

Semiparametric Reasoning over Structured Data

Rajarshi (Raj) Das

April 17, 2024

amazon | science



Who am I? 🙌

- ◆ Currently a researcher at Amazon
- ◆ PhD at UMass Amherst with Prof. Andrew McCallum
- ◆ Postdoc at University of Washington with Prof. Hanna Hajishirzi
- ◆ Thesis on “Semiparametric Contextual Reasoning Models for QA over KBs and Text”



Structured Databases

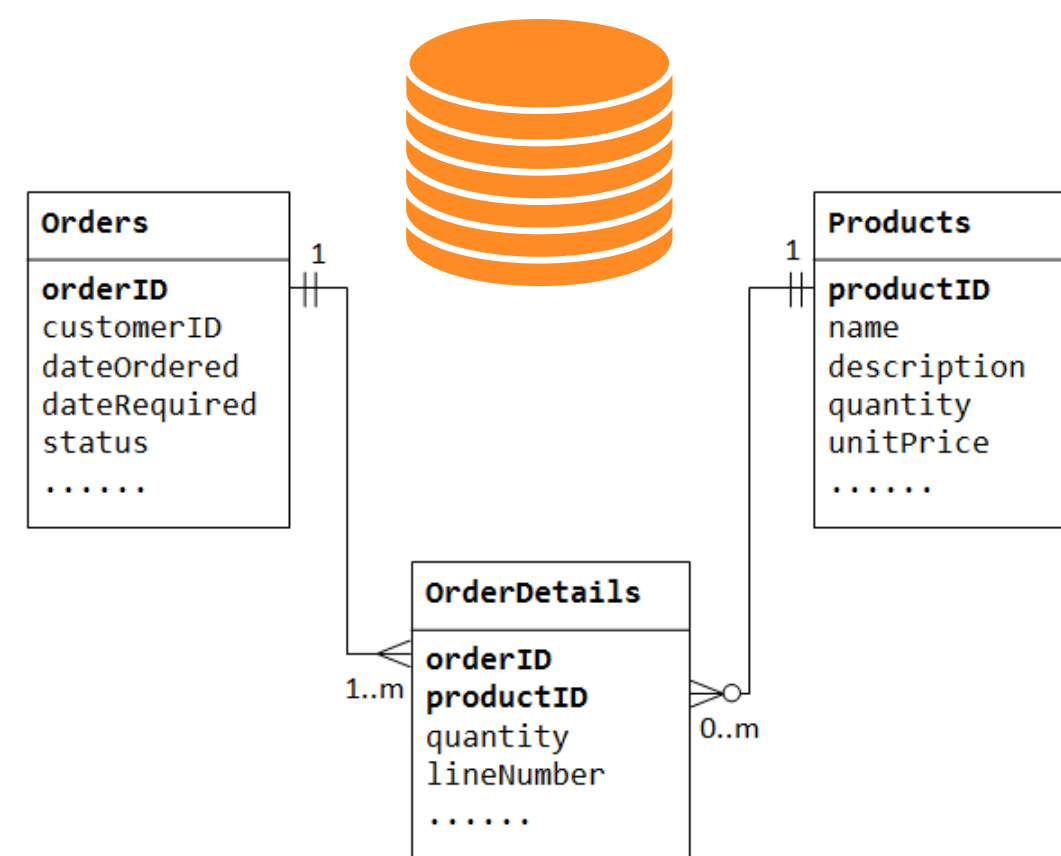
Structured Databases

◆ Structured data is ubiquitous.

Structured Databases

◆ Structured data is ubiquitous.

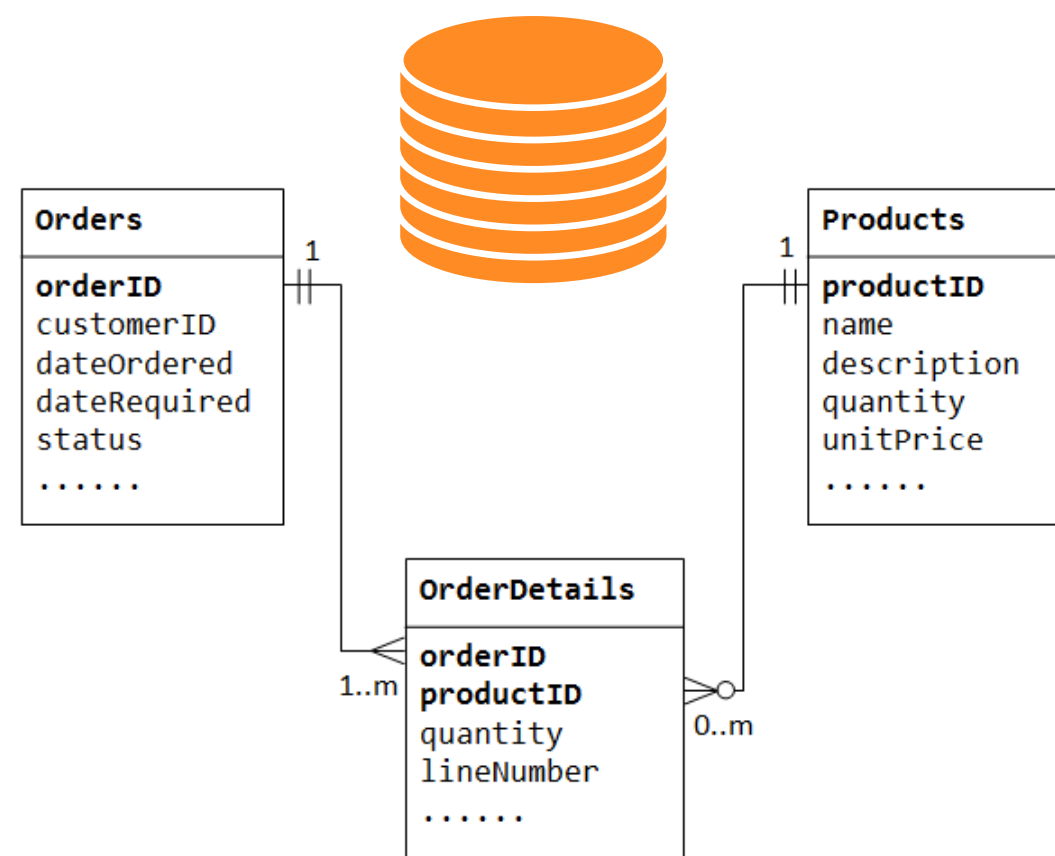
Relational Databases



Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



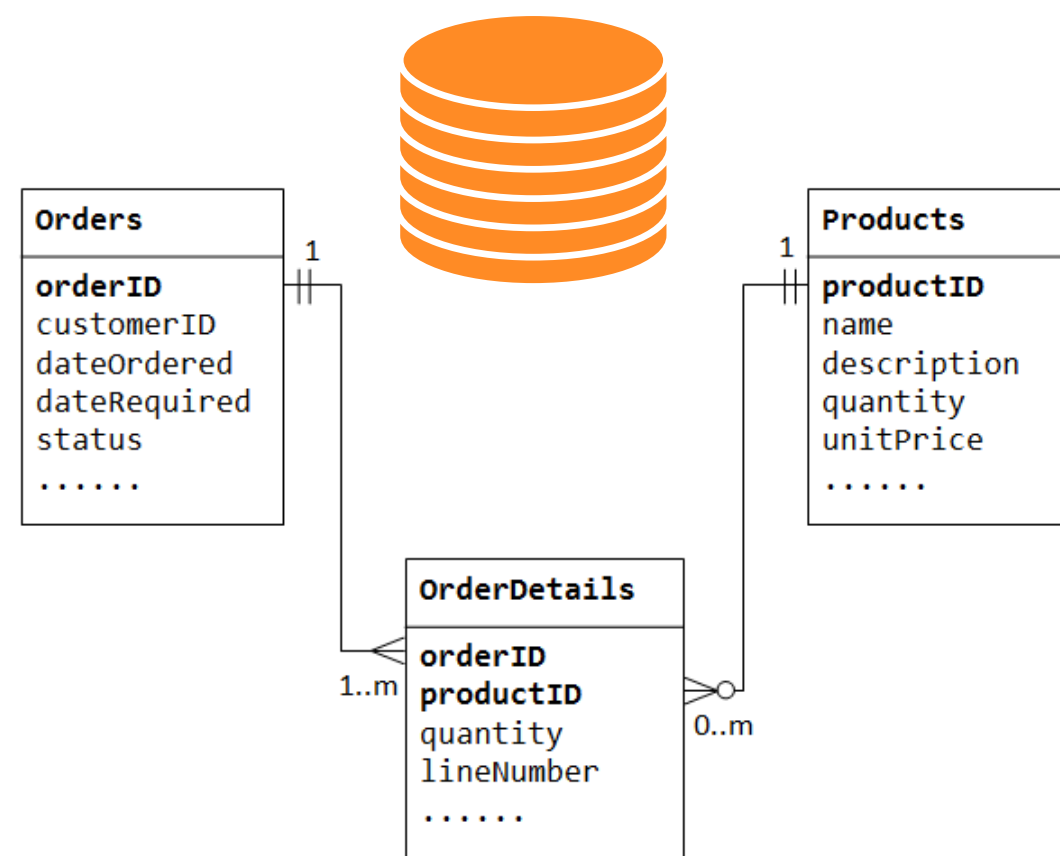
Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

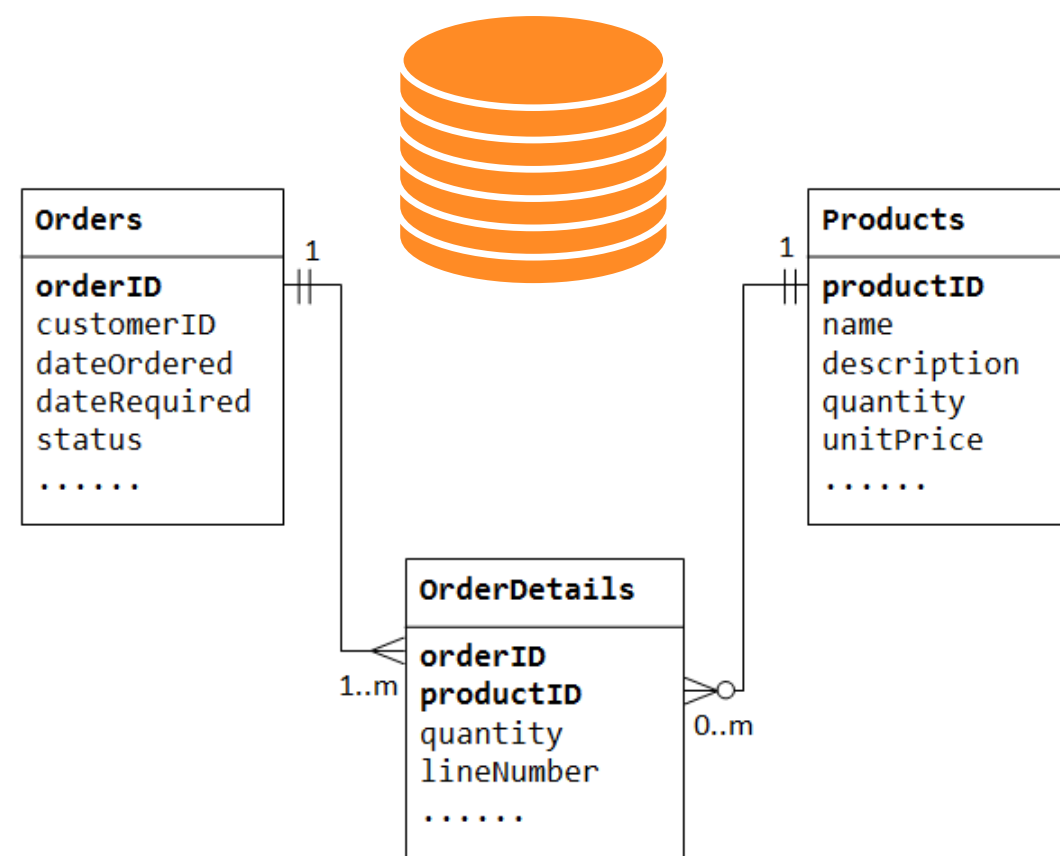


{ j s o n }

Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

Graphs

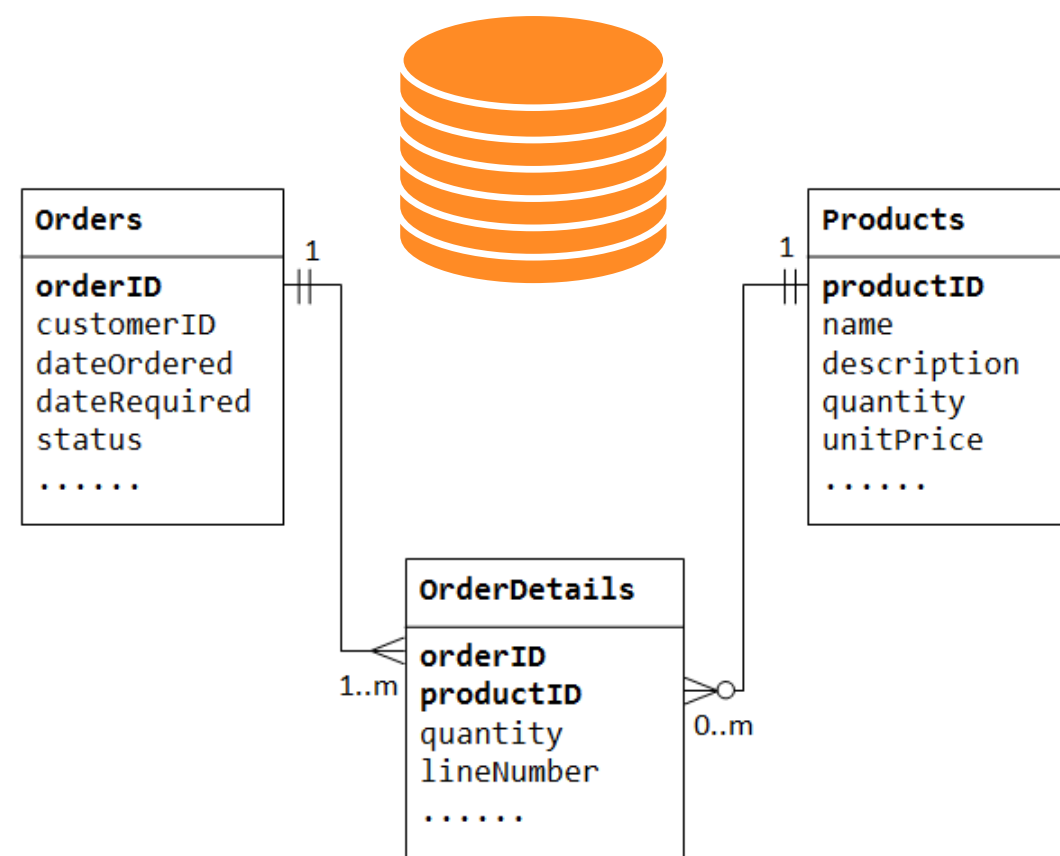


{ j s o n }

Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

Graphs

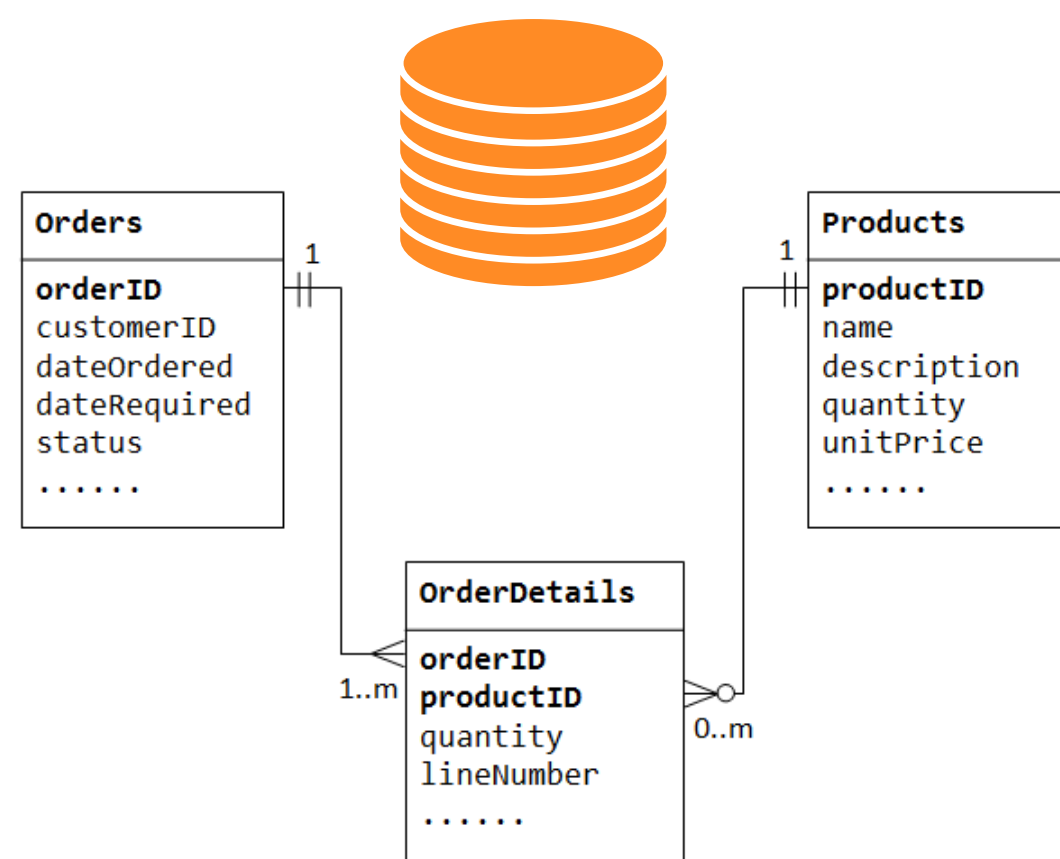


{ j s o n }

Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

Graphs

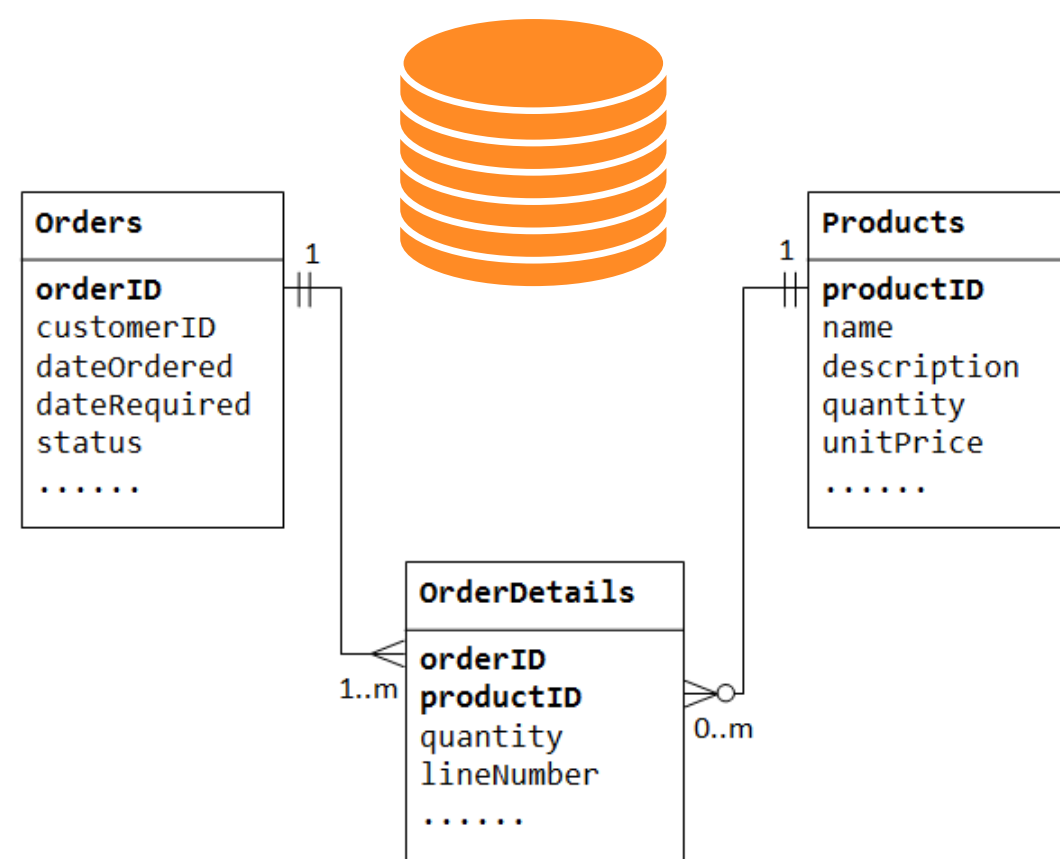


{ j s o n }

Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

Graphs



AI2 Allen Institute for AI

Mosaic Knowledge Graphs

Model: PersonX is a big deal

Knowledge Graph: Try: PersonX acts quickly, PersonX is a big deal

ATOMIC 2020 Knowledge Graph Browser

The dataset contains these relationships for 'PersonX is a big deal'

- Causes for PersonX
 - Because PersonX wanted: none
 - Before, PersonX needed:
 - to impress others
 - to learn how to do something
 - do something special
 - earn respect of people
 - none
- Attributes of PersonX
 - PersonX is seen as:
 - important
 - expert
 - famous
 - As a result, PersonX feels:
 - powerful
 - important
 - popular
 - capable
 - happy
 - As a result, PersonX wants:
 - to show off
 - to get out of the spotlight
 - to achieve great things
 - To be successful
 - To leave an impact
 - PersonX then:
 - is asked for their autograph
 - is cheered on by admirers
- Effects on others
 - As a result, others feel:
 - jealous
 - none
 - As a result, others want:
 - to spend time with PersonX
 - to praise PersonX

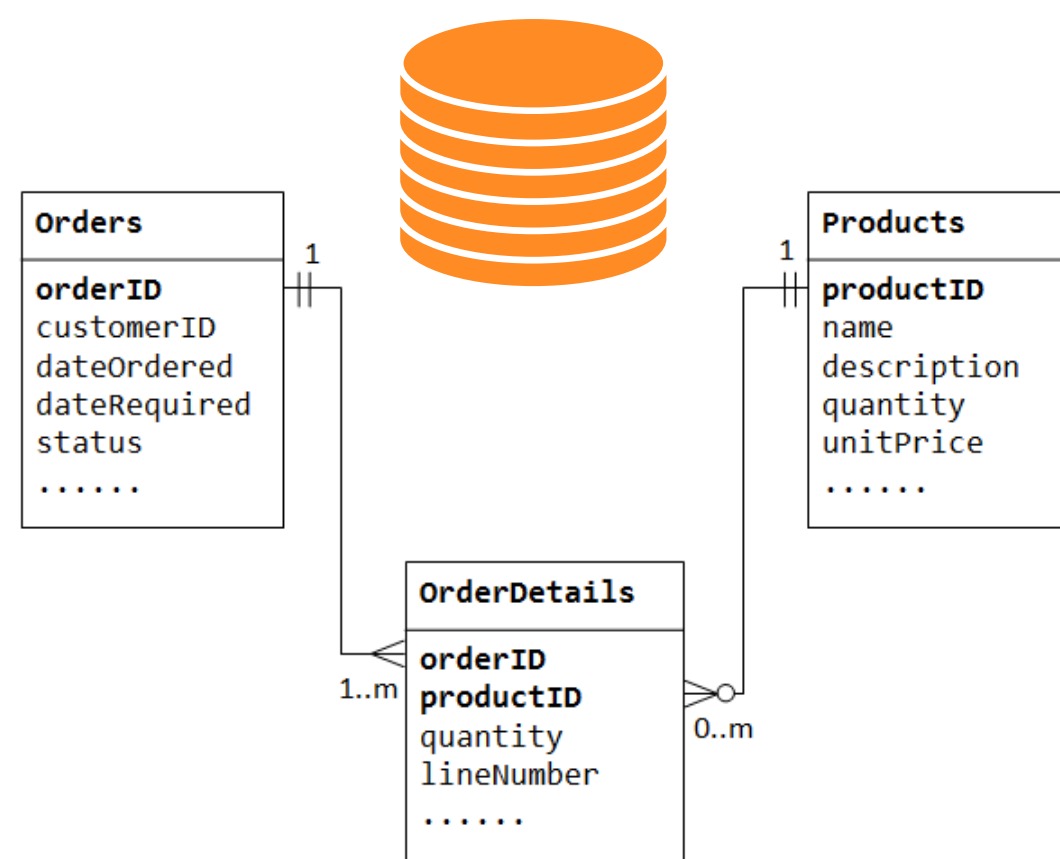


{ j s o n }

Structured Databases

◆ Structured data is ubiquitous.

Relational Databases



Web Tables

Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼
p5.48xlarge	\$98.32	192	2048 GiB	8 x 3840 GB SSD	3200 Gigabit
p4d.24xlarge	\$32.7726	96	1152 GiB	8 x 1000 SSD	400 Gigabit
p3.2xlarge	\$3.06	8	61 GiB	EBS Only	Up to 10 Gigabit
p3.8xlarge	\$12.24	32	244 GiB	EBS Only	10 Gigabit
p3.16xlarge	\$24.48	64	488 GiB	EBS Only	25 Gigabit
p2.xlarge	\$0.90	4	61 GiB	EBS Only	High
p2.8xlarge	\$7.20	32	488 GiB	EBS Only	10 Gigabit
p2.16xlarge	\$14.40	64	732 GiB	EBS Only	20 Gigabit
g6.xlarge	\$0.8048	4	16 GiB	1 x 250 GB NVMe SSD	Up to 10 Gigabit
g6.2xlarge	\$0.9776	8	32 GiB	1 x 450 GB NVMe SSD	Up to 10 Gigabit
g6.4xlarge	\$1.3232	16	64 GiB	1 x 600 GB NVMe SSD	Up to 25 Gigabit
g6.8xlarge	\$2.0144	32	128 GiB	2 x 450 GB NVMe SSD	25 Gigabit

Graphs



AI2 Allen Institute for AI

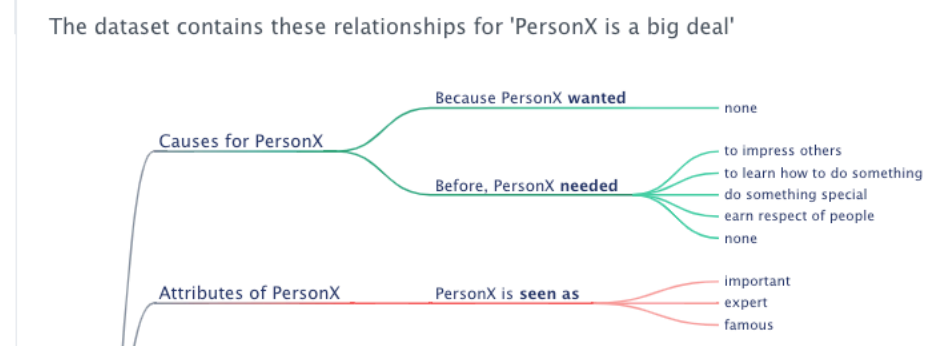
Mosaic Knowledge Graphs

Model:

Knowledge Graph:

ATOMIC 2020

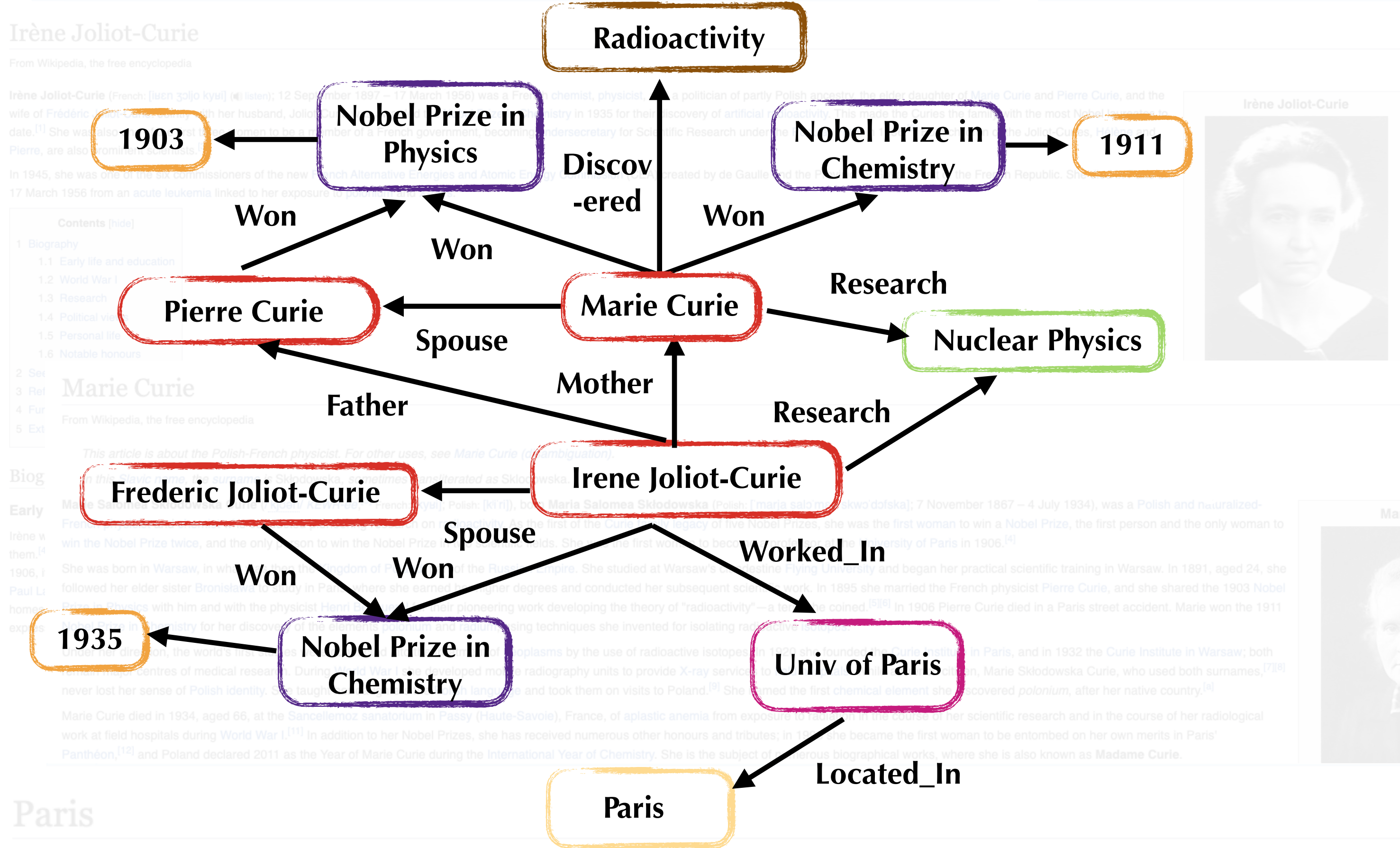
About



{ j s o n }



Knowledge Graphs



Irène Joliot-Curie

From Wikipedia, the free encyclopedia

Irène Joliot-Curie (French: [iʁɛn ʒoljo kʁiʁ] (ⓘ); 12 September 1897 – 3 August 1956) was a French chemist, physicist, and politician of partly Polish ancestry, the elder daughter of Marie Curie and Pierre Curie, and the wife of Frédéric Joliot-Curie. With her husband, Joliot-Curie, she was awarded the Nobel Prize in Chemistry in 1935 for the discovery of artificial radioactivity. This made the Curies the first and the only couple to win the most prestigious award in science, and the only woman to win the Nobel Prize twice. She was also the first woman to become a member of a French government, becoming undersecretary for Scientific Research under the Joliot-Curie government in 1945. She died of leukemia on 3 August 1956, aged 58.



Irène Joliot-Curie

- Contents (hide)
- 1 Biography
 - 1.1 Early life and education
 - 1.2 World War I
 - 1.3 Research
 - 1.4 Political views
 - 1.5 Personal life
 - 1.6 Notable honours
- 2 See also
- 3 References
- 4 Further reading
- 5 External links

Marie Curie

From Wikipedia, the free encyclopedia

This article is about the Polish-French physicist. For other uses, see Marie Curie (disambiguation).

Marie Salomea Skłodowska-Curie (Polish: [ˈmarja sɛlɔˈmɛa skɔdɔwˈska]; 7 November 1867 – 4 July 1934), was a Polish and naturalized-French physicist and chemist who worked on radioactivity. Her research, conducted jointly with her husband Pierre Curie, led to the discovery of the elements polonium and radium. She was the first woman to win a Nobel Prize, the first person and the only woman to win the Nobel Prize twice, and the only person to win the Nobel Prize in two different sciences. She was also the first woman to become a member of the Académie des Sciences.

She was born in Warsaw, in what was then part of the Russian Empire. She studied at Warsaw's University and began her practical scientific training in Warsaw. In 1891, aged 24, she followed her elder sister Bronisława to Paris, where she earned two degrees and conducted her subsequent scientific work. In 1895 she married the French physicist Pierre Curie, and she shared the 1903 Nobel Prize for Physics with him and with the physicist Henri Becquerel for their pioneering work developing the theory of "radioactivity"—a term she coined.^[28] In 1906 Pierre Curie died in a Paris street accident. Marie won the 1911 Nobel Prize for Chemistry for her discovery of polonium and radium, and for the refining techniques she invented for isolating radioactive isotopes.

Under her direction, the world's first artificial radioisotopes were produced by the use of radioactive isotopes in 1932. She founded the Curie Institute in Paris, and in 1932 the Curie Institute in Warsaw; both remain major centres of medical research. During World War I, Marie Curie developed mobile radiography units to provide X-ray services to the front lines. She never lost her sense of Polish identity. She taught in Polish language and took them on visits to Poland.^[9] She named the first chemical element she discovered *polonium*, after her native country.^[6]

Marie Curie died in 1934, aged 66, at the Sancellemoz sanatorium in Passy (Haute-Savoie), France, of aplastic anemia from exposure to radiation. Her death was the result of her scientific research and in the course of her radiological work at field hospitals during World War I.^[11] In addition to her Nobel Prizes, she has received numerous other honours and tributes; in 1995 she became the first woman to be entombed on her own merits in Paris' Panthéon,^[12] and Poland declared 2011 as the Year of Marie Curie during the International Year of Chemistry. She is the subject of numerous biographical works, where she is also known as Madame Curie.



Marie Curie

Radioactive decay

From Wikipedia, the free encyclopedia

For particle decay in a more general context, see Particle decay. For more information on hazards of various kinds of radioactive decay, see Nuclear safety. "Radioactive" and "Radioactivity" redirect here. For other uses, see Radioactive (disambiguation) and Radioactivity (disambiguation).

Radioactive decay (also known as **nuclear decay**, **radioactivity**, **radioactive disintegration** or **nuclear disintegration**) is the process by which an unstable nuclide spontaneously changes into a more stable nuclide. This process is considered **radioactive**. Three of the most common types of decay are alpha decay, beta decay, and gamma decay. Alpha decay is the only mechanism that is responsible for beta decay, while the other two are governed by the usual electromagnetic and strong force.

Radioactive decay is a stochastic (i.e. random) process at the level of single atoms. According to quantum theory, it is impossible to predict when a particular atom will decay. However, for a sufficiently large number of atoms, the decay rate can be expressed as a decay constant or as half-life.

- The decaying nuclide is transformed into a different nuclide, which is called the daughter nuclide. The process produces at least one ionizing particle and, depending on the type of decay, an antineutrino in a process that changes a neutron to a proton or a neutrino in a process that changes a proton to a neutron. The daughter nucleus may itself be radioactive and undergo further decay.
- Alpha decay
 - Beta decay
 - Gamma decay
 - In gamma decay

Nobel Prize

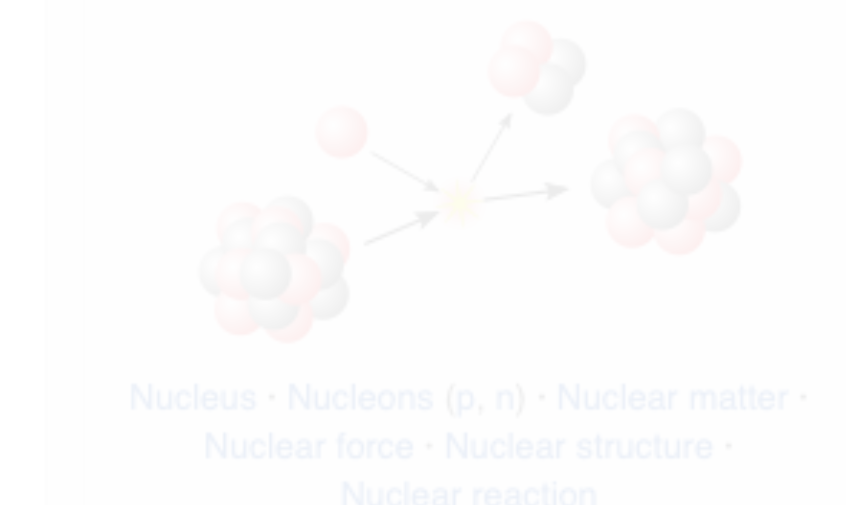


Awarded for Contributions that have conferred the greatest benefit to humankind in the areas of Physics, Chemistry, Physiology or Medicine, Literature, and Peace.

Country Sweden (all prizes except the Peace Prize)
Norway (Peace Prize only)

Presented by Royal Swedish Academy of Sciences (Physics and Chemistry)

Nuclear physics



Paris

From Wikipedia, the free encyclopedia

This article is about the capital of France. For other uses, see Paris (disambiguation).

Paris (French pronunciation: [paʁi] (ⓘ)) is the capital and most populous city of France, with an estimated population of 2,175,601 residents as of 2018, in an area of more than 105 square kilometres (41 square miles).^[4] Since the 17th century, Paris has been one of Europe's major centres of finance, diplomacy, commerce, fashion, gastronomy, science and arts. The City of Paris is the centre and seat of government of the Île-de-France, or Paris Region, which has an estimated population of 12,174,880, or about 18 percent of the population of France as of 2017.^[5] The Paris Region had a GDP of €709 billion (\$808 billion) in 2017.^[6] According to the Economist Intelligence Unit Worldwide Cost of Living Survey in 2018, Paris was the second most expensive city in the world, after Singapore and ahead of Zürich, Hong Kong, Oslo and Geneva.^[7] Another source ranked Paris as most expensive, on a par with Singapore and Hong Kong, in 2018.^{[8][9]}

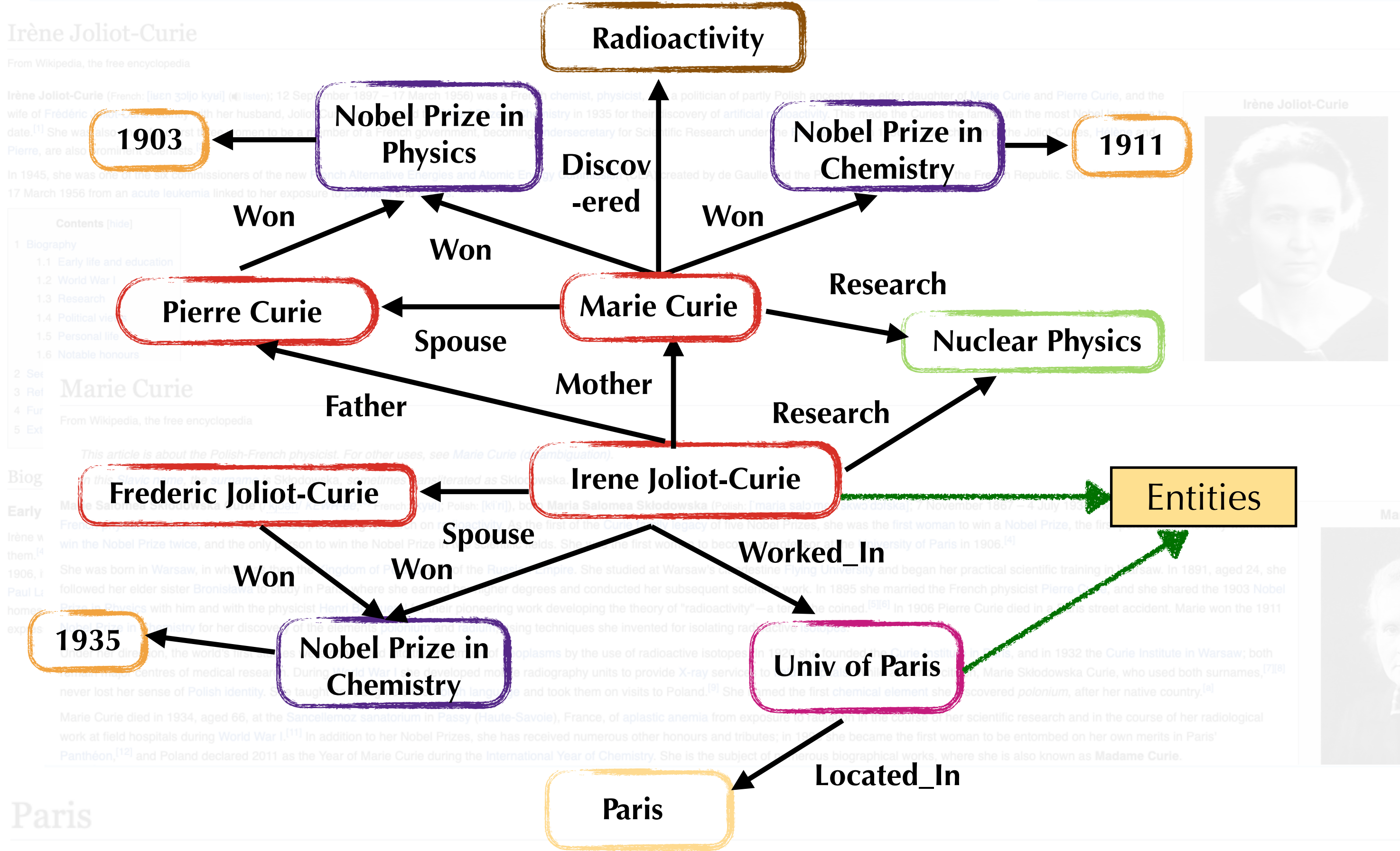
Paris is a major railway, highway and air-transport hub served by two international airports: Paris–Charles de Gaulle (the second busiest airport in Europe) and Paris–Orly.^{[10][11]} Opened in 1900, the city's subway system, the Paris Métro, serves 5.23 million passengers daily;^[12] it is the second busiest metro system in Europe after the Moscow Metro. Gare du Nord is the 24th busiest railway station in the world, but the first located outside

Paris

Capital city, department and commune



Knowledge Graphs



Radioactive decay

From Wikipedia, the free encyclopedia

For particle decay in a more general context, see Particle decay. For more information on hazards of various kinds of radiation, see Radiation. "Radioactive" and "Radioactivity" redirect here. For other uses, see Radioactive (disambiguation) and Radioactivity (disambiguation).

Radioactive decay (also known as nuclear decay, radioactivity, radioactive disintegration or nuclear disintegration) is a process by which an unstable nucleus loses energy by emitting ionizing radiation. A nucleus that is capable of undergoing this process is said to be radioactive. Three of the most common types of decay are alpha decay, beta decay, and gamma decay. Alpha decay is a process by which an alpha particle is emitted from a nucleus. Beta decay is a process by which a beta particle is emitted from a nucleus. Gamma decay is a process by which a gamma ray is emitted from a nucleus. The mechanism that is responsible for beta decay, while the other two are governed by the usual electromagnetic and strong forces.

Radioactive decay is a stochastic (i.e. random) process at the level of single atoms. According to quantum theory, it is impossible to predict when a particular atom will decay. However, for a sample containing a large number of atoms, the decay rate can be expressed as a decay constant or as half-life.

- Alpha decay
- Beta decay
- Gamma decay
- (i) beta-minus
- (ii) beta-plus
- In gamma decay

Nobel Prize

Awarded for Contributions that have conferred the greatest benefit to humankind in the areas of Physics, Chemistry, Physiology or Medicine, Literature, and Peace.

Country Sweden (all prizes except the Peace Prize)
Norway (Peace Prize only)

Presented by Royal Swedish Academy of Sciences (Physics and Chemistry)

isotope^(note 1)), and the process produces at least one daughter nucleus (or nuclei) with a different number of protons or neutrons (or both). When a nucleus decays, it emits ionizing radiation (alpha particles, beta particles, gamma rays, or neutrinos).

an antineutrino in a process that changes a neutron to a proton, or a neutrino in a process that changes a proton to a neutron. The daughter nucleus then may be stable or it may itself be radioactive.

Nuclear physics

Nucleus · Nucleons (p, n) · Nuclear matter · Nuclear force · Nuclear structure · Nuclear reaction

Models of the nucleus [show]

ion [show]

[show]

[show]

Coordinates: 48°51′24″N 2°21′08″E﻿ / ﻿48.85667°N 2.35222°E﻿ / 48.85667; 2.35222

s [show]

ses [show]

d [show]

s [show]

physics [show]

[show]

V·T·E

Paris

From Wikipedia, the free encyclopedia

This article is about the capital of France. For other uses, see Paris (disambiguation).

