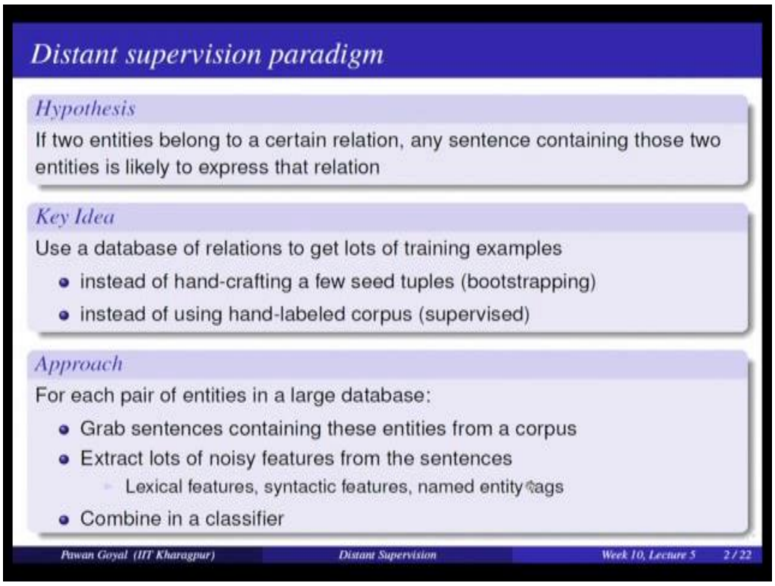
Natural Language Processing
Prof. Pawan Goyal
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 50
Distant Supervision

Welcome back for the final lecture of this week. So, for relation extraction, we have talked about hand built patterns, we have talked about bootstrapping approaches and also supervised approaches. So, in this lecture, we will talk about a very interesting approach that uses the previous 2 approaches nicely, it is called distant supervision and we will see what is the basic idea hoping you will find it very interesting so what is the idea?

(Refer Slide Time: 00:42)



Distant supervision paradigm

Hypothesis

If two entities belong to a certain relation, any sentence containing those two entities is likely to express that relation

Key Idea

Use a database of relations to get lots of training examples

- instead of hand-crafting a few seed tuples (bootstrapping)
- instead of using hand-labeled corpus (supervised)

Approach

For each pair of entities in a large database:

- Grab sentences containing these entities from a corpus
- Extract lots of noisy features from the sentences
  - Lexical features, syntactic features, named entity tags
- Combine in a classifier

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 2 / 22

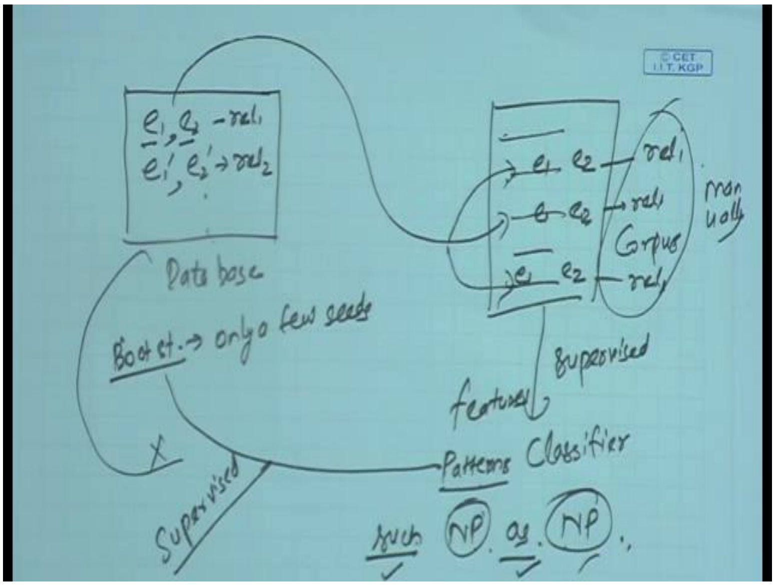
We start with the hypothesis that is similar to what you were using also in bootstrapping. So, what is the hypothesis? If there are 2 entities that are connected by a particular relation, whatever sentence in the corpus they occur in that sentence conveys that relation that is the hypothesis that is what this model is built on that I have 2 entities, I know there is relation between them if you find a sentence in the corpus where these entities are there, I can think of these are connected by that relation only. So, yeah it is likely to express that relation. So, what is the key idea here? So, we are trying to make use of both bootstrapping and supervised ideas. So, that is so here what we do? We start by using a database where you already have lot of instances of a particular relation. So, lot of seed tuples are there.

You use database of relations to get lots of training examples, instead of handcrafting a few seed tuples that you do in bootstrapping in bootstrapping you take only probably few seed tuples and you start doing some iterative method in or and instead of using hand labeled corpus that we do in the case of supervised approach. So, you do not do either of this, but you start with a database that contains lot of substitution examples.

So, what is approach? So, you have an database where you know what are the entities that are connected by different relations now you automatically grab various entity pairs from this database and you will lot of unannotated data you go through that corpus and find out wherever each entities occur you try to extract various features from there you do not take simple patterns you grab them you know you take them as the labeled sentences extract features from that.

You extract grab sentences containing these entities from a corpus extract lots of noisy features from the sentences these can be lexical features syntactic features named entity tags this is what you were doing in supervised approach, but now you are doing that without any hand labeling and then combine this in a classifier.

