Building a Broad Knowledge Graph for Products

XIN LUNA DONG, AMAZON

ICDE, 4/2019
Acknowledgement
Product Graph vs. Knowledge Graph
Knowledge Graph Example for 2 Movies

Entity

mid345

“Forrest Gump”

name

type

Movie

mid346

“Larry Crowne”

name
type

Relationship

mid127

mid128

mid129

starring

directed_by

name

“Robin Wright”

“Robin Wright Penn”

“Tom Hanks”

“Julia Roberts”

birth_date

type

Entity type

Person

July 9th, 1956
Knowledge Graph in Search

List of Tom Hanks performances - Wikipedia
https://en.wikipedia.org/wiki/List_of_Tom_Hanks_performances

Tom Hanks (@tomhanks) - Twitter
https://twitter.com/tomhanks

And don't miss this songstress at the famous Cafe Carlyle. Through Saturday nite! Hanx @RitaWilson
pic.twitter.com/J70Xjbf...
12 hours ago · Twitter

Beware! Crass self-serving Social Media Post! This book goes on sale tomorrow! Hanx
pic.twitter.com/V2EqPKL...
16 hours ago · Twitter

Lost (g)love. Looking for a mate. Good luck. Hanx.
pic.twitter.com/ApH7tEg...
1 day ago · Twitter

Thomas Jeffrey Hanks is an American actor and filmmaker. He is known for his various comedic and dramatic film roles, including Splash, Big, Turner & Hooch, A League of Their Own, Sleepless in Seattle, ...

Wikipedia

Born: July 9, 1956 (age 61), Concord, CA
Awards: Academy Award for Best Actor, MORE
Spouse: Rita Wilson (m. 1988), Samantha Lewes (m. 1978–1987)
TV shows: Bosom Buddies, Celebrity Jeopardy!, MORE
Knowledge Graph in Personal Assistant

Alexa, play Taylor Swift in the past year

Taylor Swift > Songs

- Love Story
  - Fearless · 2008

- Look What You Made Me…
  - Reputation · 2017

- Shake It Off
  - 1989 · 2014

- Delicate
  - Reputation · 2017
Mission: To answer any question about products and related knowledge in the world
Product Graph vs. Knowledge Graph
Product Graph vs. Knowledge Graph

PG

Generic KG

(D)
Knowledge Graph Example for 2 Movies

- **“Forrest Gump”**
  - Movie
  - mid345
  - name
  - directed_by
  - mid128
  - starring
  - mid127
  - name
  - “Tom Hanks”
  - birth_date
  - July 9th, 1956

- **“Larry Crowne”**
  - Movie
  - mid346
  - name
  - starring
  - mid129
  - name
  - “Julia Roberts”
  - type
  - Person

- **“Robin Wright”**
  - name
  - “Robin Wright Penn”

- **“罗宾·怀特”**
  - name

- **“Tom Hanks”**
  - name

- **“Julia Roberts”**
  - name
  - type
  - Person
Product Graph vs. Knowledge Graph

“Forrest Gump”
- mid345
  - name
  - starring
- mid127
  - name
  - directs

“Larry Crowne”
- mid346
  - name
  - starring
- mid128
  - name
  - directed_by

“Robin Wright”
- mid127
  - name
- mid129
  - name
  - birth_date
  - type

“Robin Wright Penn”
- mid127
  - name

“Tom Hanks”
- mid127
  - name

“Julia Roberts”
- mid129
  - name

July 9th, 1956
- mid129
  - birth_date

Person
Product Graph vs. Knowledge Graph
Another Example of Product Graph
Knowledge Graph vs. Product Graph

(A) Generic KG

(B) Generic KG
   PG (Movie, Music, Book)

(C) Generic KG
   Product Graph
      Movie, Music, Book, etc.
      (Hardline, softline, consumables, etc.)
But, Is The Problem Harder?
Challenges in Building Product Graph I

- No major sources to curate product knowledge from
  - Wikipedia does not help too much
  - A lot of structured data buried in text descriptions in Catalog
  - Retailers gaming with the system so noisy data
Challenges in Building Product Graph II

- Large number of new products everyday
  - Curation is impossible
  - Freshness is a big challenge
Challenges in Building Product Graph III

- Large number of product categories
- A lot of work to manually define ontology
- Hard to catch the trend of new product categories and properties
Challenges in Building Product Graph IV