Paris (French pronunciation: [paʁi] (listen[ⓘ])) is the capital and most populous city of France, with an estimated population of 2,175,601 residents as of 2018, in an area of more than 105 square kilometres (41 square miles).^[4] Since the 17th century, Paris has been one of Europe's major centres of finance, diplomacy, commerce, fashion, gastronomy, science and arts. The City of Paris is the centre and seat of government of the Île-de-France, or Paris Region, which has an estimated population of 12,174,880, or about 18 percent of the population of France as of 2017.^[5] The Paris Region had a GDP of €709 billion (\$808 billion) in 2017.^[6] According to the Economist Intelligence Unit Worldwide Cost of Living Survey in 2018, Paris was the second most expensive city in the world, after Singapore and ahead of Zürich, Hong Kong, Oslo and Geneva.^[7] Another source ranked Paris as most expensive, on a par with Singapore and Hong Kong, in 2018.^{[8][9]}

Paris is a major railway, highway and air-transport hub served by two international airports: Paris–Charles de Gaulle (the second busiest airport in Europe) and Paris–Orly.^{[10][11]} Opened in 1900, the city's subway system, the Paris Métro, serves 5.23 million passengers daily;^[12] it is the second busiest metro system in Europe after the Moscow Metro. Gare du Nord is the 24th busiest railway station in the world, but the first located outside

Paris

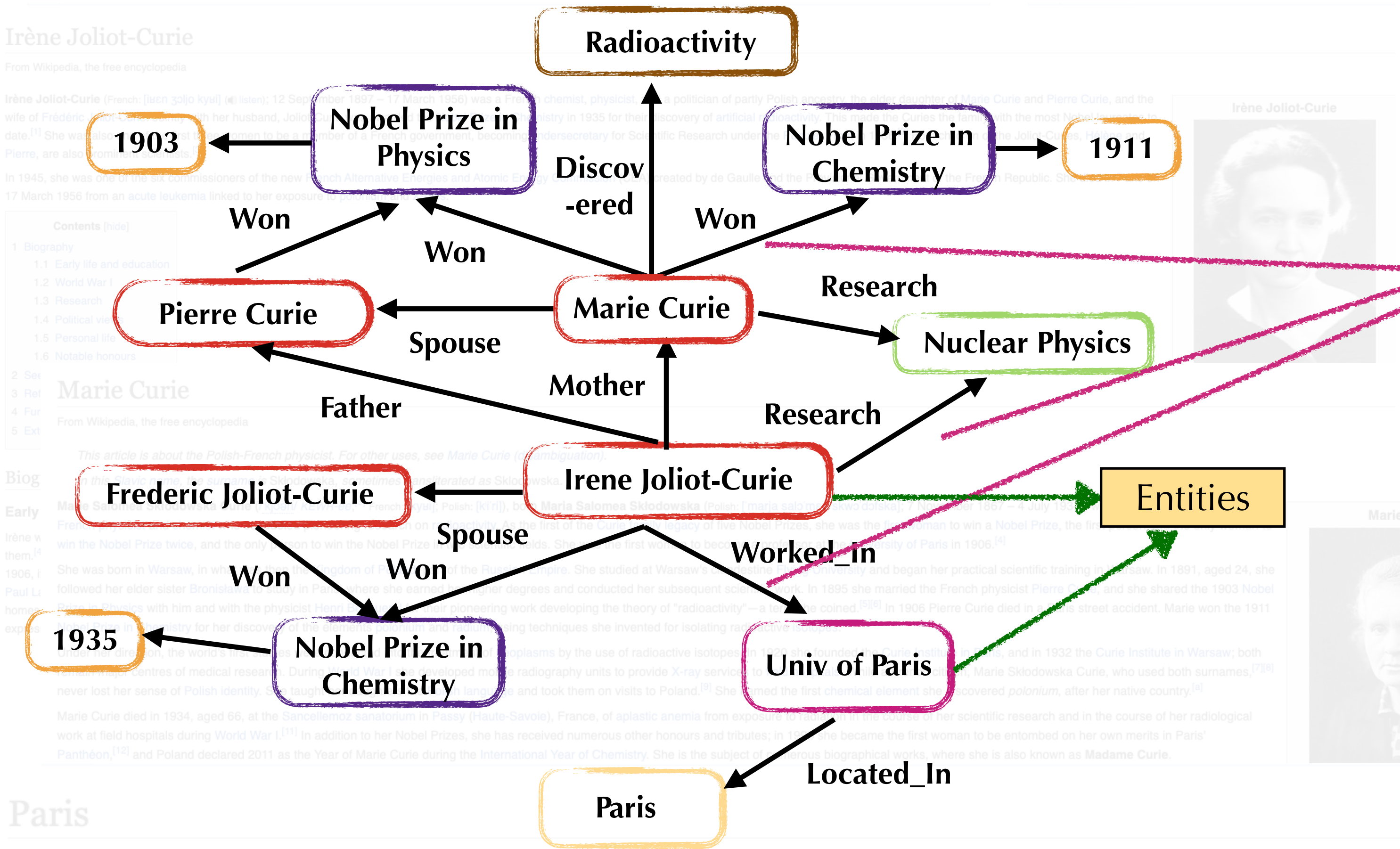
Capital city, department and commune

physics [show]

[show]

V·T·E

Knowledge Graphs



Radioactive decay

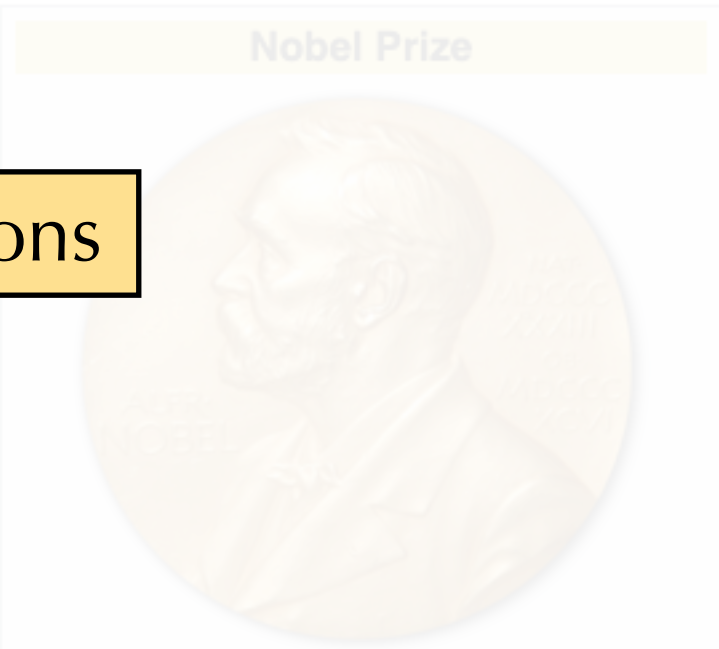
From Wikipedia, the free encyclopedia

For particle decay in a more general context, see Particle decay. For more information on hazards of various kinds of radioactive materials, see Radioactive hazards. "Radioactive" and "Radioactivity" redirect here. For other uses, see Radioactive (disambiguation) and Radioactivity (disambiguation).

Radioactive decay (also known as nuclear decay, radioactivity, radioactive disintegration or nuclear disintegration) is a process by which an unstable nucleus loses energy by emitting ionizing radiation. A nucleus that is capable of undergoing this process is considered radioactive. Three of the most common types of decay are alpha decay, beta decay, and gamma decay. Alpha decay is a process by which an alpha particle is emitted from a nucleus. Beta decay is a process by which a beta particle is emitted from a nucleus. Gamma decay is a process by which a gamma ray is emitted from a nucleus. The mechanism that is responsible for beta decay, while the other two are governed by the usual electromagnetic and strong forces.

Radioactive decay is a stochastic (i.e. random) process at the level of single atoms. According to quantum theory, it is impossible to predict when a particular atom will decay. However, for a sample of many atoms, the decay rate can be expressed as a decay constant or as half-life.

Nobel Prize



The decaying nucleus emits ionizing radiation. The decaying nucleus emits ionizing radiation. The decaying nucleus emits ionizing radiation.

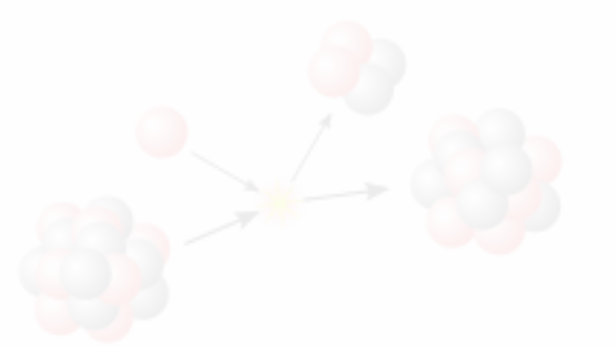
Relations

Awarded for Contributions that have conferred the greatest benefit to humankind in the areas of Physics, Chemistry, Physiology or Medicine, Literature, and Peace.

Country Sweden (all prizes except the Peace Prize)
Norway (Peace Prize only)

Presented by Royal Swedish Academy of Sciences (Physics and Chemistry)

Nuclear physics



Nucleus · Nucleons (p, n) · Nuclear matter · Nuclear force · Nuclear structure · Nuclear reaction

Paris

From Wikipedia, the free encyclopedia

This article is about the capital of France. For other uses, see Paris (disambiguation).

Paris (French pronunciation: [paʁi] (listen)) is the capital and most populous city of France, with an estimated population of 2,175,601 residents as of 2018, in an area of more than 105 square kilometres (41 square miles). Since the 17th century, Paris has been one of Europe's major centres of finance, diplomacy, commerce, fashion, gastronomy, science and arts. The City of Paris is the centre and seat of government of the Île-de-France, or Paris Region, which has an estimated population of 12,174,880, or about 18 percent of the population of France as of 2017. The Paris Region had a GDP of €709 billion (\$808 billion) in 2017. According to the Economist Intelligence Unit Worldwide Cost of Living Survey in 2018, Paris was the second most expensive city in the world, after Singapore and ahead of Zürich, Hong Kong, Oslo and Geneva. Another source ranked Paris as most expensive, on a par with Singapore and Hong Kong, in 2018.

Paris is a major railway, highway and air-transport hub served by two international airports: Paris–Charles de Gaulle (the second busiest airport in Europe) and Paris–Orly. Opened in 1900, the city's subway system, the Paris Métro, serves 5.23 million passengers daily; it is the second busiest metro system in Europe after the Moscow Metro. Gare du Nord is the 24th busiest railway station in the world, but the first located outside

Paris

Capital city, department and commune

Coordinates: 48°51′24″N 2°21′08″E

ion [show]

s [show]

ses [show]

d [show]

s [show]

physics [show]

V·T·E

Knowledge Graphs - Are they Still Relevant?

Knowledge Graphs - Are they Still Relevant?

◆ Yes!

Knowledge Graphs - Are they Still Relevant?

- ◆ Yes!

- ◆ Industry:

 - ◆ Question Answering powering chatbots and search engines

 - ◆ Product Knowledge Graphs

 - ◆ Cloud resources

Knowledge Graphs - Are they Still Relevant?

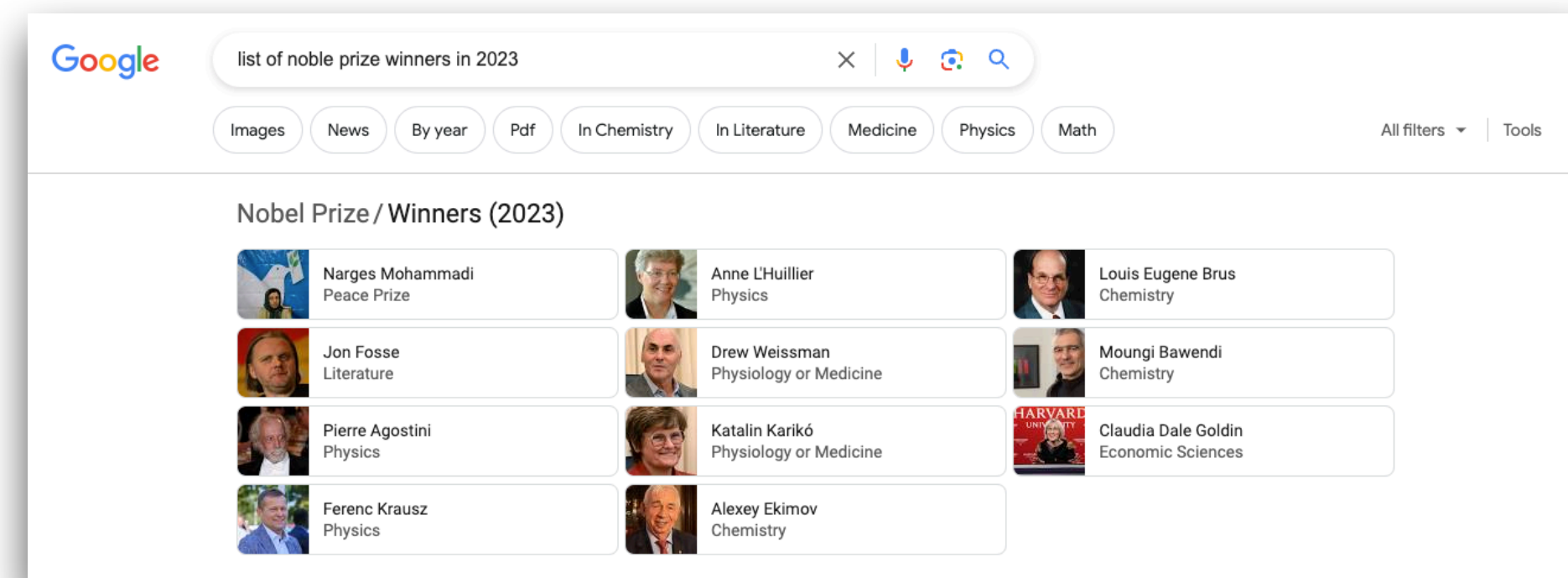
◆ Yes!

◆ Industry:

◆ Question Answering powering chatbots and search engines

◆ Product Knowledge Graphs

◆ Cloud resources



Knowledge Graphs - Are they Still Relevant?

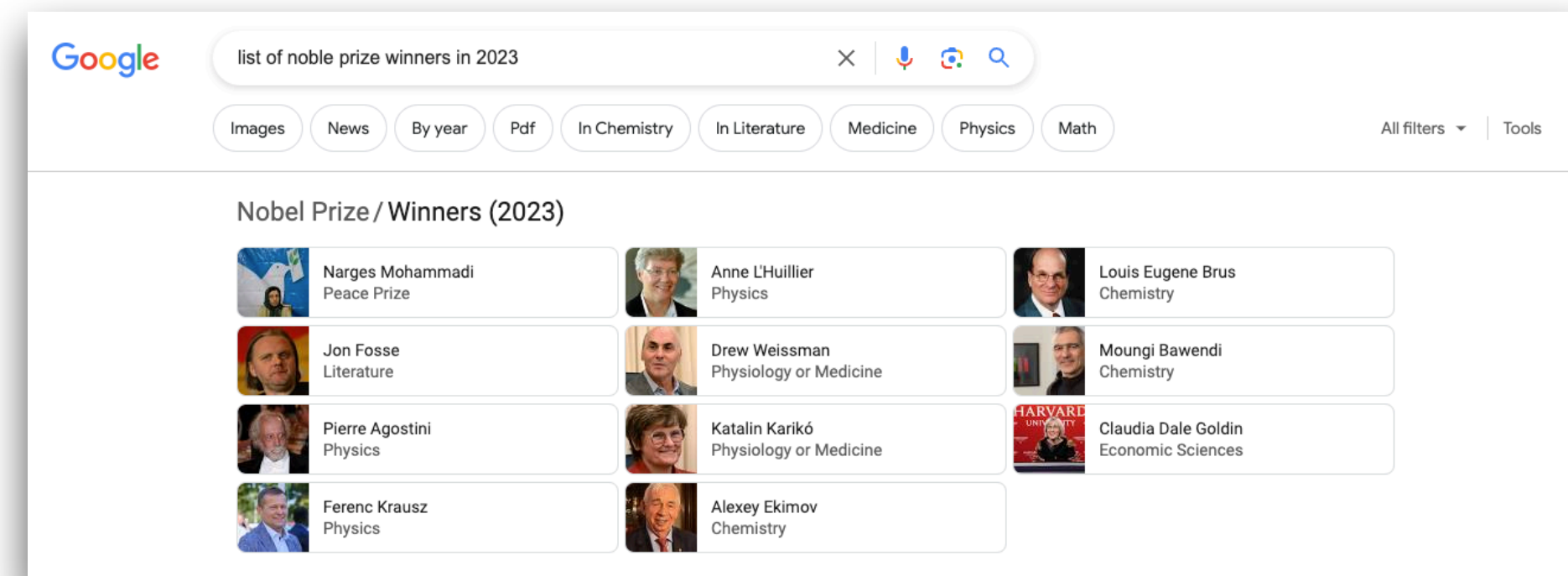
◆ Yes!

◆ Industry:

◆ Question Answering powering chatbots and search engines

◆ Product Knowledge Graphs

◆ Cloud resources



Knowledge Graphs - Are they Still Relevant?

◆ Yes!

◆ Industry:

◆ Question Answering powering chatbots and search engines

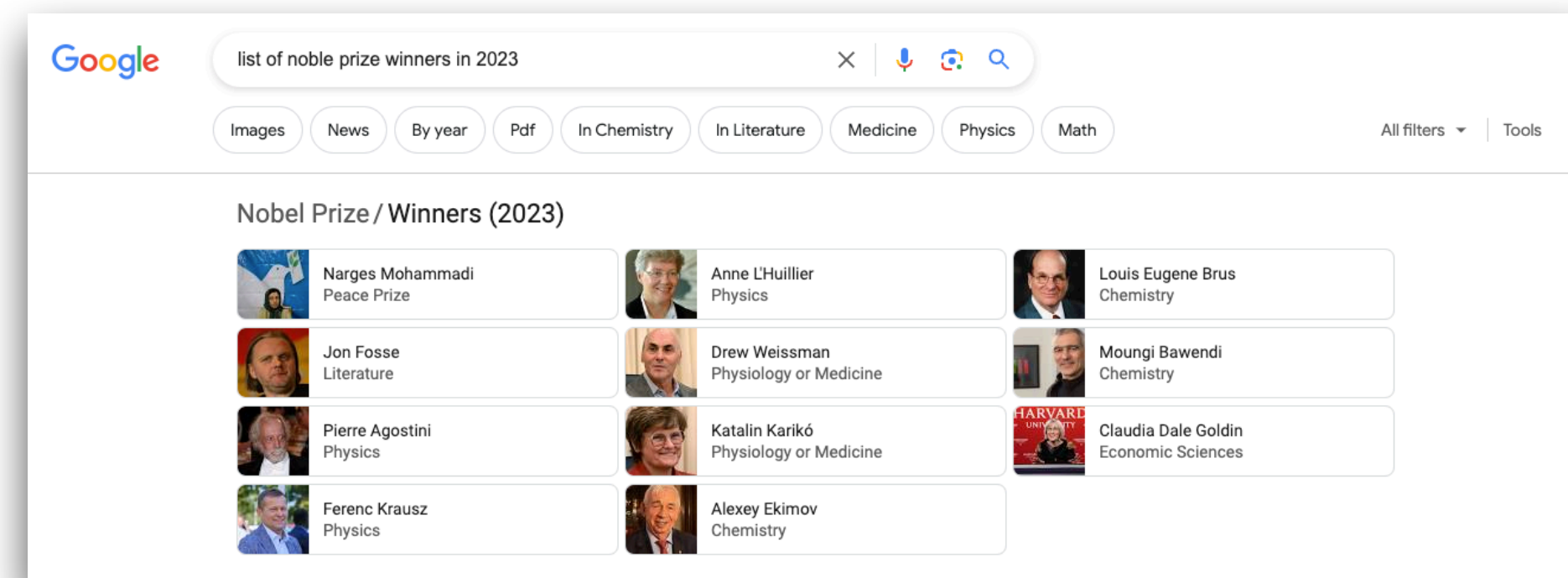
◆ Product Knowledge Graphs

◆ Cloud resources

◆ Specialized domains:

◆ Drug Discovery

◆ Material Science



5



Querying KGs

Querying KGs

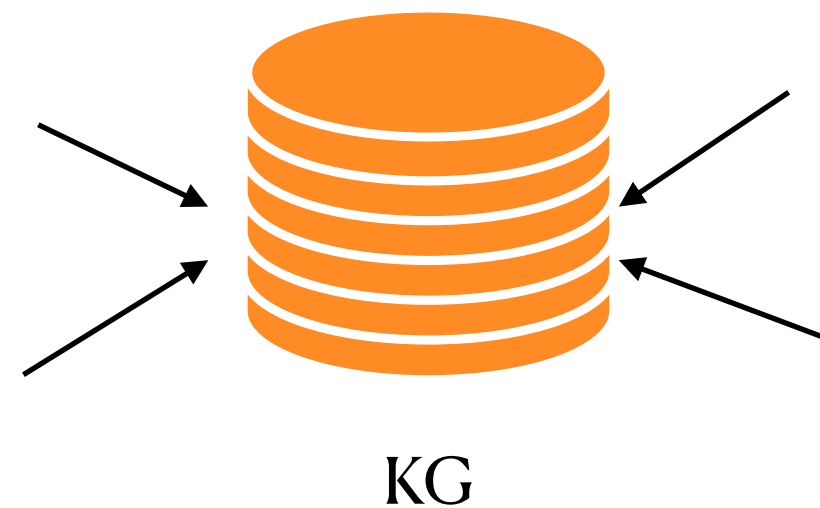
◆ How do we get information from KGs?

Querying KGs

◆ How do we get information from KGs?

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {
SELECT DISTINCT ?x0 WHERE {
?x0 :type.object.type :opera.opera_designer_role .
?x1 :type.object.type :opera.opera_designer_gig .
VALUES ?x2 { :m.0pm2fgf }
?x1 :opera.opera_designer_gig.design_role ?x0 .
?x2 :opera.opera_production.designers ?x1 .
FILTER ( ?x0 != ?x1 && ?x0 != ?x2 && ?x1 != ?x2 )}}
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {
SELECT DISTINCT ?x0 WHERE {
?x0 :type.object.type :measurement_unit.measurement_system .
VALUES ?x1 { :m.02sj5gj }
?x0 :measurement_unit.measurement_system.radiance_units ?x1 .
FILTER ( ?x0 != ?x1 )}}
```



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {\nSELECT DISTINCT ?x0
WHERE {
?x0 :type.object.type :rail.rail_network .
VALUES ?x1 { :m.03qcvdj }
?x0 :rail.rail_network.railways ?x1 .
FILTER ( ?x0 != ?x1 )\n}\n}
```

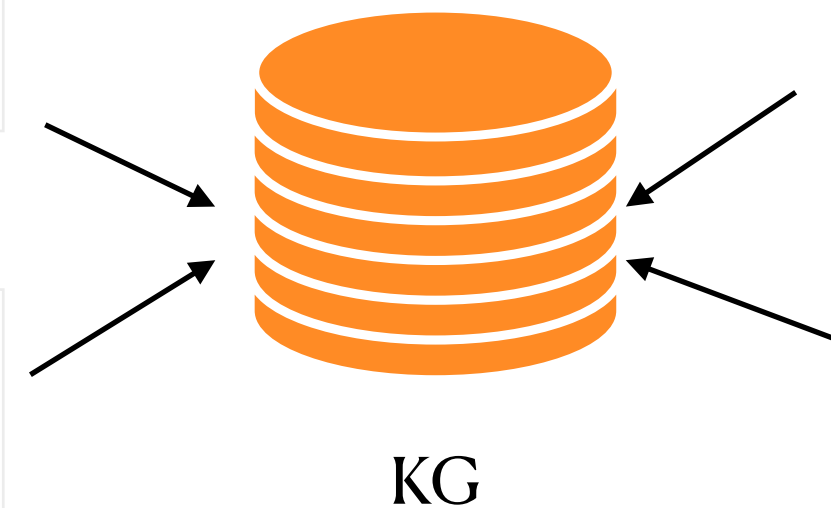
```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (COUNT(?x0) AS ?value) WHERE {\nSELECT DISTINCT ?
x0 WHERE {
?x0 :type.object.type :cvg.computer_videogame . \nVALUES
?x1 { :m.02hptvh }
?x1 :cvg.cvg_designer.games_designed ?x0 . FILTER ( ?
x0 != ?x1 )\n}\n}
```

Querying KGs

◆ How do we get information from KGs?

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {
SELECT DISTINCT ?x0 WHERE {
?x0 :type.object.type :opera.opera_designer_role .
?x1 :type.object.type :opera.opera_designer_gig .
VALUES ?x2 { :m.0pm2fgf }
?x1 :opera.opera_designer_gig.design_role ?x0 .
?x2 :opera.opera_production.designers ?x1 .
FILTER ( ?x0 != ?x1 && ?x0 != ?x2 && ?x1 != ?x2 )}}
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {
SELECT DISTINCT ?x0 WHERE {
?x0 :type.object.type :measurement_unit.measurement_system .
VALUES ?x1 { :m.02sj5gj }
?x0 :measurement_unit.measurement_system.radiance_units ?x1 .
FILTER ( ?x0 != ?x1 )}}
```



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {\nSELECT DISTINCT ?x0
WHERE {
?x0 :type.object.type :rail.rail_network .
VALUES ?x1 { :m.03qcvdj }
?x0 :rail.rail_network.railways ?x1 .
FILTER ( ?x0 != ?x1 )\n}\n}
```

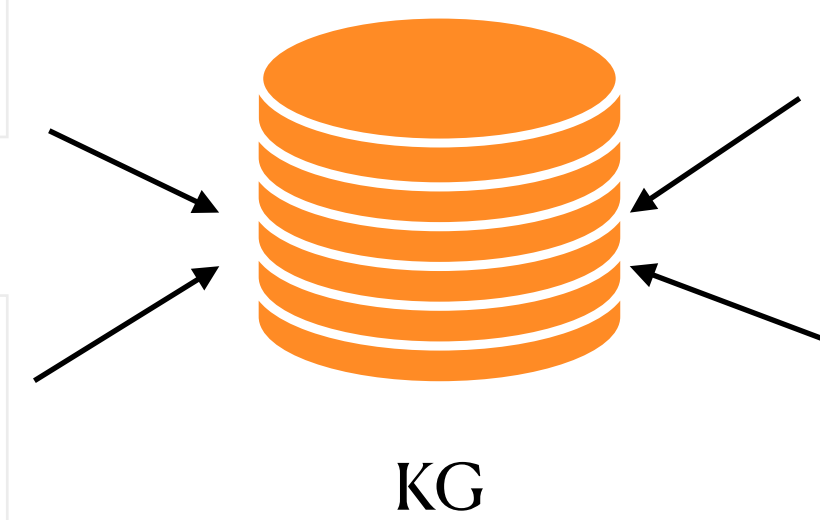
```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (COUNT(?x0) AS ?value) WHERE {\nSELECT DISTINCT ?
x0 WHERE {
?x0 :type.object.type :cvg.computer_videogame . \nVALUES
?x1 { :m.02hptvh }
?x1 :cvg.cvg_designer.games_designed ?x0 . FILTER ( ?
x0 != ?x1 )\n}\n}
```


Querying KGs

◆ How do we get information from KGs?

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {
What is the role of the designers in the opera
production The Telephone / The Medium?
?x0 :type.object.type :opera.opera_designer_gig .
VALUES ?x2 { :m.0pm2fgf }
?x1 :opera.opera_designer_gig.design_role ?x0 .
?x2 :opera.opera_production.designers ?x1 .
FILTER ( ?x0 != ?x1 && ?x0 != ?x2 && ?x1 != ?x2 )}}
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
Which measurement system has the radiance unit watt
per square metre per steradian?
SELECT DISTINCT ?x0 WHERE {
?x0 :type.object.type :measurement_unit.measurement_system .
VALUES ?x1 { :m.02sj5gj }
?x0 :measurement_unit.measurement_system.radiance_units ?x1 .
FILTER ( ?x0 != ?x1 )}}
```



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {\nSELECT DISTINCT ?x0
WHERE {
?x0 :type.object.type :rail.rail_network .
VALUES ?x1 { :m.03cwwdi }
?x0 :rail.rail_network.line_name ?x1 .
FILTER ( ?x0 != ?x1 )\n}\n}
```

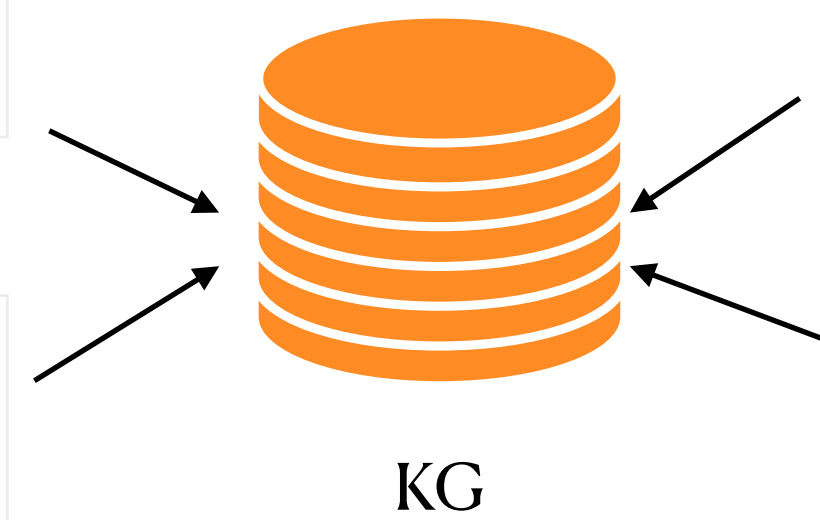
```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
How many video games has randy condon designed?
SELECT (COUNT(?x0) AS ?value) WHERE {\nSELECT DISTINCT ?
x0 WHERE {
?x0 :type.object.type :cvg.computer_videogame . \nVALUES
?x1 { :m.02hptvh }
?x1 :cvg.cvg_designer.games_designed ?x0 . FILTER ( ?
x0 != ?x1 )\n}\n}
```

Querying KGs

◆ How do we get information from KGs?