(Refer Slide Time: 03:13)



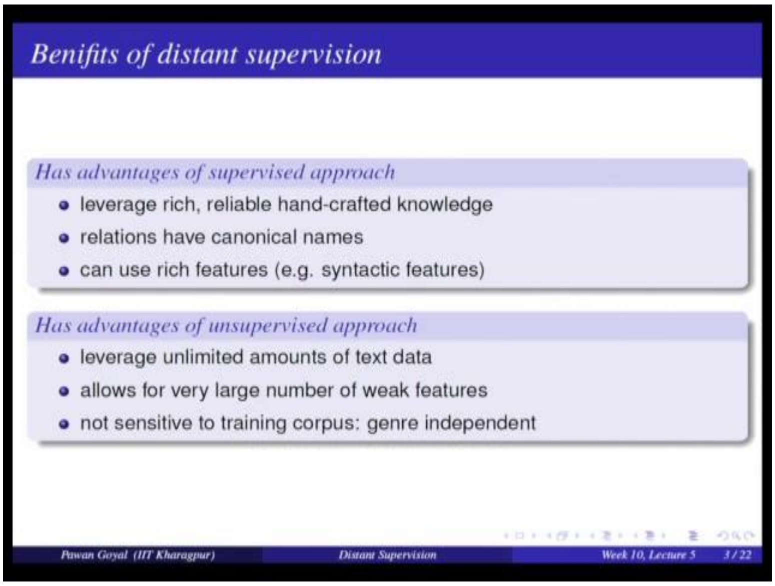
If we try to understand that from a block diagram, what we will do? So, we will have a database that will contain entity 1, entity 2 and the relation 1 entity; 1 prime entity. 2 prime relations 2 and so on and this is what you will have in the database then you will have a lot of corpus there are lot of sentences. So, what you will do you will use the hypothesis to say that wherever these entities are found in this corpus they are connected by this relation pi one. So, here you have the entity 1, entity 2, entity 1, entity 2, entity 1, entity 2, in these sentences.

Now, you start assuming that this is your labeled data set. So, now, you know this is your labeled data set. So, you will say you will label it by the label one or by this hypothesis this is relation one this is relation one this is your labeled data set now you do exactly what you do using supervised approach. So, we will try to use this is my relations now I will extract features from here sorry. So, that is what words come before what words come after what is the parse tree everything you can use. So, you use these features to build the classifier and that classifier when it runs over new sentences can give you labels these 2 entities are connected by this relation and so on.

So, what is so now, also we can see how it is different from bootstrapping, in bootstrapping you start with only a few seeds, now you go to the corpus, here you do not take the features in bootstrapping, you only take the patterns simple patterns and here not as good as using the features in a classifier and what you do in the case of supervised approach in supervised approach this has to be done manually you have to manually tag each sentence with the relation entities with the relation and this you are avoiding because by using database. So, this database is not there in the case of supervised approach. So, you directly go to the corpus you start labeling and use those labels to create the classifier.

So what? So, you see here trying to combine bootstrapping and supervised in a nice way such that you are avoiding the problems with each of these you are avoiding the problem with bootstrapping that is no problems with interpretation you do not know how that will do good or bad there are only a few seed tuples by taking a lot of examples; and doing with features instead of patterns and you are avoiding problem with supervised approaches by avoiding your labeling task and taking from database the labels. So, that is how you combine both of these approaches together.

(Refer Slide Time: 06:33)



Benifits of distant supervision

Has advantages of supervised approach

- leverage rich, reliable hand-crafted knowledge
- relations have canonical names
- can use rich features (e.g. syntactic features)

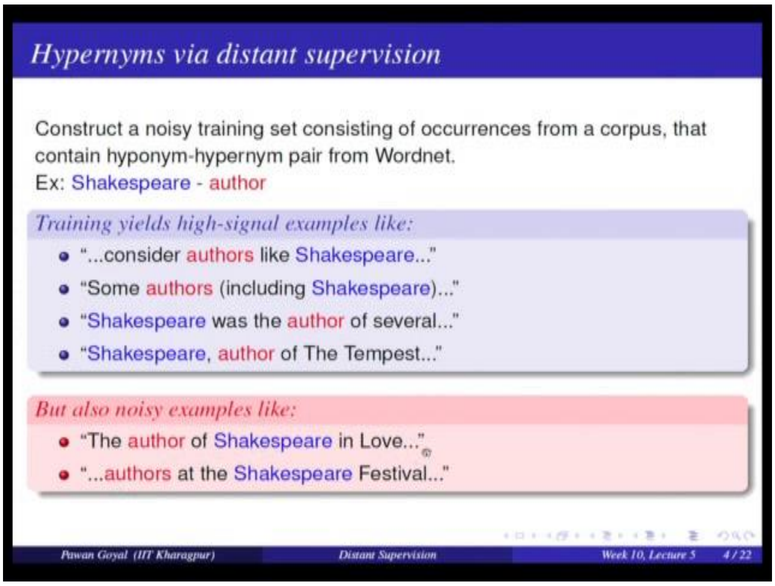
Has advantages of unsupervised approach

- leverage unlimited amounts of text data
- allows for very large number of weak features
- not sensitive to training corpus: genre independent

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 3 / 22

Now, let us say so this has advantage of supervised approach by leveraging rich handcrafted knowledge and you can use these features and also of the unsupervised approach because you can leverage on unlimited amount of text data, it may not been labeled then also you can use it in this approach and you can use a lot of different weak features. So, syntactic features lexical features and so on and you know it will not be sensitive to the training corpus. So, it will work on whatever corpus you are you are trying to use it on.

(Refer Slide Time: 07:06)



Hypernyms via distant supervision

Construct a noisy training set consisting of occurrences from a corpus, that contain hyponym-hypernym pair from Wordnet.
Ex: Shakespeare - author

Training yields high-signal examples like:

- "...consider authors like Shakespeare..."
- "Some authors (including Shakespeare)..."
- "Shakespeare was the author of several..."
- "Shakespeare, author of The Tempest..."

But also noisy examples like:

- "The author of Shakespeare in Love..."
- "...authors at the Shakespeare Festival..."

