Distant Supervision

Pawan Goyal

CSE, IIT Kharagpur

Week 10, Lecture 5

Distant supervision paradigm

Hypothesis

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

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)

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

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)

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

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

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..."

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..."

Learning hypernym patterns

- Take corpus sentence

  ... doubly heavy hydrogen atom called deuterium ...

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

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

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

Syntactic dependency paths

Patterns are based on paths through dependency parses generated by MINIPAR.

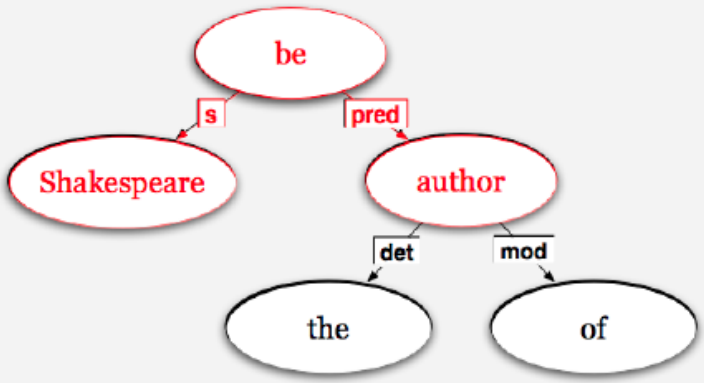
Syntactic dependency paths

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...”

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:

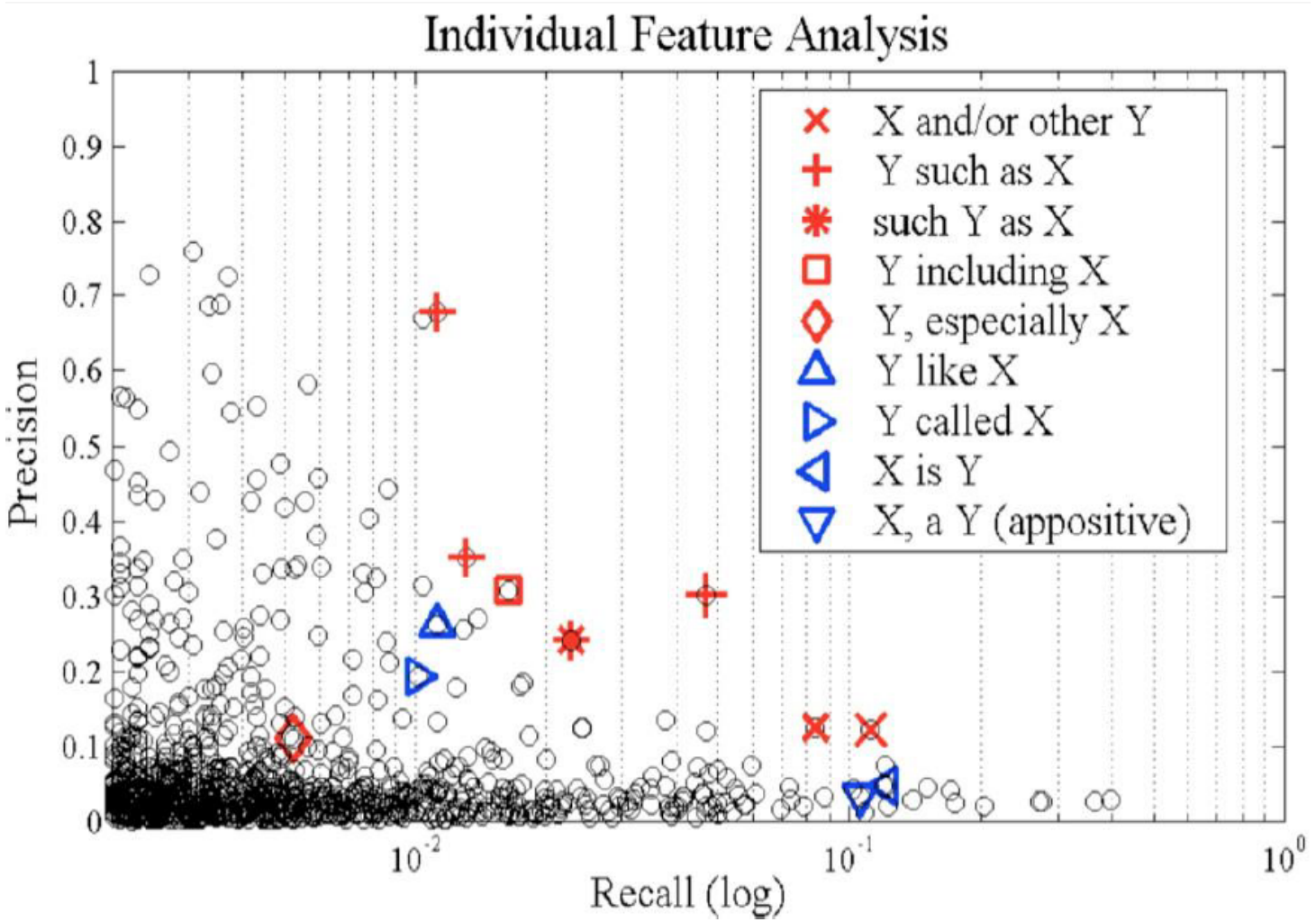


Extract shortest path:
-N:s:VBE, be, VBE:pred:N

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

Precision-Recall of hypernym extraction patterns



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