- Many entities are not named entities
  - Named Entity Recognition does not apply
  - New challenges for extraction, linking, and search
A 100-Year Project
Our Solution: Building a Broad Graph
A Broad & Shallow Version of the Same Graph

- Tide Plus Feb...
- Tide Plus Col...
- Tide
- Liquid
- No
- Detergent
- Odor remover
- type
- function
- brand
- form
- HE
- function

Detergent
Design Principles for Broad Graph

- Start simple
  - Bi-partite graph
  - Core types and relationships
- Grow and clean the graph in a pay-as-you-go fashion
  - Ontology: user log analysis, web extraction
  - Data: product profile extraction, web extraction
  - Cleaning
Input and Output of Broad Graph

**INPUT**

1. A given vertical
2. Amazon Eco System
3. Semi-structured Web

**OUTPUT**

1. Ontology
2. Broad Graph
3. Comprehensive quality metrics

subject to: \( \text{hard\_work} \leq \text{fun} \)
Stages in Building a Broad Graph

Stage 1: Ingestion

Stage 2a: OpenTag text extraction

Coverage: 4X

[Zheng et al., KDD’18]

Size: +7%, predicates: +10%, ExtAccu: 99%

Stage 2b: Ceres web extraction

[Lockard et al., VLDB’18]

Stage 3: Cleaning

Accu: +6%
Stages in Building a Broad Graph

Stage 1: Ingestion

Stage 2a: OpenTag text extraction
[Zheng et al., KDD’18]
Coverage: 4X
Size: +7%, predicates: +10%, ExtAccu: 99%

Stage 2b: Ceres web extraction
[Lockard et al., VLDB’18]

Stage 3: Cleaning
Accu: +6%
Stage 2a. OpenTag Extraction from Product Profiles [KDD’18]

<table>
<thead>
<tr>
<th>name</th>
<th>form</th>
<th>scent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tide Detergent with Febreze Freshness</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gain Apple Mango Tango Liquid Laundry Detergent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gain Joyful Expression Powder Detergent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tide PODS Original Scent HE Turbo Laundry Detergent PACs 81-load Tub</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tide PODS Free &amp; Gentle HE Turbo Laundry Detergent PACs 35-load Bag</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Stage 2a. OpenTag Extraction from Product Profiles [KDD’18]

\[ x = \{w_1, w_2, \ldots, w_n\} \text{ input sequence} \]
\[ y = \{t_1, t_2, \ldots, t_n\} \text{ tagging decision} \]
Stage 2a. OpenTag Extraction from Product Profiles [KDD’18]
Stage 2a. OpenTag Extraction from Product Profiles [KDD’18]

BiLSTM+CRF+Attention obtains best results

Extraction on new values is comparable to already known values
With active learning, we can obtain high precision and recall using only 190 training labels.
Stage 2a. OpenTag Extraction from Product Profiles [KDD’18]

Interpretability via attention
Stages in Building a Broad Graph

Stage 1: Ingestion

Stage 2a: OpenTag text extraction
Coverage: 4X

Stage 2b: Ceres web extraction
Size: +7%, predicates: +10%, ExtAccu: 99%

Stage 3: Cleaning
Accu: +6%
Stage 2b. Extracting Knowledge from Semi-Structured Data on the Web

Aamir Khan is receiving rave reviews for Dangal.

Dangal
Cast: Aamir Khan, Sakshi Tanwar, Fatima Sana Shaikh, Sanya Malhotra
Director: Nitesh Tiwari
Rating: 4/5

臥虎藏龍 (2000)
导演：李安
编剧：王蕙玲 / 詹姆斯·夏慕斯 / 蔡国荣
主演：周润发 / 杨紫琼 / 章子怡 / 张震 / 郑佩佩
类型：剧情 / 动作 / 爱情 / 武侠 / 古装
制片国家/地区：台湾 / 香港 / 中国大陆
语言：汉语普通话
片长：120 分钟
又名：Crouching Tiger, Hidden Dragon
IMDb链接：tt0190332

豆瓣评分：7.9
166740人评价
5星 26.7% 25.6%
4星 45.0% 45.0%
3星 25.6% 25.6%
2星 1.3% 1.3%
1星 0.4% 0.4%

