Unit-III

Semantic Parsing

- Introduction
- Semantic Interpretation
- System Paradigms
- Word Sense

Semantic Parsing

- Semantics is the study of meaning and parsing is the examination of the meaning in a minute way.
- Semantic parsing is the process of identifying meaning chunks contained in an information signal in an attempt to transform it so that it can be manipulated by a computer program to perform higher level tasks.
- In the current case the information signal is human language text.
- The term semantic parsing has been used by researchers to represent various levels of granularity of meaning representations.

Introduction

- Research in language understanding is the identification of a meaning representation that is detailed enough to allow reasoning systems to make deductions.
- It is general enough that it can be used across many domains with little to no adaptation.
- Two approaches have emerged in natural language processing for language understanding.
- In the first approach a specific rich meaning representation is created for a limited domain for use by applications that are restricted to that domain.
- Example: air travel reservations, football game simulations, querying a geographic database.

Introduction

- In the second approach a related set of intermediate meaning representations is created from low level, to midlevel and the bigger understanding task is divided into multiple smaller pieces that are more manageable such as word sense disambiguation.
- In the first approach the meaning representations are tied to a specific domain.
- In the second approach the meaning representations cover the overall meaning.
- We do not yet have a detailed overall representation that would cover across domains.

Introduction

- Here we treat the world as though it has exactly two types of meaning representations:
  - A domain dependent deeper representation (deep semantic parsing)
  - A set of relatively shallow but general purpose low level and intermediate representations. (shallow semantic parsing)
- The first approach reusability of the representation across domains is very limited.
- In the second approach it is difficult to construct a general purpose ontology and create symbols that are shallow enough to be learnt but detailed enough to be useful for all possible applications.
- Now the community has moved from the more detailed domain dependent representation to more shallow ones.

Semantic Interpretation

- Structural Ambiguity
- Word Sense
- Entity and Event Resolution
- Predicate-Argument structure
- Meaning Representation

Semantic Interpretation

- Semantic parsing is considered as a part of a large process **semantic interpretation**.
- Semantic interpretation is a kind of representation of text that can be fed into a computer to allow further computational manipulations and search.
- Here we discuss about some of the main components of this process.
- We begin the discuss with **Syntactic structures** by Chomsky which introduced the concept of a transformational phrase structure grammar.

Semantic Interpretation

- Later Katz and Fodor wrote a paper “The structure of a semantic theory” where they proposed a few properties a semantic theory should possesses. A semantic theory should be able to :
  - Explain sentences having ambiguous meaning. (Example: the sentence “the bill is large” can represent money or the beak of a bird)
  - Resolve the ambiguities of the words in the context.(Example: the sentence “the bill is large but need not be paid” can be disambiguated)
  - Identify meaningless but syntactically well-formed sentences.(Example “colorless green ideas sleep furiously”)
  - Identify syntactically or transformationally unrelated paraphrases of a concept having the same semantic content.
- We now look at some requirements for achieving a semantic representation.

Structural Ambiguity

- When we talk of structure, we refer to the syntactic structure of sentences.
- Since syntax and semantics have such strong interaction most theories of semantic interpretation refer to the underlying syntactic representation.
- Syntax has become the first stage of processing followed by various other stages in the process of semantic interpretation.

Word Sense

- In any given language the same word type or word lemma is used in different contexts and with different morphological variants to represent different entities or concepts in the world.
- For example the word nail represents a part of the human anatomy and also to represent an iron object.
- Humans can easily identify the use of nail in the following sentences:
  - He nailed the loose arm of the chair with a hammer.
  - He bought a box of nails from the hardware store.
  - He went to the beauty salon to get his nails clipped
  - He went to get a manicure. His nails had grown very long.
- Resolving the sense of words in a discourse, therefore, constitutes one of the steps in the process of semantic interpretation.