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

Collecting training data

Corpus text

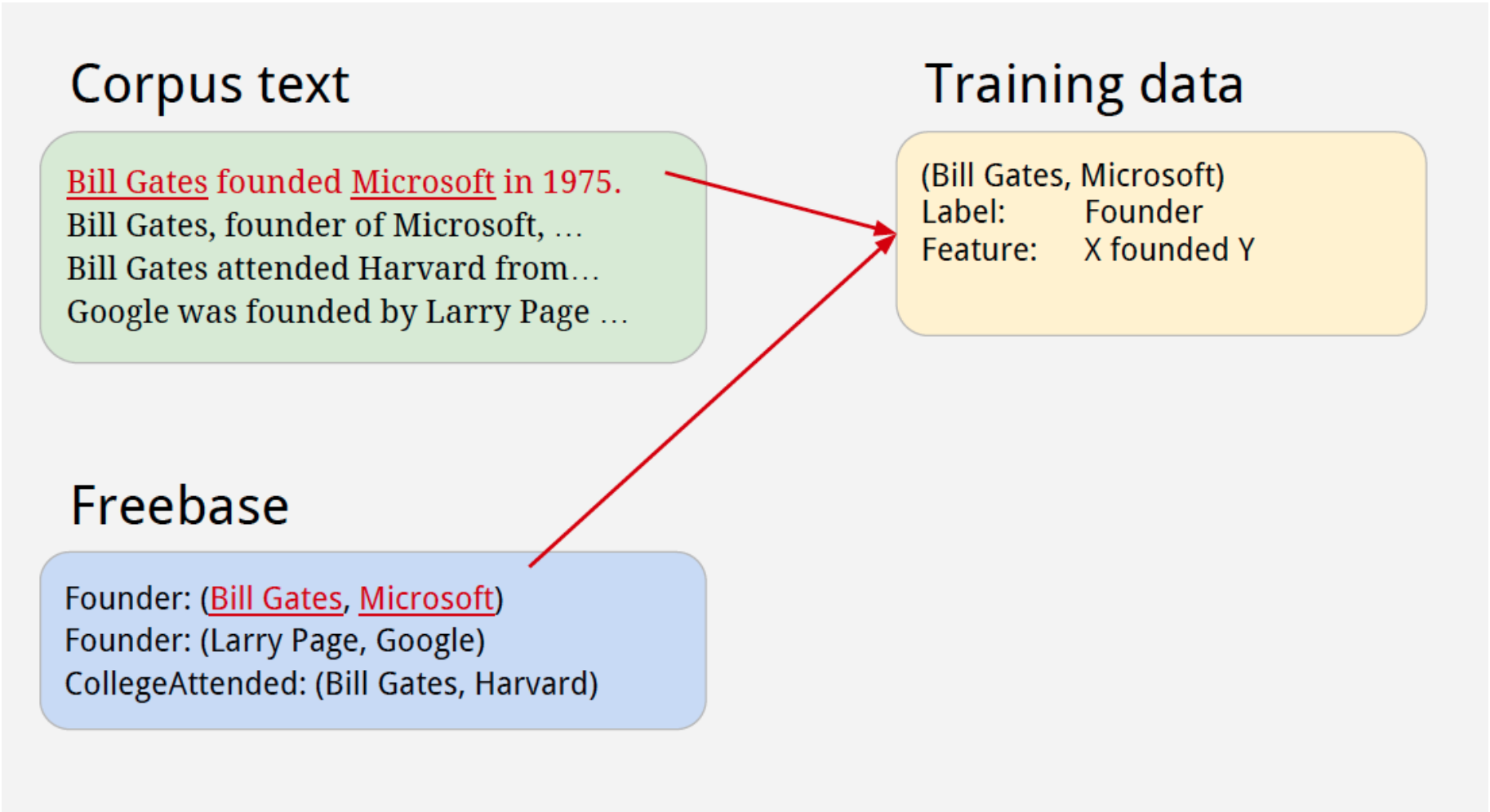
Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Training data