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 4 / 22

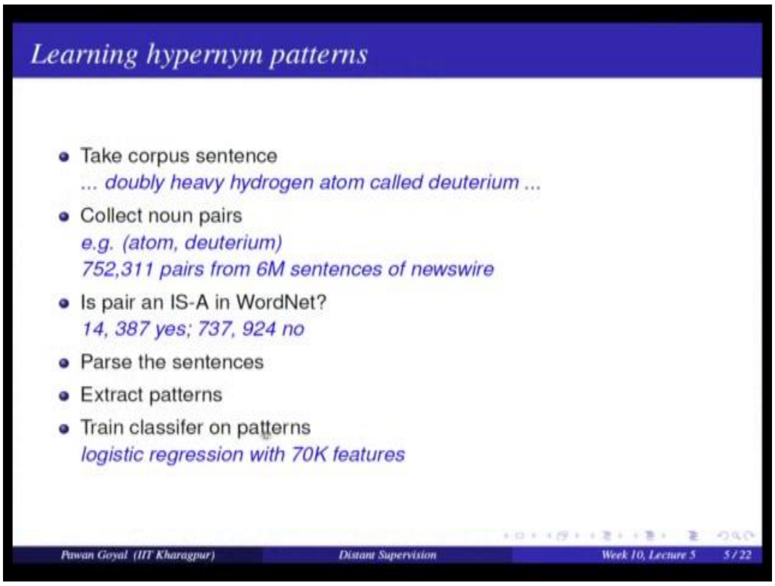
Let us take an example of finding Hypernyms via this distant supervision idea. So, here so what we will do? So, what is a database which will contain the Hypernym pairs and you can see wordnet is the database that contains super concept sub concept in a nice machine readable manner. So, I will try to use that database to find out more Hypernyms. So, let us take an example by a simple single Hypernym suppose one example is Shakespeare and author Shakespeare is an author. So, what I will do similar to bootstrapping I will go through my corpus and find out where all these entities occur together in a sentence.

Suppose I find out some sentences like this consider authors like Shakespeare some authors including Shakespeare, Shakespeare was the author of several Shakespeare the author of temptation so on so, these are the very sentences where I am finding these entities and I will immediately label them with the relation of hypernym that entity one Shakespeare here related to entity 2 author by this relation of hypernym. So, this labeling can be done automatically now because I have these relations and the hypothesis.

Now, one problem here might be that there are some noisy examples like the author of Shakespeare in love authors at the Shakespeare festival. So, can you see why this is noisy? So, author and Shakespeare both words occurring together, but they are not occurring in the relation of hypernym you see I am talking about here the author of a book Shakespeare in love it is not talking about Shakespeare being an author similarly here some authors at a festival we are not talking about Shakespeare being an author. So, what will happen suppose you take it as your pattern authors of Shakespeare or authors at the Shakespeare festival if you take it take these as your pattern if you try to extract new entities they will they will not be correct. So, think about a pair of words that occur with this relation or this pattern.

This can be someone was at somewhere. So, it does not say that this is sub concept super concept relation, no. So, this is your noisy example now how will my distant supervision handle these noisy examples. So, in the case of bootstrapping there is no easy way to handle these examples, but here what I will do because I have many many seed pairs for this relation. So, this noisy example will appear in may be one or 2 cases, but they will not occur in most of the cases and that is what I will make use of. So, they will not they will occur may be 1 or 2; 2 hypernym; hypernym pairs, but they are not occurring with many other pairs and if I take random pairs they might occur with them. So, so that is why these noisy examples will not detoriate the performance of my system.

(Refer Slide Time: 10:14)



Learning hypernym patterns

- Take corpus sentence
  ... doubly heavy hydrogen atom called deuterium ...
- Collect noun pairs
  e.g. (atom, deuterium)
  752,311 pairs from 6M sentences of newswire
- Is pair an IS-A in WordNet?
  14, 387 yes; 737, 924 no
- Parse the sentences
- Extract patterns
- Train classifer on patterns
  logistic regression with 70K features

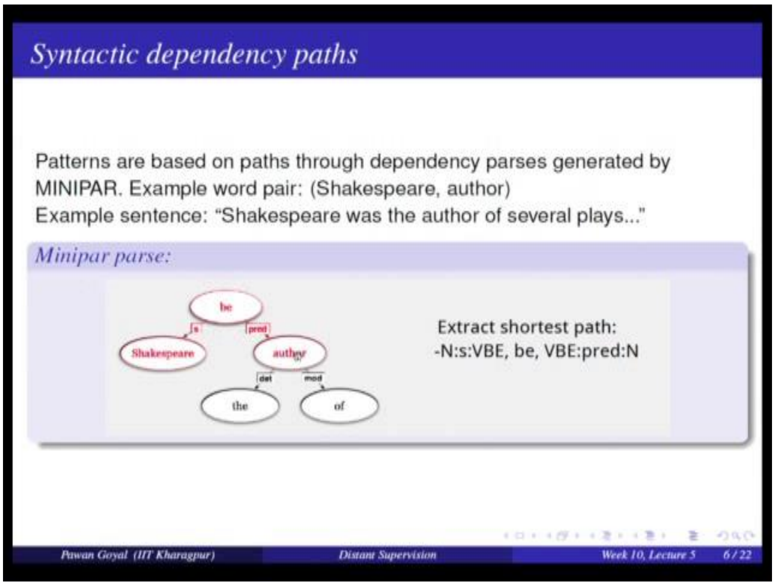
Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 5 / 22

How do I learn the hypernym patterns? I take corpus sentence like doubly heavy nitro hydrogen atom called deuterium and I have the entities like various nouns like atom deuterium etcetera. So, so if I take. So, here what was done 6 million sentences were taken from newswire and 752311 pairs were formed of entities noun pairs. So, this is starting from the data. So, you have a lot of data from that you are finding noun pairs and you have found you are seeing the 750000 noun pairs.

