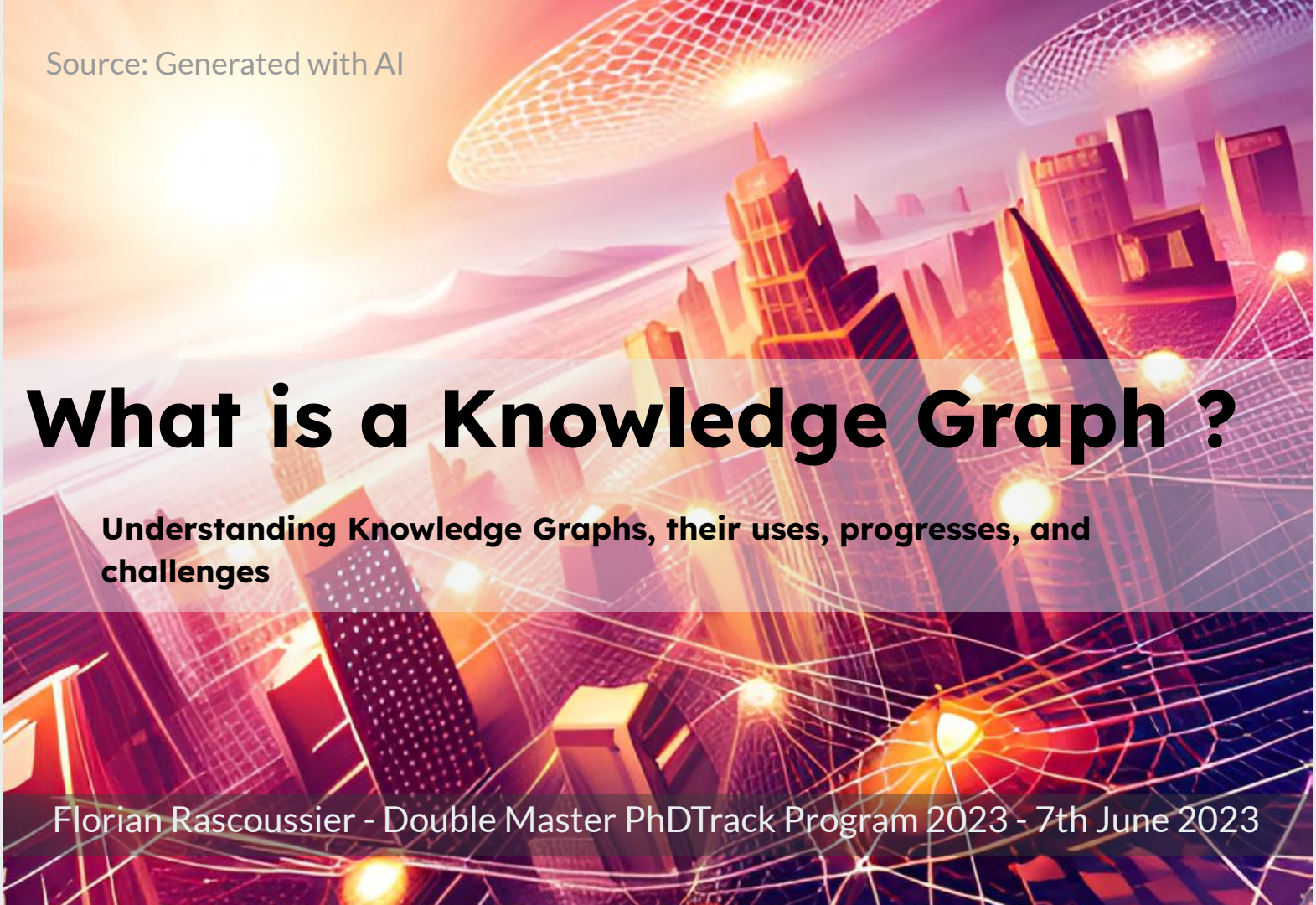


What is a Knowledge Graph ?

Understanding Knowledge Graphs, their uses, progresses, and challenges

Florian Rascoussier - Double Master PhDTrack Program 2023 - 7th June 2023

Plan

1. Introduction & History
2. So, what is a KG ?
3. KG construction introduction
4. Advanced KG-related concepts
5. Critical Evaluation of the papers
6. Implementation example
7. Conclusion

Source: [Wikipedia]

Sources & References

Main source:

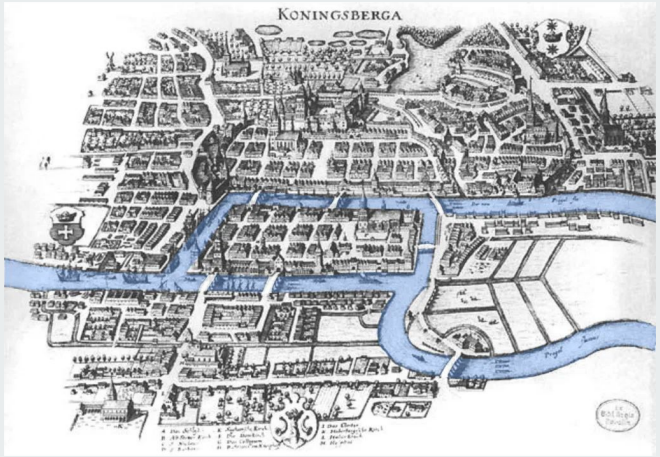
[KG21] <ACM Compt.> Aidan Hogan u. a. „*Knowledge Graphs*“. In: ACM Comput. Surv. 54.4 (Juli 2021). issn: 0360-0300. doi: 10.1145/3447772. url: https://doi.org/10.1145/3447772.3

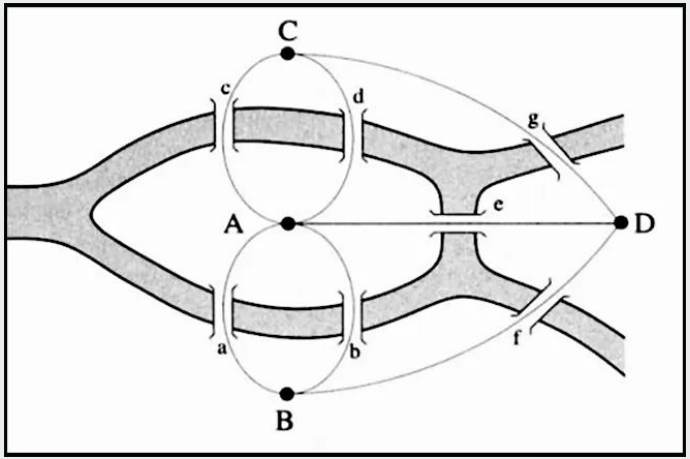
Additional sources:

[KGKE22] <Dagstuhl Reports> Paul Groth u. a. „*Knowledge Graphs and their Role in the Knowledge Engineering of the 21st Century*“. In: Dagstuhl Reports 12.9 (2022). Report from Dagstuhl Seminar 22372, S. 60–120. doi: 10.4230/DagRep.12.9.60. Specific usage: pp. 60-72, Subsection "3.2 A Brief History of Knowledge Engineering: A Practitioner’s Perspective", doi: 10.4230/DagRep.12.9.60.

[CKG23] <preprint> Marvin Hofer u. a. „*Construction of Knowledge Graphs: State and Challenges*“. In: arXiv preprint arXiv:2302.11509 (2023). url: https://doi.org/10.48550/arXiv.2302.11509.

Introduction & History





Source: [towardsdatascience.com]

This section is based on [KGKE22]

Introduction

Source: [Wikipedia]

Knowledge Graph:

A tool for **storing** and **organizing** information…

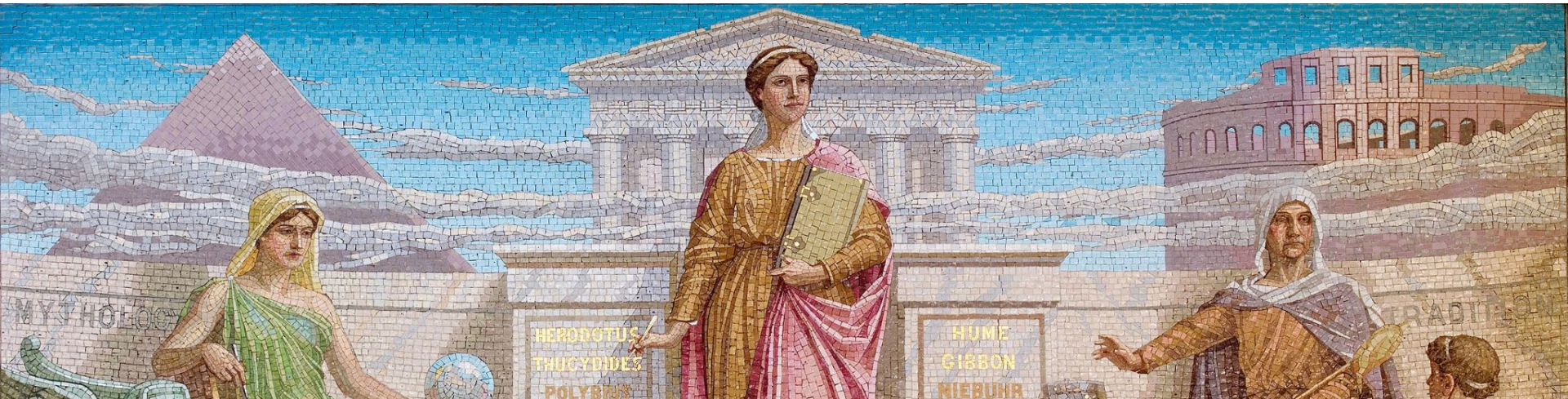
… with many real-world use cases

Aim: Understanding knowledge graphs - its function, operation, and increasing relevance in businesses and organizations.

