ISSN No:-2456-2165

Analysis of Statistical Parsing in Natural Language Processing

Krishna Karoo
Research Scholar, Department of Electronics & Computer Science, R.T.M. Nagpur University, Nagpur

Dr Girish Katkar
Head of Department, Department of Computer Science
Taywade College Koradi Dist:- Nagpur

Abstract:- A statistical language model is a probability distribution P(s] over all possible word sequences (or any other linguistic unit like words, sentences, paragraphs, documents, or spoken utterances). A number of statistical language models have been proposed in literature. The dominant approach in statistical language modeling is the n-gram model.

I. INTRODUCTION

A. n-gram Model

The goal of a statistical language model is to estimate the probability (likelihood) of a sentence. This is achieved by decomposing sentence probability into a product of conditional probabilities using the chain rule as follows:

P(s)= P(w1, w2, w3,.....,wn)
P(s)= P(w1) P(w2/w1) P(w3/w1 w2)..... P(wn/w1 w2 .....wn-1)

$= \prod_{i=1}^{n} P\left(\frac{wi}{hi}\right)$ Language Processing and Information Retrieval

Where h, is history of word w, defined as W_1 W2... W_{i_1}

Example-1

- *Training set*

The Arabian Knights
These are the fairy tales of the east
The stories of the Arabian knights are translated in many languages

- *Bi-gram model*

P(the/<s>) = 0.67 P(Arabian/the) = 0.4 P(knights /Arabian) = 1.0
P(are/these) = 1.0 P(the/are) = 0.5 P(fairy/the) = 0.2
P(tales/fairy) = 1.0 P(of/tales) = 1.0 P(the/of) = 1.0
P(east/the) = 0.2 P(stories/the) = 0.2 P(of/stories) - 1.0
P(are/knights) = 1.0 P(translated/are) = 0.5 *P(in* /translated) = 1.0
^(many/in) = 1.0
P(languages/many) = 1.0

Test sentence (s): The Arabian knights are the fairy tales of the east.

P(The/<s>) x P(Arabian/the) x P(Knights/Arabian) x P(are/knights)
x P(the/are) x P(fairy/the) x P(tales/fairy) x P(of/tales) x P(the/of)
x P(east/the)
= 0.67 x 0.4 x 1.0 x 1.0 x 0.5 x 0.2 x 1.0 x 1.0 x 1.0 x 0.2
=0.0268

As each probability is necessarily less than 1, multiplying the probabilities might cause a numerical underflow, particularly in long sentences. To avoid this, calculations are made in log space, where a calculation corresponds to adding log of individual probabilities and taking antilog of the sum.

B. Add-one Smoothing

This is the simplest smoothing technique. It adds a value of one to each n-gram frequency before normalizing them into probabilities. In general, add-one smoothing is not considered a good smoothing technique. It assigns the same probability to all missing n-grams, even though some of them could be more intuitively appealing than others. Gale and Church (1994) reported that variance of the counts produced by the add-one smoothing is worse than the unsmoothed MLE method. Another problem with this technique is that it shifts too much of the probability mass towards the unseen n-grams (n-grams with 0 probabilities) as there number is usually quite large, Good-Turing smoothing (Good 1953 attempts to improve the situation by looking at the number of n-grams with a high frequency in order to estimate the probability mass that needs to be assigned to missing or low-frequency n-grams.

C. Good-Turing Smoothing

Good-Turing smoothing (Good 1953) adjusts the frequency *faf am* re-gram using the count of re-grams having a frequency of occurrence /+!-It converts the frequency of an re-gram from *ftof** using the following expression:

$$^{n}f+I \quad /\bullet=(/+!)\text{-}\pounds\text{-}^{n}f$$

where *ni* is the number of re-grams that occur exactly / times in the training corpus. As an example, consider that the number of re-grams that occur 4 times is 25,108 and the number of re-grams that occur 5 times is 20,542. Then, the smoothed count for 4 will be H*™

D. Caching Technique

Another improvement over basic re-gram model is caching. The frequency of re-gram is not uniform across the text segments or corpus. Certain words occur more frequently in certain segments (or documents) and rarely in others. For example, in this section, the frequency of the word 're-gram' is high, whereas it occurs rarely in earlier sections. The basic re-gram model ignores this sort of variation of re-gram frequency. The cache model combines the most recent re-gram frequency with the standard re-gram model to improve its performance locally. The underlying assumption here is that the recently discovered words are more likely to be repeated.

