Natural Language Processing

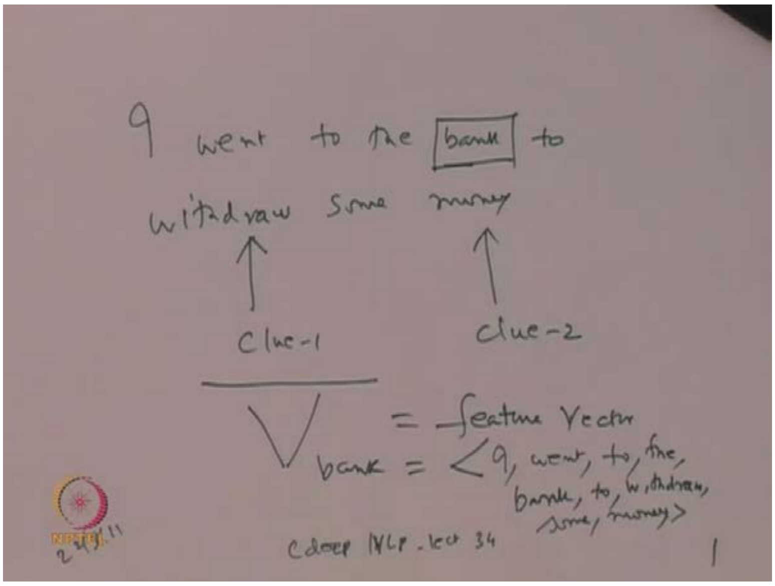
Prof. Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology, Bombay

Lecture No - 34
Word Sense Disambiguation: Supervised and Unsupervised Methods

We are coming towards the end of our discussion on word sense disambiguation. And through that we will be covering a very fundamental topic of natural language processing. Now, we have looked at word net, which is the repository from where we obtain the senses of a word, which needs disambiguation. And we have also studied a number of techniques for word sense disambiguation. And the last lecture actually covered knowledge based techniques for word sense disambiguation.

In today’s lecture, we concentrate on word sense disambiguation using supervised and unsupervised methods. The supervised methods was touched upon in the last lecture, where we said that this needs machine learning techniques and training corpora. We would like to obtain the senses of a word, for a new sentence from whatever we have learnt through the training corpus, just to give you an example.

(Refer Slide Time: 01:45)



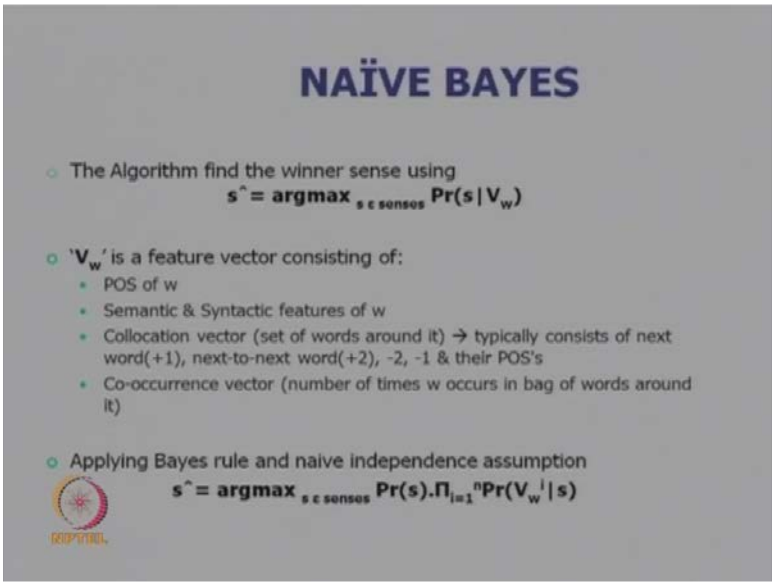
Suppose we have this sentence, I will write it down, I went to the bank to withdraw some money, bank is the word which needs disambiguation. And what we find from the sentence that there are clues in the form of one is with strong clue, clue 1; another clue is money, which is clue 2. So, these are strong clues by which bank can be disambiguated. Now, supervised approach would like to make this very fundamental notion, that words, around the word needing disambiguation, provide clues and this words are, what are called the features. So, these features are learnt along with their weightage, in the environment of the word to be disambiguated. And from there, when a new sentence comes up, we would like to produce the disambiguation of the target word that is the basic idea. So, we proceed with the slides.

(Refer Slide Time: 03:15)

SUPERVISED APPROACHES

First, we take up supervised approaches to word sense disambiguation.

(Refer Slide Time: 03:23)



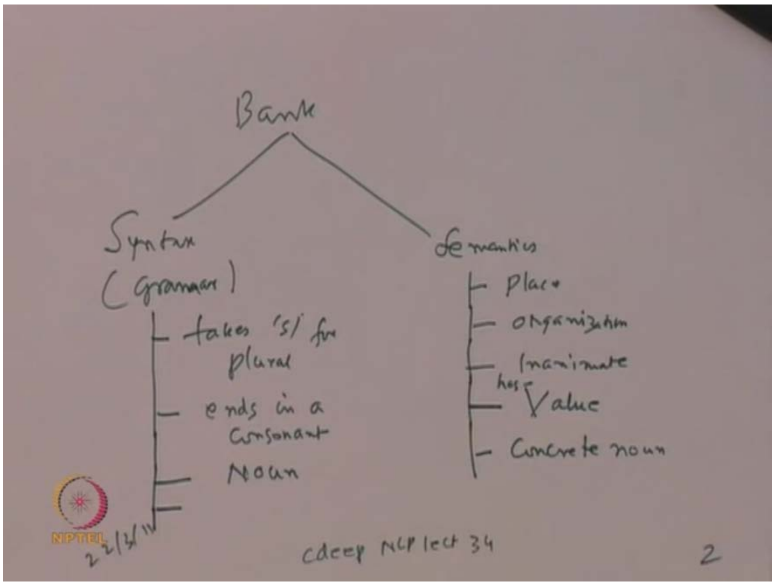
NAÏVE BAYES

- The Algorithm find the winner sense using
  $s\hat{} = \text{argmax}_{s \in senses} \Pr(s|V_w)$
- 'V_w' is a feature vector consisting of:
  - POS of w
  - Semantic & Syntactic features of w
  - Collocation vector (set of words around it) → typically consists of next word(+1), next-to-next word(+2), -2, -1 & their POS's
  - Co-occurrence vector (number of times w occurs in bag of words around it)
- Applying Bayes rule and naive independence assumption
  $s\hat{} = \text{argmax}_{s \in senses} \Pr(s).\Pi_{i=1}^{n}\Pr(V_w^{i}|s)$