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {
What is the role of the designers in the opera
production The Telephone / The Medium?
?x0 :type.object.type :opera.opera_designer_gig .
VALUES ?x2 { :m.0pm2fgf }
?x1 :opera.opera_designer_gig.design_role ?x0 .
?x2 :opera.opera_production.designers ?x1 .
FILTER ( ?x0 != ?x1 && ?x0 != ?x2 && ?x1 != ?x2 )}}
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
Which measurement system has the radiance unit watt
per square metre per steradian?
SELECT DISTINCT ?x0 WHERE {
?x0 :type.object.type :measurement_unit.measurement_system .
VALUES ?x1 { :m.02sj5gj }
?x0 :measurement_unit.measurement_system.radiance_units ?x1 .
```



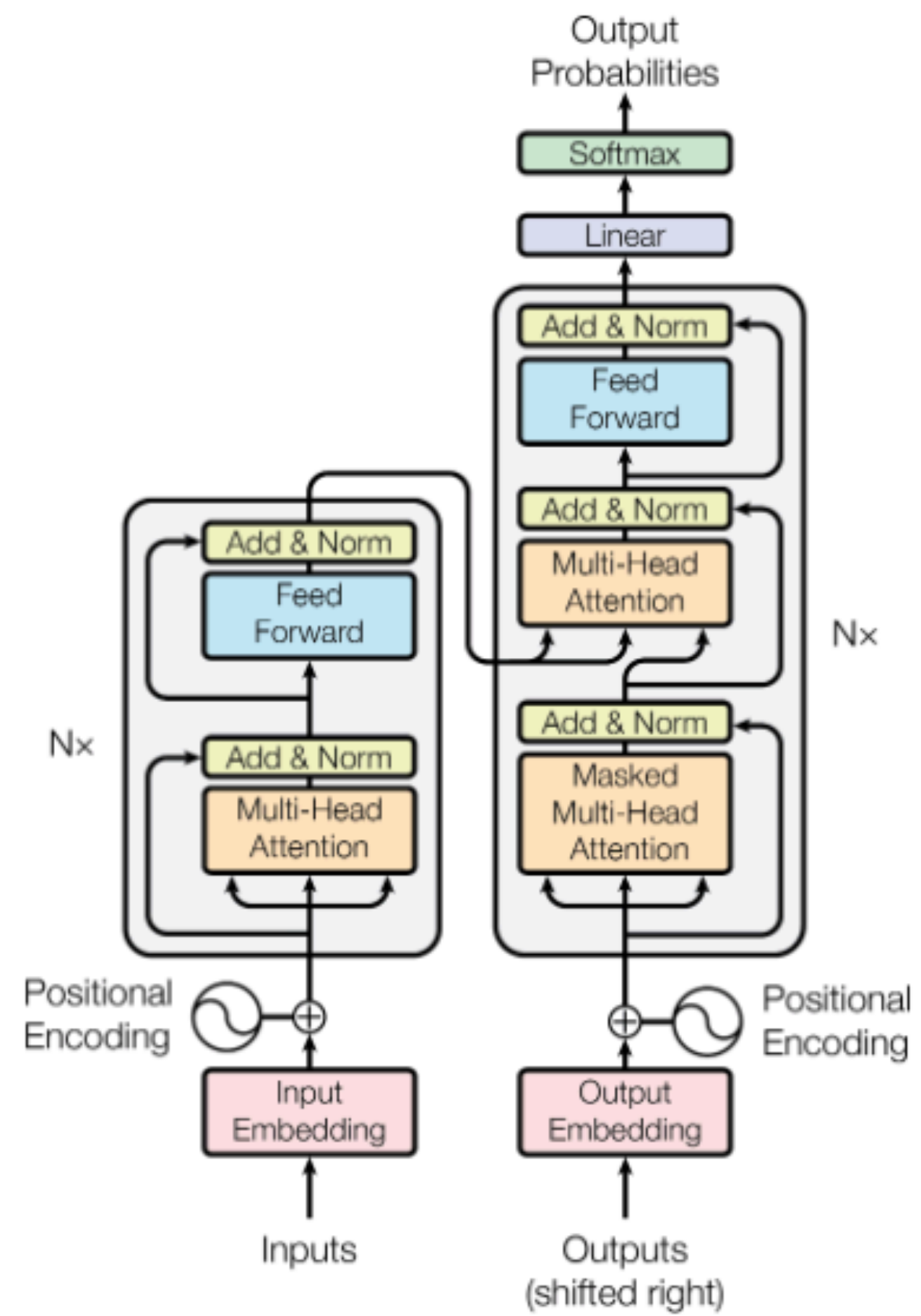
```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-
ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
SELECT (?x0 AS ?value) WHERE {\nSELECT DISTINCT ?x0
WHERE {
?x0 :type.object.type :rail.rail_network .
VALUES ?x1 { :m.03cwwdi }
?x0 :rail.rail_network.name ?x1 .
FILTER ( ?x0 != ?x1 )\n}}}
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-
ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://rdf.freebase.com/ns/>
How many video games has randy condon designed?
SELECT (COUNT(?x0) AS ?value) WHERE {\nSELECT DISTINCT ?
x0 WHERE {
?x0 :type.object.type :cvg.computer_videogame . \nVALUES
?x1 { :m.02hptvh }
```

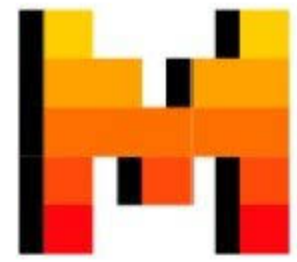
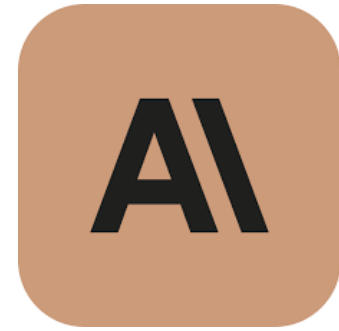
Goal #1 : Make NL Interfaces for querying KGs

Parametric Models for NLP

Parametric Models for NLP

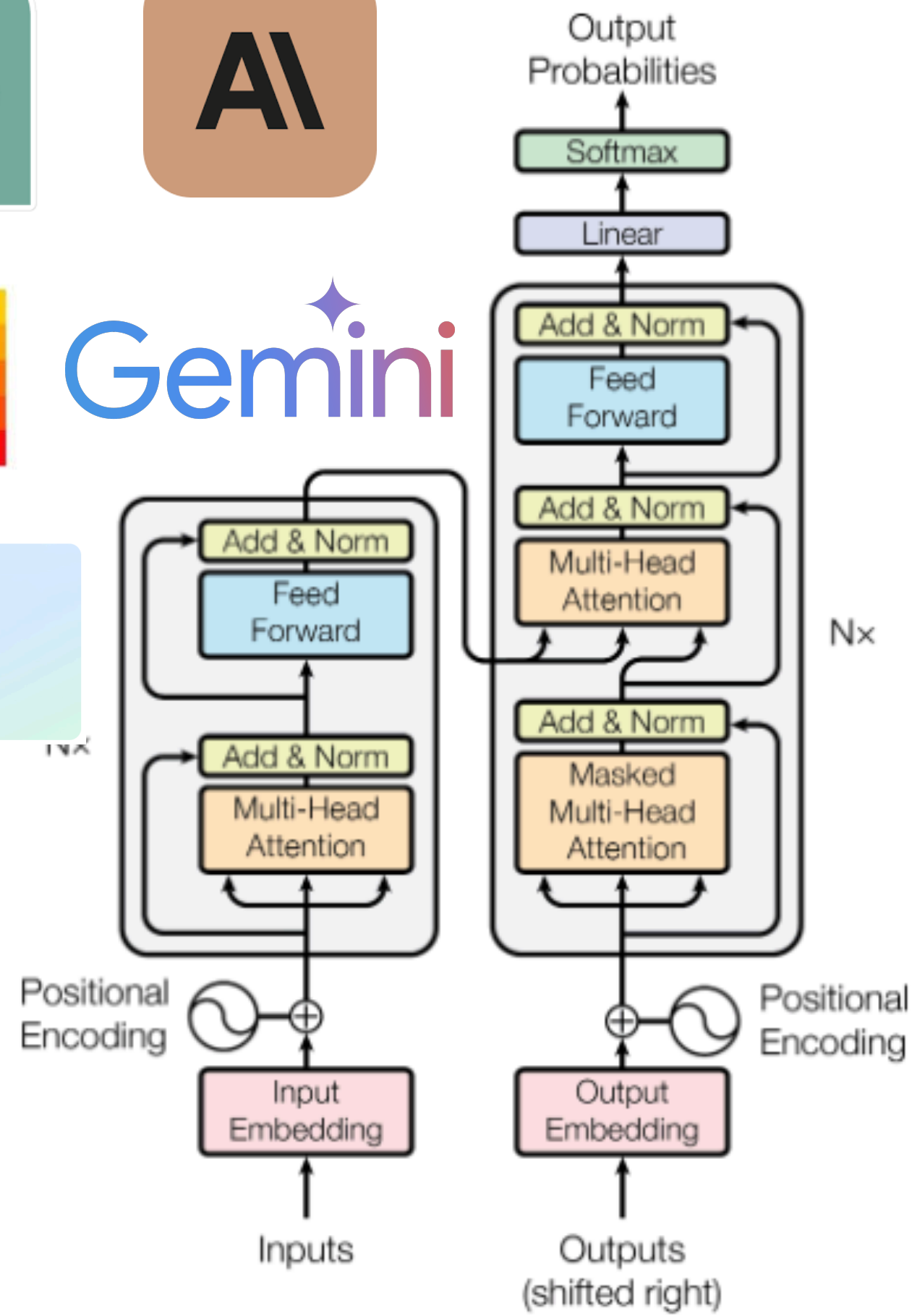


Parametric Models for NLP

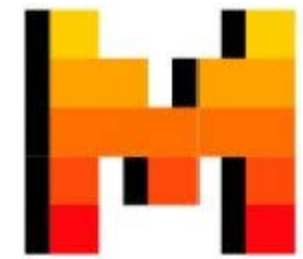


Gemini

Llama

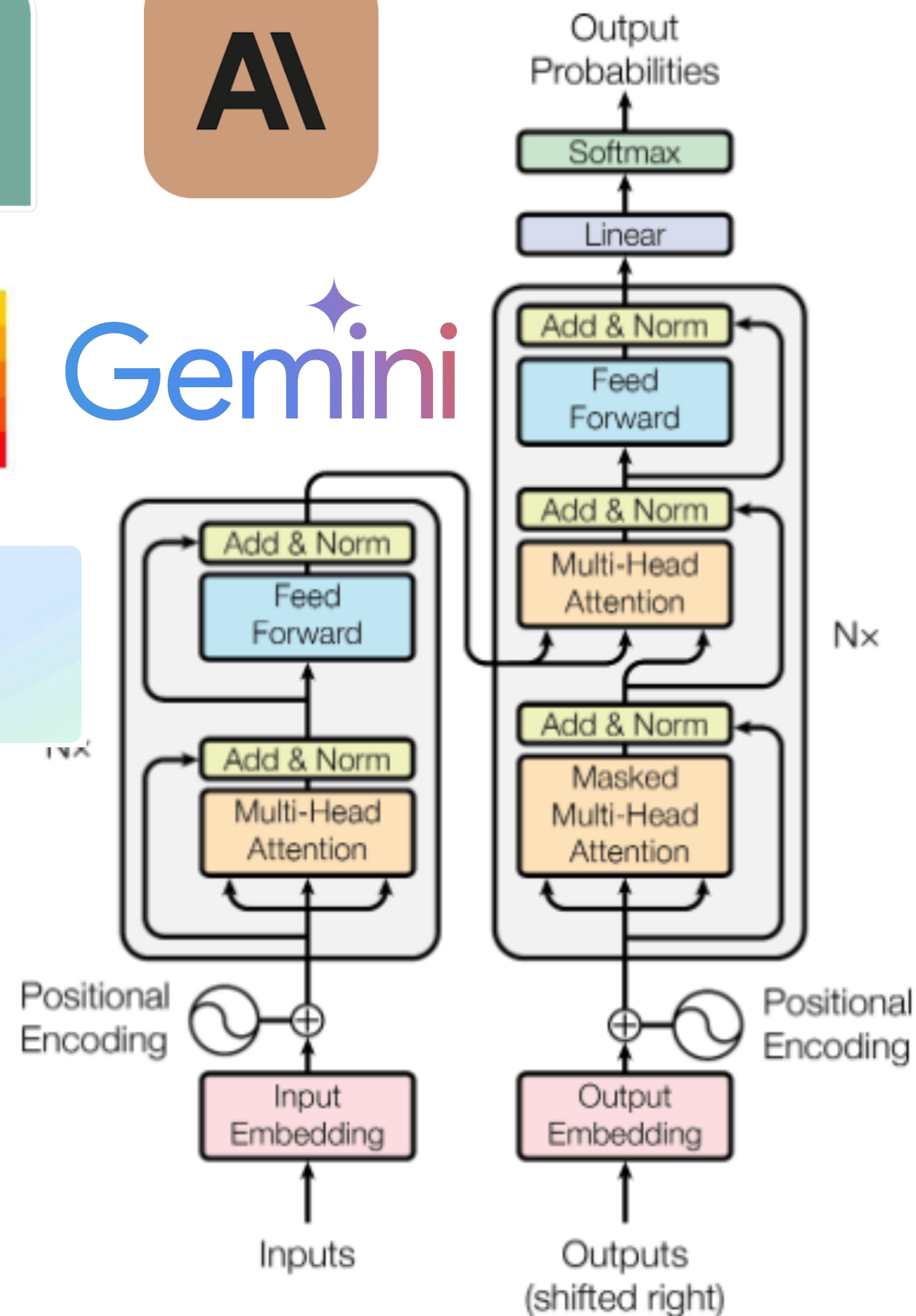


Parametric Models for NLP



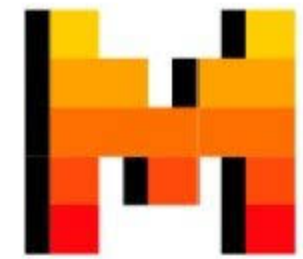
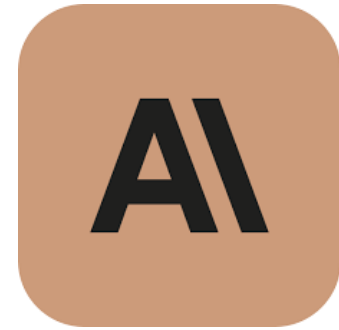
Gemini

Llama



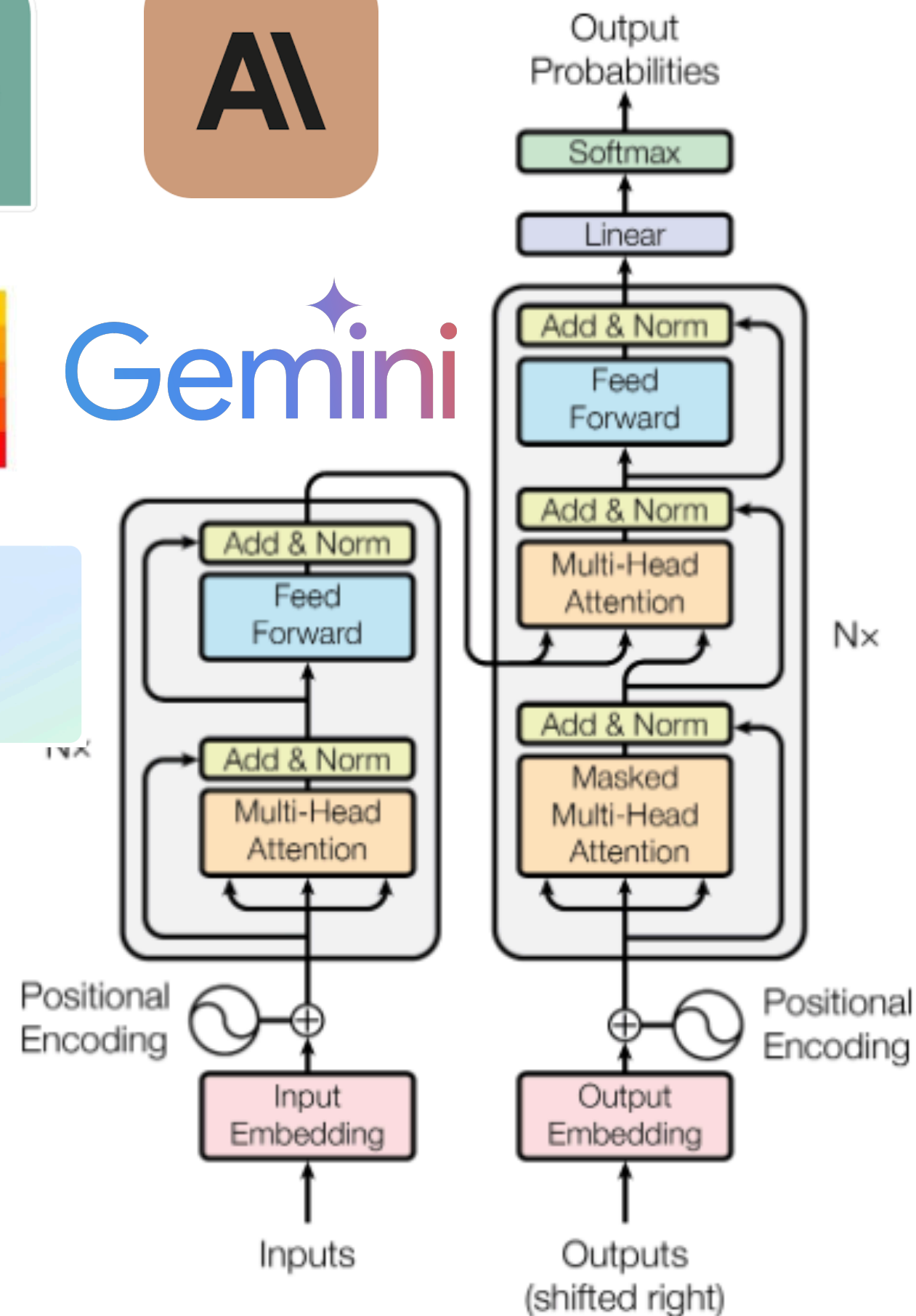
- ✓ Dramatic success of pre-trained LMs on many NLP tasks
- ✓ Model parameters pack a lot of knowledge
- ✓ Emergent abilities at massive scales.

Parametric Models for NLP



Gemini

Llama

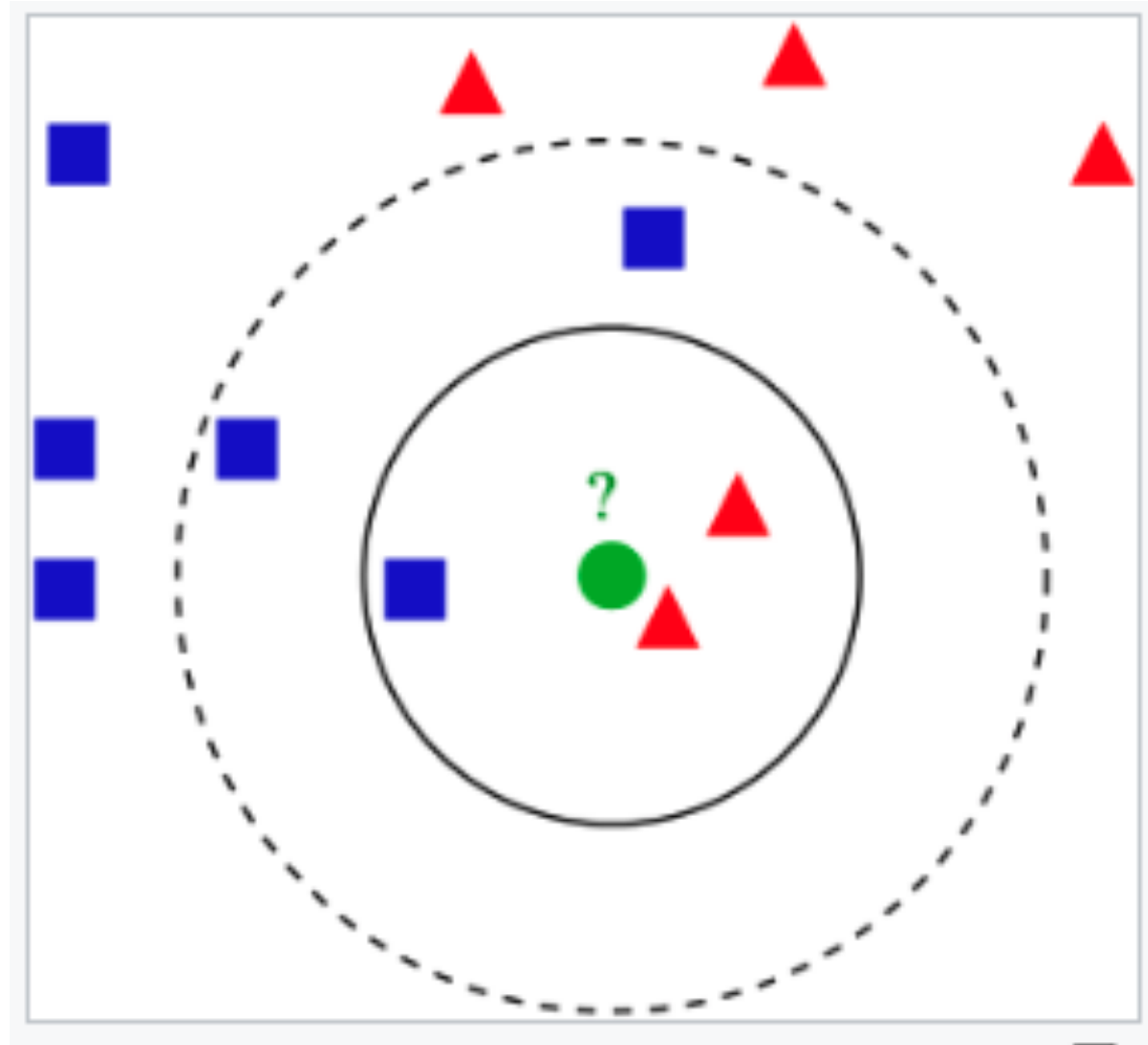


- ✓ Dramatic success of pre-trained LMs on many NLP tasks
- ✓ Model parameters pack a lot of knowledge
- ✓ Emergent abilities at massive scales.

- Lack of transparency into model mechanisms
- Hard to update/add new knowledge
- Less controllable

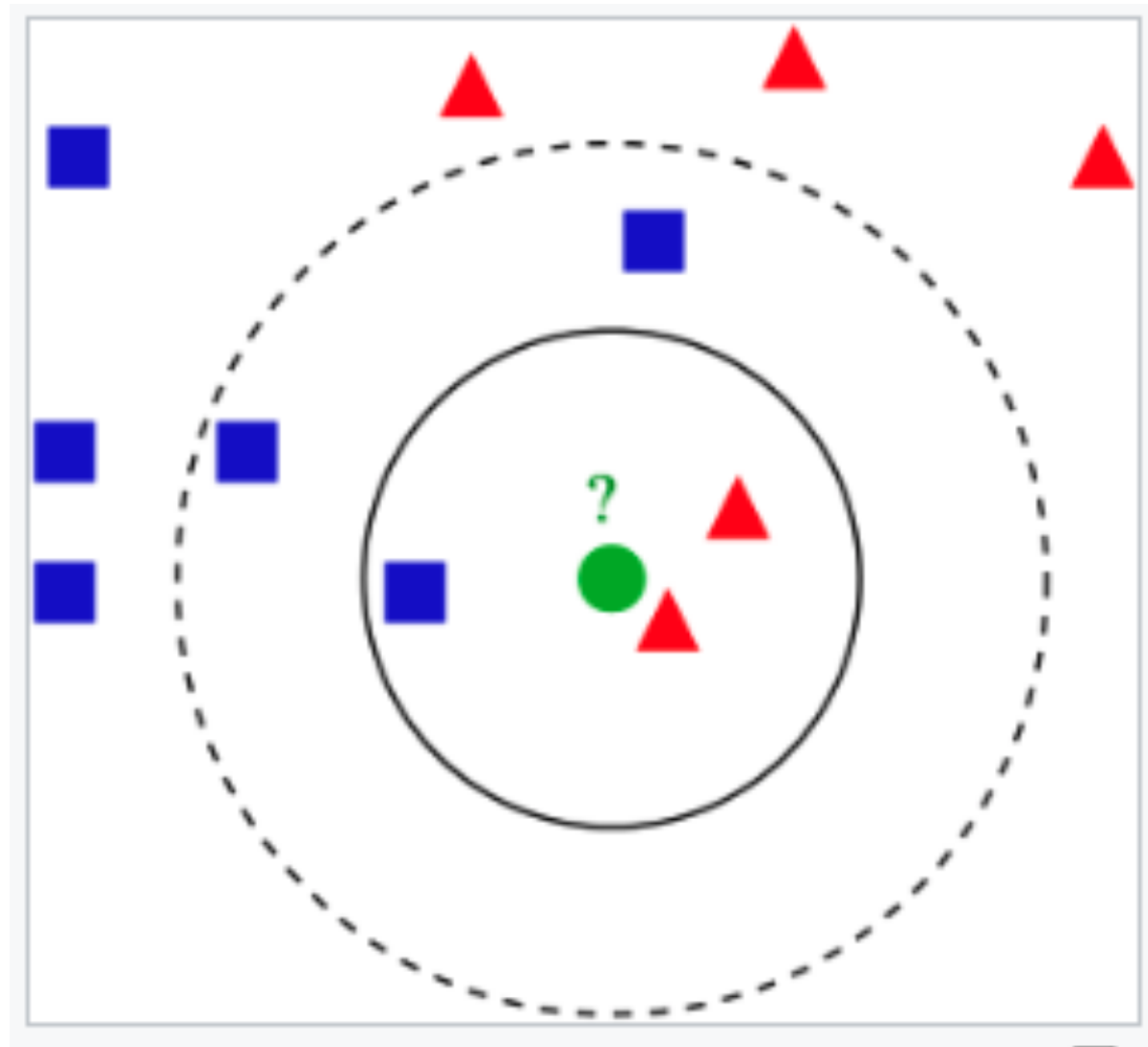
Nonparametric NLP Models

Nonparametric NLP Models



K-nearest neighbor classification
(Image from Wikipedia)

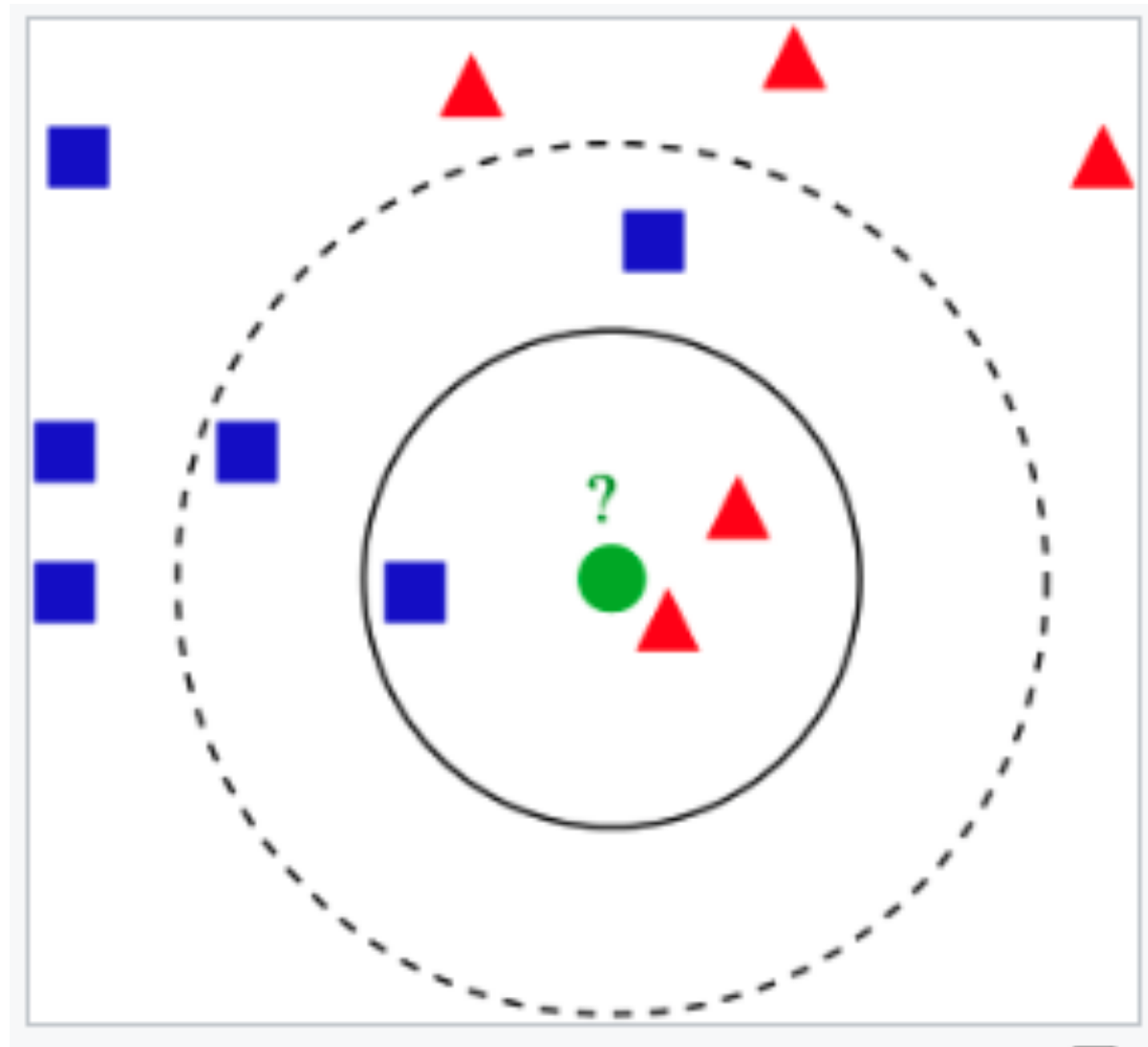
Nonparametric NLP Models



K-nearest neighbor classification
(Image from Wikipedia)

- ✓ Addition and deletion of data is easy!
- ✓ Model prediction explained by observing nearest neighbor
- ✓ Controllable - a prediction gone wrong can be fixed by adding more nearest neighbors

Nonparametric NLP Models



K-nearest neighbor classification
(Image from Wikipedia)

- ✓ Addition and deletion of data is easy!
- ✓ Model prediction explained by observing nearest neighbor
- ✓ Controllable - a prediction gone wrong can be fixed by adding more nearest neighbors

- Much less accurate than parametric models

Semiparametric Models

Semiparametric Models

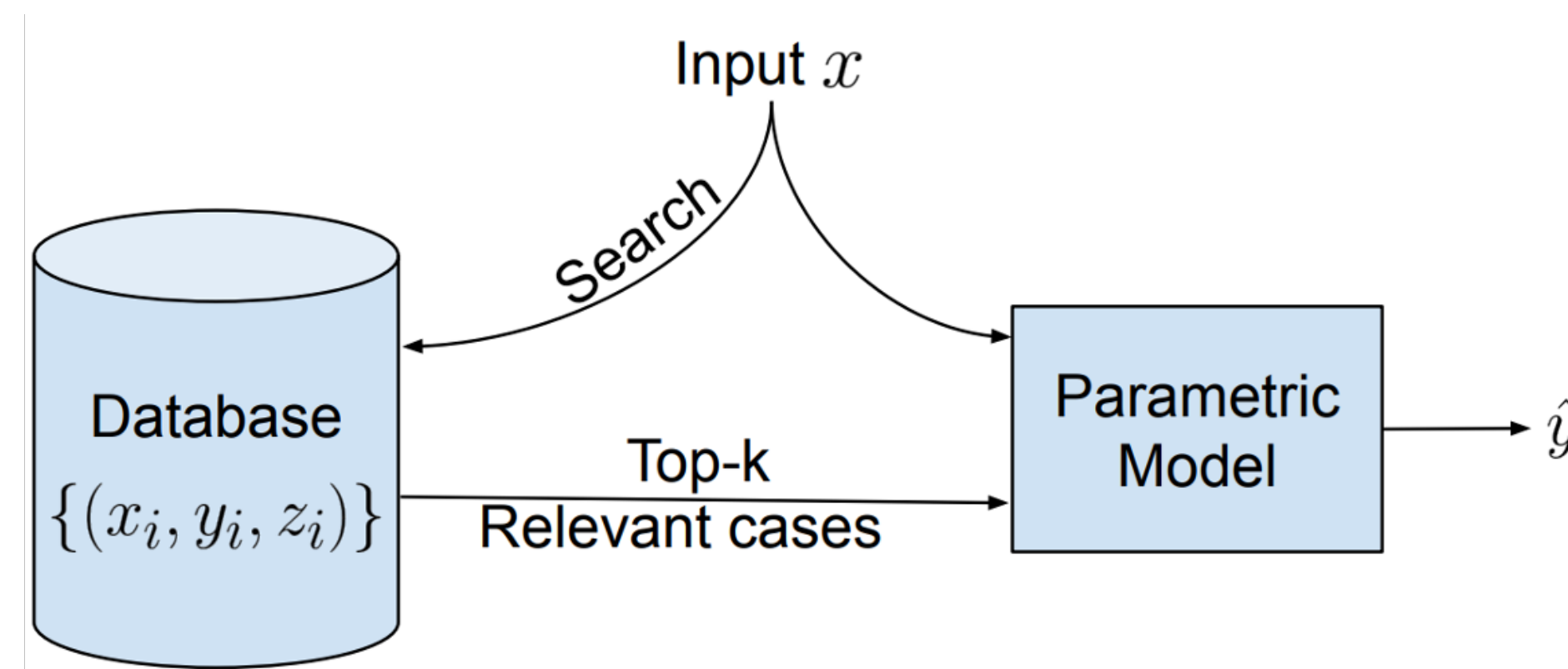
From Wikipedia, the free encyclopedia

In [statistics](#), a **semiparametric model** is a [statistical model](#) that has [parametric](#) and [nonparametric](#) components.

Semiparametric Models

From Wikipedia, the free encyclopedia

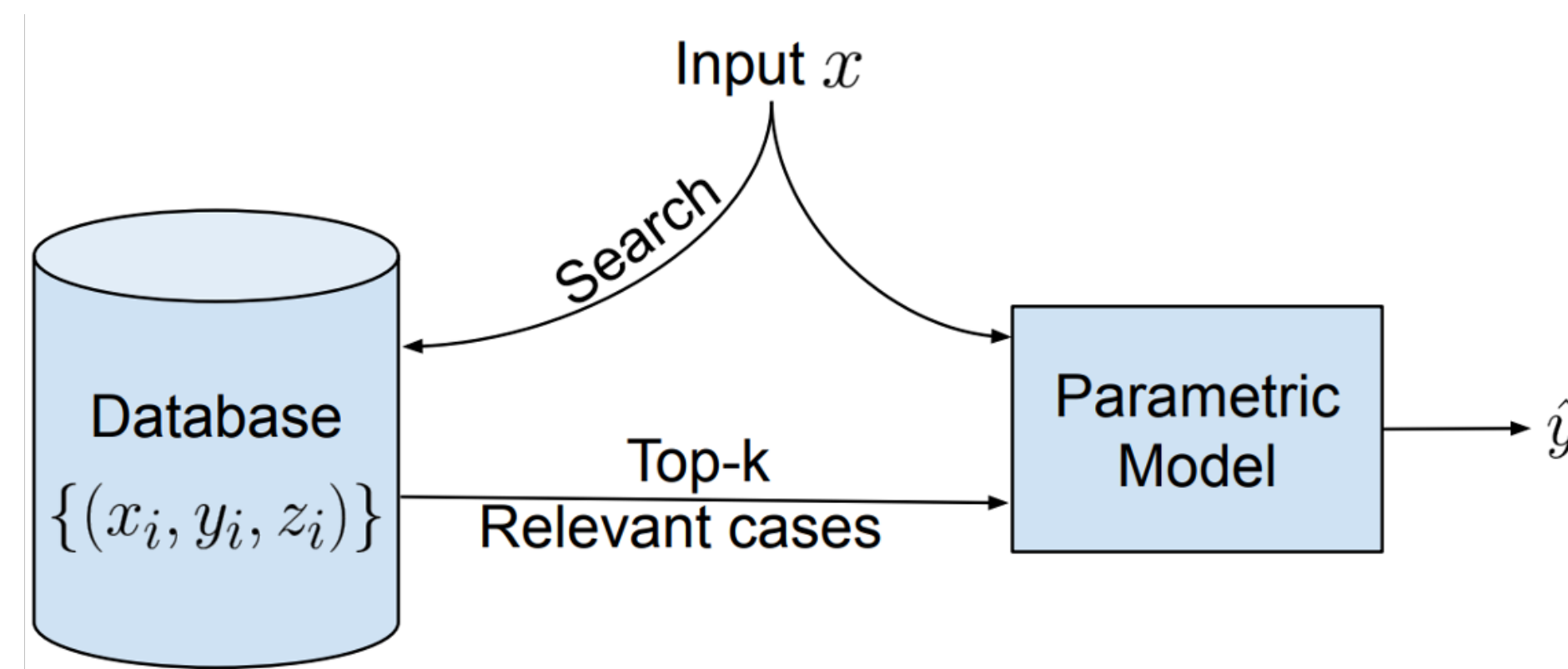
In [statistics](#), a **semiparametric model** is a [statistical model](#) that has [parametric](#) and [nonparametric](#) components.



Semiparametric Models

From Wikipedia, the free encyclopedia

In **statistics**, a **semiparametric model** is a **statistical model** that has **parametric** and **nonparametric** components.

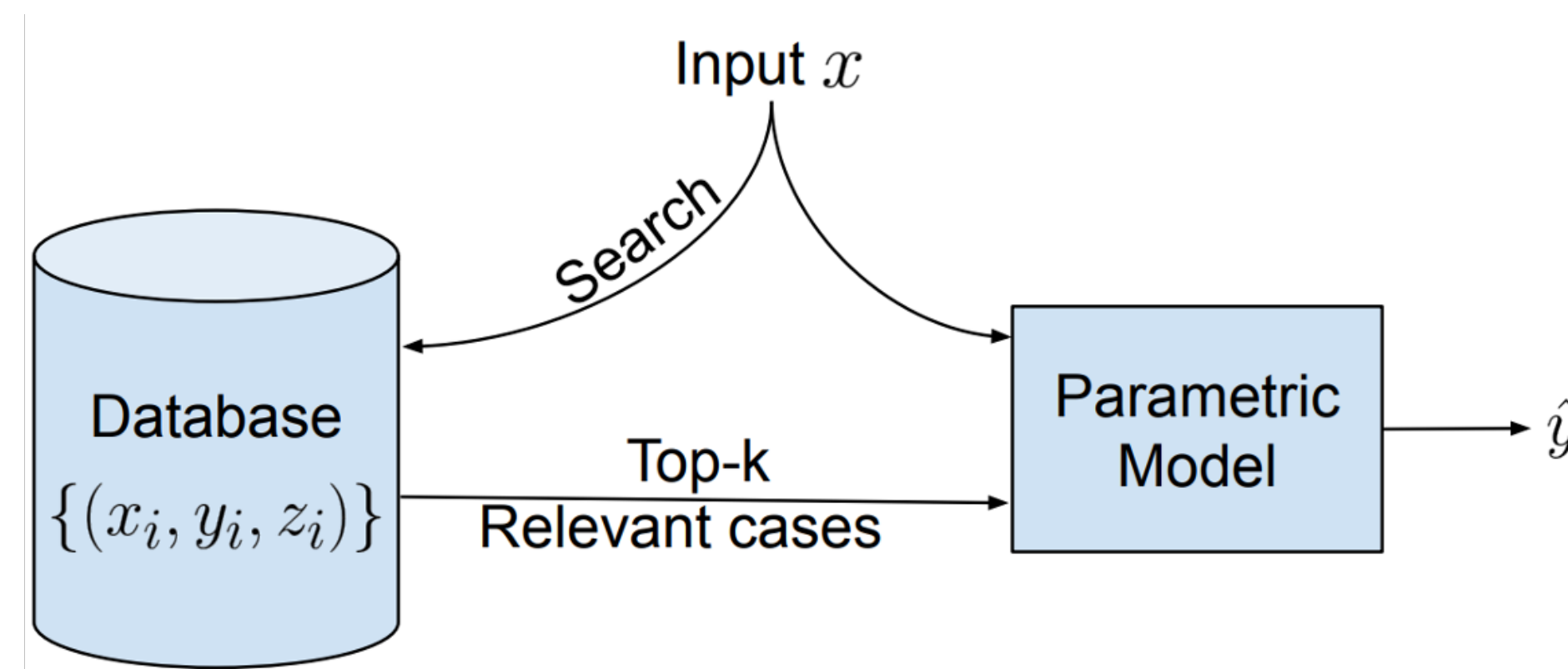


- ◆ Parametric component \rightarrow encodes logic
- ◆ Nonparametric component \rightarrow retrieves knowledge

Semiparametric Models

From Wikipedia, the free encyclopedia

In **statistics**, a **semiparametric model** is a **statistical model** that has **parametric** and **nonparametric** components.



- ◆ Parametric component \rightarrow encodes logic
- ◆ Nonparametric component \rightarrow retrieves knowledge

Question Answering (Chen et al 2017, Karpukhin et al 2020, Lewis et al 2021, etc)
Machine Translation (Gu et al 2018, Khandelwal et al 2021)
Language Modeling (Lee et al 2019, Khandelwal et al 2020)
Semantic Parsing (Das et al 2021)
Protein Structure Prediction (Jumper et al 2021)

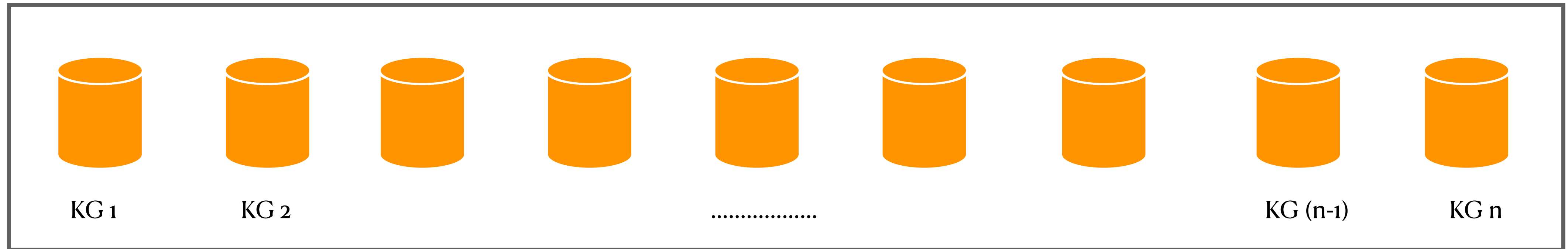
Challenges in Building NL Interfaces

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.

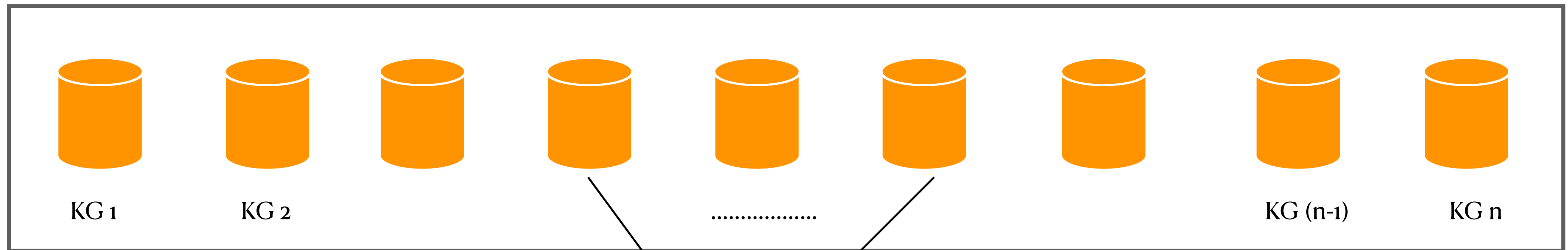
Challenges in Building NL Interfaces

◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.



Challenges in Building NL Interfaces

◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.



Can have entirely different schemas

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions

What are the top 5 countries by GDP that have a free trade agreement with the European Union, and what are the key industries that drive their economic growth?

Which countries have the highest percentage of their electricity generation coming from renewable sources?

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions
- ◆ Lack of training data

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions
- ◆ Lack of training data
 - ◆ Need experts to annotate, hence expensive

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions
- ◆ Lack of training data
 - ◆ Need experts to annotate, hence expensive
 - ◆ For broad KGs (e.g. Wikidata) hard to obtain coverage for all query types

Challenges in Building NL Interfaces

- ◆ In an enterprise, plethora of KGs (each with their own schema), created by each team.
- ◆ Complex Questions
- ◆ Lack of training data
 - ◆ Need experts to annotate, hence expensive
 - ◆ For broad KGs (e.g. Wikidata) hard to obtain coverage for all query types
 - ◆ Privacy concerns sometimes does not allow data collection.

Desiderata for a NL interface for KGs

Desiderata for a NL interface for KGs

- ◆ Build a natural language interface for querying KGs that

Desiderata for a NL interface for KGs

- ◆ Build a natural language interface for querying KGs that
 - ◆ Works for *Any Knowledge Graph* (irrespective of their schema)

Desiderata for a NL interface for KGs

- ◆ Build a natural language interface for querying KGs that
 - ◆ Works for *Any Knowledge Graph* (irrespective of their schema)
 - ◆ Requires *no training data* (i.e. completely unsupervised)

Desiderata for a NL interface for KGs

- ◆ Build a natural language interface for querying KGs that
 - ◆ Works for *Any Knowledge Graph* (irrespective of their schema)
 - ◆ Requires *no training data* (i.e. completely unsupervised)
 - ◆ Ready to be used in *reasonable time* (~1 day)

Key Difference from Prior Work

Key Difference from Prior Work

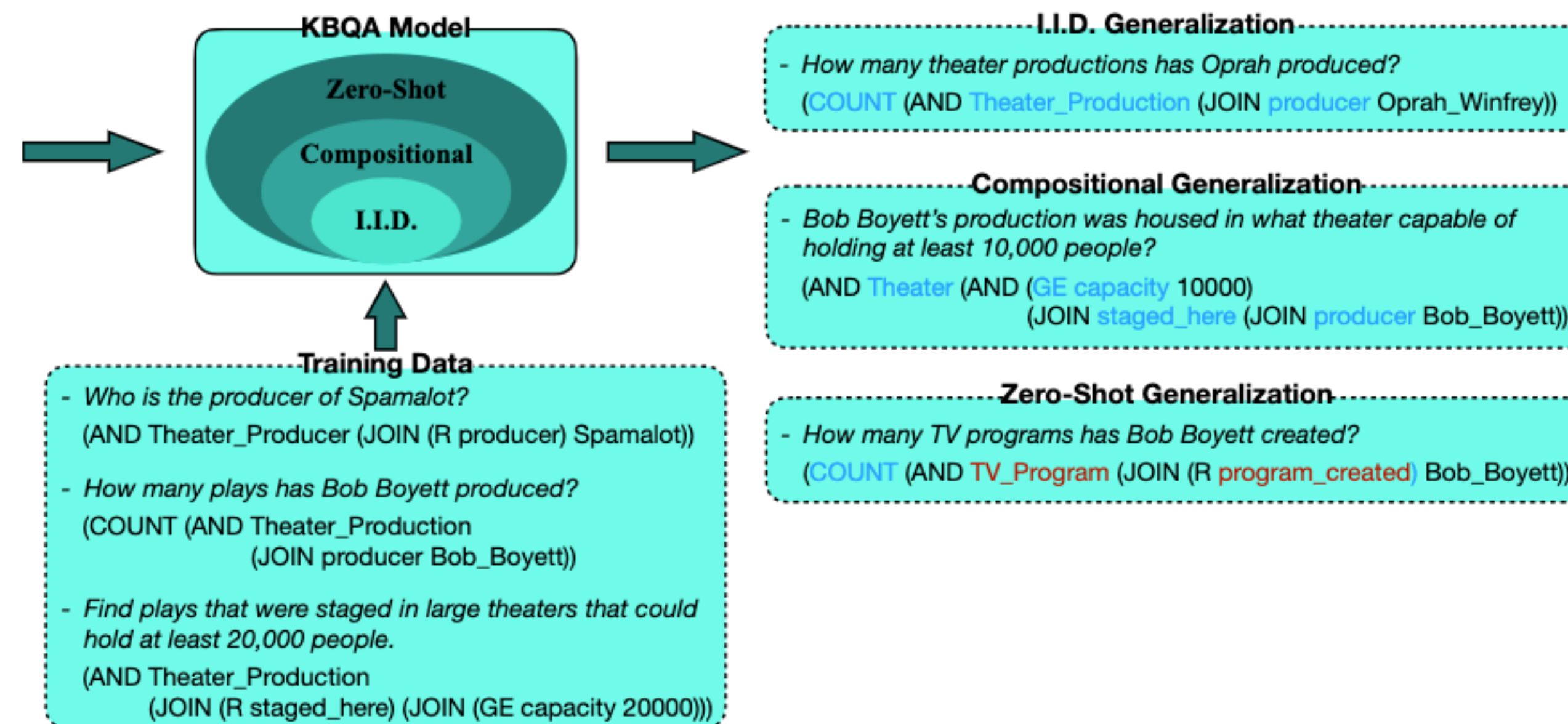
- ◆ Zero-shot QA over KGs is not new (Su et al 2021)

Key Difference from Prior Work

- ◆ Zero-shot QA over KGs is not new (Su et al 2021)
- ◆ However prior work assumes availability of *some training data* and queries that need a single unseen schema item were considered as zero-shot.

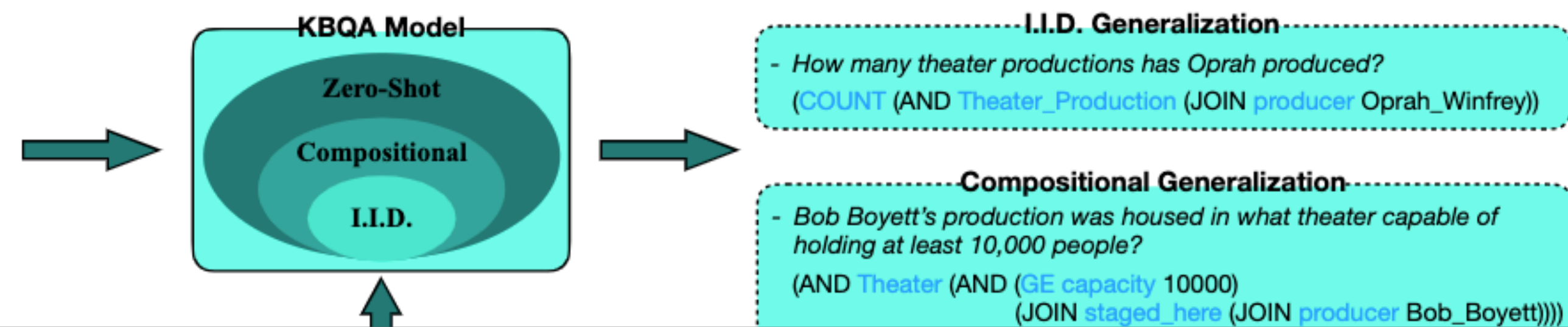
Key Difference from Prior Work

- ◆ Zero-shot QA over KGs is not new (Su et al 2021)
- ◆ However prior work assumes availability of *some training data* and queries that need a single unseen schema item were considered as zero-shot.



Key Difference from Prior Work

- ◆ Zero-shot QA over KGs is not new (Su et al 2021)
- ◆ However prior work assumes availability of *some training data* and queries that need a single unseen schema item were considered as zero-shot.



No training questions; no prior knowledge of KG schema and query distribution.

- Find plays that were staged in large theaters that could hold at least 20,000 people.
(AND Theater_Production (JOIN (R staged_here) (JOIN (GE capacity 20000))))

Our Approach:
Bring Your Own KG

BRING YOUR OWN KG: Self-Supervised Program Synthesis for Zero-Shot KGQA

Dhruv Agarwal^{1,*}, Rajarshi Das^{2,†}, Sopan Khosla^{2,†}, Rashmi Gangadharaiah²

¹University of Massachusetts Amherst, ²AWS AI Labs

dagarwal@cs.umass.edu, {dasrajar, sopankh, rgangad}@amazon.com

Abstract

We present BYOKG, a universal question-answering (QA) system that can operate on *any* knowledge graph (KG), requires no human-annotated training data, and can be ready to use within a day — attributes that are out-of-scope for current KGQA systems. BYOKG draws

of *some* training data (query-program pairs) (Tal-
mor and Berant, 2018; Keysers et al., 2020; Gu
et al., 2021; Dutt et al., 2023a; Sen et al., 2023),
which, in practice, might be unrealistic. For exam-
ple, in domains such as biomedicine and clinical
decision-making, training data may be completely
unavailable due to high collection costs and stringent



BRING YOUR OWN KG: Self-Supervised Program Synthesis for Zero-Shot KGQA

Dhruv Agarwal^{1,*}, Rajarshi Das^{2,†}, Sopan Khosla^{2,†}, Rashmi Gangadharaiah²

¹University of Massachusetts Amherst, ²AWS AI Labs

dagarwal@cs.umass.edu, {dasrajar, sopankh, rgangad}@amazon.com

Abstract

We present BYOKG, a universal question-answering (QA) system that can operate on *any* knowledge graph (KG), requires no human-annotated training data, and can be ready to use within a day — attributes that are out-of-scope for current KGQA systems. BYOKG draws

of *some* training data (query-program pairs) (Tal-
mor and Berant, 2018; Keysers et al., 2020; Gu
et al., 2021; Dutt et al., 2023a; Sen et al., 2023),
which, in practice, might be unrealistic. For exam-
ple, in domains such as biomedicine and clinical
decision-making, training data may be completely
unavailable due to high collection costs and stringent

How might a *Human* approach this?

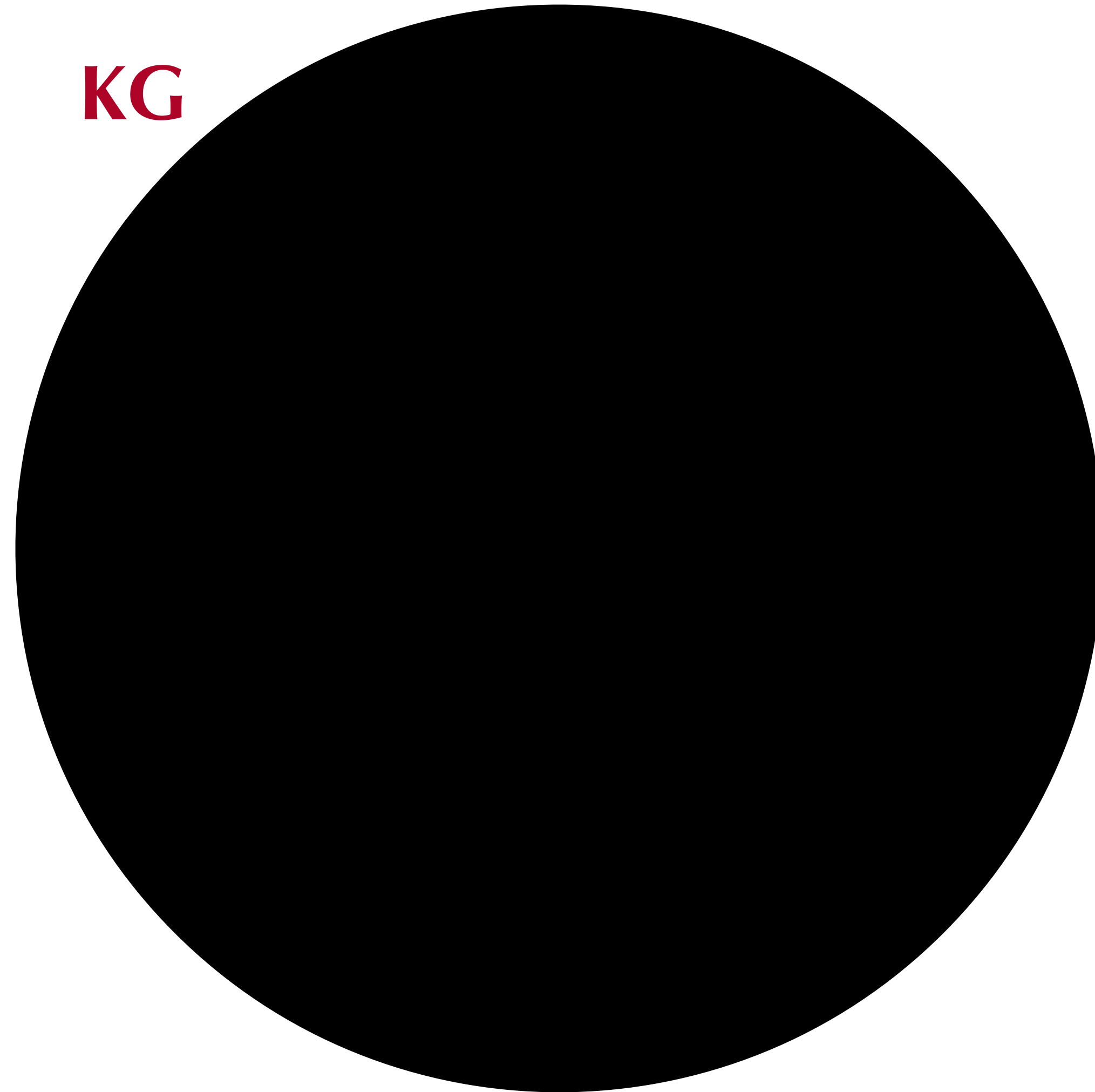
How might a *Human* approach this?

KG

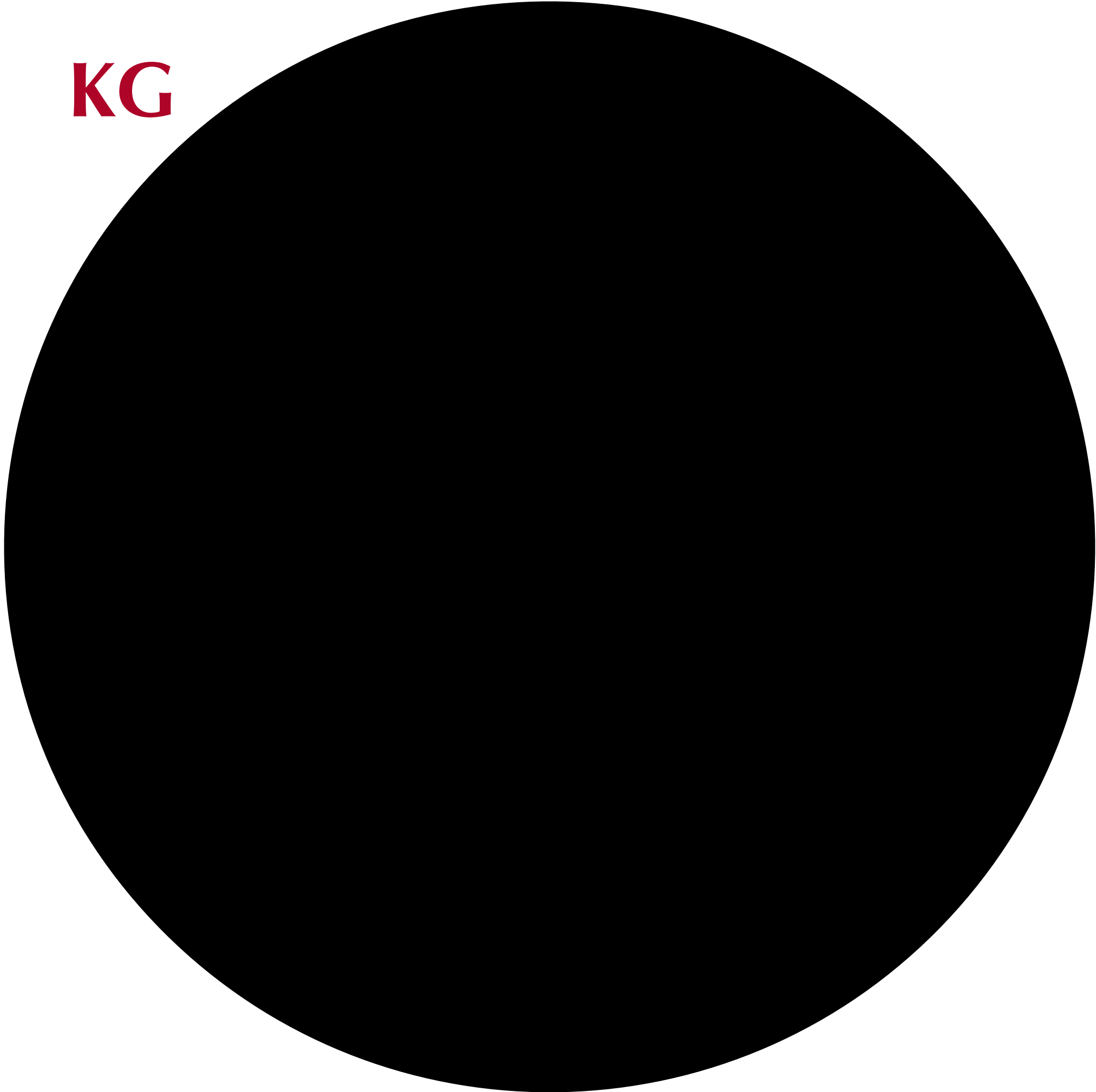