NN	noun	student,	chair,	proof,	mechanism
VB	verb	study,	increase,	produces	
ADJ	adjective	large,	high,	tall,	few,
JJ	adverb	carefully,	slowly,	uniformly	
IN	preposition	ir.,	on,	to,	of
PRP	pronoun	I,	me,	they	
DET	determiner	the,	a,	an,	this, those

Fig 1:- Part-of-speech example

II. PART-OF-SPEECH TAGGING

Part-of-speech tagging is the process of assigning a part-of-speech (such as a noun, verb, pronoun, preposition, adverb, and adjective), to each word in a sentence. The input to a tagging algorithm is the sequence of words of a natural language sentence and specified tag sets (a finite list of part-of-speech tags). The output is a single best part-of-speech tag for each word. Many words may belong to more than one lexical category. For example, the English word 'book' can be a noun as in '/ *am reading a good book*[1] or a verb as in *'Thepolice booked the snatcher*\ The same is true for other languages. For example, the Hindi word *'soan'* may mean 'gold' (noun) or 'sleep' (verb). However, only one of the possible meanings is used at a time. In tagging, we try to determine the correct lexical category of a word in its context. No tagger is efficient enough to identify the correct lexical category of each word in a sentence in every case. The tag assigned by a tagger is the most likely for a particular use of word in a sentence.

A. Hybrid taggers

Hybrid taggers combine features of both these approaches. Like rule-based systems, they use rules to specify tags. like stochastic systems, they use machine-learning to induce rules from a tagged training corpus automatically. The transformation-based tagger or Brill tagger is an example of the hybrid approach.

B. Rule-based Tagger

Most rule-based taggers have a two-stage architecture. The first stage is simply a dictionary look-up procedure, which returns a set of potential tags (parts-of-speech) and appropriate syntactic features for each word. The second stage uses a set of hand-coded rules to discard contextually illegitimate tags to get a single part-of-speech for each word. For example, consider the noun-verb ambiguity in the following sentence: *The show must go on.*

C. Stochastic Tagger

The standard stochastic tagger algorithm is the HMM tagger. A Markov model applies the simplifying assumption that the probability of a chain of symbols can be approximated in terms of its parts or w-grams. The simplest n-gram model is the unigram model, which assigns the most likely tag (part-of-speech) to each token.

D. Hybrid Taggers

Hybrid approaches to tagging combine the features of both the rule-based and stochastic approaches. They use rules to assign tags to words. Like the stochastic taggers, this is a machine learning technique and rules are automatically induced from the data. Transformation-based learning (TBL) of tags, also known as Brill tagging, is an example of hybrid approach. TBL is a machine learning method introduced by E. Brill (in 1995). Transformation-based error-driven

ISSN No:-2456-2165

learning has been applied to a number of natural language problems, including part-of-speech tagging, speech generation, and syntactic parsing (Brill 1993, 1994, Huang et al. 1994).

III. WORD SENSE DISAMBIGUATION

Having discussed various types of ambiguities we now focus on identifying the correct sense of words in a particular use. The first attempt at automatic sense disambiguation was made in the context of machine translation. The famous *Memorandum*, Weaver (1949) discusses the need for word sense disambiguation (WSD) in machine translation, and outlines an approach to WSD, which underlies all subsequent work on the topic.

A. Selectional Restriction-based Word Sense Disambiguation

Selectional restrictions or preferences can be used in parsing to eliminate flawed meaning representations. This can be viewed as a form of indirect word sense disambiguation. We now explore this idea. Consider the following sentences:

The institute will *employ* new employees, ('to hire')
The committee *employed* her proposal, ('to accept')

One can intuitively differentiate the senses of *employ in* sentences (a) and (b) with the complements of each *employ*. To be more precise, *employ* in (a) restricts its subject and object nouns to those associated with the semantic features human/organization and human, respectively. On the other hand, *employ* in (b) restricts its subject and object nouns to those associated with the semantic features human/organization and idea, respectively. Consequently, given employees as the object, the sense to hire is selected as the interpretation of employ in (a), and the sense *to accept* is ruled out. The same reasoning can be used to select the sense *to accept* as the interpretation of *employ* in (b).