The first technique there is the Naive Bayes approach, and we have this formula here, which is very familiar to students of artificial intelligence, natural language processing machine learning, computer vision and so on. So, as the formula shows, we have s hat, which is the target sense of a word w and the argmax maximizes on the probability, for various senses of the word w, what is this term V w, w as such is never used for disambiguation V stands for vector. If feature vector which is constructed based on w, and the words in its environment, just like we wrote the sentence in the paper, I went to the bank to withdraw some money, and here the feature vector is constructed for bank. So, we are constructing V bank which is the feature vector, and this consists of maybe words like, I went to the bank to withdraw some money. So, these words which are in the environment of bank would form the feature vector V, and we will operate with that.

So, it is useful for you to remember that, we never work with the word only or the word as such, this is never the entity on which argmax computation is done, instead we work with the words in the environment. So, proceeding with the slide now, s hat is argmax s belonging to senses of w, probability of the sense given V w. So, V w is a feature vector consisting of part of speech of w, whether w is a noun or verb or adjective or adverb and so on. The sentence we wrote, I went to the bank to withdraw some money, here bank, the part of speech of bank is noun, so this important for disambiguation. Then comes the feature called semantic and syntactic feature of w what would that mean, so this are word properties of w, and the grammatical properties of w. So, let us record some of the properties of the example, we are discussing, let us take the word bank, I will write it here.

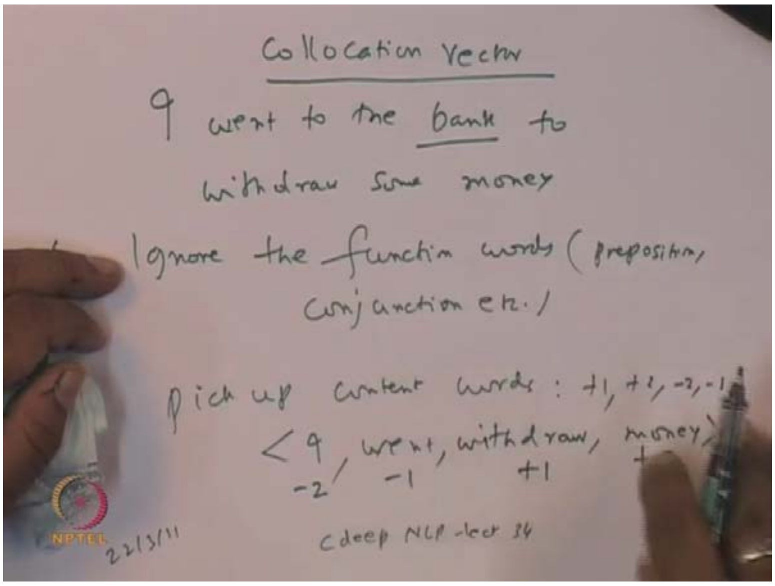
(Refer Slide Time: 06:51)



What properties on bank do, we have under syntax, that is grammar and semantics what, we have, under semantics maybe, we can record that it is a place, it is an organization, it is inanimate, it has value and so on. Under syntax, we can say that takes s for plural, then ends in a consonant; it is a noun, under semantics you can say concrete noun, and so on.

So, these are the features, which are used for constructing, the featured vector let us proceed with other feature descriptions, collocation vector set of words, around it typically consists of next word, next to next word, previous to previous word, previous word and their part of speeches. So, collocation vector consists of these things, and co-occurrence vector is the number of times w occurs in bag of words around it. So, co-occurrence vector consists of typically numbers, collocation vector on the other hand, consists of the actual words around the target word. So, we will again use this example.

(Refer Slide Time: 09:08)



Collocation Vector

I went to the bank to withdraw some money

Ignore the function words (preposition, conjunction etc.)

Pick up content words : +1, +2, -2, -1

< I, went, withdraw, money >

-2, -1, +1, +

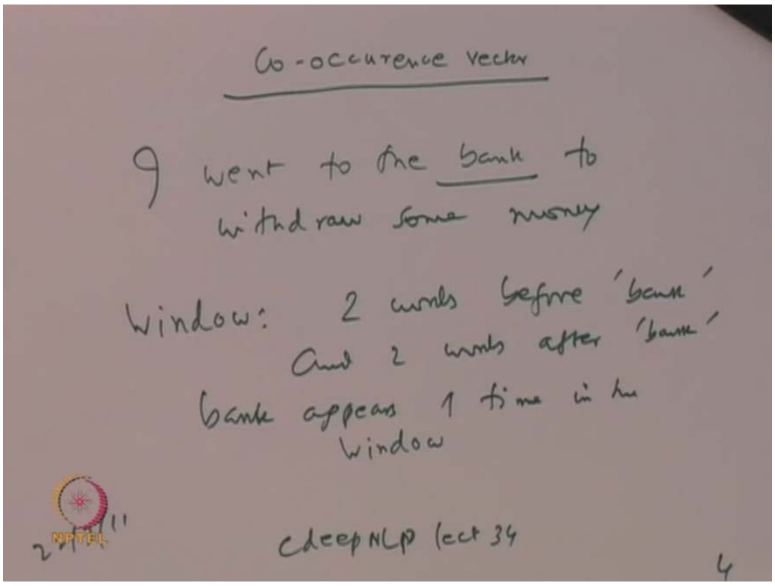
22/3/11

cdeep NLP-lect 34

I went to the bank to withdraw some money, so suppose, we ignore the function words, so things like preposition, conjunction etcetera. Pick up content words plus 1, plus 2 minus 2, minus 1. So, if you look at bank, then the 2 content words before it, minus 2 and minus 1 are i and went, and plus 2 would be money, and plus on would be withdraw. So, withdraw and money, so this becomes plus 2, this becomes plus 1; this is minus 1; this is minus 2.

So, collocation vector is a very important concept in natural language processing and it is important to have a clear idea, about what is collocation vector and how it is constituted. So, from the example that we writing on the paper, we see that the collocation vector for bank consists of I went withdraw and money, so this would be used for sense disambiguation. Next we come the co-occurrence vector, so this is the, discussion on collocation vector.