Historical Evolution of Knowledge Engineering (KE)

- From Expert Systems in the 1980s to the current Language Model Era
- [KGKE22] identifies 4 key periods for Knowledge engineering
- Each phase introduced new requirements for knowledge production

Source: [Wikipedia]



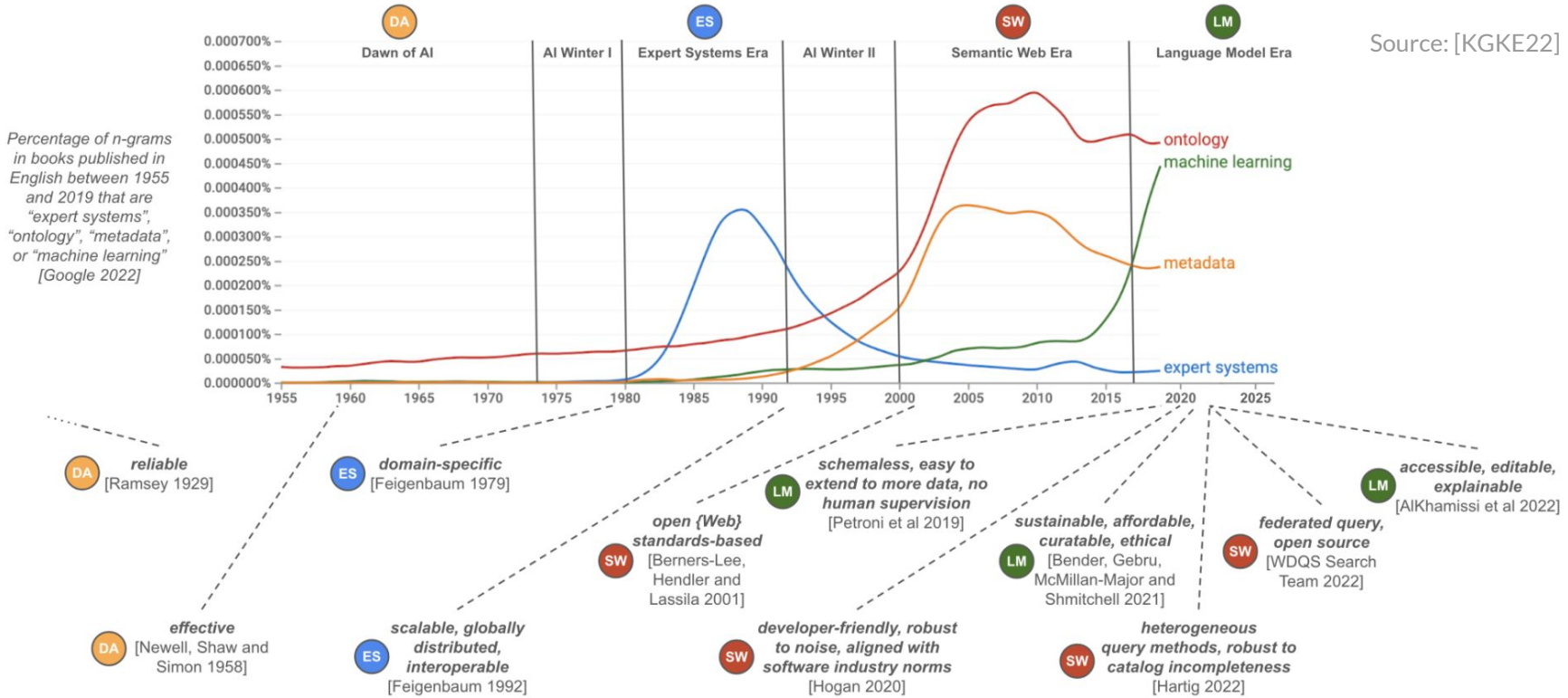
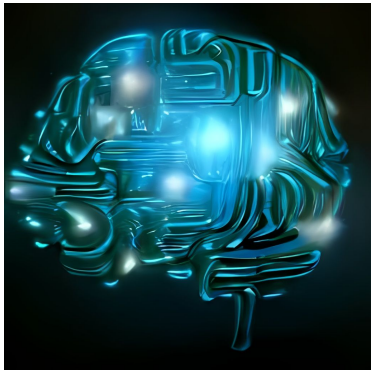


Figure 2 Seventy years of evolving requirements for knowledge production processes [1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13].

From early developments...



Source: [Wikipedia]

1. Dawn of AI

- Focus on reliability and effectiveness in problem-solving processes

2. Expert Systems Era

- Emphasis on domain-specific focus for automated knowledge production
- Challenges: brittle systems, hard to maintain

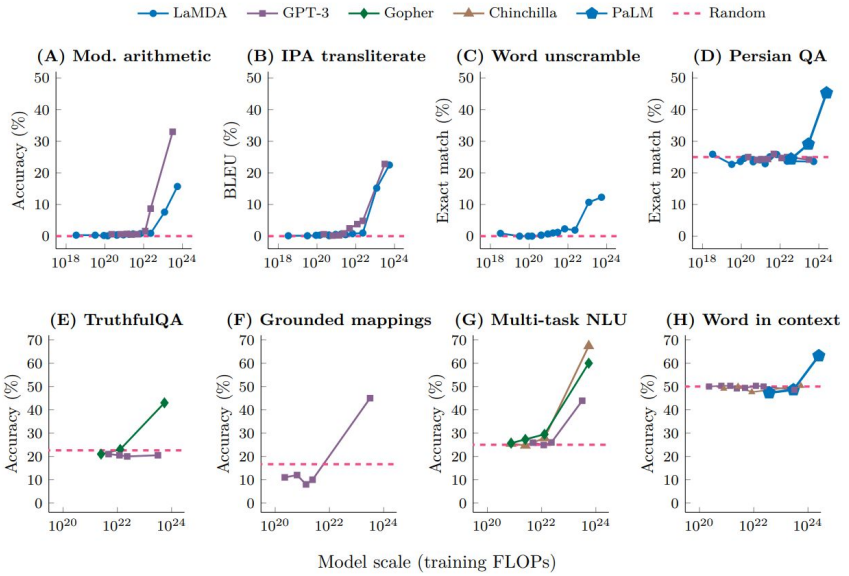
… to the XXIst century

3. Semantic Web Era

- Advocated for a “Web of Data” - linked data principles, standard ontologies, data sharing protocols
- Creation of a globally federated open linked data cloud
- Challenges: **slow adoption**, need for developer-friendly tools

4. Language Model Era

- Advancements in neural network architectures and hardware
- Language models as knowledge bases, or components in a knowledge production workflow



Source: [Wikipedia]

So, what *is* a KG ?

This section is based on [KG21]

Knowledge ? Graph Theory ?

“Explicit” Knowledge:

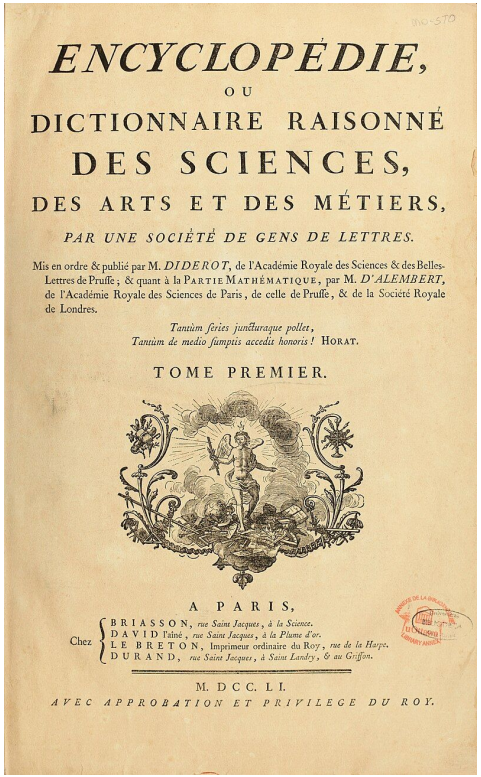
- Known, **recordable** information

  "something that is known and can be written down" [KG21 p4]

- Sequences of words establishing relationships between concepts and data

Graph Theory:

- Bridges computer science and mathematics
- Graphs: Data structures composed of **nodes** (vertices) and **edges** (arcs)
- Analyzes various relationship types and structures across multiple fields


ENCYCLOPÉDIE,
OU
DICTIONNAIRE RAISONNÉ
DES SCIENCES,
DES ARTS ET DES MÉTIERS,
PAR UNE SOCIÉTÉ DE GENS DE LETTRES.
Mis en ordre & publié par M. DIDEROT, de l'Académie Royale des Sciences & des Belles-Lettres de Prusse; & quant à la PARTIE MATHÉMATIQUE, par M. D'ALEMBERT, de l'Académie Royale des Sciences de Paris, de celle de Prusse, & de la Société Royale de Londres.
Tantùm series juncturaque pollet,
Tantùm de medio sumptis accedit honoris! HORAT.
TOME PREMIER.
A PARIS,
Chez BRIASSON, rue Saint Jacques, à la Science.
DAVID l'aîné, rue Saint Jacques, à la Plume d'or.
LE BRETON, Imprimeur ordinaire du Roy, rue de la Harpe.
DURAND, rue Saint Jacques, à Saint Landry, & au Griffon.
M. DCC. LI.
AVEC APPROBATION ET PRIVILEGE DU ROY.