B. Context-based Word Sense Disambiguation Approaches

Approaches to stand-alone WSD that make use of context of ambiguous word basically fall into one of the following two general categories:

- Knowledge-based
- Corpus-based

IV. BAYESIAN CLASSIFICATION

The specific algorithm we describe here was introduced by Gale (1992). The classifier assumes that we have a corpus in which each occurrence of an ambiguous word is labelled with its correct sense. The words around the ambiguous word are used to define a context window. The classifier treats the context of word w as a bag of words without structure. No feature selection is done. All the words occurring in the context window contribute in deciding which sense of the ambiguous word is likely to be used with it. What we'' want to find is the most likely sense s^f for an input context c of an ambiguous word w. This is obtained as

$$\$' = \arg\max P(s_k/c]h$$

As it is difficult to collect statistics for this equation, we apply the Bayesian formula to compute it.

A. Bootstrapping

The Bayes classifier attempts to combine evidence from all words in the context window to help disambiguation. This requires a large sense tagged training set to collect evidences. Hearst (1991) proposed the bootstrapping approach to eliminate the need for a large training set. The bootstrapping method relies on a relatively small number of instances labeled with senses having a high degree of confidence. This could be accomplished by manually tagging those instances of an ambiguous word for which the sense is clear (Hearst 1991). These labeled instances are used as seeds to train an initial classifier. The classifier is then used to extract more training instances from the remaining untagged corpus. As the process is repeated, the training corpus grows and the numbers of untagged instances are reduced. The iteration continues until the remaining untagged corpus is empty or no new instance can be annotated.

B. Bilingual Corpora

A bilingual corpus consists of two corpora, one of which is a translation of the other. As different senses of an ambiguous word often translate differently in another language, a bilingual corpus can be used for disambiguating word senses. For example, the Hindi word cp<*i<H is translated as *pen* in the writing sense and *graft* in the transplant sense. Gale et al. (1992b, 1993) use the bilingual Hansard corpus to avoid manual sense tagging of a corpus. The Handsard corpus consists of transcriptions in French and English of the proceedings of the Canadian parliament. They first automatically aligned the bilingual corpus and then tagged the words of the aligned corpus using the basic assumptions that translations of a word reflect the senses of that word.

V. UNSUPERVISED METHODS OF WSD

Unsupervised methods of WSD eliminate the need for sense tagged training data. Instead, these approaches take feature-value representations of unlabelled contexts (instances) and group them into clusters. Each cluster can be assumed to represent one sense of an ambiguous word. These clusters can be represented as the average of their constituent feature vectors. Unknown instances are classified as having the sense of the cluster to which they are closest according to the similarity measure. Strictly speaking, using a completely Unsupervised sense disambiguation task, we can only discriminate word senses. That is, we can group together instances of a word used in different senses without knowing what those senses are. However, Yarowsky (1995) proposed an Unsupervised algorithm that can accurately disambiguate word senses in a large completely untagged corpus. He exploited two powerful properties of human language in an iterative bootstrapping setup to avoid the need of manually tagged training data (adapted from Yarowsky 1995):

- One sense per discourse: The sense of a target word is highly consistent within any given document or discourse.
- One sense per collocation: Nearby words provide strong and consistent clues to the sense of a target word, conditional on relative distance, order and syntactic relationship.

A. Knowledge Sources in WSD

A variety of information, including syntactic (part-of-speech, grammatical structure), semantic (selection restriction) and pragmatic (topics) information as well as dictionary (definitions), and corpus (collocation) specific information, can be utilized as a knowledge source in WSD. Here is a list of some of the information sources deemed useful in disambiguation.

- *Context of a word*

The context of a word can be regarded as the words surrounding the ambiguous word. A word only can be disambiguated in its context. The context is therefore useful in determining the meaning of a word in a particular usage.

➢ *Frequency of a sense*

This information is generally used in statistical approaches to measure the likelihood of each possible sense. Usually, this statistics is gathered over some sense-tagged corpus.