(Refer Slide Time: 11:20)



Co-occurrence vector

I went to the bank to withdraw some money

Window: 2 words before 'bank'
And 2 words after 'bank'
bank appears 1 time in the window

2/[illegible]/11

cdeepNLP lect 34

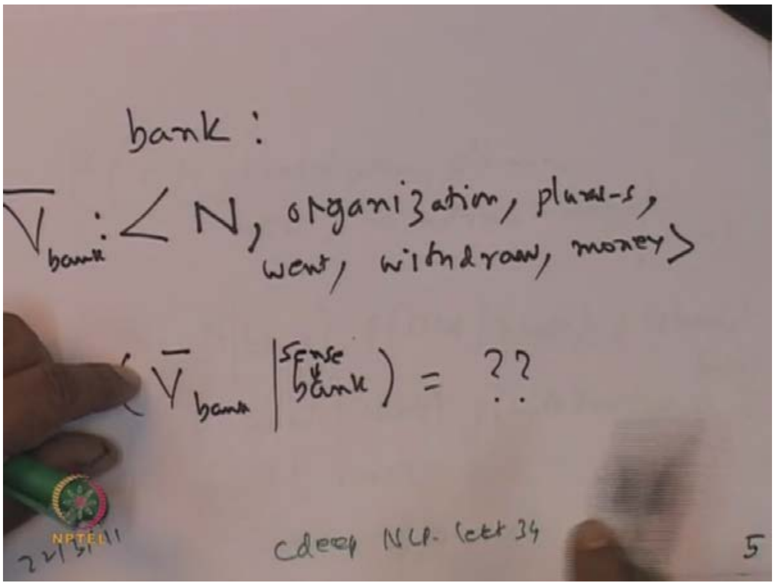
4

Here, we now discuss co-occurrence vector, this is actually if I take the example, I am working with I went to the bank to withdraw some money. So, if I look at this example, then what is the co-occurrence vector, co-occurrence vector is simply the number of times bank appears, in a neighborhood. So, suppose I choose my window to be 2 words before bank, and 2 words after bank.

So, window is 2 words before bank and 2 words after bank if, we have this as the window, then 2 words before bank is to the 2 words after bank are to withdraw. So, bank appears 1 time in the window, this information is quite crucial for sense disambiguation and we would like to make use of that. So, when we, look at the slide once again, we have now understood, what we are working, with part of speech of w semantic and syntactic features of w, collocation vector and the co-referencing vector.

Now, we apply a very well know technique, namely the Bayes theorem, the Bayes rule and the Naive independence assumption. So, if you look at the formula here now, s hat is equal to argmax, s belongs to the senses of w. So, first I isolate the prior probability Pr s, and then the vector is inverted it becomes V w given s. However, there is naïve independence assumption, so each feature, now is converted into a conditional probability. And that gives us the new formula, we will again write it down in the paper, to make our concepts clear, so writing it up so we will now look at the vector and understand this expression.

(Refer Slide Time: 14:29)



bank :

$\bar{V}_{bank} : \langle N, \text{organization}, \text{plural-s}, \text{went}, \text{withdraw}, \text{money} \rangle$

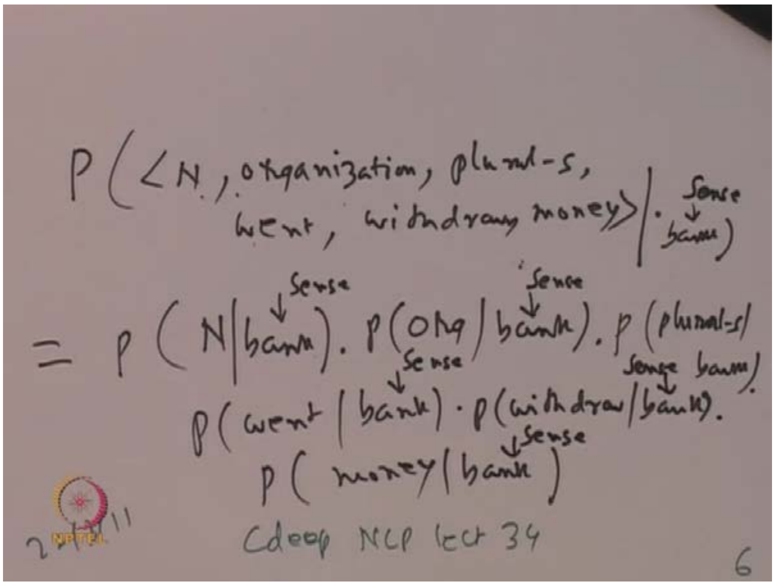
$(\bar{V}_{bank} \mid \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}}) = ??$

Cdeep NLP lect 34

5

So, for bank the features vectors are, the part of speech of bank, which is noun, other properties like organization, plural s that means takes s in plural, collocation vector is went withdraw and money. So, this is the feature vector of bank.

(Refer Slide Time: 15:30)



$P(\langle N, \text{organization}, \text{plural-s}, \text{went}, \text{withdraw money} \rangle \mid . \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}})$

$= P(N \mid \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}}) . P(\text{org} / \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}}) . P(\text{plural-s} / \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}}) .$

$P(\text{went} \mid \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}}) \cdot P(\text{withdraw} / \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}}) .$

$P(\text{money} \mid \overset{\text{sense}}{\underset{\text{bank}}{\downarrow}})$

