Implementation of Reduction of Ambiguity due to Synonyms in Punjabi Language

Navdeep Kaur[1]
Student of master Technology
Department of CSE
Desh Bhagat University
Mandi Gobindgarh, Punjab, India
navdeep.panag384@gmail.com

Vandana Pushe[2]
Assistant Professor
Department of CSE
Desh Bhagat University
Mandi Gobindgarh, Punjab, India
vandanapushe@gmail.com

Abstract— ***We present a probabilistic generative model for learning semantic parsers from ambiguous supervision. Our approach learns from natural language sentences paired with world states consisting of multiple potential logical meaning representations. It disambiguates the meaning of each sentence while simultaneously learning a semantic parser that maps sentences into logical form. Compared to a previous generative model for semantic alignment, it also supports full semantic parsing.***

Keywords— ***reranking, syntactic parsing, semantic parsing, semantic role labeling, named entity recognition***

1. INTRODUCTION

Building a computer system that can understand human languages has been one of the long-standing goals of artificial intelligence. Currently, most state-of-the-art natural language processing (NLP) systems use statistical machine learning methods to extract linguistic knowledge from large, annotated corpora. However, constructing such corpora can be expensive and time-consuming due to the expertise it requires to annotate such data. In this thesis, we explore alternative ways of learning which do not rely on direct human supervision. In particular, we draw our inspirations from the fact that humans are able to learn language through exposure to linguistic inputs in the context of a rich, relevant, perceptual environment. We first present a system that learned to sportscast for RoboCup simulation games by observing how humans commentate a game. Using the simple assumption that people generally talk about events that have just occurred, we pair each textual comment with a set of events that it could be referring to. By applying an EM-like algorithm, the system simultaneously learns a grounded language model and aligns each description to the corresponding event. The system does not use any prior language knowledge and was able to learn to sportscast in both English and Korean. Human evaluations of the generated commentaries indicate they are of reasonable quality and in some cases even on par with those produced by humans.

For the sportscasting task, while each comment could be aligned to one of several events, the level of ambiguity was low enough that we could enumerate all the possible alignments. However, it is not always possible to restrict the set of possible alignments to such limited numbers. Thus, we present another system that allows each sentence to be aligned to one of exponentially many connected subgraphs without explicitly enumerating them. The system first learns a lexicon and uses it to prune the nodes in the graph that are unrelated to the words in the sentence. By only observing how humans follow navigation instructions, the system was able to infer the corresponding hidden navigation plans and parse previously unseen instructions in new environments for both English and Chinese data with the rise in popularity of crowdsourcing, we also present results on collecting additional training data using Amazon's Mechanical Turk. Since our system only needs supervision in the form of language being used in relevant contexts, it is easy for virtually anyone to contribute to the training data. Being able to communicate with a computer in human languages is one of the ultimate goals of artificial intelligence (AI) research. Instead of learning special commands or control sequences (e.g. a series of mouse clicks, typing, or gestures), we could articulate what we want in our own words. In response, the computer could also present information to us or ask questions verbally without those responses having been programmed into the system. In order to achieve this goal, there are two tasks the computer must become competent at: the ability to interpret human languages and the ability to generate coherent natural language content.

The choice of the representation language depends on the specific application domain and can range from predicate logic to SQL statements to any other formal language that supports automated reasoning. There are typically two parts to building a semantic parser. One is building a lexicon that defines the meanings of words or short phrases. The other part is building compositional rules that successively combine smaller meaning representations into larger, coherent representations of complete sentences.

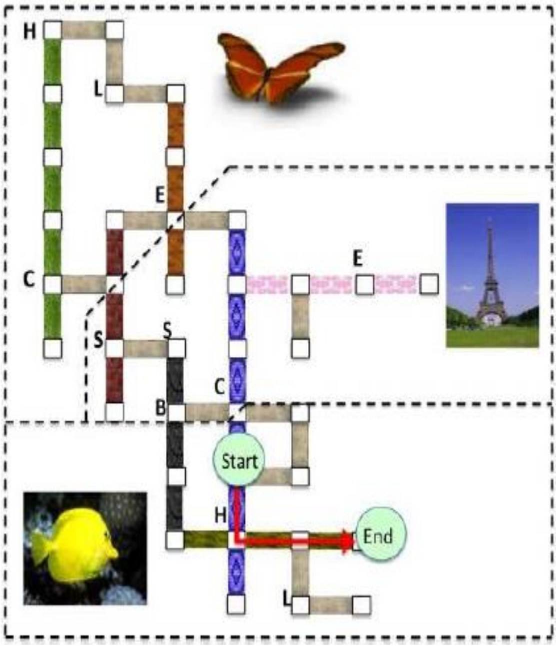


Figure 1: This is an example of a route in our virtual world. The world consists of interconnecting hallways with varying floor tiles and paintings on the wall (butterfly, fish, or Eiffel Tower.) Letters indicate objects (e.g. 'C' is a chair) at a location.

