SCHOOL OF COMPUTING
THE UNIVERSITY OF UTAH

Klemen Simonic
klemen.simonic@gmail.com

Concept Aware Co-occurrence And Its Applications

Plan

- What is co-occurrence?
- Applications
- Limitations of co-occurrence
- What is concept co-occurrence?
- How can we obtain concept co-occurrence?
- Short text understanding
- Conclusion
- Future work

What is co-occurrence of a word?

Distribution over words that frequently occur in the same context.

apple	
orange	0.02
banana	0.015
iPhone	0.013
CEO	0.011
iPad	0.01
vitamin	0.008
…	

book	
title	0.03
author	0.025
reader	0.02
page	0.015
ink	0.009
…	

Co-occurrence = Context that we expect to see

Applications

Concept Identification / Disambiguation

"**xyz** is a big metropolitan area with many rich cultural societies ..."

What is xyz?

xyz can be a city, because cities often interact with words:

- Metropolitan area
- Society
- Culture

We can infer what is xyz, without ever seeing it before

Applications (cont'd)

Word Chunking / Segmentation

Which partition is more likely?

"jordan 10 day weather forecast"

country

weather

"jordan 10 day weather forecast"

basketball shoe

weather

Applications (cont'd)

Coreference Resolution

"The play in the city was very nice. **It** included many great actors.",

What does "It" refer to? city play

Applications (cont'd)

Short Text Understanding

Query: **"april in paris lyrics"**

What is the intent of the query?
What does the user want?

Interpretation:

april in paris [song] lyrics

Limitations of Co-occurrence

Sparseness

Obtain co-occurrence by extracting co-occurring words within a sentence (window) over a large text corpus.

For a majority of pairs the probability that they occur is small.

apple	
orange	0.02
banana	0.015
iPhone	0.013
CEO	0.011
iPad	0.01
vitamin	0.008
…	

pepino	
melon	1.0

Popular fruit => many co-occurrences
Unknown fruit => almost no co-occurrences

Limitations of Co-occurrence (cont'd)

Words are atomic units

apple	
orange	0.02
banana	0.015
iPhone	0.013
CEO	0.011
iPad	0.01
vitamin	0.008
…	

No relations to concepts

apple (fruit)	
orange	0.02
banana	0.015
vitamin	0.008

apple (company)	
iPhone	0.013
CEO	0.011
iPad	0.01

Separated by concepts

Limitations of Co-occurrence (cont'd)

No syntactic structure

Noun Phrases

apple	
iPhone	0.013
CEO	0.011
iPad	0.01
…	

Verb Phrases

apple	
make	0.02
sell	0.015
release	0.013
…	

Prepositional Phrases

apple	
with	0.04
along	0.03
Instead of	0.028
…	

Noun phrases alone are not enough.

Verb and prepositional phrases.

Limitations of Co-occurrence (cont'd)