Cdeep NLP lect 34

6

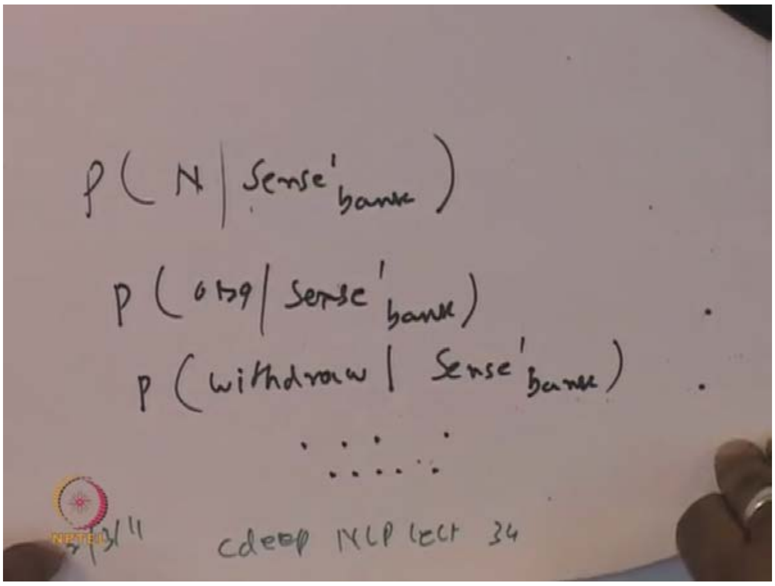
Now probability of V bar bank, given bank, what does it come out to be, that will be written as probability of noun, organization, plural s, went, withdraw, money, given the word bank, this Naive Bayes assumption will become probability of noun, given bank probability of organization, given bank probability of plural s, given bank into probability of went, given bank probability of withdraw, given bank into probability of money given bank. So, this whole vector given the word bank has become a product of the features each is a feature, n is a feature of bank, noun is a feature bank, organization is a feature of bank. So, this product of features given the word bank, this decomposition has taken place because of Naive Bayes assumption.

So, now what is the independence assumption here, we have the understand that clearly, what the independence assumption is saying, is that the feature of being noun is independent of the feature of being organization, which again is independent of feature of being plural, independent if feature of being went, independent of feature of being withdraw, independent of the feature money. These features have nothing to do with each other, so they are independent, therefore this conditional probability has given rise to product of conditional probabilities.

Now, clearly you can see that these, independence assumption is Naïve, how can it be that the features are independent of each other, that is not possible. Now, the property of being noun, and the property of being organization, are clearly depend on each other, in particular, the property of being organization is impossible. If the property of noun is not satisfied anything, that is an organization, has to be a noun, if something is not a noun, it cannot be an organization. So, these kind of, independence assumption is definitely not true, however it can lead to some convenient formulization, and ease of calculating parameters.

And therefore, these assumption, is many times made, in spite of all its inaccuracies, it still delivers results because of the fact that, we are actually ranking the senses. When, we rank the senses a little bit of inaccuracy in terms of independence assumption does not really hurt. So, we proceed and go to the slide now, and after look at the mathematical calculations s hat is argmax, overall possible senses of P r s given V w, where V w is the feature vector, you applied BAYES rule. And then u have obtained the product of features, given the sense, so let us go to, the paper once again, where we wrote the independence expression, so here, we wrote it for the word. Actually when, we apply the formula, we have to have, the sense of bank so we will express this by sense of bank, here also what we mean is the sense of bank. So, these are senses, sense, sense everything is being done with respect to sense.

(Refer Slide Time: 20:45)



$p(N \mid Sense'_{bank})$

$p(org \mid Sense'_{bank})$

$p(withdraw \mid Sense'_{bank})$

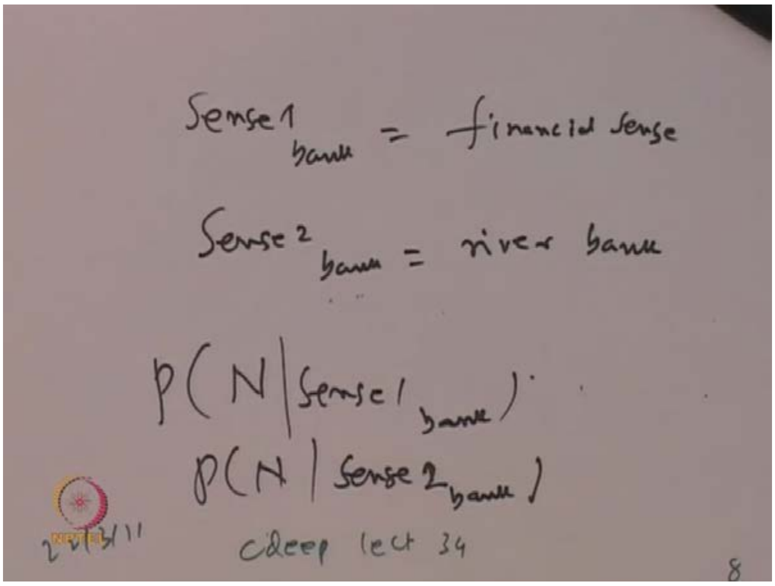
. . . .

. . . . .

3/3/11 cdeep NLP lect 34

Let us consider the independent expression once again, the probability of noun for a particular sense of bank, probability of being an organization for the sense 1 of bank, probability of withdraw, being a collocated word for the sense word of bank and so on. So, these become our parameters, these are our parameters now suppose, we have the financial sense of bank, financial sense of bank sense 1 is that, this will become more clear, when we write it down.

(Refer Slide Time: 21:46)



$\text{Sense1}_{\text{bank}}$ = financial sense

$\text{Sense2}_{\text{bank}}$ = river bank

$P(N \mid \text{sense1}_{\text{bank}})$

$P(N \mid \text{Sense2}_{\text{bank}})$

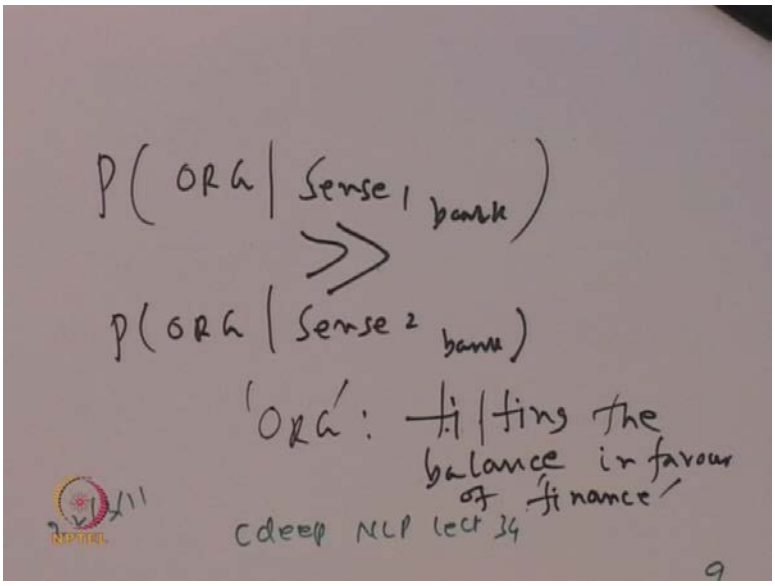
22/3/11 cdeep lect 34

8

So, suppose sense 1 of bank is the financial sense, and we also have sense 2 of bank which is the river bank sense, then what we are trying to compute or calculate is the probability that this sense has noun feature, given sense ` of bank, and the other sense also has noun feature for the word bank. So, do you understand what this expression means, it is the probability that the financial sense of bank has noun sense, probability that river bank sense of bank has noun part of speech. So, probability that financial sense of bank has noun part of speech, probability that river bank sense of bank has noun part of sense.