Source: [Wikipedia]

KG

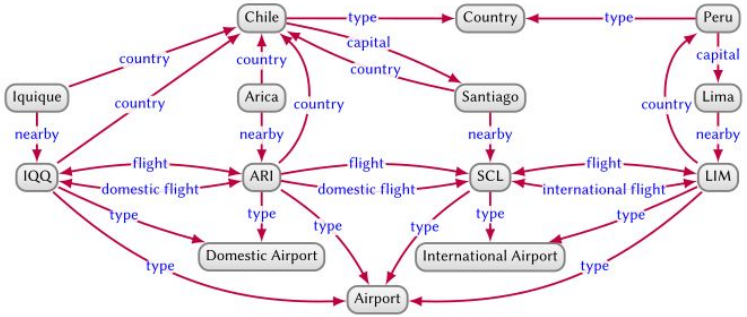


Fig. 21. An incomplete del graph describing flights between airports.

Source: [KG21]

Knowledge Graph:

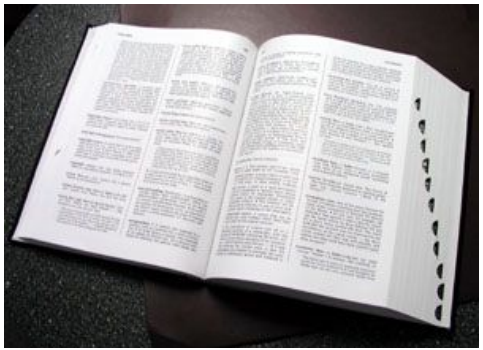
- A data graph intended to accumulate and convey real-world knowledge
- **Nodes** represent **entities**, **edges** represent **relations** between these entities
- Serves as a common substrate for knowledge representation and dissemination

"At the foundation of any knowledge graph is the principle of first modelling data as a graph" [KG21 p4]

This slide is based on [CKG23]

Defining KG is not easy...

- The term "knowledge graph" dates back to 1973. [ISIFSCD73]
- It gained popularity through a 2012 blog post about Google's Knowledge Graph.
- Several definitions of knowledge graphs have been proposed in research papers and by companies using or supporting KGs.

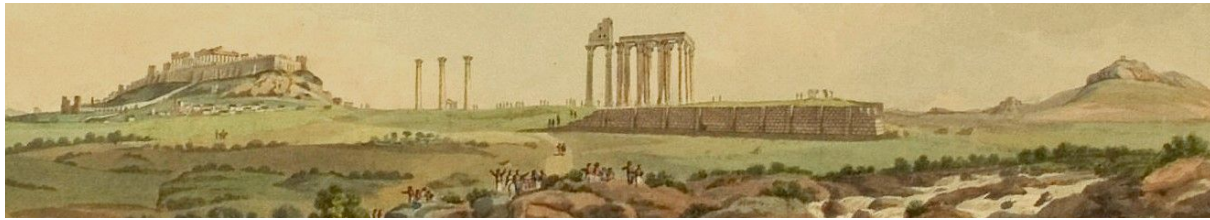


Source: [Wikipedia]

This slide is based on several sources

Many definitions

Source: [Wikipedia]



- "A knowledge graph acquires and integrates information into an ontology and applies a reasoner to derive new knowledge." [TDKG16]
- “A graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent potentially different relations between these entities.” [KG21]
- “A graph of data consisting of semantically described entities and relations of different types that are integrated from different sources. Entities have a unique identifier. KG entities and relations are semantically described using an ontology or, more clearly, an ontological representation.” [CKG23]

Understanding KG

- Fundamental principle: Modeling data as a graph
- Flexible method to represent and integrate diverse and incomplete data
- Flexible solution, benefits over relational or NoSQL models
    - **Flexibility**: Adapt to evolving relationships without changing database structure
    - Tolerance for **Incompleteness**: Handle missing or ambiguous data effectively
    - **Semantic** Interlinking: Give meaning over data and relations, context
    - **Scalability**: Accommodate large datasets and integrate new sources seamlessly
    - Advanced **Queries**: Support complex queries due to graph database nature
    - Knowledge **Representation**: Ideal for AI and Machine Learning tasks

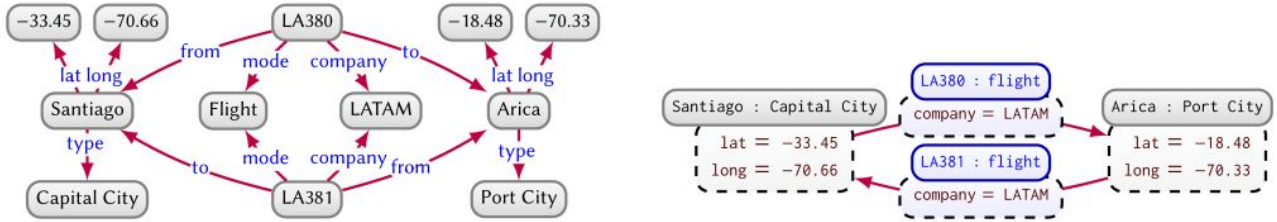
Types of KG

- Directed Edge-labelled Graphs (DEL)
- Heterogeneous Graphs
- Property Graphs

NB: KGs can adopt any graph data model; conversion possible between models.

(a) Del graph (b) Heterogeneous graph

Fig. 2. Data about capitals and countries in a del graph and a heterogeneous graph.

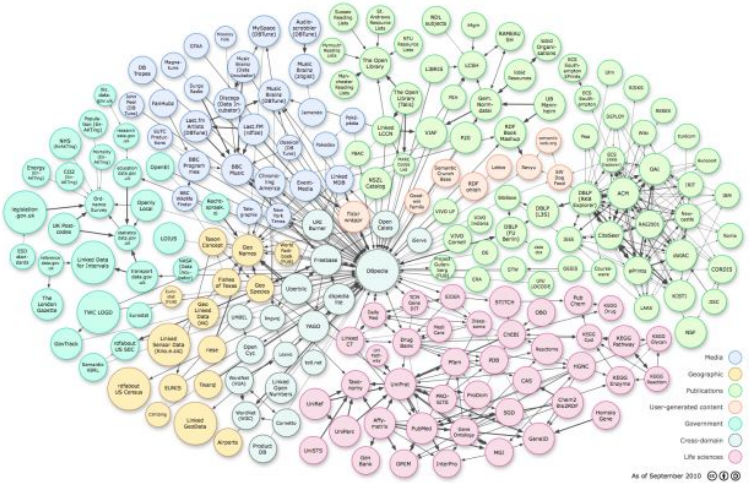


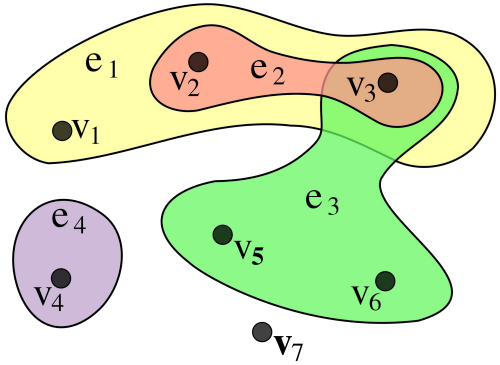
(a) Del graph (b) Property graph

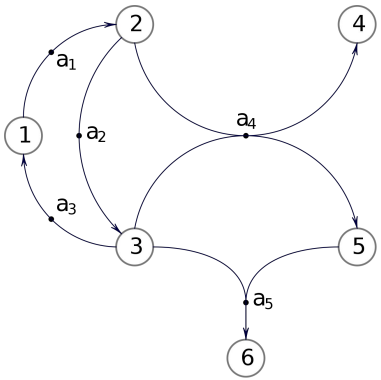
Fig. 3. Flight data in a del graph and a property graph.

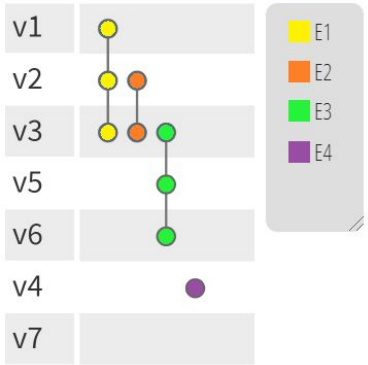
Source: [KG21]

- Graph Dataset
- Hypergraphs









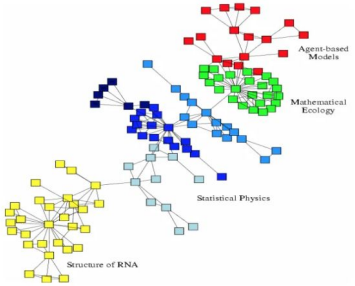
Source: [Wikipedia]

Source: [towardsdatascience.com]

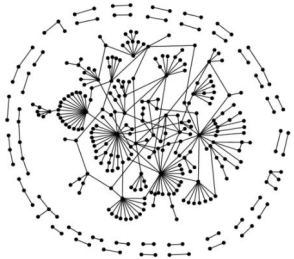
A wide range of use-case

- Flexibility allows to model data is different fields
  - Web Search
  - Commerce
  - Social Networks
  - Finance
  - …
- Other Applications like Information extraction, personal agents, advertising, automation, etc.