卧虎藏龙的剧情简介：
一代大侠李慕白（周润发饰）有退出江湖之意，托付红颜知己俞秀莲（杨紫琼饰）将青冥剑转交给贝勒爷（郑佩佩饰）收留，不料夜半遭玉娇龙（章子怡饰）袭击。俞秀莲暗中查访也大体知道是金羽府小姐玉蛟龙所为，她想办法迫使玉蛟龙归还宝剑，免受祸害。李慕白发现了玉娇龙的美貌与聪明（郑佩佩饰）的踪迹，她隐藏于金羽府并收玉蛟龙为弟子，而玉蛟龙欲以青冥剑来斩断阻碍李小龙（张震饰）的情缘。他们私定终身，关系变得错综复杂，俞秀莲和李慕白爱惜玉蛟龙人才难得，苦心引导，但玉蛟龙却性任性气不听劝阻……

卧虎藏龙
Stage 2b. Extracting Knowledge from Semi-Structured Data on the Web

- Knowledge Vault @ Google showed big potential from DOM-tree extraction [Dong et al., KDD'14][Dong et al., VLDB’14]

<table>
<thead>
<tr>
<th></th>
<th>Accu</th>
<th>Accu (conf ≥ .7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TXT (301M)</td>
<td>0.36</td>
<td>0.52</td>
</tr>
<tr>
<td>DOM (1280M)</td>
<td>0.43</td>
<td>0.63</td>
</tr>
<tr>
<td>TBL (10M)</td>
<td>0.09</td>
<td>0.62</td>
</tr>
<tr>
<td>ANO (145M)</td>
<td>13K</td>
<td>0.3M</td>
</tr>
</tbody>
</table>
Stage 2b. Extracting Knowledge from Semi-Structured Data on the Web

Extracted relationships

• (Top Gun, type.object.name, “Top Gun”)
• (Top Gun, film.film.genre, Action)
• (Top Gun, film.film.directed_by, Tony Scott)
• (Top Gun, film.film.starring, Tom Cruise)
• (Top Gun, film.film.runtime, “1h 50min”)
• (Top Gun, film.film.release_Date_s, “16 May 1986”)

Title | Genre | Release Date | Runtime | Director | Actors
Stage 2b. Extracting Knowledge from Semi-Structured Data on the Web

Same pred may corr. to diff DOM tree nodes
Stage 2b. Extracting Knowledge from Semi-Structured Data on the Web

Same DOM tree node may correspond to diff preds
Stage 2b. Extracting Knowledge from Semi-Structured Data on the Web

GULHANE ET AL, WEB-SCALE INFORMATION EXTRACTION WITH VERTEX. ICDE 2011
Stage 2b. Linking Entities Between Sources

- Near perfect precision and recall
- Annotations on 5 pages per site

<table>
<thead>
<tr>
<th>Vertical</th>
<th>Predicate</th>
<th>Vertex++</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>P</td>
<td>R</td>
</tr>
<tr>
<td>Movie</td>
<td>Title</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>Director</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>Genre</td>
<td>0.88</td>
<td>0.87</td>
</tr>
<tr>
<td></td>
<td>MPAA Rating</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>0.97</strong></td>
<td><strong>0.97</strong></td>
</tr>
<tr>
<td>NBAPlayer</td>
<td>Name</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>Team</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>Weight</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>Height</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>1.00</strong></td>
<td><strong>1.00</strong></td>
</tr>
<tr>
<td>University</td>
<td>Name</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>Type</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>Phone</td>
<td>0.97</td>
<td>0.92</td>
</tr>
<tr>
<td></td>
<td>Website</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>0.99</strong></td>
<td><strong>0.98</strong></td>
</tr>
<tr>
<td>Book</td>
<td>Title</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>Author</td>
<td>0.97</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>Publisher</td>
<td>0.85</td>
<td>0.85</td>
</tr>
<tr>
<td></td>
<td>Publication Date</td>
<td>0.90</td>
<td>0.90</td>
</tr>
<tr>
<td></td>
<td>ISBN-13</td>
<td>0.94</td>
<td>0.94</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>0.93</strong></td>
<td><strong>0.93</strong></td>
</tr>
</tbody>
</table>
Stage 2b. Linking Entities Between Sources

- Random forest on attribute-wise similarity
- Results between Freebase and IMDb movies

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movie</td>
<td>99.0%</td>
<td>98.7%</td>
</tr>
<tr>
<td>People</td>
<td>99.3%</td>
<td>99.6%</td>
</tr>
</tbody>
</table>

1.5M labels
Stage 2b. Linking Entities Between Sources

Apply active learning to minimize #labels

For 99% precision and recall, active learning reduces #labels by 2 orders of magnitude.