So, these 2 probabilities may not be very distinct, and thereby they may not be, cannot be playing such a crucial role, in deciding the financial sense versus river bank sense. However, we must note that, if the domain becomes restricted, suppose it is a tourism domain, where the river bank sense of bank possibly appears more, then, we will see a difference between these 2 probabilities.

(Refer Slide Time: 23:47)



$$P(ORG \mid sense_{1\,bank}) \gg P(ORG \mid sense_{2\,bank})$$

'ORG': tilting the balance in favour of 'finance'

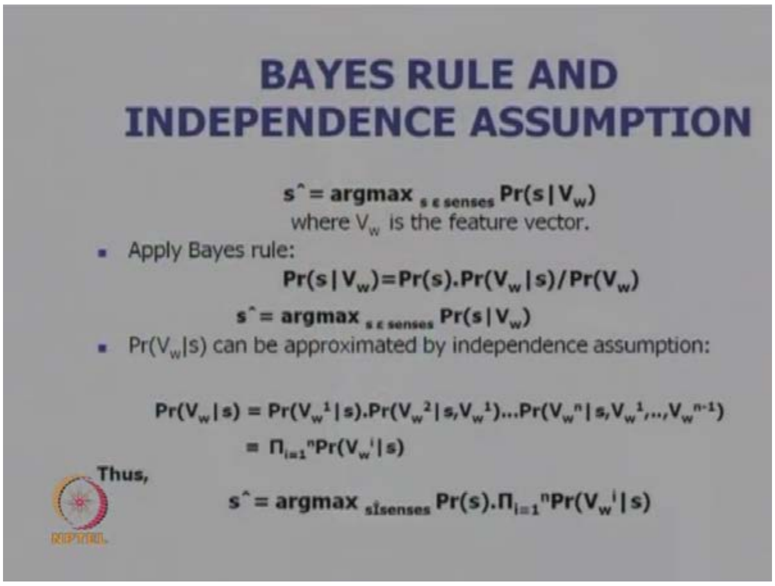
cdeep NLP lect 34

9

But another feature will have much more remarkable influence, this others probability value is let us say probability that the property, is organization and we have sense 1 of bank and property of being organization for sense 2 of bank. So, thus the first sense of bank which is financial sense, does it have organization property, property of being an organization, and thus the other sense of bank which is river bank sense, does it have organization property. Clearly, the first sense of bank, which is financial, will have a much higher probability here. We can really write that this probability will be much, much more than the probability of sense 2 on bank, being organization.

So, O RG indeed will be a tilting the balance ORG, will surely have the influence of tilting, the balance in favor of finance. So, now I think you are getting an idea about what roles, the features the play, how important role the features play, when we use the Naive Bayes assumption, and break the feature vector into smaller components or single features by which, we have the probability values uncovered, with their high values for 1 sense and low value for another. So this indeed is a tipping factor, a factor that tilts the balance for disambiguation. So, I guess this, ideas are bought out.

(Refer Slide Time: 26:03)



BAYES RULE AND INDEPENDENCE ASSUMPTION

$$s\hat{} = \text{argmax}_{s \in senses} \Pr(s|V_w)$$

where V_w is the feature vector.

- Apply Bayes rule:

$$\Pr(s|V_w) = \Pr(s).\Pr(V_w|s)/\Pr(V_w)$$

$$s\hat{} = \text{argmax}_{s \in senses} \Pr(s|V_w)$$

- $\Pr(V_w|s)$ can be approximated by independence assumption:

$$\Pr(V_w|s) = \Pr(V_w^1|s).\Pr(V_w^2|s,V_w^1)\ldots\Pr(V_w^n|s,V_w^1,..,V_w^{n-1})$$

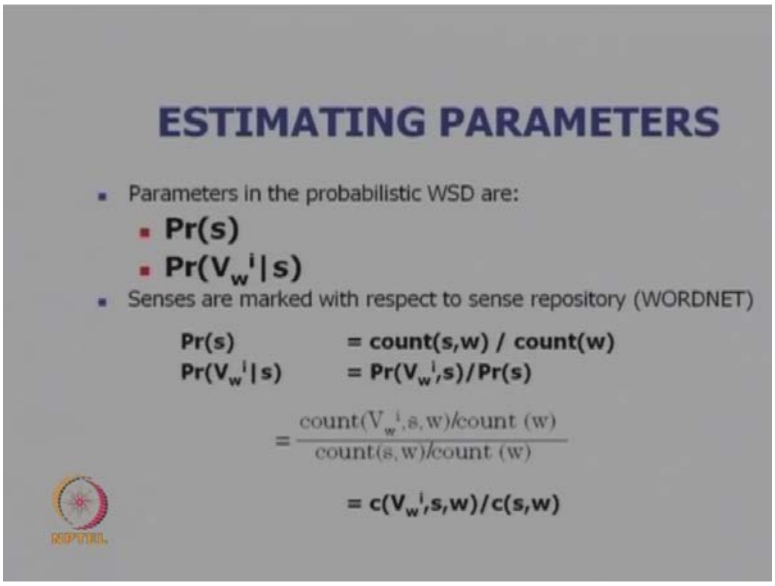
$$= \Pi_{i=1}^n \Pr(V_w^i|s)$$

Thus,

$$s\hat{} = \text{argmax}_{\text{sIsenses}} \Pr(s).\Pi_{i=1}^n \Pr(V_w^i|s)$$

Clearly proceeding further.