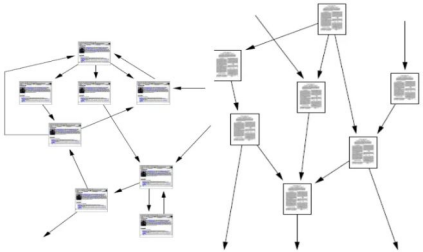
Social networks



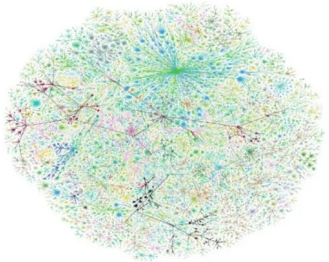
Economic networks



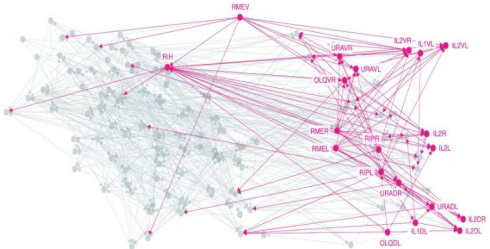
Biomedical networks



Information networks:
Web & citations



Internet



Networks of neurons

Famous KGs

Open Knowledge Graphs:

- **BabelNet**: Integrates several resources including Wikipedia and WordNet for multilingual lexical knowledge
- **DBpedia**: Extracts structured content from Wikipedia to make it accessible on the Web
- **Freebase**: Crowdsourced database of well-known people, places, and things
- **Wikidata**: Central storage for the structured data of Wikimedia projects
- **YAGO**: Automatically extracts and integrates knowledge from Wikipedia and other sources

Enterprise Knowledge Graphs:

- **Google Knowledge Graph**: Enhances Google Search's results with semantic-search information gathered from various sources
- **Amazon Product Knowledge Graph** (PKG) is a large-scale, semi-structured knowledge graph that organizes information about products sold on Amazon and relationships between them.

	Year	Domain	Srcs.	Model	Entities	Relations	Types	R-Types	Vers.	Update
Closed KG										
Google KG [195]	2012	Cross,MLang	>>>1	Custom,RDF	1B	>100B	?	?	?	?
Diffbot.com	2019	Cross	>>>1	RDF	5.9B	>1T	?	?	?	?
Amazon PG [196]	2020	Products	>1	Custom	30M	1B	19K	1K	?	?
Open Access KG										
*Freebase [197]	2007	Cross	>>1	RDF	22M	3.2B	53K	70K	>1	2016
DBpedia [198]	2007	Cross,MLang	140	RDF	50M	21B	1.3K	55K	>20	2023
YAGO [199] [200]	2007	Cross	2-3	RDF(-Star)	67M	2B	10K	157	5	2020
NELL [201]	2010	Cross	⩾1	Custom,RDF	2M	2.8M	1.2K	834	>1100	2018
*Wikidata [202]	2012	Cross,MLang	>>>1	RDB/RDF	100M	14B	300K	10.3K	>100	2023
DBpedia-EN Live [203]	2012	Cross	1	RDF	7.6M	1.1B	800	1.3K	>>>1	2023
Artist-KG [204]	2016	Artists	4	Custom	161K	15M	>1	18	1	2016
*ORKG [205]	2019	Research	>>1	RDF	130K	870K	1.3K	6.3K	>1	2023
AI-KG [206]	2020	AI Science	3	RDF	820K	1.2M	5	27	2	2020
CovidGraph [207]	2020	COVID-19	17	PGM	36M	59M	128	171	>1	2020
DRKG [208]	2020	BioMedicine	>7	CSV	97K	5.8M	17	107	1	2020
VisualSem [209]	2020	Cross,MLang	2	Custom	90k	1.5M	(49K)	13	2	2020
WorldKG [210]	2021	Geographic	1	RDF	113M	829M	1176	1820	1	2021

Source: [CKG23]

KG construction introduction

This section is based on [CKG21]

What's the issue ?

- Goal: having a "useful" and "usable" organisation of information

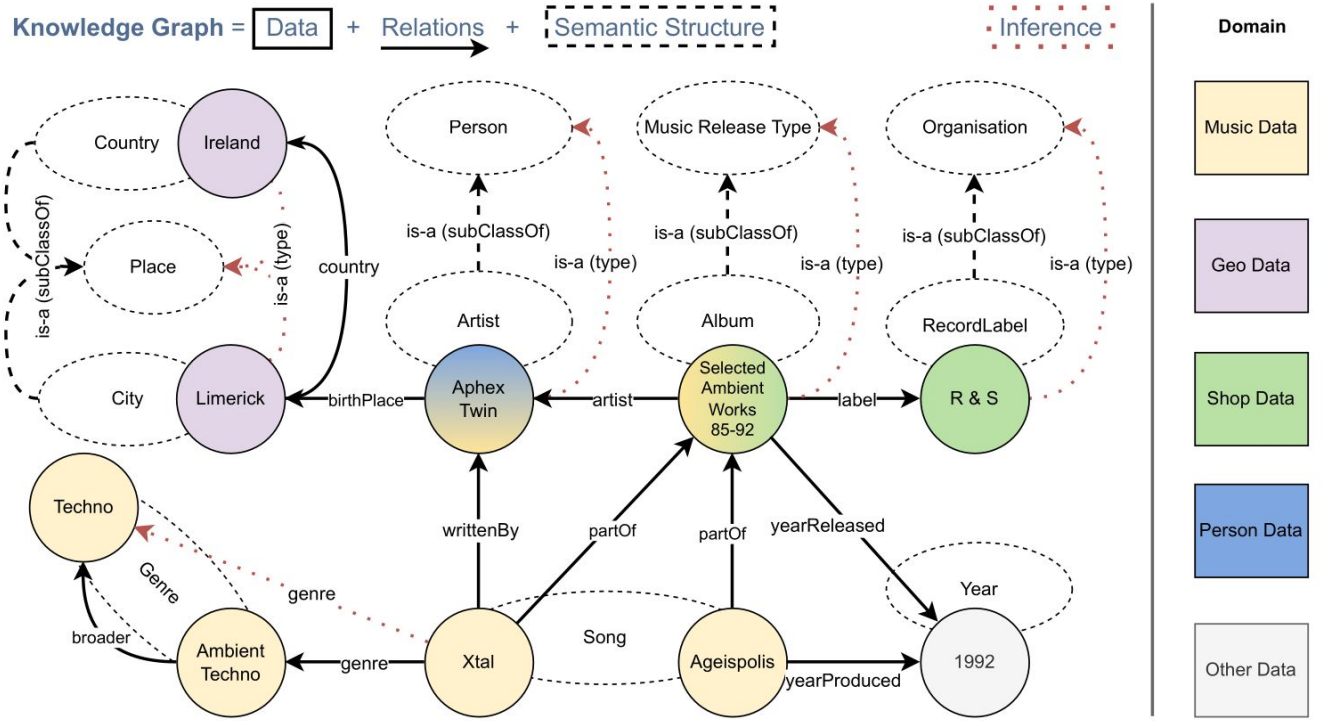


Fig. 1. **Simplified KG example** with integrated data from several domains. Entities and relations are described by an underlying ontology that allows the inference of additional relations (dashed red lines).

Source: [CKG23]

From data to KG

- Need to integrate consistently different sources
- Metadata can be quite diverse
- Temporality, redundancies, missing or incorrect information

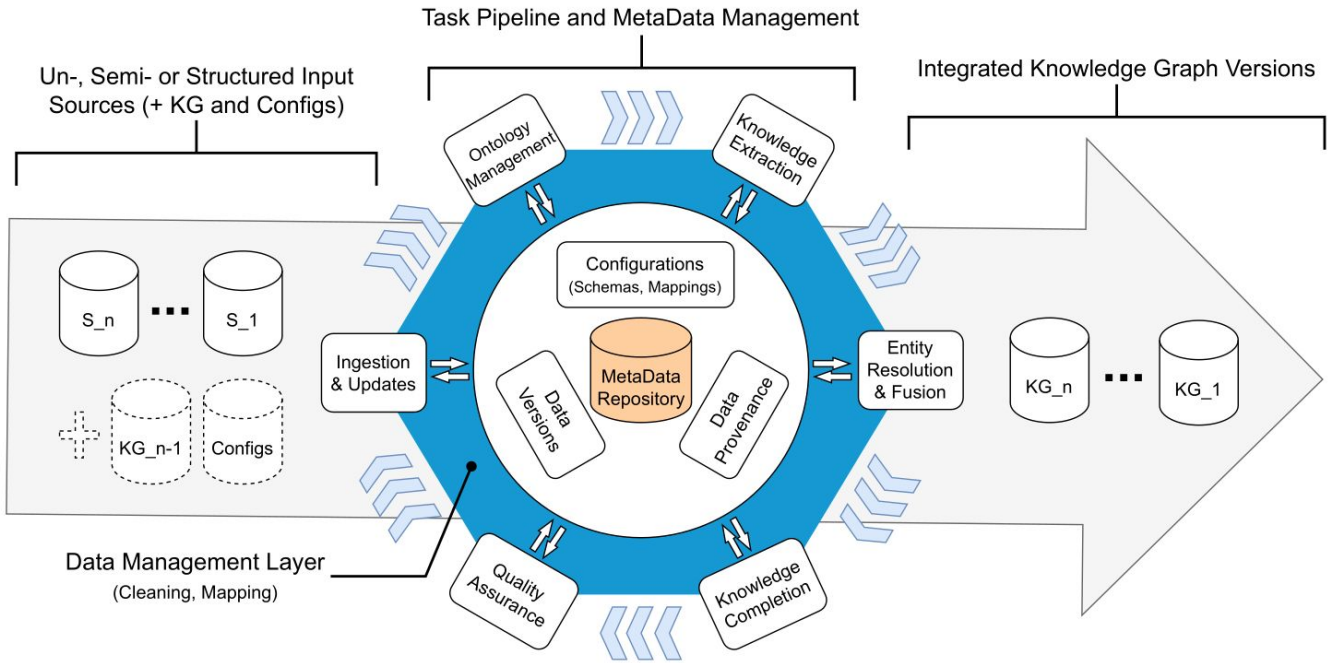


Fig. 2. Incremental Knowledge Graph Construction Pipeline

Source: [CKG23]

Overview of KG Construction Tasks and Approaches

- **Data Acquisition & Preprocessing**: Selection and transformation of relevant source data, initial data cleaning.
- **Metadata Management**: Acquisition and management of different kinds of metadata.
- **Ontology Management**: Creation and incremental evolution of a KG ontology.
- **Knowledge Extraction (KE)**: Derivation of structured information from unstructured or semi-structured data.
- **Entity Resolution (ER) and Fusion**: Identification and fusion of matching entities within the KG.
- **Quality Assurance (QA)**: Identification and repair of data quality problems in the KG.
- **Knowledge Completion**: Extension of a given KG, learning missing type information, predicting new relations, enhancing domain-specific data.