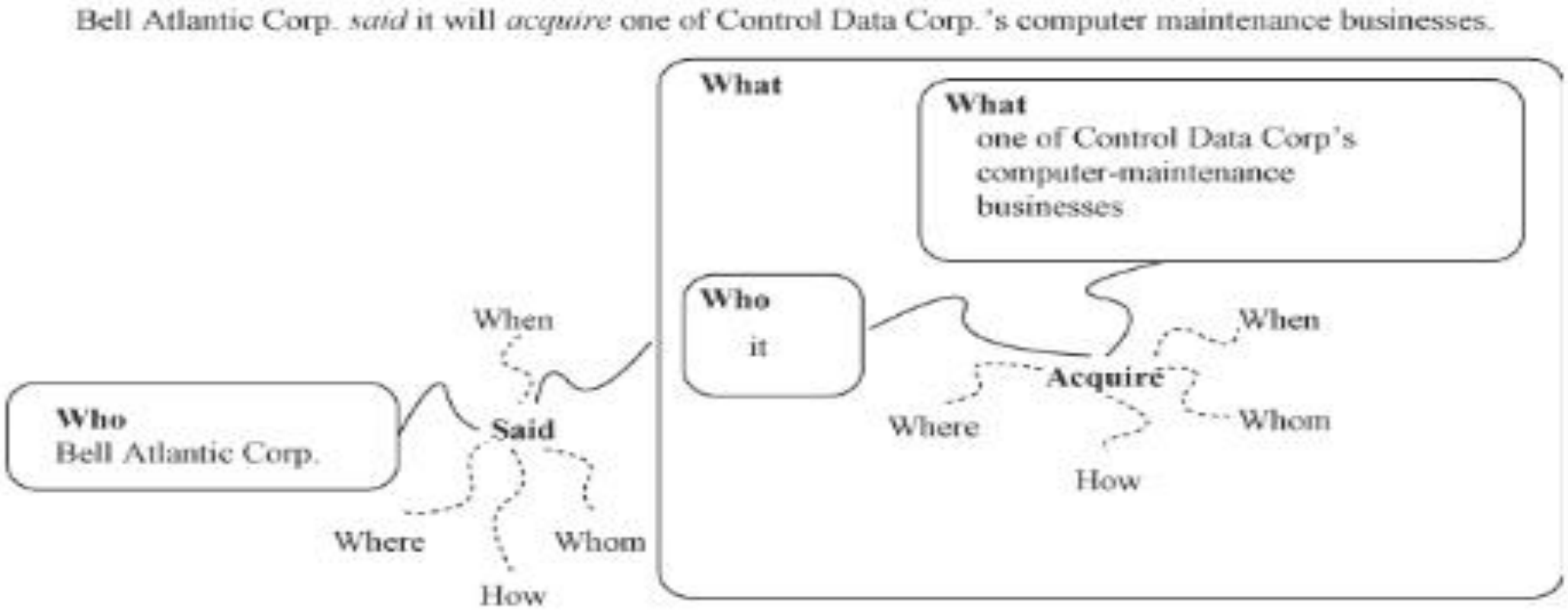
Entity and Event Resolution

- Any discourse consists of a set of entities participating in a series of explicit or implicit events over a period of time.
- The next important component of semantic interpretation is the identification of various entities that are sprinkled across the discourse using the same or different phrases.
- Two predominant tasks have become popular over the years:
  - Named entity recognition
  - Coreference resolution

Predicate-Argument Structure

- Another level of semantic structure is identifying the participants of the entities in these events.
- Resolving the argument structure of the predicates in a sentence is where we identify which entities play what part in which event.
- This process can be defined as the identification of who did what to whom, when where why and how.

Predicate-Argument Structure



Meaning Representation

- The final process of the semantic interpretation is to build a semantic representation or meaning representation that can be manipulated by algorithms to various application ends.
- This process is called the deep representation.
- The following examples show sample sentences and their meaning representation for the RoboCup and GeoQuery domains:
  - If our player 2 has the ball, then position our player 5 in the midfield. ((bowner(player our 2)) (do(player our 5)(pos(midfield)))).
  - Which river is the longest? answer(x1,longest(x1 river(x1)))
- This is a domain-specific approach. Our further discussion here will be about domain independent approaches.

System Paradigms

- Researchers from linguistics community have examined meaning representations at different levels of granularity and generality exploring the space of numerous languages.
- Many of the experimental conditions do not have hand annotated data.
- The approaches of semantic interpretation generally fall into the following three categories:
- **System Architectures**
  - **Knowledge based:** These systems use a predefined set of rules or a knowledge base to obtain a solution to a new problem.
  - **Unsupervised:** These systems tend to require minimal human intervention to be functional by using existing resources that can be bootstrapped for a particular application or problem domain.

System Paradigms

- **Supervised:** These systems involve manual annotation of some phenomena that appear in a sufficient quantity of data so that machine learning algorithms can be applied. Feature functions are created to allow each problem instance to be projected into a space of features.
- **Semi-supervised:** In instances where human annotation is difficult we can use machine generated output or bootstrap an existing model by having humans correct its output.

- **Scope**
  - **Domain Dependent:** These systems are specific to certain domains such as air travel reservations or simulated football coaching.
  - **Domain Independent:** These systems are general enough that the techniques can be applicable to multiple domains without little or no change.

System Paradigms

- **Coverage**
  - **Shallow:** These systems tend to produce an intermediate representation that can then be converted to one that a machine can base its actions on.
  - **Deep:** These systems usually create a terminal representation that is directly consumed by a machine or application.

Word Sense

- Resources
- Systems
- Software

Word Sense - Introduction

- In a compositional approach the semantics is composed of the meaning of its parts in a discourse.
- The smallest parts in textual discourse are the words themselves:
  - Tokens that appear in the text
  - Lemmatized parts of the tokens
- It is not clear whether it is possible to identify a finite set of senses that each word in a language exhibits in various contexts.
- Attempts to solve this problem range from rule based, knowledge based to completely unsupervised, supervised and semi-supervised learning.

Word Sense - Introduction

- The early systems were either rule based or knowledge based and used dictionary definitions of senses of words.
- Unsupervised word sense disambiguation techniques induce the sense of a word as it appears in various corpora.
- Supervised approaches for word sense disambiguation assume that a word can evoke only one sense in a given context.
- The coarser the granularity of senses for a word the more consistent the annotation and more learnable they become.
- There is an increased chance that this lower granularity might not identify nuances that are fine enough for the consuming application.

