indicate objects (e.g. 'C' is a chair) at a location.

For example, in probabilistic logic, the synonymy relation between "man" and "guy" is represented by: 8x. man(x) , guy(x) | w1 and the hyponymy relation between "car" and "vehicle" is: 8x. car(x)) vehicle(x) | w2 where w1 and w1 are some certainty measure estimated from the distributional semantics. For inference, we use probabilistic logic frameworks like Markov Logic Networks (MLN) (Richardson and Domingos, 2006) and Probabilistic Soft Logic (PSL) (Kimmig et al., 2012). They are Statistical Relational Learning (SRL) techniques (Getoor and Taskar, 2007) that combine logical and statistical knowledge in one uniform framework, and provide a mechanism for coherent probabilistic inference. We implemented this semantic parser (Beltagy et al., 2013; Beltagy et al., 2014) and used it to perform two tasks that require deep semantic analysis, Recognizing Textual Entailment (RTE), and Semantic Textual Similarity (STS).

II. RELATED WORK

Building systems that learn to interpret navigation instructions has recently received some attention due to its application in building mobile robots. Our work is the most similar to that of Matuszek et al. (2010). Their system learns to follow navigation instructions from example pairs of instructions and map traces with no prior linguistic knowledge. They used a general-purpose semantic parser learner WASP (Wong and Mooney 2006) to learn a semantic parser and constrain the parsing results with physical limitations imposed by the environment. However, their virtual world is relatively simple with no objects or attribute information as it is constructed from laser sensors. Similarly, Shimizu and Haas (2009) built a system that learns to parse navigation instructions. They restrict the space of possible actions to 15 labels and treat the parsing problem as a sequence labeling problem. This has the advantage that context of the surrounding instructions are taken into account. However, their formal language is very limited in that there are only 15 possible parses for an instruction. There is some recent work that explores direction following in more complex environments. Vogel and Jurafsky (2010) built a learning system for the HCRC Map Task corpus (Anderson et al. 1991) that uses reinforcement learning to learn to navigate from one landmark to another. The environment consists of named locations laid out on a map. Kollar et al. (2010) presented a system that solves the navigation problem for a real office environment. They use LIDAR and camera data collected from a robot to build a semantic map of the world and to simulate navigation. However, both of these systems were directly given object names or required other resources to learn to identify objects in the world. Moreover, both systems used lists of predefined spatial terms. In contrast, we do not assume any existing linguistic knowledge or resource. Besides navigation instructions, there has also been work on learning to interpret other kinds of instructions. Recently, there has been some interest in learning how to interpret English instructions describing how to use a particular website or perform other computer tasks (Branavan et al. 2009; Lau, Drews, and Nichols 2009). These systems learn to predict the correct computer action (pressing a button, choosing a menu item, typing into a text field, etc.) corresponding to each step in the instructions. Our work also fits into the broader area of *grounded language acquisition*, in which language is learned by simply observing its use in some naturally occurring perceptual context (see Mooney (2008) for a review). Unlike most work in statistical NLP which requires annotating large corpora with detailed syntactic and/or semantic markup, this approach tries to learn language without explicit supervision in a manner more analogous to how children acquire language. This approach also grounds the meaning of words and sentences in perception and action instead of arbitrary semantic tokens. One of the core issues in grounded language acquisition is solving the correspondence between language and the semantic context. Various approaches have been used including supervised training (Snyder and Barzilay 2007), iteratively retraining a semantic parser/language generator to disambiguate the context (Kate and Mooney 2007; Chen, Kim, and Mooney 2010), building a generative model of the content selection process (Liang, Jordan, and Klein 2009; Kim and Mooney 2010), and using a ranking approach (Bordes, Usunier, and Weston 2010). Our work differs from these previous approaches in that we explicitly model the relationships between the semantic entities rather than treating them as individual items.

