Lexical Semantics

Lizhen Qu

Semantic Analysis

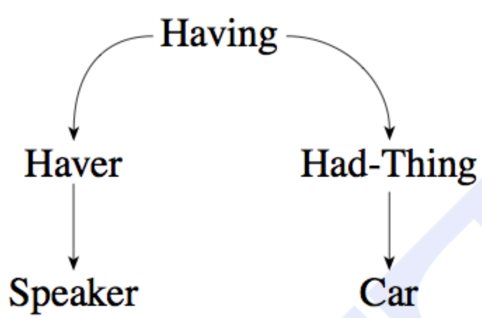
- Meaning representation: formal structures to represent the meanings of linguistic utterances.
- Semantic analysis: the process of constructing and assigning meaning representations to linguistic inputs.

I have a car.

semantic analysis

Meaning Representations

$$\exists x,y\ Having(x) \wedge Haver(Speaker,x) \wedge HadThing(y,x) \wedge Car(y)$$



Car
$\Uparrow$ POSS-BY
Speaker

```
Having
 Haver:    Speaker
 HadThing:   Car
```

What are Good Meaning Representations?

- Verifiability: system's ability to compare, or match, the representations of the meaning of linguistic inputs against the representations in its knowledge base.

Does John have a car? →

Knowledge Base
have(John_SMITH, Audi Q3)
have(Mary_LEE, NISSAN X-Trail)

- Unambiguous representations.

I wanna eat someplace near ANU.

- Doctrine of canonical form: linguistic utterances that mean the same thing should have the same meaning representation.

Does Akiba have sea food?

Are sea food served at Akiba?

- Perform inference based on meaning representations.
- Expressiveness.

Lexical Semantics

- Lexical semantics: linguistic study of word meaning.

- Key questions:

  – What is the meaning of words?
    - Most words have more than one senses.

  – How are the meanings of different words related?
    - Specific relations between senses.
      – *Vehicle* is more general than *car*.
    - Semantic fields.
      – *travel* is related to *flight*.

Terminology

- Wordforms: any form of a word.
  - good, better, best, decide, decided.
- Lexeme: an abstract representation of a word and corresponds to a set of all its forms.
  - RUN = {run, runs, ran and running}.
- Lemma: a particular grammatical form that is used to represent a lexeme. This is often the base form.
  - book, car, fly
- Word sense: a discrete representation of one aspect of the meaning of a word.

Relations between Senses

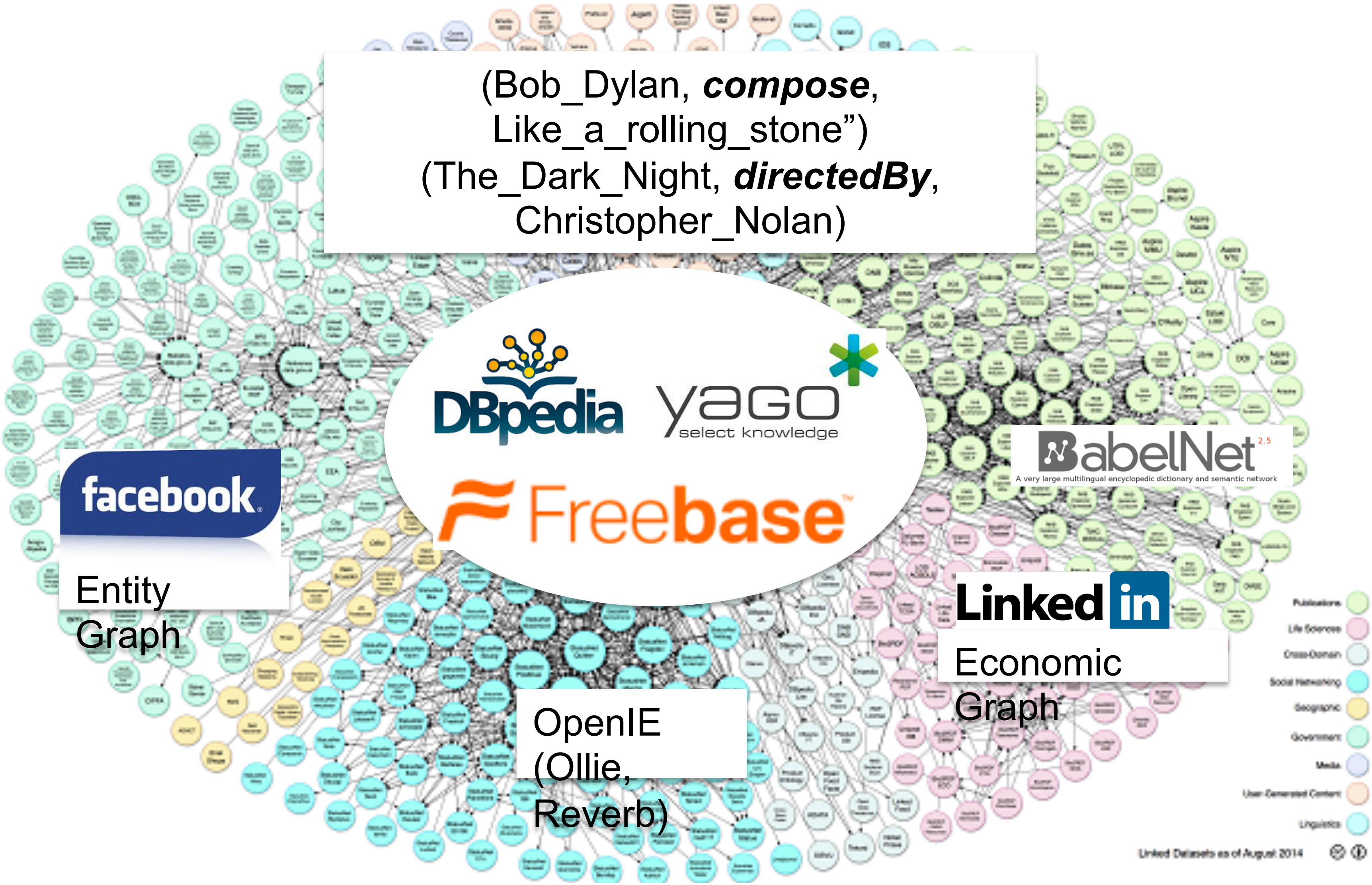
- Symmetric relations:
  - Synonyms: meaning of two senses of two different words are identical or nearly identical.
    - car/automobile, couch/sofa
  - Antonyms: two senses have opposite meaning.
    - long/short, rise/fall
- Hierarchical relations:
  - Hypernyms: sense A is a hypernym of sense B if A is more general. E.g. pet/dog.
  - Hyponyms: sense A is a hyponym of sense B if A is more specific. E.g. mango/fruit.
  - Meronyms: sense A is a meronym sense B if A is part of B.
    - wheel/car
  - Holoynms: sense A is a holoynm sense B if B is part of A.

WordNet

- Very large lexical database of English.
  - 117k nouns, 115k verbs, 21k adjectives, and 4k adverbs.
- Synset: the set of near-synonyms for a sense.
  - 82k noun synsets, 13k verb synsets, 18k adj. synsets, 3k adv. synsets.
  - Avg. # synsets : noun 1.24, verb 2.17, adj. 1.4, adv. 1.25
- Website : http://wordnet.princeton.edu/
- WordNet in other languages: German, Italian …

- Lemmatization: map a wordform to a lemma.
  - http://textanalysisonline.com/nltk-wordnet-lemmatizer
  - http://nlp.stanford.edu/software/corenlp.shtml

Knowledge Bases (Open Linked Data)



(Bob_Dylan, ***compose***, Like_a_rolling_stone”)
(The_Dark_Night, ***directedBy***, Christopher_Nolan)