Word Sense - Introduction

- The number of applications which need word sense disambiguation are few and for that reason very few manually sense-tagged text corpora have been produced.
- The absence of standard criteria has also prevented the merging of various resources that have sense information.
- In information retrieval it is an accepted fact that multiple words in a query matching with multiple words in the document provide an implicit disambiguation.
- In speech recognition also context classes have proven to be applicable than word classes.

Word Sense - Introduction

- In the case of domain specific applications words generally map to a unique concept and therefore no need for word sense disambiguation.
- The reasons for the lack of progress in word sense disambiguation are:
  - Lack of standardized evaluations
  - The range of resources needed to provide the required knowledge as compared to other tasks
  - The difficulty of obtaining adequately large sense-tagged data sets.
- Techniques used to measure the performance of an automatic word sense disambiguation system is an important issue.

Word Sense - Introduction

- One proposal for performance measure is called most frequent sense (MFS) baseline.
- A very important property of a gold standard sense-tagged corpus is that it should be replicable (multiple annotators accepting it) to a high degree.
- Word sense ambiguities can be of three principal types:
  - Homonymy- words share the same spelling but different meaning (bank).
  - Polysemy- fine nuances can be assigned to the word depending on the context (financial bank and bank of clouds).
  - Categorical ambiguity- the word book can be a noun or a verb.

Word Sense - Introduction

- In English word senses have been annotated for each POS separately.
- In a few languages like Chinese sense annotation is done per lemma and ranges across all POS because distinction between a noun and a verb is not clear.

Resources

- The availability of resources is the key factor in the disambiguation of word senses in corpora.
- There is no hand-tagged sense data available until recently.
- Early work on word sense disambiguation used machine readable dictionaries or thesaurus as knowledge sources.
- Prominent sources were:
  - Longman dictionary of contemporary English
  - Roget's thesaurus
- Later a significant lexicographical resource WordNet was created.

Resources

- WordNet in addition to being a lexicographical resource has a rich taxonomy connecting words across many different relationships such as :
  - Hypernymy- a broader class of things (bird)
  - Hyponymy- a subclass of hypernymy (crow,pegion etc)
  - Homonymy- words share the same spelling but different meaning (bank)
  - Meronymy- it denotes a constituent of something (finger is a meronymy of hand)
- WordNet in addition to being used for sense disambiguation can also be used to create a semantic concordance (SEMCOR of Brown Corpus).

Resources

- Other corpuses that were developed for English language are:
  - DSO corpus of sense-tagged English- It was created by tagging WordNet version 1.5 senses on the Brown and Wall Street Journal Corpara for 121 nouns and 70 verbs.
  - Onto Notes corpus- It is released through the Linguistic Data Consortium (LDC). It has tagged a significant number of verb (2,700) and noun (2200) lemmas covering lemmas.
  - Cyc- It is a good example of a useful resource that creates a representation of common sense knowledge about objects and events.

Resources

- Efforts for creating resources for other languages are as follows:
  - HowNet- a network of words for Chinese
- The Global WordNet association keeps track of WordNet development across various languages.
- A few semiautomatic methods were used for expanding coverage were used for languages like Greek.

Systems

- Rule based
- Supervised
- Unsupervised
- Algorithms motivated by Cross-linguistic Evidence
- Semi-Supervised

Systems

- Let us discuss about some sense disambiguation systems.
- We can classify the sense disambiguation system into four main categories:
- Rule based or knowledge based
- Supervised
- Unsupervised
- Semi-supervised

Rule Based

- The first generation of word sense disambiguation systems were primarily based on dictionary sense definitions.
- Access to the exact rules and the systems was very limited.
- Here we look at a few techniques and algorithms which are accessible.
- The first generation word sense disambiguation algorithms were mostly based on computerized dictionaries.
- We now have a look at Lesk algorithm which can be used as a baseline for comparing word sense disambiguation performance.

Rule Based

Algorithm 4–1. Pseudocode of the simplified Lesk algorithm The function COMPUTEOVERLAP returns the number of words common to the two sets

---

Procedure: SIMPLIFIED_LESK(word, sentence) **returns** best sense of *word*

```
1: best-sense ← most frequent sense of word
2: max-overlap ← 0
3: context ← set of words in sentence
4: for all sense ∈ senses of word do
5:    signature ← set of words in gloss and examples of sense
6:    overlap ← COMPUTEOVERLAP(signature, context)
7:    if overlap gt max-overlap then
8:       max-overlap ← overlap
9:       best-sense ← sense
10:   end if
11: end for
12: return best-sense
```

Rule Based

- A few modifications can be made to the Lesk algorithm so that synonyms, hypernyms, hyponyms, meromynys, and so on of the words in the context as well as in the dictionary definition are used to get accurate overlap statistic.
- One more algorithm for dictionary-based word sense algorithm used Roget's Thesaurus categories. (Yarowsky)
- It was able to classify unseen words into 1042 categories using the corpus 10-million-word Grolier's Encyclopedia.

Rule Based

- The method consists of three steps:
  - The first step is a collection of contexts
  - The second step computes weights for each of the salient words.
    - The amount of context used was 50 words on each side of the target word.
    - P(w/Rcat) is the probability of a word w occurring in the context of a Roget's category Rcat.
  - In the third step, the unseen words in the test set are classified into the category that has the maximum weight.