A large, solid black circle is centered on the page. The text 'KG' is positioned to the upper left of the circle's top edge.

How might a *Human* approach this?

KG

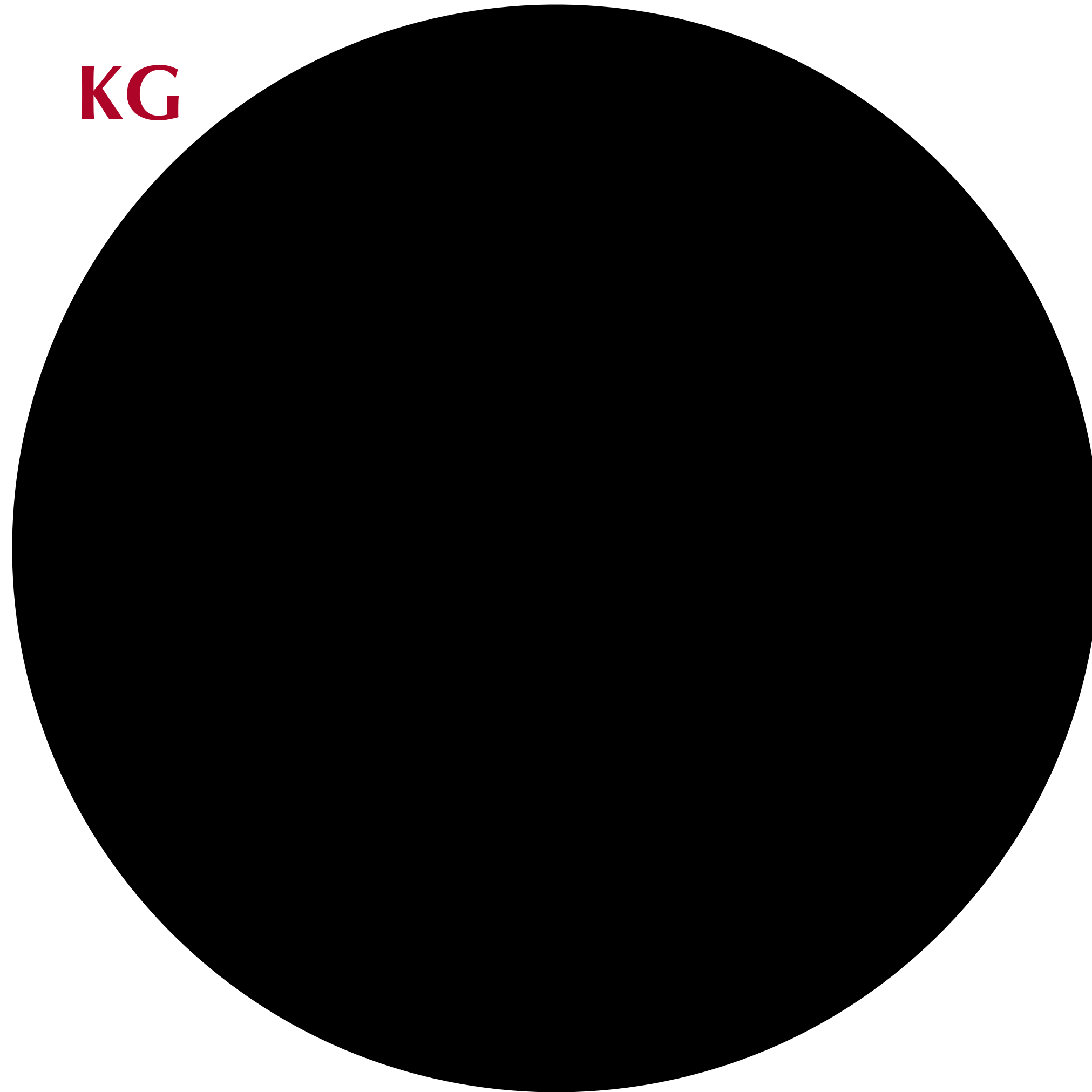


KG



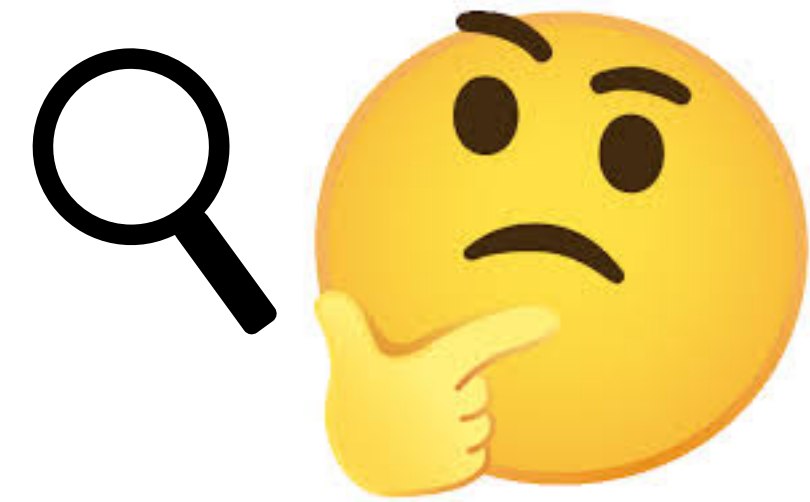
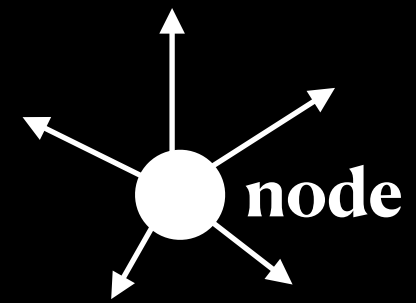
Curiosity!

KG

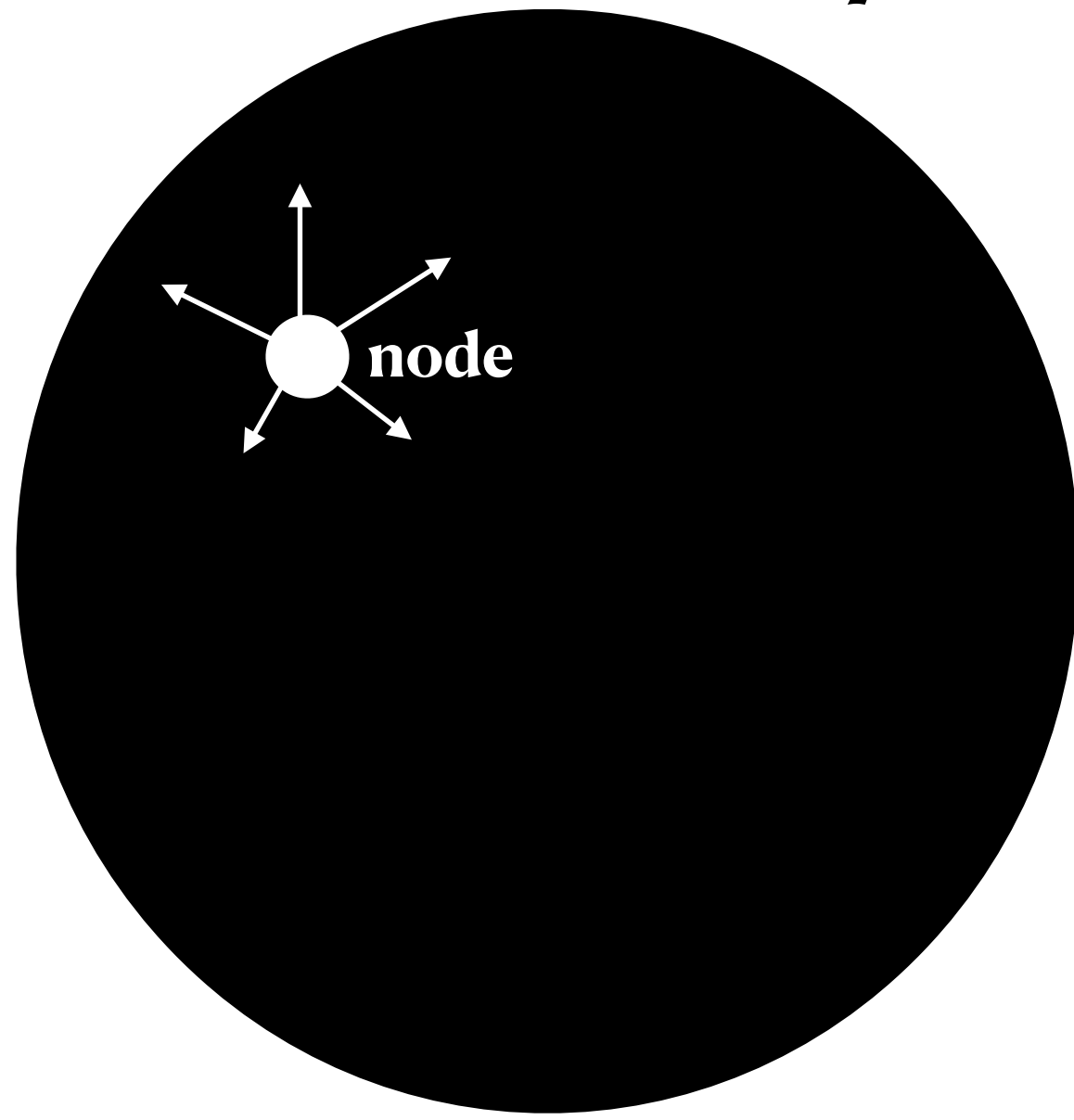


Curiosity!

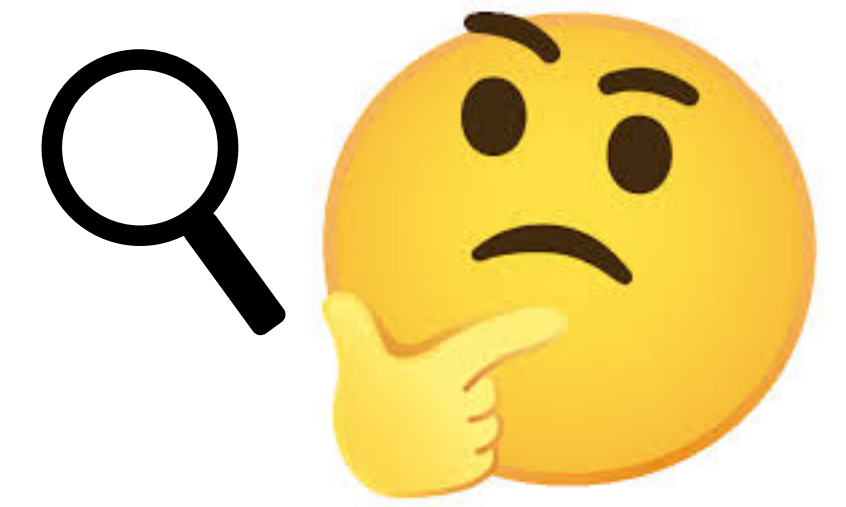
KG



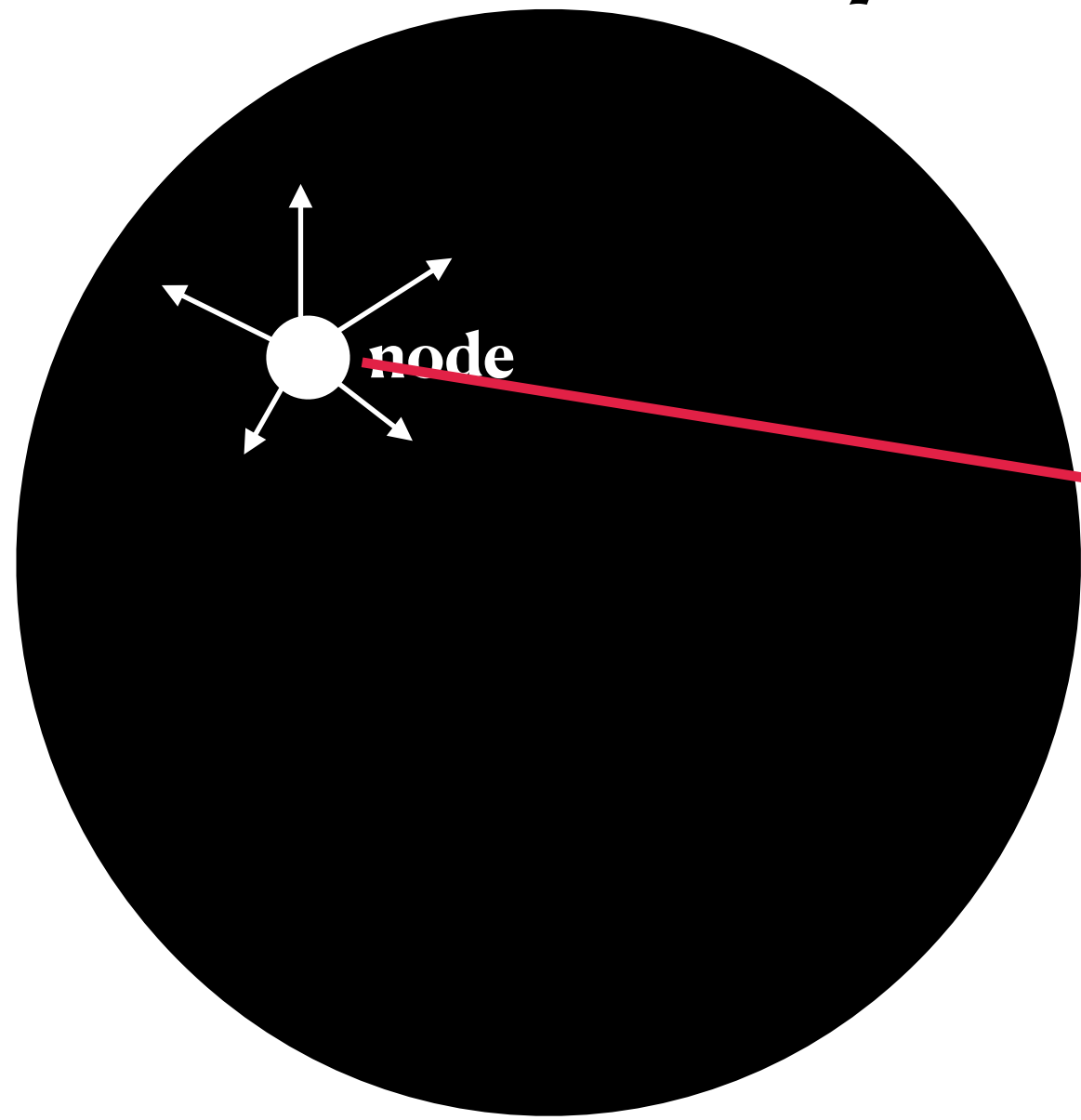
Curiosity!



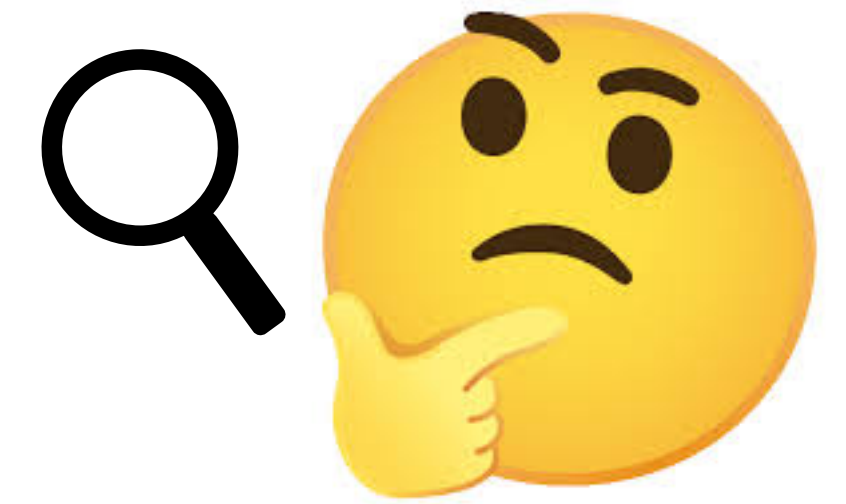
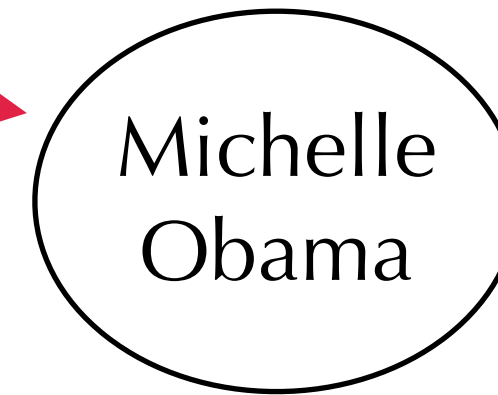
KG



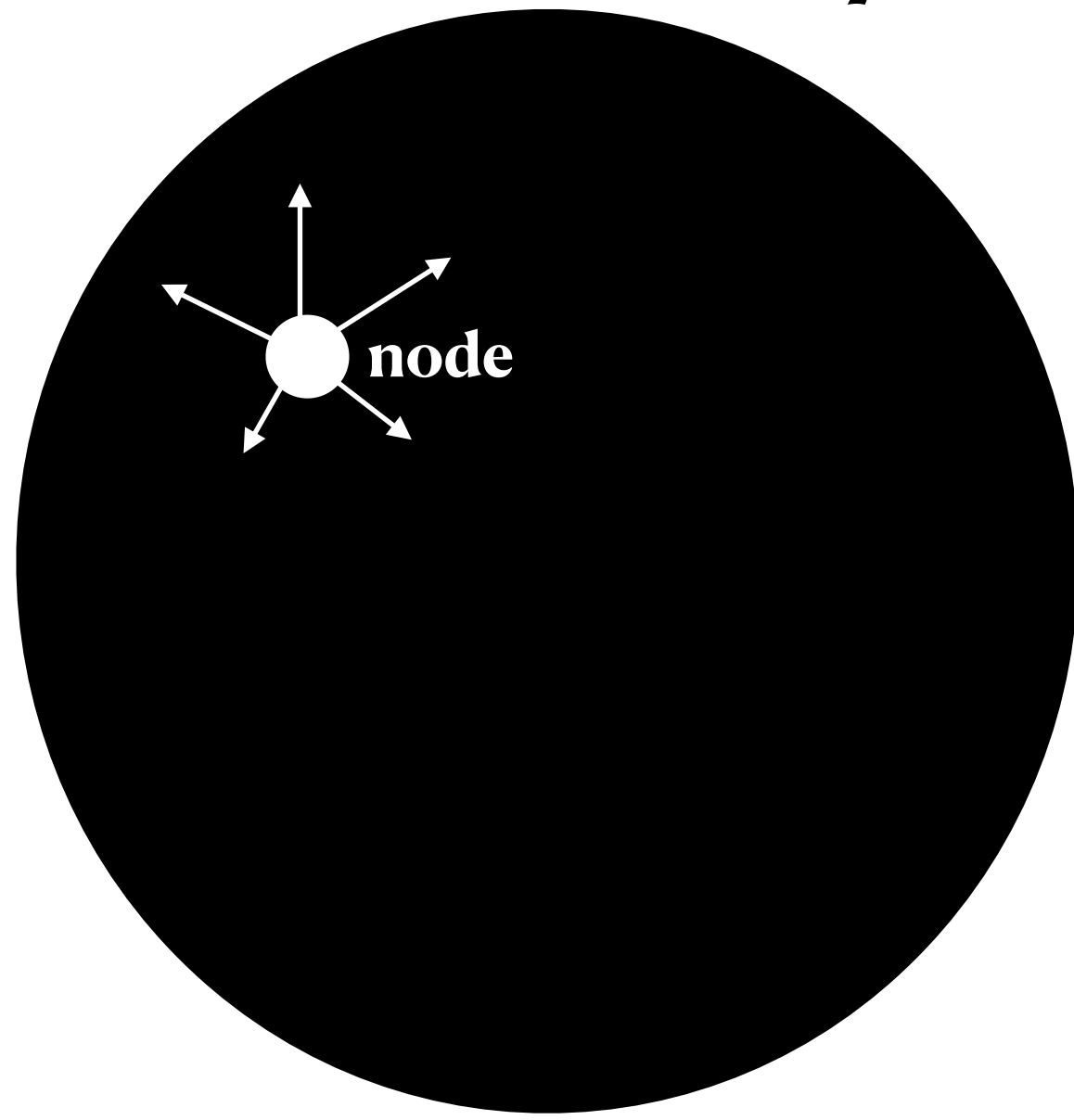
Curiosity!



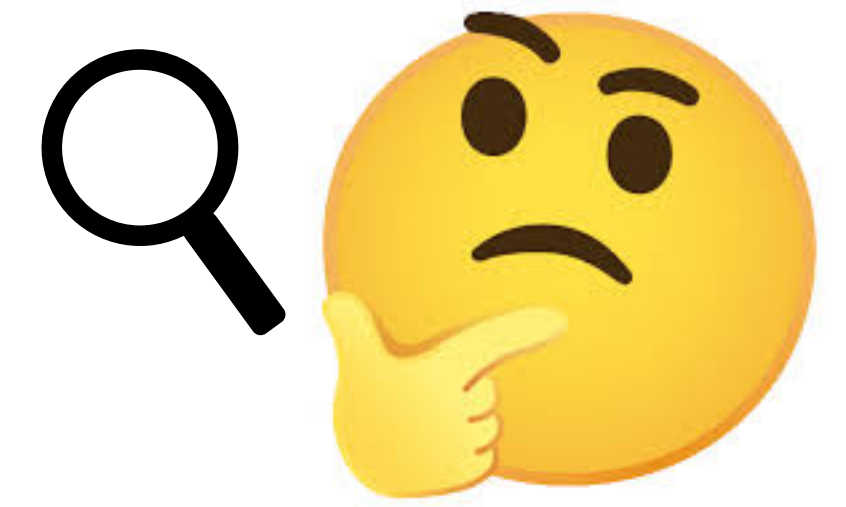
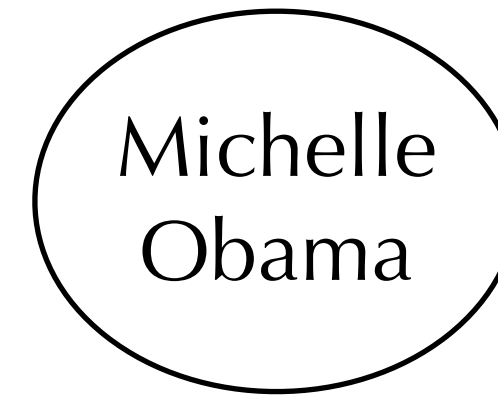
KG



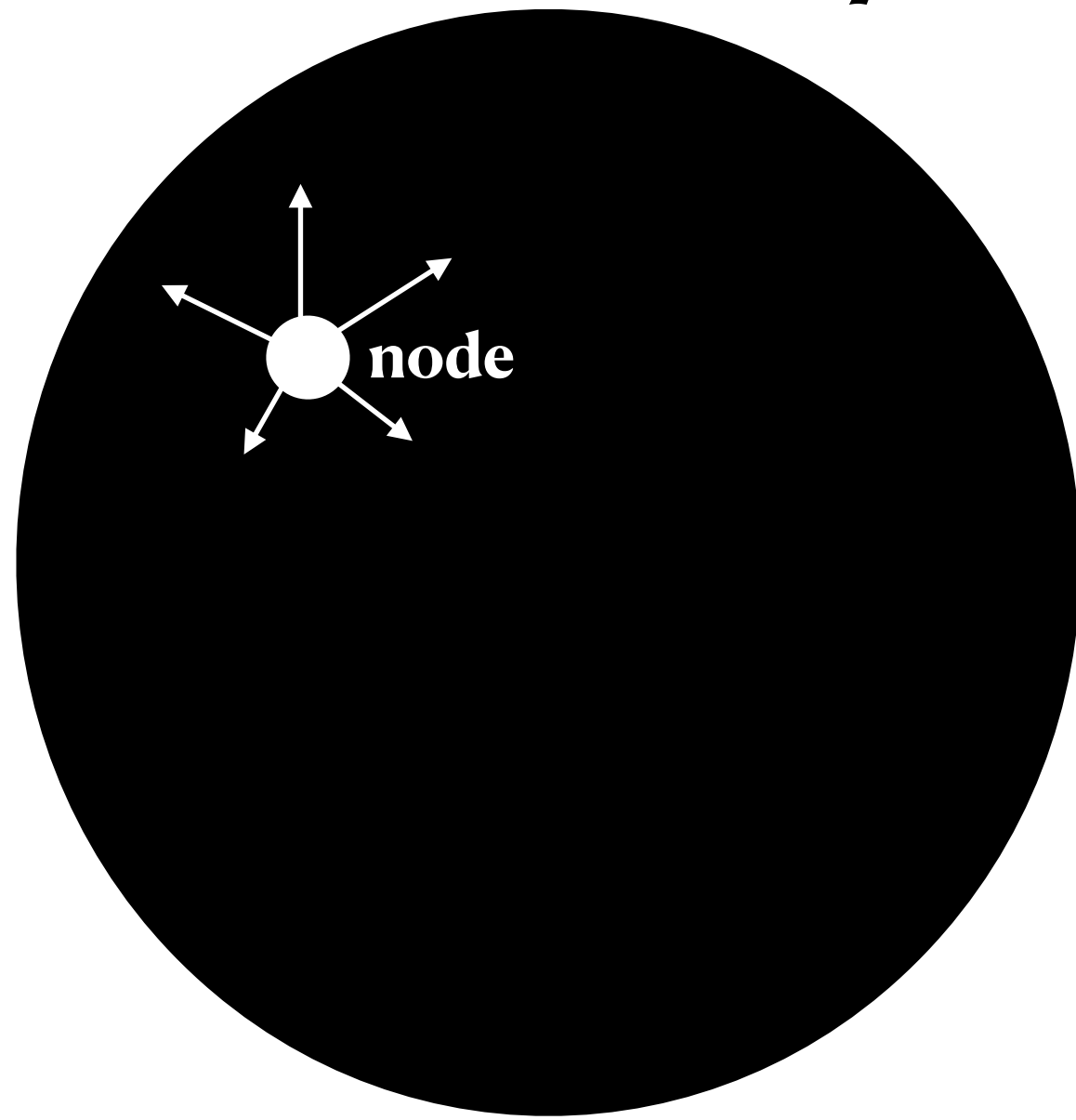
Curiosity!



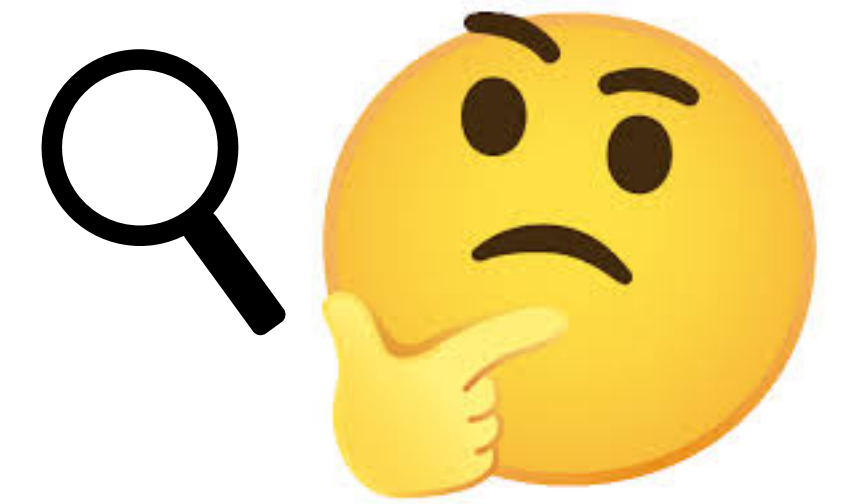
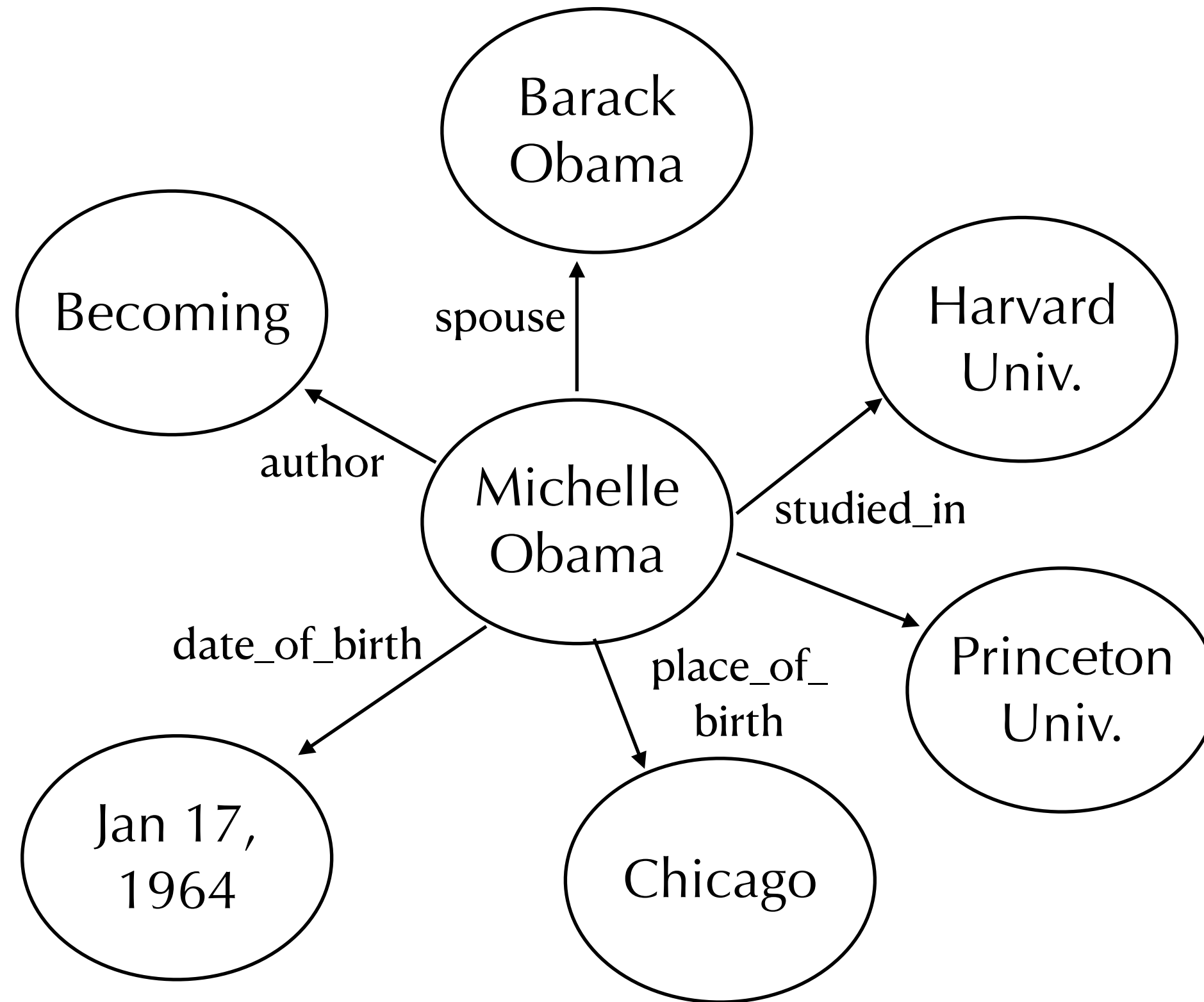
KG



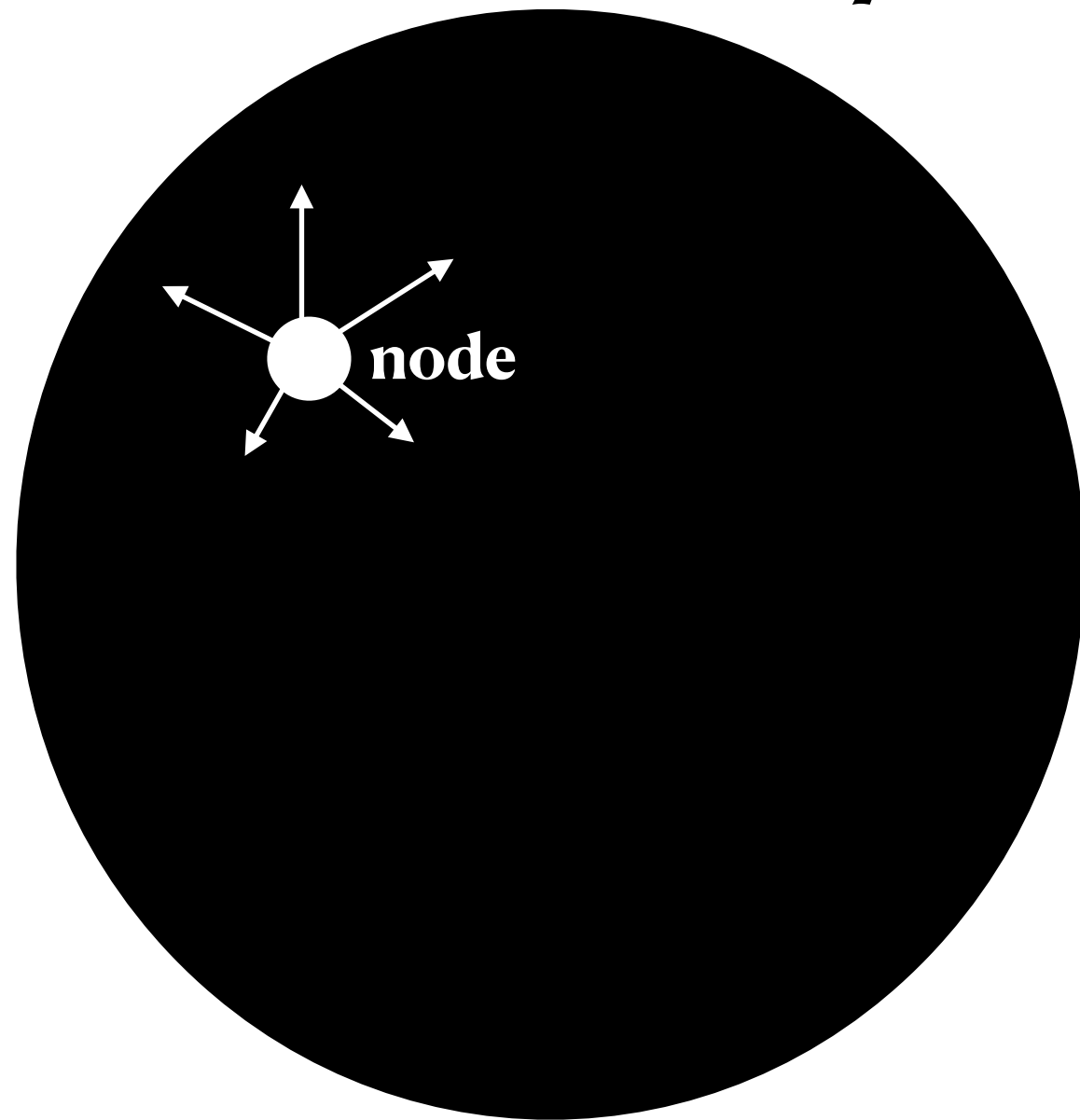
Curiosity!



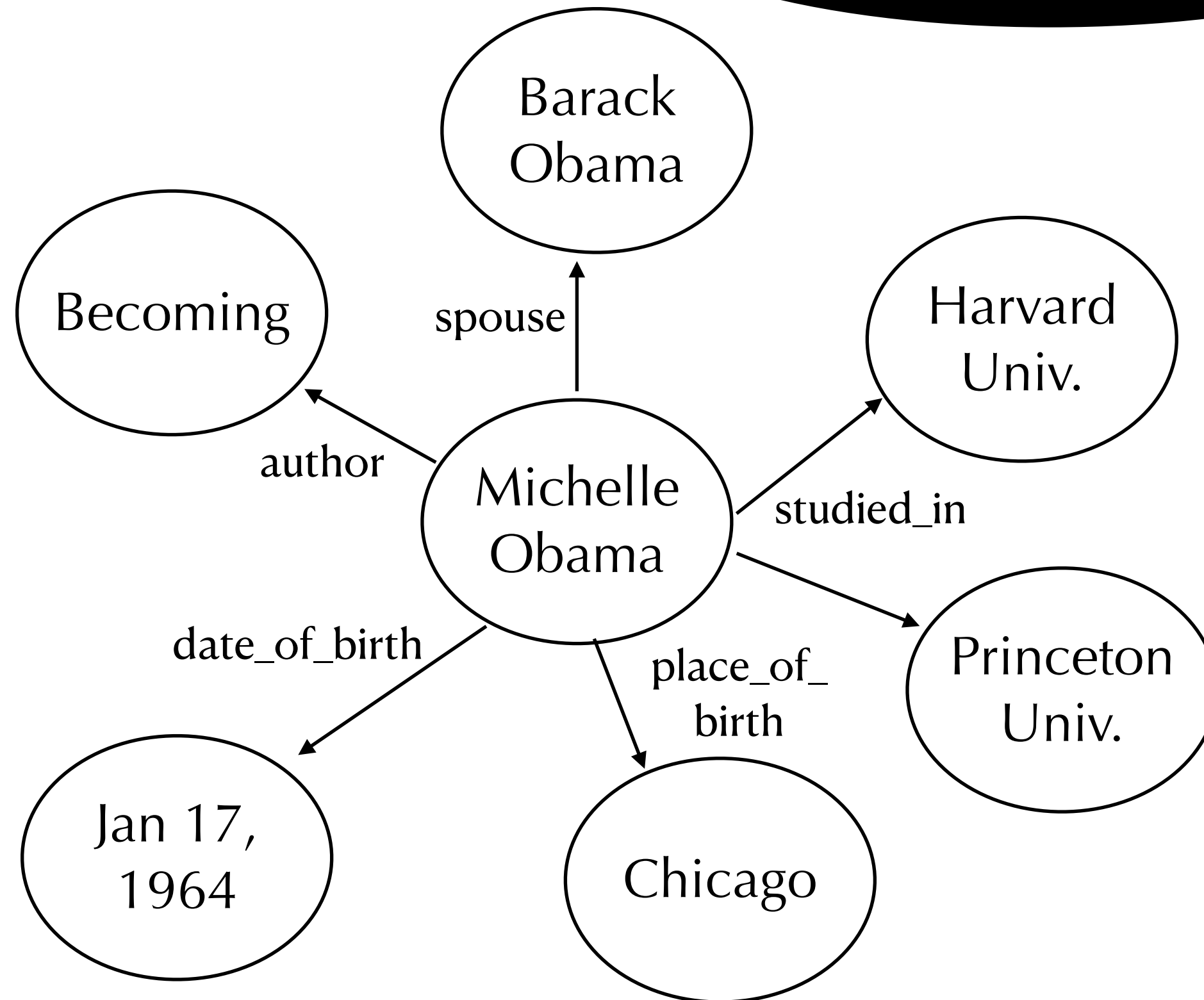
KG



Curiosity!



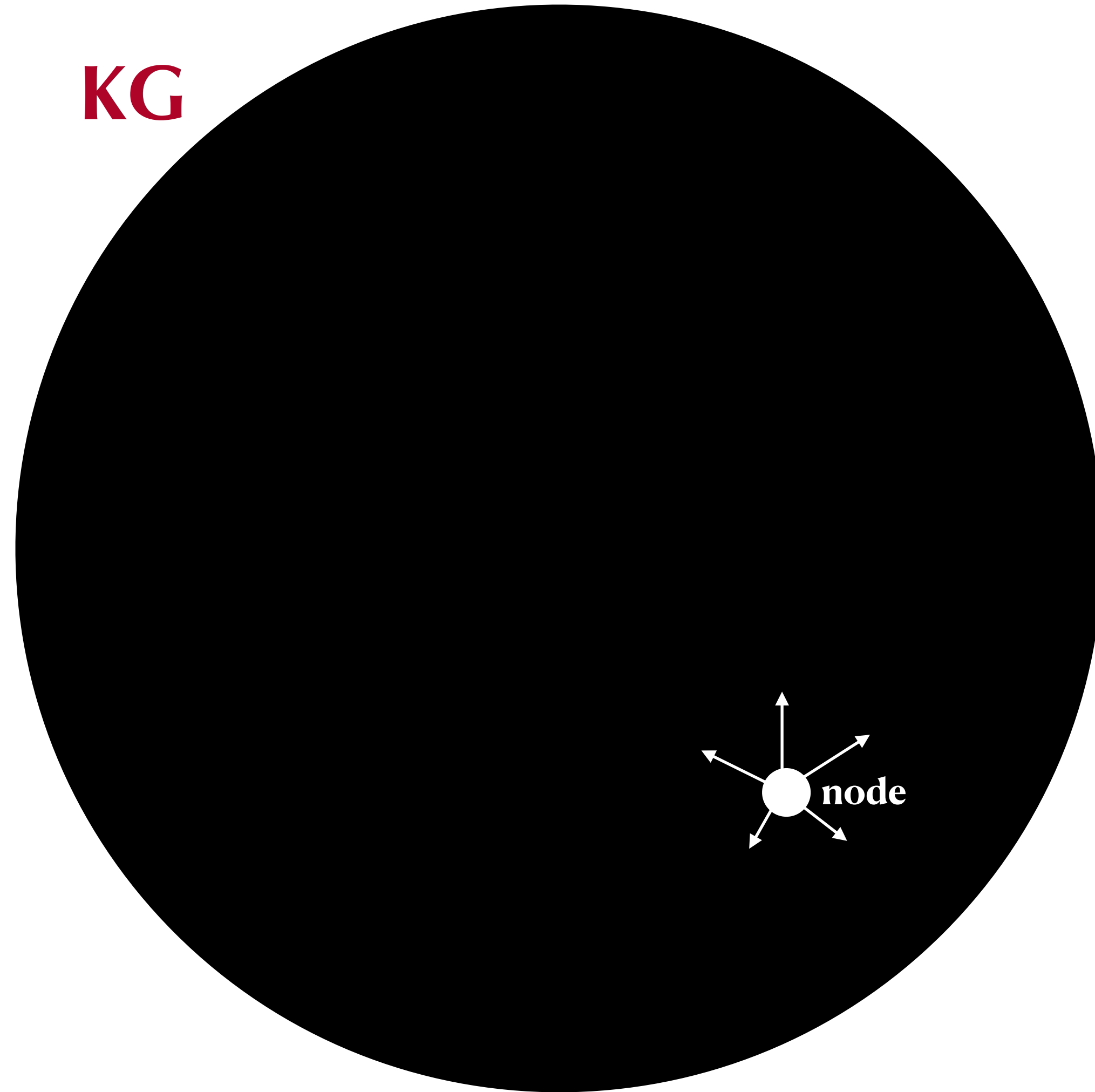
KG



The KG has information about famous people such as their place & date of birth, spouse and their professional achievements

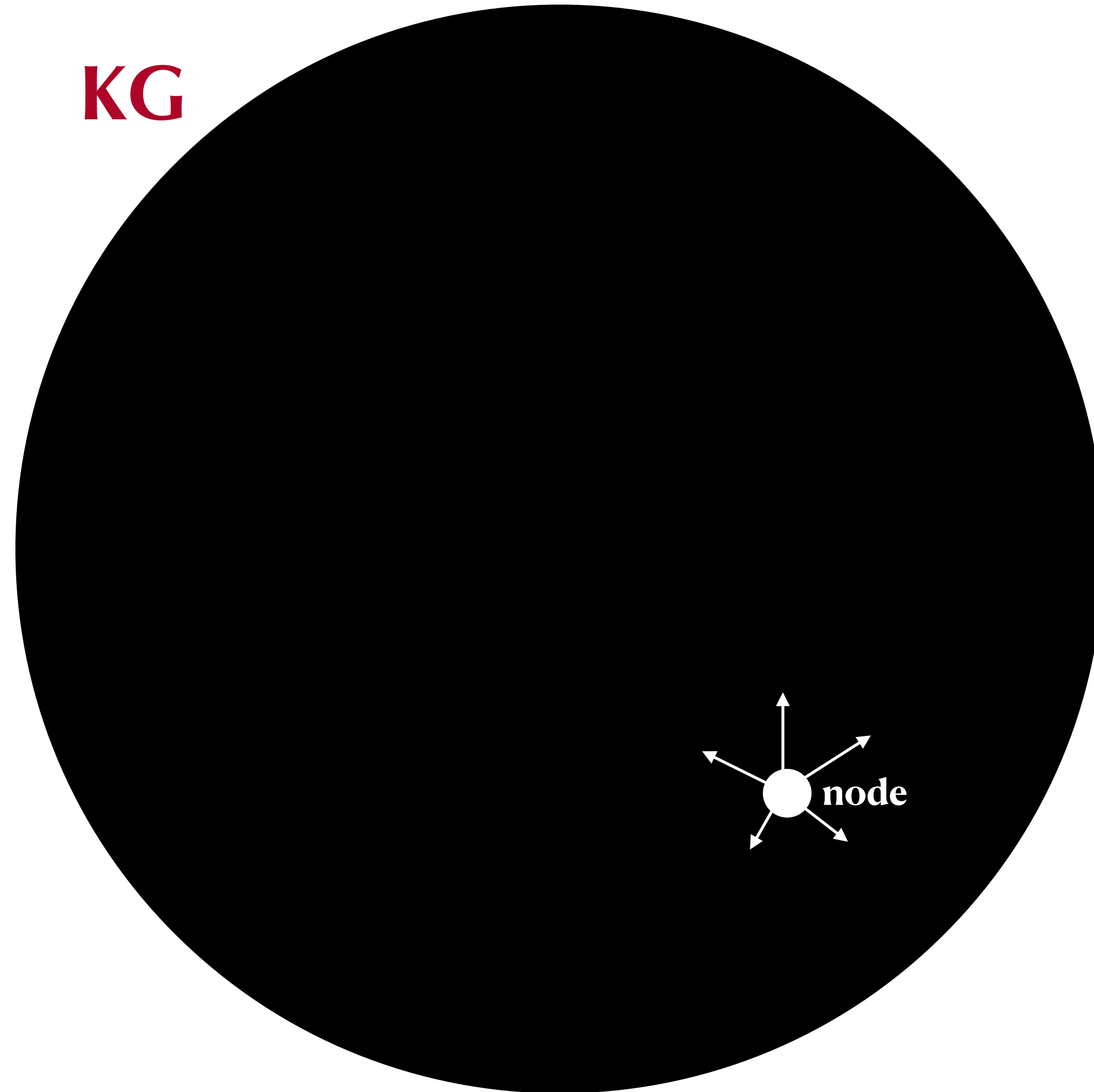


KG



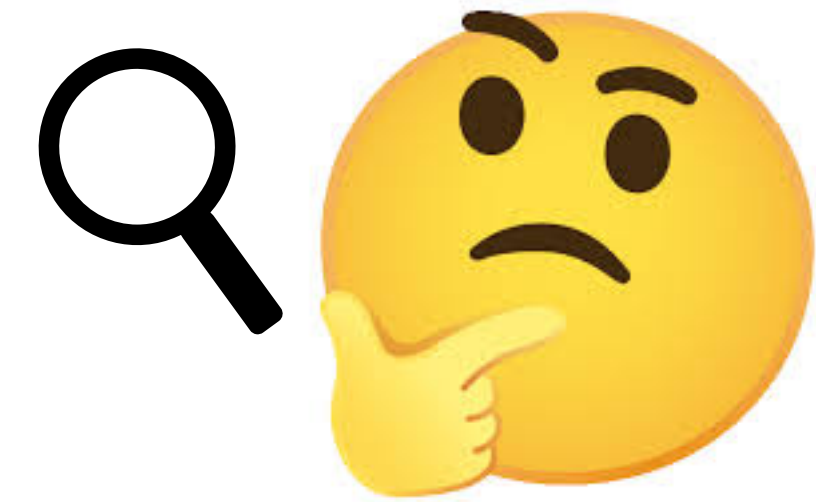
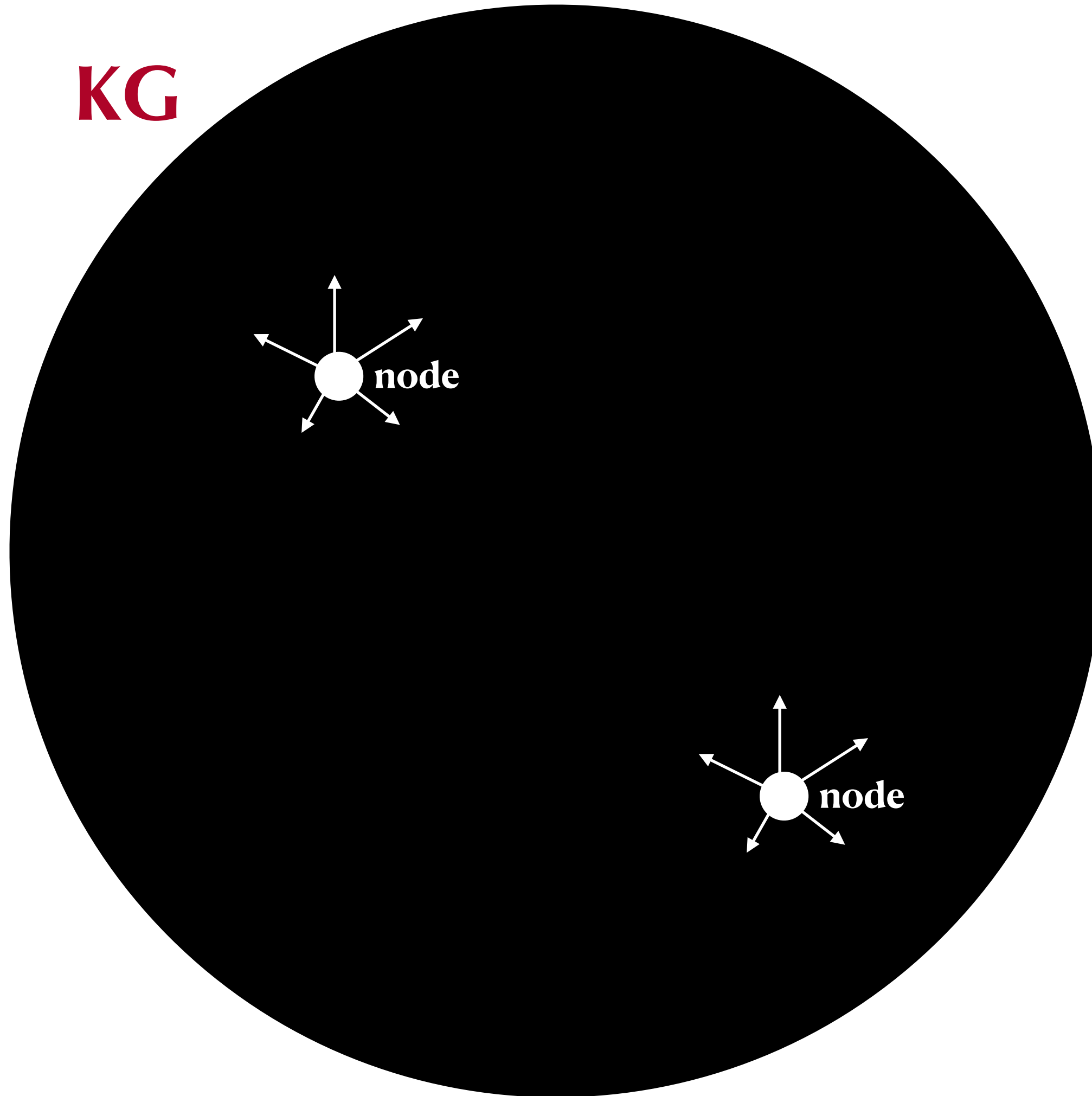
Curiosity!

KG

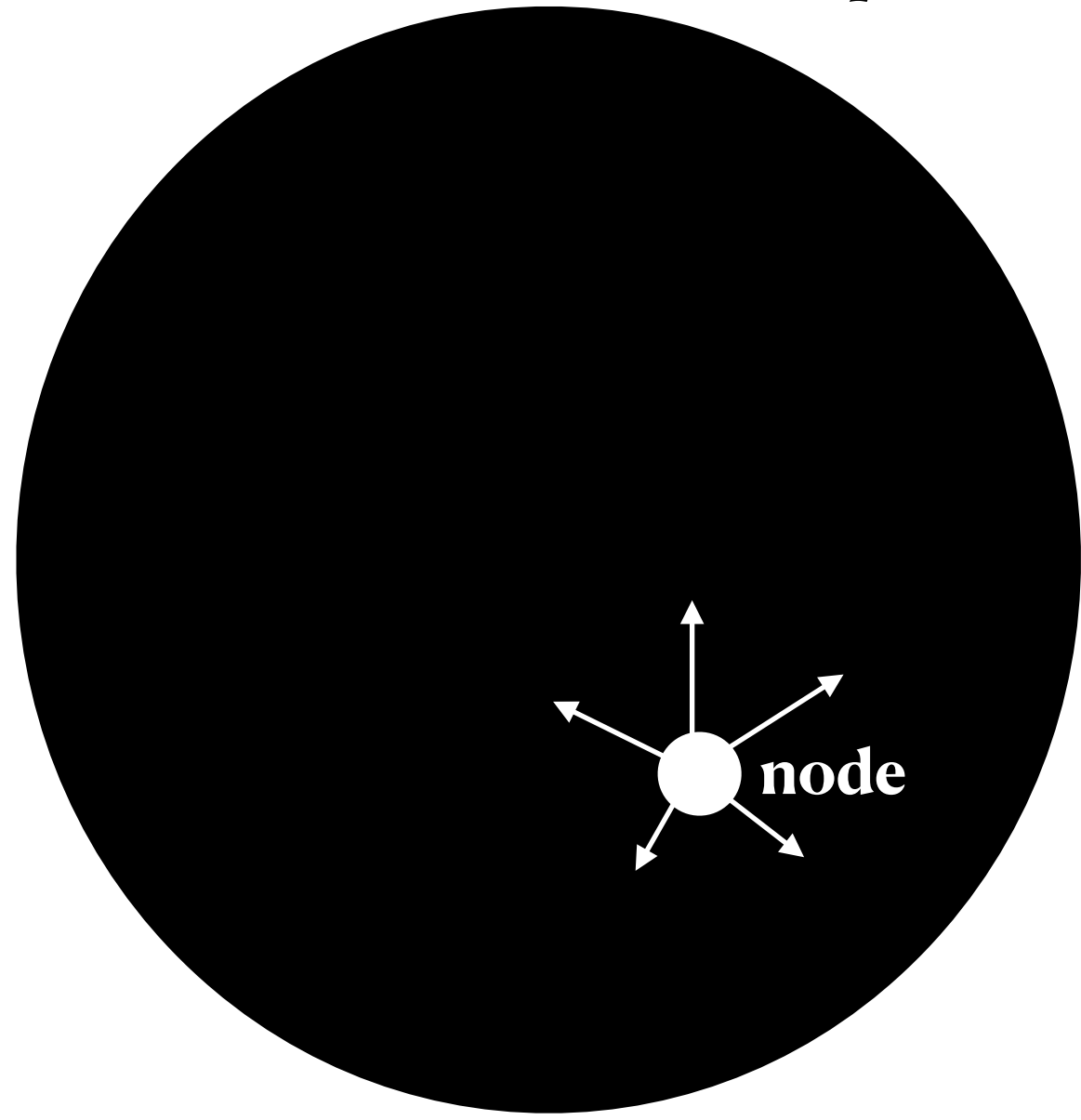


Curiosity!

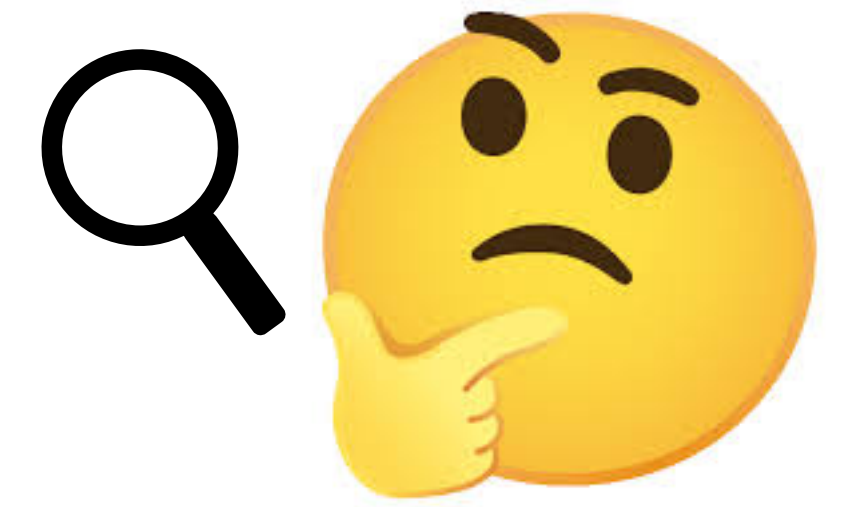
KG



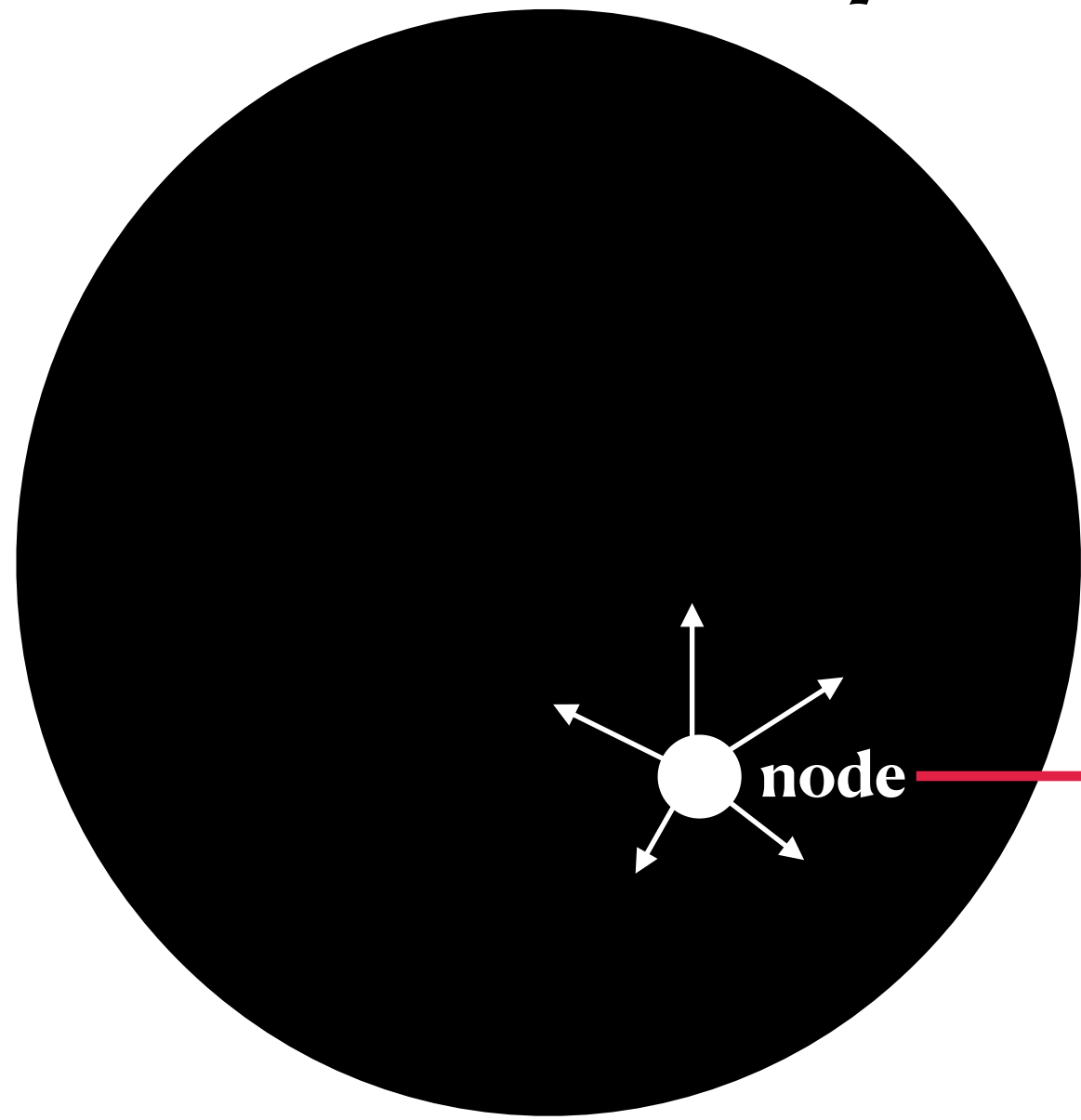
Curiosity!



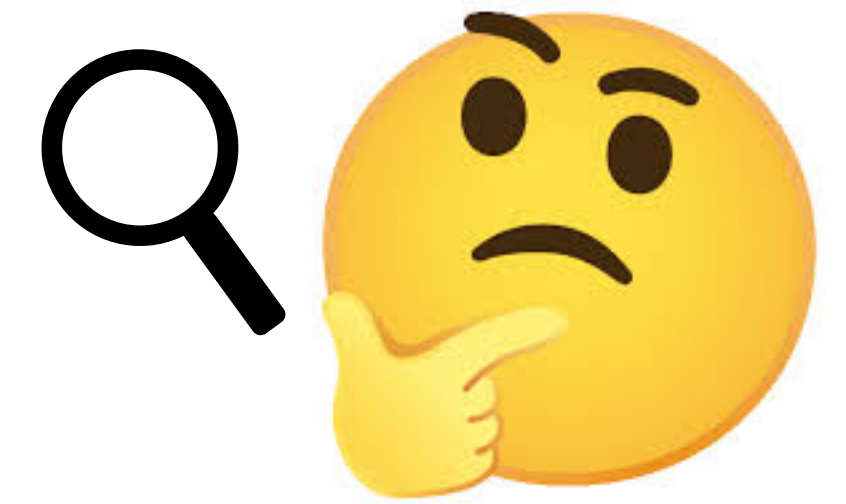
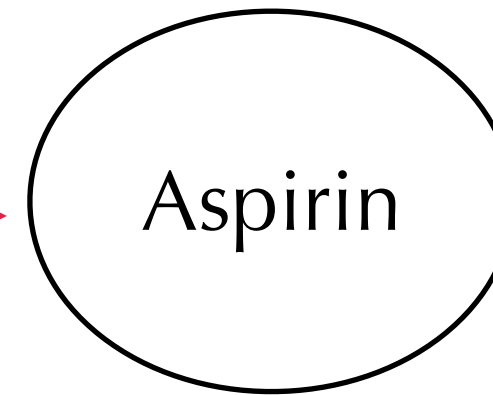
KG



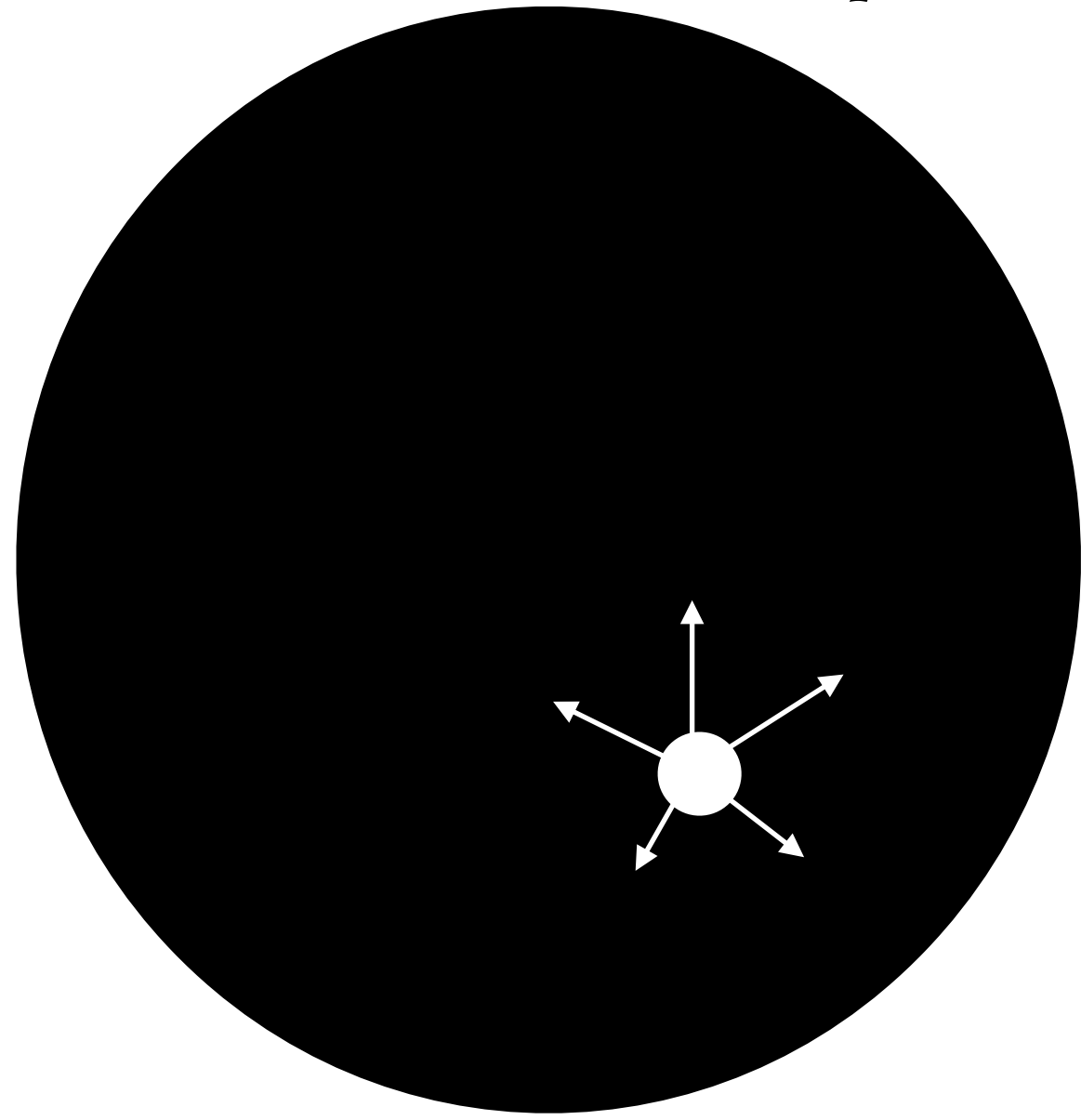
Curiosity!



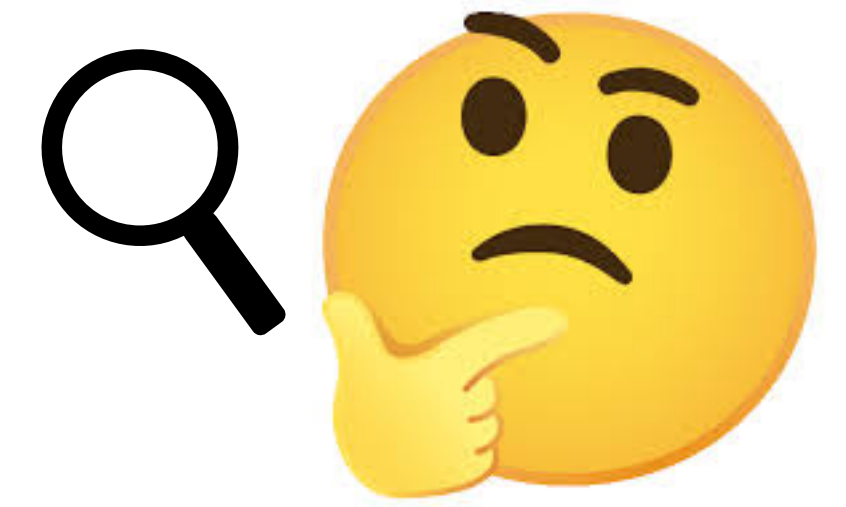
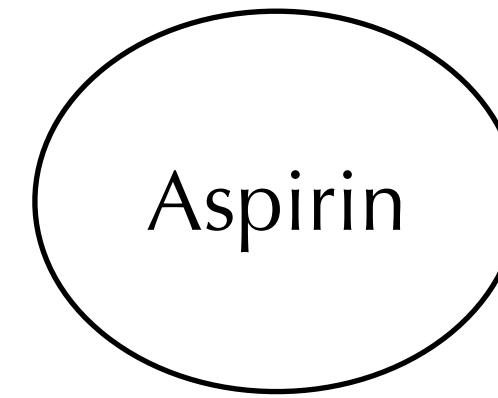
KG



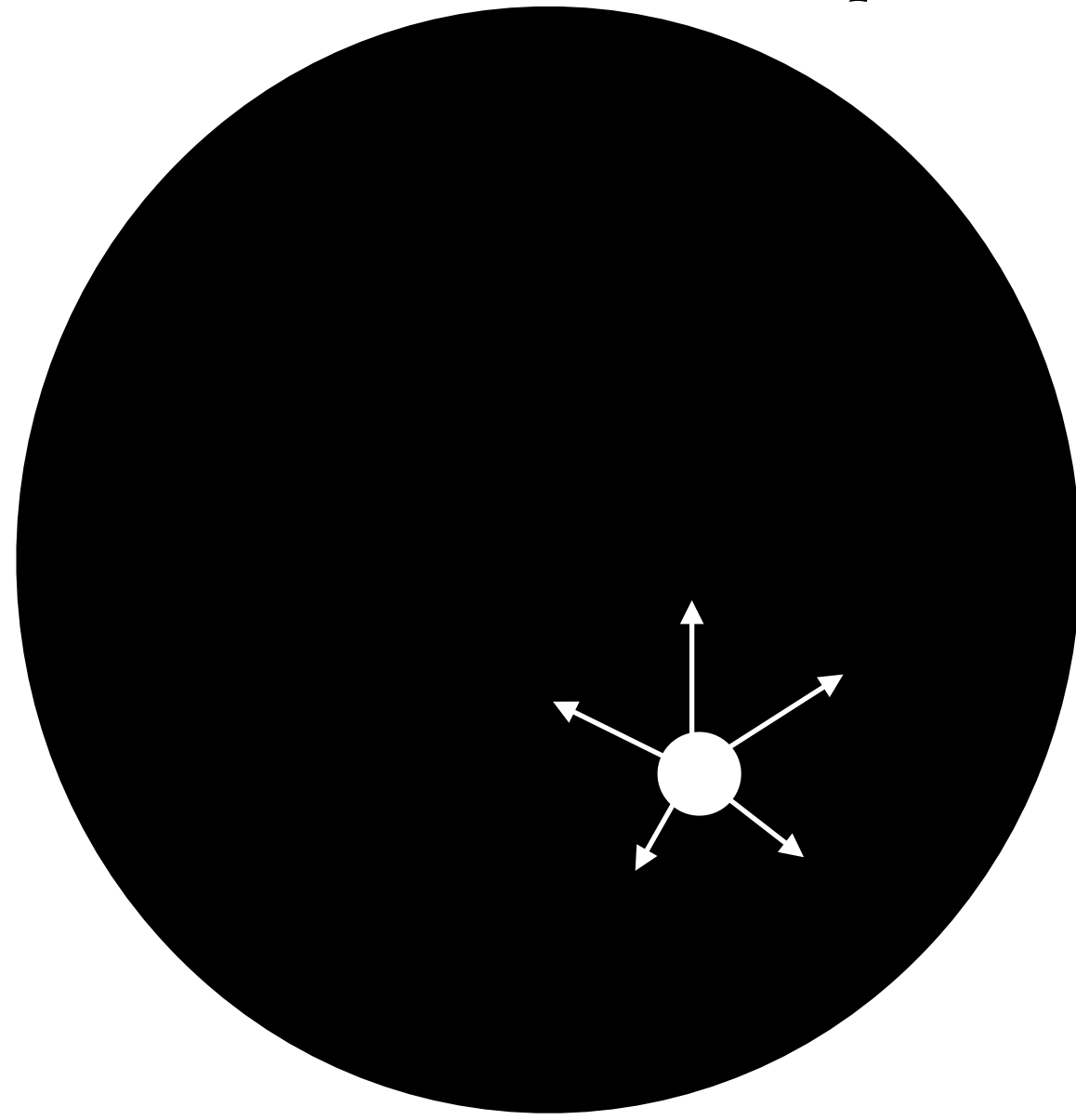
Curiosity!



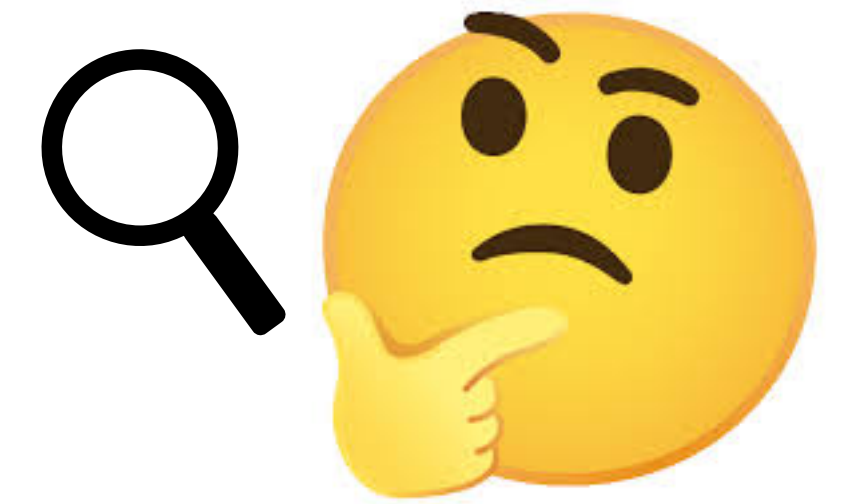
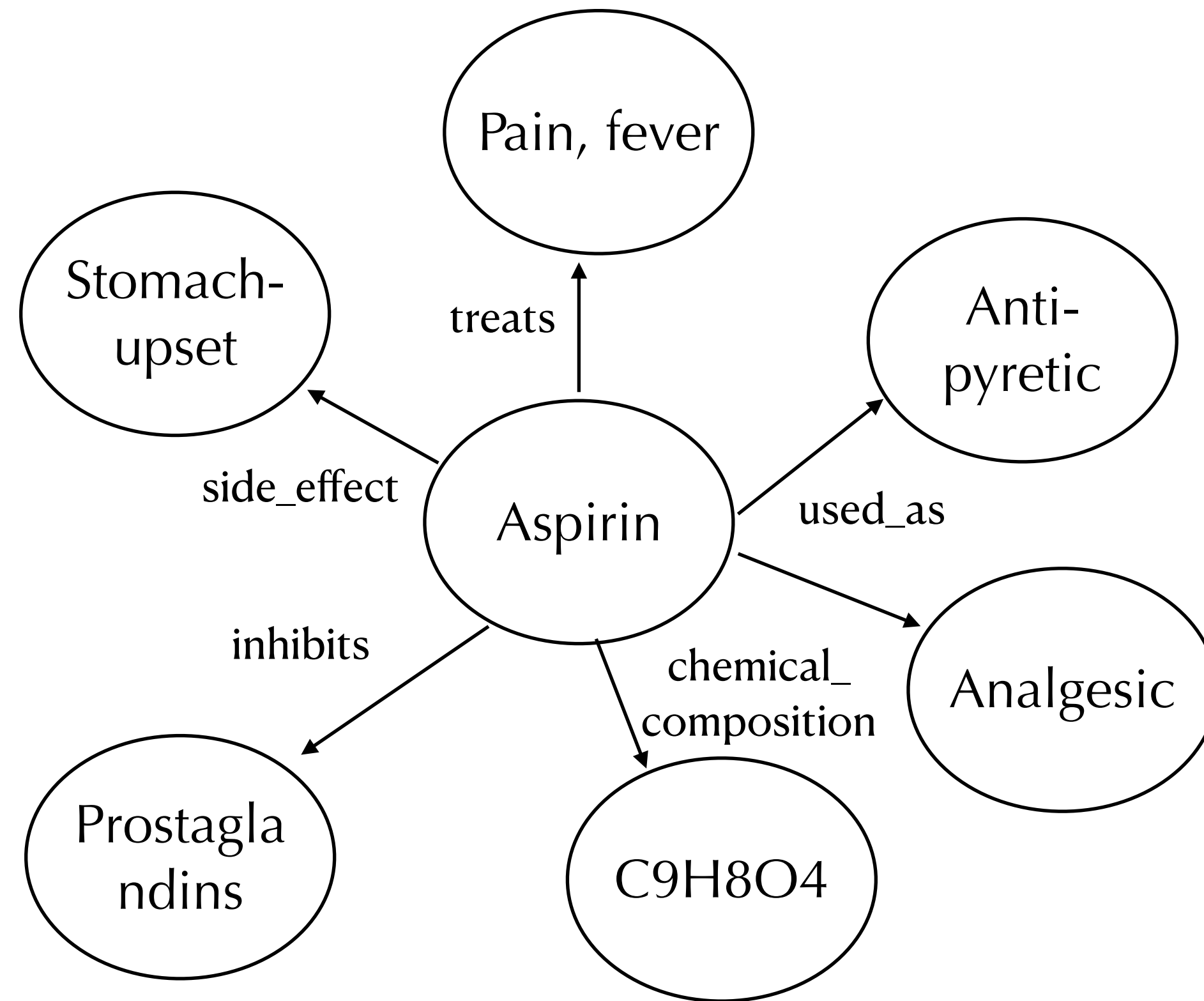
KG



Curiosity!



KG



Emerging Clarity

Emerging Clarity



Emerging Clarity

The KG has information about famous people such as their place & date of birth, spouse and their professional achievements



Emerging Clarity

The KG has information about famous people such as their place & date of birth, spouse and their professional achievements



The KG also has information about drugs, their side effects, their treatments, etc..

Emerging Clarity

The KG has information about famous people such as their place & date of birth, spouse and their professional achievements

The KG also has information about drugs, their side effects, their treatments, etc..

Therefore, the KG can answer questions like...



Emerging Clarity

The KG has information about famous people such as their place & date of birth, spouse and their professional achievements

The KG also has information about drugs, their side effects, their treatments, etc..



Therefore, the KG can answer questions like...

- ◆ Which year was Person X born?
- ◆ What is X's age?
- ◆ How many books X has written?
- ◆ Who is X married to?

Emerging Clarity

The KG has information about famous people such as their place & date of birth, spouse and their professional achievements

The KG also has information about drugs, their side effects, their treatments, etc..

Therefore, the KG can answer questions like...



- ◆ Which year was Person X born?
- ◆ What is X's age?
- ◆ How many books X has written?
- ◆ Who is X married to?

- ◆ What is the chemical composition of Drug A?
- ◆ What are the side effects of A?
- ◆ How many distinct diseases can be treated by A?

Emerging Clarity

The KG has information about famous people such as their place & date of birth, spouse and their professional achievements

The KG also has information about drugs, their side effects, their treatments, etc..



Therefore, the KG can answer questions like...

◆ Which year was Person X born?

◆ What is the chemical composition of Drug A?

Can we mechanize this human behavior?

◆ Who is X married to?

◆ How many distinct diseases can be treated by A?

Outline: Rest of the talk

- ◆ Task Formalization
- ◆ BYOKG Approach
- ◆ Stage 1: Exploration
- ◆ Stage 2: Question Generation
- ◆ Stage 3: Reasoning
- ◆ Results
- ◆ Future directions

Task: KGQA (Program Synthesis)

Given **KG**: $\mathcal{K} \subseteq \mathcal{E} \times \mathcal{R} \times (\mathcal{E} \cup \mathcal{L} \cup \mathcal{C})$

Find answer set \mathcal{A}_q for a natural language query q
by mapping q to a program p_q

s.t. $\text{eval}^{\mathcal{K}}(p_q) = \mathcal{A}_q$

Task: KGQA (Program Synthesis)

Example:

Q Who are the sponsors of the Stanford Medicine X conference series?

P_q `(AND conferences.conference_sponsor (JOIN
conferences.conference_sponsor.conferences
m.0j2fyjs))`

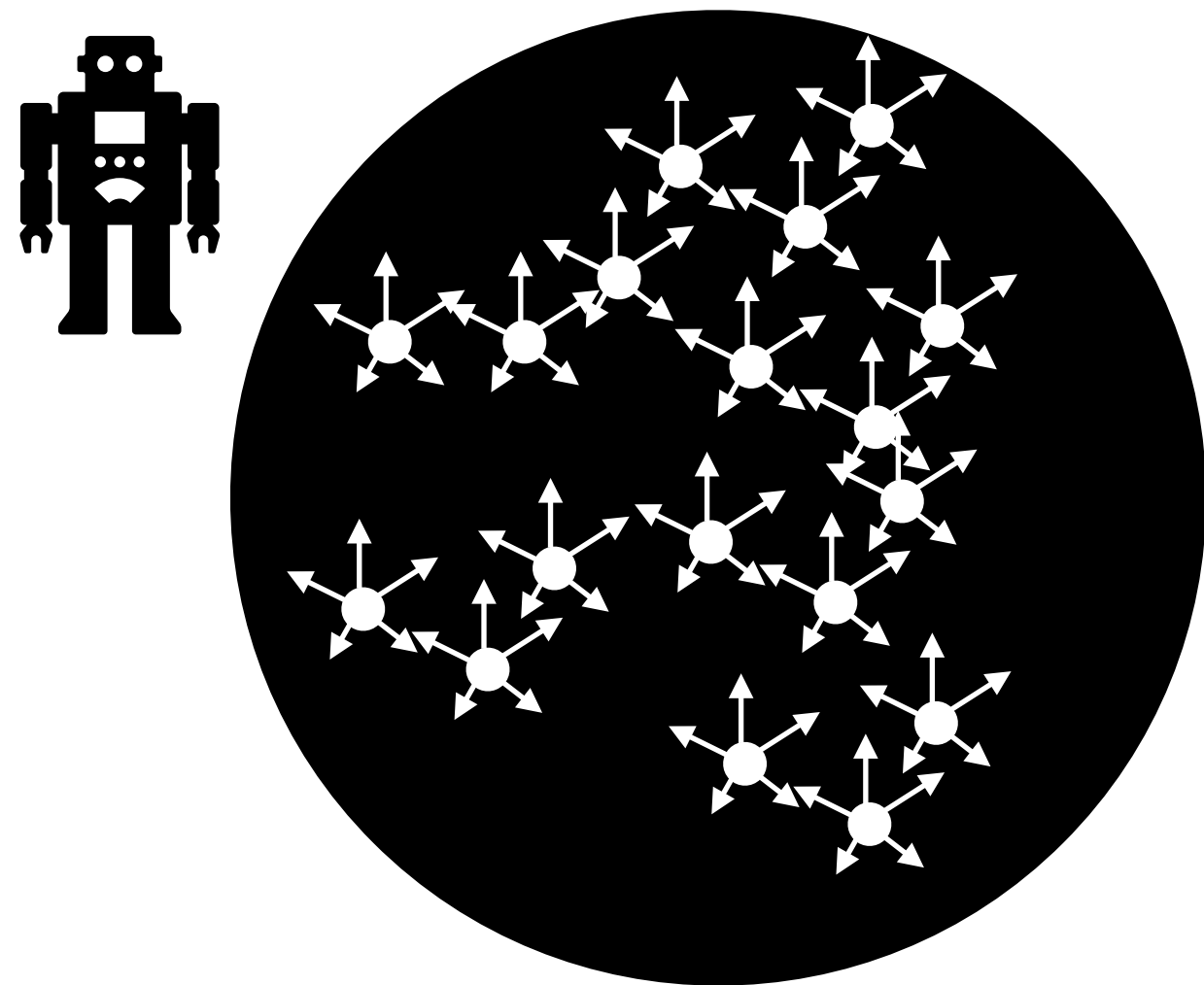
A_q `{m.0c1d2_9 (Stanford Anesthesia),
m.02rkyb4 (Stanford Hospital & Clinics)}`

 **S-expression**

BYOKG: Approach

BYOKG: Approach

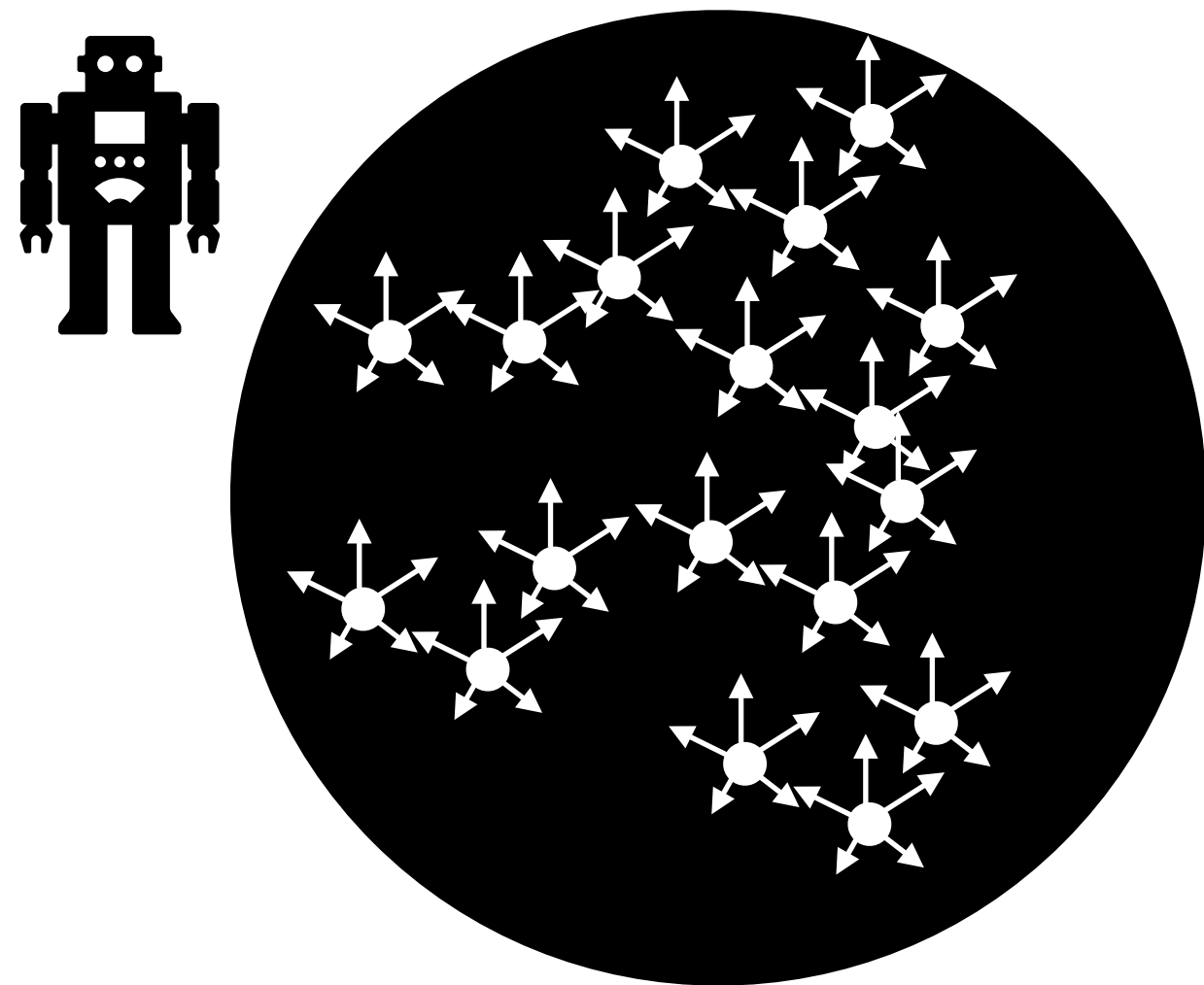
Stage 1: Symbolic Graph Exploration



$$\mathcal{X}^P := \{p_i\}$$

BYOKG: Approach

Stage 1: Symbolic Graph Exploration

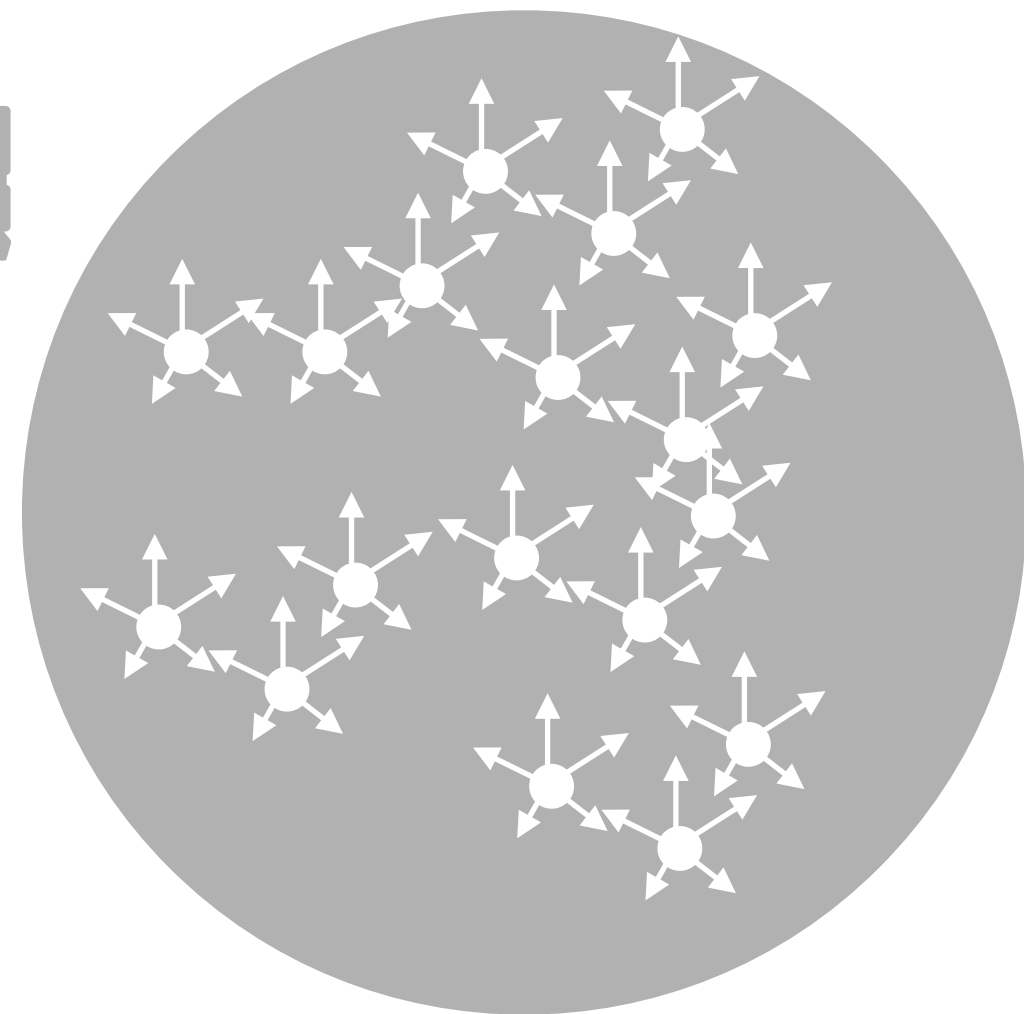
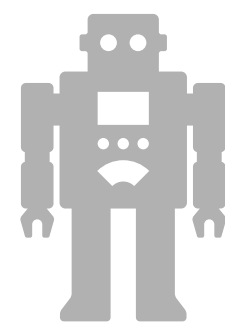


$$\mathcal{X}^P := \{p_i\}$$

BYOKG: Approach

Stage 1:

Symbolic Graph Exploration