Wait for more...

topic	student	Monitor
What is a KG?	Rascoussier, Florian Guillaume Pierre	Prof. Algergawy
KG construction from text	Boudemagh, Houria-Chiraz	Prof. Algergawy
KG construction from tabular	Tayebi, Ilnaz	Prof. Algergawy
KG construction from both	Natesh Vijay, Chirag	Asha

Advanced KG-related concepts

This section is based on [KG21]

Inductive and deductive reasoning: What is that ?

Source: [Wikipedia]

Types of Reasoning

- 2 fundamental methods: Deductive and Inductive reasoning
- Different in process and results

Importance of Reasoning in KGs

- Essential for enhancing the utility of KGs.
- Enable predictions: missing links,
  - discovery of new knowledge, rules
  - improvement of the overall quality and completeness
- Facilitate complex queries and advanced analytics on the graph.

Deductive Reasoning

- **Top-down** logical flow: General to specific
- Conclusion is certain if premises are true
- Example:
  1. All men are mortal. (General premise)
  2. Socrates is a man. (Specific premise)
  3. Therefore, Socrates is mortal. (Specific conclusion)
- Used to test theories and hypotheses
- Used to “deduce” new knowledge

Modus ponens

Type	Deductive argument form Rule of inference
Field	Classical logic Propositional calculus
Statement	P implies Q. P is true. Therefore Q must also be true.
Symbolic statement	$P \to Q,\ P \vdash Q$

Source: [Wikipedia]

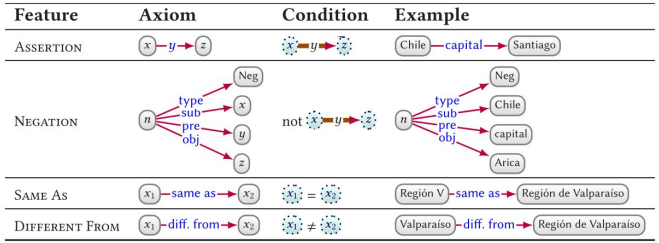
Deductive Reasoning in KGs

- Infer new facts based on existing facts and rules in the graph.
- Follows a logical process: if all premises are true, then the conclusion must also be true.
- Example: If we know that "Paris is a city" (fact) and "All cities are populated areas" (rule), we can deduce that "Paris is a populated area" (new fact).
- This process is also known as "knowledge graph completion" or "link prediction".

A range of methods

Table 2. Ontology Features for Individuals

Source: [KG21]



Ontologies

- **Definition:** An ontology is a formal representation of what terms mean within the scope in which they are used.
- Shared ontologies make KGs more interoperable.
- **Examples:** Web Ontology Language (OWL) by W3C, Open Biomedical Ontologies Format (OBOF).

Deduction, Inference, and Entailment

- **Deduction**: Process of deriving new data from what is already given and some implicit or explicit rules.
- **Inference**: Process of deriving or deducing new facts or knowledge from the existing data in the graph.
- **Entailment**: Deductive process where a relationship between statements or sets of statements where the truth of one statement or set necessarily implies the truth of another.

Inference Rules and Description Logics (DLs)

- **Inference Rules**: If-then (body-head) like statements with body and head being graph patterns.
- **Description Logics** (DLs): DLs come from First Order Logic (FOL), based on 3 types: Individuals, Classes, Properties
- DLs allow for making claims (known as axioms) about these elements.

Inductive Reasoning

- **Bottom-up** logical flow: Specific to general
- Conclusion is probable, based on truth of premises
- Example:
  - The sun has risen in the east every morning so far. (Specific observation)
  - Therefore, the sun will rise in the east tomorrow. (General conclusion)
- Used in the formation of hypotheses and theories

Source: [Wikipedia]

Inductive Reasoning in KGs

- Inductive reasoning in KGs is about learning new rules based on patterns observed in the data.
- It follows a probabilistic process: specific observations are generalized into rules, which are likely but not guaranteed to be true.
- Example: If we observe many instances of a relationship like "Person works at Company" and "Company is located in City", we might induce a rule like "Person lives in City".
- This process is often used in "rule mining" or "entity prediction" in KGs.

A range of possible approaches

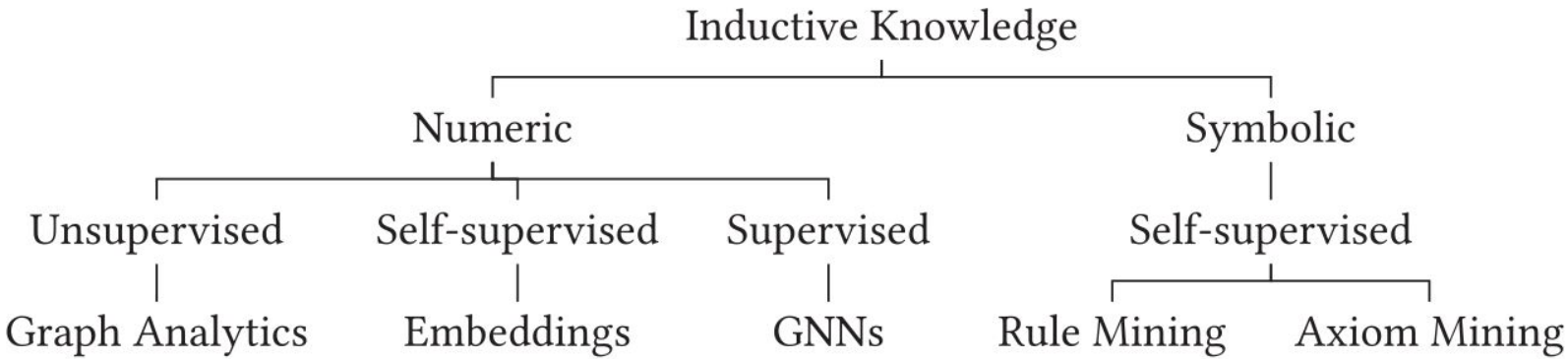


Fig. 14. Conceptual overview of popular inductive techniques for knowledge graphs.

Source: [KG21]

Critical Evaluation of the papers

This section is based on [KG21]

Contributions & Improvements [KG21]

Paper Overview

- Title: "Knowledge Graphs"
- Comprehensive exploration of Knowledge Graphs (KGs)
- Overview on structures, applications, and related concepts of KGs

Contributions

- Provides a detailed **overview** of KGs for readers with varying expertise
- **Meta-analysis** of 13 external papers and books
- Extended online version available with **concrete examples** on GitHub

Contributions & Improvements [KG21]

Improvements to the State-of-the-Art

- In-depth discussion of complex topics
- Meta-analysis on 13 papers
- Extensive bibliography for further reading, **examples** and extended online version

Main Results

- Detailed **overview of KG concepts** from basics to advanced
- Extensive overview of **inductive and deductive methods** for KG
- Discussion of **learning methods** for deductive and inductive reasoning: graph kernels, graph neural networks, label propagation, multimodal learning

Evaluation [KG21]

Quality of the Paper

- Comprehensive coverage and clear explanations of complex concepts
- Valuable resource for anyone interested in understanding KGs
- Not perfect, small mistakes
- could benefit from more practical examples or case studies

Readability of the Paper

- Well-structured, clearly presented content
- Complexity of topics may challenge readers new to the field
- Deep and insightful exploration of KGs related concepts

Overall Assessment

- Significant contribution to the field
- Thorough and detailed examination of KGs
- Useful for researchers and practitioners alike

Contributions & Improvements [CKG23]

Slide 1: Paper Overview

- Comprehensive review of current state of Knowledge Graph (KG) construction
- Discussion of main requirements, tasks, and challenges
- Comparison of existing KG construction pipelines and toolsets

Slide 2: Contributions

- Introduction and categorization of **general requirements** for incremental KG construction
- Overview of **main tasks in incremental KG construction** pipelines and proposed solution approaches
- Investigation and **comparison of existing construction efforts** for selected KGs and recent tools for KG construction

Contributions & Improvements [CKG23]

Improvements to the State-of-the-Art

- Emphasis on need for more open-source toolsets for KG development
- Highlighting importance of good data and metadata management
- Stress on high data quality and evaluation of complete KG construction pipelines

Main Results

- Identification of various challenges for improved incremental KG construction
- Discussion of engineering questions, task-specific problems, and support for incremental construction
- Addressing these challenges promises significant advances for future KG construction pipelines and reduced effort for creating and maintaining high-quality KGs

Evaluation [CKG23]

Quality of the Paper

- Comprehensive overview and comparison of the state of KG construction
- **Strength:** Identification of key challenges and requirements for KG construction
- **Weakness:** Lack of practical examples or case studies to illustrate the discussed concepts
- **WARN: Still a preprint ! No guaranty of peer review**