Rule Based

1. Collect contexts for each of the *Roget's Thesaurus* categories.
2. Determine weights for each of the salient words in the context.

$$\frac{P(w_i|RCat)}{P(w_i)}$$

3. Use the weights for predicting the appropriate category of the word in the test corpus.

$$\arg\max_{RCat} \sum_{w} \log \frac{P(w_i|RCat)P(RCat)}{P(w_i)}$$

Figure 4–2. Algorithm for disambiguating words into *Roget's Thesaurus* categories

Rule Based

- One more knowledge based algorithm uses graphical representation of senses of words in context to disambiguate the term under consideration. (Navigli and Velardi)
- This is called the structural semantic interconnections (SSI) algorithm.
- It uses various sources of information like:
  - WordNet
  - Domain labels
  - Annotated corpora to from structural specifications of concepts or semantic graphs

Rule Based

- The algorithm consists of two steps:
  - An initialization
  - An iterative step- the algorithm attempts to disambiguate all the words in the context iteratively.
- Its performance is very close to that of supervised algorithms and surpasses the best unsupervised algorithms.
- We now look at an example semantic graph for the two senses of the term bus.
  - The first one is the vehicle sense
  - The second one is the connector sense.

Rule Based

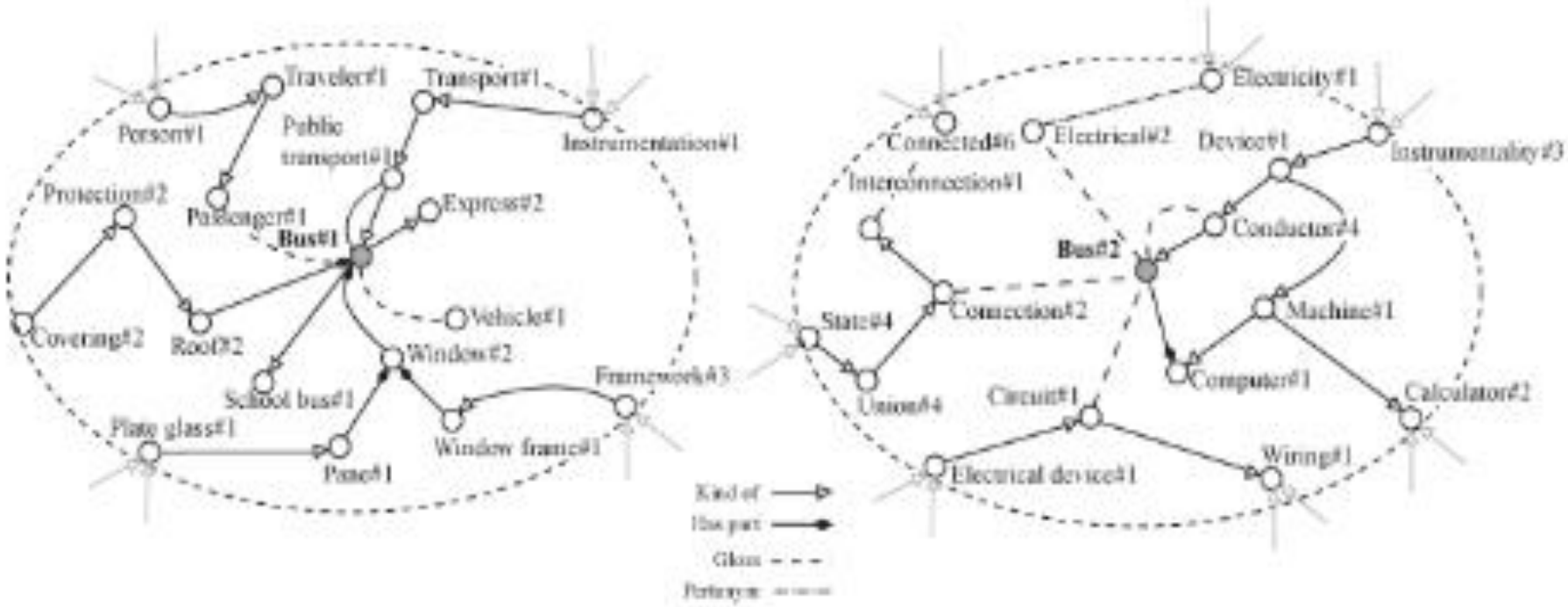


Figure 4–3. The graphs for sense 1 and 2 of the noun *bus* as generated by the SSI algorithm

Rule Based

- Notation:
  - T(Lexical context) is the list of terms in the context of the term t to be disambiguated T=[t1,t2,……,tn].
  - S_1^t,S_2^t,……S_n^t are structural specifications of the possible concepts (or senses) of t.
  - I (the semantic context) is the list of structural specifications of the concepts associated with each of the terms in T\{t} (except t). I=[S^{t1},S^{t2},…..,S^{tn}], that is the semantic interpretation of T.
  - G is the grammar defining the various relations between the structural specifications.(semantic inter-connections among the graphs).
  - Determine how well the structural specifications in I match that of S_1^t,S_2^t,……S_n^t using G.
  - Select the best match S_i^t .