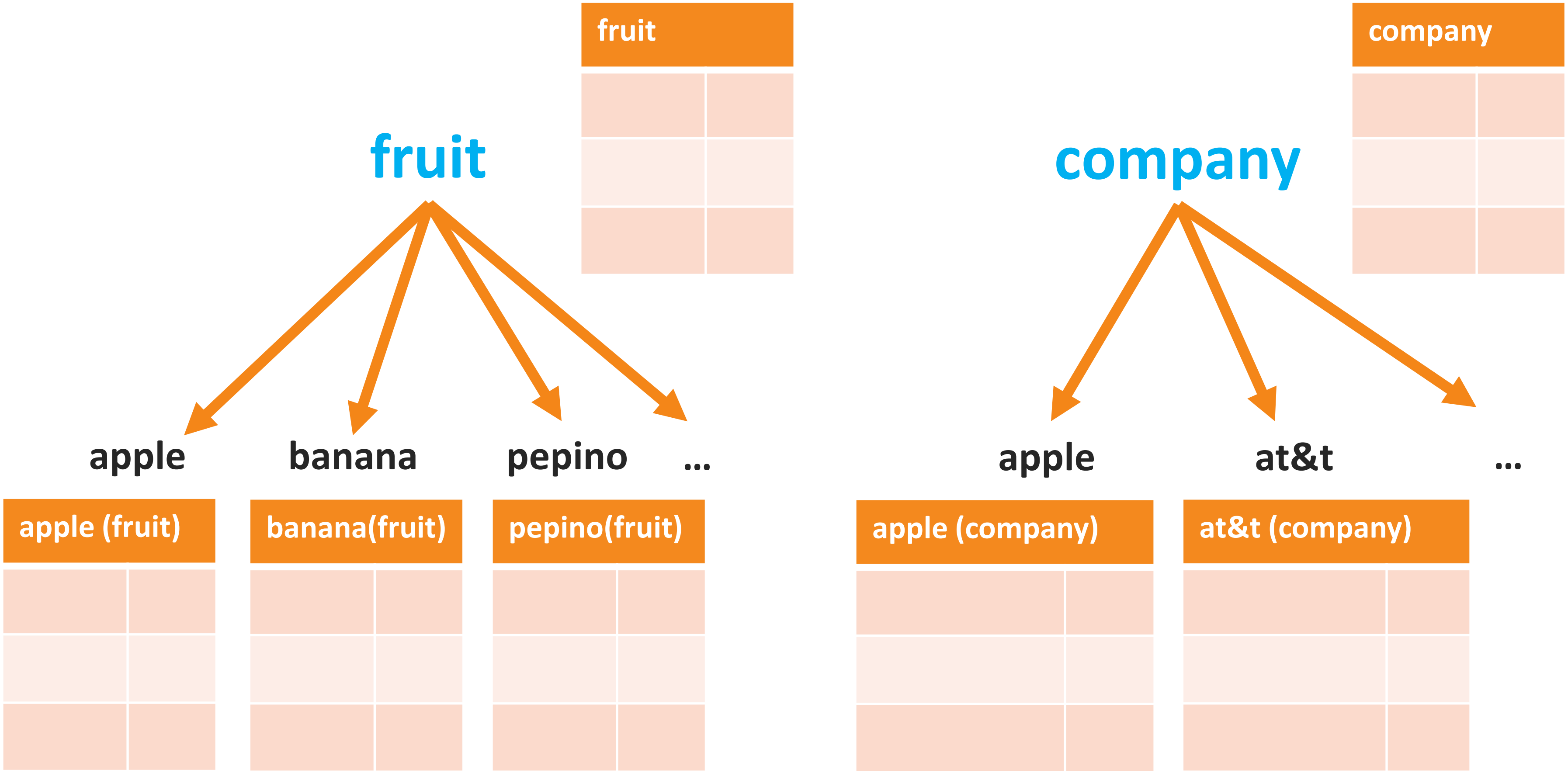
Sparseness

No relations to concepts

No syntactic structure

Concept Aware Co-occurrence

Learn concept co-occurrence from instance co-occurrence



=> Also solves the sparseness problem

How do we do this?

Problems:

- Instance concept relationships?
- Co-occurring noun, verb and prepositional phrases?
- Acquire instance co-occurrence?
- Acquire concept co-occurrence?

Instance Concept Relationships

IsA Network: a collection of isA relations

IsA relation:

(i, c)

i = instance

c = concept

(apple, company)
(apple, fruit)

Typicality scores:

p(i | c) = typicality of i as c

p(c | i) = typicality of c as i

P(apple | company)
P(company | apple)

Very important for concept learning

p(robin | bird) > p(penguin | bird)

Instance Concept Relationships (cont'd)

IsA Network

How do we obtain isA relationships?

Use sentences containing Hearst Pattern.

Example:

I have visited countries **such as** China, Japan and South Korea.

=> (China, country)

(Japan, country)

(South Korea, country)