Now, to use distant supervision you have to see some of these noun pairs are already involved in hypernym relation. So, you can use wordnet to find out how many of these are having this relation. So, suppose from there you find out fourteen thousand roughly are having this relation and remaining seven thirty seven thousand do not have this relation. So, immediately you have the positive examples and the negative examples from the data.

Wherever one of these 15000 entities occurs, you will say these are connected by the relation of hypernym and wherever these entities occur they are there is no hypernym relation between these entities and this you can use to build your classifier and extract various features. So, you parse the sentence extract patterns build various features and train classifier on these patterns and suppose, in this experiment people so they run logistic regression classifier with 70000 different features that was considered using patterns and everything else.

(Refer Slide Time: 12:08)



Syntactic dependency paths

Patterns are based on paths through dependency parses generated by MINIPAR. Example word pair: (Shakespeare, author)
Example sentence: “Shakespeare was the author of several plays...”

Minipar parse:

Extract shortest path:
-N:s:VBE, be, VBE:pred:N

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 6 / 22

Now patterns are so in this approach what the patterns? They used were based on a specific dependency parse it is called Minipar parse. So, so this will tell you how you can find out nice patterns and use them as your features also and. So, what is the idea for using Minipar for considering your patterns as features? So, suppose we have the sentence Shakespeare was the author of several plays and you want to find out a feature between the 2 entities Shakespeare and author you want to find a feature between these 2 entities.

What you will do? You will run this parser; this Minipar parser and you will get the parse of this sentence. So, you have this parse of the sentence. So, you have you have the lemmas here Shakespeare author of and be and they are connected by various relations different words are connected now you want to extract a relation between or a pattern that connects Shakespeare and author. So, what is the idea you will abstract towards the words that are that you want to connect because you want to find out what are the other words that are connected by the same sort of pattern. So, you will extract over Shakespeare and author call them x and y.

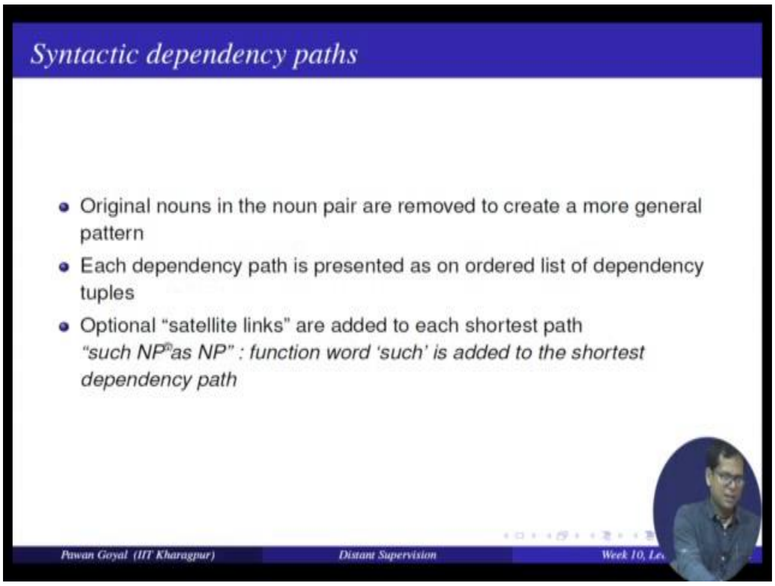
You are starting from Shakespeare. So, you will see what is the path that connects the 2? And you are going in the opposite direction here. So, you are seeing, what is this path?

What is the first dependency relation that connects Shakespeare and in Minipar? What happens, you write down what is the part of speech of the first word, the relation and part of speech of the next word that is how the relations are labeled the relations are labeled.

Here Shakespeare is a noun, be is a verb and the relation is subject and you are going in the opposite direction. So, you are having a minus. So, you denoting this relation by minus noun subject and VBE, VBE is the part of speech which tags for be then you will take the intermediate words. So, you will take the word be here then you have another path similar again take v b e predicate and the noun and you will stop here because you want to connect Shakespeare and author. So, by this simple approach you can try to find out patterns between any 2 possible words from the dependency parser.

What they did? They found out all the possible patterns between different words in the corpus and so, there was 70 k; 70 k different patterns that the users feature for training their dependency curve for sorry not dependency curve, but their logistic regression for finding out if 2 words have a relation of hypernym.

(Refer Slide Time: 15:02)



Syntactic dependency paths

- Original nouns in the noun pair are removed to create a more general pattern
- Each dependency path is presented as on ordered list of dependency tuples
- Optional “satellite links” are added to each shortest path
  “such NP as NP” : function word ‘such’ is added to the shortest dependency path

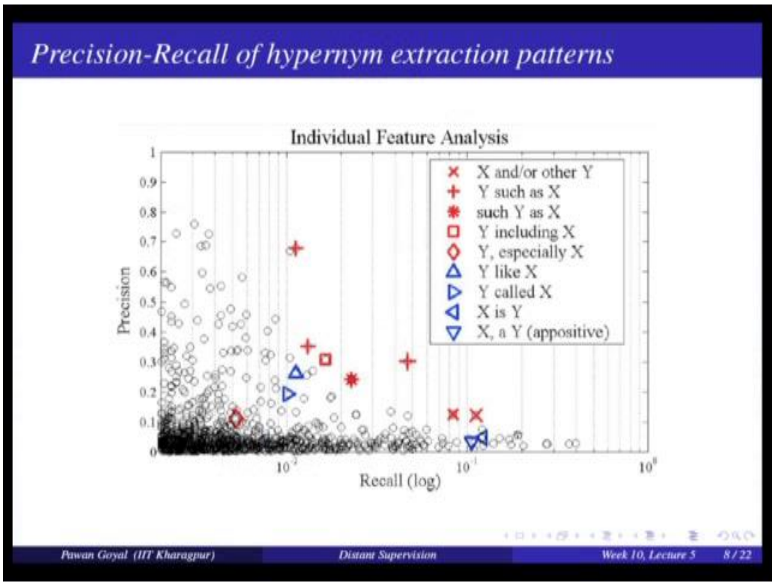
Pawan Goyal (IIT Kharagpur)   Distant Supervision   Week 10, Le

Now, what was so and they did some more tricks like as we discussed original nouns in the noun pair were removed to create a more general pattern and each dependency path is presented in ordered list of dependency tuples that you were seeing and sometimes they have some satellite links like such and as.