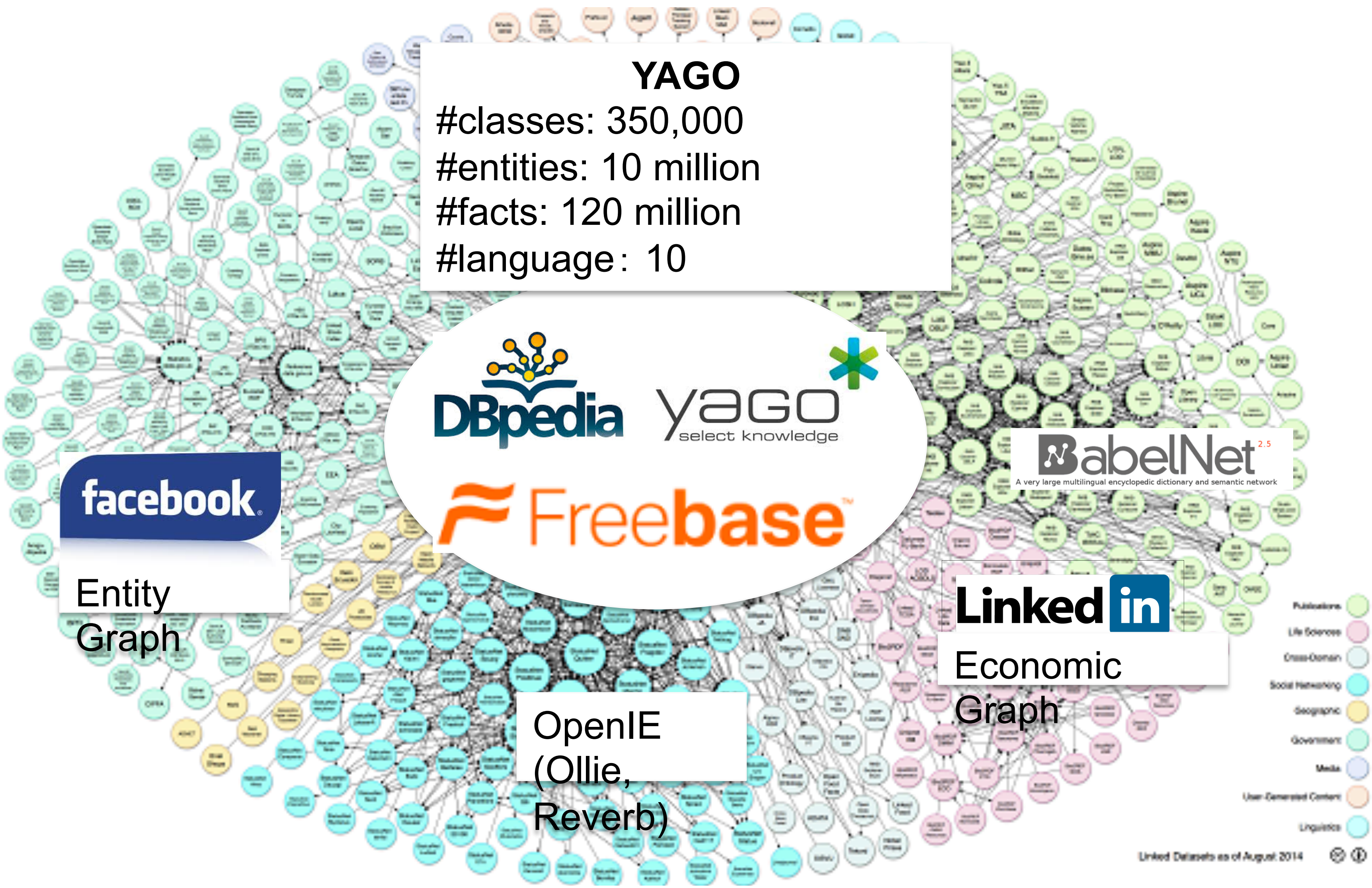
Entity Graph

Economic Graph

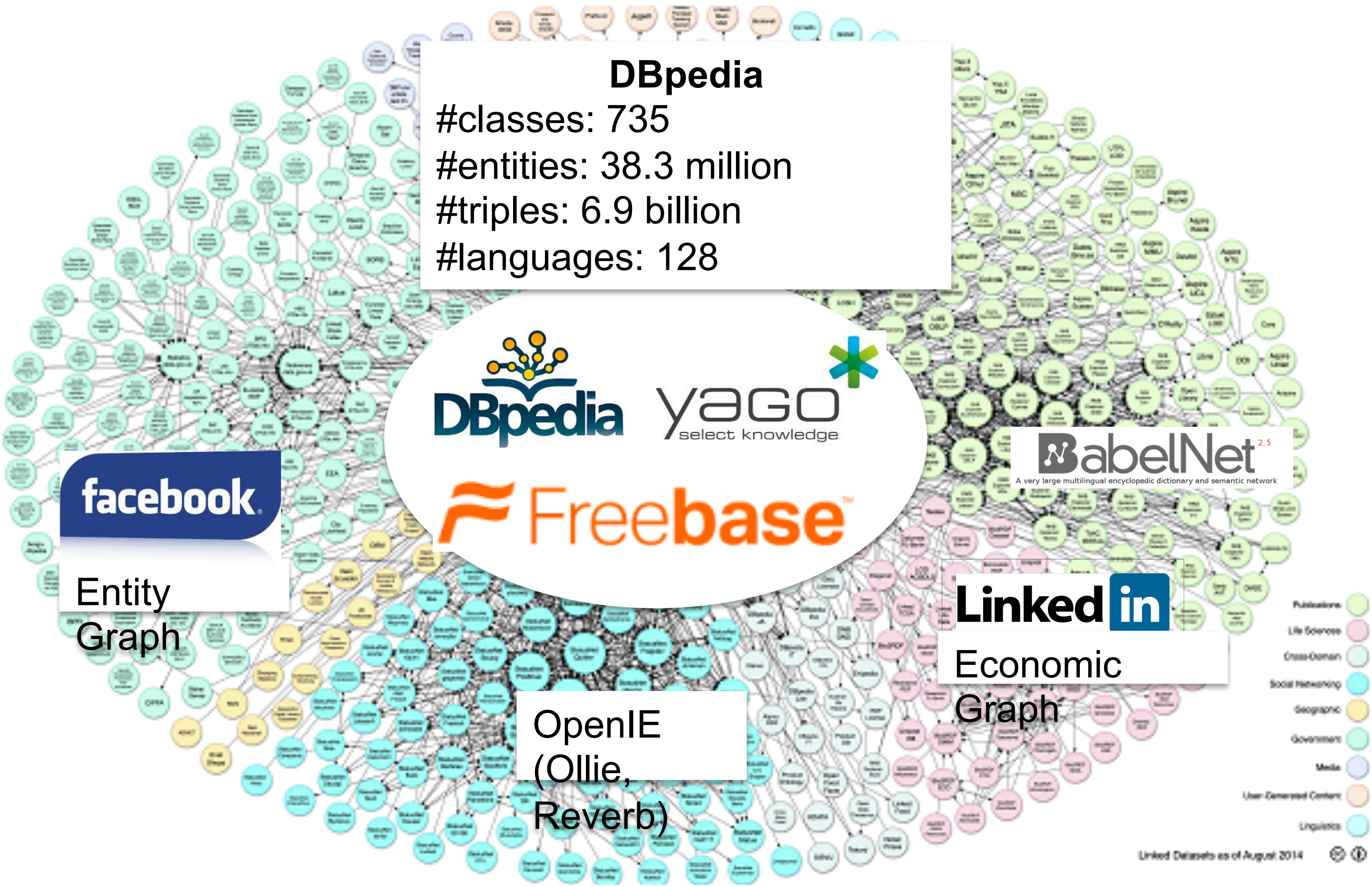
OpenIE (Ollie, Reverb)

Knowledge Bases (Open Linked Data)