(Refer Slide Time: 26:05)



ESTIMATING PARAMETERS

- Parameters in the probabilistic WSD are:
  - $\Pr(s)$
  - $\Pr(V_w^i|s)$
- Senses are marked with respect to sense repository (WORDNET)

$$\Pr(s) = \text{count}(s,w) / \text{count}(w)$$

$$\Pr(V_w^i|s) = \Pr(V_w^i,s)/\Pr(s)$$

$$= \frac{\text{count}(V_w^i,s,w)/\text{count }(w)}{\text{count}(s,w)/\text{count }(w)}$$

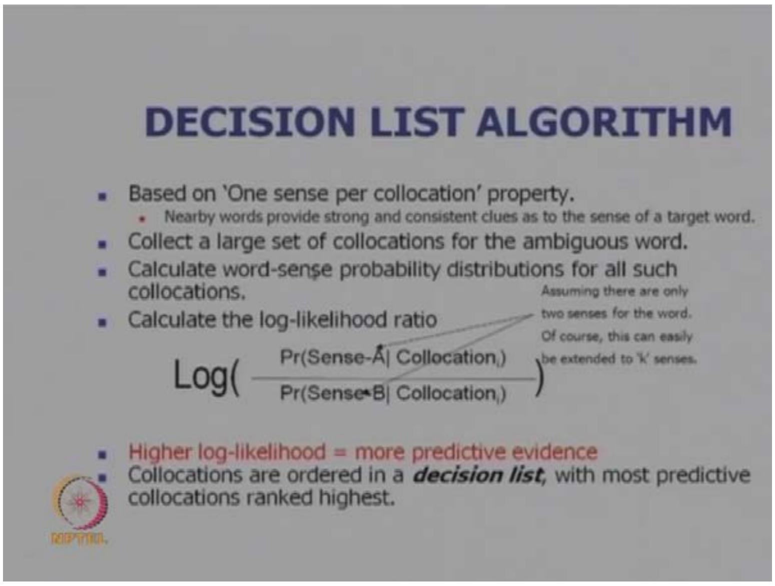
$$= c(V_w^i,s,w)/c(s,w)$$

Next task, that we are faced with, is the task of estimating the parameters namely, the probability values. So, the parameters that we are prior probability of sense P r s, and another probability of an individual feature V w i diet feature given s. How do we calculate this senses are now marked with respect to the sense repository which is the WORDNET, now the probability of s how would we get this probability? Probability of s will be equal to the count of s and w, and divided by the count of w, on the other hand the probability of V w i given s is equal to probability of V w i comma s divided by probability of s. So, this comes out to be equal to, count of the feature sense and w combination, divided by count of sense and w combining. So, this is the way it is calculated we note that for the count this 3 things appear together the word a particular sense of the word, and a particular feature of the word divided by word, and the sense of the word.

So, if we consider our current example, the word bank, the financial sense of bank, let us say the organizational feature of the environment in which word w appears, divided by the count of the times, the word bank appears in the financial sense. So, number of times bank is in financial sense in the corpus, and this is the number of times financial sense, the word bank and the feature of being an organization, appears together. Now, here 1 thing must be very clear to you, look at this count and the numerator, now if the word bank appears in the river bank sense, then it is very unlikely that the, feature of organization, will come along with this sense, and the word.

So, this count will be very small in fact, we will not be surprised, if it 0 for the river bank sense of the word bank, why at all, will the word bank, in the sense of finance, why at all, will it have the feature of organization or other why at all, will the river bank sense of the word bank, have the feature of organization, river is not a feature, river is not an organization, river bank is not an organization. So, this count is likely to be very small so when that happens, we can see that the competing sense of the word bank, which is the river bank. It will lose out in favor of the finance sense of bank, provided the context is indicating that, because of the influence of this feature. So, I suppose you understand the importance of a good quality features for the purpose of disambiguation, and the importance of their being represented and used in the sense determination.

(Refer Slide Time: 30:11)



DECISION LIST ALGORITHM

- Based on 'One sense per collocation' property.
  - Nearby words provide strong and consistent clues as to the sense of a target word.
- Collect a large set of collocations for the ambiguous word.
- Calculate word-sense probability distributions for all such collocations.
- Calculate the log-likelihood ratio

$$Log\left(\frac{Pr(Sense\text{-}A \mid Collocation_i)}{Pr(Sense\text{-}B \mid Collocation_i)}\right)$$

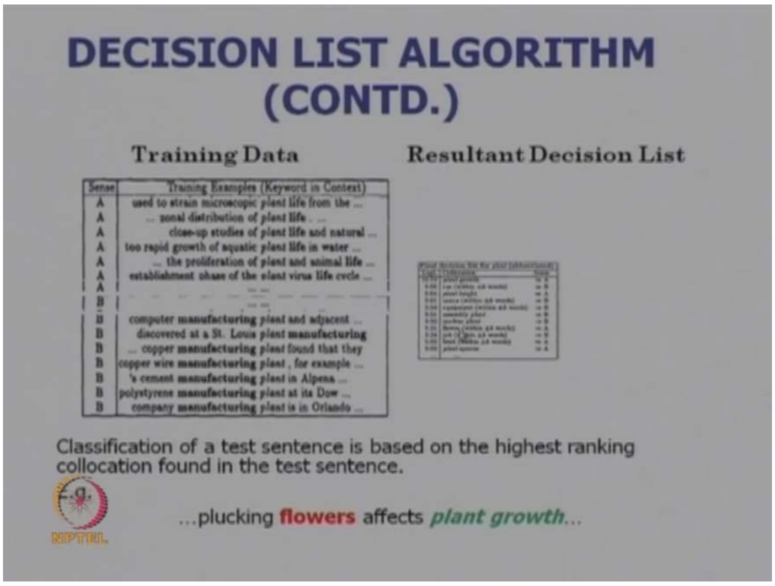
Assuming there are only two senses for the word. Of course, this can easily be extended to 'k' senses.

- Higher log-likelihood = more predictive evidence
- Collocations are ordered in a ***decision list***, with most predictive collocations ranked highest.

So, this was the Naive Bayes algorithm we now change the gear, and look at another interesting algorithm, again based supervised machine learning technique. So, this is the algorithm of decision list based algorithm here, we make an important assumption which is 1 sense per collocation property. This is that nearby words provide strong and consistent clues as to the sense of a target word. And all of them are so arranged, they are placed along with the word to form a meaningful sentence, in such a way, that their association points to only 1 sense of the word, so nearby words provide strong and consistent clues as to the sense of a target word.