Rule Based

- The algorithm works as follows:
- A set of pending terms in the context $P=\{t_i/S^{ti}=null\}$ is maintained and I is used in each iteration to disambiguate terms in P.
- The procedure iterates and each iteration either:
  - Disambiguates one term in P and removes it from the pending list
  - Stops because no more terms can be disambiguated
- The output I is updated with the sense of t.
- Initially I contains structures for monosemous terms in T\{t} and any possible disambiguated synsets.

Rule Based

- If this set is null then the algorithm makes an initial guess at what the most likely sense of the least ambiguous term in the contest is.
- The algorithm selects those terms t in P that show semantic interconnections with at least one sense of S of t and one or more senses in I.
- A function $F_I(S,t)$ determines the likelihood of S being the correct interpretation of t and is defined as :

$$f_I(S,t) = \begin{cases} \rho(\{\varphi(S,S') | S' \in I\}), & \text{if } S \in Senses(t) \\ 0, & \text{otherwise} \end{cases} \quad (4.1)$$

where $Senses(t)$ are the senses associated with the term t, and

$$\varphi(S,S') = \rho'(\{w(e_1 \cdot e_2 \cdots e_n) | S \xrightarrow{e_1} S_1 \xrightarrow{e_2} \ldots \xrightarrow{e_{n-1}} S_{n-1} \xrightarrow{e_n} S'\}) \quad (4.2)$$

Rule Based

- A function ρ’ of the weights w is the path connecting S and S’.
- A good choice for ρ and ρ’ can be sum or average sum function.
- A CFG G = (E,N,SG,PG) encodes all the meaningful semantic patterns where:

$$E = \{e_{kind-of}, e_{has-kind}, e_{part-of}, \dots\}$$

are the edge labels,

$$N = \{S_G, S_s, S_g, S_1, S_2, \dots, E_1, E_2, \dots\}$$

are nonterminal symbols that encode paths between the senses,

$$S_G$$

is the start symbol of the graph G, and

$$P_G = \{S_G \rightarrow S_s/S_g, S_s \rightarrow S_1/S_2/S_3, S_1 \rightarrow E_1S_1/E_1, E_1 \rightarrow e_{kind-of}/e_{part-of}, S_g \rightarrow e_{gloss}S_5/S_4/S_5, \dots\}$$

Rule Based

- The other algorithms which are rule based are:
- Patwardhan, Benerjee and Pedersen compared several similarity measures based on WordNet.
- The emergence of Wikipedia has led to better applications in this area.
- Strube and Penzetto provided an algorithm called WikiRelate to estimate the distance between two concepts using the Wikipedia taxanomy.
- Navigli and Penzotto introduced an approach for creating a multilingual lexical knowledge base that establishes a mapping between Wikipedia and WordNet.

Supervised

- Supervised approach to disambiguation transfers all the complexity to the machine learning algorithm, but still requires hand annotation.
- The sense inventory for supervised approach must be predetermined and any changes needs a reannotation.
- The rules and knowledge can be incorporated in the form of features.
- A particular knowledge source and the classifier may have a few issues and the sense-tagged data could be noisy to varying degrees.
- We now look at a few algorithm which are useful for word sense disambiguation.

Supervised

- Brown was among the first few to use supervised learning and used the information in the form of parallel corpora.
- Yearowsky used a rich set of features and decision lists to tackle the word sense problem.
- A few researchers like Ng and Lee have come up with several variations including different levels of context and granularity.
- Now we look at the different **classifiers** and **features** that are relatively easy to obtain.

Supervised-Classifiers

- The most common and high performing classifiers are:
- Support Vector Machines
- Maximum Entropy Classifiers
- Since each lemma has a separate sense inventory a separate model is trained for each lemma and POS combination.

Supervised-Features

- Here we discuss a few commonly found subset of features that have been useful in supervised learning of word sense.
- The list provides a base that can be used to achieve nearly state-of-the-art performance.
- The features are as follows:
- **Lexical context**- This feature comprises the words and lemmas of the words occurring in the entire paragraph or a smaller window of five words.
- **Parts of Speech**- This feature comprises the POS information for words in the window surrounding the word that is being sense tagged.

Supervised-Features

- **Bag of Words Context**- This feature comprises using an unordered set of words in the context window. A threshold is typically tuned to include the most informative words in the large context.
- **Local Collocations**
  - They are an ordered sequence of phrases near the target word that provide semantic context for disambiguation.
  - A very small window of about three tokens on each side of the target word in contiguous pairs of triplets are added as a list of features.
- For example for a target word w $C_{i,j}$ would be a collocation where I and j are the start and offset with respect to the word w.
- A positive sign indicates words to the right and negative sign indicates words to the left.

Supervised-Features-Example Local Collocations

- Let us look at an example:
- Ng and Lee used the following features:
- $C_{-1,-1}$, $C_{1,1}$,$C_{-2,-2}$,$C_{2,2}$,$C_{-2,-1}$,$C_{-1,1}$,$C_{1,2}$,$C_{-3,-1}$,$C_{-2,1}$,$C_{-1,2}$ and $C_{1,3}$
- The example sentence “He bought a box of nails from the hardware store.” where we try to disambiguate **nails.**
- The collection $C_{1,1}$ would be “from” and $C_{1,3}$ would be the string “from the hardware”.
- Stop words and punctuations are not removed before creating the collocations and for boundary conditions a null word is collocated.