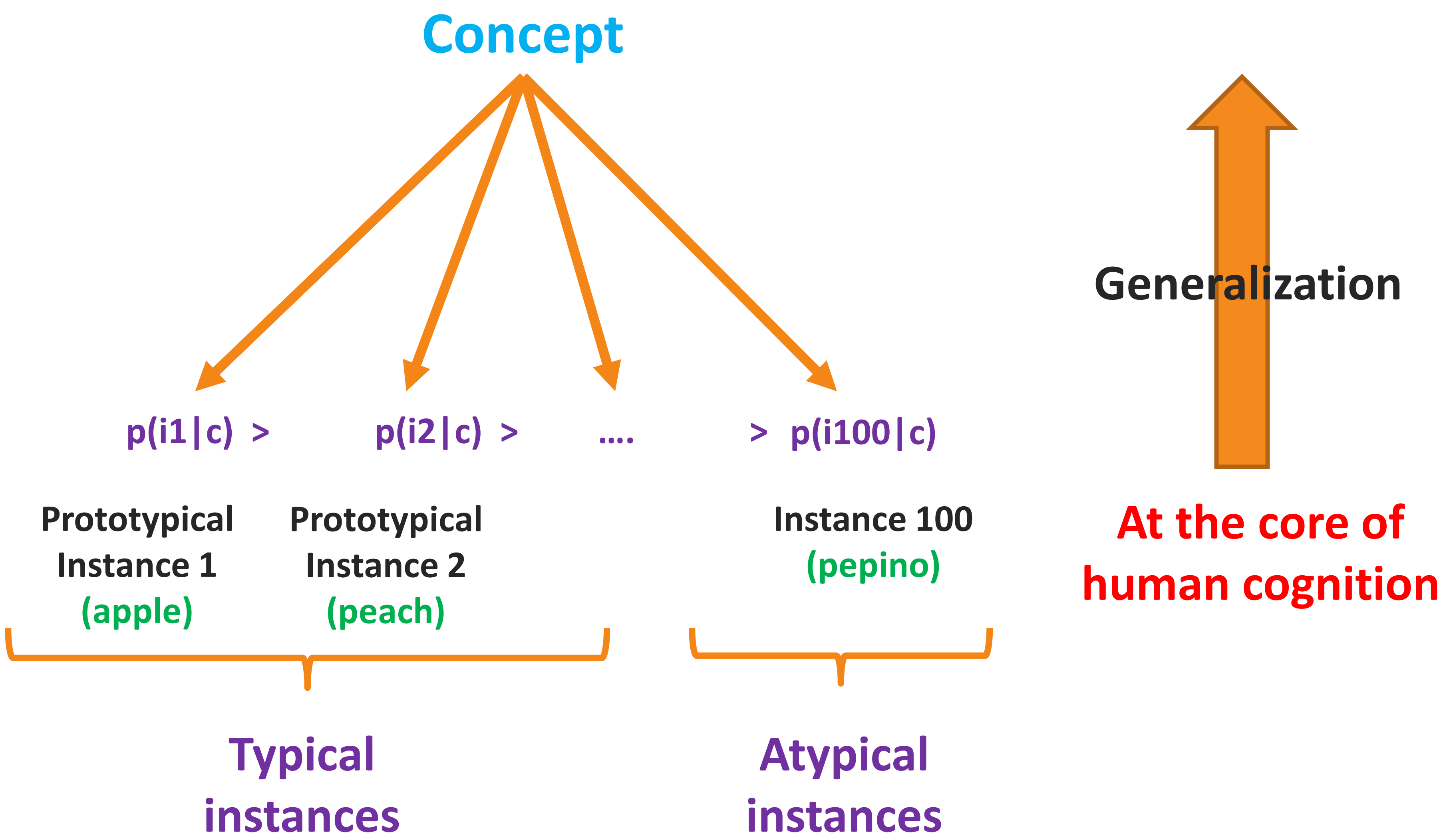
We use Probase from Microsoft Research:

#concepts = 2,653,872

#isA pairs = 20,757,545

(Extracted from: 1.6B web pages, 326M "such as" sentences)

General Concept Learning



Instance Co-occurrence

Given: **(i, c) pairs from isA network**

How do we get the co-occurrence for (apple, company)?

Noun Phrases

apple (company)	
iPhone	0.013
CEO	0.011
iPad	0.01
…	

Verb Phrases

apple (company)	
make	0.02
sell	0.015
release	0.013
…	

Prepositional Phrases

apple (company)	
with	0.04
along	0.03
Instead of	0.028
…	

Instance Co-occurrence

How can we identify apple is of concept company?

company fruit

apple

Silicon Valley company Apple produces great phones.

One concept of apple => no conflicts

Many companies are involved in fruit business of apple juice.

Two concepts of apple => conflict

=> Disregard sentences with conflicts.

Instance Co-occurrence

Silicon Valley <u>**company**</u> **Apple** produces great phones.