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).

2. BACKGROUND

This section describes existing models and algorithms employed in the current research. Our model is built on top of the generative semantic parsing model developed by Lu et al. (2008). After learning a probabilistic alignment and parsing model, we also used the WASP and WASP −1 systems to produce additional parsing and generation results. In particular, since our current system is incapable of effectively generating NL sentences from MR logical forms, in order to demonstrate how our matching results can aid NL generation, we use WASP −1 to learn a generator. This follows the experimental scheme of Chen et al. (2010), which demonstrated that an improved NL–MR matching from Liang et al. (2009) results in better overall parsing and generation. Finally, our overall generative model uses the IGSL (Iterative Generation Strategy Learning) method of Chen and Mooney (2008) to initially estimate the prior probability of each event-type generating a natural-language comment.

3. 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 semanticmarkup, 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 andMooney 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.

4. 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. **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:

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

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.

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

```
Input: A set of training examples ($e_i$; $y_i$ ),
where $e_i$ is a NL word and $y_i$ =arg max$_y$ belongs to GEN($e_i$) EXEC(y)
Output: The parameter vector W , averaged over all iterations 1:::T
procedure PERCEPTRON
Initialize _W = 0
for t = 1….T; i = 1….n do
    $y_i$ = arg max$_y$ belongs to GEN($e_i$) _($e_i$; y) _ _W
    if $y_i$ = $y_i$ then
        W = W + ($e_i$; $y_i$ ) != phi($e_i$; $y_i$)
    end if
    end for
end procedure
```

5. Implementation of the System

In this section the results generation and the implementation phase in the form of GUI is represented. In the figure defined below this is a system that is representing the GUI part of the research and is defined below.

5.1 Results

It represents the GUI(**Graphical User Interface**) part of the research system.

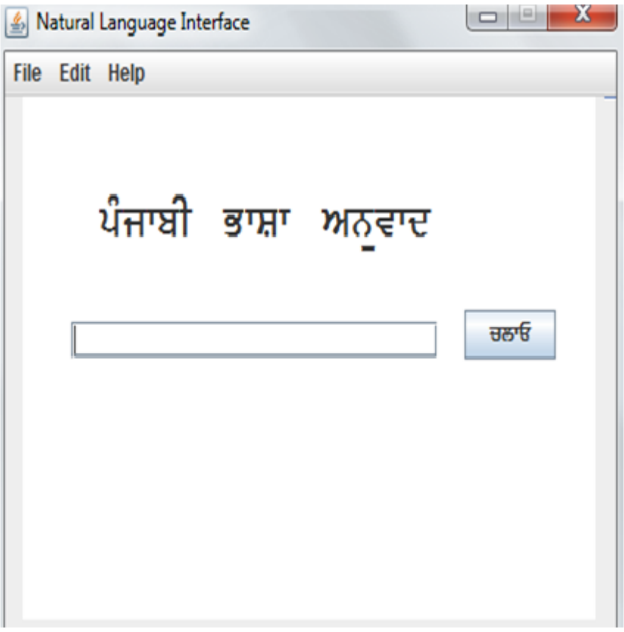
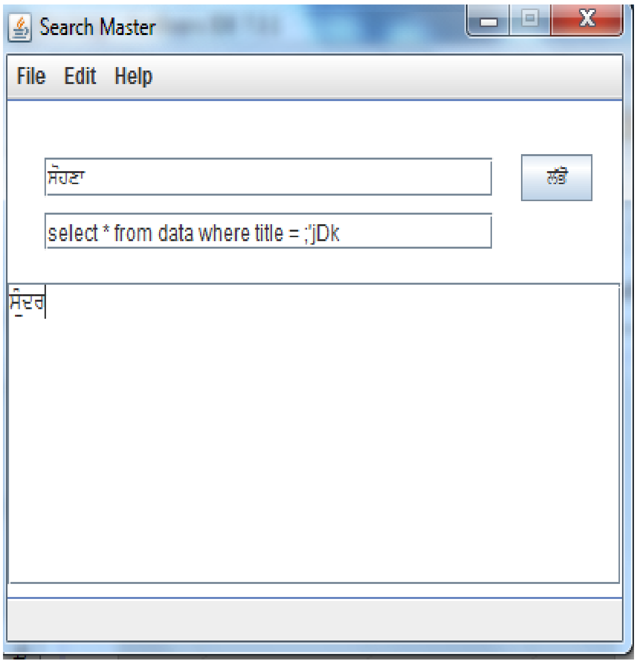