Readability of the Paper

- **Well structured:** The paper is well-structured with clear sections and subsections
- **Strength:** Use of clear and concise language makes it accessible to readers familiar with the topic
- **Weakness:** The paper could be difficult for readers without a background in KGs or related fields
- Some technical terms and concepts are not explained in detail, which could hinder understanding for some readers

Overall Assessment

- The paper provides a **valuable contribution** to the field of KG construction
- Successfully identifies the current state of KG construction and highlights areas for future research and improvement
- Valuable resource for researchers and practitioners in the field of KG construction

Contributions & Short Evaluation [KGKE22] (3.2 only)

Paper Overview

- Evolution of knowledge engineering since the 1980s.
- Four distinct periods with their consequences summarized

Quality / Readability

- **Clarity**: well-structured, presents a clear progression of ideas
- **Figures and Tables**: The paper includes figures to illustrate the evolving requirements for knowledge production processes

Overall Assessment

- **Relevance: The paper provides valuable insights into the history and evolution of knowledge engineering, making it relevant for anyone in the field.**
- **Contribution: The paper contributes to the understanding of the changing requirements and challenges in knowledge engineering over time.**
- **Implications: The paper raises important questions about the future of knowledge engineering, particularly in relation to knowledge graphs.**

Implementation example

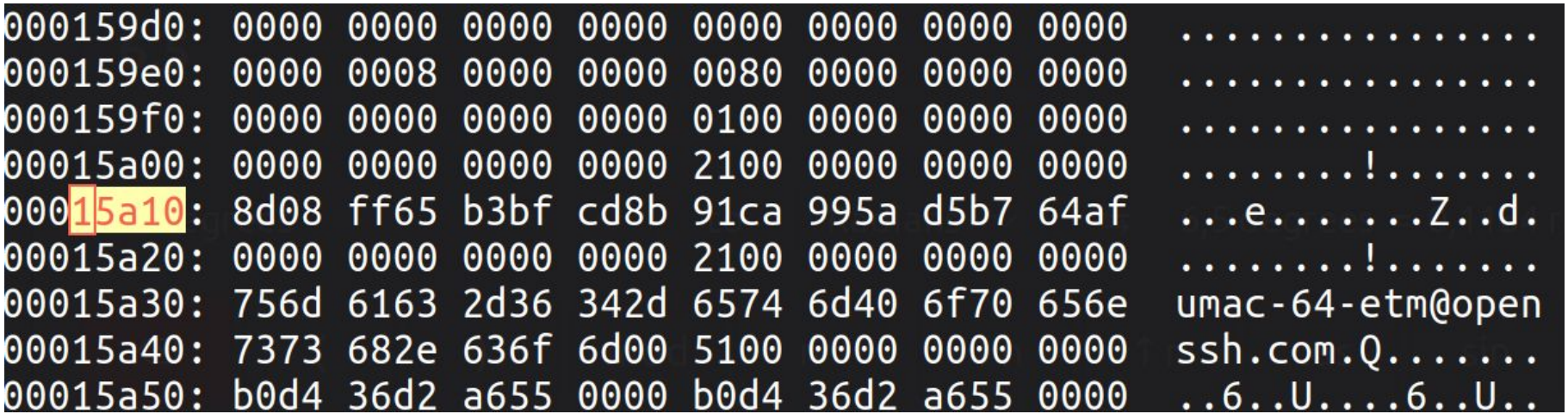
KG in my research project

Research project title: **Heap dump memory data modeling for embedding and security key detection learning**

This section is based on personal work

KG in real life

Research project title: **Heap dump memory data modeling for embedding and security key detection learning**



KG data sources

- Heap dump raw data
- JSON annotations
- Knowledge about memory representation

```
00000350:90957e968b550000b0957e968b550000..~..U....~..U..
00000360:d0957e968b55000000967e968b550000..~..U....~..U..
00000370:30967e968b55000080967e968b5500000.~..U....~..U..
00000380:a0967e968b550000c0967e968b550000..~..U....~..U..
00000390:0000000000000000210000000000000.........!.......
000003a0:5348454c4c3d2f62696e2f6261736800SHELL=/bin/bash.
000003b0:0000000000000000210000000000000.........!.......
000003c0:4c414e47554147453d656e5f55533a65LANGUAGE=en_US:e
000003d0:6e0000000000000021000000000000n.......!.......
000003e0:505744 3d2f726f6f740000 0000000000PWD=/root.....
000003f0:0000000000000000210000000000000.........!.......
00000400:4c4f474e414d 453d726f6f740000 0000LOGNAME=root....
00000410:0000000000000000210000000000000.........!.......
00000420:5844475f53455353494f4e5f54595045XDG_SESSION_TYPE
00000430:3d74747900000000210000000000000=tty....!.......
00000440:484f4d45 3d2f726f6f740000 00000000HOME=/root.....
00000450:0000000000000000210000000000000.........!.......
00000460:4c414e473d656e5f55532e5554462d38LANG=en_US.UTF-8
00000470:0000000000000000210000000000000.........!.......
00000480:4c435f5445524d494e414c3d69546572LC_TERMINAL=iTer
00000490:6d3200000000000041000000000000m2.....A.......
000004a0:5353485f434f4e4e454354494f4e3d31SSH_CONNECTION=1
000004b0:302e34322e302e322035383130302031 0.42.0.2581001
000004c0:39322e3136382e31322e32313320323292.168.12.21322
```

```
00000360:d06535d2a6550000006635d2a6550000.e5..U...f5..U..
00000370:306635d2a6550000806635d2a65500000f5..U...f5..U..
00000380:a06635d2a6550000c06635d2a6550000.f5..U...f5..U..
00000390:0000000000000000210000000000000.........!.......
000003a0:5348454c4c3d2f62696e2f6261736800SHELL=/bin/bash.
000003b0:0000000000000000210000000000000.........!.......
000003c0:4c414e47554147453d656e5f55533a65LANGUAGE=en_US:e
000003d0:6e0000000000000021000000000000n.......!.......
000003e0:5057443d2f726f6f740000000000 00PWD=/root.....
000003f0:0000000000000000210000000000000.........!.......
```

```
(base) [onyr@kenzael phdtrack_data]$ cat ./Training/Training/scp
/V_7_8_P1/16/1010-1644391327.json | json_pp
{
   "ENCRYPTION_KEY_1_NAME" : "aes128-ctr",
   "ENCRYPTION_KEY_1_NAME_ADDR" : "558b967f7620",
   "ENCRYPTION_KEY_2_NAME" : "aes128-ctr",
   "ENCRYPTION_KEY_2_NAME_ADDR" : "558b967fb160",
   "HEAP_START" : "558b967e9000",
   "KEY_A" : "119bd34f49d27bbbc0f9af400d4edc39",
   "KEY_A_ADDR" : "558b967fefe0",
   "KEY_A_LEN" : "16",
   "KEY_A_REAL_LEN" : "16",
   "KEY_B" : "8a77835eb2007a46a776ae0c183253b9",
   "KEY_B_ADDR" : "558b967f5ce0",
   "KEY_B_LEN" : "16",
   "KEY_B_REAL_LEN" : "16",
   "KEY_C" : "528f6dbd2907b3b4cfbd02fb32b852e7",
   "KEY_C_ADDR" : "558b967f51f0",
   "KEY_C_LEN" : "16",
   "KEY_C_REAL_LEN" : "16",
   "KEY_D" : "427f04149eed7029f031e58f3fde9844",
   "KEY_D_ADDR" : "558b967fb180",
   "KEY_D_LEN" : "16",
   "KEY_D_REAL_LEN" : "16",
   "KEY_E" : "17b6c799b5639ce5ea60c7f67cf6177f",
   "KEY_E_ADDR" : "558b967ff070",
   "KEY_E_LEN" : "16",
   "KEY_E_REAL_LEN" : "16",
   "KEY_F" : "fb75f5776184794ca92624ec6a36fd62",
   "KEY_F_ADDR" : "558b967f3d90",
   "KEY_F_LEN" : "16",
   "KEY_F_REAL_LEN" : "16",
   "NEWKEYS_1_ADDR" : "558b96800fd0",
   "NEWKEYS_2_ADDR" : "558b967fef10",
   "SESSION_STATE_ADDR" : "558b967f7f30",
   "SSH_PID" : "1010",
   "SSH_STRUCT_ADDR" : "558b967f6c20",
   "enc_KEY_OFFSET" : "0",
   "iv_ENCRYPTION_KEY_OFFSET" : "40",
   "iv_len_ENCRYPTION_KEY_OFFSET" : "24",
   "key_ENCRYPTION_KEY_OFFSET" : "32",
   "key_len_ENCRYPTION_KEY_OFFSET" : "20",
   "mac_KEY_OFFSET" : "48",
   "name_ENCRYPTION_KEY_OFFSET" : "0",
   "newkeys_OFFSET" : "344",
   "session_state_OFFSET" : "0"
}
```