Supervised-Features

- **Syntactic Relations**- If the parse of the sentence containing the target word is available then we can use syntactic features.

Algorithm 4–2. Rules for selecting syntactic relations as features

---

1: **if** *w* is a *noun* **then**
2: *select* parent head word (*h*)
3: *select* part of speech of *h*
4: *select* voice of *h*
5: *select* position of *h* (left, right)
6: **else if** *w* is a *verb* **then**
7: *select* nearest word *l* to the left of *w* such that *w* is the parent head word of *l*
8: *select* nearest word *r* to the right of *w* such that *w* is the parent head word of *r*
9: *select* part of speech of *l*
10: *select* part of speech of *r*
11: *select* part of speech of *w*
12: *select* voice of *w*
13: **else if** *w* is a *adjective* **then**
14: *select* parent head word (*h*)
15: *select* part of speech of *h*
16: **end if**

Supervised-Features

- **Topic Features**- The broad topic or domain of the article that the word belongs to is also a good indicator of what sense of the word might be most frequent.

Supervised- Additional Features

- Chen and Palmer proposed some additional features for disambiguation:
- **Voice of the Sentence**- This feature indicates whether the word occurs in passive, semipassive or active sentence.
- **Presence of subject/object**-
  - This binary feature indicates whether the target word has a subject or object.
  - We can also use the actual lexeme and semantic roles rather than the syntactic subject/object.
- **Sentential Complement**- This feature indicates whether the word has a sentential complement. (I know that he will do it.)

Supervised- Additional Features

- **Prepositional Phrase Adjunct**- This feature indicates whether the target word has a prepositional phrase and if so selects the head of the noun phrase inside the prepositional phrase.
- **Named Entity**- This feature is the named entity of the proper nouns and certain types of common nouns.
- **WordNet**- WordNet synsets of the hypernyms of head nouns of the noun phrase arguments of verbs and prepositions.

Supervised-Features–Verb Sense Disambiguation

- **Path**- This feature is the path from the target verb to the verb's argument.
- **Subcategorization**- The subcategorization frame is essentially the string formed by joining the verb phrase type with that of its children.

Unsupervised

- There is a problem in the progress of word sense disambiguation due to the lack of labeled training data to train a classifier.
- There are a few solutions to this problem:
  - Devise a way to cluster instances of a word so that each cluster effectively constrains the examples of the word to a certain sense. This could be considered sense induction through clustering.
  - Use some metrics to identify the proximity of a given instance with some sets of known senses of a word and select the closest to be the sense of that instance.
  - Start with seeds of examples of certain senses then iteratively grow them to form clusters.

Unsupervised

- We assume that there is already a predefined sense inventory for a word and that the unsupervised methods use very few hand-annotated examples and then attempt to classify unseen test instances into one of their predetermined sense categories.
- We first look at the category of algorithms that use some form of distance measure to identify senses.
  - Rada and others introduced a metric for computing the shortest distance between the two pairs of senses in WordNet.
  - This metric assumes that multiple co-occurring words exhibit senses that would minimize the distance in a semantic network of hierarchical relations. Ex: IS-A from WordNet.

Unsupervised

- Resnik proposed a new measure of semantic similarity: **information content** in an IS-A taxonomy which produces much better results than the edge-counting measure.
- Agirre and Rigau refined this measure calling it **conceptual density** which not only depends on the number of separating edges but is also sensitive of the hierarchy and the density of its concepts.
- It is independent of the number of concepts being measured.
- Conceptual density is defined for each of the subhierarchies.
- The sense that falls in the subhierarchy with the highest conceptual density is chosen to be the correct sense.