So, the algorithm is as follows collect a large set of collocation for the ambiguous word, in this case let us say bank, which we were working with calculate word sense, probability distribution for all such collocation, here is a formula assuming there are only 2 senses for the word of course, this can easily be extended to k senses. So, probability of sense A given the collocation, I divided by probability of sense B given the collocation, I again take this ratio, and take the logarithm. Now, if the probability of sense A is smaller, then the probability of sense B, then we will have a number less than 1, and the log of that number be minus, otherwise it will be positive. So, higher the log- likelihood more is the predictive evidence, collocations are ordered in a decision list, with most predictive collocations, ranked the highest. So, here is a training data, and the training data is for the word plant.

(Refer Slide Time: 32:25)



DECISION LIST ALGORITHM (CONTD.)

Training Data

Sense	Training Examples (Keyword in Context)
A	used to strain microscopic plant life from the ...
A	... zonal distribution of plant life . ...
A	close-up studies of plant life and natural ...
A	too rapid growth of aquatic plant life in water ...
A	... the proliferation of plant and animal life ...
A	establishment phase of the plant virus life cycle ...
A	...
B	...
B	computer manufacturing plant and adjacent ...
B	discovered at a St. Louis plant manufacturing
B	... copper manufacturing plant found that they
B	copper wire manufacturing plant , for example ...
B	's cement manufacturing plant in Alpena ...
B	polystyrene manufacturing plant at its Dow ...
B	company manufacturing plant is in Orlando ...

Resultant Decision List

Final decision list for plant (abbreviated)

LogL	Collocation	Sense
[illegible]	plant growth	⇒ A
[illegible]	car (within ±k words)	⇒ B
[illegible]	plant height	⇒ A
[illegible]	union (within ±k words)	⇒ B
[illegible]	equipment (within ±k words)	⇒ B
[illegible]	assembly plant	⇒ B
[illegible]	nuclear plant	⇒ B
[illegible]	flower (within ±k words)	⇒ A
[illegible]	job (within ±k words)	⇒ B
[illegible]	fruit (within ±k words)	⇒ A
[illegible]	plant species	⇒ A

Classification of a test sentence is based on the highest ranking collocation found in the test sentence.

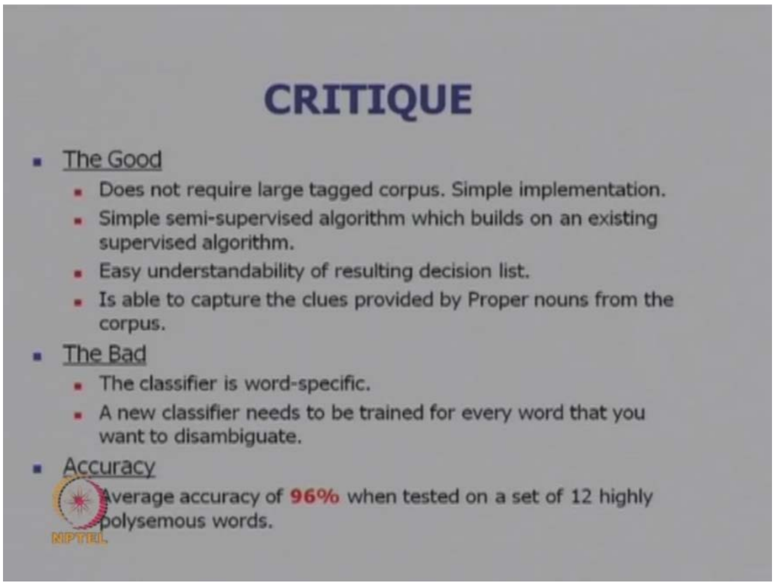
...plucking **flowers** affects *plant growth*...

So, first few sentence is take the botanical sense of the word plant, used to strain microscopic plant life from the whatever, zonal distribution of plant life, close up studies of plant life, and natural whatever, too rapid growth of aquatic plant, life in water, the proliferation of plant, and animal life establishment phase of plant, virus life cycle etcetera. The other sense of plant is the sense of a manufacturing unit, so computer manufacturing plant and adjacent, whatever discovered at a saint louis plant manufacturing, copper manufacturing plant found that etcetera, copper wire manufacturing plant and so on, cement manufacturing plant, polystyrene manufacturing plant, company manufacturing plant and so.

The collocations which are collected are shown here, plant growth is the sense a this is a collocation car within plus minus k words this goes to sense B, plant height is sense A union within plus minus k word sense B, equipment within plus minus k word sense B again, simply plant A is sense B, nuclear plant is sense B, flower if you add the word flower within plus minus k words it is sense A, job within plus minus k words sense B fruit, within plus minus k words sense A plant species is again sense A. So, this is an interesting table which shows, the kind of collocation, and their influence on determining the sense. If we find the word equipment, within a range of the word plant, we are almost sure log like is 9.54 that is the sense B, or the manufacturing unit sense.

If we find the word flower then we are almost sure that plant is used in a botanical sense here, so classification of test sentence is based on the highest ranking collocation, found in the test sentence. So, if we have this new sentence here, plucking flowers affects plant growth, then since within near vicinity of the plant, we find the word flowers, we also find the word growth then, we have a strong reason to say, that this word plant is in the sense of botanical sense, this is the botanical sense of plant as of course, to manufacturing units. So, what it shows here is that, we have the training data with marking of sense, and we have the resulted decision list, obtained from the training data, and this data is processed to obtain the decision list, so this was an algorithm and it is used quite extensively.