Graph modelisation

Heterogeneous Directed Edge-labelled Graph

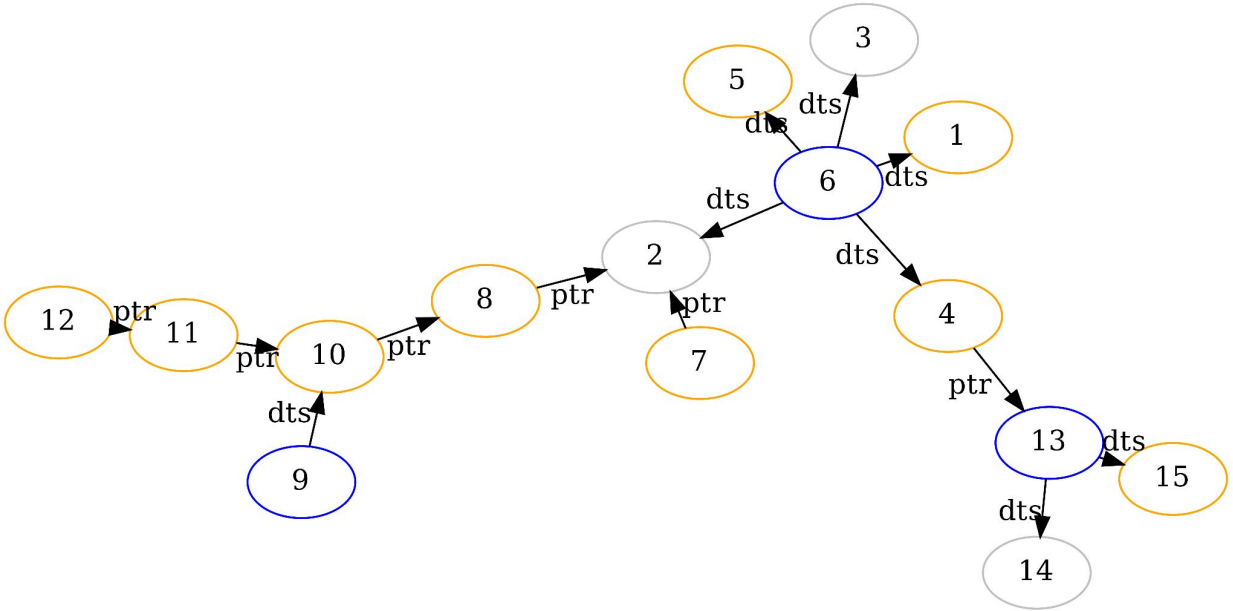
- Different types for each node
- Different types of edges

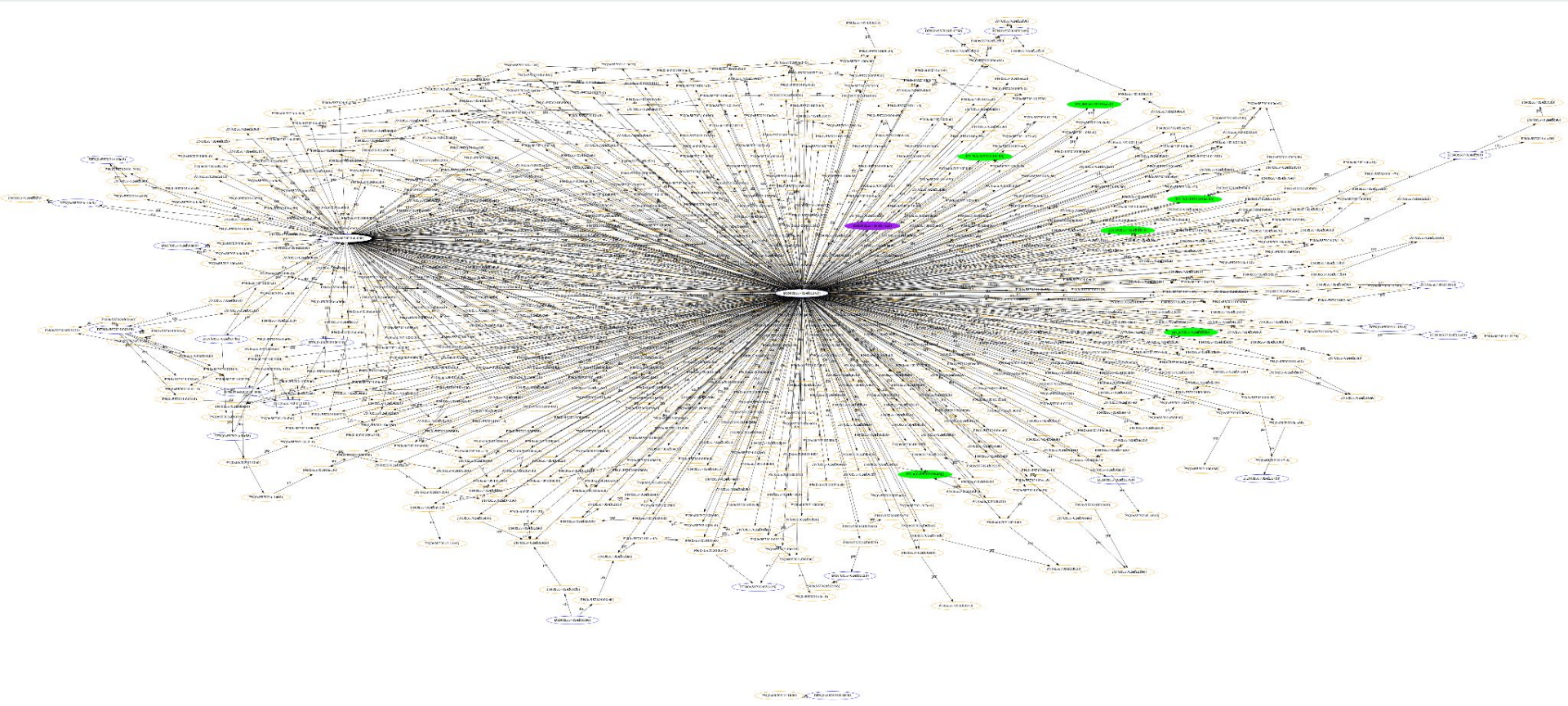
Nodes:

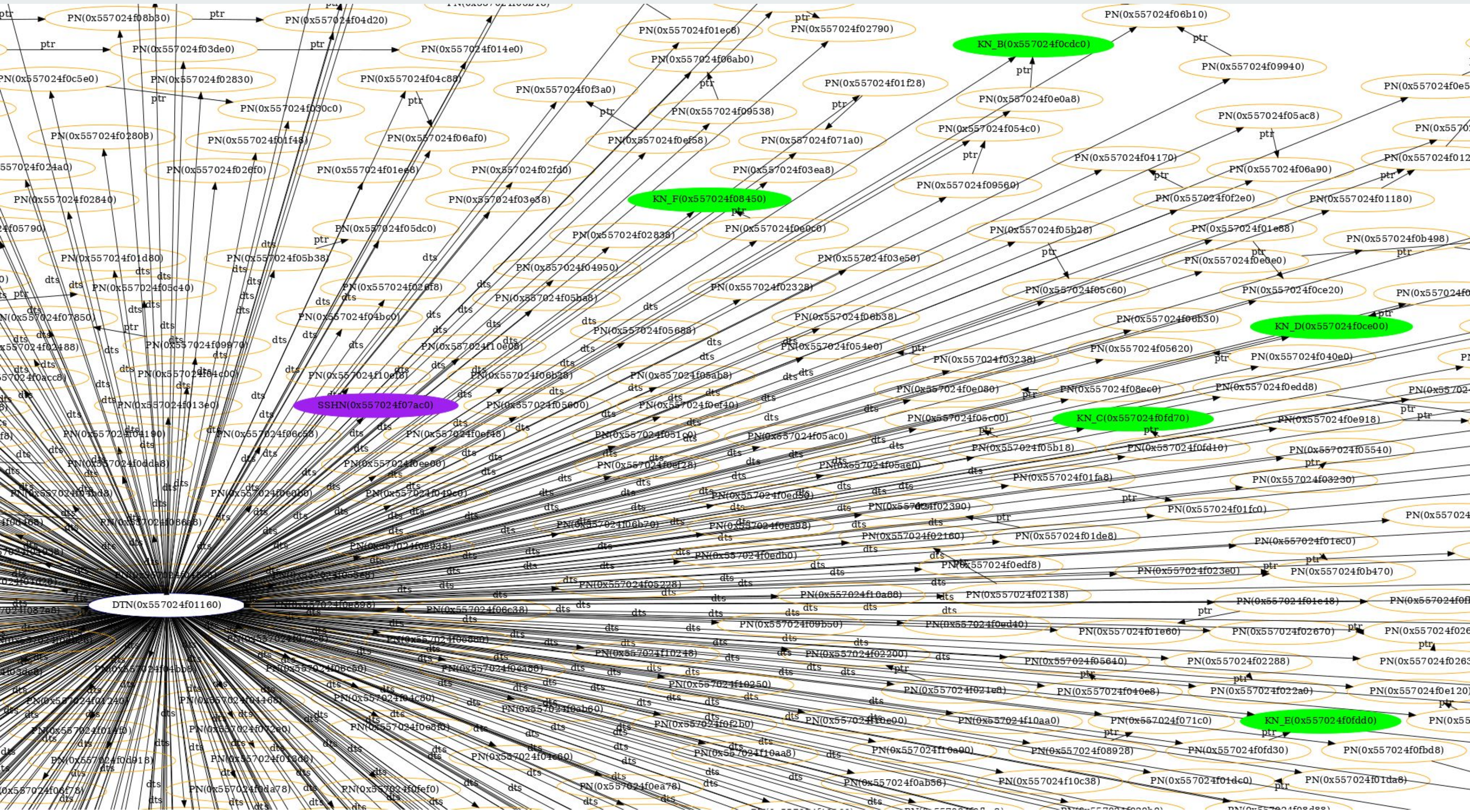
- DTN - pointer node
- VN - value node
- PN - pointer node
- KN - key node
- Other special nodes from annotations

Edges:

- ptr - pointer edge
- dts - data structure edge






DTN(0x557024f01160)
SSHN(0x557024f07ac0)
KN_B(0x557024f0cdc0)
KN_F(0x557024f08450)
KN_D(0x557024f0ce00)
KN_C(0x557024f0fd70)
KN_E(0x557024f0fdd0)
PN(0x557024f08b30)
PN(0x557024f04d20)
PN(0x557024f03de0)
PN(0x557024f014e0)
PN(0x557024f0c5e0)
PN(0x557024f02830)
PN(0x557024f030c0)
PN(0x557024f06b10)
PN(0x557024f09940)
PN(0x557024f02790)
ptr
dts