Unsupervised

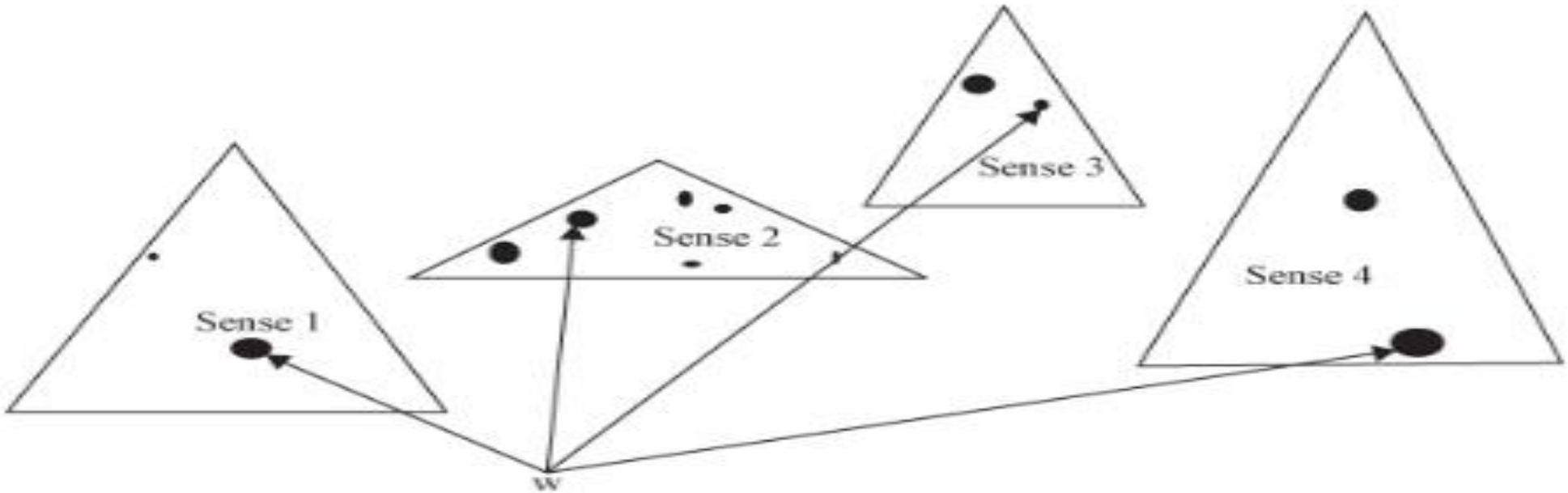


Figure 4–4. Conceptual density

$$CD(c,m) = \frac{\sum_{i=0}^{m-1} \text{hyponyms}^{i^{-0.20}}}{\text{descendants}_c} \qquad (4.3)$$

Unsupervised

- In the figure in the previous slide sense 2 is the one with the highest conceptual density and is therefore the chosen one.
- Resnik observed that selectional constraints and word sense are related and identified a measure by which to compute the sense of a word on the basis of predicate argument statistics.
- This algorithm is primarily limited to the disambiguation of nouns that are arguments of verb predicates.
- Let A_R be the selectional association of the predicate p to the concept c with respect to argument R. A_R is defined as:

$$A_R(p,c) = \frac{1}{S_R(p)} P(c|p) \log \frac{P(c|p)}{P(c)}$$

Unsupervised

- If n is the noun that is in an argument relation R to predicated p, and {s1,s2,……,sk} are its possible senses, then, for i from 1 to k compute:

$$C_i = \{c | c \text{ is an ancestor of } s_i\}$$
$$a_i = \max_{c \in C_i} A_R(p, c)$$

- where ai is the score for sense si. The sense si which has the largest value of ai is sense for the word. Ties are broken by random choice.
- Leacock, Miller, and Chodorow provide another algorithm that makes use of corpus statistics and WordNet relations, and show that monosemous relatives can be exploited for disambiguating words.

Algorithms Motivated by Crosslinguistic Evidence

- There are a family of unsupervised algorithms based on crosslinguistic information or evidence.
- Brown and others were the first to make use of this information for purposes of sense disambiguation.
- They were interested in sense differences that required translating into other languages in addition to sense disambiguation.
- They provide a method to use the context information for a given word to identify its most likely translation in the target language.

Algorithms Motivated by Crosslinguistic Evidence

- Dagan and Itai used a bilingual lexicon paired with a monolingual corpus to acquire statistics on word senses automatically.
- They also proposed that syntactic relations along with word co-occurances statistics provide a good source to resolve lexical ambiguity.
- Diab performed experiments using machine translated English-to-Arabic translations to extract sense information for training a supervised classifier.
- Now let us look at a crosslinguistic algorithm.

Algorithms Motivated by Crosslinguistic Evidence

1. L1 words that translate into the same L2 word are grouped into clusters.
2. SALAAM (Sense Assignment Leveraging Alignment and Multilinguality) identifies the appropriate senses for the words in those clusters according to the words senses' proximity in WordNet. The word sense proximity is measured in information theoretic terms on the basis of an algorithm by Resnik [57].
3. A sense selection criterion is applied to choose the appropriate sense label or set of sense labels for each word in the cluster.
4. The chosen sense tags for the words in the cluster are propagated back to their respective contexts in the parallel text. Simultaneously, SALAAM projects the propagated sense tags for L1 words onto their L2 corresponding translations.

Figure 4–5. SALAAM algorithm for creating training using parallel English-to-Arabic machine translations

Semi-supervised

- The next category of algorithms we look at are those that start from a small seed of examples and an iterative algorithm that identifies more training examples using a classifier.
- This additional automatically labeled data can be used to augment the training data of the classifier to provide better predictions for the next cycle.
- Yorowsky algorithm is such an algorithm which introduced semi-supervised methods to the word sense disambiguation problem.

Semi-supervised

- The algorithm is based on the assumption that two strong properties are exhibited by corpora:
- **One sense per collocation**: Syntactic relationship and the types of words occurring nearby a given word tend to provide a strong indication as to the sense of that word.
- **One sense per discourse**: In a given discourse all instances of the same lemma tend to invoke the same sense.
- Based on the assumptions that these properties exist the Yarowsky algorithm iteratively disambiguates most of the words in a given discourse.