One concept of apple => no conflicts

1.) We declare apple is a company

2.) Extract co-occurring noun phrases
(Silicon Valley, phones)

3.) Extract co-occurring verb phrases
(produces)

4.) Extract co-occurring prepositional phrases

5.) Repeat steps 1 - 4 for every isA pair over all the sentences.

Instance Co-occurrence

Real world experiment:

1.) Rent ~ 50 machines
2.) Download web pages:
  CommonCrawl dataset
  #web pages > 2 B
  Over 200TB of data
3.) Clean HTML and extract text from every web page
4.) Annotate extracted text of every web page with:
  Tokenization
  Sentence splitting
  POS tagging
  Dependency parsing
5.) Extract instance co-occurrence

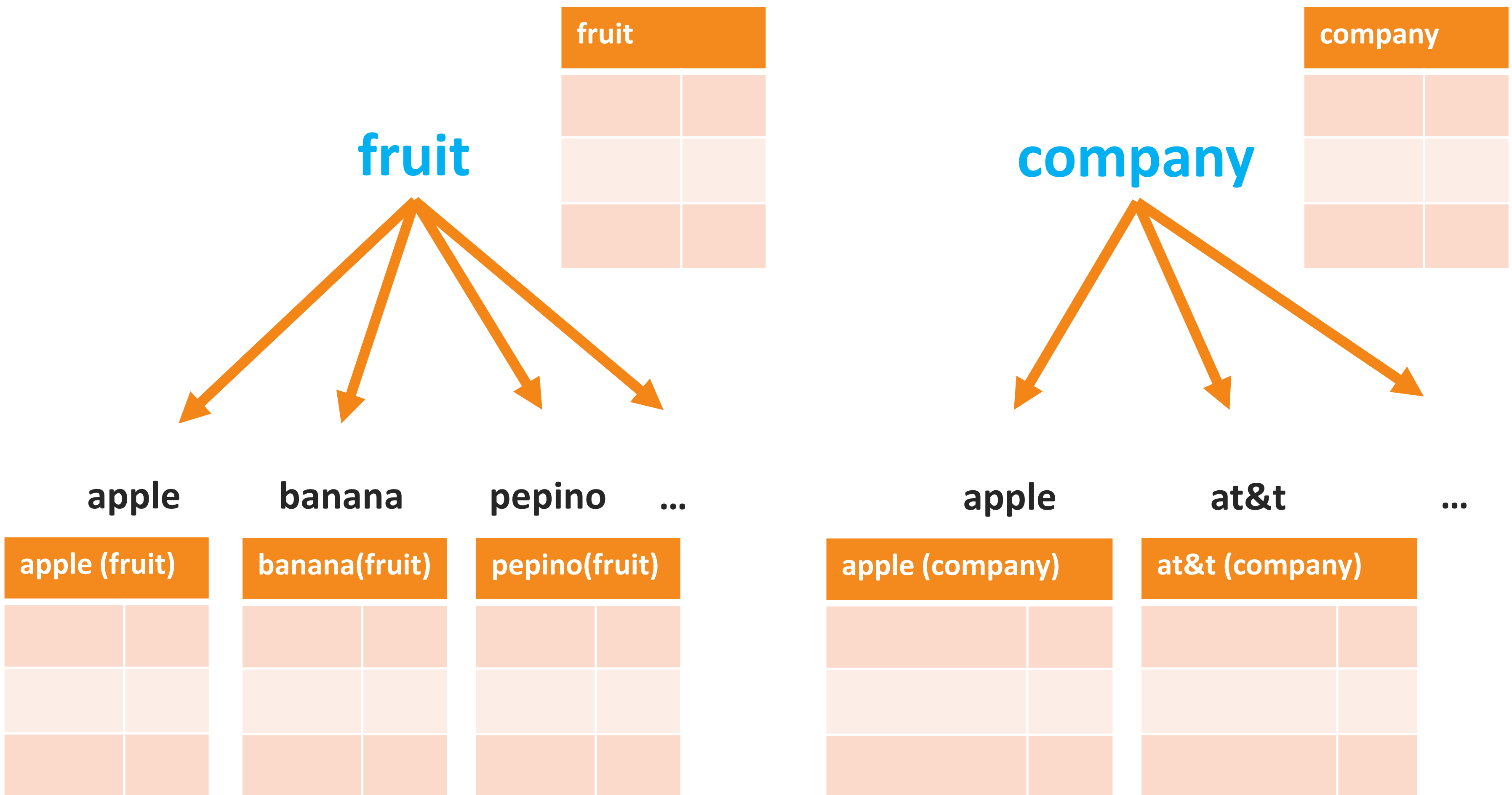
Map
Reduce

Long, complicated, computational very expensive pipeline.