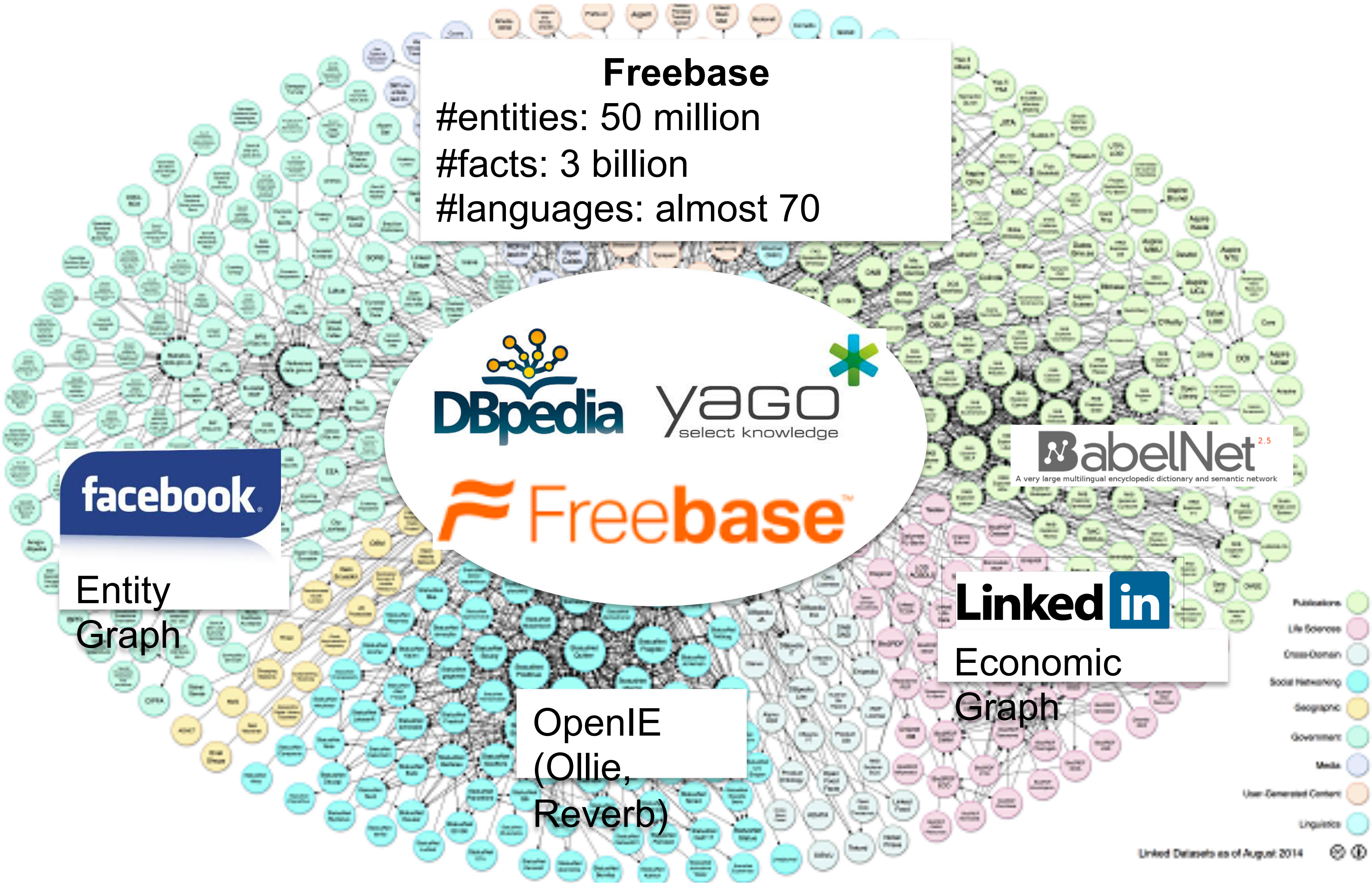
YAGO
#classes: 350,000
#entities: 10 million
#facts: 120 million
#language：10



Knowledge Bases (Open Linked Data)



Knowledge Bases (Open Linked Data)



Weaknesses of Discrete Representations

- Missing new words.
- Hard to compute word similarity.
- Subjective
- Require human labor to create and maintain.

Distributional Similarity

“You shall know a word by the company it keeps”

(J. R. Firth 1957: 11)

'study in the united states',
'live in New Zealand',
'study in the january 10',
'stay in the New England’,
'live in the US',
'study in the United Kingdom’

'work in the United States’,
'work in Australia.',
'its meeting on 05 January',
'annual meeting on 01 December',
'a meeting on January NUM',
'ordinary meeting on 9 December',
'regular meeting of February 10'

Learning Word Embeddings

- Tri-gram model.

$$P(x_1, x_2, ..., x_l) = P(x_1)P(x_2|x_1)\prod_{i=3}^{l} P(x_i|x_{i-1}, x_{i-2})$$

- Predict the word in the middle given context words.

$$\prod_{i=1}^{N} P(x_i|x_{i-k}, ..., x_{i-1}, x_{i+1}, ..., x_{i+k})$$

k is the size of the context window.

Continuous Bag-of-Words Model (CBOW) [1,2]

- Basic form: $P(x_i|x_{i-2}, x_{i-1}, x_{i+1}, x_{i+2}) = \dfrac{\exp(\mathbf{e}_i^{\mathrm{T}}\mathbf{e}_{\mathrm{sum}})}{\sum_{j \in V} \exp(\mathbf{e}_j^{\mathrm{T}}\mathbf{e}_{\mathrm{sum}})}$

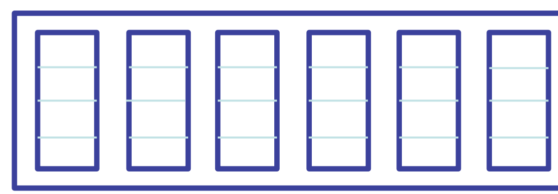
  where $k = 2$ and V is the vocabulary.

softmax layer : $\dfrac{\exp(\mathbf{e}_i^{\mathrm{T}}\mathbf{e}_{\mathrm{sum}})}{\sum_{j \in V} \exp(\mathbf{e}_j^{\mathrm{T}}\mathbf{e}_{\mathrm{sum}})}$

sum layer : $\mathbf{e}_{\mathrm{sum}} = \mathbf{e}_{i-2} + \mathbf{e}_{i-1} + \mathbf{e}_{i+1} + \mathbf{e}_{i+2}$

$(\mathbf{e}_{i-2}, \mathbf{e}_{i-1}, \mathbf{e}_i, \mathbf{e}_{i+1}, \mathbf{e}_{i+2})$

Look-up table:



$(x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2})$

High Scalability with Negative Sampling

- Approximate the softmax loss with a set of binary classification losses.
  - Positive example: the loss of seeing word i given context words.
  - Negative examples : the loss of not observing $|M|$ word j given context words, where M is a set of randomly picked words from vocabulary.

$$L_{\text{neg\_sam}}(x_i) = -\log \sigma(\mathbf{e}_i^{\mathrm{T}} \mathbf{e}_{\text{sum}}) - \sum_{u \in M} \log(1 - \sigma(\mathbf{e}_u^{\mathrm{T}} \mathbf{e}_{\text{sum}}))$$