➢ *Part-of-speech*

Part-of-speech information can reduce the number of possible senses a word can have. For example, in WordNet 2.0 *bitter* has 3 senses as noun, 7 senses as adjective, and one sense as a verb. The use of *bitter* as a verb does not lead to ambiguity.

➢ *Collocations*

These may provide useful information about the sense of a word. For instance, the noun *match* has 9 senses listed in WordNet but only one of these applies to football match.

➢ *Selectional preferences*

Semantic restrictions that predicates place on their argument can be used for disambiguation. For instance, *eat* in the *have a meal* sense prefers humans as subjects. This knowledge is similar to the argument-head relation, but selectional preferences are given in terms of semantic classes, instead of plain words.

➢ *Domain*

In a particular domain, only one sense of a word is likely to be used. Thus, information about domain furnishes useful information for disambiguation. For example, in the domain of sports, the *cricket bat* sense of *bat* is preferred.

Besides these, thematic role of a word (subject or object), sentence structure, semantic word properties, and pragmatic information may also be utilized in sense disambiguation. All this information can be used together with general knowledge about the situation to rule out impossible readings.

B. Applications of WSD

Word sense disambiguation (WSD) is only an intermediate task in NLP, like POS tagging or parsing. Accurate WSD is important for many applications, e.g., machine translation and information retrieval.

One of the first applications of WSD was machine translation, for which, disambiguating the sense of a source language word is crucial for accurately selecting its translation equivalent in the target language. The Hindi word ^PcJT, for example, can either have the sense of the English *word fruit*, or the sense of Mfeuurft (result). In order to correctly translate a text containing cpaf, we need to know which sense is intended.

C. WSD Evaluation

Evaluation is important in all NLP tasks. It has always been a problem in disambiguation research, as the only way to judge the performance of a disambiguator is to manually check its output. Manual checking is time consuming and because of this, most disambiguators have been evaluated only on a small number of words. The SENSEVAL initiatives have simplified the evaluation task. The basic metric used for evaluating word sense disambiguation algorithm is precision and recall. Precision measures the fraction of correctly tagged instances in the total set. This requires access to an annotated corpus. Two such corpuses are now available: the SEMCOR (Landes et al. 1998) corpus and SENSEVAL (Kilgariff and Rosenzweig 2000) corpus. These metrics fail to give any credit to an algorithm that makes only broad distinctions between senses, as they consider sense match to be exact. Some metrics have been proposed to give partial credit to instances where a broader sense is selected.

VI. CONCLUSION

Semantic analysis is concerned with meaning representation of linguistic inputs. A meaning representation bridges the gap between linguistic and commonsense knowledge. A meaning representation language must be verifiable and unambiguous. It should support the use of variable and inferencing and must be expressive enough to handle the wide variety of content found in natural language. Syntax driven semantic analysis uses the syntactic constituents of a sentence to build its meaning representation. Semantic grammar provides an alternative way for creating meaning representation. Word sense disambiguation is concerned with identifying the correct sense of a word. The knowledge sources used by word sense disambiguation algorithms include context of word, sense frequency, selectional preferences, collocation and domain.

REFERENCES

[1]. Bourns, G. 1987. A Unification-based Analysis of Unbounded Dependencies in Categorial Grammar, in J.Groenendijk, M. Stokhof, & F. Veltman (eds.) *Proceedings of the sixth Amsterdam Colloquium*, University of Amsterdam, Amsterdam, 1-19.

[2]. Bourns, G., 1988, Modifiers and Specifiers in Categorial Unification Grammar, *Linguistics*, vol 26, 21-46.Bourns, G., E. KSnig, & H. Uszkoreit, 1988. A Flexible Graph-Unification Formalism and its Application to Natural Language Processing, *IBM Journal of Research and Development*, 32, 170-184

[3]. Calder, J., E. Klein, & H. Zeevat 1988. Unification Categoriai Grammar: a concise, extendable grammar for natural language processing. *Proceedings of Coling 1988, Hungarian Academy of Sciences*, Budapest, 83-86.

[4]. 4. Haas, A. 1989. A Parsing Algorithm for Unification Grammar. *Computational Linguistics* 15-4, 219-232.