III. BACKGROUND

A. *Logical Semantics*

Logic-based representations of meaning have a long tradition (Montague, 1970; Kamp and Reyle, 1993). They handle many complex semantic phenomena such as relational propositions, logical operators, and quantifiers; however, they can not handle "graded" aspects of meaning in language because they are binary by nature. Also, the logical predicates and relations do not have semantics by themselves without an accompanying ontology, which we want to replace in our semantic parser with distributional semantics. To map a sentence to logical form, we use Boxer (Bos, 2008), a tool for wide-coverage semantic analysis that produces uninterpreted logical forms using Discourse Representation Structures (Kamp and Reyle, 1993). It builds on the C&C CCG parser (Clark and Curran, 2004). Distributional Semantics Distributional models use statistics on contextual data from large corpora to predict semantic similarity of words and phrases (Turney and Pantel, 2010; Mitchell and Lapata, 2010), based on the observation that semantically similar words occur in similar contexts (Landauer and Dumais, 1997; Lund and Burgess, 1996). So words can be represented as vectors in high dimensional spaces generated from the contexts in which they occur. Distributional models capture the graded nature of meaning, but do not adequately capture logical structure (Grefenstette, 2013). It is possible to compute vector representations for larger phrases compositionally from their parts (Landauer and Dumais, 1997; Mitchell and Lapata, 2008; Mitchell and Lapata, 2010; Baroni and Zamparelli, 2010; Grefenstette and Sadrzadeh, 2011). Distributional similarity is usually a mixture of semantic relations, but particular asymmetric similarity measures can, to a certain extent, predict hypernymy and lexical entailment distributionally (Lenci and Benotto, 2012; Kotlerman et al., 2010).

B. *Markov Logic Network*

Markov Logic Network (MLN) (Richardson and Domingos, 2006) is a framework for probabilistic logic that employ weighted formulas in first order logic to compactly encode complex undirected probabilistic graphical models (i.e., Markov networks). Weighting the rules is a way of softening them compared to hard logical constraints. MLNs define a probability distribution over possible worlds, where a world's probability increases exponentially with the total weight of the logical clauses that it satisfies. A variety of inference methods for MLNs have been developed, however, their computational complexity is a fundamental issue.

C. *Probabilistic Soft Logic*

Probabilistic Soft Logic (PSL) is another recently proposed framework for probabilistic logic (Kimmig et al., 2012). It uses logical representations to compactly define large graphical models with continuous variables, and includes methods for performing efficient probabilistic inference for the resulting models. A key distinguishing feature of PSL is that ground atoms have soft, continuous truth values in the interval [0, 1] rather than binary truth values as used in MLNs and most other probabilistic logics. Given a set of weighted inference rules, and with the help of Lukasiewicz's relaxation of the logical operators, PSL builds a graphical model defining a probability distribution over the continuous space of values of the random variables in the model. Then, PSL's MPE inference (Most Probable Explanation) finds the overall interpretation with the maximum probability given a set of evidence. It turns out that this optimization problem is second-order cone program (SOCP) (Kimmig et al., 2012) and can be solved efficiently in polynomial time. Recognizing Textual Entailment Recognizing Textual Entailment (RTE) is the task of determining whether one natural language text, the premise, Entails, Contradicts, or not related (Neutral) to another, the hypothesis.

D. *Semantic Textual Similarity*

Semantic Textual Similarity (STS) is the task of judging the similarity of a pair of sentences on a scale from 1 to 5 (Agirre et al., 2012). Gold standard scores are averaged over multiple human annotations and systems are evaluated using the Pearson correlation between a system's output and gold standard scores.

IV. APPROACH

A semantic parser is three components, a formal language, an ontology, and an inference mechanism. This section explains the details of these components in semantic parser. It also points out the future work related to each part of the system.

A. *Formal Language*

First-order logic Natural sentences are mapped to logical form using Boxer (Bos, 2008), which maps the input sentences into a lexically-based logical form, in which the predicates are words in the sentence. For example, the sentence "A man is driving a car" in logical form is:

$$\exists x, y, z. \; man(x) \wedge agent(y, x) \wedge drive(y) \wedge patient(y, z) \wedge car(z)$$

We call Boxer's output alone an uninterpreted logical form because predicates do not have meaning by themselves. They still need to be connected with an ontology.