Positive example

negative examples

where $\sigma(z) = \frac{1}{1+\exp(-z)}$

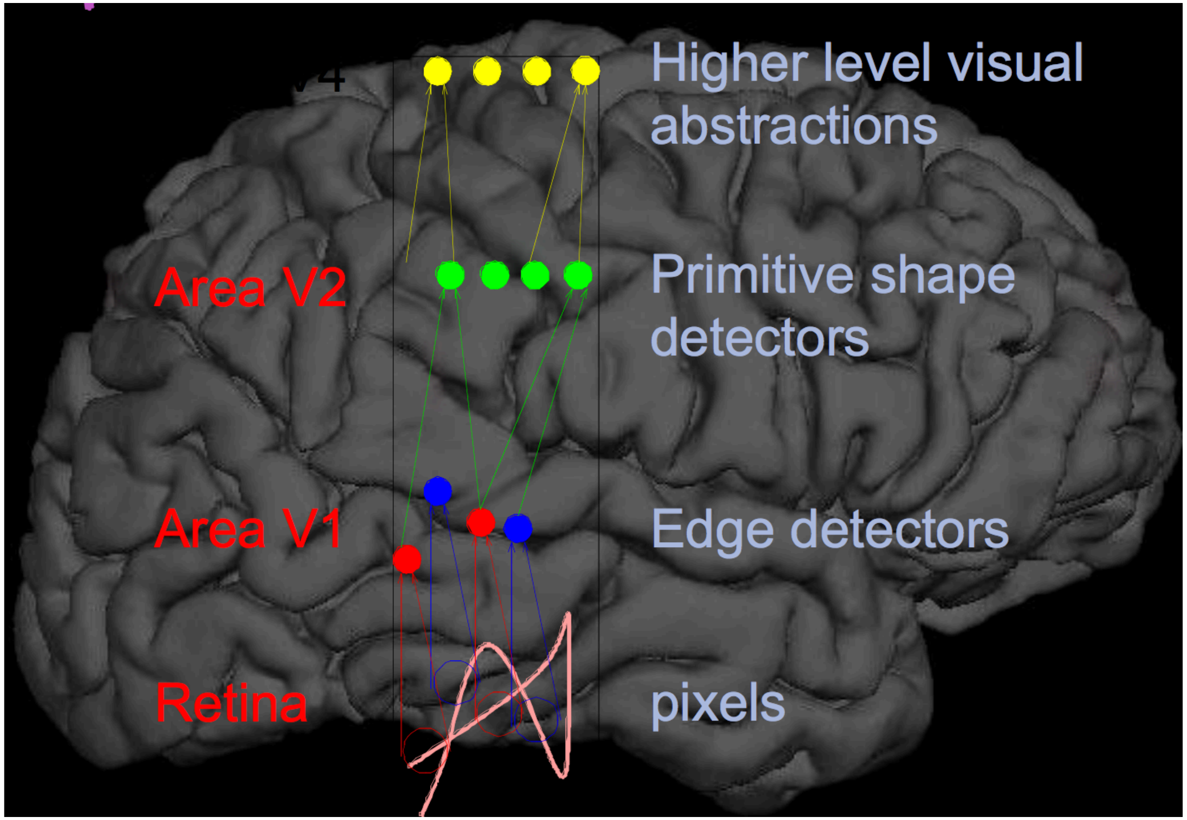
Evaluation with Word Analogy [1]

$$\mathbf{e}_{?} = \mathbf{e}_{\text{Athens}} - \mathbf{e}_{\text{Greece}} + \mathbf{e}_{\text{Norway}}$$

Type of relationship	Word Pair 1		Word Pair 2	
Common capital city	Athens	Greece	Oslo	Norway
All capital cities	Astana	Kazakhstan	Harare	Zimbabwe
Currency	Angola	kwanza	Iran	rial
City-in-state	Chicago	Illinois	Stockton	California
Man-Woman	brother	sister	grandson	granddaughter
Adjective to adverb	apparent	apparently	rapid	rapidly
Opposite	possibly	impossibly	ethical	unethical
Comparative	great	greater	tough	tougher
Superlative	easy	easiest	lucky	luckiest
Present Participle	think	thinking	read	reading
Nationality adjective	Switzerland	Swiss	Cambodia	Cambodian
Past tense	walking	walked	swimming	swam
Plural nouns	mouse	mice	dollar	dollars
Plural verbs	work	works	speak	speaks

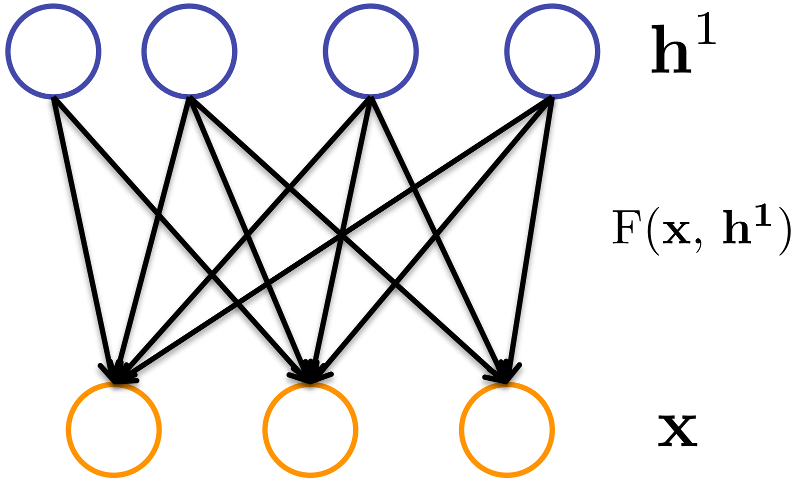
Deep Architecture

- Compositionality of knowledge.
  - Learn simpler concepts first.
  - Compose them into higher-level concepts.



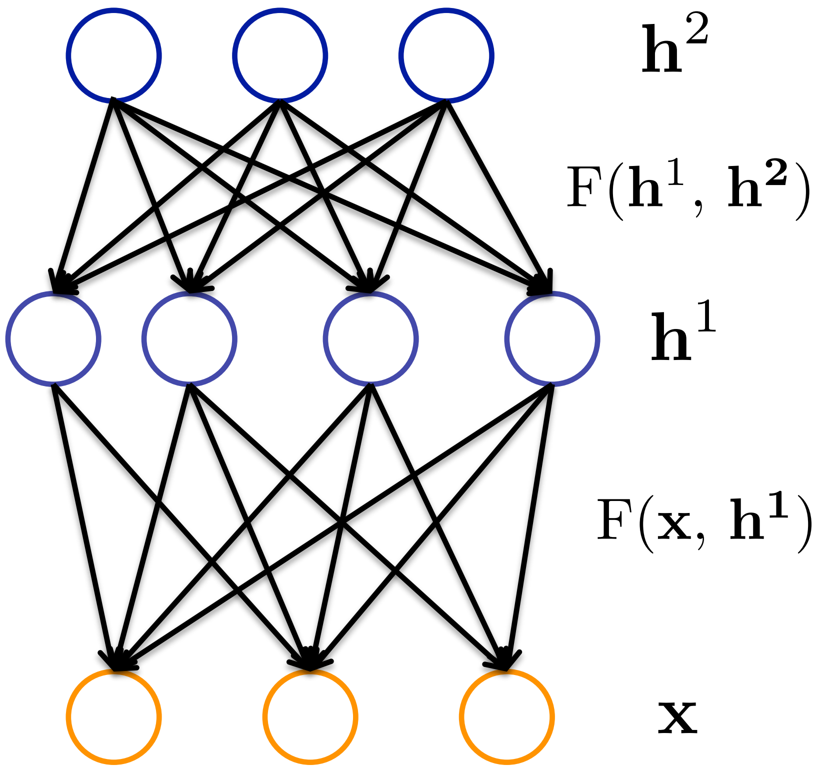
Greedy Layer-wise Pre-training [4]

- Step one:
  - Construct $\mathrm{F}(\mathbf{x}, \mathbf{h^1})$ between the visible layer $\mathbf{x}$ and the hidden layer $\mathbf{h}^1$.
  - Train $\mathrm{F}(\mathbf{x}, \mathbf{h^1})$ until it converges.