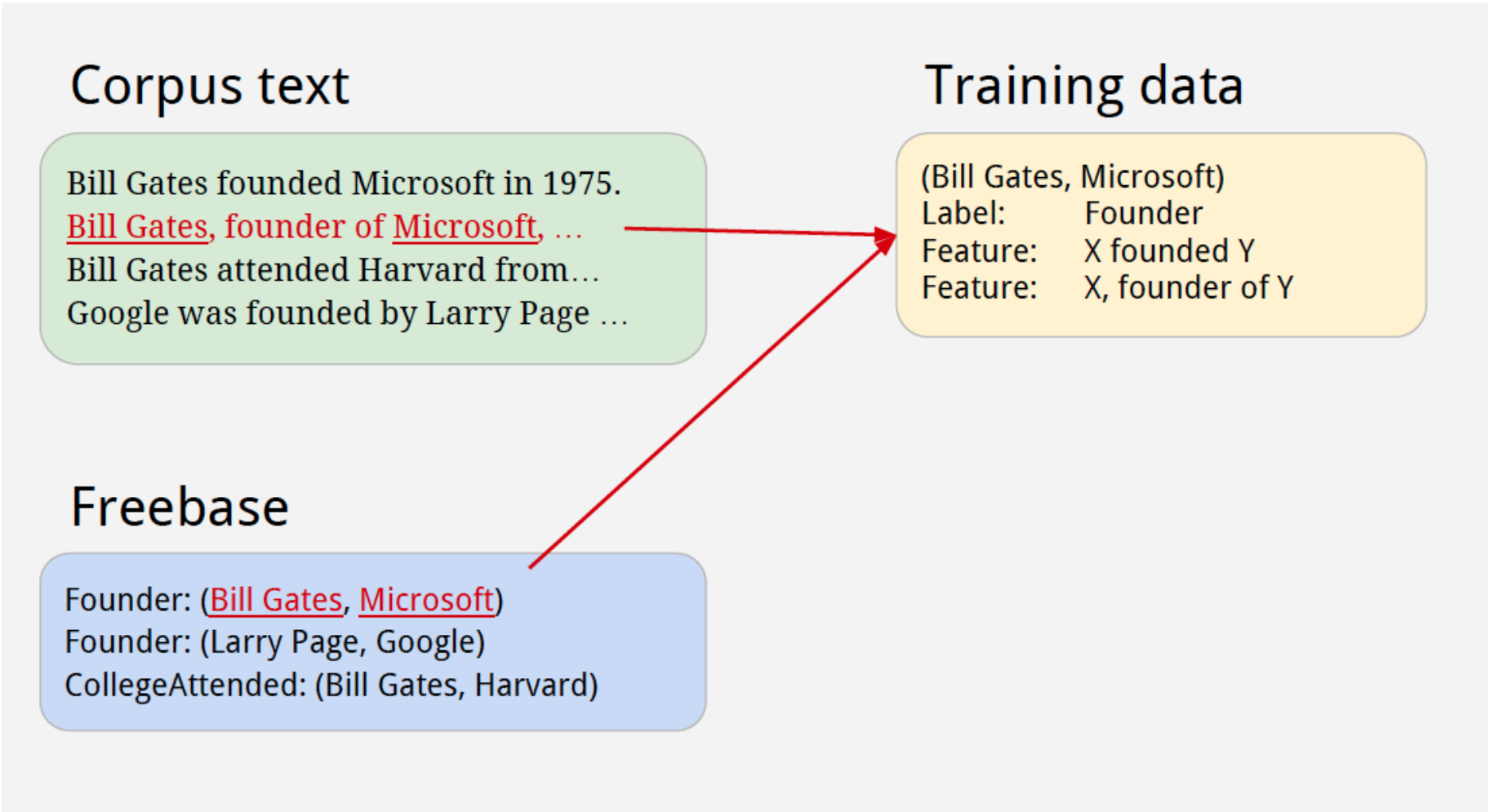
Freebase

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

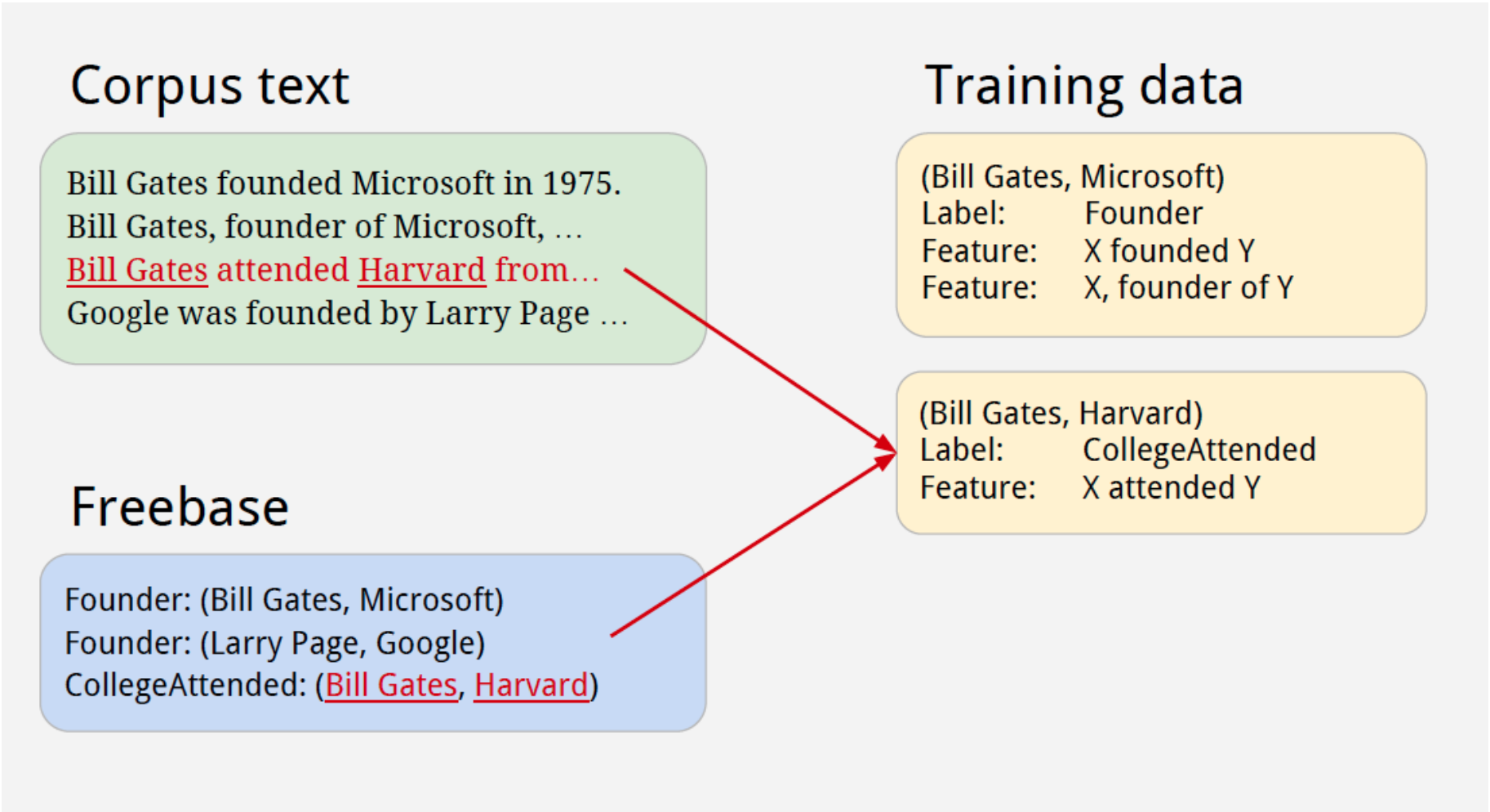
Collecting training data