Concept Co-occurrence

Concept co-occurrence from instance co-occurrence?

Generalization



Concept Co-occurrence (cont'd)

IR(i, c) = instance representation of instance i as a concept c

CR(c) = concept representation of concept c

General formula for learning concept co-occurrence:

$$CR(c) \propto \sum_i \underbrace{p(i|c)}_{\text{Typicality score}} \underbrace{IR(i,c)}_{\text{Instance representation}}$$

Example:

CR(fruit) = p(orange | fruit) IR(orange, fruit)
+ p(apple | fruit) IR(apple, fruit)
+ p(tomato | fruit) IR(tomato, fruit)
+ p(olive | fruit) IR(olive, fruit)

Verb Concept Co-occurrence

List of concepts that typically occur with the verb phrase

ride	
train	0.013
bike	0.011
horse	0.01
…	

watch	
tv show	0.013
movie	0.011
documentary	0.01
…	

$$VR_{sub}(v) \propto \sum_{c} p(c) VP_{sub}(c, v)$$

$$VR_{obj}(v) \propto \sum_{c} p(c) VP_{obj}(c, v)$$

Concept Aware Co-occurrence

Consists of :

Instance co-occurrence (instance, concept)

apple (fruit)	

Concept co-occurrence (concept)

fruit	

Verb concept co-occurrence (verb phrase)

ride	

Short Text Understanding

Examples of short text:

- Search queries (web search query, mail search query, …)
- Posts (Facebook post, Twitter posts, …)

Properties of short text:

- Very Limited Context
- Often irregular syntax

Understanding short text has huge value for:

- => Users (better user experience, …)
- => Service providers (better service, adds, …)

Need intensive knowledge processing techniques

Short Text Understanding

Focus on understanding search queries

Examples:

april in paris lyrics => april in paris [song] lyrics [music]

harry potter watch => harry potter [brand] watch [accessory]

watch harry potter => watch [verb] harry potter [movie]

read harry potter => read [verb] harry potter [book]

pink shoes => pink [color] shoes [accessory]

pink song => pink [artist] song [music]

We built a system that is very good at understanding search queries!

Short Text Understanding

Problem Definition:

Input:

Tokens: T = { t_1, … t_N }

Output (jointly):

Segmentation: S = { s_1, …, s_n }

Disambiguation: C = {c_1, …, c_n }

Output S and C have to be semantically meaningful

Example:

Input: april in paris lyrics

Output:

s_1 = april in paris, s_2 = lyrics

c_1 = song, c_2 = music

Joint Structured Prediction Model

Big Picture:

1. **Enumerate all (S, C) from T**
2. **Score every (S, C)**
3. **Choose (S, C) with the highest score**

1.) Enumerate all possible (S, C) from T:

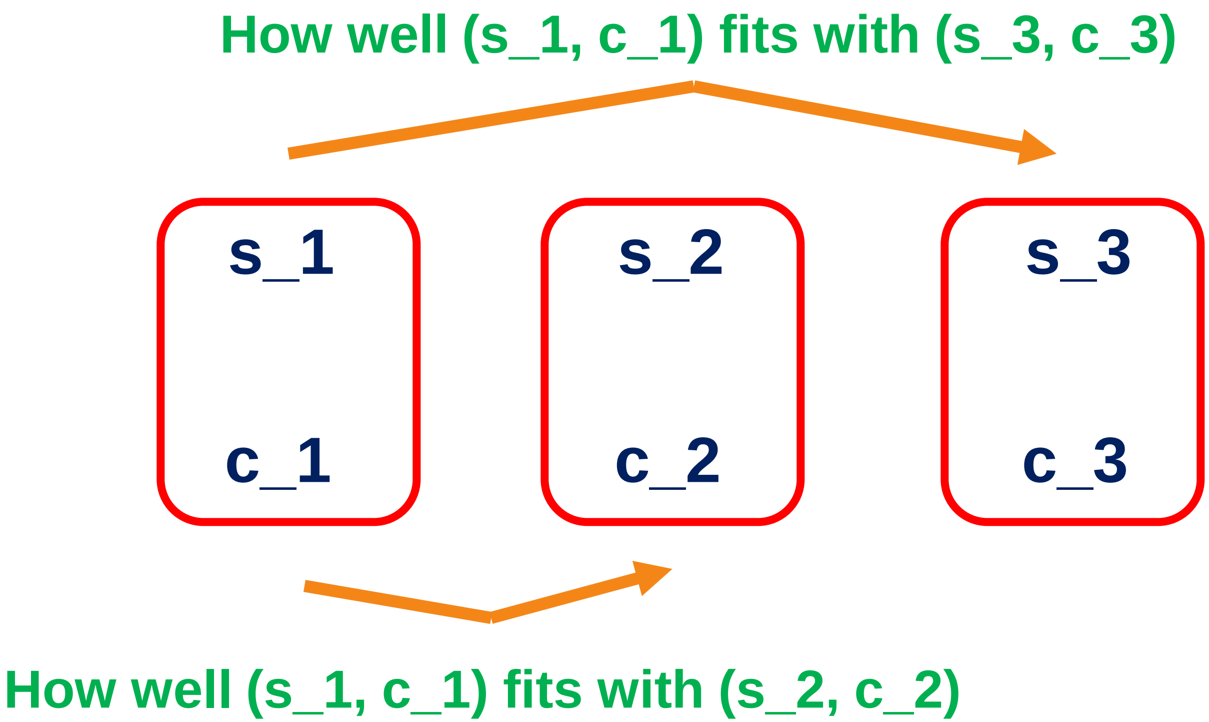
We can afford to do exhaustive enumeration because queries are very short (< than 10 tokens).

2.) Score (S, C) with a model

3.) Choose (S, C) with the highest score (simple)

Joint Structured Prediction Model (cont’d)

Score (S, C) with a model:

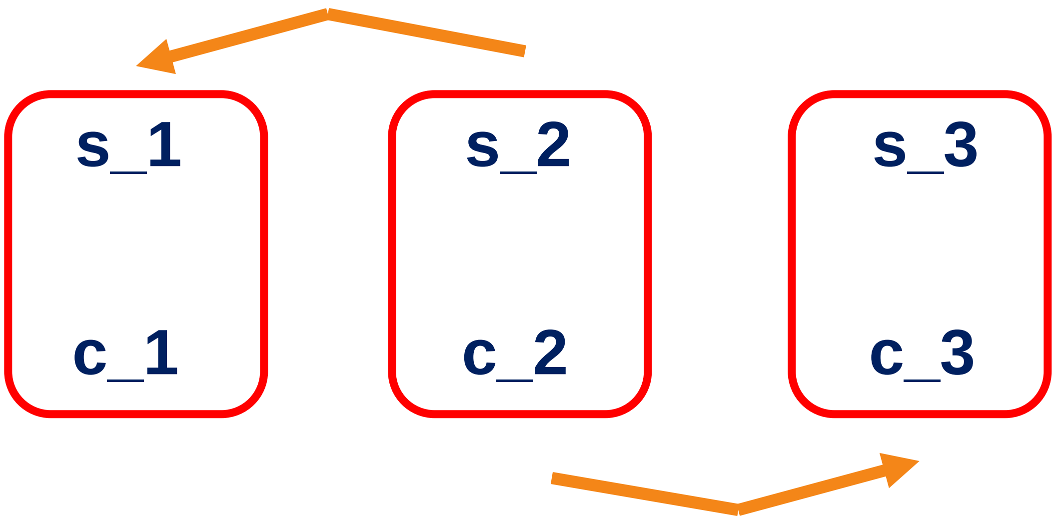


=> Score H(s_1, c_1)

Joint Structured Prediction Model (cont'd)

Score (S, C) with a model:

How well (s_2, c_2) fits with (s_1, c_1)