Figure 1 Graphical User Interface

The above defined figure represents the GUI part of the research system, In this part a query is feeded into the system and then it is converted into tokens from which the processing is done to check which word is of common in data base and which is to be defined separately.
In this figure the query is executed and result is defined.

In the figure 2 defined below a query string is defined called and the output is generated .



ਸੋਹਣਾ

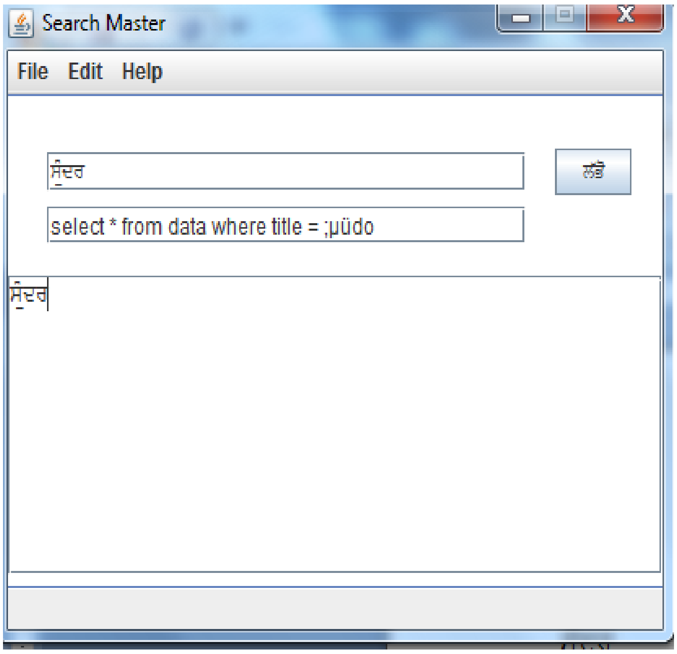
select * from data where title = ;'jDk

ਸੁੰਦਰ

Figure 2 Example of a Word

In the figure 2 that is shown above is the result of the query that is generated and the result that is produced by the query. In the above defined figure the query is displayed in the gurmukhi script and the word is entered which is a synonym of some of the words. Now all the synonyms of the above defined words will produce same results as this word is producing.

This figure 3 represented the ambiguity reduction due to synonym of the system against the query string **.**



ਸੁੰਦਰ

select * from data where title = ;µüdo

ਸੁੰਦਰ

Figure 3 Example of synonym

In the figure 3 defined above the synonym of the word that is represented in figure 2 is generated and the proposed system is generating the same result as the defined in the figure 2. The two words that may be used as query are synonym of one another and producing same results that helped in reducing the space complexity of the system**.**