What is found when you are building patterns function words are very important to the patterns, but the content words are not so important. So, wherever you have a pattern where you have a word like such there is a noun phrase as noun phrase. So, what you will do? You will retain the function words, but the noun phrases you will put them as noun phrases instead of the word. So, instead of saying such red algae as Gelidium, you will say such NP as NP and such and as can remain as it is because these function words are very helpful in describing your pattern and this is a nice trick that you can use for building your patterns. So, abstract over the content words and retain the function words.

Even content words you might take some content words that are having very high frequency because then you might find them more often, but if something is not having high frequency there is no point in keeping the word as it is better to abstract, it as a part of speech tag or the particular phrase and so on and this helps a lot in building these patterns and this you can use as your features or whatever.

(Refer Slide Time: 16:39)



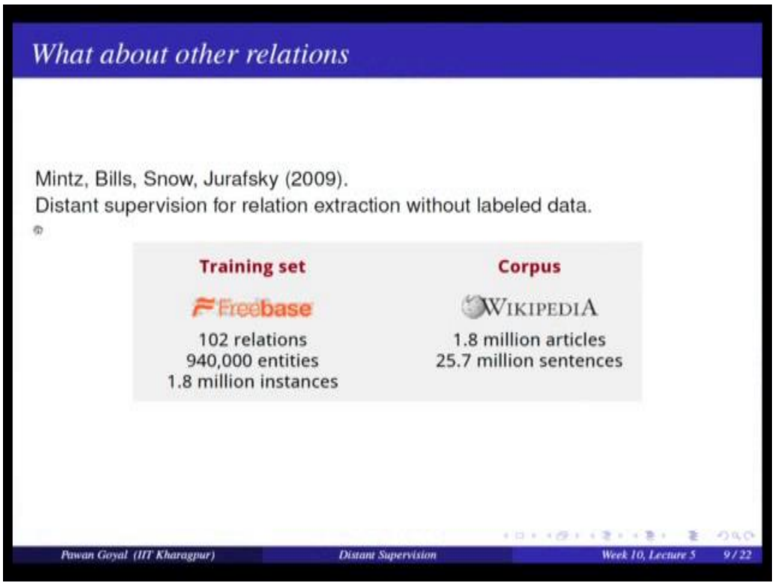
Precision-Recall of hypernym extraction patterns

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 8 / 22

Now once they did that so this in this figure, what you are seeing? Individual feature analysis, so if you take only one feature at a time what is the precision recall that you get on your data and you are having. So, these are like seventy k different features and you are seeing most of the features had very poor precision recall independently, but there are some features that is stood out that were giving very high precision recall and when they analyzed they found something very interesting.

These are the features that were having nice recall in precision and these are the features X and the other Y is such as X such Y as X Y including X Y especially X now do you remember these features. So, when we talked about the patterns that Hearst felt manually he was using these features. So, so these features you may not be identified in this approach we never gave these features manually, but the algorithm was able to identify these features automatically what were the some other features like Y like X Y called X, X is Y X A Y. So, these were also some other features that Hearst did not find, but algorithm was finding and if you show them to various linguists or to various people they will say oh yes these features make sense for this particular task. So, this was very very interesting about this approach.

(Refer Slide Time: 18:08)



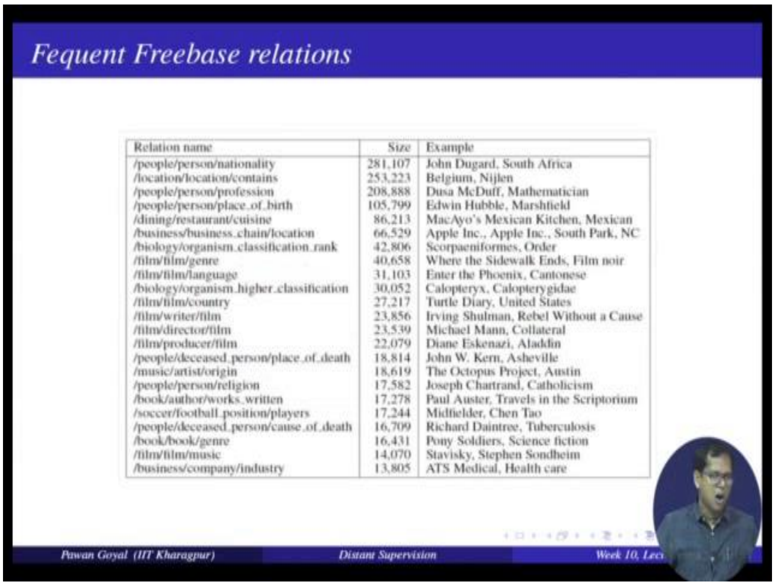
What about other relations

Mintz, Bills, Snow, Jurafsky (2009).
Distant supervision for relation extraction without labeled data.

Training set	Corpus
Freebase	WIKIPEDIA
102 relations 940,000 entities 1.8 million instances	1.8 million articles 25.7 million sentences

Now this was just for a single relation hypernym but suppose you want to do that for many many different relations at the same. So, together I want to train them together how will I do that. So, that is where in 2 thousand nine this was this paper by the Jurafskys group distant supervision for relation extraction without labeled data. So, what they did in place of the corpus they used Wikipedia. So, that had at that time 1.8 million articles and 25.7 million sentences that was there unlabeled data set what was their database containing the relations they used freebase now freebase is a very large repository of various entity pairs and their relations. So, that time it contained 102 relations 940000 entities and 1.8 million instances that many different pairs of entities are present with their labeled relations.