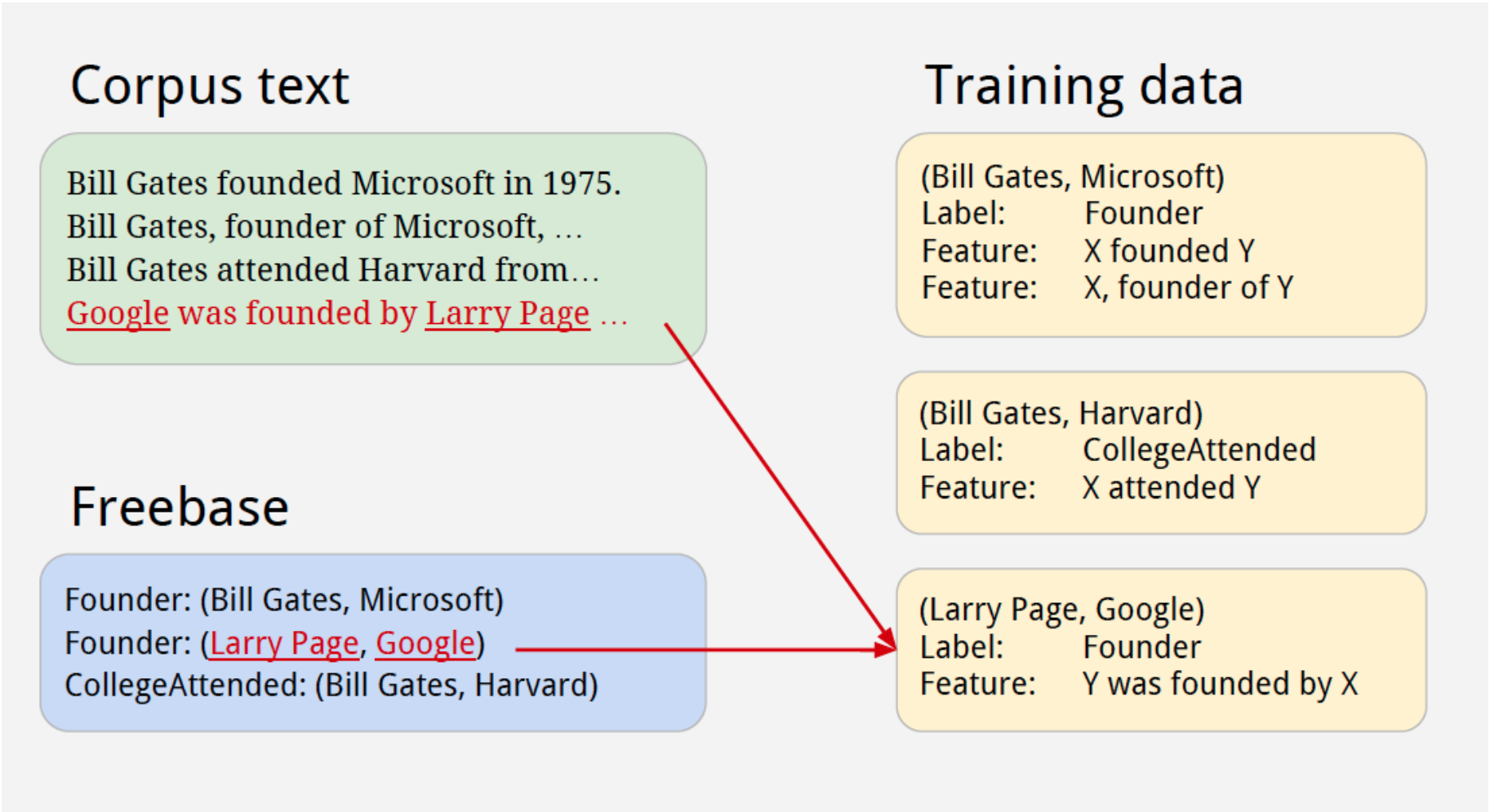
Collecting training data



Collecting training data



Collecting training data



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 ...