742

Future work: While Boxer has wide coverage, additional linguistic phenomena like generalized quantifiers need to be handled.

B. Ontology: on-the-fly ontology

Distributional information is used to generate the needed part of an on-the-fly ontology for the given input sentences. It is encoded in the form of weighted inference rules describing the semantic relations connecting words and phrases in the input sentences. For example, for sentences "A man is driving a car", and "A guy is driving a vehicle", we would like to generate rules like 8x.man(x) , guy(x) |w1 indicating that "man" and "guy" are synonyms with some certainty w1, and 8x. car(x)) vehicle(x) | w2 indicating that "car" is a hyponym of "vehicle" with some certainty w2. Other semantic relations can also be easily encoded as inference rules like antonyms 8x. tall(x) , ¬short(x) |w, contextonymy relation 8x. hospital(x)) 9y. doctor(y) | w. For now, we generate inference rules only as synonyms (Beltagy et al., 2013), but we are experimenting with more types of semantic relations. In (Beltagy et al., 2013), we generate inference rules between all pairs of words and phrases. Given two input sentences T and H, for all pairs (a, b), where a and b are words or phrases of T and H respectively, generate an inference rule: a ! b | w, where the rule's weight w = sim(−!a , −! b), and sim is the cosine of the angle between vectors −!a and −! b . Note that this similarity measure cannot yet distinguish relations like synonymy and hypernymy. Phrases are defined in terms of Boxer's output to be more than one unary atom sharing the same variable like "a little kid" which in logic is little(k) ^ kid(k), or two unary atoms connected by a relation like "a man is driving" which in logic is man(m) ^ agent(d,m) ^ drive(d). We used vector addition (Mitchell and Lapata, 2010) to calculate vectors for phrases. Inference: probabilistic logical inference The last component is probabilistic logical inference. Given the logical form of the input sentences, and the weighted inference rules, we use them to build a probabilistic logic program whose solution is the answer to the target task. A probabilistic logic program consists of the evidence set 9 E, the set of weighted first order logical expressions (rule base RB), and a query Q. Inference is the process of calculating Pr(Q|E,RB). Probabilistic logic frameworks define a probability distribution over all possible worlds. The number of constants in a world depends on the number of the discourse entities in the Boxer output, plus additional constants introduced to handle quantification. Mostly, all constants are combined with all literals, except for rudimentary type checking.

V. CONCLUSION

In this paper an on-the-fly ontology of semantic relations between predicates is derived from distributional information and encoded in the form of soft inference rules in probabilistic logic. We evaluated this approach on two task, RTE and STS, using two probabilistic logics, MLNs and PSL respectively. The semantic parser can be extended in different direction, especially in predicting more complex semantic relations, and enhancing the inference mechanisms.

REFERENCES

[1]. Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In Proceedings of Semantic Evaluation (SemEval-12).

[2]. Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-10).

[3]. Islam Beltagy, Cuong Chau, Gemma Boleda, Dan Garrette, Katrin Erk, and Raymond Mooney. 2013.

[4]. Montague meets Markov: Deep semantics with probabilistic logical form. In Proceedings of the Second Joint Conference on Lexical and Computational Semantics (*SEM-13).

[5]. Islam Beltagy, Katrin Erk, and Raymond Mooney. 2014. Probabilistic soft logic for semantic textual similarity. In Proceedings of Association for Computational Linguistics (ACL-14).

[6]. Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-13).

[7]. Johan Bos. 2008. Wide-coverage semantic analysis with Boxer. In Proceedings of Semantics in Text Processing (STEP-08).

[8]. Stephen Clark and James R. Curran. 2004. Parsing the WSJ using CCG and log-linear models. In Proceedings of Association for Computational Linguistics (ACL-04).

[9]. Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In Proceedings of North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-13).

[10]. L. Getoor and B. Taskar, editors. 2007. Introduction to Statistical Relational Learning. MIT Press, Cambridge, MA.