$$\mathcal{X}^P := \{p_i\}$$

Stage 2:

NL Query Generation

`(JOIN radio.radio...m.01mxcd7)`

LLM: What is the...?

`(COUNT (AND computer.des...))`

LLM: How many...?

...

`(ARGMAX (AND film.dire...))`

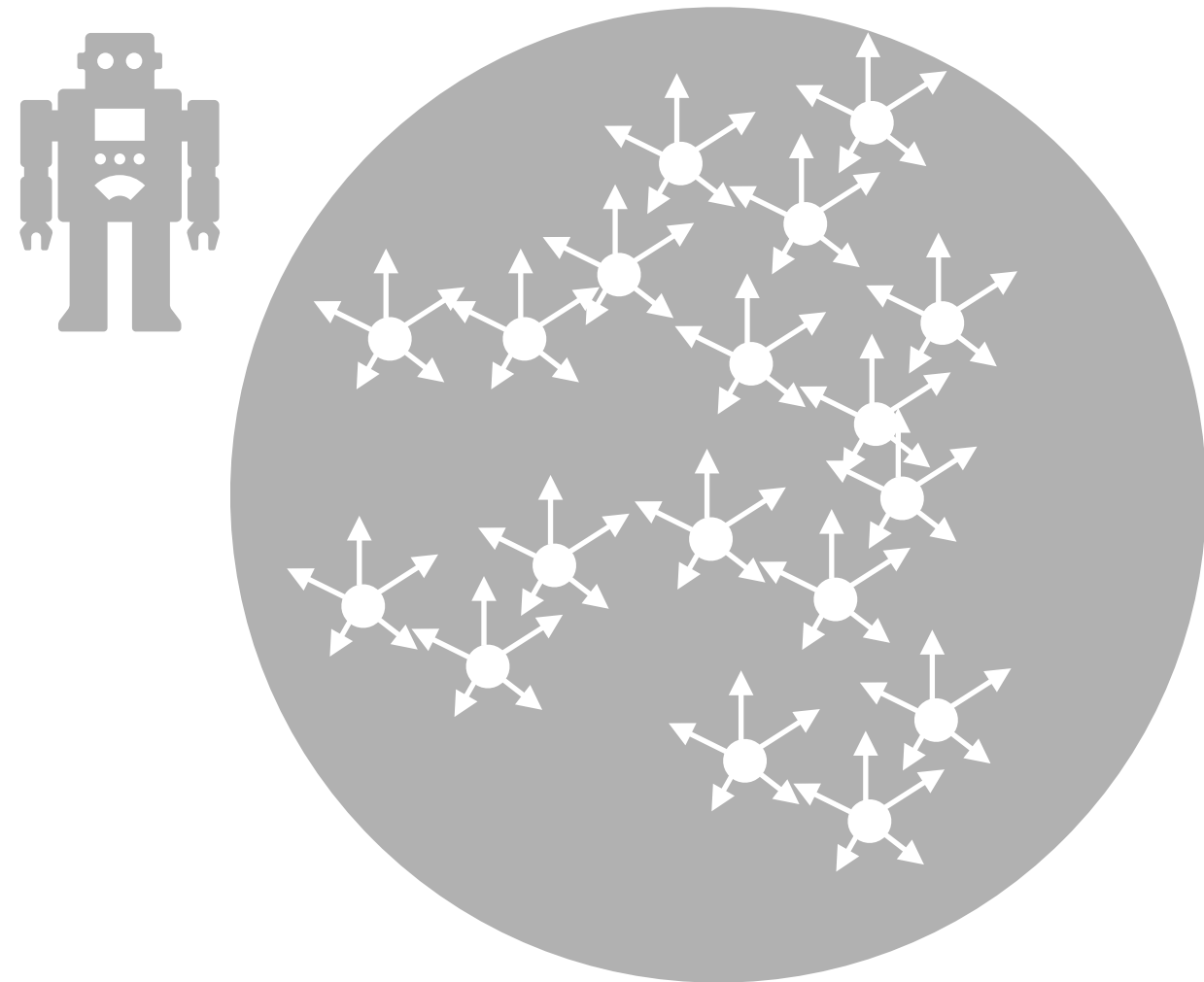
LLM: Who is the most...?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\}$$

BYOKG: Approach

Stage 1:

Symbolic Graph Exploration



$$\mathcal{X}^P := \{p_i\}$$

Stage 2:

NL Query Generation

1. L2M prompting

2. Inverse-Consistency

(JOB (AND computer.des...))
LLM: what is the...^{d7)}

(COUNT (AND computer.des...))

LLM: How many...?

...

(ARGMAX (AND film.dire...))

LLM: Who is the most...?

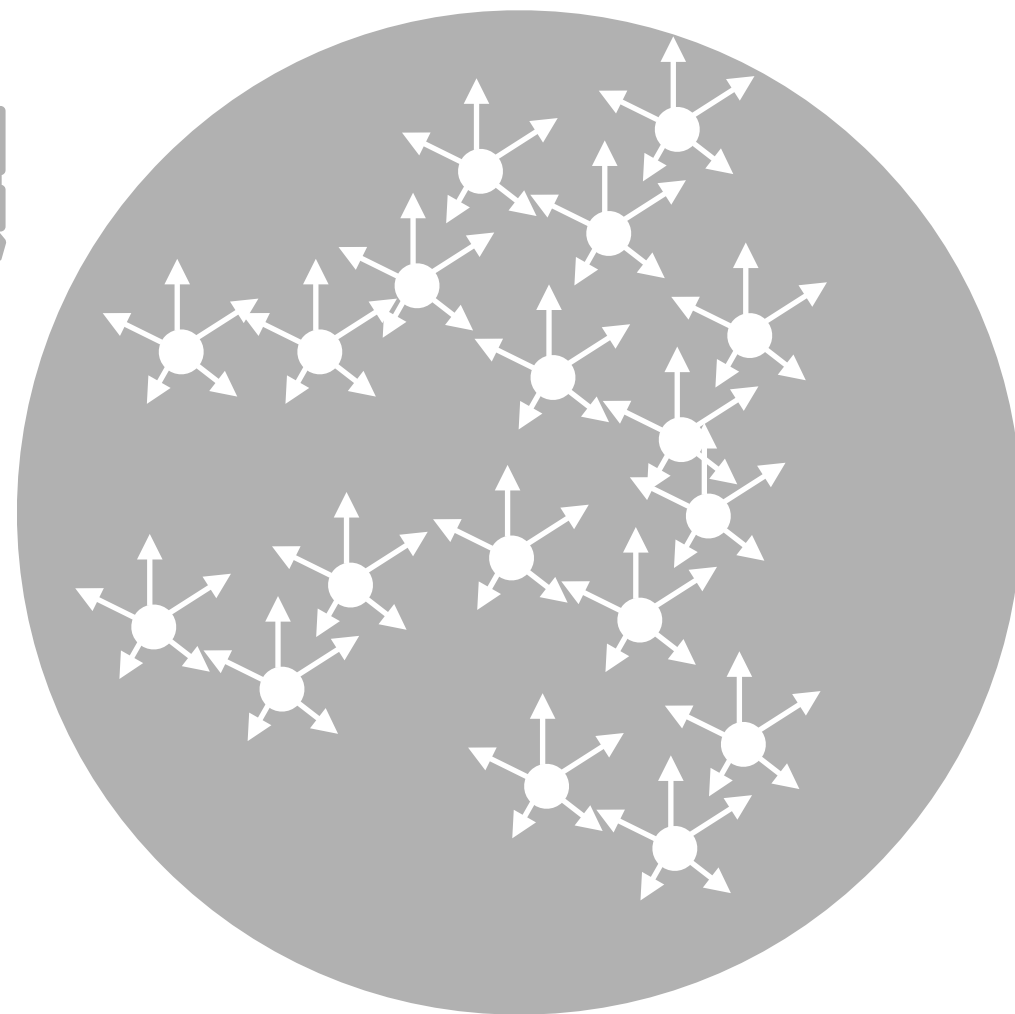
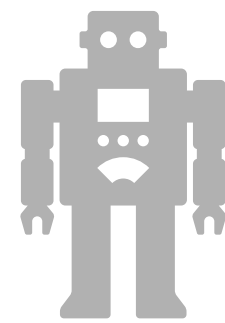
$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\}$$

BYOKG: Approach

Required to be done once for a KG

Stage 1:

Symbolic Graph Exploration



$$\mathcal{X}^P := \{p_i\}$$

Stage 2:

NL Query Generation

1. L2M prompting

2. Inverse-Consistency

(JOB (AND computer.designer))
LLM: what is the...

(COUNT (AND computer.designer))

LLM: How many...?

...

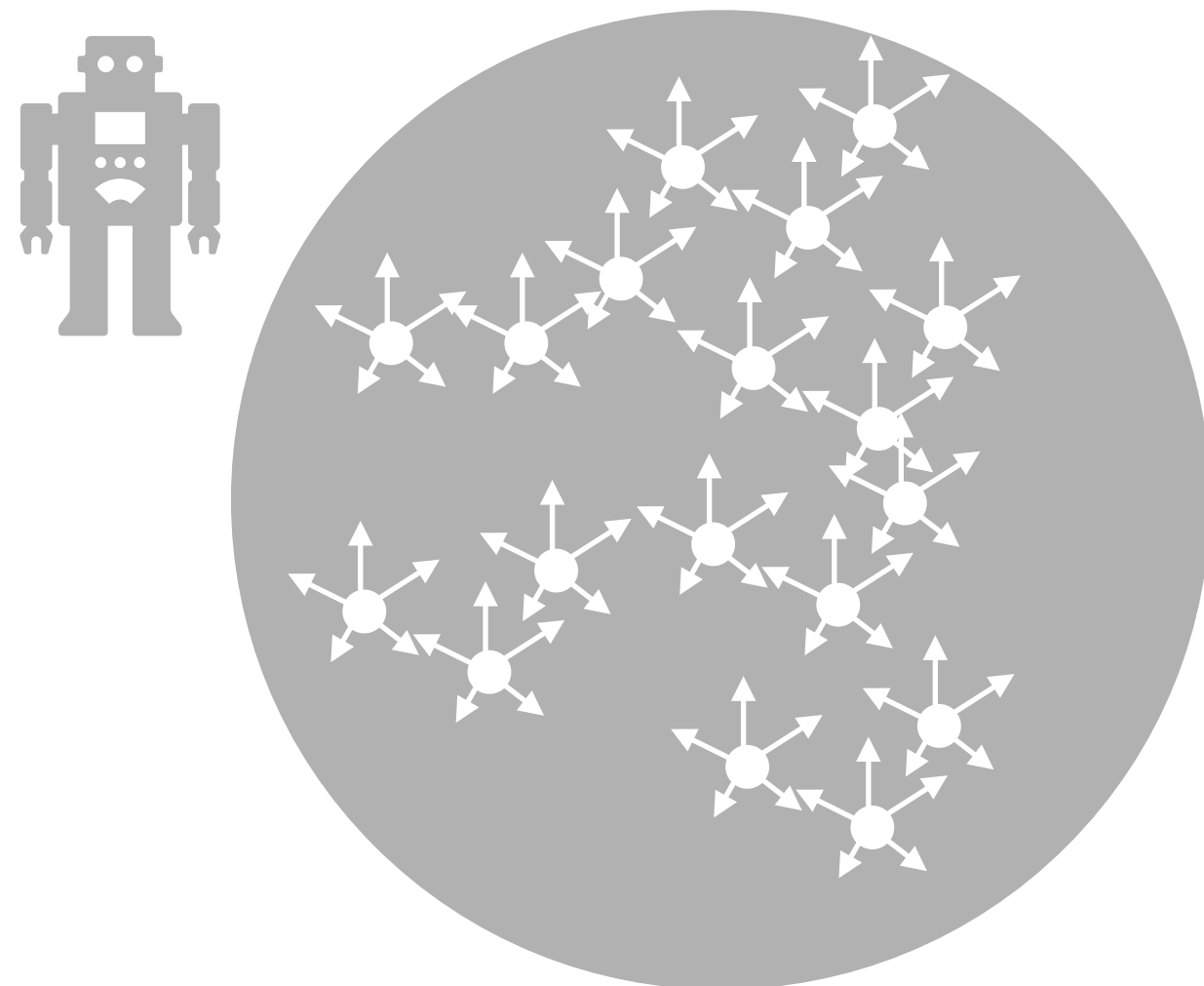
(ARGMAX (AND film.director))

LLM: Who is the most...?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\}$$

BYOKG: Approach

Stage 1: Symbolic Graph Exploration



$$\mathcal{X}^P := \{p_i\}$$

Stage 2: NL Query Generation

1. L2M prompting
2. Inverse-Consistency

(JOIN ...)
LLM: what is the...
(COUNT (AND computer.des...))
LLM: How many...?
...
(ARGMAX (AND film.dire...))
LLM: Who is the most...?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\}$$

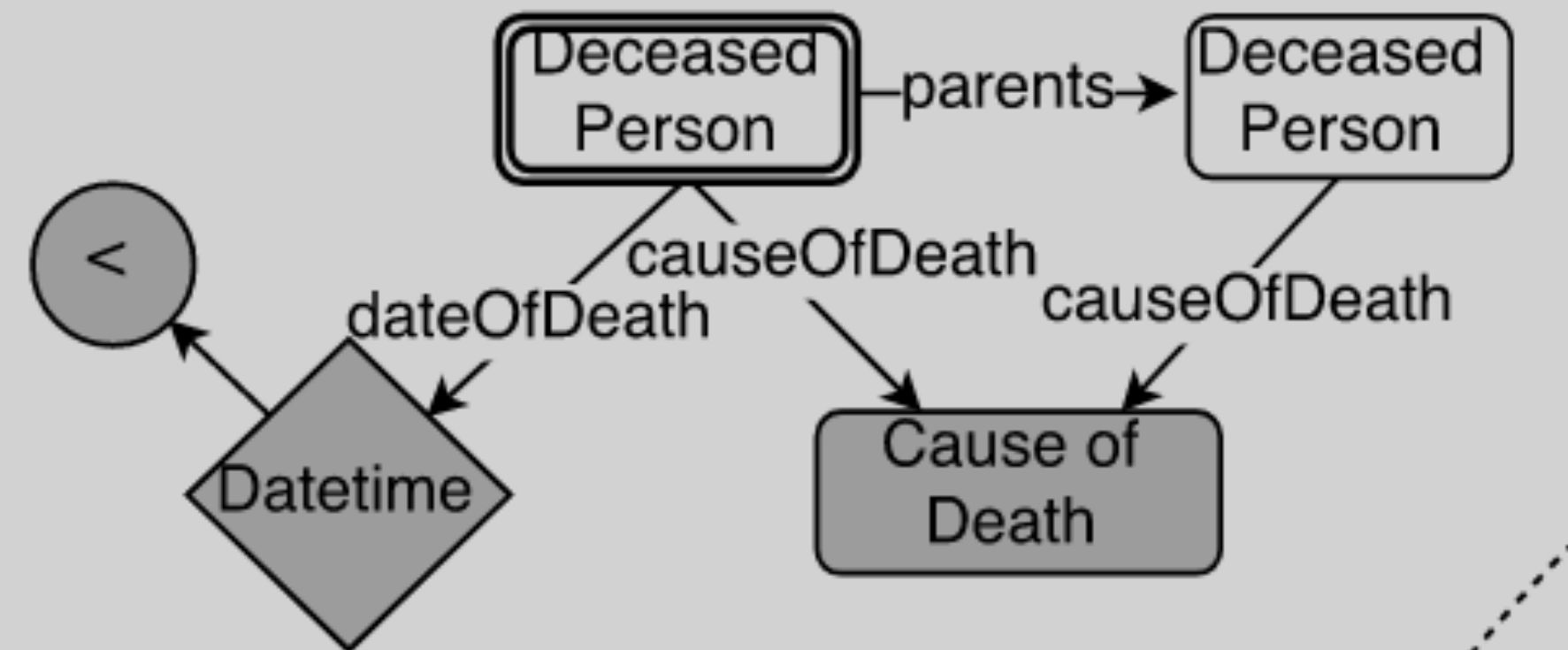
Stage 3: Bottom-up Reasoning

Q: What is the name of the tallest art director whose profession is director of photography?

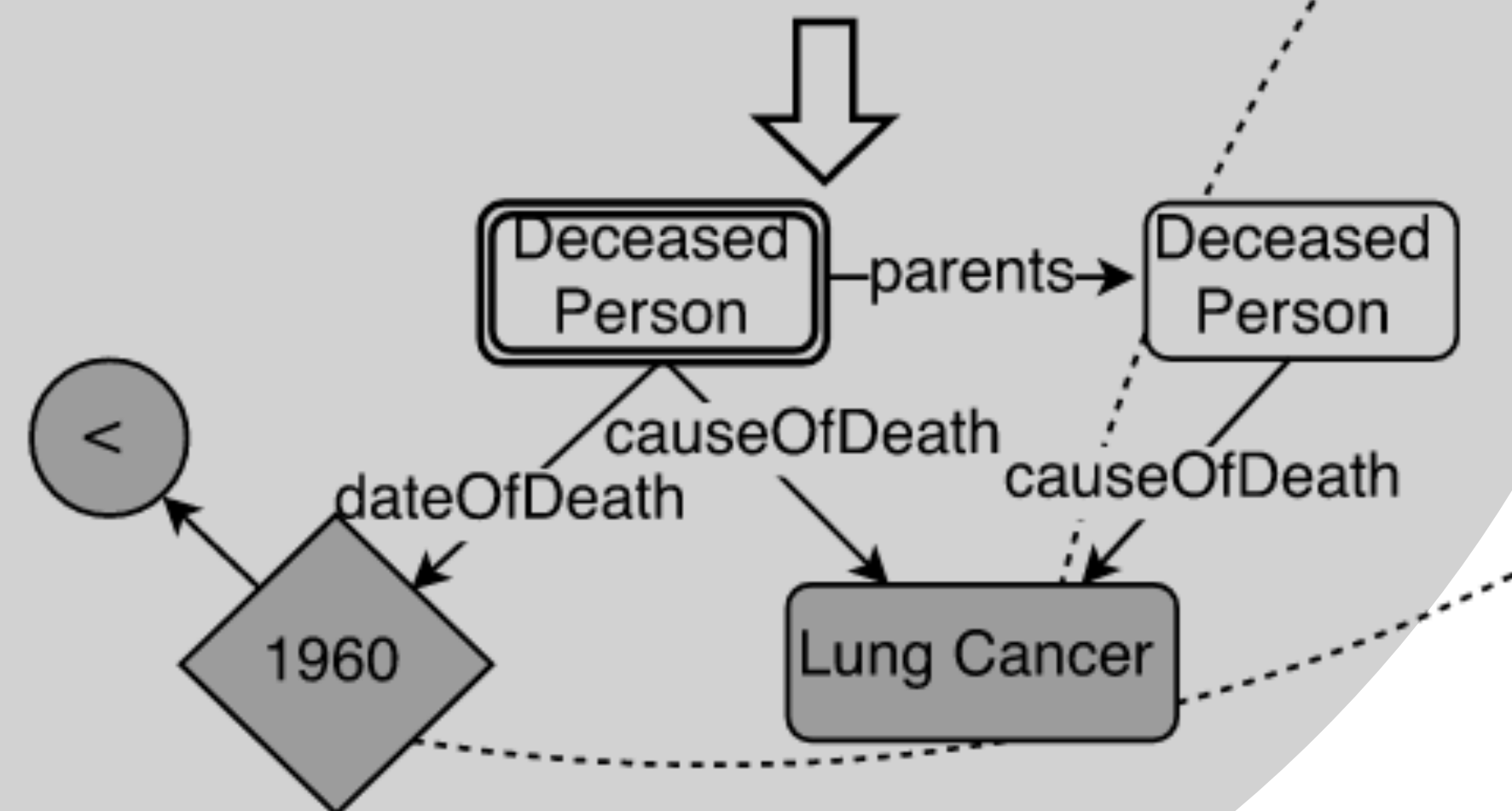


p1: m.0dgd_
↓
p2: (JOIN profession **p1**)
↓
p3: (AND art_director **p2**)
↓
p4: (ARGMAX **p3** person.height)

Stage 1: Symbolic Graph Exploration

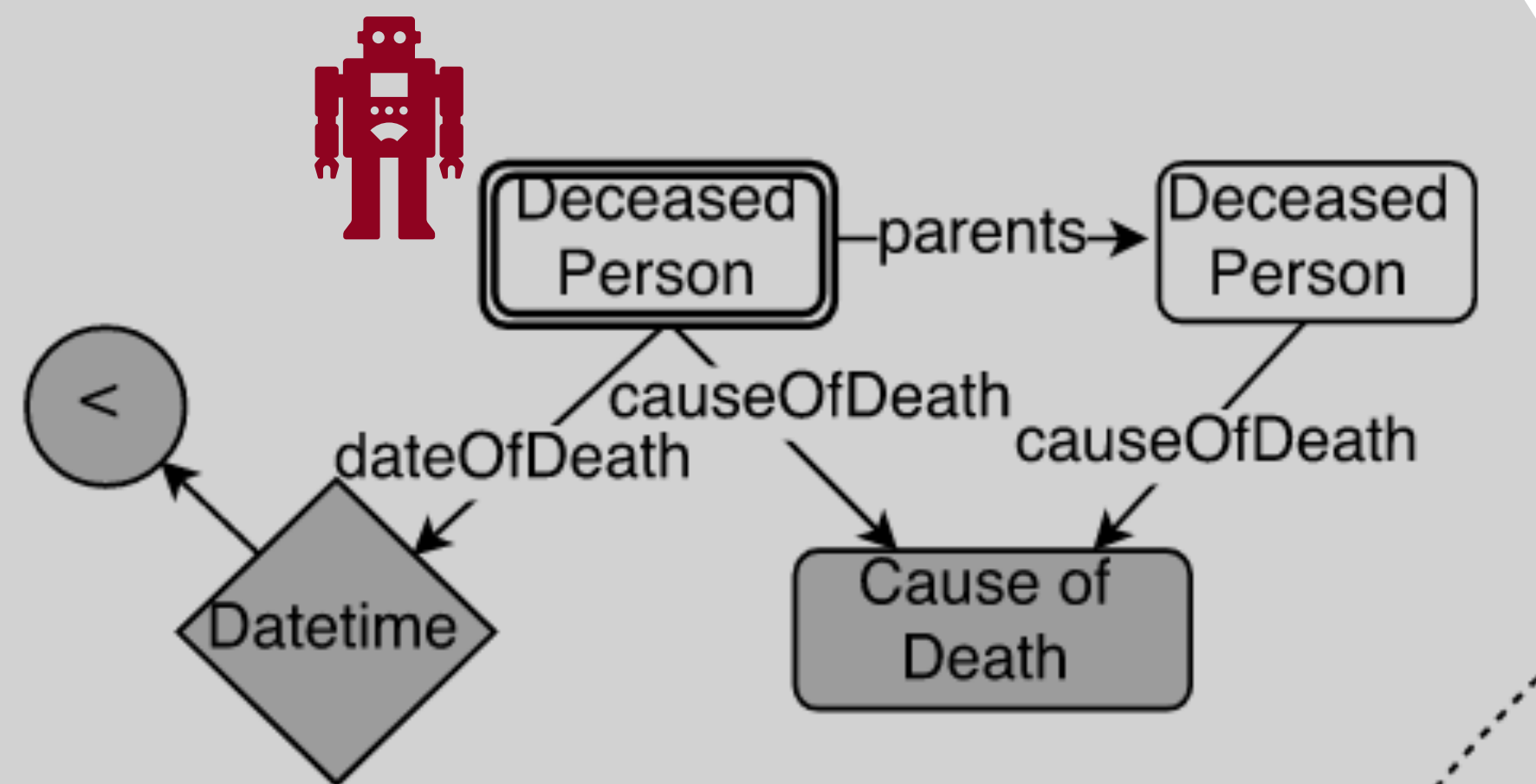


(b) Query template

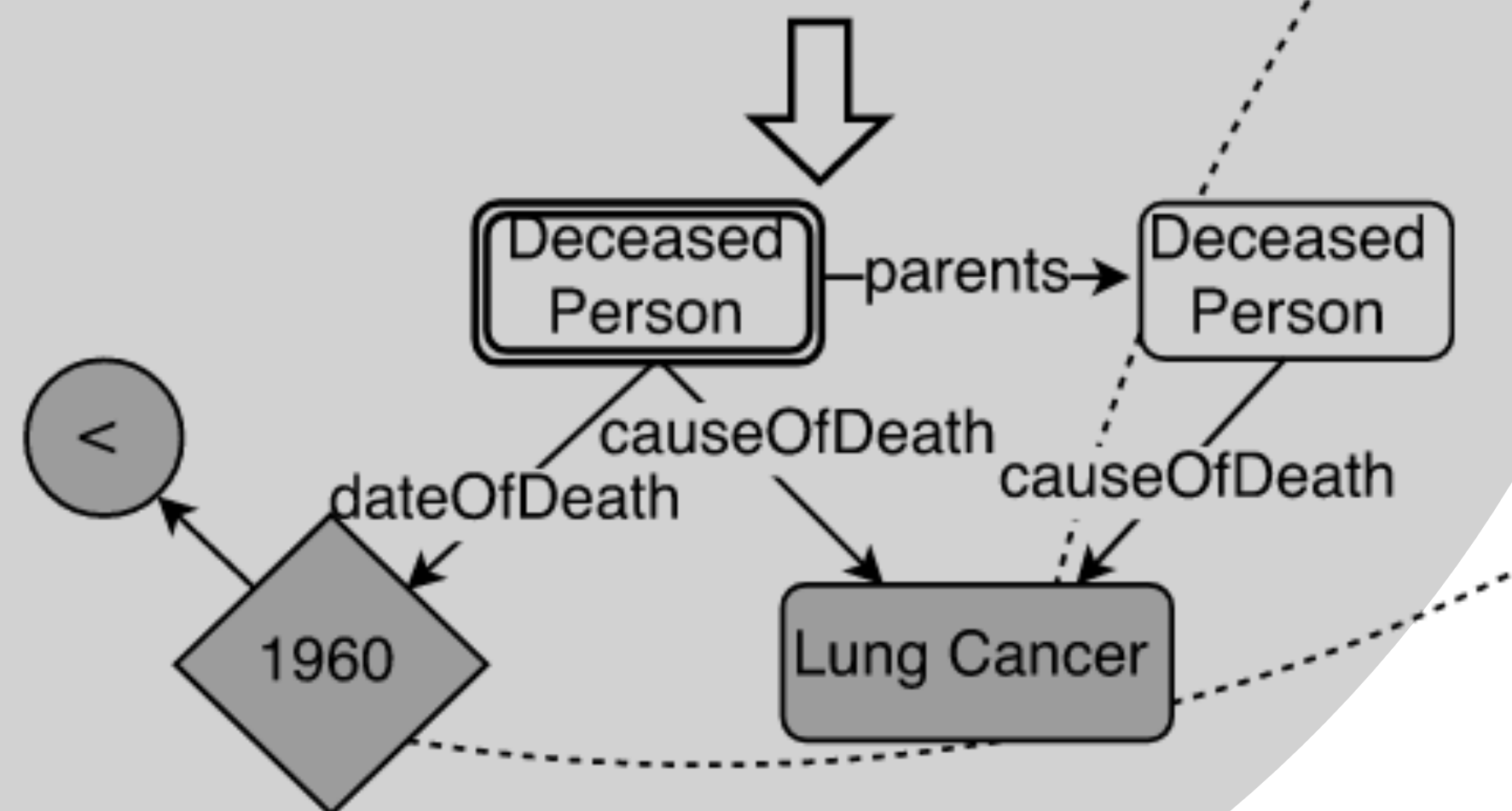


(c) Graph query

Stage 1: Symbolic Graph Exploration



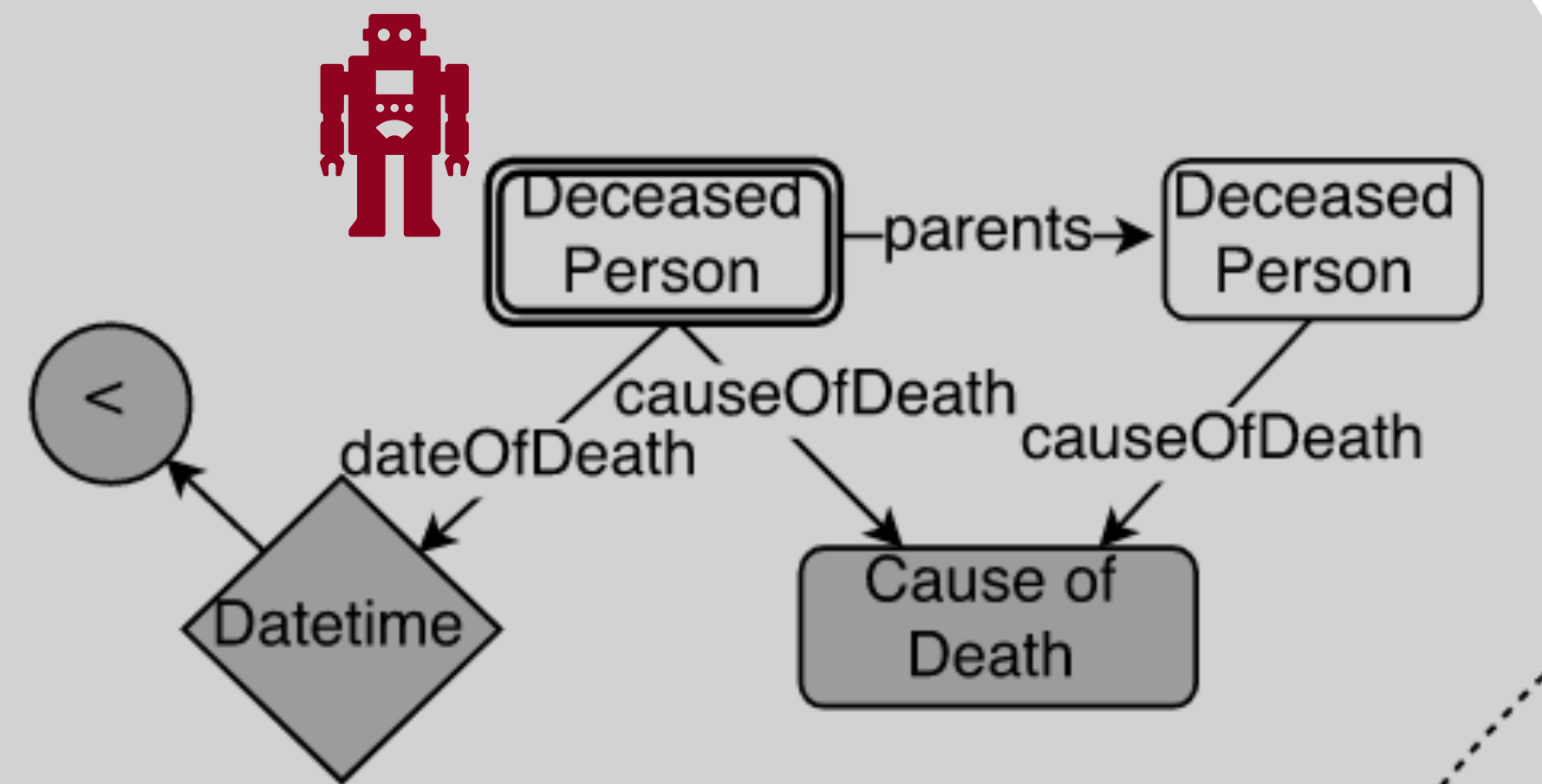
(b) Query template



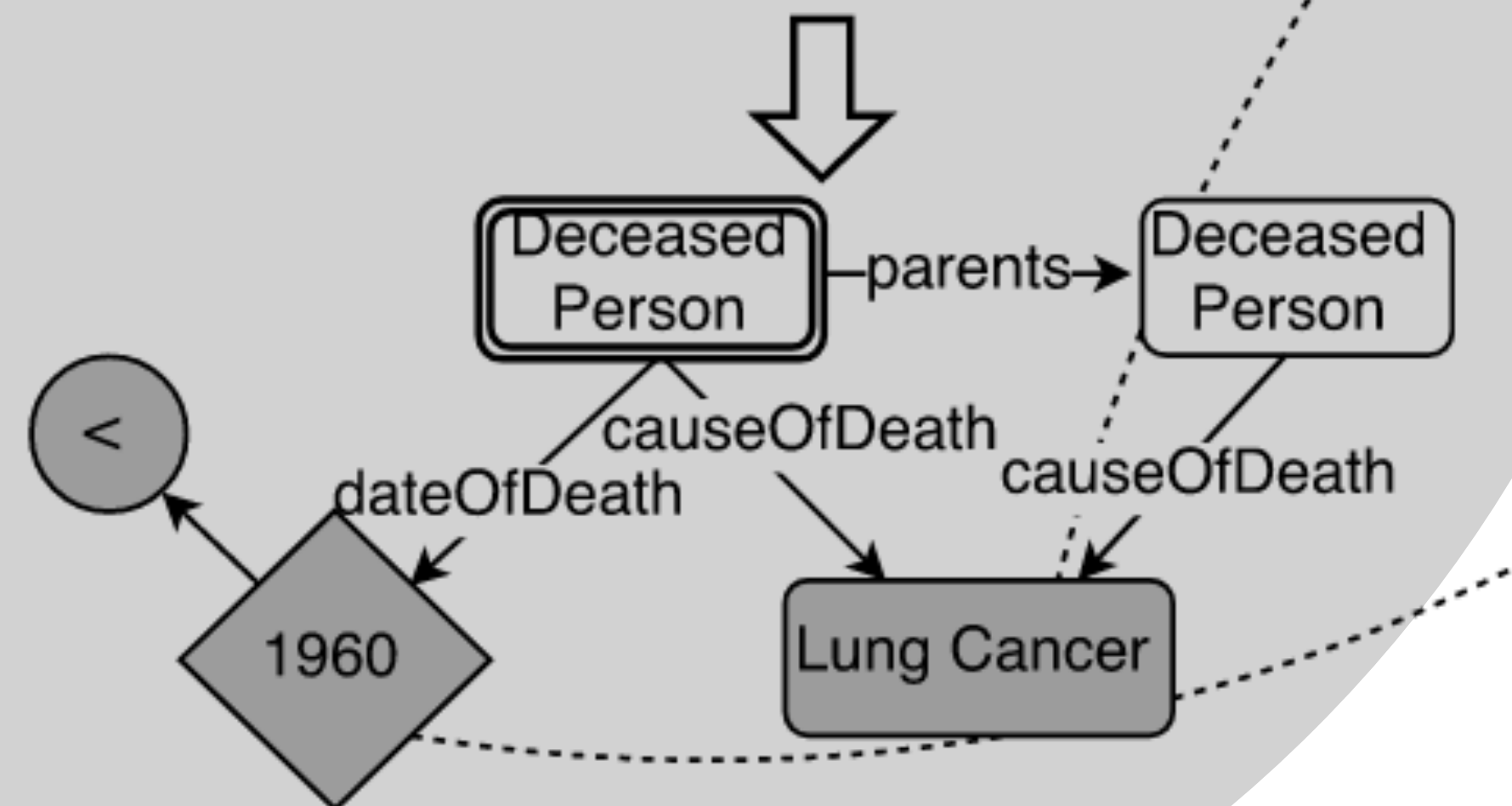
(c) Graph query

$$p_0 := c_0 \sim \mathcal{C}$$

Stage 1: Symbolic Graph Exploration



(b) Query template

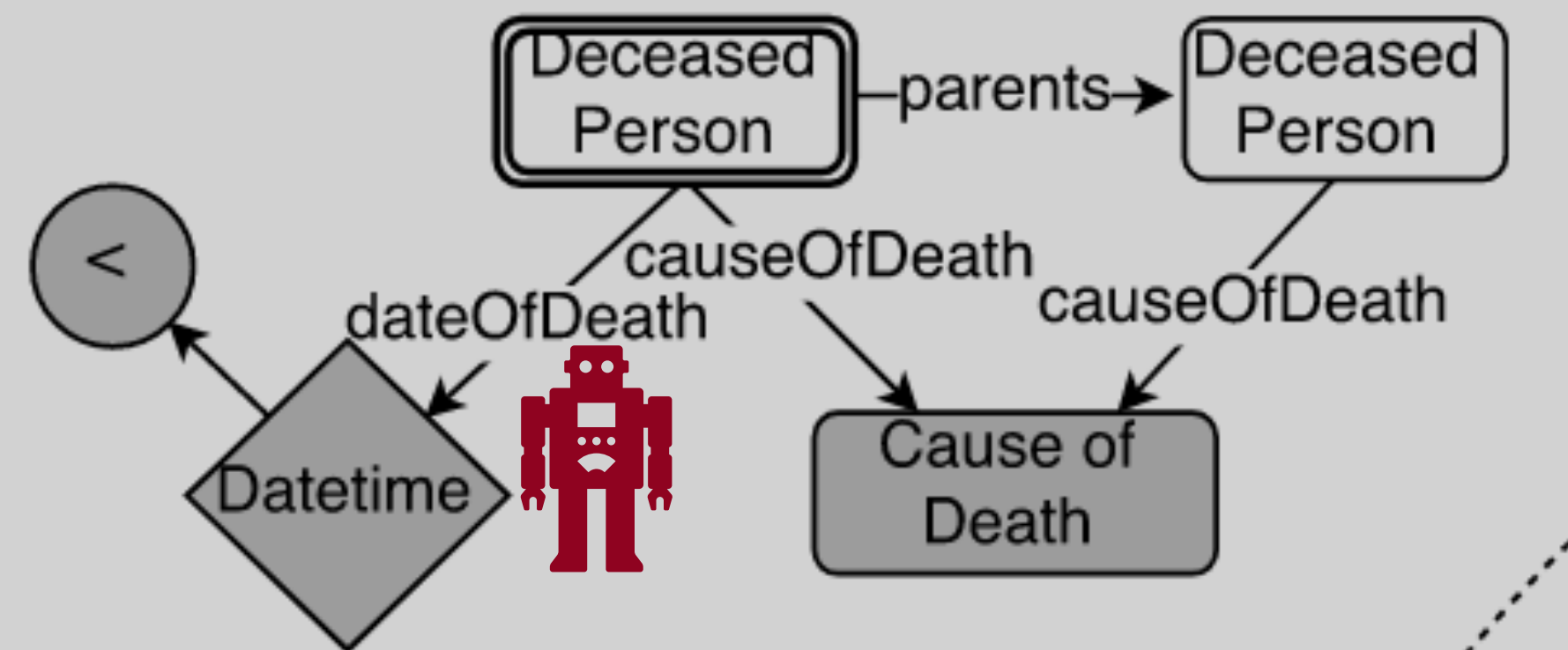


(c) Graph query

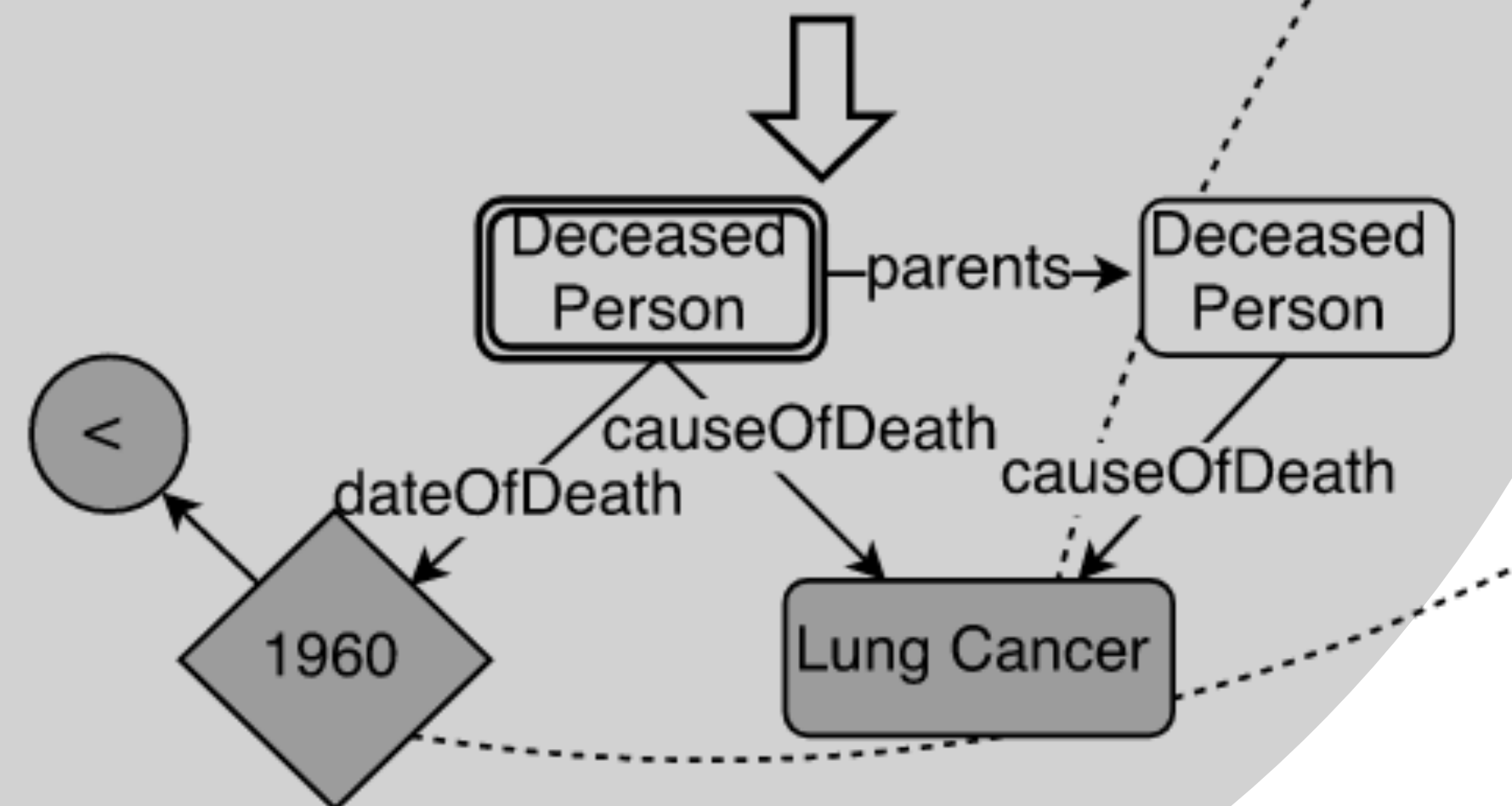
$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

Stage 1: Symbolic Graph Exploration



(b) Query template



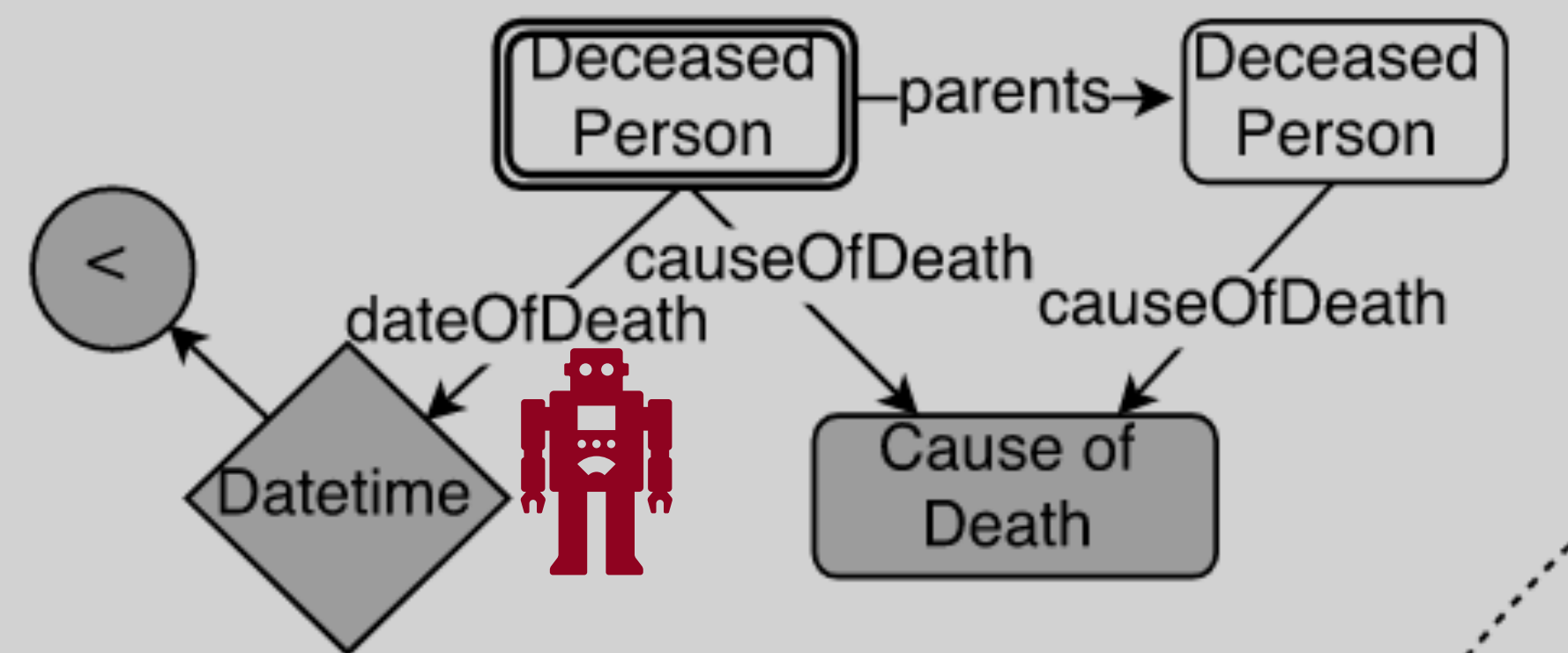
(c) Graph query

$$p_0 := c_0 \sim \mathcal{C}$$

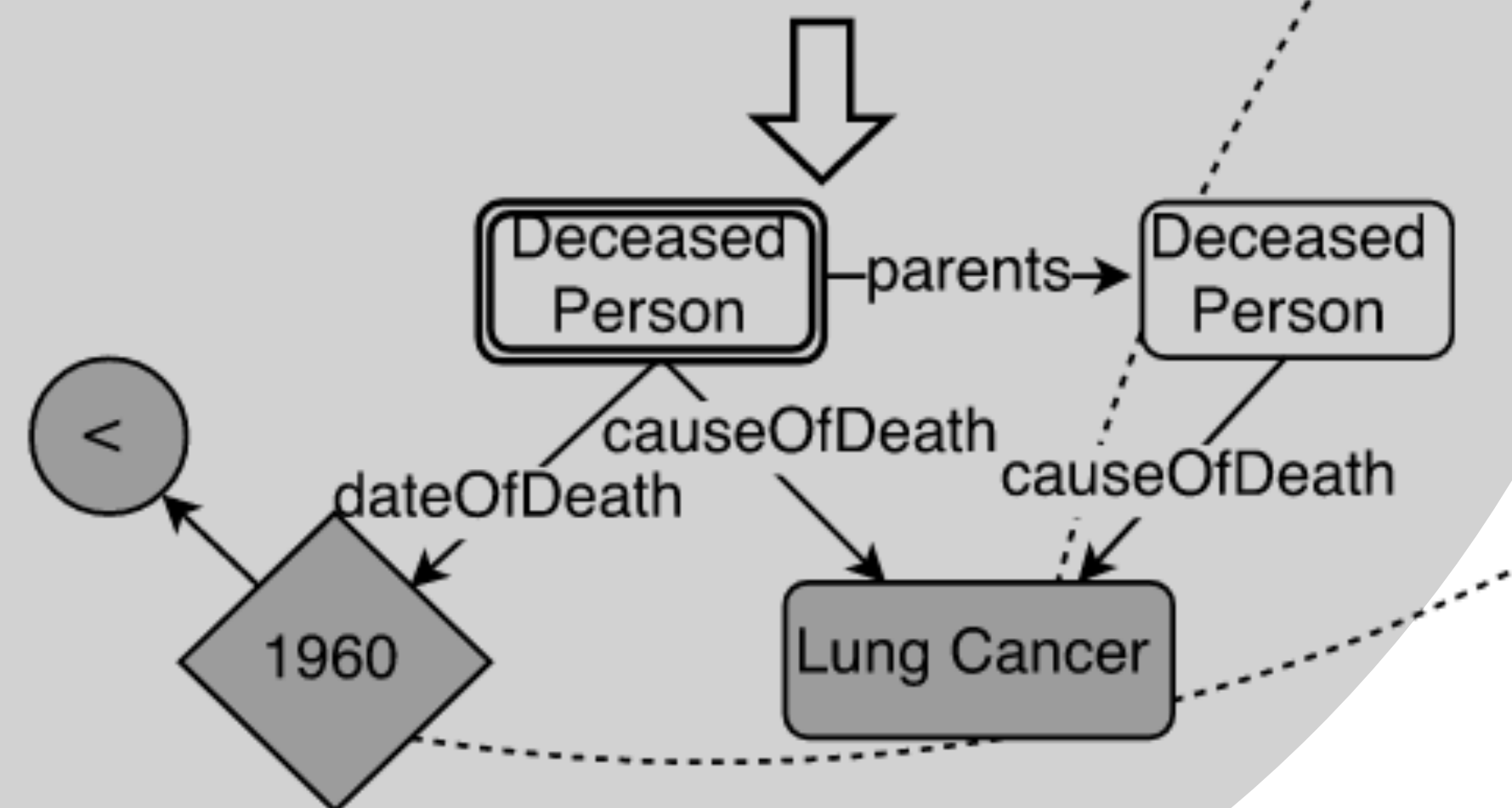
$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

$$p_1 := \text{extend}(p_0, s_0)$$

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

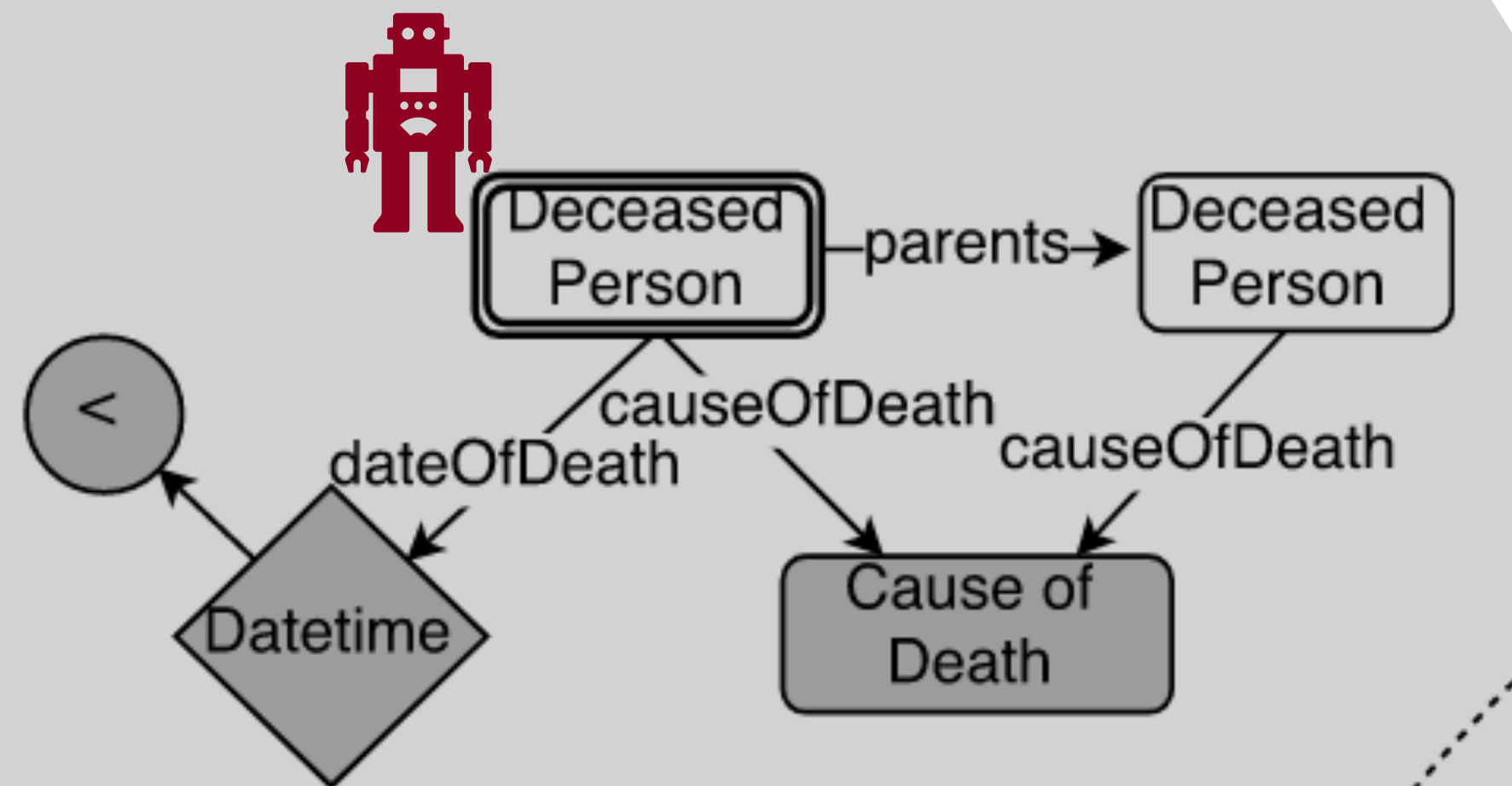
$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

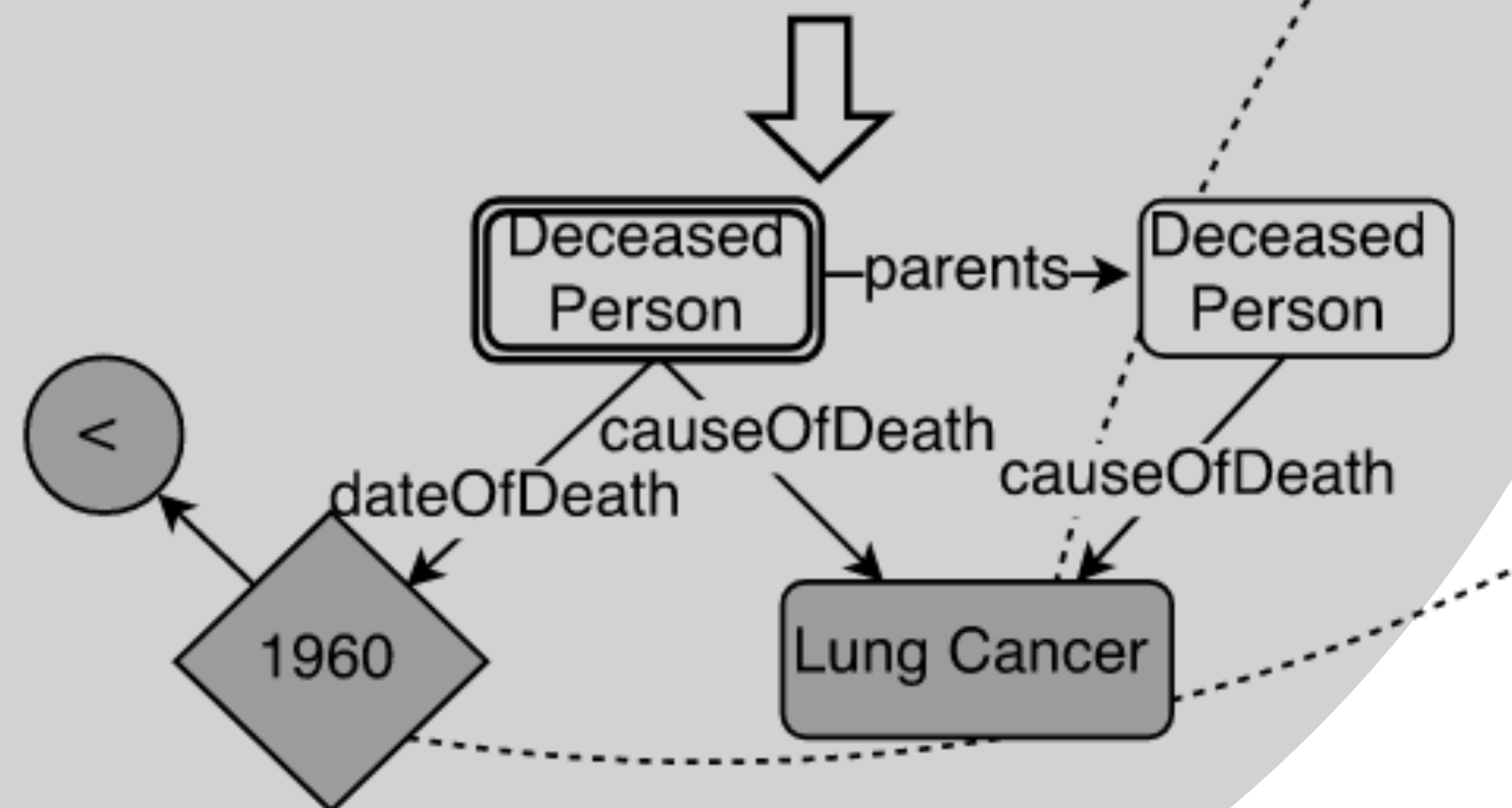
$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

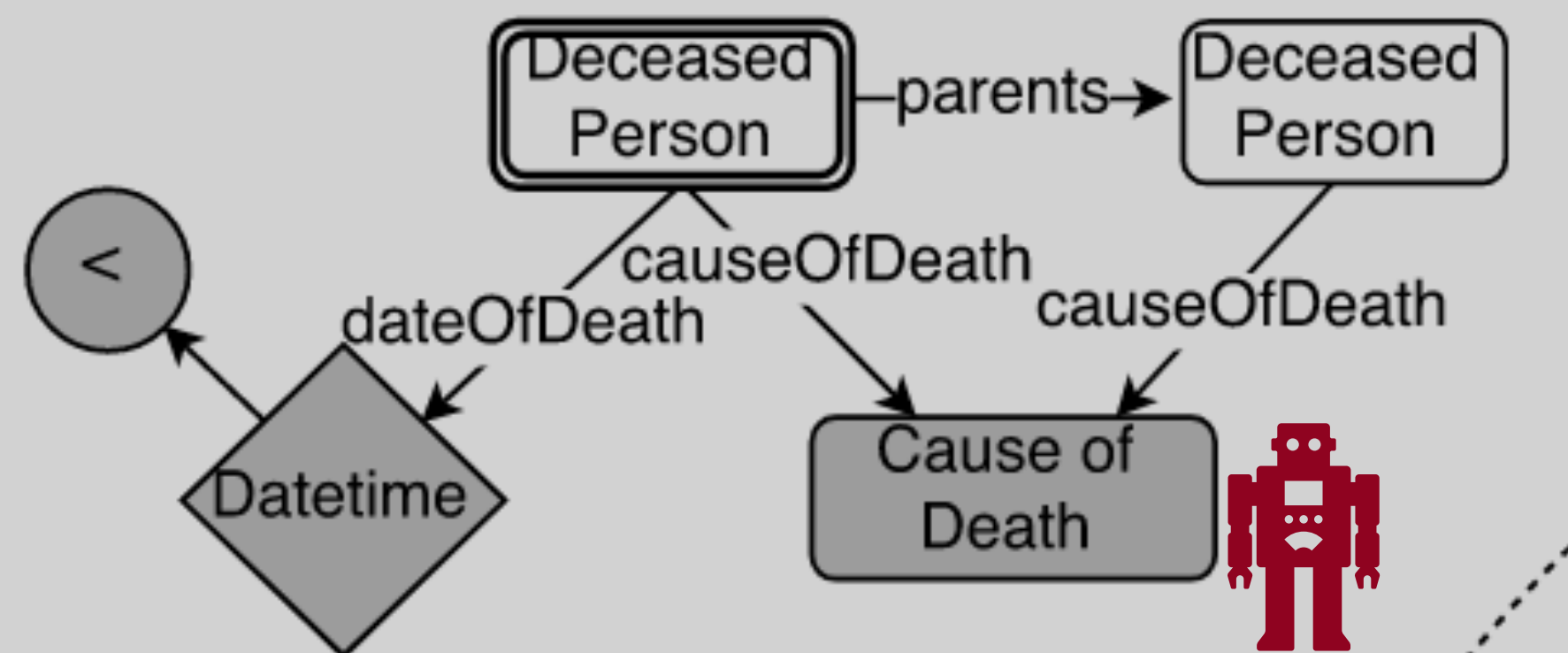
$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

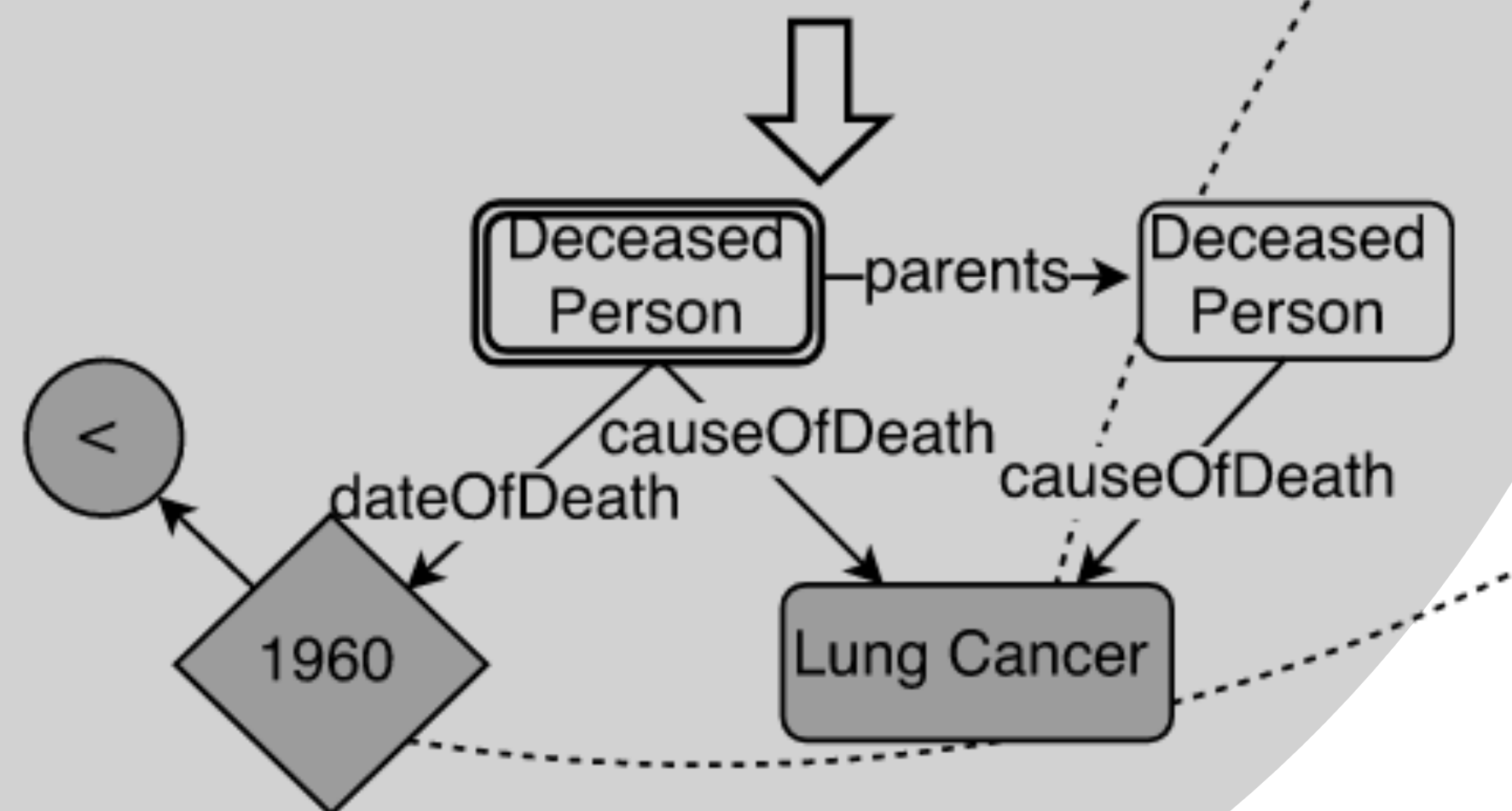
$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

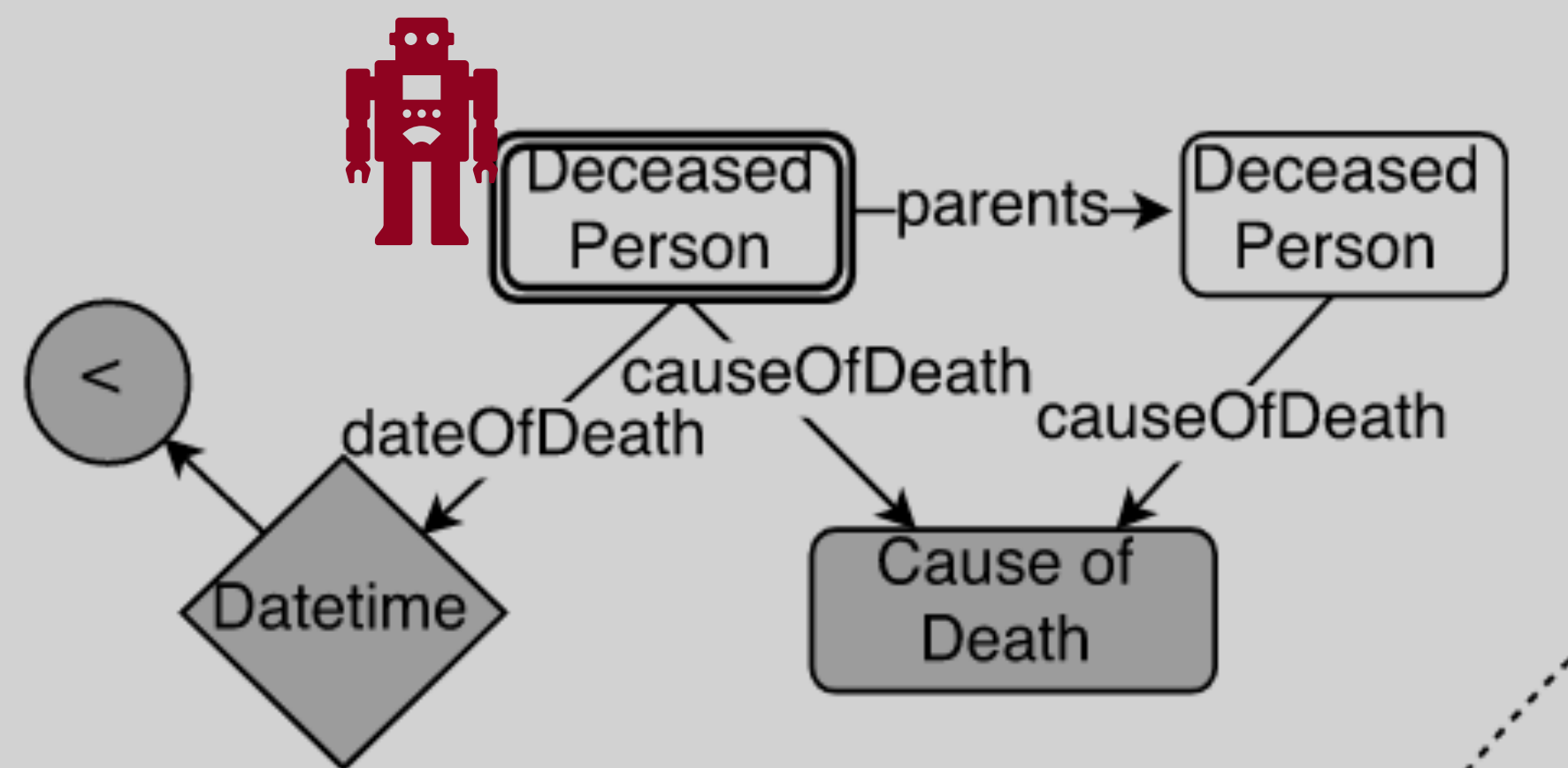
$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

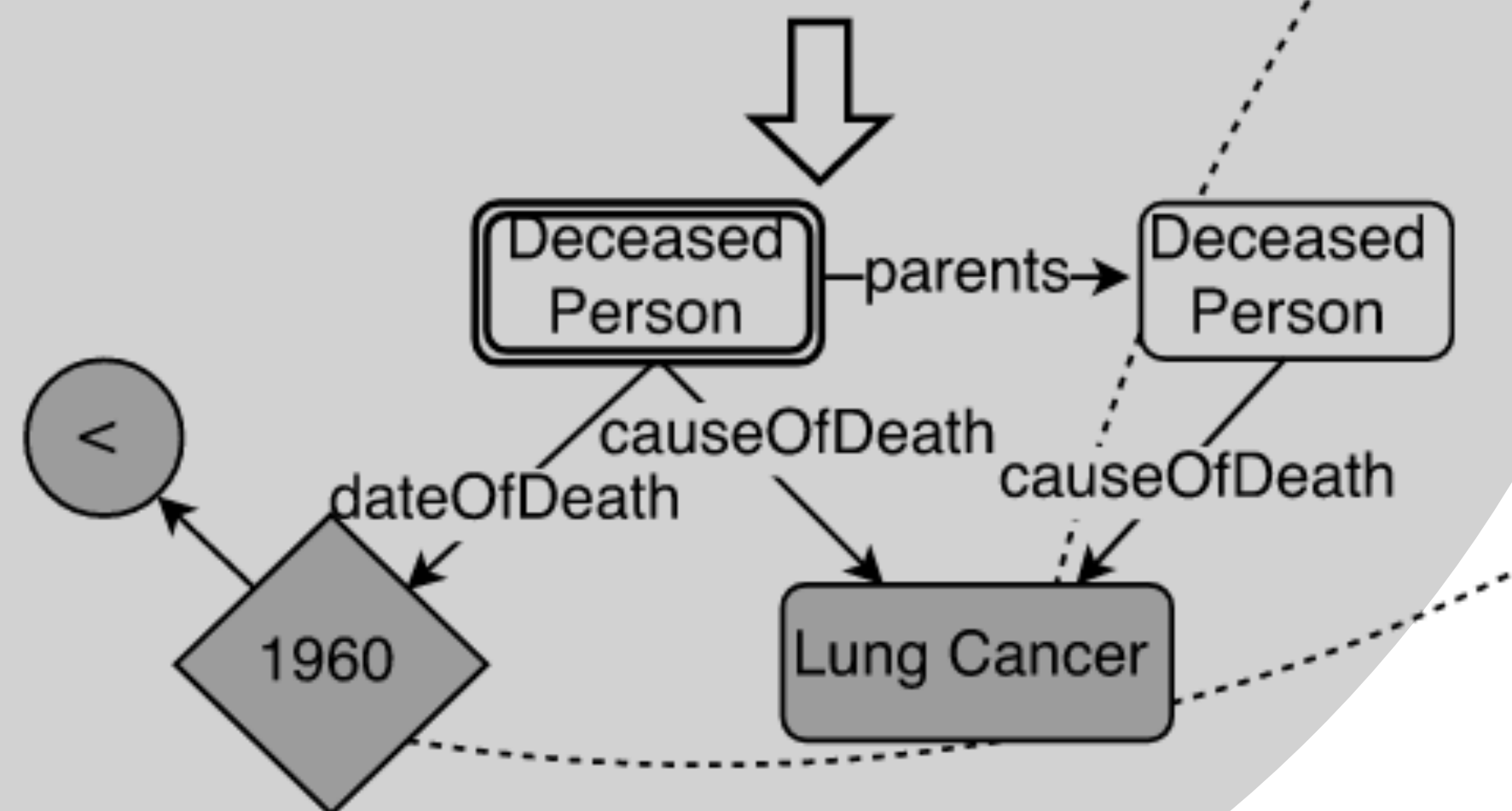
$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

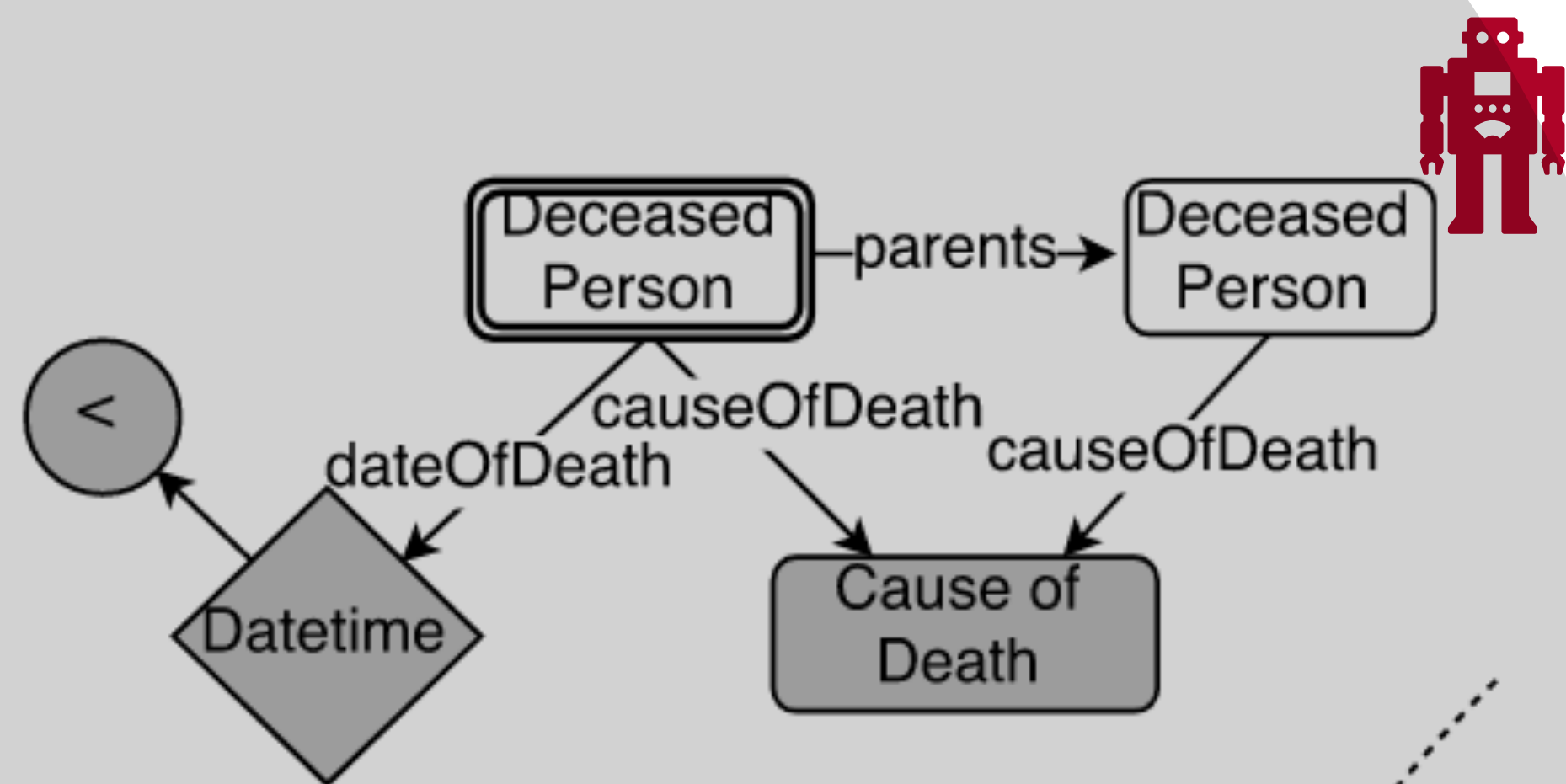
$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

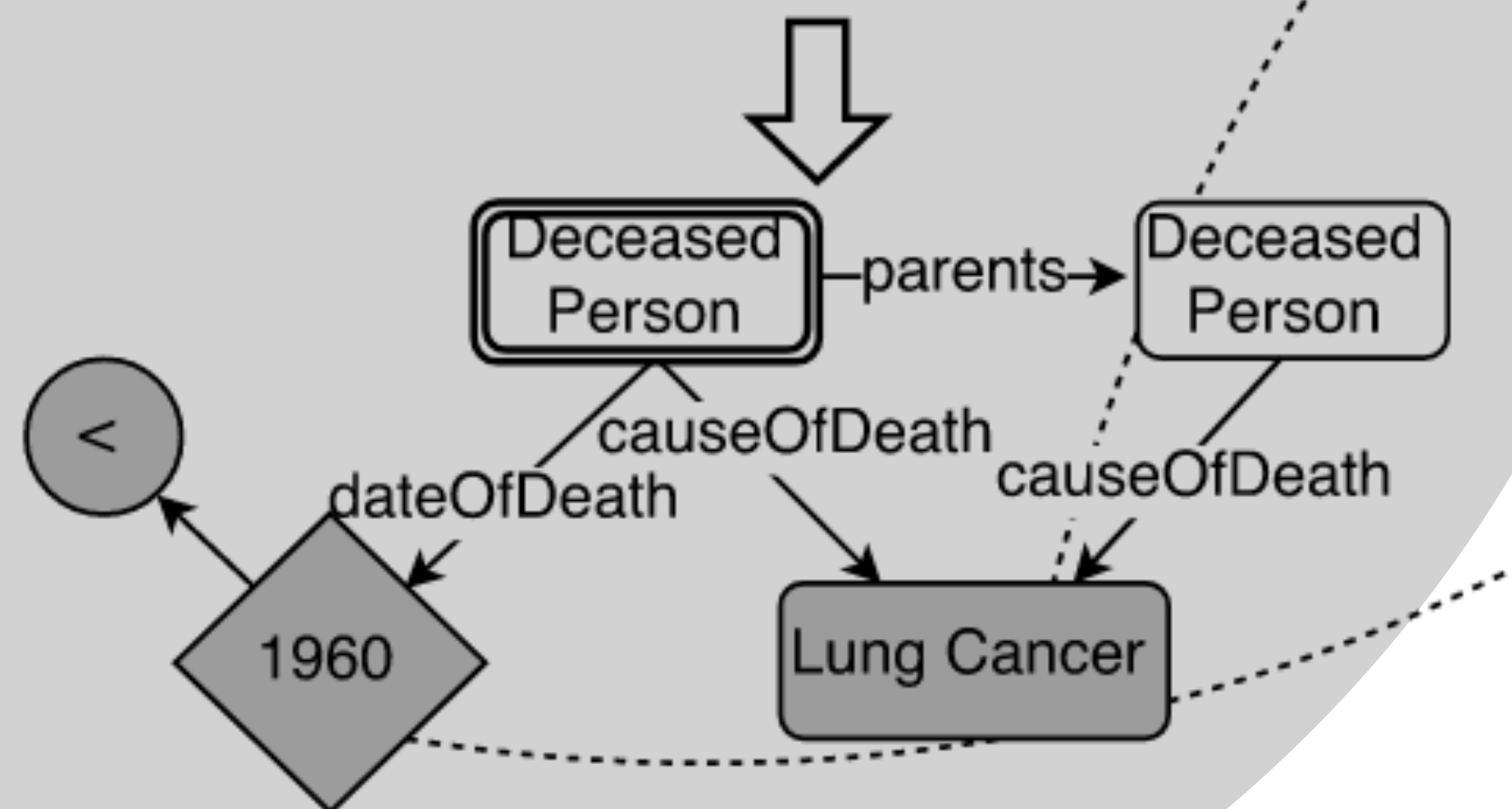
$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

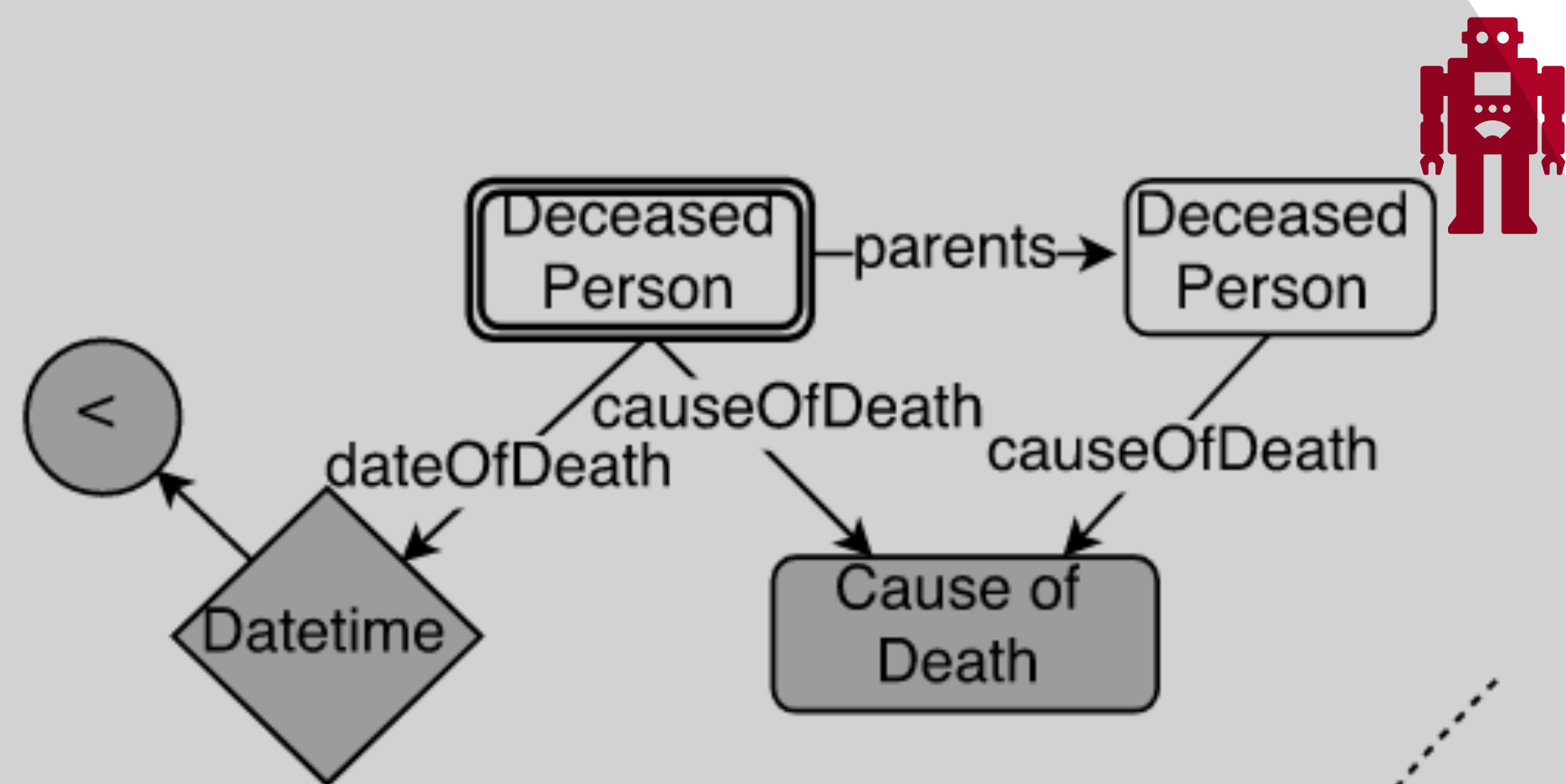
$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

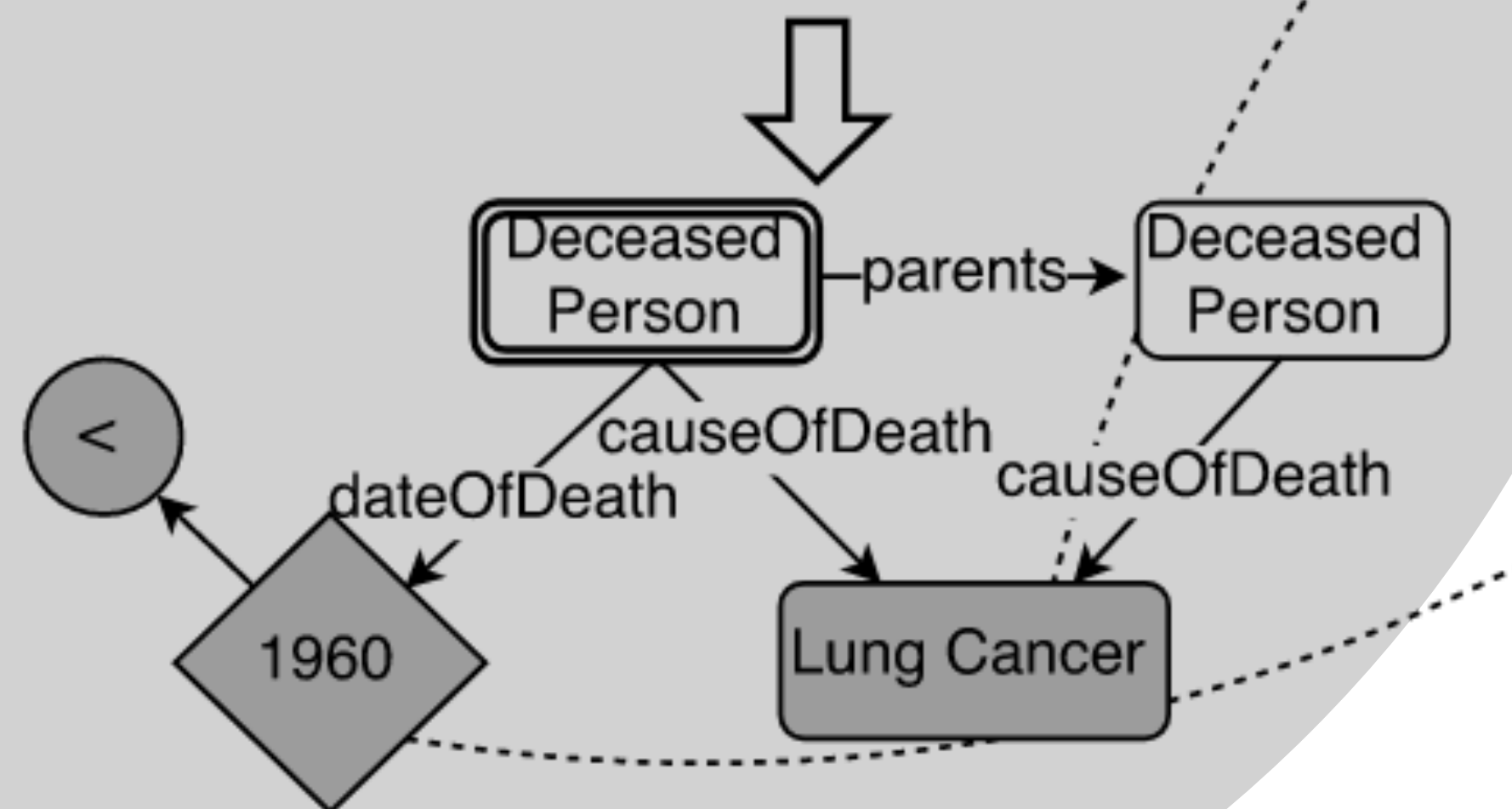
$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

$$p_0 := c_0 \sim \mathcal{C}$$

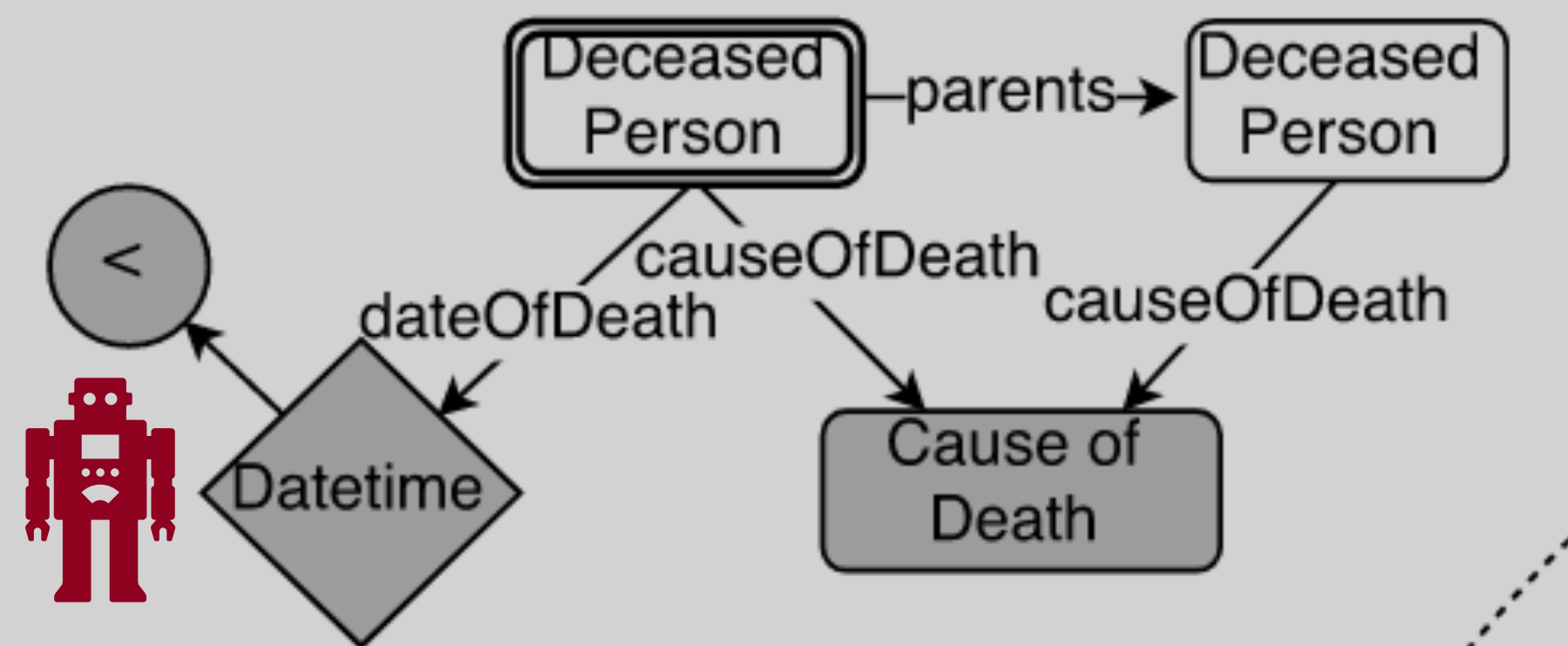
$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

$$p_1 := \text{extend}(p_0, s_0)$$

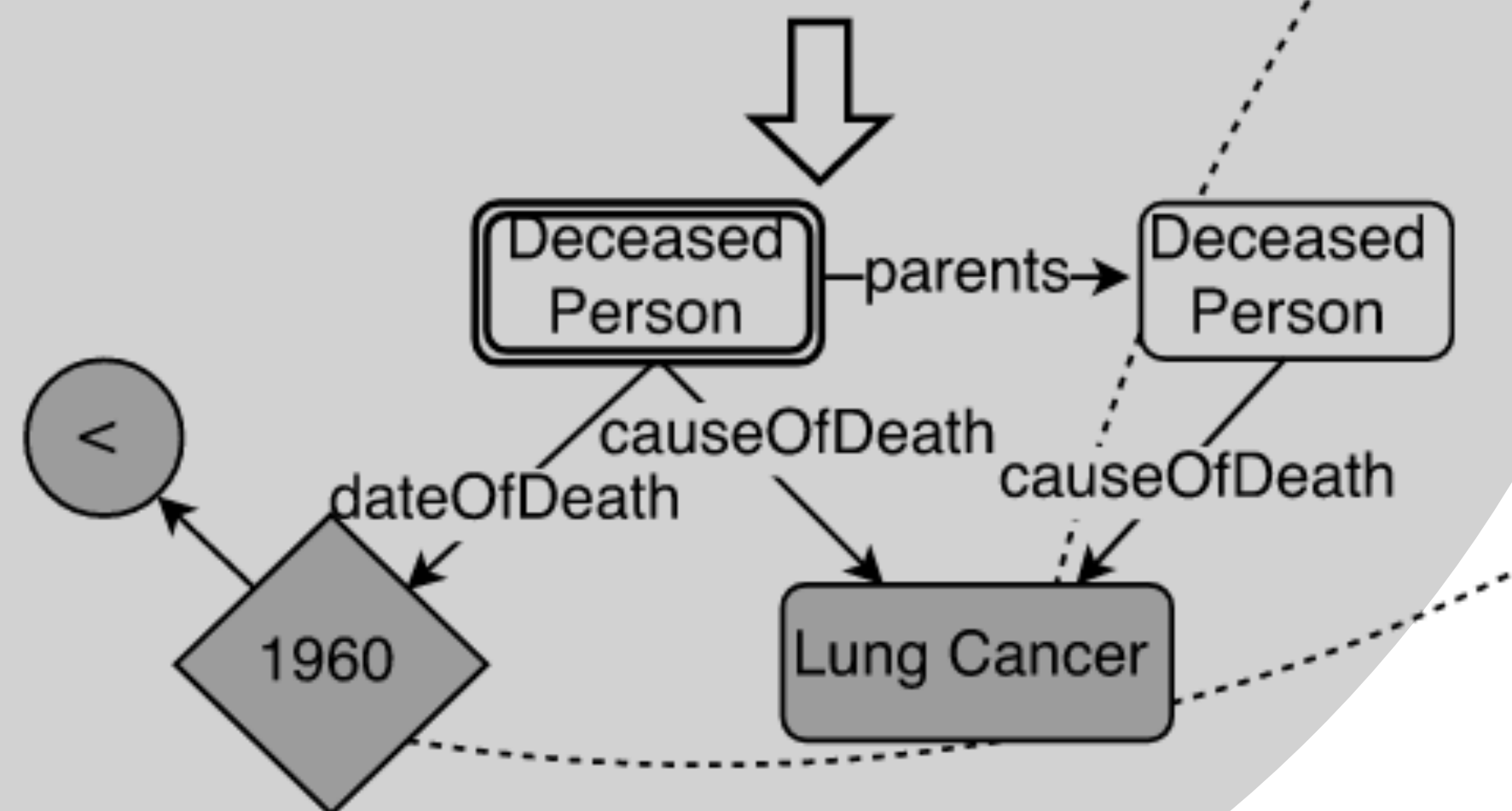
Repeat till t for t -hop complexity

$$f \sim \mathcal{F}$$

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

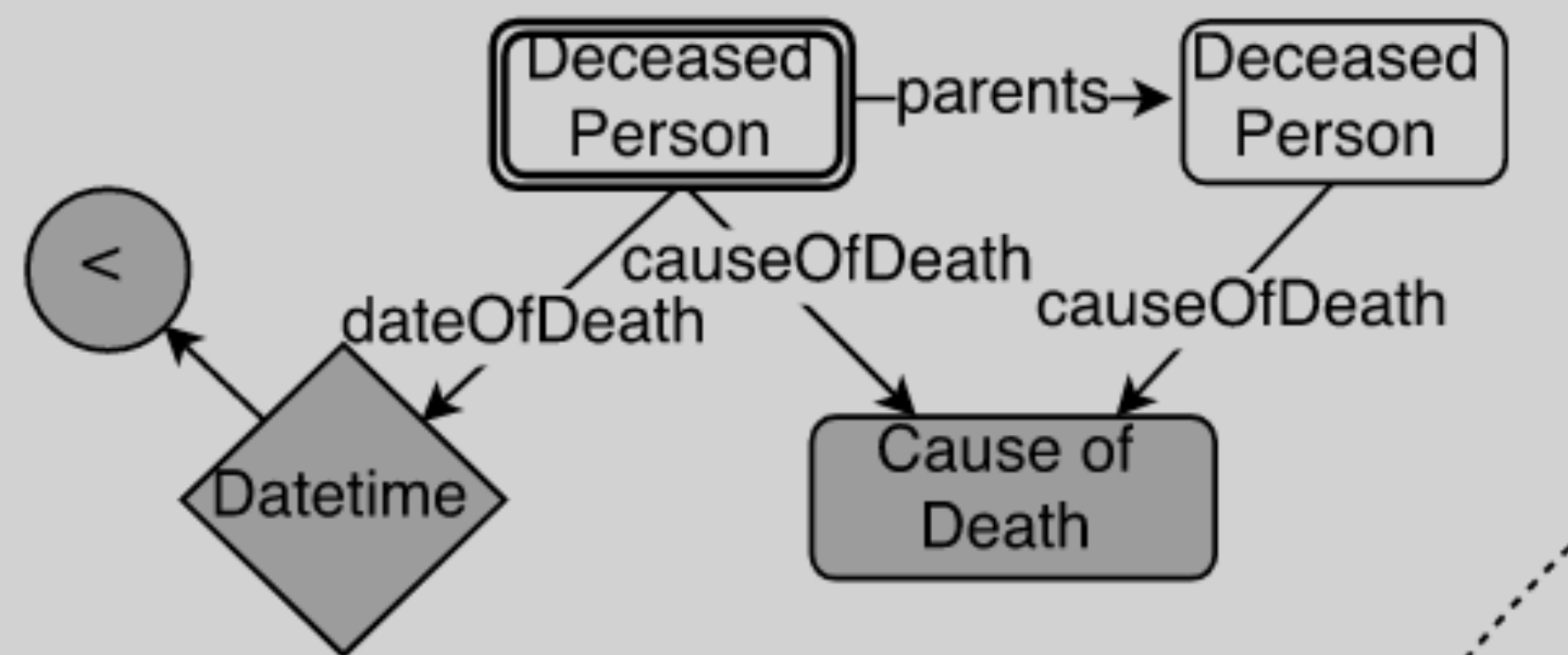
$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

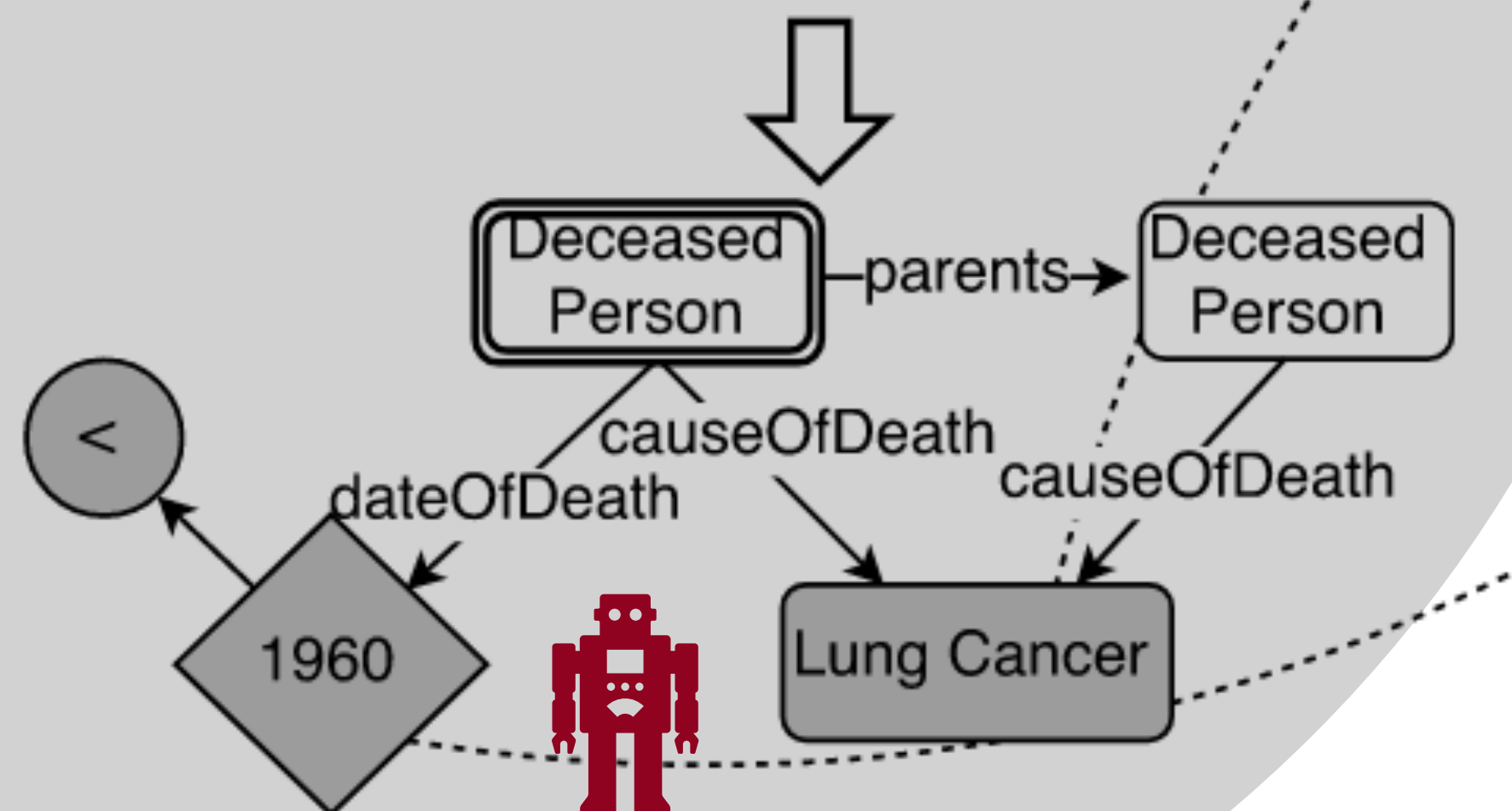
$$f \sim \mathcal{F}$$

$$p := \text{extend}(p_t, f)$$

Stage 1: Symbolic Graph Exploration



(b) Query template



(c) Graph query

$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

$$p_1 := \text{extend}(p_0, s_0)$$

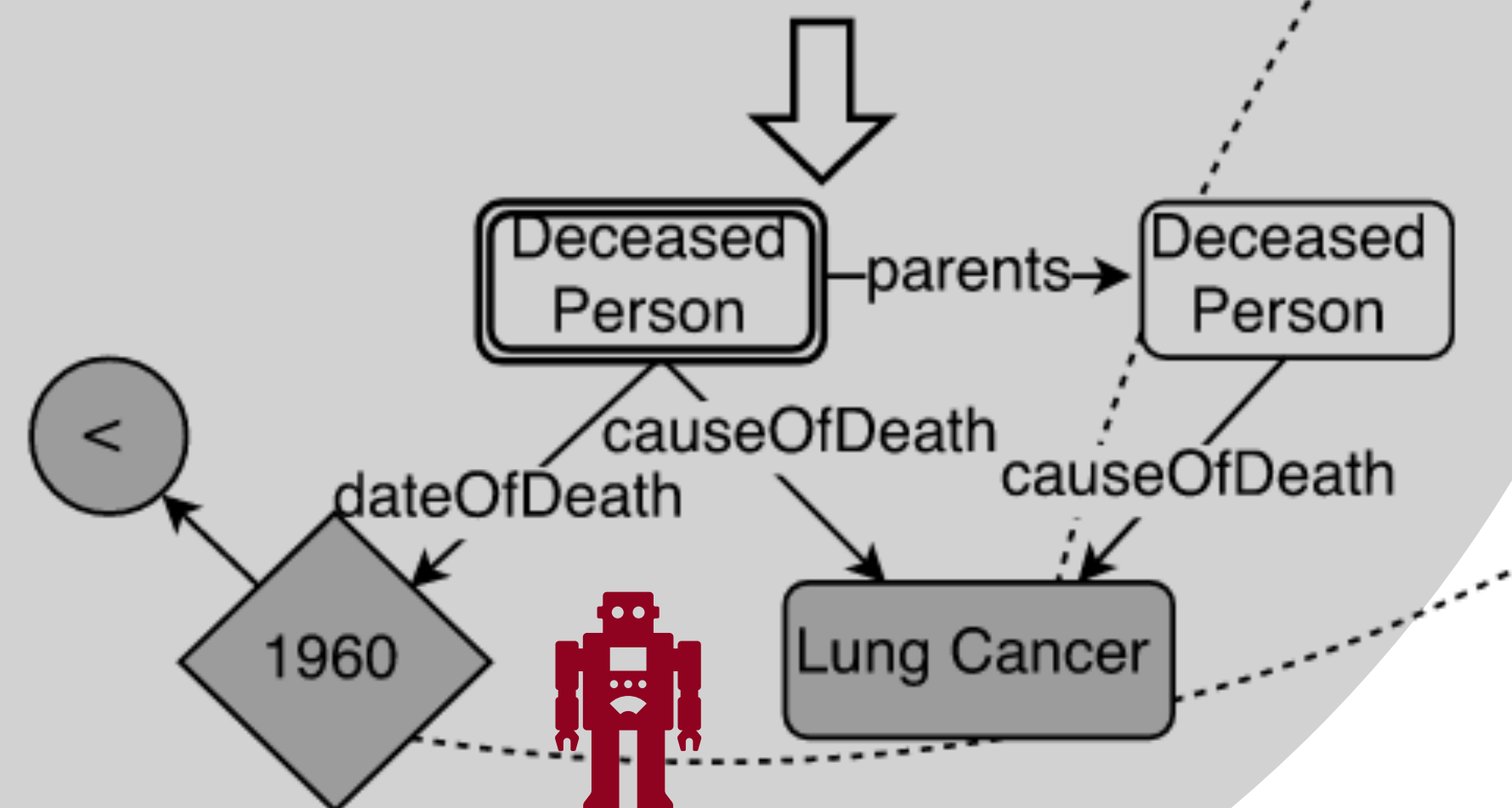
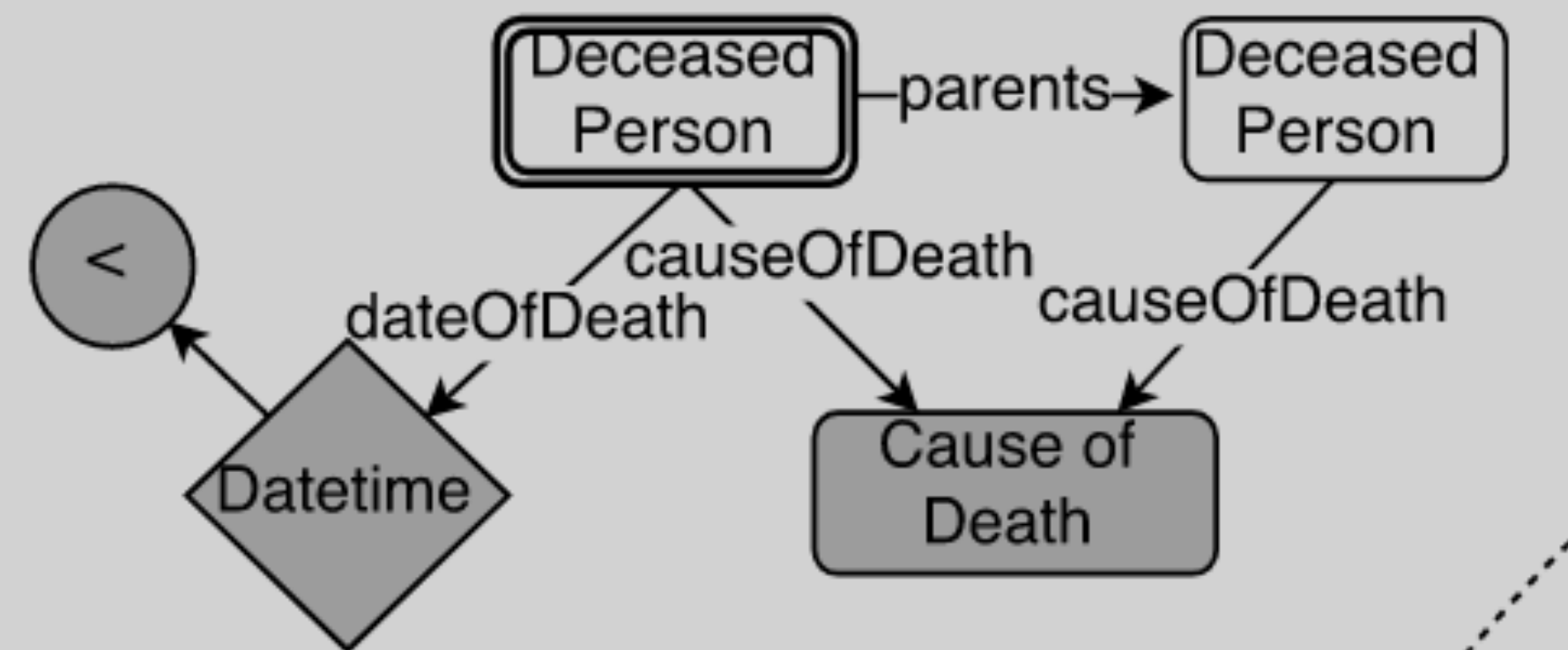
Repeat till t for t -hop complexity

$$f \sim \mathcal{F}$$

$$p := \text{extend}(p_t, f)$$

Ground classes to entities

Stage 1: Symbolic Graph Exploration



$$p_0 := c_0 \sim \mathcal{C}$$

$$s_0 \sim \{s \mid s \in \mathcal{R} \cup \mathcal{C} : \text{reachable}(p_0, s)\}$$

$$p_1 := \text{extend}(p_0, s_0)$$

Repeat till t for t -hop complexity

$$f \sim \mathcal{F}$$

$$p := \text{extend}(p_t, f)$$

Ground classes to entities

Add p to \mathcal{X}^P **Program Exploration Set**

Stage 1: Symbolic Graph Exploration