FT / Embedding

Feature Engineering (FT): process of using **domain knowledge** to create **features** (characteristics, properties, attributes) that enhance the performance of ML algorithms. Those features are often vectors

```
f_dtn_byte_size,f_position_in_dtn,f_dtn_pt
rs,f_dtn_vns,f_dtns_ancestor_1,f_ptrs_ance
stor_1,f_dtns_ancestor_2,f_ptrs_ancestor_2
,f_dtns_ancestor_3,f_ptrs_ancestor_3,f_dtn
s_ancestor_4,f_ptrs_ancestor_4,f_dtns_ance
stor_5,f_ptrs_ancestor_5,f_dtns_ancestor_6
,f_ptrs_ancestor_6,f_dtns_ancestor_7,f_ptr
s_ancestor_7,label
```

Embedding: Learned **lower-dimensional representation of data** that captures complex relationships, often used in GCNs to represent nodes based on their features and their position in the graph structure.

```
142682  48,1,0,5,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142683  192,1,3,20,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142684  47,2,0,4,1,1,1,1,0,0,0,0,0,0,0,0,0,0
142685  272,1,1,32,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0
142686  48,1,0,5,1,1,1,0,0,0,0,0,0,0,0,0,0,0,1
142687  96,1,3,8,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142688  544,1,1,66,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142689  32,1,0,3,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0
142690  48,1,0,5,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142691  48,5,0,5,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142692  64,1,0,7,1,2,2,0,0,0,0,0,0,0,0,0,0,0,1
142693  64,5,0,7,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0
142694  32,1,0,3,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0
142695  32,1,0,3,1,2,2,1,1,0,0,0,0,0,0,0,0,0,0
142696  95,8,3,7,1,2,2,1,1,0,0,0,0,0,0,0,0,0,0
```

Machine Learning

- Key prediction
- Binary classification problem
- Many models can be used:
  - MultinomialNB
  - BernoulliNB
  - Perceptron
  - SGDClassifier
  - PassiveAggressiveClassifier
  - MLPClassifier
  - ...
- Imbalance dataset

```
2023_05_31_15_54_30 - common_logger - INFO - 📋 [f: 152 / 154] Loading file /home/onyr/cod
e/phdtrack/phdtrack_project_3/src/mem_to_graph/data/samples_and_labels/Validation__chunck_
idx-109_samples.csv
2023_05_31_15_54_30 - results_logger - INFO - Removing 1 columns with only one unique valu
e: ['f_dtns_ancestor_1']
2023_05_31_15_54_30 - common_logger - INFO - 📋 [f: 153 / 154] Loading file /home/onyr/cod
e/phdtrack/phdtrack_project_3/src/mem_to_graph/data/samples_and_labels/Validation__chunck_
idx-118_samples.csv
2023_05_31_15_54_30 - results_logger - INFO - Removing 1 columns with only one unique valu
e: ['f_dtns_ancestor_1']
2023_05_31_15_54_30 - common_logger - INFO - Number of empty files: 24
2023_05_31_15_54_33 - results_logger - INFO - Precision: 0.0033172557660275164, Recall: 0.
6098213583064452, F1-score: 0.006598616921718383
2023_05_31_15_54_33 - results_logger - INFO - Time elapsed since the begining of pipeline
(PipelineNames.ML_SGD): 17.69077968597412 s
```

Code

- Initial testing in Python
- KG creation & embedding: Rust
- ML: Python

KG well supported on both languages

```python
import networkx as nx
```

```rust
use petgraph::graphmap::DiGraphMap;
```

```rust
/// This struct contains the graph data
/// linked to a given heap dump file.
pub struct GraphData {
    pub graph: DiGraphMap<u64, graph_structs::Edge>,
    pub addr_to_node: HashMap<u64, graph_structs::Node>,
    /// list of all the addresses of the nodes that are dtn
    pub dtn_addrs: Vec<u64>,
    /// list of the addresses of the nodes that are values (and potential keys)
    pub value_node_addrs: Vec<u64>,

    /// special nodes are the ones that are not values (and not keys)
    pub special_node_to_annotation: HashMap<u64, SpecialNodeAnnotation>,

    pub heap_dump_data: Option<HeapDumpData>, // Some because it is an optional field, for testing purposes
}
```

```python
class GraphEmbedding:
    graph_data: GraphData
    depth: int # hyperparameter, affect the length of the vectors

    # aliases
    graph: nx.DiGraph
    heap_dump_data: HeapDumpData
    params: ProgramParams
```

Conclusion

Source: [KG21]

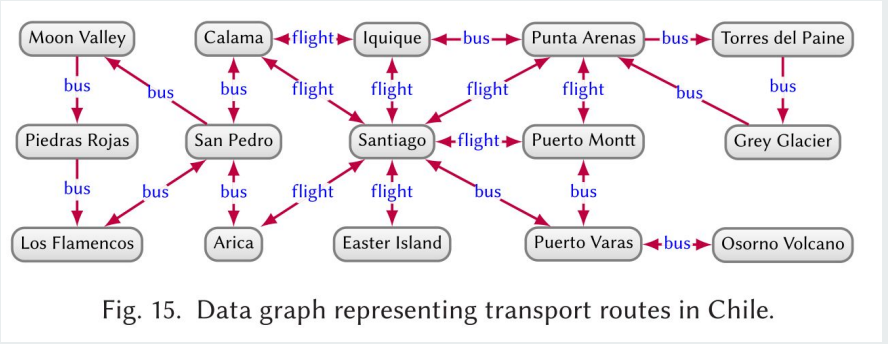


Fig. 15. Data graph representing transport routes in Chile.

Knowledge Graphs

- Powerful tool for modeling, storing, organizing, and accessing complex information
- Represents data as a network of interconnected nodes and edges
- Applications across many industries and domains

Sources & References

Main source:

[KG21] <ACM Compt.> Aidan Hogan u. a. „*Knowledge Graphs*“. In: ACM Comput. Surv. 54.4 (Juli 2021). issn: 0360-0300. doi: 10.1145/3447772. url: https://doi.org/10.1145/3447772.3

Additional sources:

[KGKE22] <Dagstuhl Reports> Paul Groth u. a. „*Knowledge Graphs and their Role in the Knowledge Engineering of the 21st Century*“. In: Dagstuhl Reports 12.9 (2022). Report from Dagstuhl Seminar 22372, S. 60–120. doi: 10.4230/DagRep.12.9.60. Specific usage: pp. 60-72, Subsection "3.2 A Brief History of Knowledge Engineering: A Practitioner’s Perspective", doi: 10.4230/DagRep.12.9.60.

[CKG23] <preprint> Marvin Hofer u. a. „*Construction of Knowledge Graphs: State and Challenges*“. In: arXiv preprint arXiv:2302.11509 (2023). url: https://doi.org/10.48550/arXiv.2302.11509.

Additional References

[TDKG16] L. Ehrlinger and W. Wöß, Towards a Definition of Knowledge Graphs., in: SEMANTiCS (Posters, Demos, SuCCESS), 2016.

[ISIFSCD73] E.W. Schneider, Course Modularization Applied: The Interface System and Its Implications For Sequence Control and Data Analysis. (1973).

NB: Images with no source are personal ones

Online resources for additional images

https://math.stackexchange.com/questions/1173328/eulers-solution-of-seven-bridges-of-k%C3%B6nigsberg-in-layman-terms

https://en.wikipedia.org/wiki/Seven_Bridges_of_K%C3%B6nigsberg

https://www.learner.org/courses/mathilluminated/units/11/textbook/02.php

https://www.storyofmathematics.com/16th_tartaglia.html

https://cs.mcgill.ca/~wlh/comp551/slides/25-gnns.pdf

https://en.wikipedia.org/wiki/Encyclop%C3%A9die

https://towardsdatascience.com/graph-theory-and-data-science-ec95fe2f31d8

https://www.stateofdigital.com/search-in-the-knowledge-graph-era/

https://www.aboutamazon.com/news/innovation-at-amazon/making-search-easier

Remark: **Wikipedia** has been used **only** as a source for illustration and images

Q&A

Thanks for your attention!

Challenges about KGs

A complex topic for handling complex data

Introduction to Challenges in KGs

- Knowledge Graphs (KGs) face numerous challenges in their construction and maintenance
- These challenges range from data volume to ontology incompatibilities and data quality

Handling Huge Data Streams

- KGs often need to integrate large volumes of data from various sources
- This requires efficient and scalable data processing pipelines
- Current KG construction often batch-like, unfit for incorporating new facts without full re-computation

Ontology Incompatibilities

- KGs rely on ontologies to provide a structured vocabulary for data
- However, different data sources may use different ontologies, leading to incompatibilities
- Aligning and mapping between different ontologies is a significant challenge

A hot scientific topic

Incorrectness and Incompleteness

- KGs can suffer from incorrect or incomplete data
- This can be due to errors in the original data sources or during the data integration process
- Ensuring data quality and completeness is a major challenge in KG construction

Current State of Research

- Research is ongoing to address these challenges and improve KG construction and maintenance
- Areas of focus include incremental KG construction, ontology alignment, and data quality assurance
- However, many challenges remain and further research is needed

Conclusion

- Despite these challenges, KGs remain a powerful tool for data integration and analysis
- Addressing these challenges will enhance the utility and reliability of KGs
- Continued research and development in this area is crucial.