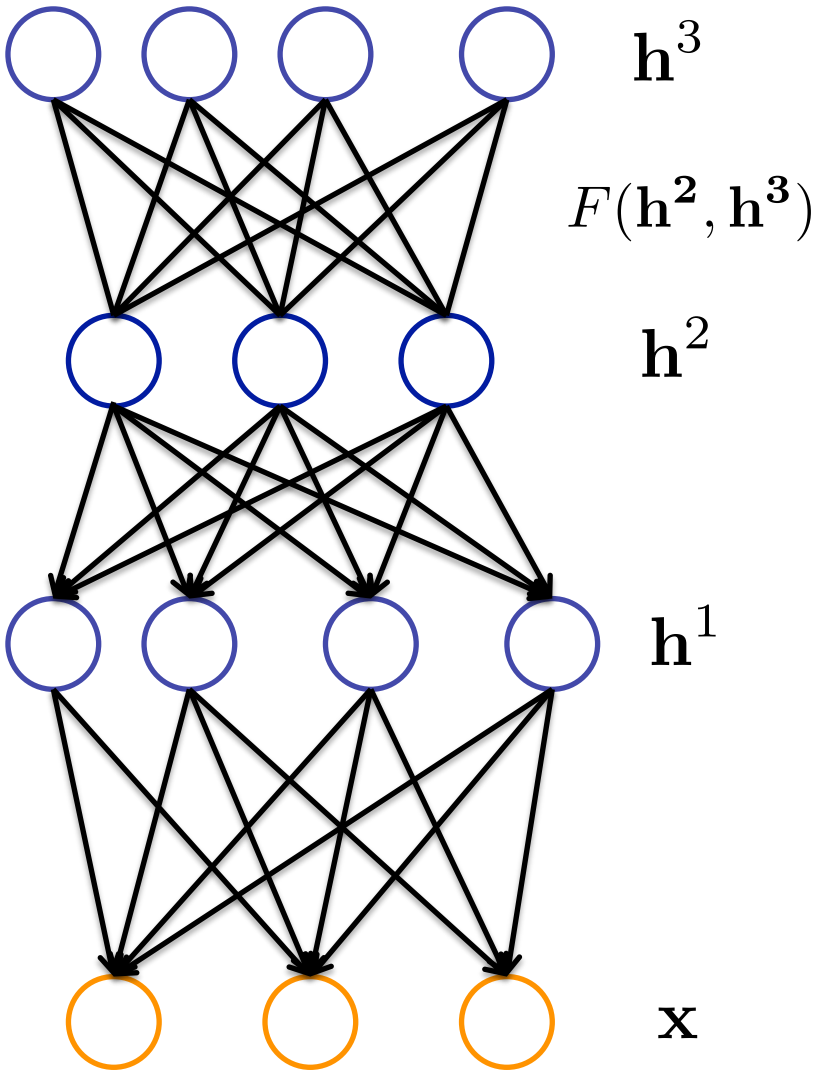
Greedy Layer-wise Pre-training

- Step two:
  - Stack $\mathrm{F}(\mathbf{h}^1, \mathbf{h^2})$ on top of $\mathrm{F}(\mathbf{x}, \mathbf{h^1})$.
  - Fix parameters of the previous layer, sample $\mathbf{h}^1$ as input , train $\mathrm{F}(\mathbf{h}^1, \mathbf{h^2})$



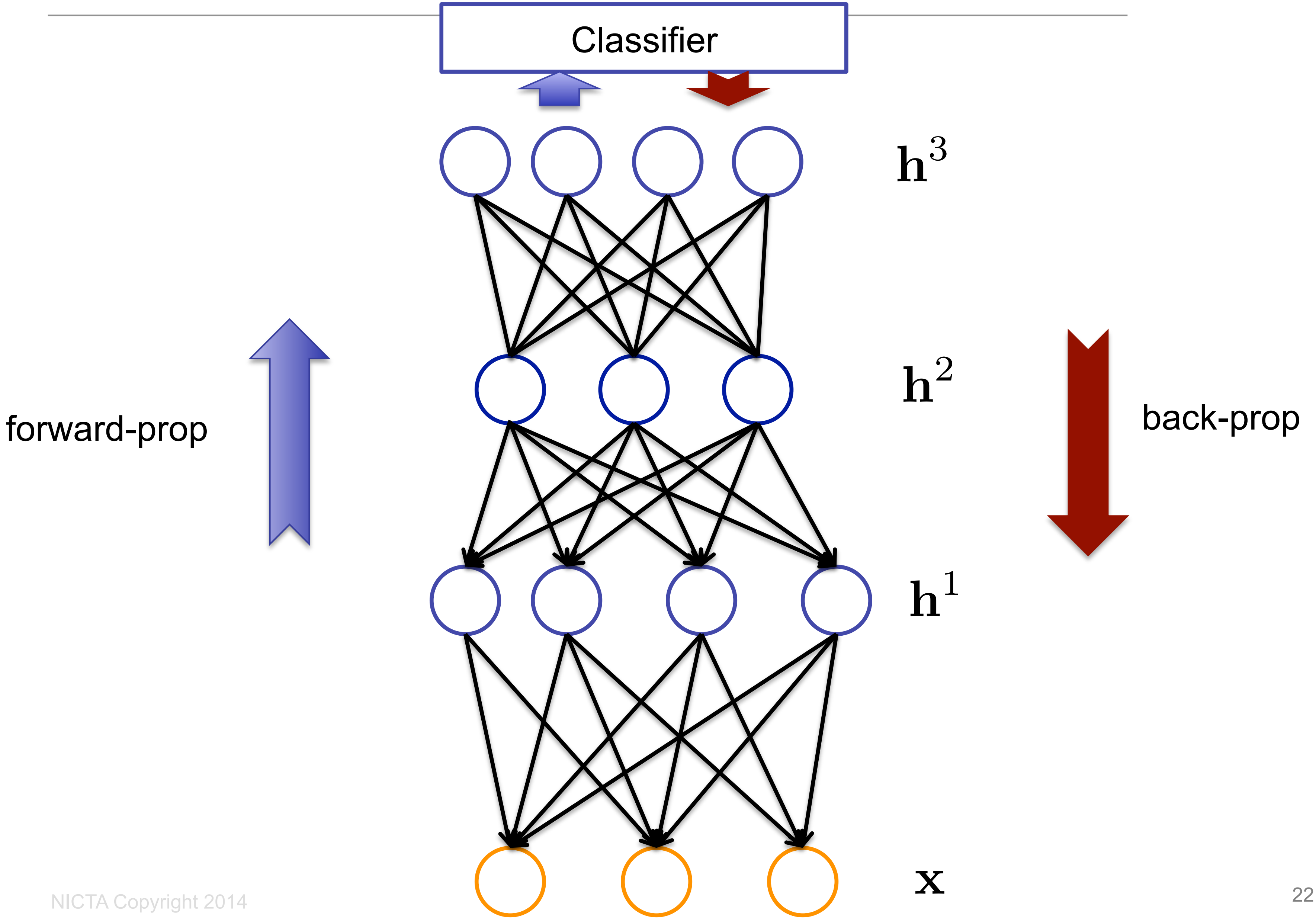
Greedy Layer-wise Pre-training

- Step three:
  - Stack $F(\mathbf{h^2}, \mathbf{h^3})$ on top of $\mathrm{F}(\mathbf{h}^1, \mathbf{h^2})$.
  - Fix parameters of the previous layer, sample $\mathbf{h}^2$ as input, train $F(\mathbf{h^2}, \mathbf{h^3})$.



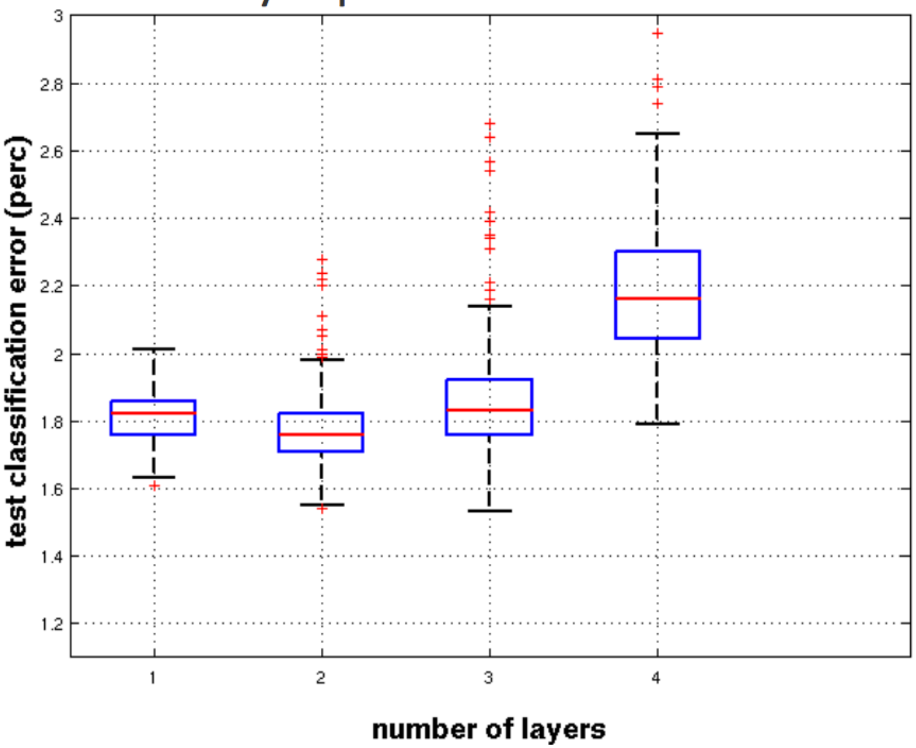
- Repeat the steps until all layers are trained.