```
(AND medicine.manufactured_drug_form (JOIN medicine.manufactured_drug_form.marketing_end_date 2013-11-30^^http://www.w3.org/2001/XMLSchema#date))
```

```
(AND biology.organism_classification (JOIN biology.organism_classification.fossil_specimens m.0n8_wf9))
```

```
(AND measurement_unit.measurement_system (JOIN measurement_unit.measurement_system.pressure_units m.0h5qxr7))
```

```
(AND food.beer_style_category (JOIN food.beer_style_category.styles m.02hv1zv))
```

```
(AND food.wine_style (JOIN food.wine_style.wines (JOIN wine.wine.wine_producer m.03wz5rd)))
```

```
(COUNT (AND exhibitions.exhibition (JOIN (R exhibitions.exhibition_producer.exhibitions_produced) m.059wk)))
```

```
(ARGMIN music.music_video music.music_video.initial_release_date)
```


Stage 1: Symbolic Graph Exploration

```
(AND medicine.manufactured_drug_form (JOIN medicine.manufactured_drug_form.marketing_end_date 2013-11-30^^http://www.w3.org/2001/XMLSchema#date))
```

```
(AND biology.organism_classification (JOIN biology.organism_classification.fossil_specimens m.0n8_wf9))
```

```
(AND measurement_unit.measurement_system (JOIN measurement_unit.measurement_system.pressure_units m.0h5qxr7))
```

```
(AND food.beer_style_category (JOIN food.beer_style_category.styles m.02hv1zv))
```

```
(AND food.wine_style (JOIN food.wine_style.wines (JOIN wine.wine.wine_producer m.03wz5rd)))
```

```
(COUNT (AND exhibitions.exhibition (JOIN (R exhibitions.exhibition_producer.exhibitions_produced) m.059wk)))
```

```
(ARGMIN music.music_video music.music_video.initial_release_date)
```

Stage 1: Symbolic Graph Exploration

```
(AND medicine.manufactured_drug_form (JOIN medicine.manufactured_drug_form.marketing_end_date 2013-11-30^^http://www.w3.org/2001/XMLSchema#date))
```

```
(AND biology.organism_classification (JOIN biology.organism_classification.fossil_specimens m.0n8_wf9))
```

```
(AND measurement_unit.measurement_system (JOIN measurement_unit.measurement_system.pressure_units m.0h5qxr7))
```

```
(AND food.beer_style_category (JOIN food.beer_style_category.styles m.02hv1zv))
```

```
(AND food.wine_style (JOIN food.wine_style.wines (JOIN wine.wine.wine_producer m.03wz5rd)))
```

```
(COUNT (AND exhibitions.exhibition (JOIN (R exhibitions.exhibition_producer.exhibitions_produced) m.059wk)))
```

```
(ARGMIN music.music_video music.music_video.initial_release_date)
```

Stage 1: Symbolic Graph Exploration

```
(AND medicine.manufactured_drug_form (JOIN medicine.manufactured_drug_form.marketing_end_date 2013-11-30^^http://www.w3.org/2001/XMLSchema#date))
```

```
(AND biology.organism_classification (JOIN biology.organism_classification.fossil_specimens m.0n8_wf9))
```

Fast! 10,000 programs in ~1.5hrs on Freebase

```
(AND food.wine_style (JOIN food.wine_style.wines (JOIN wine.wine.wine_producer m.03wz5rd)))
```

```
(COUNT (AND exhibitions.exhibition (JOIN (R exhibitions.exhibition_producer.exhibitions_produced) m.059wk)))
```

```
(ARGMIN music.music_video music.music_video.initial_release_date)
```

Stage 2: Question Generation

Goal: $\mathcal{X} := \{(q_p, p \mid p \in \mathcal{X}^P)\}$

Stage 2: Question Generation

Goal: $\mathcal{X} := \{(q_p, p \mid p \in \mathcal{X}^P)\}$

Stage 2: Question Generation

Goal: $\mathcal{X} := \{(q_p, p \mid p \in \mathcal{X}^P)\}$



Stage 2: Question Generation

Goal: $\mathcal{X} := \{(q_p, p \mid p \in \mathcal{X}^P)\}$



**Schema
Descriptions**

$S_p := \{\text{desc}(s) \mid s \in \mathcal{R} \cup \mathcal{C} \cup \mathcal{F} : \text{contains}(p, s)\}$

Stage 2: Question Generation

Goal: $\mathcal{X} := \{(q_p, p \mid p \in \mathcal{X}^P)\}$

1. Most of our experiments are with small-mid LLMs (7B)
2. We use open-source models.



**Schema
Descriptions**

$S_p := \{\text{desc}(s) \mid s \in \mathcal{R} \cup \mathcal{C} \cup \mathcal{F} : \text{contains}(p, s)\}$

Stage 2: Question Generation

Goal: $\mathcal{X} := \{(q_p, p \mid p \in \mathcal{X}^P)\}$

where



**Schema
Descriptions**

$S_p := \{\text{desc}(s) \mid s \in \mathcal{R} \cup \mathcal{C} \cup \mathcal{F} : \text{contains}(p, s)\}$

Stage 2: Question Generation

Goal: $\mathcal{X} := \{ (a, n) \mid n \in \mathcal{X}^P \}$

However, zero-shot LLM generation is challenging:

- #1 Erroneous generations for complex, multi-hop programs
- #2 Incorrect top-1 predictions from the model

Schema
Descriptions $S_p := \{ \text{desc}(s) \mid s \in \mathcal{R} \cup \mathcal{C} \cup \mathcal{F} : \text{contains}(p, s) \}$

Stage 2: Question Generation

Solution for #1

Least-to-Most Prompting

Query: (JOIN (R movie.written_by) (JOIN movie.starred_actors (JOIN (R movie.starred_actors) "Titanic"))))

L2M prompting:

Step 1:

(JOIN (R movie.starred_actors) m.6594)

Prediction: Who starred in Titanic?

Step 2:

(JOIN (R movie.starred_actors) m.6594)

Who starred in Titanic?

(JOIN movie.starred_actors (JOIN (R movie.starred_actors) "Titanic"))

Prediction: What movies have the actors who starred in Titanic starred in?

Step 3:

(JOIN (R movie.starred_actors) m.6594)

Who starred in Titanic?

(JOIN movie.starred_actors (JOIN (R movie.starred_actors) "Titanic"))

What movies have the actors who starred in Titanic starred in?

(JOIN (R movie.written_by) (JOIN movie.starred_actors (JOIN (R movie.starred_actors) "Titanic"))))

Prediction: Who wrote the movies that have actors who⁴⁹ starred in the Titanic?

Stage 2: Question Generation

Problem #2

Beam Search Decoding

p
+
 S_p



$q_p^{(1)}$

Highest log-probability score
(but **incorrect**)

$q_p^{(2)}$

Lower log-probability score
(but **correct**)

⋮

$q_p^{(n)}$

Stage 2: Question Generation



Solution for #2

Inverse-Consistency Re-ranking

Given a generative task

$$T := y \mid I, D, x$$

For our setting, $D = \emptyset$

Standard top-1 prediction
from a decoding algorithm

$$y_{\text{pred}} := \arg \max_{y \in \mathbf{y}_{\text{cands}}} \log \Pr(y \mid I, D, x) / |y|$$

Score using the *inverse* task

$$T^{-1} := x \mid I^{-1}, D^{-1}, y$$

$$y_{\text{pred}} := \arg \max_{y \in \mathbf{y}_{\text{cands}}} \log \Pr(x \mid I^{-1}, D^{-1}, y) / |x|$$

Stage 2: Question Generation



Solution for #2

Inverse-Consistency Re-ranking

```
"query": "(AND religion.founding_figure (JOIN religion.founding_figure.religion_founded
(JOIN religion.religion.founding_figures \"st. peter\")))",
"reference": "who were the founding figures of the religion that st. peter founded"
"standard_decoding": [
  "who is paul the apostle?",
  "who founded christianity?",
  "who was the founder of christianity?",
  "what is the name of the person who founded christianity?",
  "who are the founding figures of the religion founded by st. peter?",
  "who is the founding figure of the religion founded by st. peter?",
]
"inverse_consistency": [
  "who is the founding figure of the religion founded by st. peter?",
  "who are the founding figures of the religion founded by st. peter?",
  "who was the founder of christianity?",
  "who founded christianity?",
  "who is paul the apostle?",
  "what is the name of the person who founded christianity?",
]
```


Stage 2: Question Generation



Solution for #2

Inverse-Consistency Re-ranking

Necessary for fast inference since question generation is the bottleneck in exploration

```
"query": "(AND religion.founding_figure (JOIN religion.founding_figure.  
(JOIN religion.religion.founding_figures \"st. peter\")))",
```

Reliably improves top-1 prediction!
Particularly for models with **lower-parameter** counts.