Semi-supervised

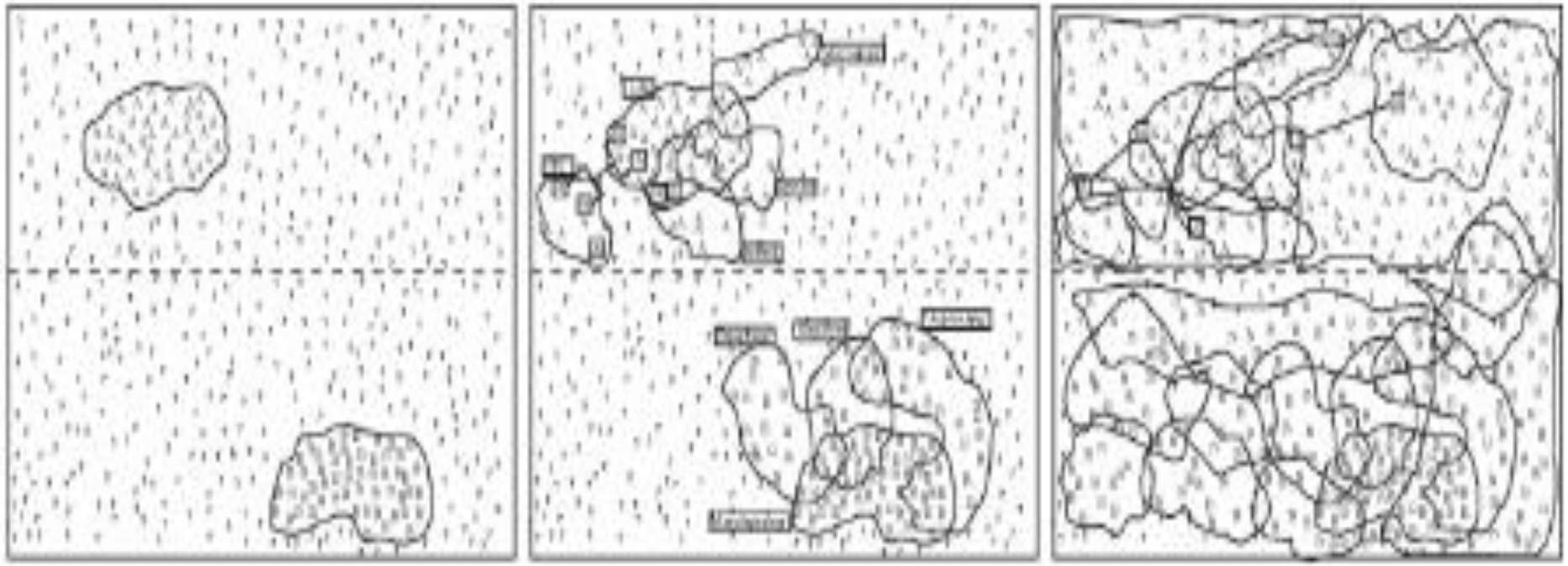


Figure 4–6. The three stages of the Yarowsky algorithm

Semi-supervised

Step 1. In a sufficiently large corpus, identify all the instances of a particular polysemous word that needs to be disambiguated, storing its context alongside.
Step 2. Identify a small set of instances that are strongly representative of one of the senses of the word. This can either be done in a completely unsupervised fashion by identifying collocations that give a strong indication of the sense usage for the word under consideration or by manually tagging a small portion of the data. In this example, we assume a polysemous word with only two senses, but this algorithm can be extended to n senses.
Step 3.

Step 3a. Train a supervised classifier on this set of examples.

Step 3b. Using these classifiers, classify the remaining instances of the word in the corpus and select those that are classified above a certain level of confidence.

Step 3c. Filter out the possible misclassifications using one sense per discourse constraint, and identify possible new collocations to be added to the list of seed collocations.

Step 3d. Repeat step 3 iteratively, thereby slowly shrinking the residual.

Step 4. Stop. At some point, a small, stable residual will remain.
Step 5. The trained classifier can now be used to classify new data, and that in turn can be used to annotate the original corpus with sense tags and probabilities.

Figure 4–7. The Yarowsky algorithm

Semi-supervised

- Another variation of semi-supervised systems is the use of unsupervised methods for the creation of data combined with supervised methods to learn models for that data.
- There are two presumptions here:
  - One that the potential noise of wrong examples selected from a corpus during this process would be low enough so as not to effect learnability.
  - Two that the overall discriminative ability of the model is superior to purely unsupervised methods or to situations in which not enough hand-annotated data is available to train a purely supervised system.

Semi-supervised

- Mihalcea and Moldovan describe an algorithm which is used to obtain examples from large corpora for particular senses from WordNet.
- Mihalcea proposed the following method using Wikipedia for automatic word sense disambiguation.
  - Extract all the sense in Wikipedia in which the word under consideration is a link.
  - There are two types of links: a simple link such as [[bar]] or a piped link such as [[musical_notation|bar]].
  - Filter those links that point to a disambiguation page. This means that we need further information to disambiguate the word. If the word does not point to a disambiguation page the word itself can be the label.
  - For all piped links the string before the pipe serves as the label.

Semi-supervised

- Collect all the labels associated with the word and then map them to possible WordNet senses.
- They might all map to the same sense essentially making the verb monosemous and not useful for this purpose.
- The categories can be mapped to a significant number of WordNet categories there by providing sense-disamguated data for training.

- This algorithm provides a cheap way to extract sense information for many words that display the required properties.

Software

- Several software programs are available for word sense disambiguation.
- IMS (It makes Sense): This is a complete word sense disambiguation system
- WordNet Similarity-2.05: These WordNet similarity modules for Perl provide a quick way of computing various word similarity measures.
- WikiRelate: This is a word similarity measure based on categories in Wikipedia.