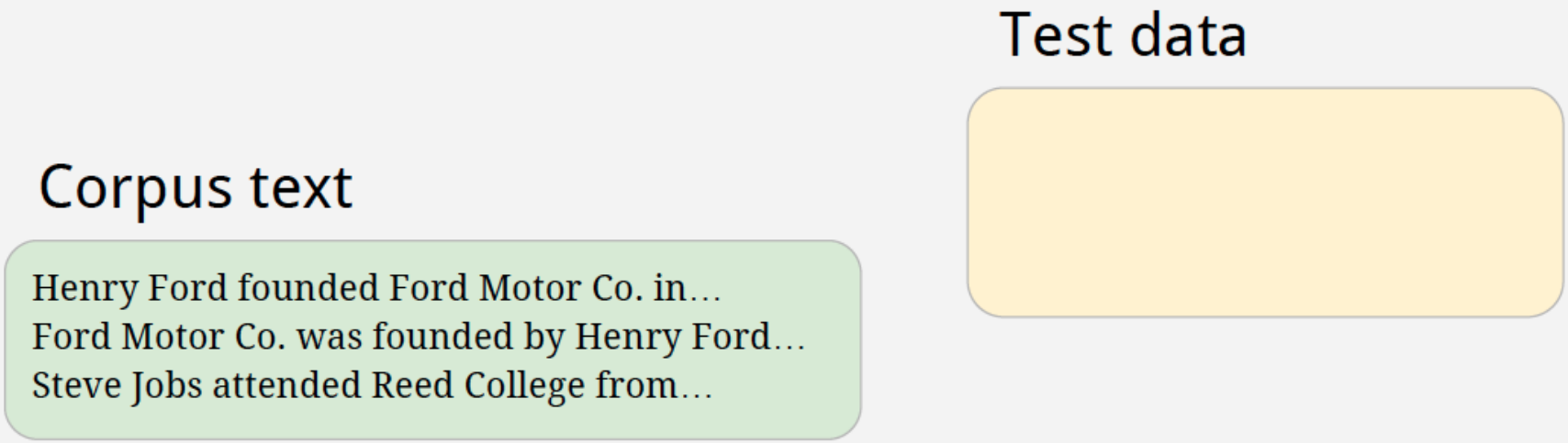
Training data

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

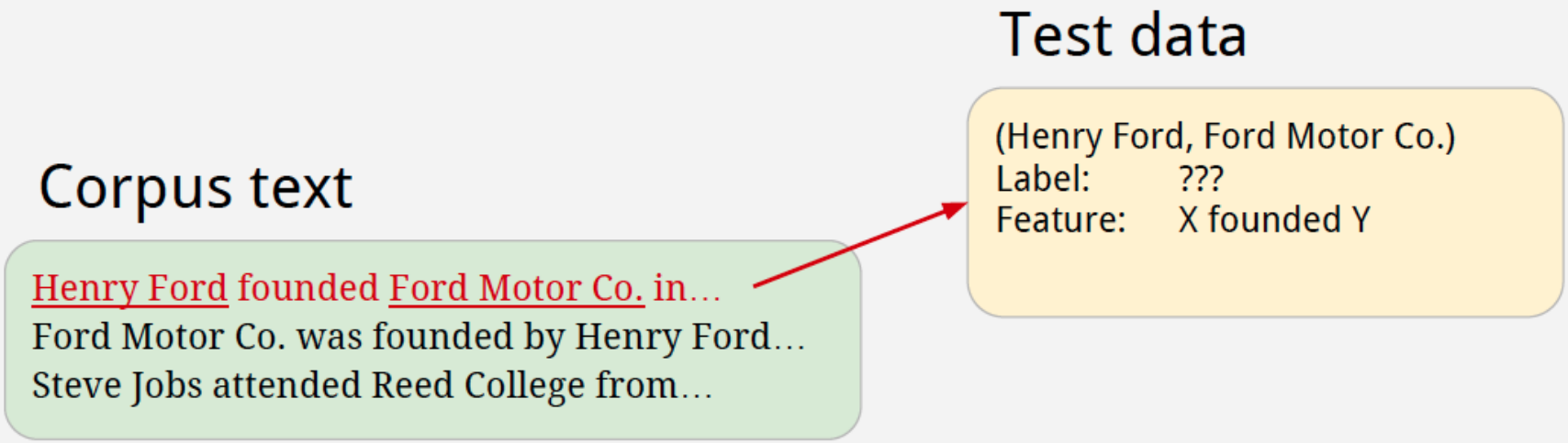
(Larry Page, Harvard)
Label: NO_RELATION
Feature: Y invited X