```
"inverse_consistency": [  
  "who is the founding figure of the religion founded by st. peter?",  
  "who are the founding figures of the religion founded by st. peter?",  
  "who was the founder of christianity?",  
  "who founded christianity?",  
  "who is paul the apostle?",  
  "what is the name of the person who founded christianity?",  
]
```

We have Q&A pairs, lets train a model!

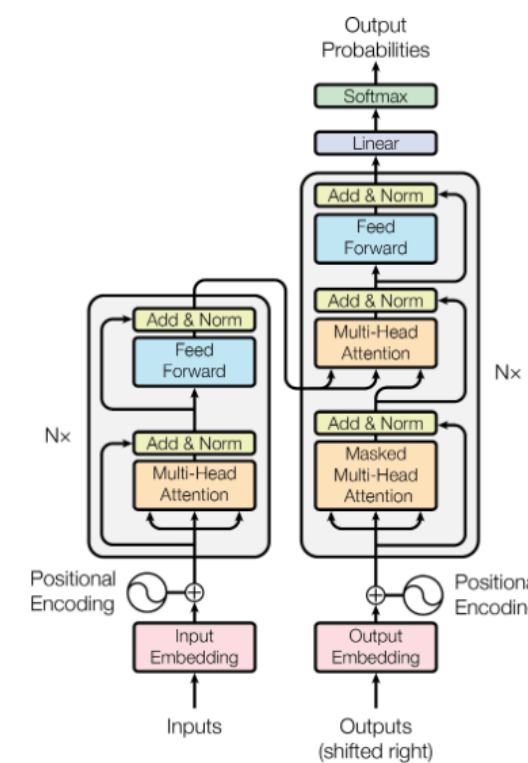
We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\} +$$



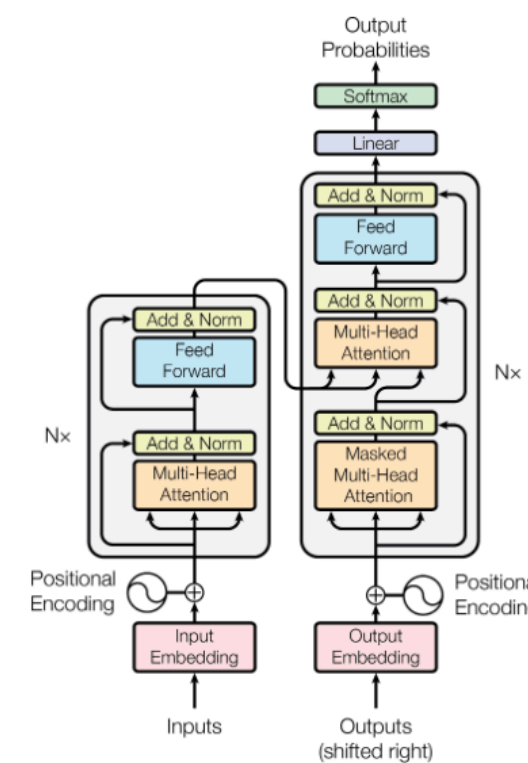
=

A parametric model
specific to the KG!

We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\} +$$



=

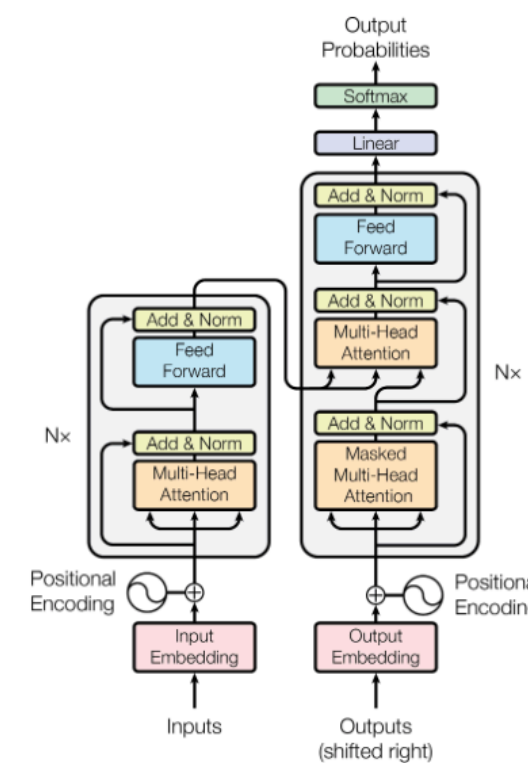
A parametric model
specific to the KG!

Hence will not work for other KGs 🥲

We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\} +$$



=

A parametric model specific to the KG!

Hence will not work for other KGs 🥲

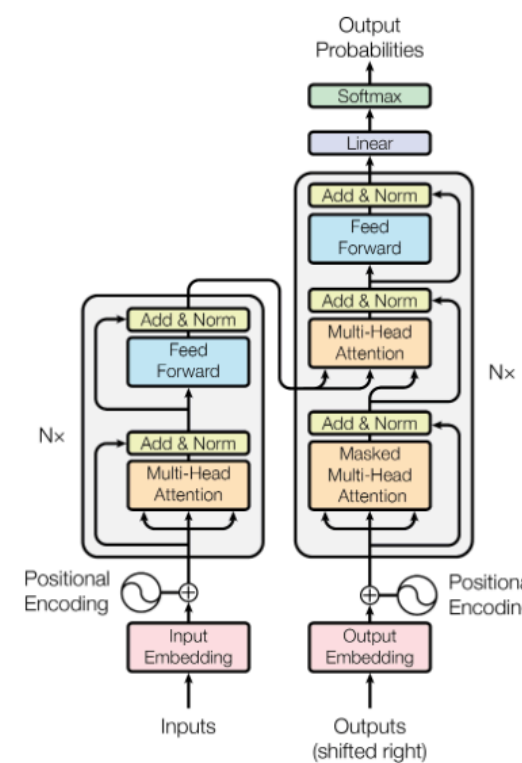
- Semiparametric approach with the generated (Q,P) pairs in nonparametric memory.

We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\}$$

+

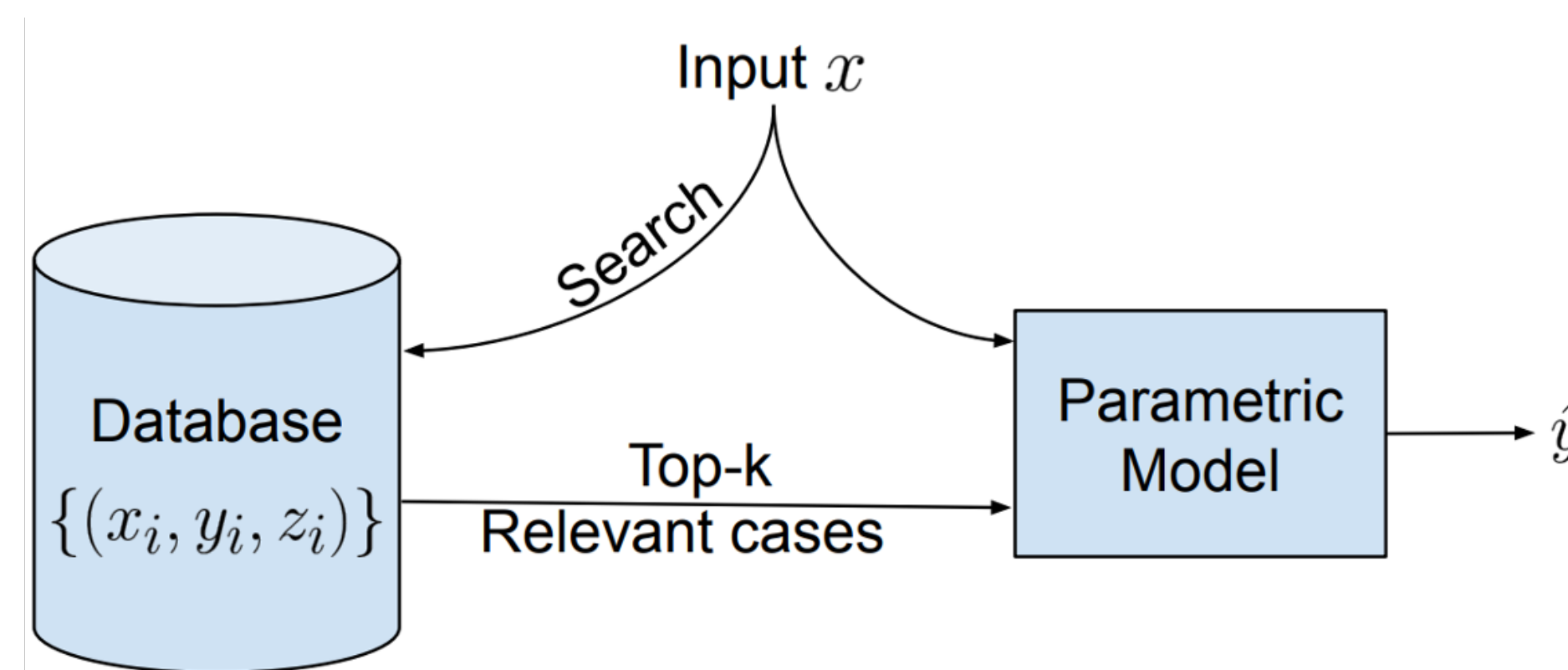


=

A parametric model
specific to the KG!

Hence will not work for other KGs 🥲

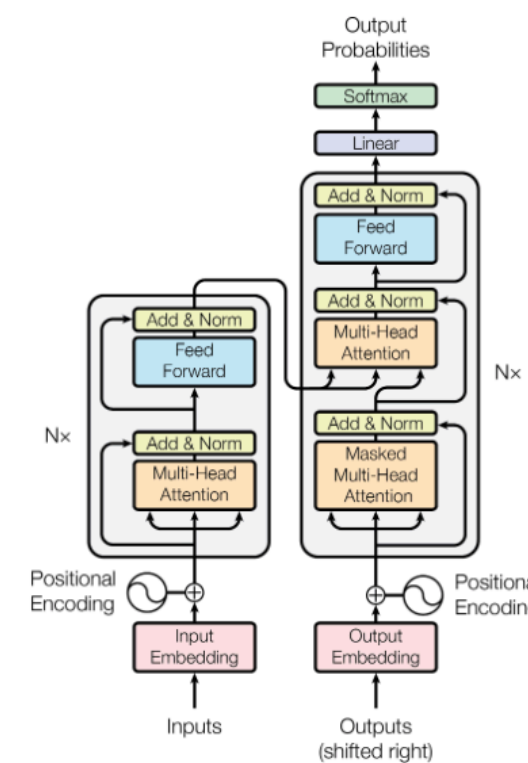
- Semiparametric approach with the generated (Q,P) pairs in nonparametric memory.



We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\} +$$



=

A parametric model
specific to the KG!

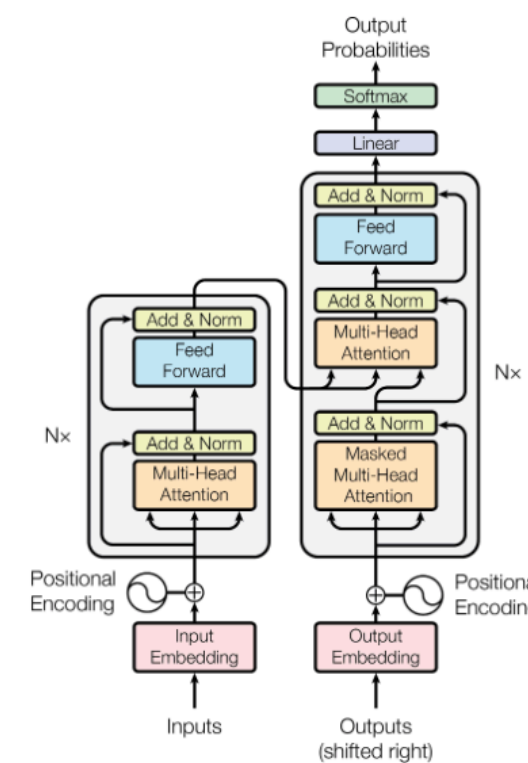
Hence will not work for other KGs 🥲

- Semiparametric approach with the generated (Q,P) pairs in nonparametric memory.

We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\} +$$



=

A parametric model
specific to the KG!

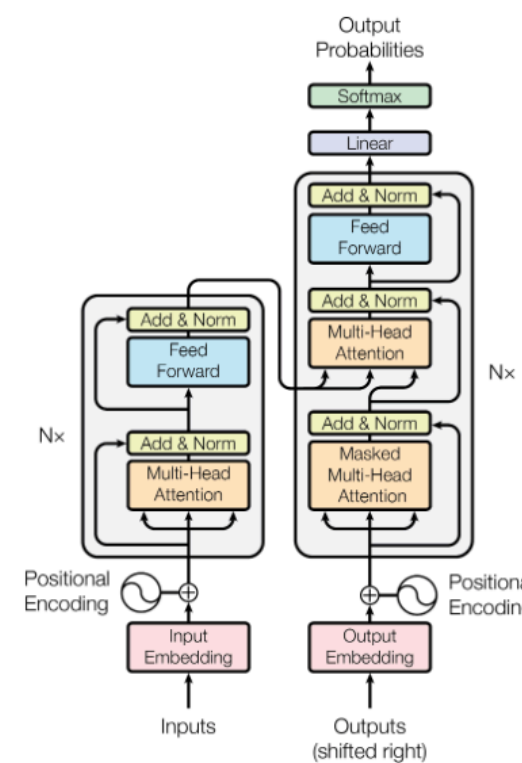
Hence will not work for other KGs 🥲

- Semiparametric approach with the generated (Q,P) pairs in nonparametric memory.
- The parametric component is an LLM that uses the demonstration retrieved from the memory

We have Q&A pairs, lets train a model!

- We have query and program pairs that we generated from the graph. So we can now train supervised model on this graph, right?

$$\mathcal{X} := \{(q_p, p) \mid p \in \mathcal{X}^P\} +$$



=

A parametric model
specific to the KG!

Hence will not work for other KGs 😓

- Semiparametric approach with the generated (Q,P) pairs in nonparametric memory.
- The parametric component is an LLM that uses the demonstration retrieved from the memory
- No fine-tuning - Therefore our method works for any KG 😊

Stage 3: Bottom-up Grounded Reasoning

Stage 3: Bottom-up Grounded Reasoning

Input utterance:

 What is the latest released computer emulator developed in *Java*?

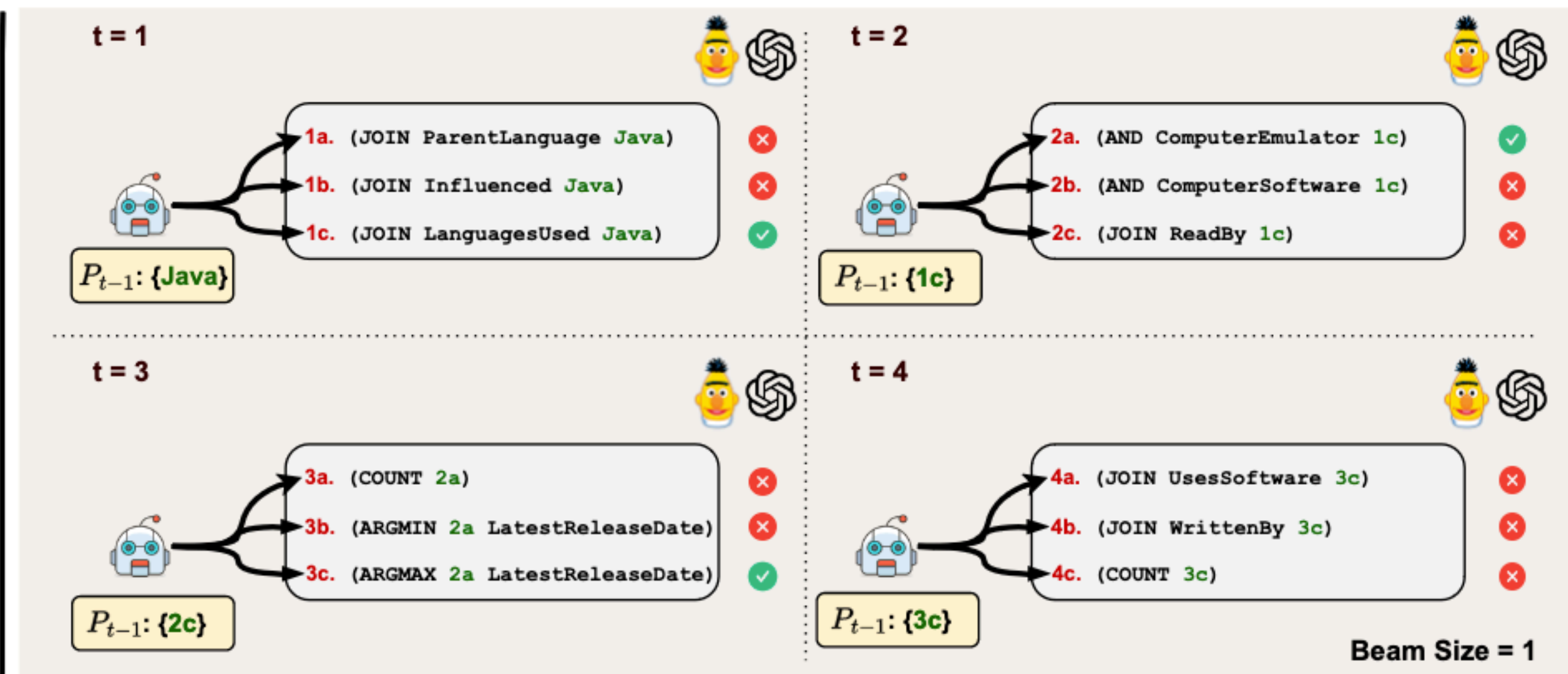
Environment:



Target plan:

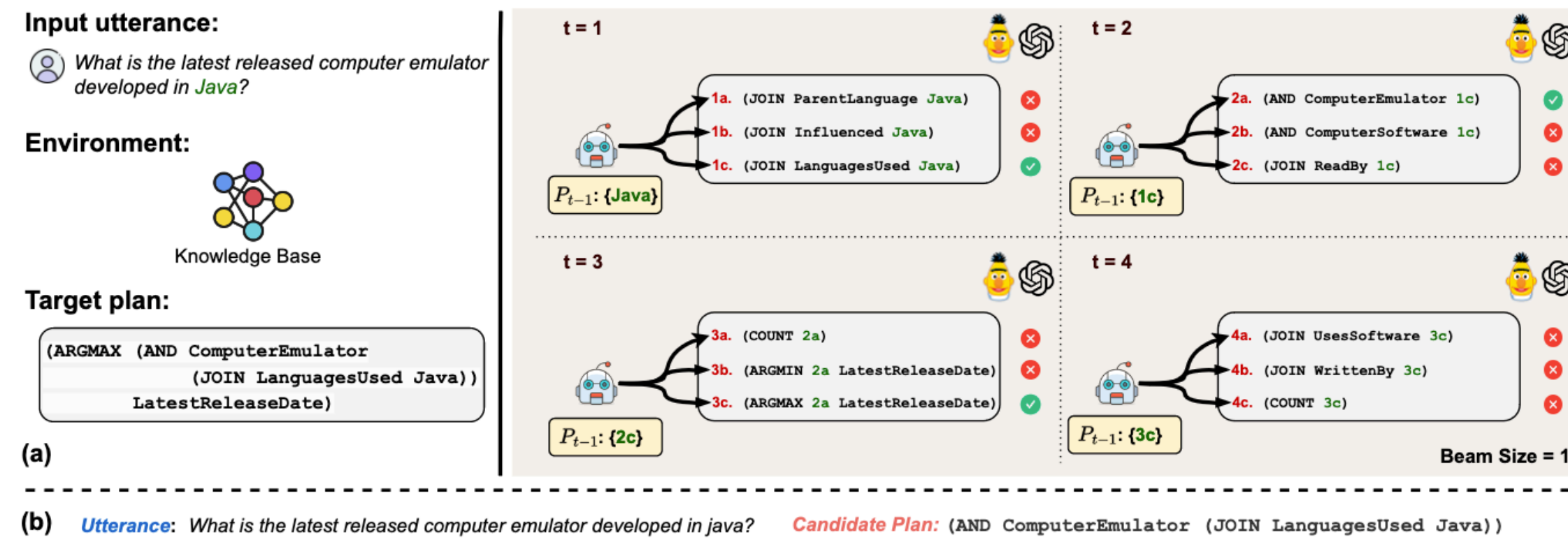
```
(ARGMAX (AND ComputerEmulator
           (JOIN LanguagesUsed Java))
         LatestReleaseDate)
```

(a)

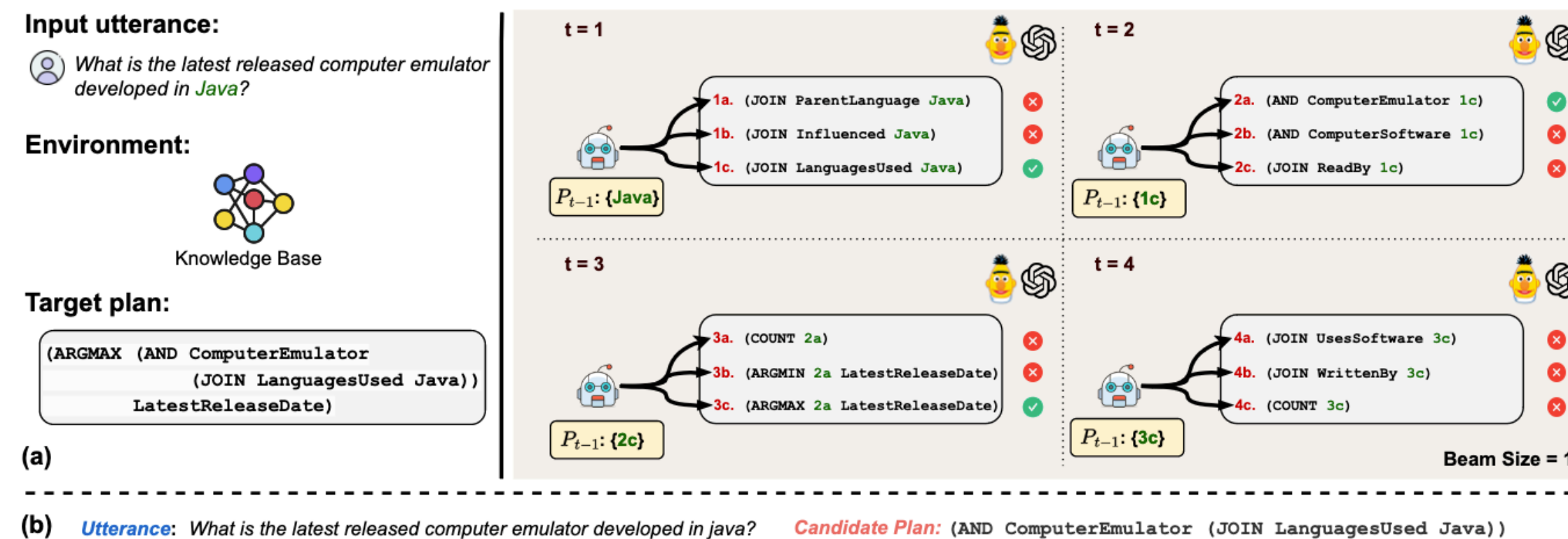


(b) **Utterance:** What is the latest released computer emulator developed in java? **Candidate Plan:** (AND ComputerEmulator (JOIN LanguagesUsed Java))

Stage 3: Bottom-up Grounded Reasoning

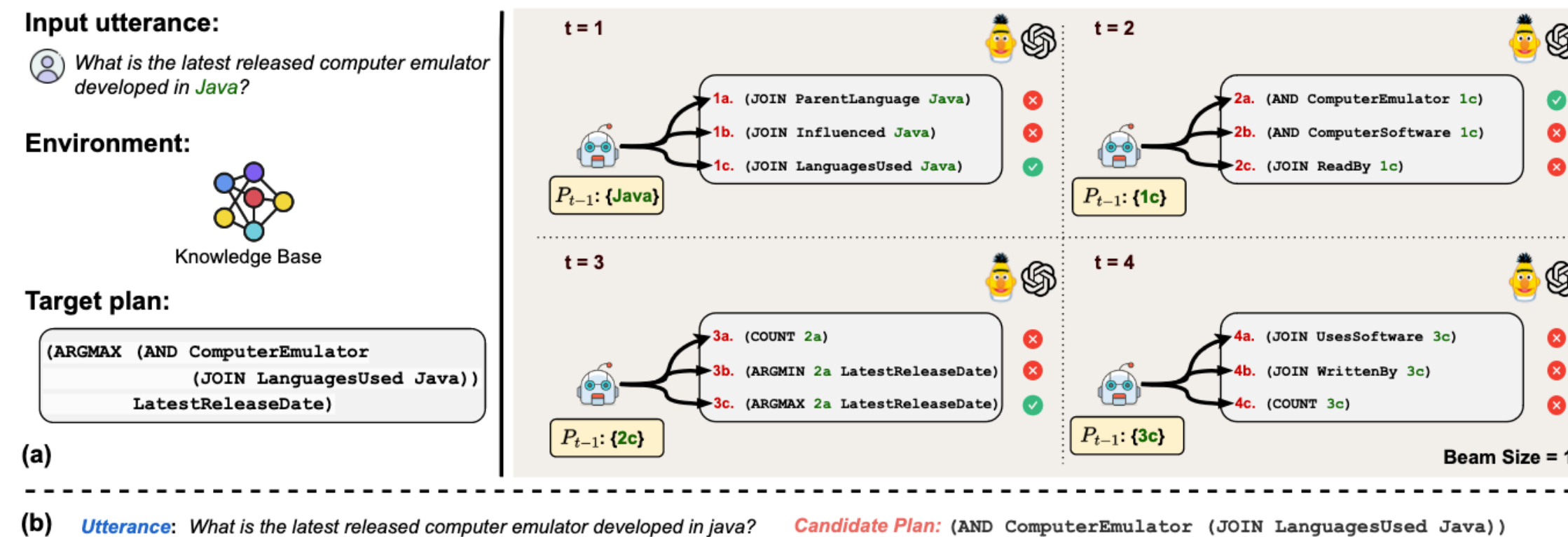


Stage 3: Bottom-up Grounded Reasoning



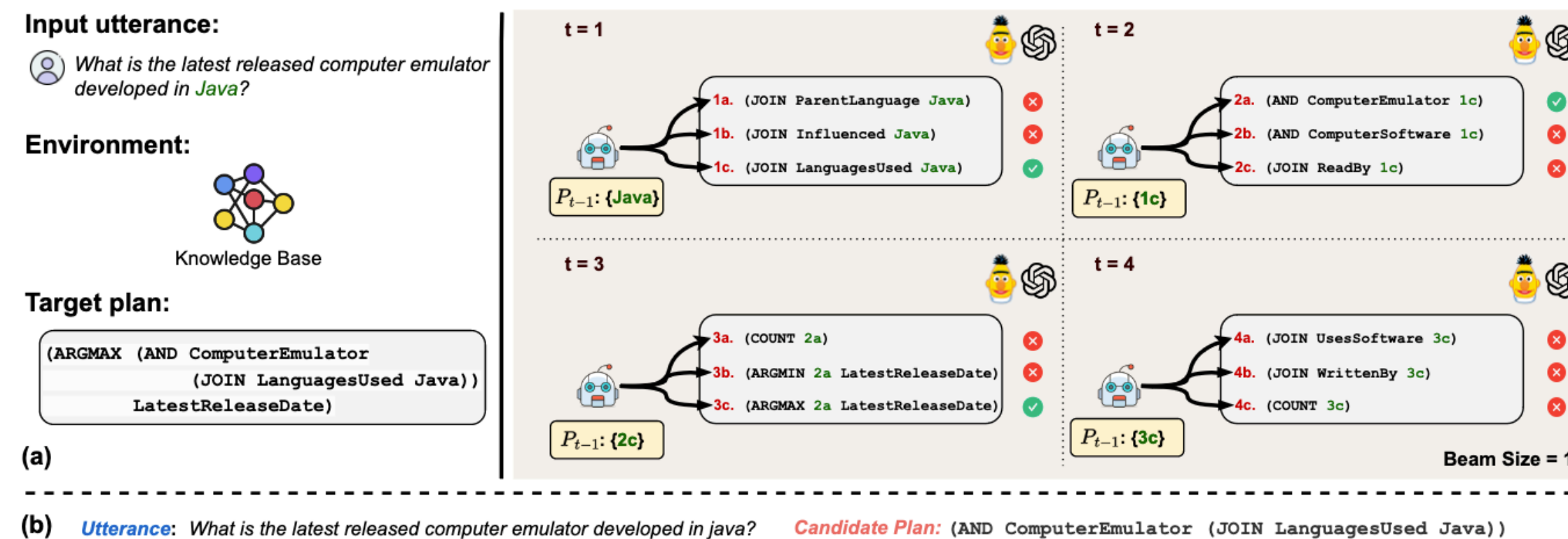
- 2 key changes that improves both speed and accuracy

Stage 3: Bottom-up Grounded Reasoning



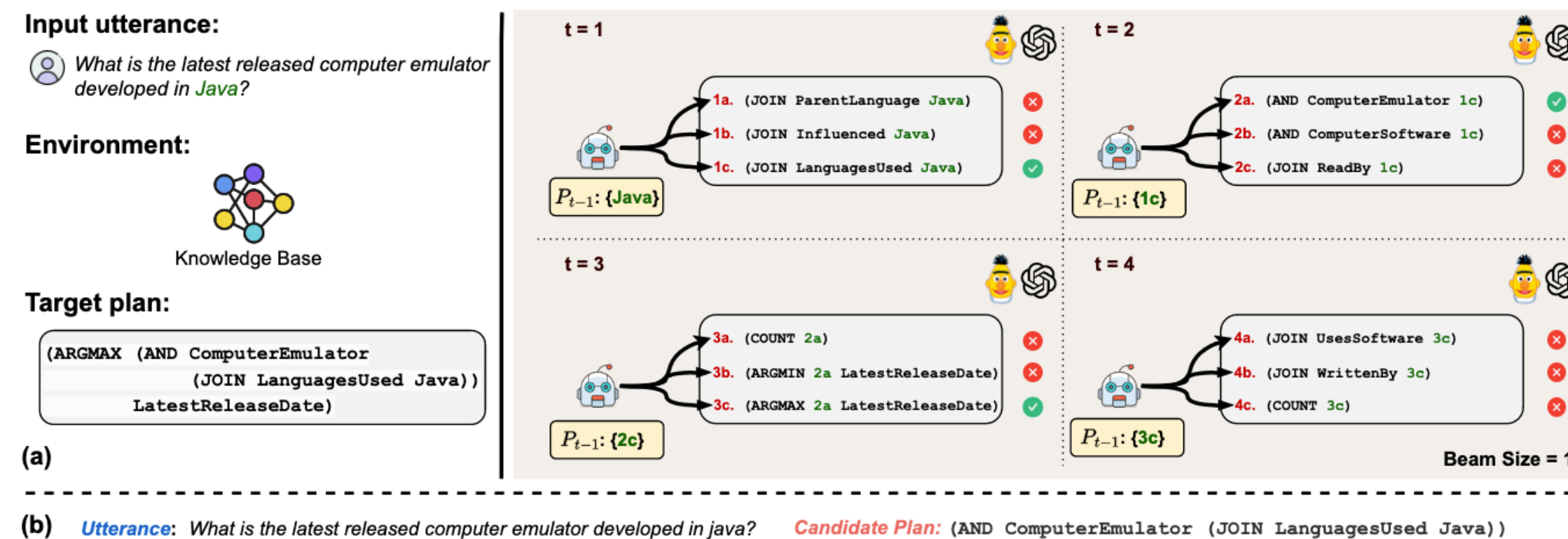
- 2 key changes that improves both speed and accuracy
 - Pruning at each step (improves speed by 8.33x)

Stage 3: Bottom-up Grounded Reasoning



- 2 key changes that improves both speed and accuracy
 - Pruning at each step (improves speed by 8.33x)
 - Inverse-consistency re-ranking (improves accuracy)

Stage 3: Bottom-up Grounded Reasoning



- 2 key changes that improves both speed and accuracy
 - Pruning at each step (improves speed by 8.33x)
 - Inverse-consistency re-ranking (improves accuracy)

Inverse-consistency is pretty general. You should try out for your generation experiments (esp. w/ smaller models)

Results

Datasets and Graphs

- **GrailQA:** 13,231 test questions containing questions up to 4-hops
 - **Freebase KG (Commons subset):** 3.7k relations, 1.5k classes, 32k entities
 - **MetaQA:** 39,093 test questions containing questions up to 3-hops
 - **MoviesKG:** 9 relations, 7 classes, 43k entities
 - **MatKG:** 100 test questions . Unseen KG, 21 relations, 7 classes, 70k entities
-

Metrics

- **F1-score**
 - **Answer-EM**
 - **Hits@1**
-

Models

- **Open-source:** MPT-7B
- **Closed-source:** GPT-3.5 (sub-sampled experiments)

Result #1 Exploration lead to substantial gain in unsupervised setting

Result #1 Exploration lead to substantial gain in unsupervised setting

Method	GrailQA (F1)	MetaQA (F1)
--------	-----------------	----------------

Result #1 Exploration lead to substantial gain in unsupervised setting

Method	GrailQA (F1)	MetaQA (F1)
Zero-shot	18.58	15.43

Result #1 Exploration lead to substantial gain in unsupervised setting

Method	GrailQA (F1)	MetaQA (F1)
Zero-shot	18.58	15.43
Pangu + exploration (10K)	42.44	54.68

Result #1 Exploration lead to substantial gain in unsupervised setting

Method	GrailQA (F1)	MetaQA (F1)
Zero-shot	18.58	15.43
Pangu + exploration (10K)	42.44	54.68
BYOKG + exploration (10K)	46.47	75.31

Results #2: Competitive results with supervised setting

Results #2: Competitive results with supervised setting

Method	GrailQA (F1)	MetaQA (F1)
BYOKG + training data (10K)	46.61	82.10
BYOKG + exploration (10K)	46.47	75.31

Results #3: Better consistency with splits


Results #3: Better consistency with splits

Method	IID	Compositional	Zero-shot	Overall
BYOKG + training data (10K)	58.29	45.14	41.89	46.61

Results #3: Better consistency with splits

Method	IID	Compositional	Zero-shot	Overall
BYOKG + training data (10K)	58.29	45.14	41.89	46.61


High Variance



Results #3: Better consistency with splits

Method	IID	Compositional	Zero-shot	Overall
BYOKG + training data (10K)	58.29	45.14	41.89	46.61
BYOKG + exploration (10K)	48.91	43.22	46.80	46.47

High Variance



Results #3: Better consistency with splits

Method	IID	Compositional	Zero-shot	Overall
BYOKG + training data (10K)	58.29	45.14	41.89	46.61
BYOKG + exploration (10K)	48.91	43.22	46.80	46.47

High Variance

Much lower variance consistent results

Results #3: Better consistency with splits

Method	IID	Compositional	Zero-shot	Overall
BYOKG + training data (10K)	58.29	45.14	41.89	46.61
BYOKG + exploration (10K)	48.91	43.22	46.80	46.47

High Variance

Much lower variance
consistent results

Everything is out-of-distribution for BYOKG => more consistent results

Results #4: BYOKG improves with model scale

Results #4: BYOKG improves with model scale

Method	IID	Compositional	Zero-shot	Overall
Pangu (Codex + Training data)	73.7	64.9	61.1	65.0

Results #4: BYOKG improves with model scale

Method	IID	Compositional	Zero-shot	Overall
Pangu (Codex + Training data)	73.7	64.9	61.1	65.0
BYOKG (GPT-3.5) + exploration (10K)	73.89	70.33	80.99	75.16

Results #4: BYOKG improves with model scale

Method	IID	Compositional	Zero-shot	Overall
Pangu (Codex + Training data)	73.7	64.9	61.1	65.0
BYOKG (GPT-3.5) + exploration (10K)	73.89	70.33	80.99	75.16

Performance increases with stronger LLMs and outperforms model using supervised training data

Case Study: Material Science KG

Case Study: Material Science KG

- ◆ Specialized domain of Material Sciences
- ◆ Totally unseen KG (released after pre-training of MPT models)
- ◆ No training data - we annotate 100 NL question

Case Study: Material Science KG

- ◆ Specialized domain of Material Sciences
- ◆ Totally unseen KG (released after pre-training of MPT models)
- ◆ No training data - we annotate 100 NL question

Method	Overall
Zero-shot	15.92

Case Study: Material Science KG

- ◆ Specialized domain of Material Sciences
- ◆ Totally unseen KG (released after pre-training of MPT models)
- ◆ No training data - we annotate 100 NL question

Method	Overall
Zero-shot	15.92
BYOKG + exploration (10K)	62.25

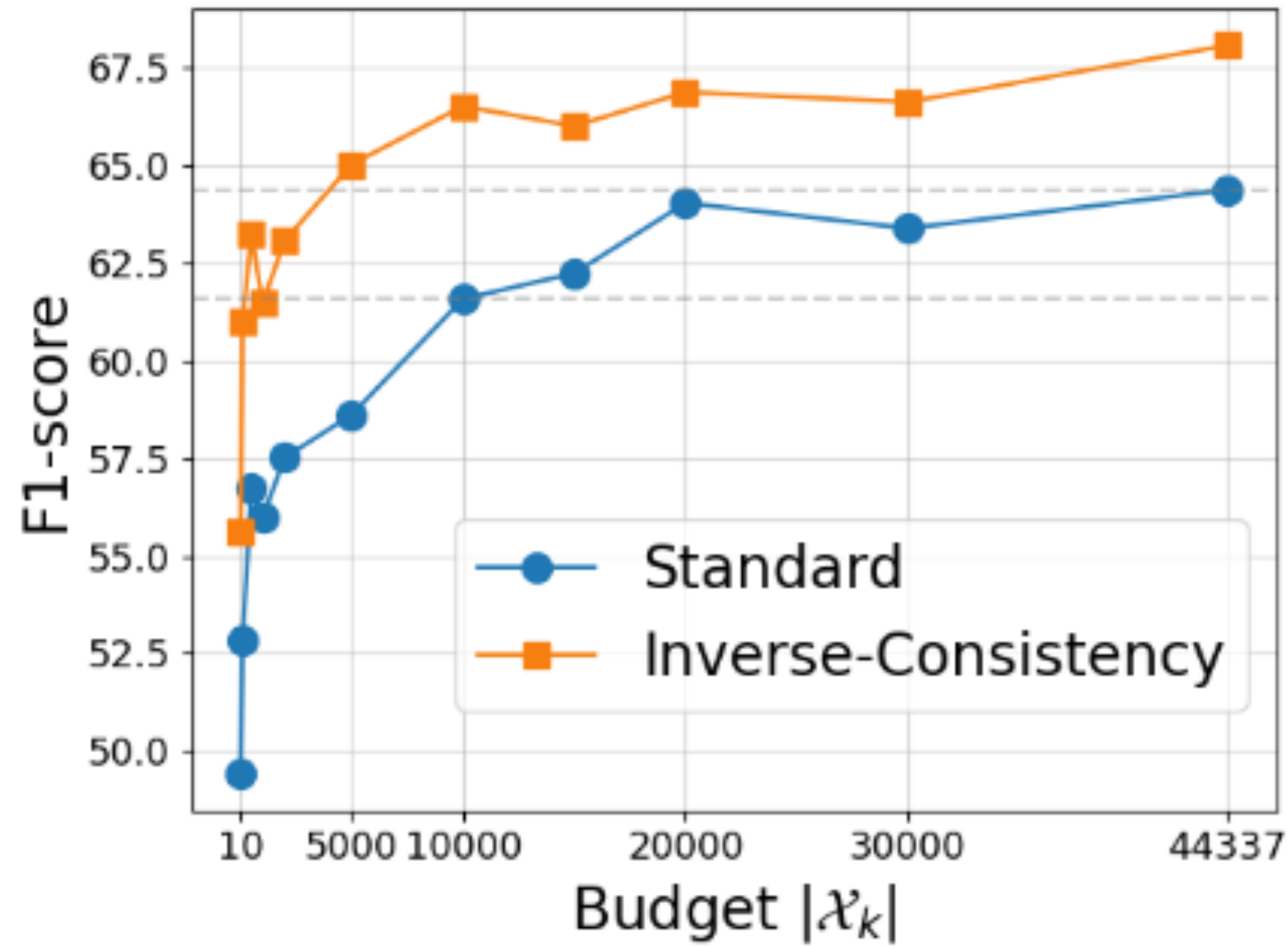
Result #5: Inverse Consistency is helpful

Result #5: Inverse Consistency is helpful

Metrics	Standard	Inverse-Consistency
ROUGE-1	48.17	52.81 ($\Delta+4.64$)
BLEU	31.54	38.63 ($\Delta+7.09$)
BERTscore	87.17	88.33 ($\Delta+1.16$)
Human Evaluation	47.50	70.00 ($\Delta+22.50$)

Table 7: **Inverse-Consistency for Question Generation.** Generation quality with inverse-consistency re-ranking compared with standard top-1 predictions from beam search using MPT-7B. Inverse-consistency improves generation quality as measured on both automatic and human evaluation metrics.

Accuracy v/s Exploration Budget



Summary & Future Directions

Summary & Future Directions

- ◆ ~~Build~~ Built a natural language interface for querying KGs that
 - ☑ Works for *Any Knowledge Graph* (irrespective of their schema)
 - ☑ Requires *no training data* (i.e. completely unsupervised)
 - ☑ Ready to be used in *reasonable time* (~1 day)

Summary & Future Directions

- ◆ ~~Build~~ Built a natural language interface for querying KGs that
 - ☑ Works for *Any Knowledge Graph* (irrespective of their schema)
 - ☑ Requires *no training data* (i.e. completely unsupervised)
 - ☑ Ready to be used in *reasonable time* (~1 day)
- ◆ Extend this idea to other structured data (databases, web tables, CSVs)
- ◆ Further reduce latency

Summary & Future Directions

- ◆ ~~Build~~ Built a natural language interface for querying KGs that
 - ☑ Works for *Any Knowledge Graph* (irrespective of their schema)
 - ☑ Requires *no training data* (i.e. completely unsupervised)
 - ☑ Ready to be used in *reasonable time* (~1 day)
- ◆ Extend this idea to other structured data (databases, web tables, CSVs)
- ◆ Further reduce latency

Thank you!



Code and data: <https://github.com/amazon-science/BYOKG-NAACL24>