[5]. Karttunen, L. 1989. Radical Lexicalism. In M. Baltin & A. Kroch (eds.), *Alternative Conceptions of Phrase Structure*, Chicago University Press, Chicago, 43-66.

[6]. Matsumoto, Y., H. Tanaka, H. Hirakawa, II. Miyoshi,& H. Yasukawa, 1983, BUP : A Bottom-Up Parser embedded in Prolog. *New Generation Computing, vol 1*,145-158.

[7]. Pereira, F., & S. Shieber (1986). *Proiog and Natural Language Analysis.* CSLI Lecture Notes 10, University of Chicago Press, Chicago.Pollard, C. • I. Sag, 1987, *Information-Based Syntax and Semantics, vol 1 : Fundamentals*, CSLI Lecture Notes 13, University of Chicago Press,

[8]. Chicago.Shieber, S. 1985. Using Restriction to Extend Parsing Algorithms for Complex-Feature-Based Algorithms.*Proceedings of the g2nd Annual Meeting of the Association for Computational Linguistics*, University of Chicago, Chicago, 145-152.

[9]. Uszkoreit, H. 1986. Categorial Unification Grammars.*Proceedings of COLING 1985*. Institute fiir angewandte Kommunikations- und Sprachforschung, Bonn, 187-194.

[10]. Zeevat, H., E. Klein, & J. Calder, 1987. An Introduction to Unification Categorial Grammar. In N. Haddock,E. Klein, & G. Morill (eds.), *Categorial Grammar,Unification grammar, and Parsing*, Edinburgh Working Papers in Cognitive Science, Vol. 1.Zwicky, A. 1986. German Adjective Agreement in GPSG. *Linguistics*, vol 24,957-990.

[11]. Dagan, I., Glickman, O. 2004 Generic applied modeling of language variability *In Proceedings of PASCAL Workshop on Learning Methods for Text Understanding and Mining* Grenoble.

[12]. Lin, D. 1998. Dependency-based evaluation of MINIPAR. *In Proceedings of the Workshop on Evaluation of Parsing Systems at LREC-98.* Granada, Spain.Lin, D. and Pantel, P. 2001. Discovery of inference rules for Question Answering. *Natural Language Engineering,7(4)*, pages 343-360.

[13]. Monz, C. and de Rijke, M. 2001. Light-Weight Entailment Checking for Computational Semantics. *The third workshop on inference in computational semantics (ICoS-3)*

[14]. Punyakanok., V.,Roth, D. and Yih, W., 2004 Mapping Dependencies Trees: An Application to Question Answering *Proceedings of AI & Math 2004* Ratnaparkhi, A. 1996 A Maximum Entropy Part-Of-Speech Tagger. *In proceeding of the Empirical Methods in Natural Language Processing Conference, May 17-18, 1996*

[15]. Szpektor I., Tanev H., Dagan I., and Coppola B. 2004 Scaling Web-based Acquisition of Entailment Relations *In Proceedings of EMNLP-04 - Empirical Methods in Natural Language Processing, Barcelona, July 2004*

[16]. K. Zhang K., Shasha D. 1990 Fast algorithm for the unit cost editing distance between trees. *Journal of algorithms, vol. 11, p. 1245-1262, December 1990.*

[17]. Asher, Nicholas. 1993. *Reference to Abstract Objects in Discourse.* Kluwer Academic Publishers, Dordrecht.Ballard, D. Lee, Robert Conrad, and Robert E. Longacre. 1971. The deep and surface grammar of interclausal relations. *Foundations of language*, 4:70-118.

[18]. Cahn, Janet. 1992. An investigation into the correlation of cue phrases, unfilled pauses and the structuring of spoken discourse. In *Proceedings of the IRCS Workshop on Prosody in Natural Speech*, pages 19-30.