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

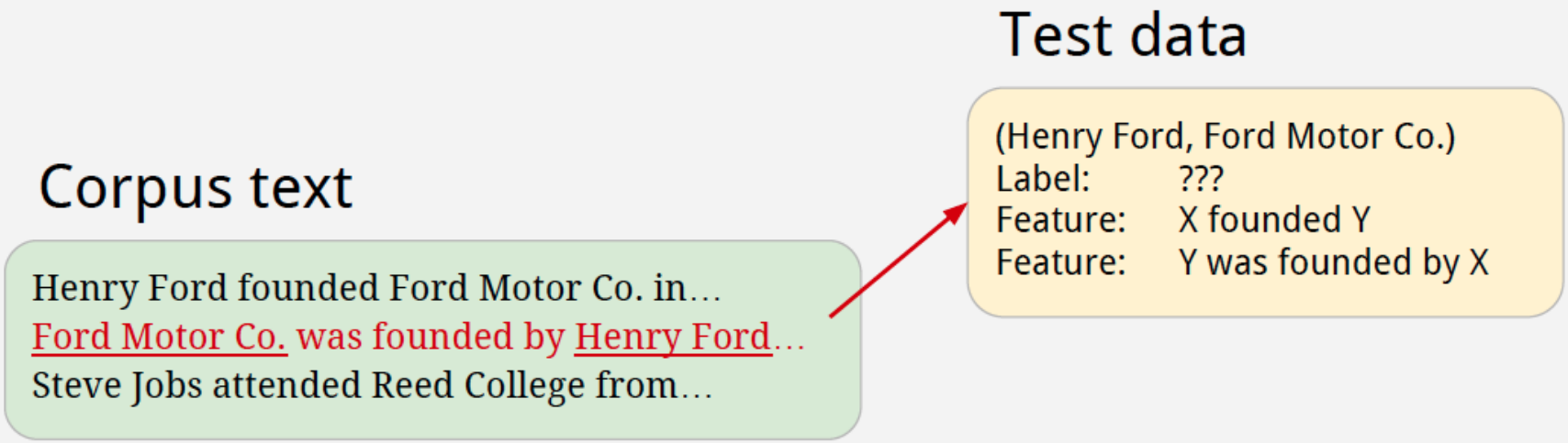
Preparing test data



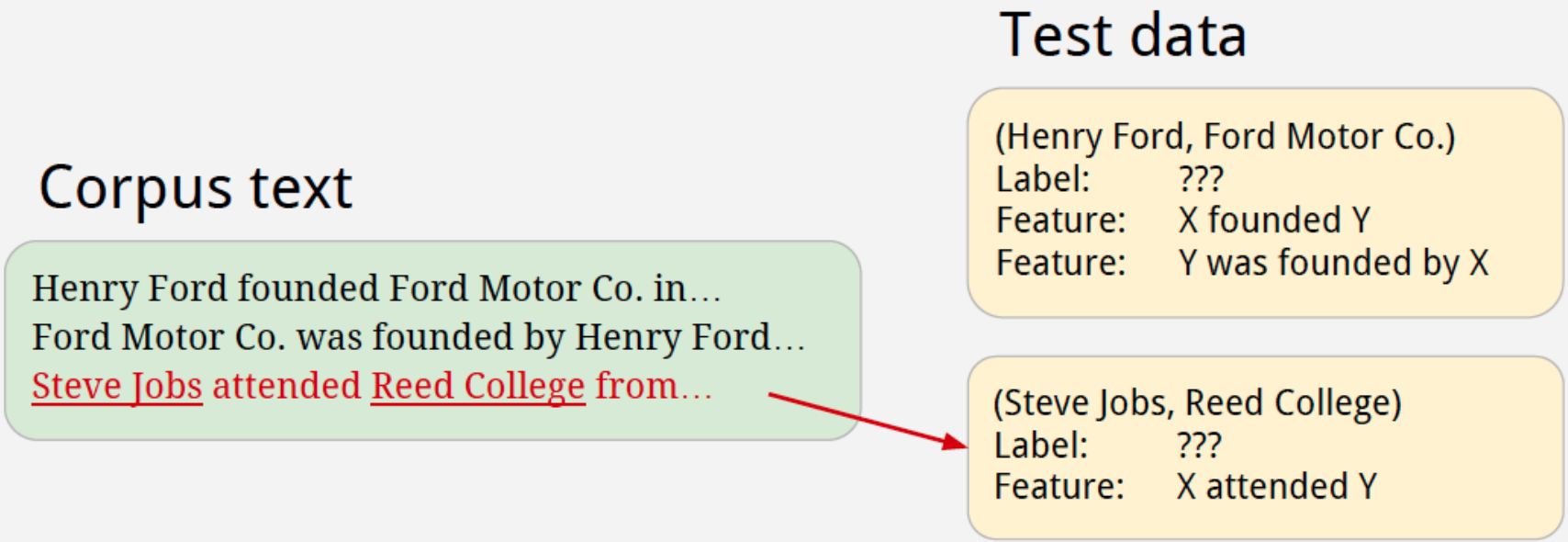
Preparing test data