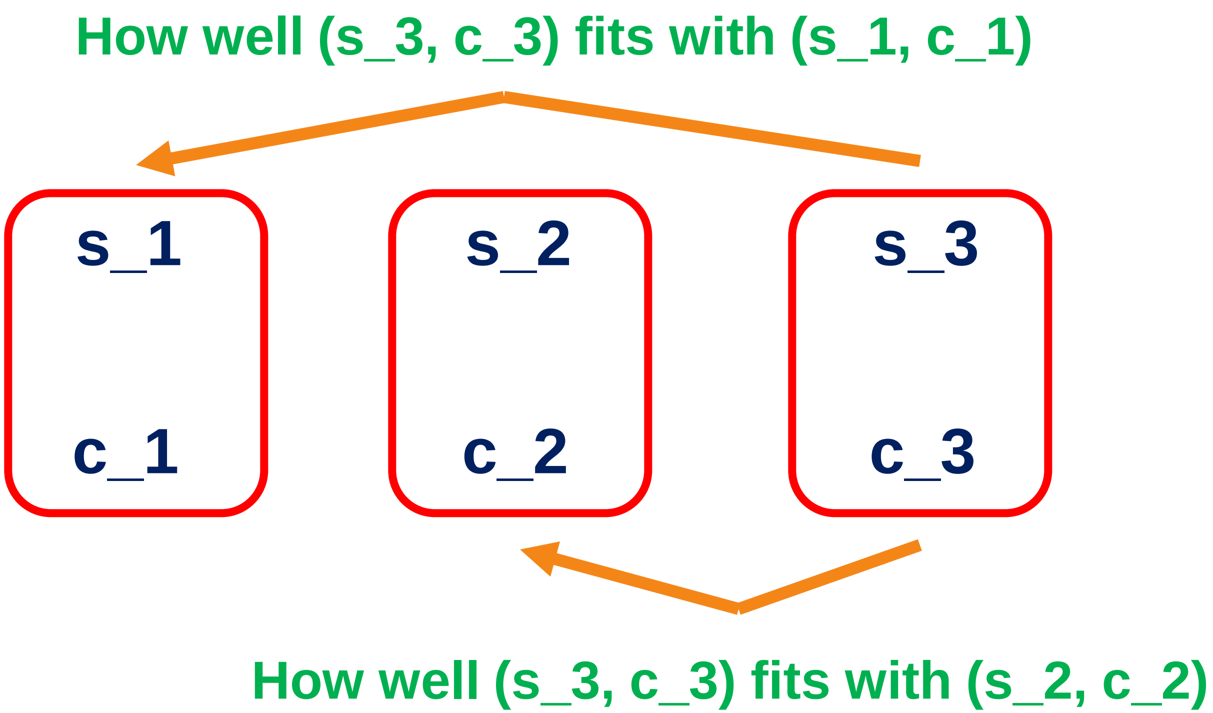


How well (s_2, c_2) fits with (s_3, c_3)

=> Score H(s_2, c_2)

Joint Structured Prediction Model (cont'd)

Score (S, C) with a model:



=> Score H(s_3, c_3)

Joint Structured Prediction Model (cont'd)

Score (S, C) with a model:

Final Score $H(S, C) = H(s_1, c_1) + H(s_2, c_2) + H(s_3, c_3)$

Combine the scores

Recap:

1. We score each (s, c) pair individually based on the other segments and concepts

2. Combine all the individual score into one score

Joint Structured Prediction Model (cont'd)

Score (S, C) with a model:

Exponential model

Scoring function of the input and output

Probability of the output given the input

$$P(Y|T) \propto \exp\left[\Phi(Y, T)\right]$$

Segment concept pairs { (s_i, c_i) }

Input tokens

Other segments and concepts

$$\Phi(Y, T) = \sum_{i=1}^{n} \hat{\Phi}_G(Y_{-i}, s_i, c_i, T)$$

Decomposition

$$\hat{\Phi}_G(Y_{-i}, s_i, c_i, T) = \sum_{j=1 \quad i \neq j}^{n} \Phi_G(s_j, c_j, s_i, c_i, T)$$

Joint Structured Prediction Model (cont'd)

Score (S, C) with a model:

standard term-term co-occurrence

Co-occurrence model:

$$\Phi_G(s_j, c_j, s_i, c_i, T) = \Phi_{G_{1ss}}(s_j, s_i) + \Phi_{G_{1cs}}(c_j, s_i)$$

instance co-occurrence (s_i, c_i)