[19]. Cohen, Robin. 1987. Analyzing the structure of argumentative discourse. *Computational Linguistics, 13 (1*-2): 11-24, January-June.Costermans, Jean and Michel Fayol. 1997. *Processing lnterclausal Relationships. Studies in the Production and Comprehension of Text.* Lawrence Erlbaum Associates,Publishers.Cumming, Carmen and Catherine McKercher. 1994.*The Canadian Reporter: News writing and reporting.*

[20]. Delin, Judy L. and Jon Oberlander. 1992. Aspectswitching and subordination: the role of/t-clefts in discourse.In *Proceedings of the Fourteenth International Conference on Computational Linguistics (COLING-92)*, pages 281-287, Nantes, France, August 23-28.Fraser, Bruce. 1996. Pragmatic markers. *Pragmatics*,6(2): 167-190.

[21]. Grosz, Barbara J., Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*,21 (2):203-226, June.

[22]. Grosz, Barbara J. and Candace L. Sidner. 1986. Attention,intentions, and the structure of discourse. *Computational Linguistics*, 12(3): 175-204, July-September.Grover, Claire, Chris Brew, Suresh Manandhar, and Marc Moens. 1994. Priority union and generalization in discourse grammars. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL-94)*, pages 17-24, Las Cruces, June 27-30.

[23]. HaUiday, Michael A.K. and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman. Harabagiu, Sanda M. and Dan I. Moldovan. 1995. A marker-propagation algorithm for text coherence. In *Working Notes of the Workshop on Parallel Processing in Artificial Intelligence*, pages 76-86, Montreal,Canada, August.

[24]. Hirschberg, Julia and Diane Litman. 1993. Empirical studies on the disambiguation of cue phrases. *Computational Linguistics*, 19(3):501-530.

[25]. Hobbs, Jerry R. 1990. *Literature and Cognition*. CSLI Lecture Notes Number 21. Kamp, Hand and Uwe Reyle. 1993. *From Discourse to Logic: Introduction to ModelTheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic Publishers,London, Boston, Dordrecht. Studies in Linguistics and Philosophy, Volume 42.

[26]. Kintsch, Walter. 1977. On comprehending stories. In Marcel Just and Patricia Carpenter, editors, *Cognitive processes in comprehension*. Erlbaum, Hillsdale, New Jersey.

[27]. Knott, Alistair. 1995. *A Data-Driven Methodology for Motivating a Set of Coherence Relations*. Ph.D. thesis, University of Edinburgh.Lascarides, Alex and Nicholas Asher. 1993. Temporal interpretation, discourse relations, and common sense entailment. *Linguistics and Philosophy*, 16(5):437-493.

[28]. Lascarides, Alex, Nicholas Asher, and Jon Oberlander. 1992. Inferring discourse relations in context. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics (ACL-92)*, pages 1-8.Longacre, Robert E. 1983. *The Grammar of Discourse*Plenum Press, New York.

[29]. Mann, William C. and Sandra A. Thompson. 1988.Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243-281.

[30]. Marcu, Daniel. 1996. Building up rhetorical structure trees. In *Proceedings of the Thirteenth National Conference on Artificial intelligence (AAA1-96)*, volume 2, pages 1069-1074, Portland, Oregon, August 4-8,.

[31]. Marcu, Daniel. 1997. *The rhetorical parsing, summarization,and generation of natural language texts*.Ph.D. thesis, Department of Computer Science, University of Toronto, Forthcoming.

[32]. Martin, James R. 1992. *English Text. System and Structure*.John Benjamin Publishing Company, Philadelphia/Amsterdam.

[33]. Moens, Marc and Mark Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics*,14(2): 15-28.

[34]. Moser, Megan and Johanna D. Moore. 1997. On the correlation of cues with discourse structure: Results from a corpus study. Submitted for publication.Polanyi, Livia. 1988. A formal model of the structure of discourse. *Journal of Pragmatics*, 12:601-638.

[35]. Prost, H., R. Scha, and M. van den Berg. 1994. Discourse grammar and verb phrase anaphora. *Linguistics and Philosophy*, 17(3):261-327, June.

[36]. Redeker, Gisela 1990. Ideational and pragmatic markers of discourse, structure. *Journal ofPragmatics*, 14:367-381.

[37]. Sanders, Ted J.M., Wilbert P.M. Spooren, and Leo G.M. Noordman. 1992. Toward a taxonomy of coherence relations. *Discourse Processes*, 15:1-35.Schiffrin, Deborah. 1987. *Discourse Markers. Cam*bridge University Press.