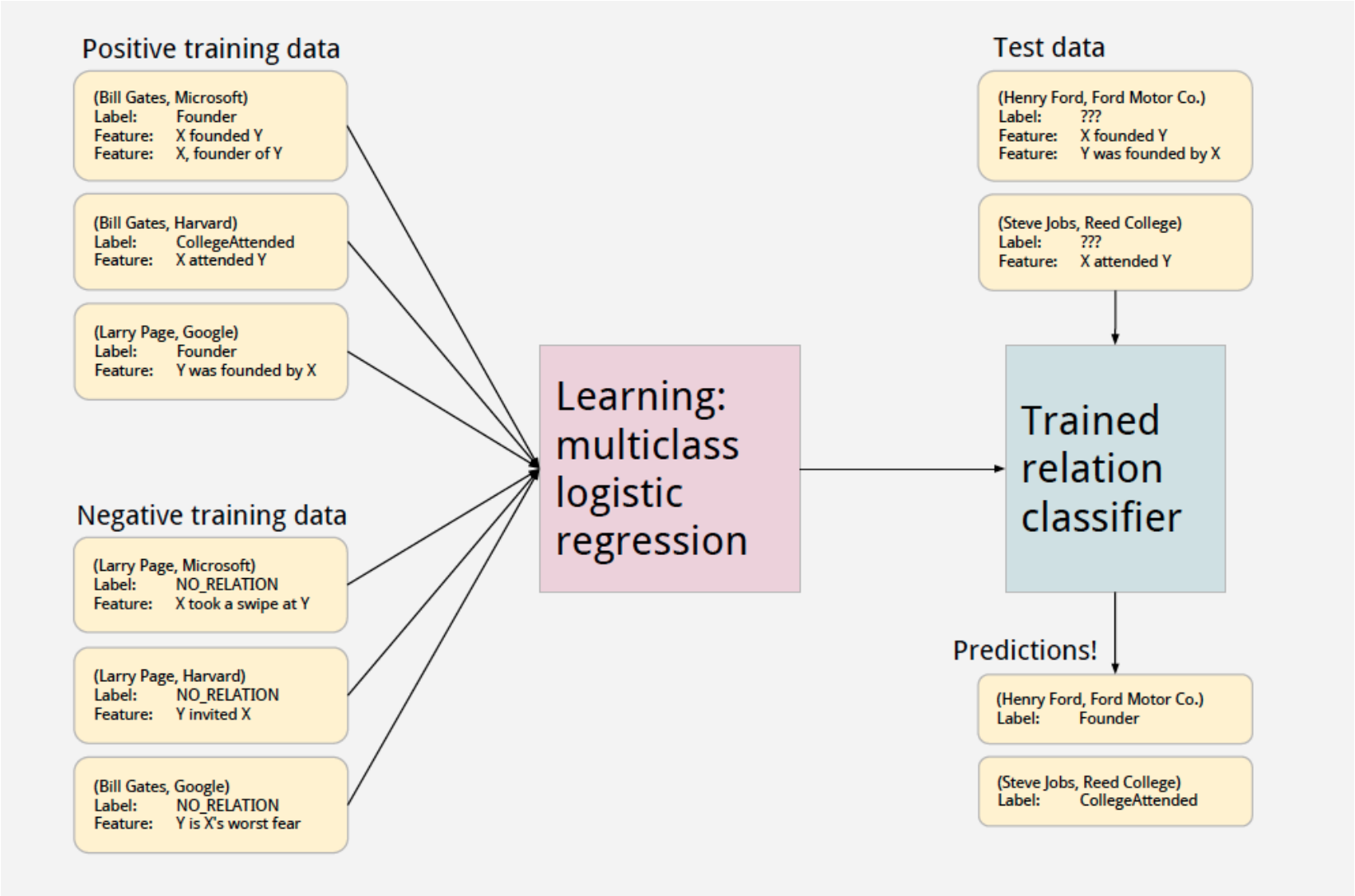
Preparing test data



Preparing test data



The experiment



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 $k \in \{0, 1, 2\}$