Concept co-occurrence model:

$$\Phi_G(s_j, c_j, s_i, c_i, T) = \Phi_{G_{2ssc}}(s_j, s_i, c_i) + \Phi_{G_{2sc}}(s_j, c_i) + \Phi_{G_{2cc}}(c_j, c_i)$$

concept co-occurrence c_i

Experiments

We want to compare the models:

- **Co-occurrence model**
- **Concept co-occurrence model**

We need:

- **Dataset of labeled queries**
- **Evaluation metric**

Experiments: Dataset

Labeled query needs to have:

- **Segments {s_1, …, s_n}**
- **Concepts {c_1, …, c_n}**

!!! No publicly available dataset

We construct the dataset our selves.

Query Pattern:

Sequence of terms and concepts

Examples:

[song] lyrics
[song] sheet
[movie] premier
[food] recipe

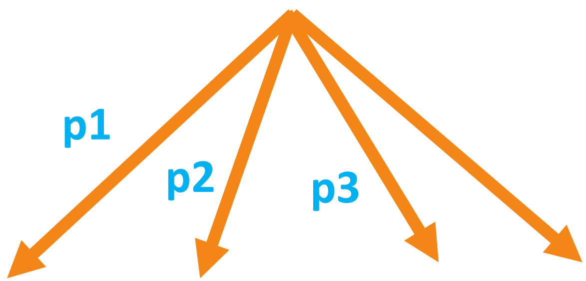
Experiments: Dataset (cont'd)

From query pattern to query:

- **Sample instances using isA network**
- **Substitute concepts with instances**

Query pattern: [song] lyrics

IsA network: song



Yesterday April in Pairs Grenade ...

Yesterday lyrics
Grenade lyrics
...

Experiments: Evaluation

We need a metric to measure the similarity of two queries

Exact metric:

$$em(q_1, q_2) = \begin{cases} 1 & q_1, q_2 \text{ same segments and concepts} \\ 0 & \text{otherwise} \end{cases}$$

Rank metric:

- More forgiving than exact metric
- Requires the same segments; otherwise it is 0
- Requires a ranking over the concepts
- The higher the concept that we expect, the better the result
- Take inverse of the rank of the concept

Example: april in paris **[song]** lyrics

Ranking: **(1) hit**
(2) song → Inverse rank: **1 / 2**
(3) book

Experiments: Overall Results

Model	Exact Metric	Rank Metric
Co-occurrence	0.479	0.597
Concept co-occurrence	0.633	0.718

Concept co-occurrence model significantly outperforms co-occurrence model

Experiments: Specific Query Patterns

Unambiguous context is not that hard

Query Pattern	Co-occ.	Concept co-occ.	
[song] lyrics	0.801	0.895	easy
[song] sheet	0.305	0.681	hard
[film] premiere	0.776	0.873	easy
[watch] watch	0.173	0.59	
[food] recipe	0.473	0.592	
[furniture] design	0.557	0.651	

sheet **is ambiguous word (e.g. sheet music, bed sheets)**

=> Concept representations help a lot!

Conclusion

- We have seen co-occurrence and its applications
- Limitations of co-occurrence
  (sparseness, no concepts, no syntactic structure)
- IsA network
- Generalization process
- Concept co-occurrence:
  - Instance co-occurrence
  - Concept co-occurrence
  - Verb phrase co-occurrence
- Short text understanding:
  - Understating the intent of the queries
  - Model definition and prediction
  - Concept co-occurrence helps a lot

Future Work

To do an intelligent prediction (AI):

- **The prediction model is important, but not that important**
- **It is mostly about the knowledge**

Knowledge:

- **Standard knowledge bases (e.g. Freebase) are not enough:**
  - **Knowledge represented in tables (e.g. triples)**
  - **High precision, no probabilistic interpretation**
  - **No relation between knowledge, language, concepts**
- **Knowledge bases have to be about how humans use knowledge**
- **Probabilistic interpretations are very important**

Two big problems:

1. **How to acquire such knowledge?**
2. **How to represent such knowledge?**

=> Unsupervised learning will play a vital role

(word, concept embeddings, new learning principles)

Questions?