(Refer Slide Time: 19:04)



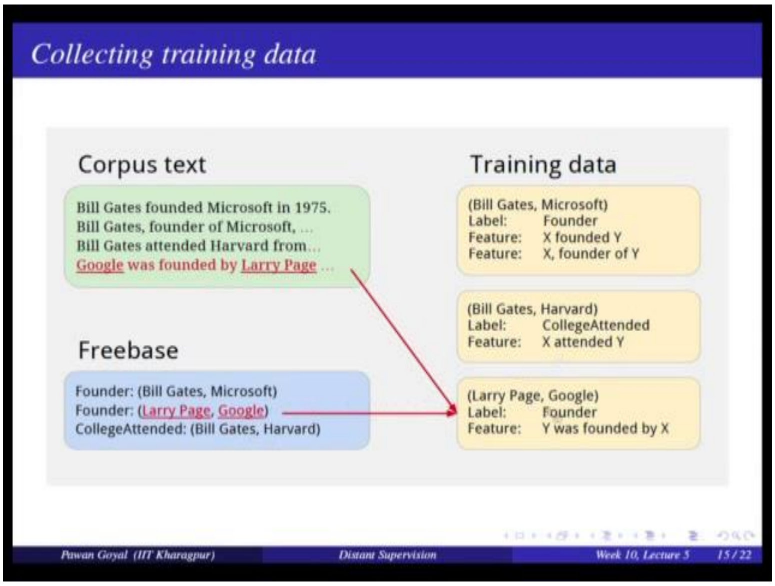
Fequent Freebase relations

Relation name	Size	Example
/people/person/nationality	281,107	John Dugard, South Africa
/location/location/contains	253,223	Belgium, Nijlen
/people/person/profession	208,888	Dusa McDuff, Mathematician
/people/person/place_of_birth	105,799	Edwin Hubble, Marshfield
/dining/restaurant/cuisine	86,213	MacAyo's Mexican Kitchen, Mexican
/business/business_chain/location	66,529	Apple Inc., Apple Inc., South Park, NC
/biology/organism_classification_rank	42,806	Scorpaeniformes, Order
/film/film/genre	40,658	Where the Sidewalk Ends, Film noir
/film/film/language	31,103	Enter the Phoenix, Cantonese
/biology/organism_higher_classification	30,052	Calopteryx, Calopterygidae
/film/film/country	27,217	Turtle Diary, United States
/film/writer/film	23,856	Irving Shulman, Rebel Without a Cause
/film/director/film	23,539	Michael Mann, Collateral
/film/producer/film	22,079	Diane Eskenazi, Aladdin
/people/deceased_person/place_of_death	18,814	John W. Kern, Asheville
/music/artist/origin	18,619	The Octopus Project, Austin
/people/person/religion	17,582	Joseph Chartrand, Catholicism
/book/author/works_written	17,278	Paul Auster, Travels in the Scriptorium
/soccer/football_position/players	17,244	Midfielder, Chen Tao
/people/deceased_person/cause_of_death	16,709	Richard Daintree, Tuberculosis
/book/book/genre	16,431	Pony Soldiers, Science fiction
/film/film/music	14,070	Stavisky, Stephen Sondheim
/business/company/industry	13,805	ATS Medical, Health care

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lec

Now, how does relations look like in freebase? So, you will have a relation name. So, in the category of people there is a person and nationality that is how you need to read it there are 2 eighty one thousand relations and 1 example John Dugard is the person with nationality South Africa, similarly location contains Belgium, contains Nijlen person profession Mister McDuff is mathematician person, place of birth, Edwin Hubble born in Marshfield. So, like that this freebase contains all these relations and what are the entities that are connected by these relations. So now so what is their task using these whole data sets try to come up with the algorithm classifier such that you can populate this database so that you can find out new and new entity pairs that are connected by relations. So, what is the idea? What they do?

(Refer Slide Time: 20:00)



Collecting training data

Corpus text

Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, ...
Bill Gates attended Harvard from...
Google was founded by Larry Page ...

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)

Training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y
Feature: X, founder of Y

(Bill Gates, Harvard)
Label: CollegeAttended
Feature: X attended Y

(Larry Page, Google)
Label: Founder
Feature: Y was founded by X

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 15 / 22

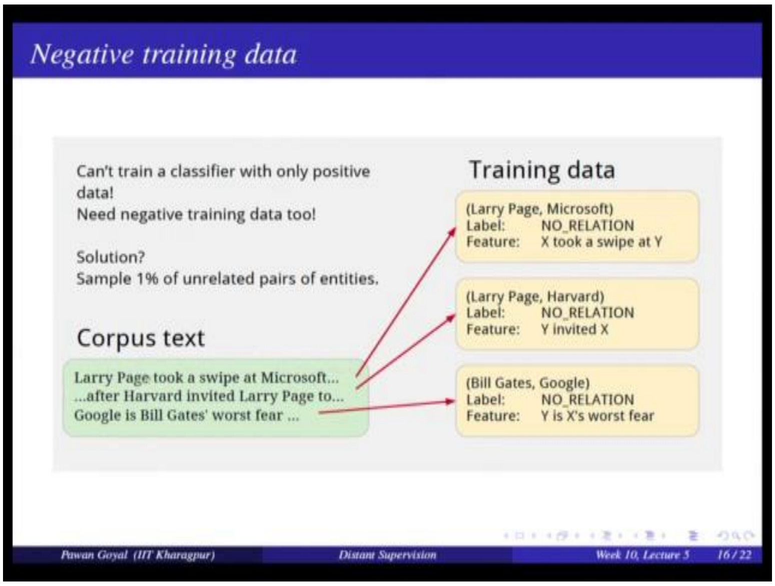
Here is the simple illustration. So, you have this corpus text like Wikipedia. So, there are various sentences that are here then you have your freebase that contains the relation and entities that are connected by the relation like founder Bill Gates founded Microsoft, Larry Page founded Google, Bill Gates attended Harvard. So, various relations and the entities, now how do you make use of both of these? You will take; you will either take the relations here and go through the corpus and find out wherever they occur together in the single sentence. So, you will say Bill Gates and Microsoft occur together in this sentence. So, you will say; that means, a Bill Gates and Microsoft are the entity pairs they occur in the sentence. So, I gave a label as founder and then I extract various features.