With 15K labels we get prec=99% and rec=95% (30 labelers for 1 week!)

Reaching prec=99% and rec=99% requires 1.5M labels.
Stages in Building a Broad Graph

Stage 1: Ingestion

Stage 2a: OpenTag text extraction
- Coverage: 4X
- [Zheng et al., KDD’18]
- Size: +7%, predicates: +10%, ExtAccu: 99%

Stage 2b: Ceres web extraction
- [Lockard et al., VLDB’18]

Stage 3: Cleaning

Accu: +6%
Stage 3. Supervised Knowledge Cleaning

- Value noise
  - Invalid brands: “1 lb”
  - Lengthy descriptive brands: “The company X was founded in...”
  - Unnormalized values: “Reese’s”, “reese”, “REESE’s”

- Inconsistency
  - E.g., isSugarFree && sugarPerServing > 0
Stage 3. Supervised Knowledge Cleaning I

- A discriminative model to tell: Brand or Not a Brand?
- Features: frequency count, string lengths, special word hit, etc.
- Precision = 99.1%
Stage 3. Supervised Knowledge Cleaning II

- Embedding-based filtering

  Taking *Flavor* as an example

  Score = \( \text{Cosine}(a, b) \)

\[ \text{Mean} \]

- Original
- Apple
- Mango
- Hershey’s

*OpenTag* Value Embedding
Stage 3. Supervised Knowledge Cleaning III

- Relation embedding filtering

Learn embedding from IMDb data and identify WikiData errors

<table>
<thead>
<tr>
<th>Subject</th>
<th>Relation</th>
<th>Target</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Moises Padilla Story</td>
<td>writtenBy</td>
<td>César Ámigo Aguilar</td>
<td>Linkage error</td>
</tr>
<tr>
<td>Bajrangi Bhaijaan</td>
<td>writtenBy</td>
<td>Yo Yo Honey Singh</td>
<td>Wrong relationship</td>
</tr>
<tr>
<td>Piste noire</td>
<td>writtenBy</td>
<td>Jalil Naciri</td>
<td>Wrong relationship</td>
</tr>
<tr>
<td>Enter the Ninja</td>
<td>musicComposedBy</td>
<td>Michael Lewis</td>
<td>Linkage error</td>
</tr>
<tr>
<td>The Secret Life of Words</td>
<td>musicComposedBy</td>
<td>Hal Hartley</td>
<td>Cannot confirm</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Broad Graph Summary

- Quick development: ~15 person X month
- Easy ontology design and extension: 2W design
- Higher coverage: +8%~+68%
- Web extraction: +7% new triples, +10% new predicates, with 99% extraction accuracy
- Higher accuracy: +2%~+22%
- Live on Alexa Shopping, being deployed in Amazon Search and Browse
From Broad Graph to Rich Graph

Broad, shallow graph

Rich, deep graph
Are We There Yet?
Success Criteria

- Success criteria for Product Graph
  - #Papers?
  - Team size?
  - #Production applications?
  - $$?

- Becoming part of people’s daily lives
Knowledge Graph Construction Requires Data Integration & Cleaning (DI & DC)?

One vertical, A few sources

Effective search, mining and analysis

Hierarchy of thousands of types

Thousands-to-millions of sources

Big challenge: Limited training labels for large-scale, rich data
How to Get DI & DC to the Next Level of Success?

- Challenges: Limited training labels for large-scale, rich data
- Solution: Unsupervised learning
How to Get DI & DC to the Next Level of Success?