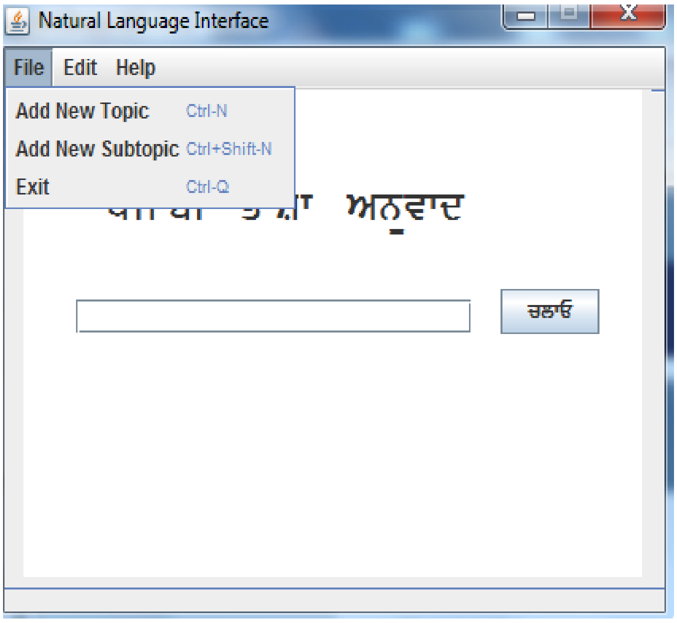


Figure 4 Result of adding a new word

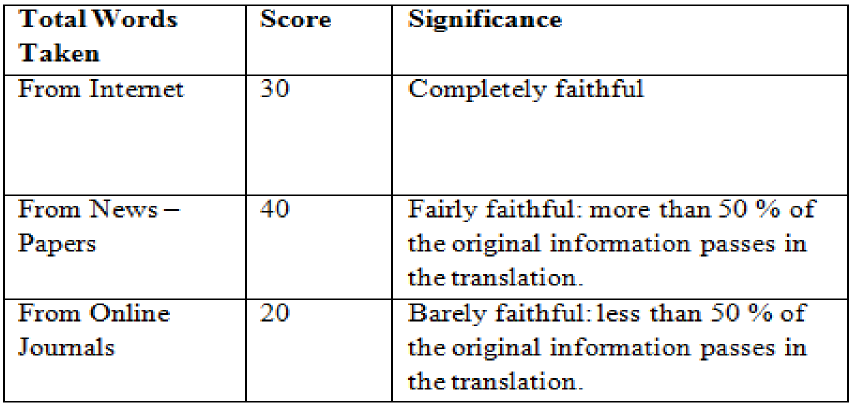
Figure 4 is the representation to add a new word in the dictionary of the system in gurmukhi script in punjabi language.

6. TESTING OF THE SYSTEM

In this figure 6.1 is the representation of the accuracy of the two systems. Existing one is drawn from the literature survey and proposed one the system that is developed now. The accuracy produced by the existing system is approx 98.6% and proposed is 99%.
For Testing the system we have take help some internet material ,newspapers and online journals or online news and books in Punjabi. Testing of single characters which gives 98% output. And words 99% accuracy.

6.1 Word Testing

In word testing we take words from articles, blog, news and literature. And finally get the output. We get the very completely faithful output for the synonym of “Punjabi Language”. Different words are compared and then try to overcome the problem of synonym.



Total Words Taken	Score	Significance
From Internet	30	Completely faithful
From News – Papers	40	Fairly faithful: more than 50 % of the original information passes in the translation.
From Online Journals	20	Barely faithful: less than 50 % of the original information passes in the translation.

Table 1 Word Testing

7. CONCLUSION

Learning the semantics of language from the perceptual context in which it is uttered is a useful approach because only minimal human supervision is required. Instead of annotating each sentence with its correct semantic representation, the human teacher only has to demonstrate to the system how to use language in a relevant context. However, resolving the ambiguity of which parts of the perceptual context are being referred to can be a difficult problem. In this thesis, we have looked at a couple of frameworks aimed to solve this problem. The first system uses an EM-like retraining loop that alternates between building a semantic model of the language and estimating the mostly likely alignments between NL sentences and MRs. We demonstrated the feasibility of this system by applying it to a sportscasting task where the training data consist of textual commentaries and streams of automatically extracted events from simulated RoboCup games. We evaluated several scoring functions for disambiguating the training data in order to learn semantic parsers and language generators. Using a generation evaluation metric as the criterion for selecting the best NL–MR pairs produced better results than using semantic parsing scores when the initial training data were very noisy. Our system also learned a simple model for content selection from the ambiguous training data by estimating the probability that each event type evokes human commentary. Experimental evaluation verified that the overall system learned to accurately parse and generate comments as well as generate complete play-by-play sportscasts that are competitive with those produced by humans. We achieved similar results learning to sportscast in Gurmukhi Script in Punjabi. We compare so many words and get different results and then with my present work in natural language processing I remove the problem of synonyms in Punjabi language.

8. FUTURE SCOPE

In the future scope the accuracy of the system in the gurmukhi script in Punjabi language can be improved. The ambiguity of the previous system can be reduced so that the synonym in a language can-not produce different results.
Prediction ability of the research system can be improved. In the prediction ability of the system the synonym will be able to produce the accurate results after executing the query. In future, quality can be improved to increase the size of corups and also to add more Indian languages.

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.