Here for simplicity we are only showing one simple pattern X founded by, but in general you can think of any other feature like what is the next word to Y it is in and what, what occurs between X and Y is X starting the sentence. So, like that you can have many many different features. So, you are giving a very simple feature here X founded Y as the pattern. So, so that is what you are doing for each entity pair you are going to corpus selecting sentences where they occur using the hypothesis they might be connected by the same relation labeling relation and extracting features.

Another sentence it occurs, so you will extract features X founder of Y now suppose you go to Bill Gates, Harvard, you have this sentence and you will extract the feature X attended by relation college attended similarly here Larry Page Google. So, founder Y was founded by X. So, by different sentences you will try to you will be able to get different sort of features and different sort of patterns now these are all positive examples of various relations you might also have to get some negative examples that these 2 entities are never connected by any relation now how.

It is interesting that how do you construct negative examples in this distant supervision approach and the idea is also very interesting that you have your database you know what entities are connected, but from your database also sample some pair of entities that are not connected by any relation. So, once you sample these entities find out the sentences in the in the in the corpus where these are appearing together and label this as no relation because we know there is no relation in the in the freebase among these entities and this will generate your negative data. So, this is your positive data and then you will get some negative data also. So, you cannot really classify with only positive data. So, you need some negative trained data.

(Refer Slide Time: 22:57)



Negative training data

Can't train a classifier with only positive data!
Need negative training data too!

Solution?
Sample 1% of unrelated pairs of entities.

Corpus text

Larry Page took a swipe at Microsoft...
...after Harvard invited Larry Page to...
Google is Bill Gates' worst fear ...

Training data

(Larry Page, Microsoft)
Label: NO_RELATION
Feature: X took a swipe at Y

(Larry Page, Harvard)
Label: NO_RELATION
Feature: Y invited X

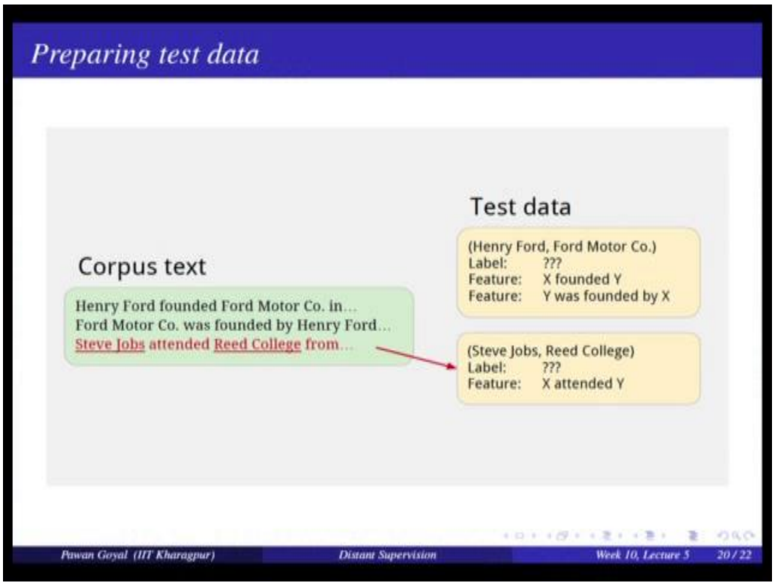
(Bill Gates, Google)
Label: NO_RELATION
Feature: Y is X's worst fear

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 16 / 22

What you do? Sample 1 percent of unrelated pairs of entities, 1 percent because otherwise there will be too much negative data. So, you do not want to make a classifier where it is more biased towards the negative examples. So, we take some sample you under sample the negative cases and here is how you will do that.

Suppose you find out in your corpus on your sorry in your database there is no relation between Larry Page and Microsoft. So, you will find out the sentence like Larry Page took a swipe at Microsoft and you will say with this feature the label is no relation X took a swipe at Y no relation here Harvard invited Larry Page to something. So, Y invited X no relation and so on we will do for all these pair of entities.

(Refer Slide Time: 23:51)



Preparing test data

Corpus text

Henry Ford founded Ford Motor Co. in...
Ford Motor Co. was founded by Henry Ford...
Steve Jobs attended Reed College from...

Test data

(Henry Ford, Ford Motor Co.)
Label: ???
Feature: X founded Y
Feature: Y was founded by X

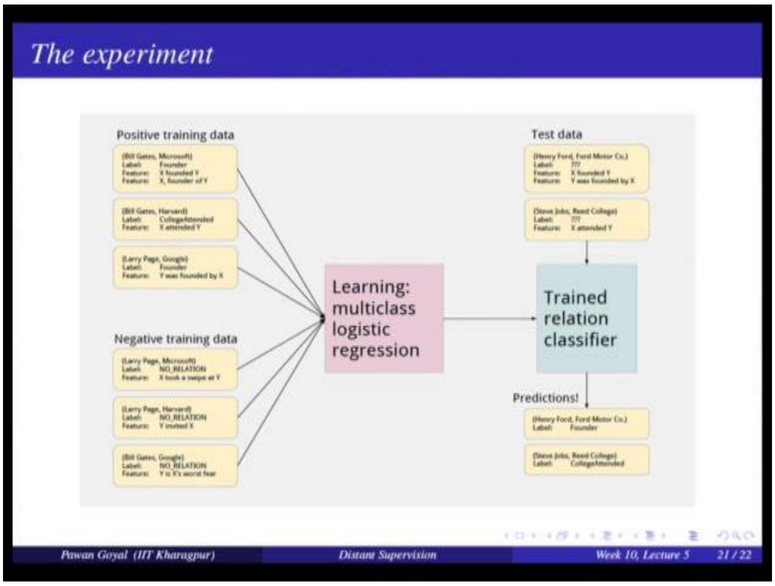
(Steve Jobs, Reed College)
Label: ???
Feature: X attended Y

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 20 / 22

You have now got and see here what will happen in the testing? Testing, you will have the corpus text. So, you will have the sentence here you find out the entities and then you will parse it through your classifier to find out is there a relation between these 2.