- Challenges: Limited training labels for large-scale, rich data
- Solution: Learning with limited labels
  - Active learning
  - Weak learning (e.g., distance supervision, data programming)
  - Semi-supervised learning (e.g., graph-based learning)
  - Transfer learning
  - One/few-shot learning
Research Philosophy

**Roofshots**: Deliver incrementally and make production impacts

**Moonshots**: Strive to apply and invent the state-of-the-art
Moonshot: Open Knowledge Extraction and QA from Semi-Structured Web
Ceres: Automatic ClosedIE from Semi-structured Web [VLDB’18]

Weak learning

Training (on noisy labels)

Movie entity

Genre

Release Date

Runtime

Director

Actors

Extracted triples

- (Top Gun, type.object.name, “Top Gun”)
- (Top Gun, film.film.genre, Action)
- (Top Gun, film.film.directed_by, Tony Scott)
- (Top Gun, film.film.starring, Tom Cruise)
- (Top Gun, film.film.runtime, “1h 50min”)
- (Top Gun, film.film.release_Date_s, “16 May 1986”)
Ceres: Automatic ClosedIE from Semi-structured Web [VLDB’18]

<table>
<thead>
<tr>
<th>#Websites / #Webpages</th>
<th>33 / 434K</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td>English and 6 other languages</td>
</tr>
<tr>
<td>Domains</td>
<td>Animated films, Documentary films, Financial performance, etc.</td>
</tr>
<tr>
<td># Annotated pages</td>
<td>70K (16%)</td>
</tr>
<tr>
<td>Annotated : Extracted #entities</td>
<td>1 : 2.6</td>
</tr>
<tr>
<td>Annotated : Extracted #triples</td>
<td>1 : 3.0</td>
</tr>
<tr>
<td># Extractions</td>
<td>1.25 M</td>
</tr>
<tr>
<td>Precision</td>
<td>90%</td>
</tr>
</tbody>
</table>
Ceres: Automatic ClosedIE from Semi-structured Web [VLDB’18]

- Extraction on 33 movie websites with 7 languages

Precision=0.9
Yield=1.25MM
OpenCeres: OpenIE to Identify New Predicates [NAACL’19]

- **ClosedIE**: Only extracting facts corresponding to ontology
  - (“When Harry Met Sally…”, `film.film.directed_by`, “Rob Reiner”)

- **OpenIE**: Extract all relations expressed on the webpage
  - (“When Harry Met Sally…”, “Director”, “Rob Reiner”)

[Image of movie poster]
OpenCeres: OpenIE to Identify New Predicates [NAACL’19]

- **ClosedIE:** Normalize predicates by ontology
  
  (“When Harry Met Sally...”, `film.film.directed_by`, “Rob Reiner”)

- **OpenIE:** Predicates are unnormalized strings
  
  (“When Harry Met Sally...”, “Directed By”, “Rob Reiner”)

---

**Rotten Tomatoes**

MOVIE INFO

Does sex make it impossible for men and women to b dilemma through the eleven year relationship between their own lives until they reconnect ten years later.

- **Rating:** R
- **Genre:** Comedy, Drama, Romance
- **Directed By:** Rob Reiner
- **In Theaters:** Jul 12, 1989 Wide
- **On Disc/Streaming:** Oct 13, 1998
- **Runtime:** 96 minutes
OpenCerces: OpenIE to Identify New Predicates [NAACL’19]

Extracted triples
- (“Top Gun”, “Director”, “Tony Scott”)
- (“Top Gun”, “Writers”, “Jim Cash”)
- (“Top Gun”, “Stars”, “Tom Cruise”)
- (“Top Gun”, “Stars”, “Tim Robbins”)

Semi-supervised learning
OpenCeres: OpenIE to Identify New Predicates [NAACL’19]

Movie
- Seed: Director, Writer, Producer, Actor, Release Date, Genre, Alternate Title
- New: Country, Filmed In, Language, MPAA Rating, Set In, Reviewed by, Studio, Metascore, Box Office, Distributor, Tagline, Budget, Sound Mix

NBA Player
- Seed: Height, Weight, Team
- New: Birth Date, Birth Place, Salary, Age, Experience, Position, College, Year Drafted

University
- Seed: Phone Number, Web address, Type (public/private)
- New: Calendar System, Enrollment, Highest Degree, Local Area, Student Services, President
OpenCeres: OpenIE to Identify New Predicates [NAACL’19]

Extraction on long-tail movie websites

OpenIE added significant amount of knowledge

Still need precision improvement on new relations
OpenKI: Universal Schema to Cluster OpenIE Predicates [NAACL’19]

Unsupervised pre-training for different granularities
QA Preliminary Results: Question → Predicate