(Refer Slide Time: 35:56)

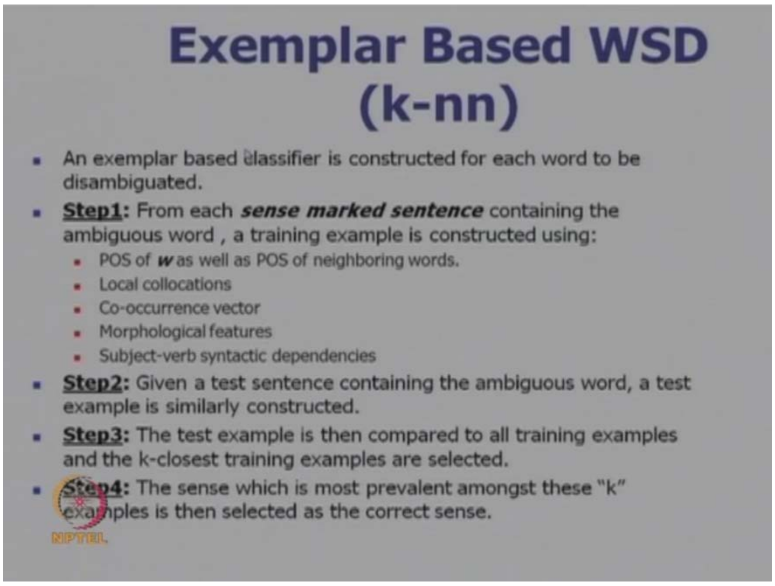


CRITIQUE

- The Good
  - Does not require large tagged corpus. Simple implementation.
  - Simple semi-supervised algorithm which builds on an existing supervised algorithm.
  - Easy understandability of resulting decision list.
  - Is able to capture the clues provided by Proper nouns from the corpus.
- The Bad
  - The classifier is word-specific.
  - A new classifier needs to be trained for every word that you want to disambiguate.
- Accuracy
  - Average accuracy of 96% when tested on a set of 12 highly polysemous words.

Now, we have some critical comments about, the supervised algorithm which are Naive Bayes and decision list the good factors are the following, it does not require large tagged corpus. The implementation is simple semi supervised algorithm, which builds on an existing supervised algorithm, is possible to be constructed from here. We look at the list here easy understandability of the resulting decision list, it able to capture the clues provided by proper nouns, from the corpus, which is a serious difficulty for other algorithms. The dab part is that, the classifier is word specific, the actual words which appear, and form the decision list, have their say in the decision for sense. A new classifier needs to be trained for every, word that you, want to disambiguate, the average accuracy comes out to be 96 percent when tested on a set of 12 highly polysemous words.

(Refer Slide Time: 36:58)



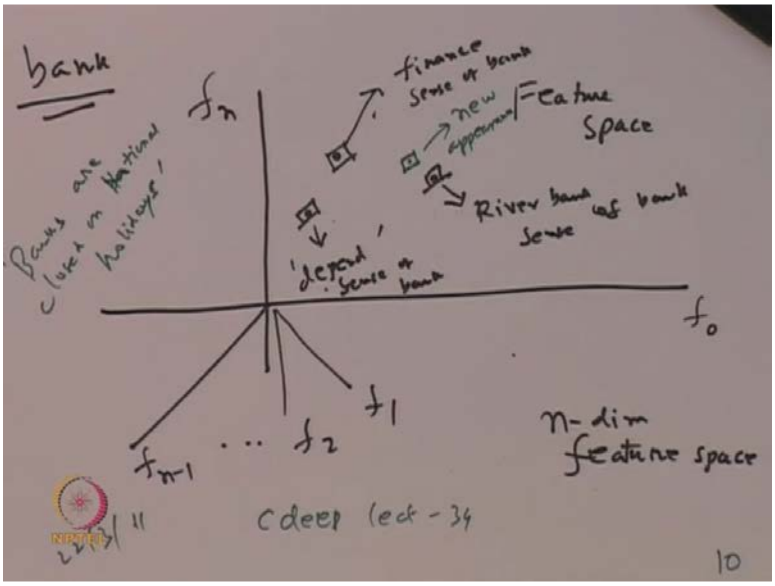
Exemplar Based WSD (k-nn)

- An exemplar based classifier is constructed for each word to be disambiguated.
- <u>**Step1**</u>**:** From each ***sense marked sentence*** containing the ambiguous word , a training example is constructed using:
  - POS of ***w*** as well as POS of neighboring words.
  - Local collocations
  - Co-occurrence vector
  - Morphological features
  - Subject-verb syntactic dependencies
- **Step2:** Given a test sentence containing the ambiguous word, a test example is similarly constructed.
- **Step3:** The test example is then compared to all training examples and the k-closest training examples are selected.
- **Step4:** The sense which is most prevalent amongst these "k" examples is then selected as the correct sense.

The next algorithm is the exemplar based WSD algorithm this is the k-nn, k nearest neighborhood algorithm, an exemplar based classifier is constructed for each word to be disambiguated, how do we do this, first step is from each sense marked sentence containing the ambiguous word a training example is constructed, using part of speech of W as well as part of speech of neighboring words, we get the local collocations, we get the co-occurrence vector, we get the morphological features and finally, subject verb syntactic dependencies.

All these features are used for this algorithm exemplar based word sense, disambiguation, step 2 is given a test sentence containing, the ambiguous word, a test example is similarly, constructed. So, we obtain a test example based on the parameters which are listed here. Step 3 is the test example is then compared to all training examples, and the k closest training examples are selected. The sense which is most prevalent amongst these k examples is then selected as the correct sense. So, a diagrammatic representation, which I will write now, will help understand, these notion k nearest neighbor, this is as follows.

(Refer Slide Time: 38:39)



What, we are doing is that, we have a feature space and for the word bank, which is a target word, we create this is f0 f1 f2 fn minus 1 fn. So, we have a n dimensional feature space, now based on the features that, we have mentioned, we obtain points. So, this is the finance sense of bank, this is the river bank sense of bank, this is the depend sense of bank. So, what happens is when a new sentence comes in, and the word bank is present in that, we get a feature vector and the corresponding point, for the new appearance. So, assume we have a new sentence, where the word bank appears, at this new sentence is assumed, the banks are closed on national holidays.

So, this is a new sentence, banks are closed on national holidays, we have to disambiguation bank here. So, from the environment of bank, we create a feature vector as has been given in the k-nn based algorithm, and this feature vector, now becomes a point in the feature space, this is the new appearance. And we are finding that, this is placed at some distance, with respect to different points, in the feature space. Now, the point it is closest to is taken to give, the sense of the new appearance of the word bank. So, in this case, we find that this is erroneously close to the river bank sense, which will be given out as the sense by the machine, this will be wrong feature vectors should have been such that, the point is close to this point.

And then they would be connect, so this is simple idea, which is called the k nearest neighborhood classifier idea. And again it is based on the features so we have seen 3 algorithms today, Naive Bayes classifier, decision list based classifier, and the k- n classifier. And all of them are based on features around the environment of the target word. We will continue in may be two more lectures to finish the topic of word sense disambiguation.