You have the feature like X founded Y for the entities, but you did not know the label in the text side. So, you will use the classifier to find out with this feature is there a relation between them and if. So, what is the relation and you will say similarly here Y was founded by X, what will be the label X attended Y? What will be the label and the classifier would be able to tell this will be college attended this will be founder of and so on.

(Refer Slide Time: 24:37)



The experiment

Positive training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y
Feature: X, founder of Y

(Bill Gates, Harvard)
Label: CollegeAttended
Feature: X attended Y

(Larry Page, Google)
Label: Founder
Feature: Y was founded by X

Negative training data

(Larry Page, Microsoft)
Label: NO_RELATION
Feature: X took a swipe at Y

(Larry Page, Harvard)
Label: NO_RELATION
Feature: Y invited X

(Bill Gates, Google)
Label: NO_RELATION
Feature: Y is X's worst fear

Learning: multiclass logistic regression

Trained relation classifier

Test data

(Henry Ford, Ford Motor Co.)
Label: ???
Feature: X founded Y
Feature: Y was founded by X

(Steve Jobs, Reed College)
Label: ???
Feature: X attended Y

Predictions!

(Henry Ford, Ford Motor Co.)
Label: Founder

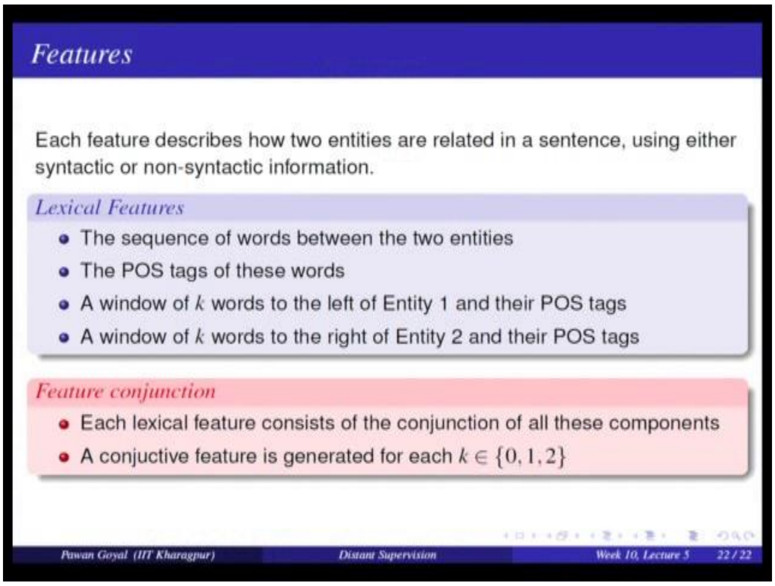
(Steve Jobs, Reed College)
Label: CollegeAttended

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 21 / 22

In a single figure if you want to see so we will have some positive training data you will have some negative training data you will use this to learn your logistic regression and here is a multi class where no relation is also a class and you have many different relations at test time what will happen for each sentence you will take the entity pairs you will try to identify the features and you will feed it to your classifier to find out with these features what is the relation between these 2 entities.

Here suppose the predictions are Henry Ford and Ford motor company, the label is founder Steve Jobs and Reed College, the label is college attended, now what are the kind of features you can use? So, all these we have talked about the features buts.

(Refer Slide Time: 25:23)



Features

Each feature describes how two entities are related in a sentence, using either syntactic or non-syntactic information.

Lexical Features

- The sequence of words between the two entities
- The POS tags of these words
- A window of k words to the left of Entity 1 and their POS tags
- A window of k words to the right of Entity 2 and their POS tags

Feature conjunction

- Each lexical feature consists of the conjunction of all these components
- A conjuctive feature is generated for each $k \in \{0, 1, 2\}$

Pawan Goyal (IIT Kharagpur) Distant Supervision Week 10, Lecture 5 22 / 22

But you can use features like. So, they should tell how the 2 entities are related in the sentence. So, you can use lexical features in terms of only the words you can you use syntactic features in terms of what their connections you can use dependency paths like the example we discussed you can use gazetteers named entities lot of different things that you can make use of.

Lexical features can be like what are the sequence of words between the 2 entities parts of speech tags of these words a window of k words of the left of entity one their part of speech tags a window of k words to the right of entity 2 their part of speech tags and you can also combine these features I will take part of speech tag of the previous word. And the part of speech tag of the next word I can also try to combine these features and generate various conjunctive features and then use a use lot of syntactic features and once you identified these features use that to train your regression classifier and get the output.

That is where we will end this week. So, we talked about some interesting application on entity linking and information extraction relation extraction and we saw that they are nice ways in which you can make use of lot of data that is available on the web and put it in a more usable manner, so entity linking so that you know given an entity what particular concept it corresponds to in a database in information extraction you can find out given entities - what is a relation in which they are what is the relation that connects these 2 entities and you can populate a knowledge base using that and these are some of the very nice and important applications used in text mining. So with that we end this week and I will then see you in the next week.

Thank you.