- **Easy questions we get right**
  - “How many calories are there in X” → caloriesPerServing
  - “Can I eat X while on a low carbohydrates diet?” → carbsPerServing
  - “Is X nonGMO?” → isGMOFree

- **Hard questions we get right**
  - “Is X OK for people with celiac disease” → isGlutenFree
  - “Was X grown without pesticides” → IsOrganic
  - “How long can I store X” → shelfLife
  - “How many ounces of X do I get” → Weight
  - “Is X a Halal food” → IsKosher
QA Preliminary Results:
Questions → Question

- Is it vegetarian?
- Are the tomatoes grown without pesticides?
- Are the seeds organic?
- Are the ingredients all vegetarian?
- Are any ingredients vegetarian?
- Does it contain MSG?
- Does it contain gelatine?
- Does it contain meat?

- Is it vegetarian friendly?
- Were the seeds grown irradiated?
- Have the seeds been chemically irradiated?
- Have the seeds been pesticide tested?
- Does the seed contain sulfites?
- Is the fennel USDA irradiated?
- Does your fennel seeds irradiated?
- Does these fennel seeds irradiated?
- Are the seeds organic?
- Are the tomatoes organic?
## QA Preliminary Results: Predicate $\rightarrow$ Question phrase

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Question phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>isGlutenFree</td>
<td>contains_wheat, gluten_free, gluten, fermented_soy, soy</td>
</tr>
<tr>
<td>packageSize</td>
<td>how_many_ounces, bulk_package, 1_lb, 10_oz, ounces, lb, bulk, oz, package</td>
</tr>
<tr>
<td>flavor</td>
<td>taste_like, taste, flavored, flavors</td>
</tr>
<tr>
<td>kosher</td>
<td>kosher_certified, kosher, halal, certified_kosher</td>
</tr>
<tr>
<td>fatPerServing</td>
<td>saturated_fat, saturated, fat</td>
</tr>
<tr>
<td>isVegetarian</td>
<td>suitable_for_vegetarians, meat, vegetarians, vegetarian</td>
</tr>
</tbody>
</table>

Unsupervised pre-training
### StarFinder: Finding Entity Importance

#### Top 15 Among 1MM Artists

<table>
<thead>
<tr>
<th>Score</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>8262</td>
<td>Johann Sebastian Bach</td>
</tr>
<tr>
<td>7039</td>
<td>Wolfgang Amadeus Mozart</td>
</tr>
<tr>
<td>5028</td>
<td>Ludwig van Beethoven</td>
</tr>
<tr>
<td>3591</td>
<td>Frank Sinatra</td>
</tr>
<tr>
<td>3118</td>
<td>Elvis Presley</td>
</tr>
<tr>
<td>3089</td>
<td>Herbert von Karajan</td>
</tr>
<tr>
<td>3052</td>
<td>Frederic Chopin</td>
</tr>
<tr>
<td>3001</td>
<td>John Williams</td>
</tr>
<tr>
<td>2944</td>
<td>Antonio Vivaldi</td>
</tr>
<tr>
<td>2746</td>
<td>Manfred Eicher</td>
</tr>
<tr>
<td>2638</td>
<td>Norman Granz</td>
</tr>
<tr>
<td>2616</td>
<td>Grateful Dead</td>
</tr>
<tr>
<td>2449</td>
<td>Ella Fitzgerald</td>
</tr>
</tbody>
</table>

#### Exploring some other nodes

<table>
<thead>
<tr>
<th>Rank</th>
<th>Score</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>228</td>
<td>726</td>
<td>hans zimmer</td>
</tr>
<tr>
<td>267</td>
<td>668</td>
<td>eminem</td>
</tr>
<tr>
<td>282</td>
<td>650</td>
<td>michael jackson</td>
</tr>
<tr>
<td>369</td>
<td>559</td>
<td>beatles</td>
</tr>
<tr>
<td>632</td>
<td>412</td>
<td>cher</td>
</tr>
<tr>
<td>678</td>
<td>398</td>
<td>taylor swift</td>
</tr>
</tbody>
</table>

- Semi-supervised learning with GCN
Take Aways

- We aim at building an authoritative knowledge graph for all products in the world
- We shoot for roofshot and moonshot goals to realize our mission
- The next-generation of KG could be a combination of rich graph and broad graph
- Learning with limited labels is the rescue to get DI & DC to the next level of success
Thank You!

QUESTIONS?