Supervised Fine-Tuning

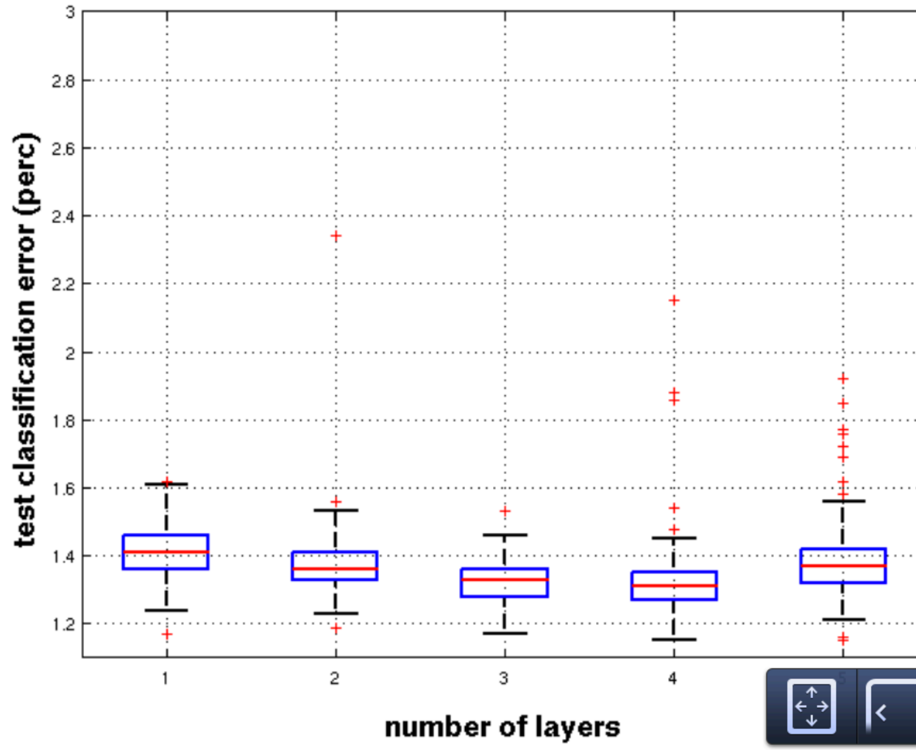


Why Greedy Layer-Wise Pre-training Works?

Purely supervised neural net



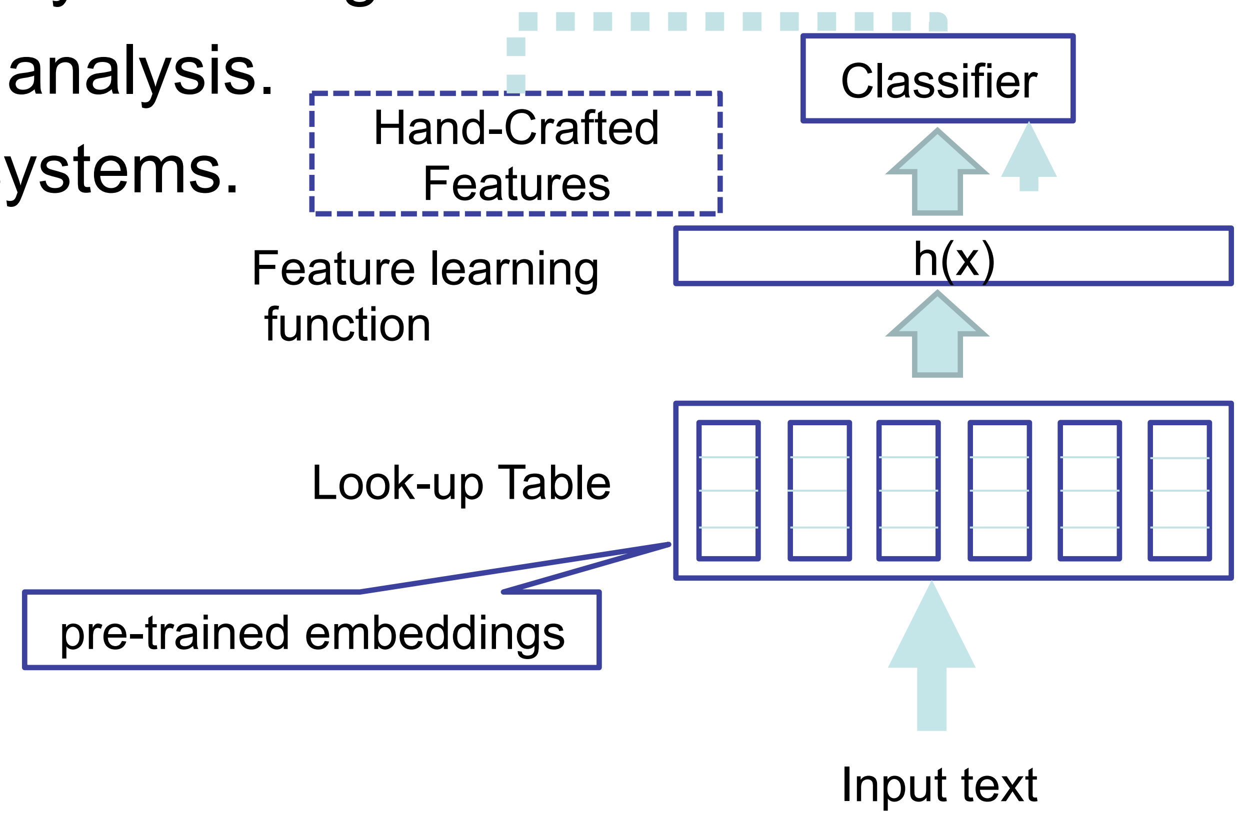
With unsupervised pre-training



(Erhan et al. JMLR 2010)

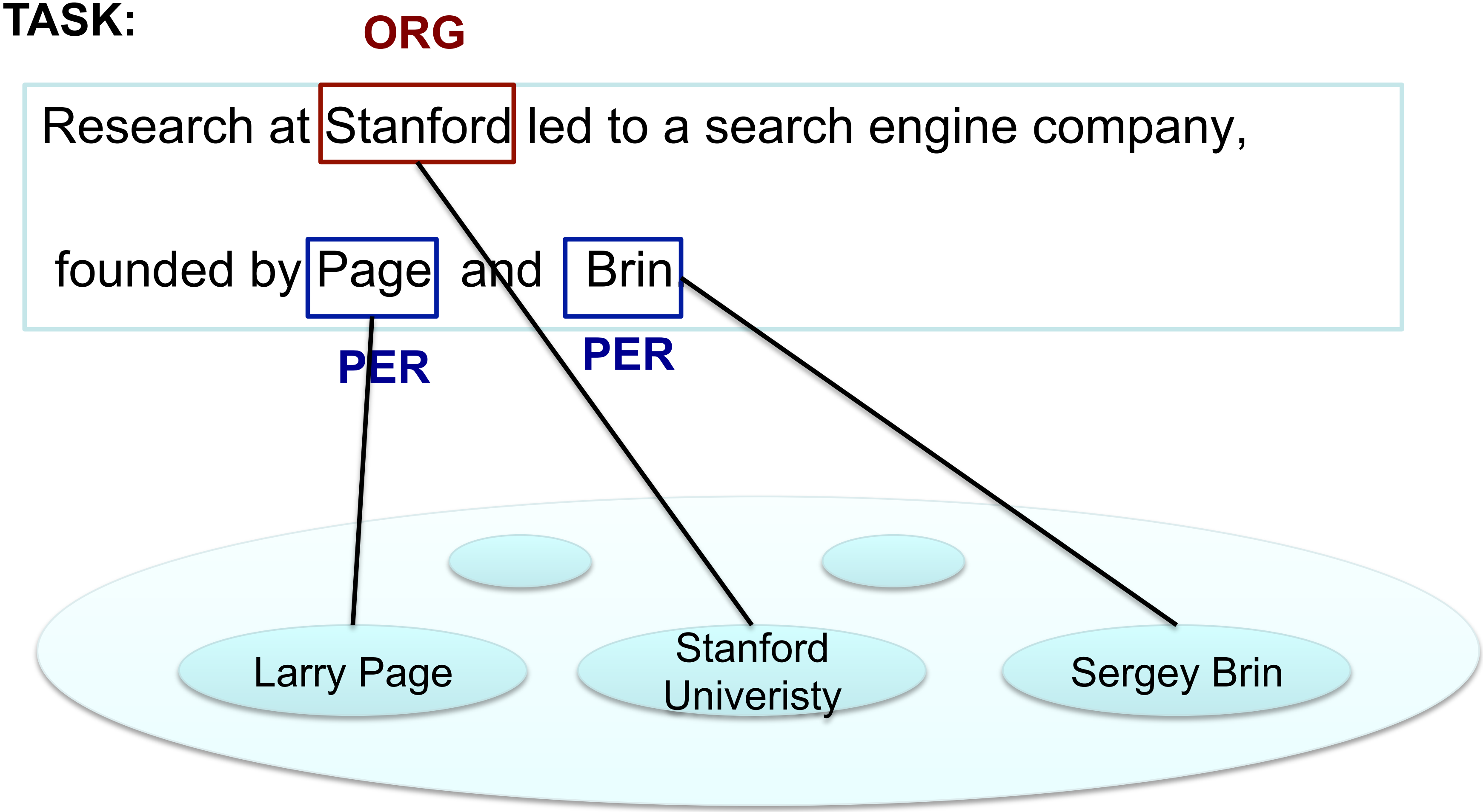
Applications of Word Embeddings

- Named entity recognition.
- Relation extraction.
- Named entity disambiguation.
- Sentiment analysis.
- Dialogue systems.
- …

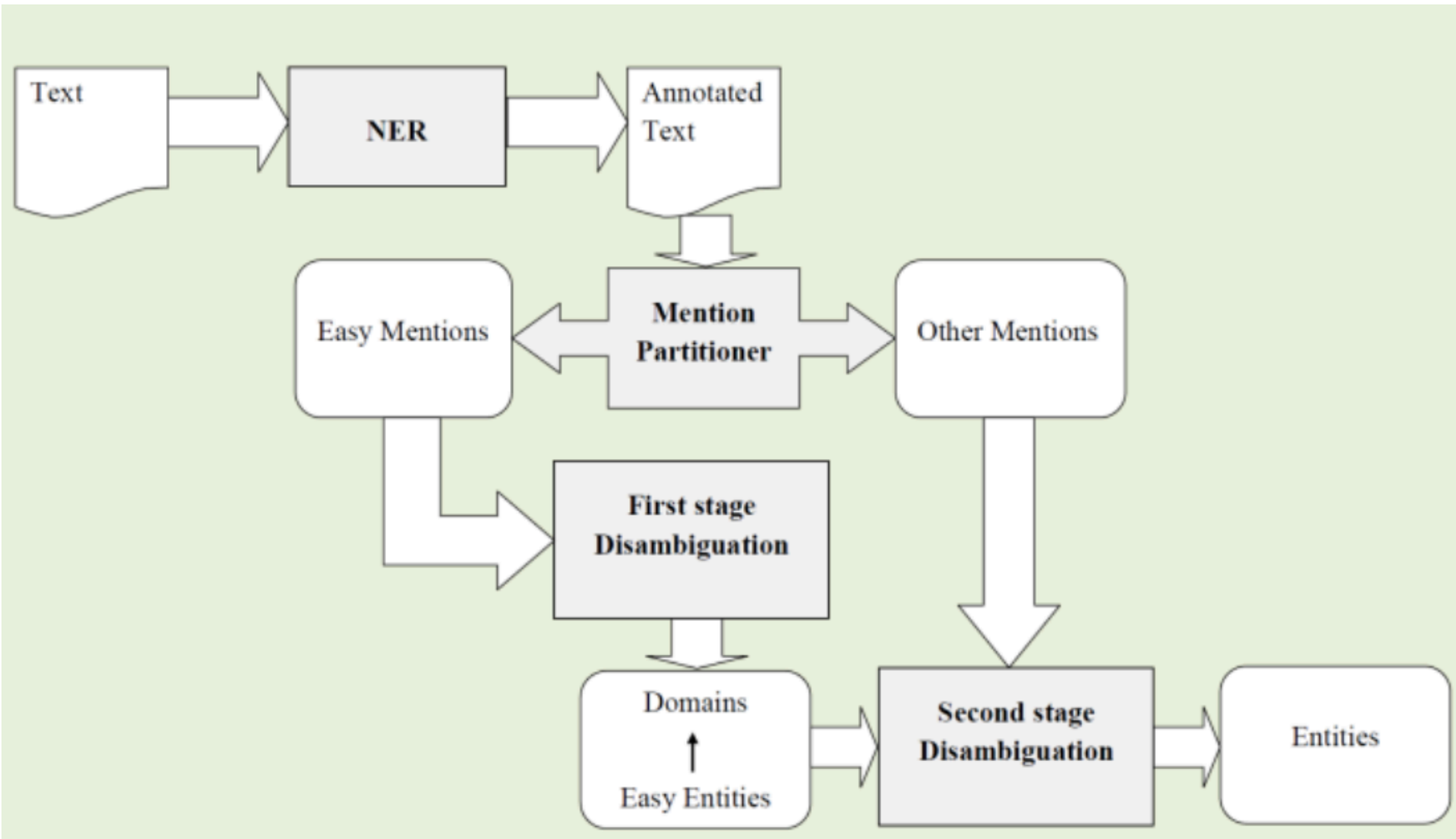


Named Entity Disambiguation (for ALTA shared task)

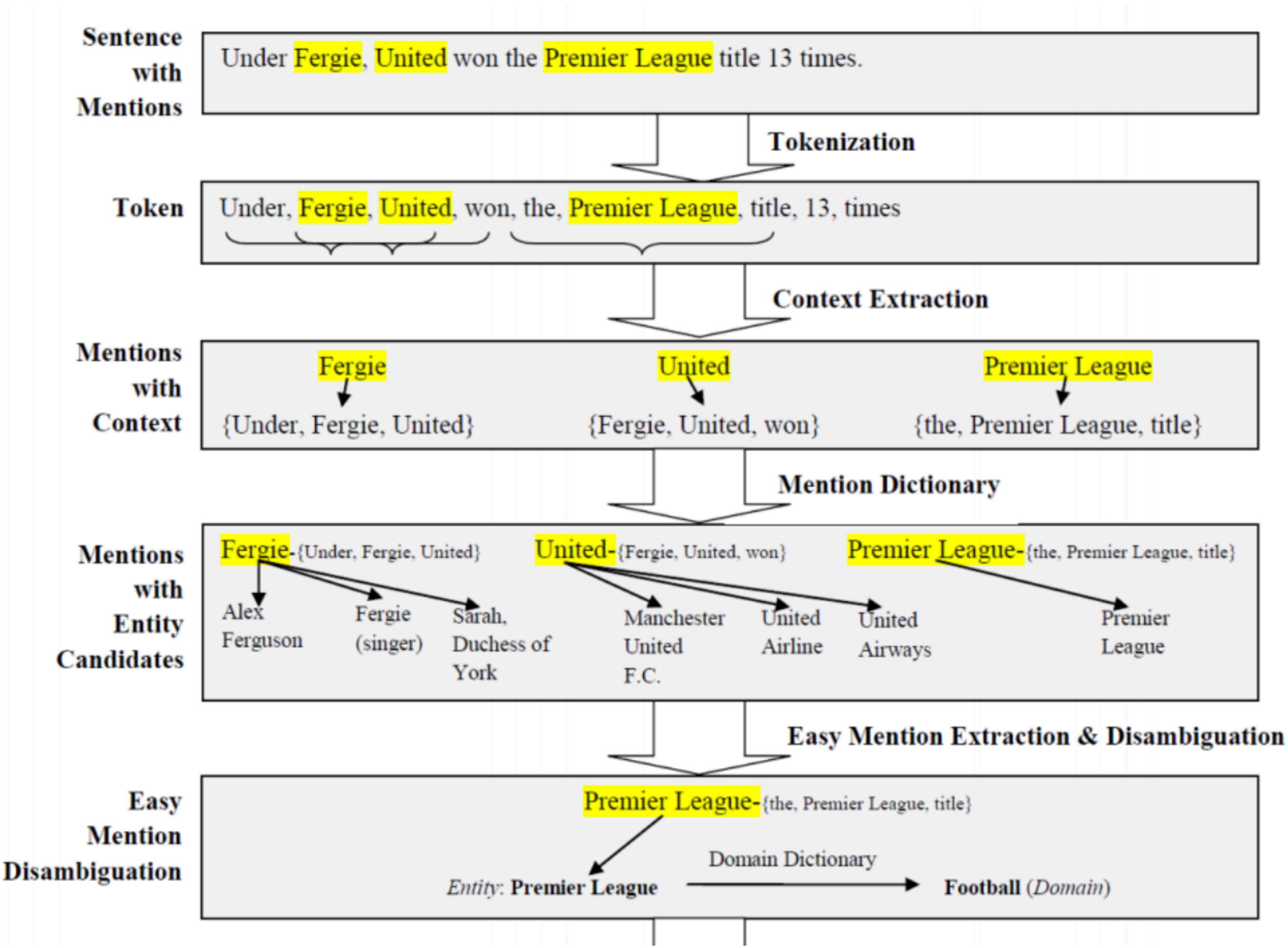
TASK:



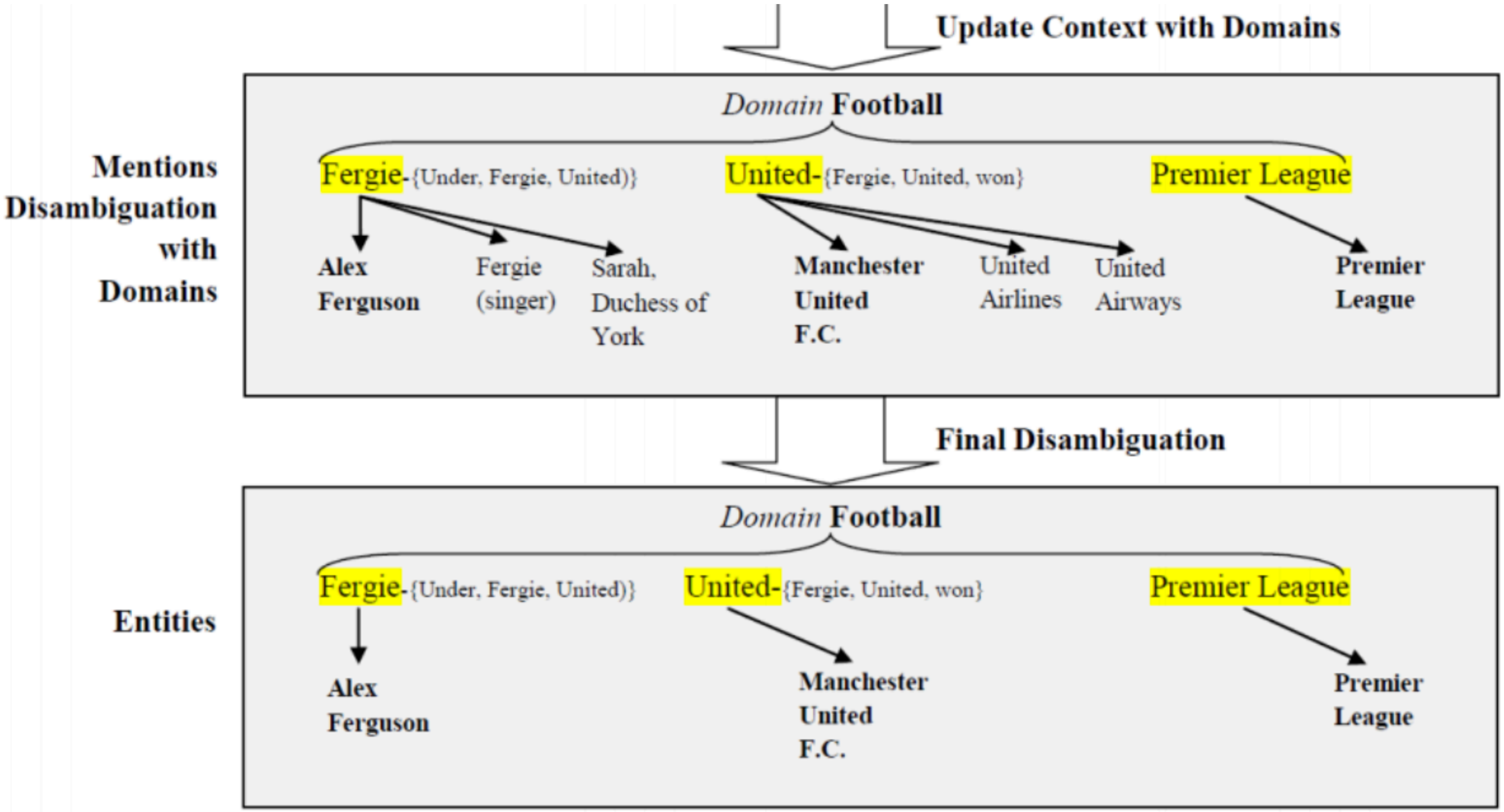
AIDA-light [5] **(for ALTA shared task)**



First Stage **(for ALTA shared task)**



Second Stage (for ALTA shared task)



AIDA-light [5]: 84.8%
DBPepdia spotlight: 75%

ALTA Shared Task

Language Technology Programming Competition 2016

ALTA Shared Task 2016

http://alta.asn.au/events/sharedtask2016

Welcome to the Language Technology Programming Competition. This competition is organised by ALTA.

What is ALTA?

ALTA is the Australasian Language Technology Association. ALTA's mission is to promote language technology research and development in Australia and New Zealand. ALTA organises workshops like ALTA 2016, and supports competitions like the Australian High-School Computational and Linguistics Olympiad (OzCLO) and this programming competition.

What is this competition about?

This is our seventh programming competition and is targeted at university students with programming experience. The competition is formatted as a "shared task": all participants compete to solve the same problem. The problem is the object of active research and nobody has (yet) obtained 100% correct results. In this competition, the winner will be the team who obtains the best results and beats the results of a sample solution that will be provided. If you participate and obtain 100% results you should definitely consider doing a PhD on the topic!

There are no limitations on the size of the teams or the means that they can use to solve the problem, **as long as the processing is fully automatic - there should be no human intervention**. As soon as you join the competition you will have access to a sample of data that you can use to develop your system. To qualify for the prize you need to submit your results and a poster that describes the methods that you used to obtain the results. Selected posters will be displayed at ALTA 2016.

What's New

15 June 2016
Welcome to the ALT

Contact:
Lizhen.Qu@data61.csiro.au
Hanna.Suominen@data61.csiro.au

http://www.alta.asn.au/events/sharedtask2016/

REFERENCES

- [1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013.
- [2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.
- [3] https://code.google.com/archive/p/word2vec/
- [4] Hinton, G. E., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. Neural Computation, 18, 1527–1554.
- [5] Nguyen, Dat Ba, et al. "AIDA-light: High-Throughput Named-Entity Disambiguation." LDOW. 2014.
  - https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/aida/
- [6] Yamada, Ikuya, et al. "Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation." arXiv preprint arXiv:1601.01343 (2016). (http://arxiv.org/pdf/1601.01343.pdf)

Learning Resources

- NLP conferences.
  - ACL, EMNLP, COLING, NAACL, EACL …
- NLP online courses.
  - https://www.coursera.org/course/nlangp
  - https://www.youtube.com/playlist?list=PL6397E4B26D00A269
- ML online courses.
  - https://www.coursera.org/course/ml
  - https://www.coursera.org/course/neuralnets
  - https://www.youtube.com/watch?v=SGZ6BttHMPw&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH