Comparing the Impact of Financial Knowledge Graphs from Financial Reports and Wikidata in Asset Recommendation

Lubingzhi Guo , Javier Sanz-Cruzado , Richard McCreadie

Fin-RecSys 2024

Motivation

- Financial Market Investors
  - Goal: Earn money
  - Challenge: Identifying good assets is difficult and time consuming

- Asset Recommendation
  - Supporting tools to quickly identify assets likely to provide market-beating returns
  - Automatically rank financial assets based on past market information

Financial Asset Recommendation

- The majority of prior solutions for FAR focus on either the market or news, e.g.
  - Past pricing data
  - Financial news articles
  - Social media

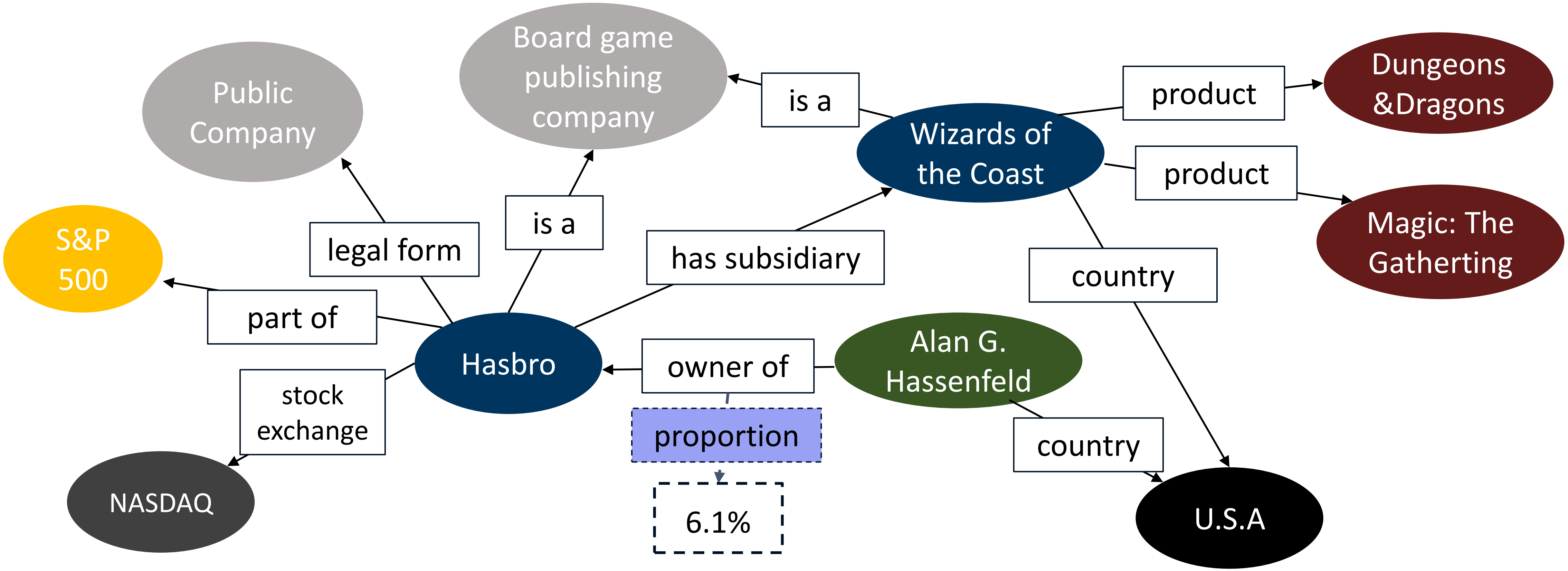
- However, **important fundamental information about each individual company is often ignored**, such as:
  - Company operations
  - The people that manage or lead the company
  - Relationships with other companies
  - Product launch events
  - … and much more

By leveraging these information, we hypothesize that the accuracy of profitability prediction can be increased

But how can we model this information if we can get it?

Financial Knowledge Graph

- A **financial knowledge graph (KG)** is a data structure that can be used to store such information, where nodes represent entities (e.g. companies, people), while edges represent relations between them (e.g. owner of).



Financial Knowledge Graph

KG Construction

General Knowledge Bases

- Produced from structured databases like Wikidata
- Provides high-quality general information

Financial Documents

- Produced from relevant news and financial reports
- Provides insights into financially-related data and trends

No prior works have quantitatively compared these two different strategies

Knowledge Gap

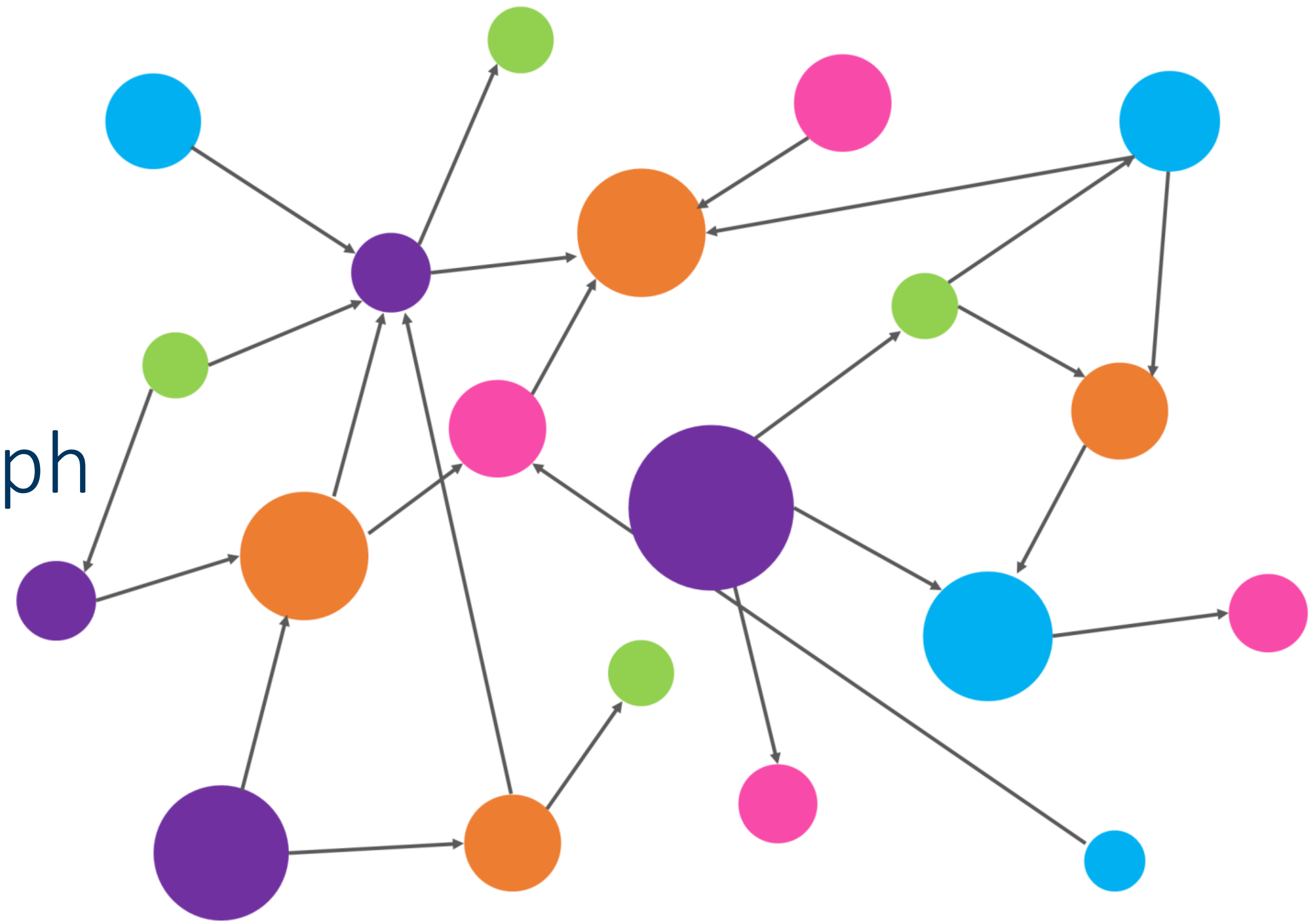
- Graphs produced by different strategies are likely to benefit different types of assets being recommended
  - ➢General knowledge bases with imbalanced coverage toward well-known/long standing companies
  - ➢Financial documents cover more up-to-date and newsworthy information

- Our Contributions
  - Crawl a financial KG from Wikidata
  - Construct a financial KG from 10K reports
  - Compare the impact these two KGs have when predicting the future profitability of U.S. stocks.

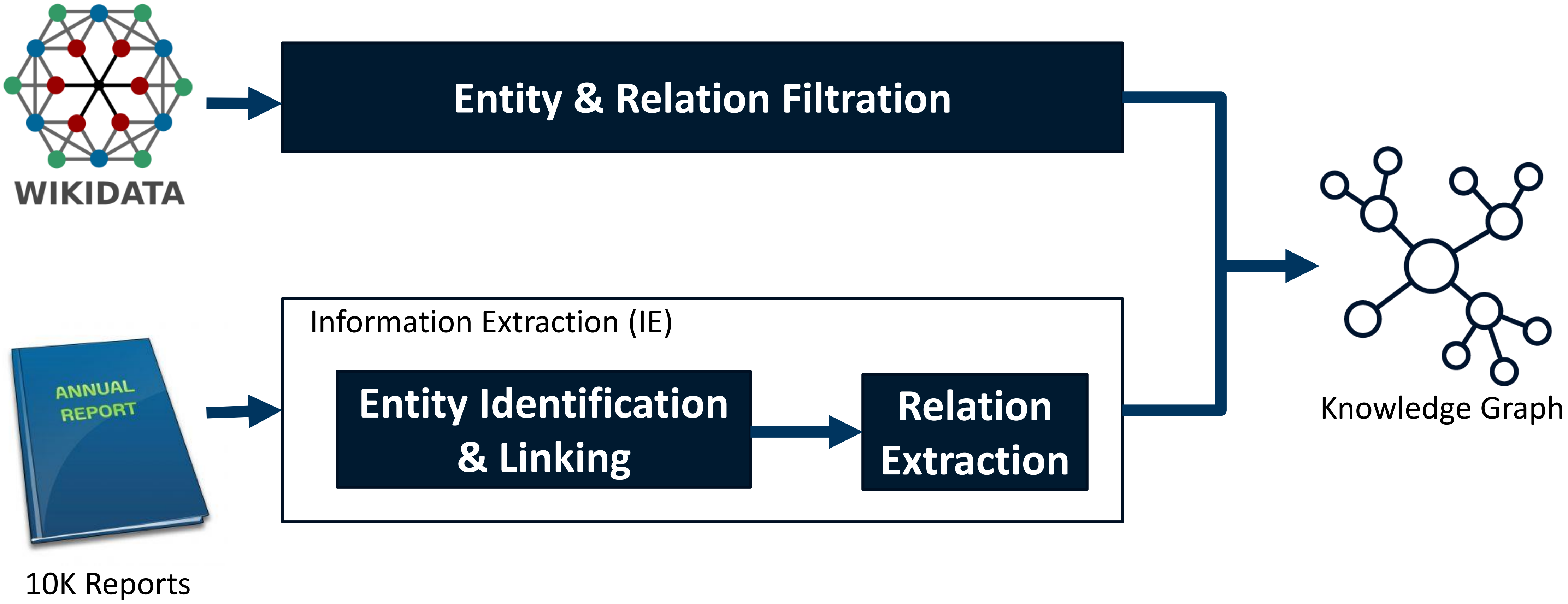
General Knowledge Base

Annual reports of public traded companies submitted to the U.S. Securities and Exchange Commission

Financial Knowledge Graph Construction

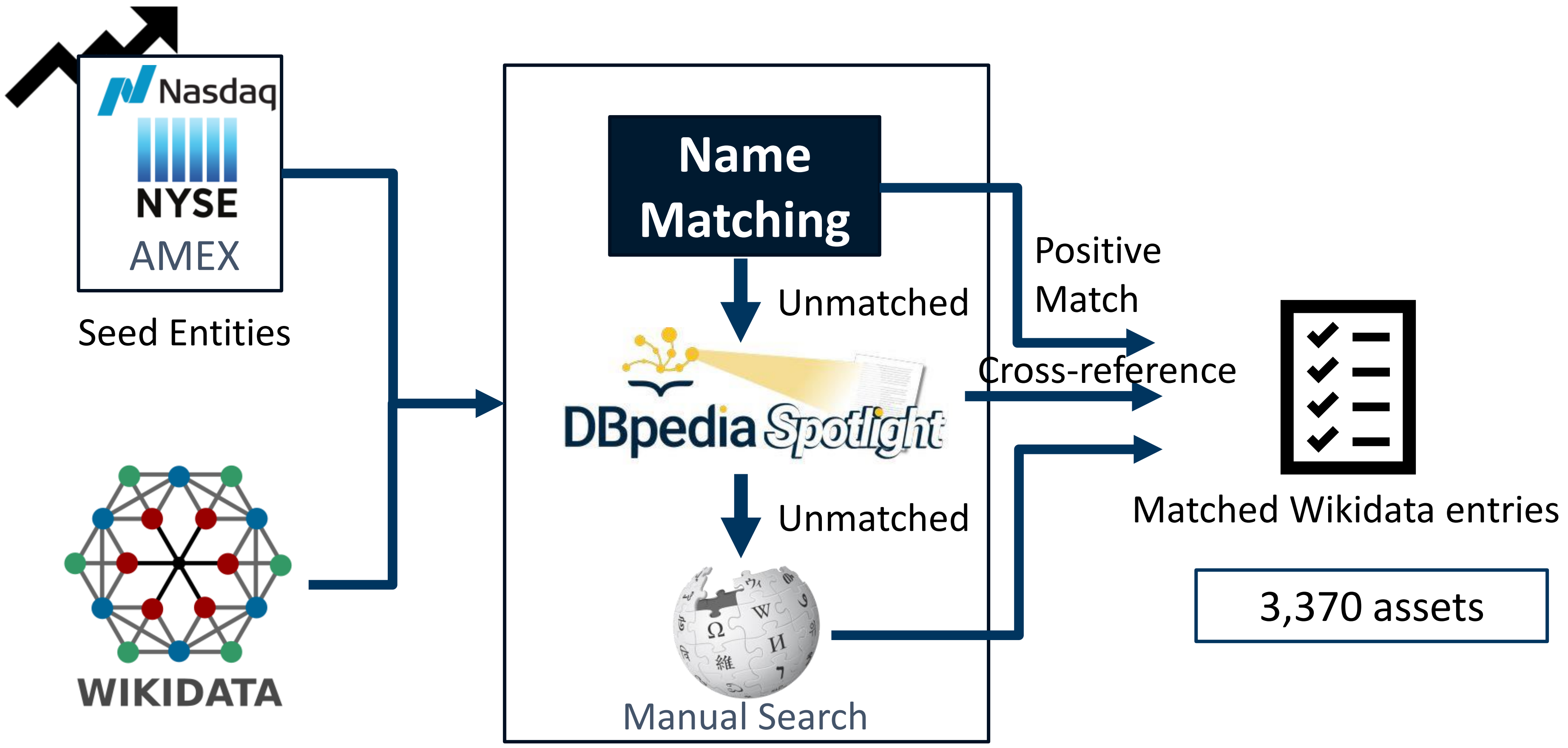


Graph Generation



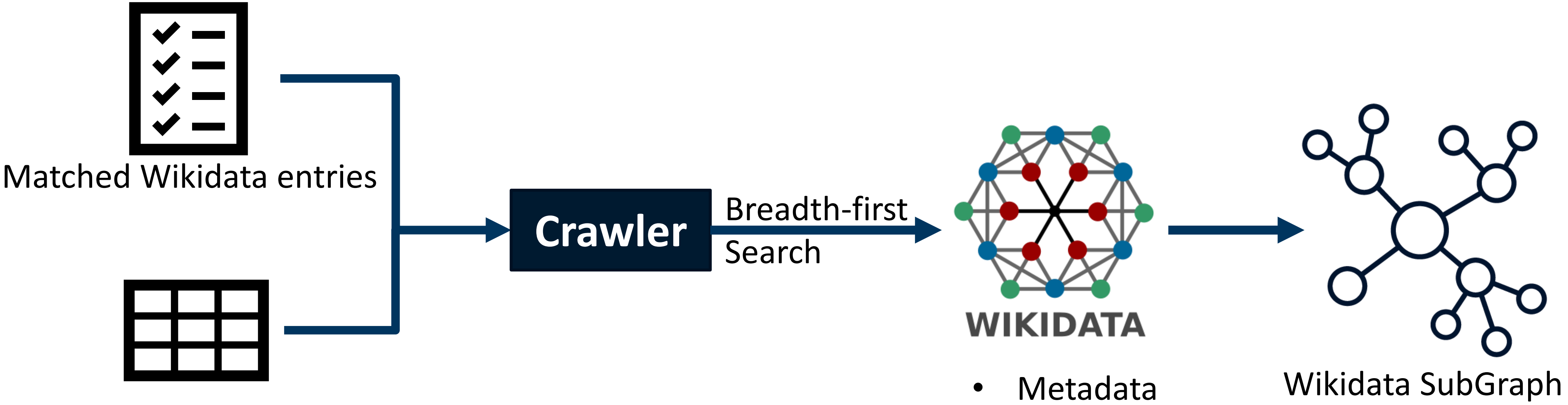
Wikidata SubGraph Construction

- **Seed Entity Matching**



Wikidata SubGraph Construction

- **Entity and Relation Filtration**



Filtering rules

- Valid financial relations
  - Ownership, Employment, Creation, …
- Valid entity types
  - Organization, Market, Product., …
- Valid qualifiers:
  - Time, Position, Location

- Metadata
- Relations
- Properties(any available temporal information)

KG Construction from 10K Reports

Information Extraction

- Graph Definition
  - Hyper-relational fact: (Head Entity, Relation, Tail Entity, (Qualifier Label, Qualifier Value))

- Extract Information from Raw Text
  - Most studies focus on closed-domain relation extraction
  - A limited set of predefined relations is not sufficient for represent complex relations in the financial documents
  - On the other hand, open-domain relation extraction identifies all possible relations from texts, extending beyond financial domain

10K Reports Graph

INFORMATION RETRIEVAL Group

- Semi-open Hyper-relational Financial Facts Extraction
  - Bridge the closed-domain and open-domain methods
  - No specified relation tags needed
  - Provide guidelines to guide the model to identify business, transaction or personnel-related relationships

- Facts Extraction Example

On **April 25 , 2014** , **Microsoft acquired** substantially all of **Nokia Corporation** ’s Devices and Services business (“ NDS ”)

Subject Organization	Microsoft
Object Organization	Nokia Corporation
Relationship	acquired
Point in time	April 25, 2014

GoLLIE[1]

Adapt GoLLIE for the financial facts extraction task

Represents both input and output using python classes

Outperforms on zero-shot Information Extraction task across various domains

Allows the user to perform inferences with custom annotation schemas

[1] Oscar Sainz, Iker Garc´ıa-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. GoLLIE: Annotation guidelines improve zero-shot information-extraction. In The 12th International Conference on Learning Representations (ICLR 2024), Vienna, Austria, 2024. OpenReview

An Example of Business Event Schema

INFORMATION RETRIEVAL Group

- Event Argument Extraction

Event type introduced as class name

Guidelines are introduced as docstrings

Trigger text span serves as relation

Representative candidates are introduced as comments

Allows to match multiple patterns

```
class BusinessEvent(Event):
 """A BusinessEvent refers to actions related to Organizations such as: creating, merging, acquiring,
owning another orga cy or ending organizations (inc
agencies)."""
        mention: str
        """The text span that most clearly expresses the event. Such as: "started", "open", "create",
"closing", "merged" """
        subject_organization: str organization in the event.
        object_organization: Optional[str] = None # Receiver or secondary organization in the event.
        location: Optional[str] = None # Where the event takes place
        point_in_time: Optional[str] = None # Descriptive or vague moment of the event, not
intended for direct comparison with start_time or end_time
        start_time: Optional[datetime] = None # The precise starting datetime of the event
        end_time: Optional[datetime] = None # The precise ending datetime of the event
```

Output

- The output annotations are represented as a list of instances
- well-structured and easy to parse

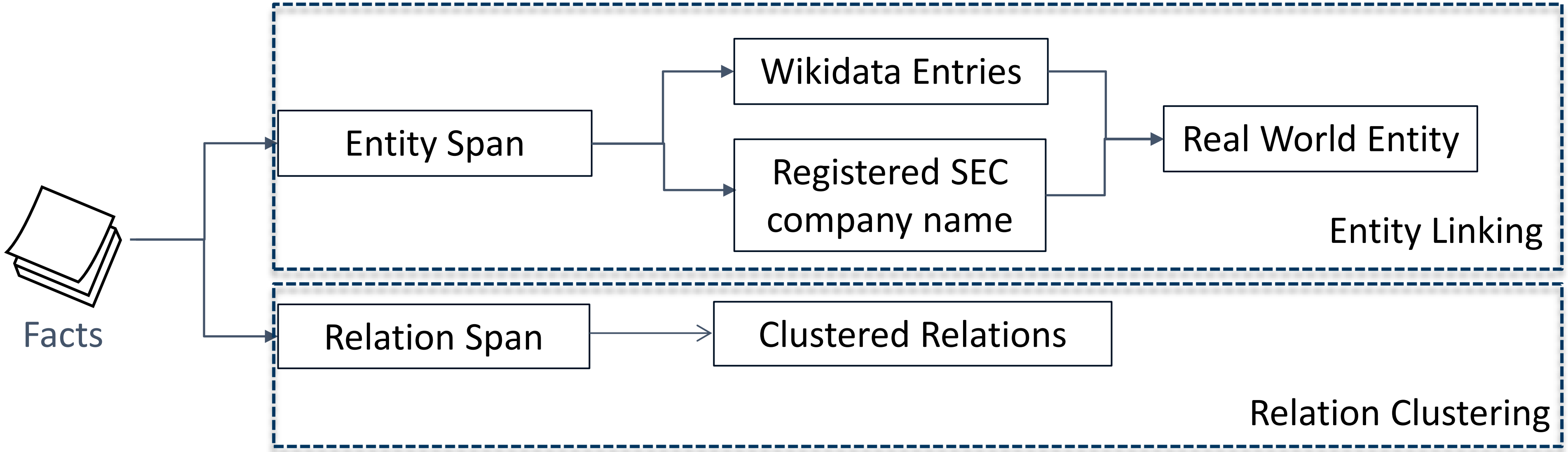
```
# This is the text to analyze
text = "On April 25 , 2014 , Microsoft acquired substantially all of Nokia Corporation ’s Devices and Services business ( “ NDS ” )"

# The annotation instances that take place in the sentence above are listed here
results = [
    BusinessEvent(mention = "acquired",
                  subject_organization = "Microsoft",
                  object_organization = "Nokia Corporation",
                  point_in_time = "April 25, 2014")
]
```

10K Reports Graph

- Construction Pipeline

- Postprocessing



Graph Properties

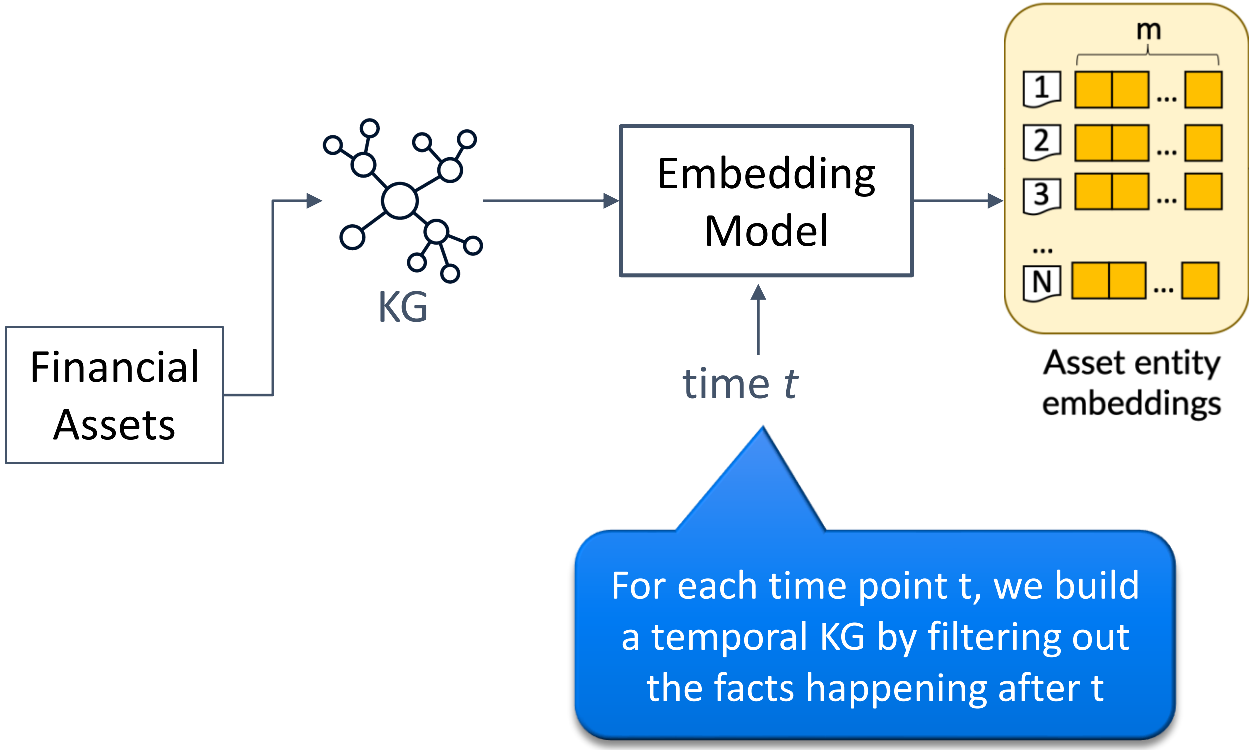
Property	Wikidata	10K
Number of entities	102,739	8,380
Number of relation types	114	450
Number of links	457,758	36,973

- Filtered Wikidata Graph
  - Large
  - Reliable
  - Static
  - General
- Extracted 10K Graph
  - Small
  - Noisy
  - Up-to-date
  - Financially-related

Financial Asset Recommendation

Encode Knowledge Graphs

- Knowledge Graph Embedding (KGE)
  - ➢ Aim to encode the information in the knowledge graph into a low dimensional space
  - ➢ Entity embeddings are computed considering the related assets and relationships



- KGE Models

  Translation-based (Popular)

  TransE, TransH, TransR, RotateE

  Factorization-based

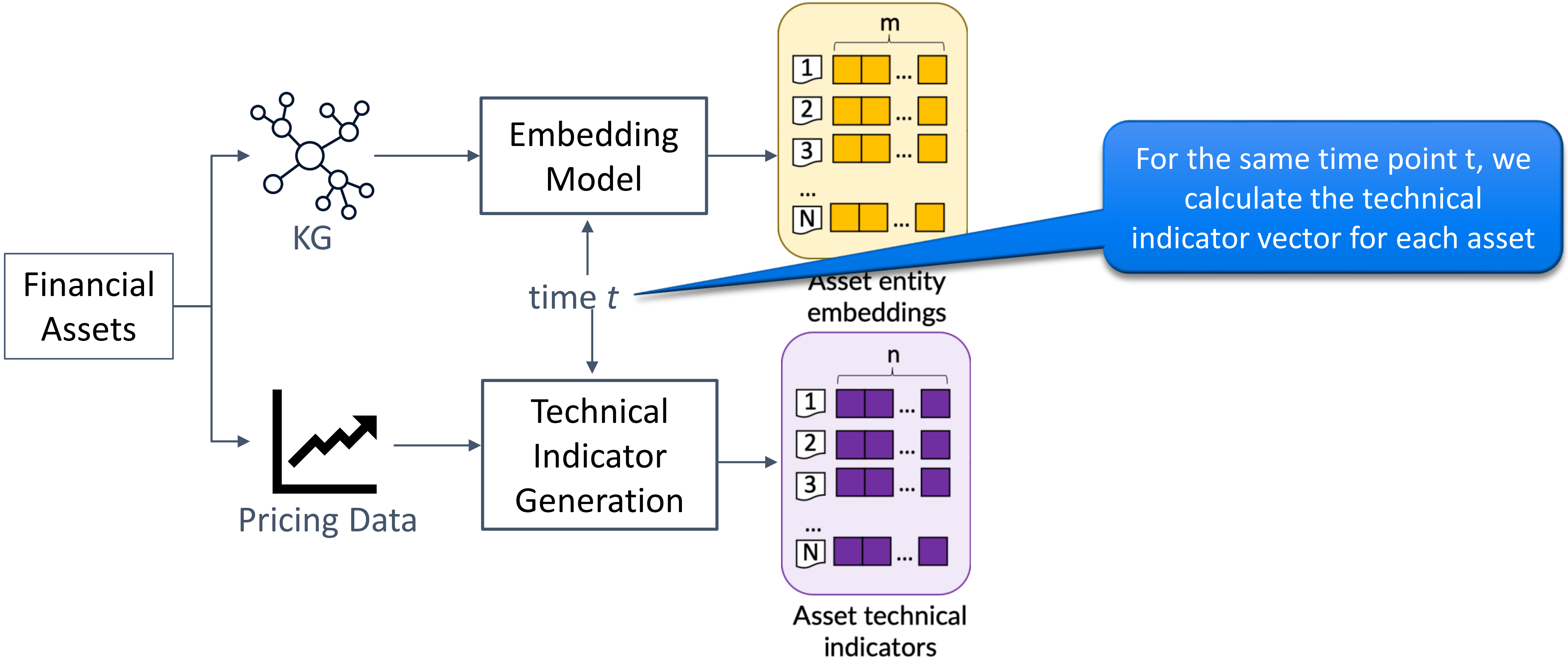
  RESCAL, HolE, TuckER

  Neural network-based (state-of-art)

  ConvE, RGCN

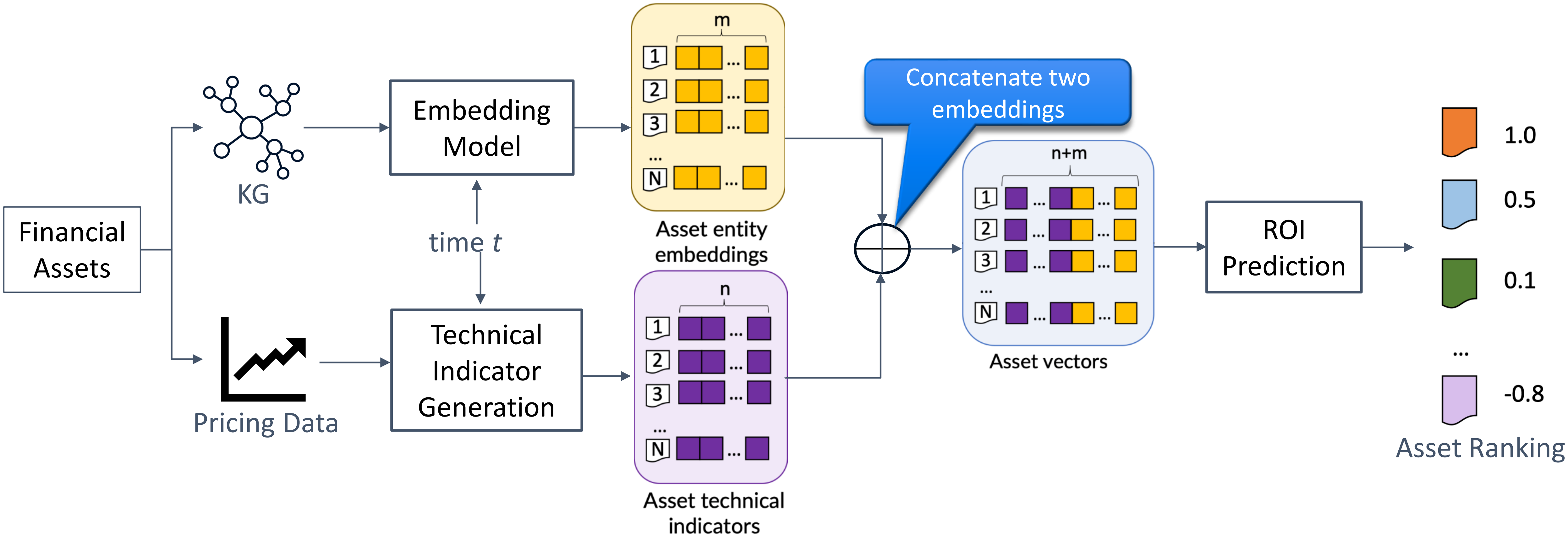
Incorporate Pricing Information

- Technical Indicators
  - Key performance indicators (KPIs) that encode some aspect of the past pricing information of a financial asset
  - Average price, Return on investment, Volatility



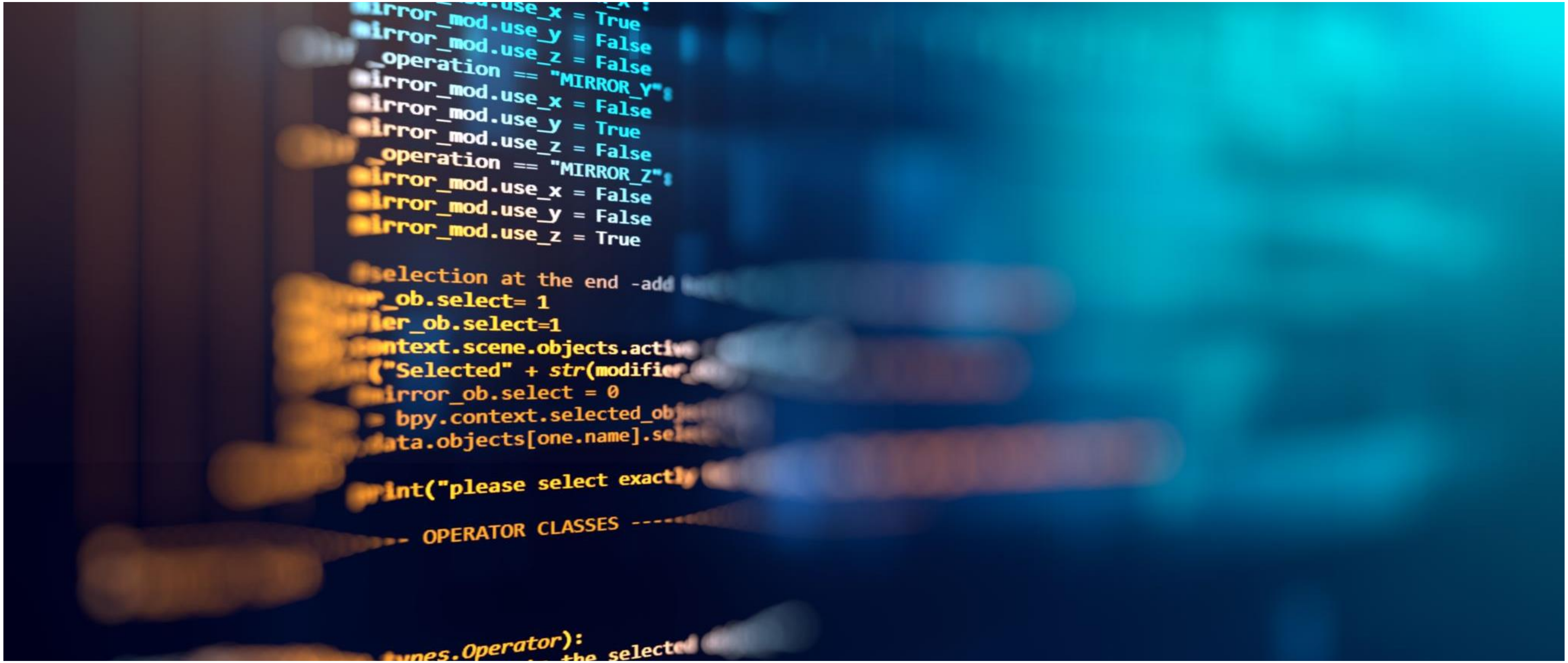
Asset Recommendation Pipeline

- Return on Investment Prediction
  - ➢ Train a random forest regressor to predict the future profitability of the assets
  - ➢ Rank assets by the estimated profitability in descending order



Experimental Setup

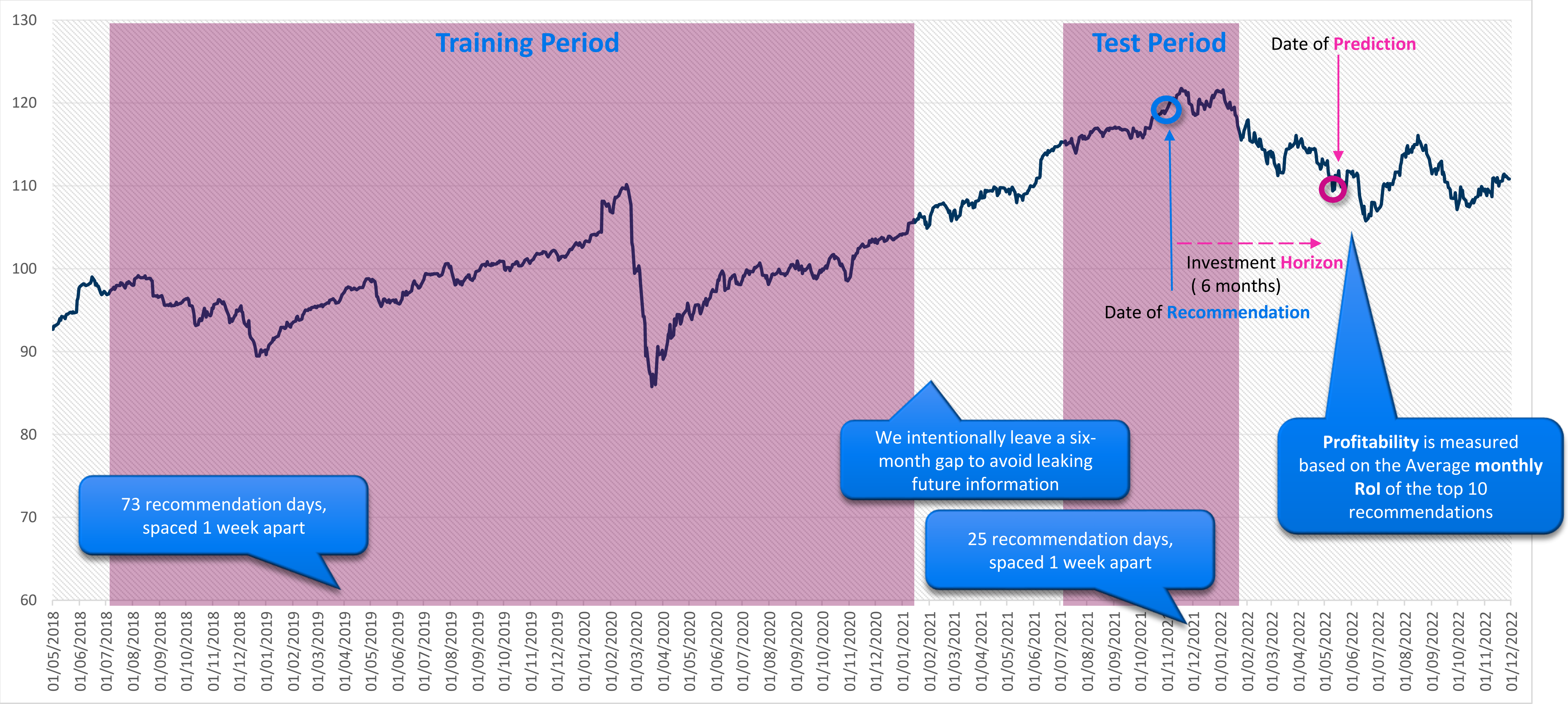
Information Retrieval Group



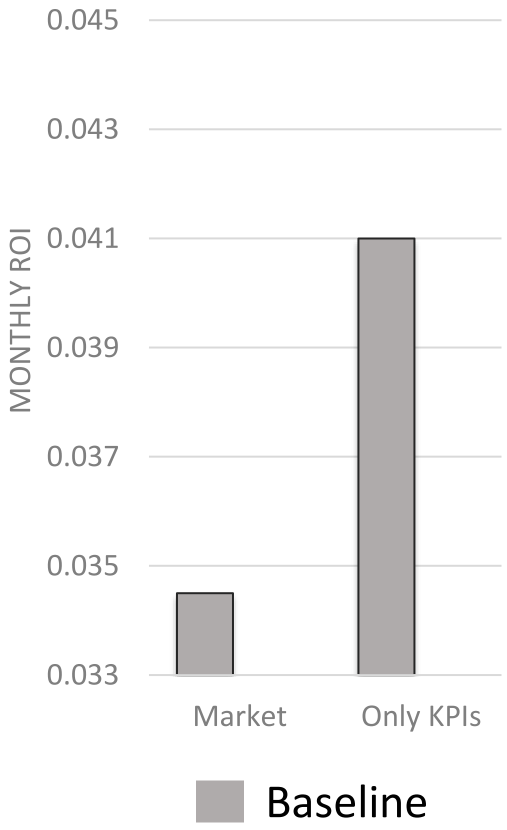
Research Questions

RQ1	How effective are knowledge graphs in predicting profitability?
RQ2	How do recommended assets differ across knowledge graphs?

Training Regime

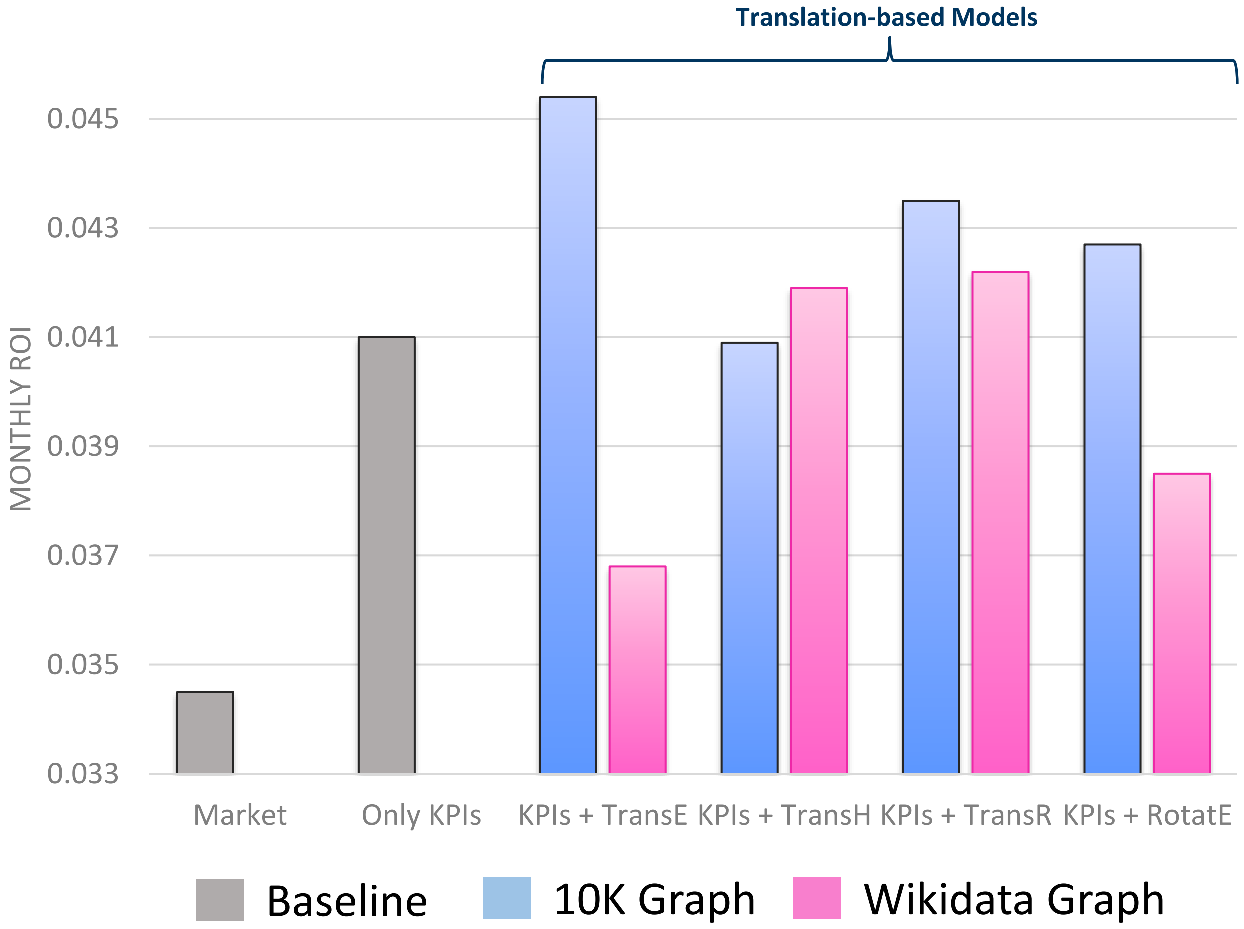


Establishing the Baseline



- Baselines
  - ➢ The market average
  - ➢ Only KPIs
    - ➢ Models only use technical indicators to predict the future profitability of an asset

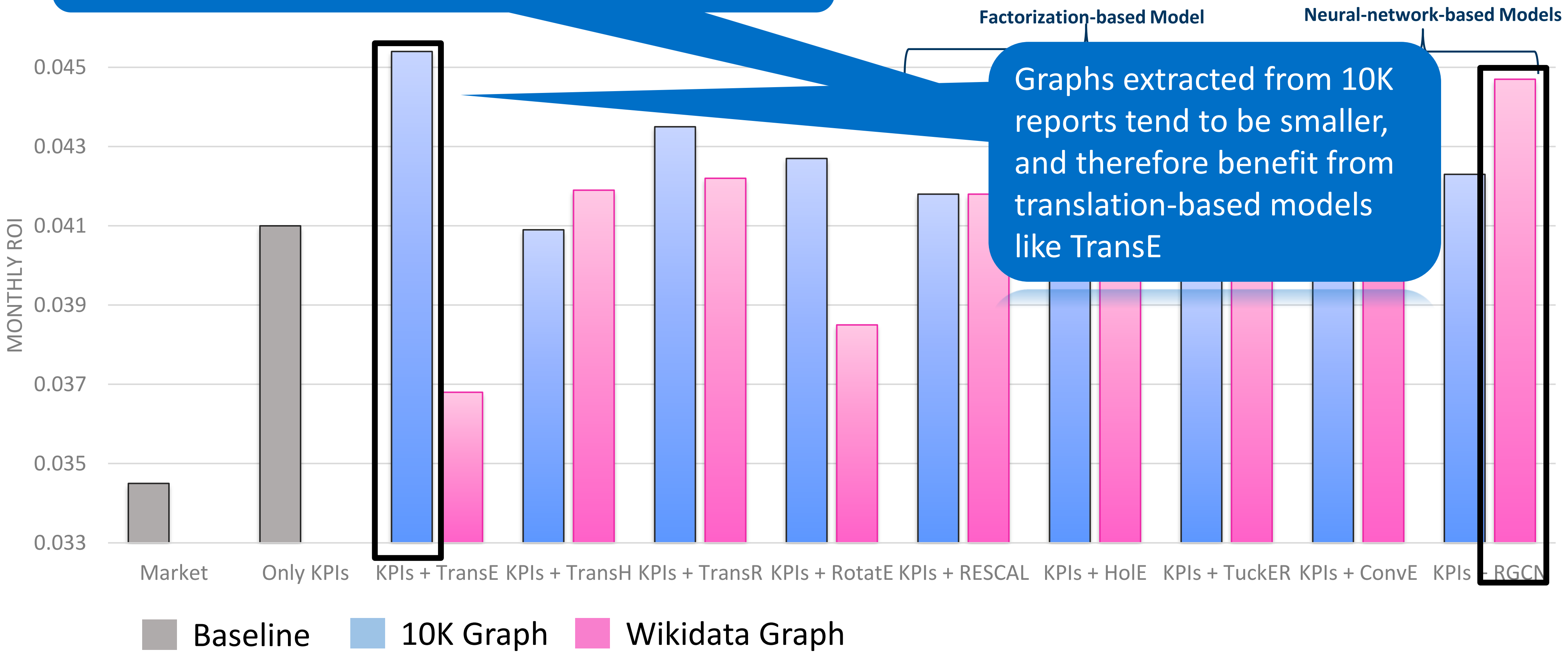
RQ1:Integrating KGE for Profitability Prediction



RQ1:Integrating KGE for Profitability Prediction

The bigger Wikidata graph favors complex neural network models

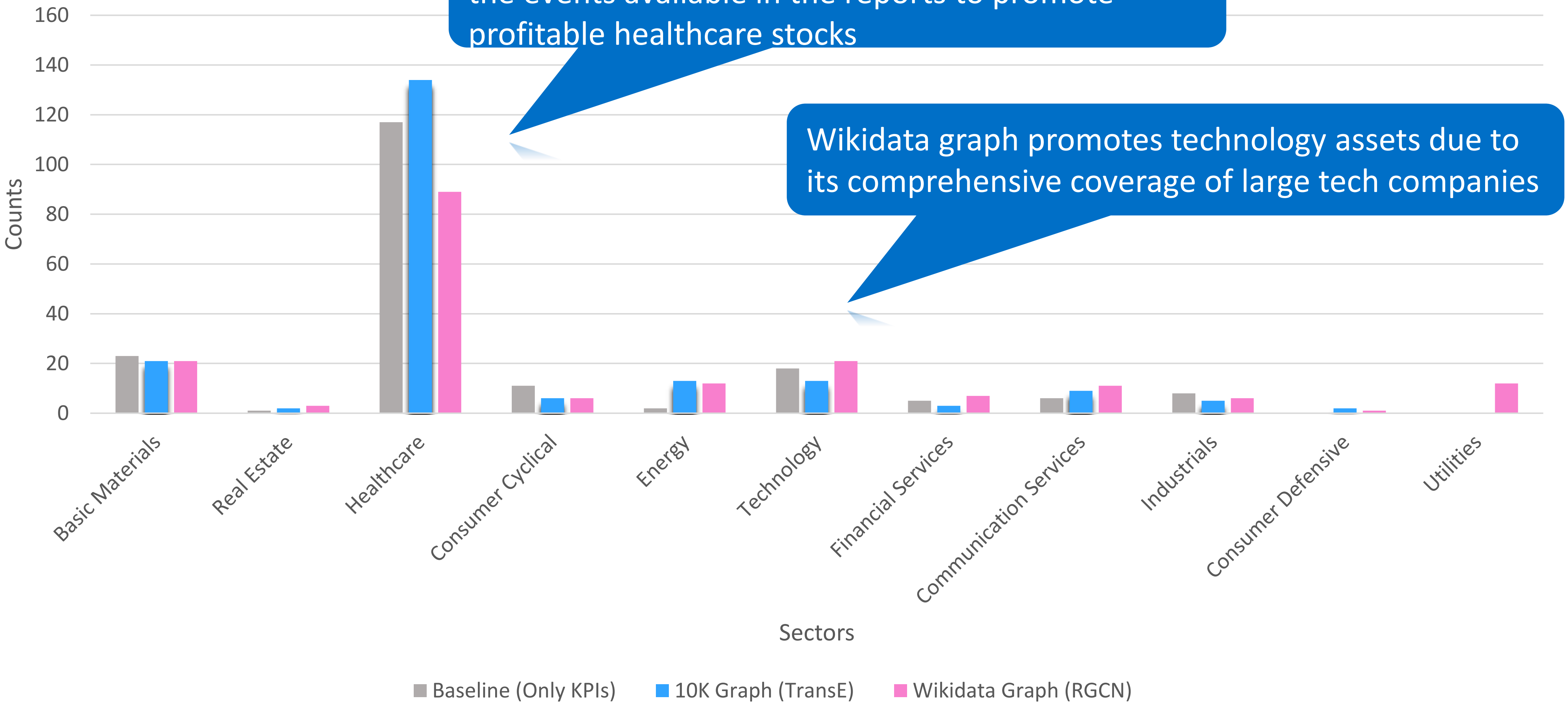
Graphs extracted from 10K reports tend to be smaller, and therefore benefit from translation-based models like TransE



RQ2: How do recommended assets differ?

10K graph can leverage the information regarding the events available in the reports to promote profitable healthcare stocks

Wikidata graph promotes technology assets due to its comprehensive coverage of large tech companies



Conclusions

- Both graphs can improve the profitability of recommendations with respect to the method only using technical indicators by up-to 10%
- Should be careful to pick the embedding methods for incorporating knowledge graph
- Different construction methods can result in promoting different types of assets

Information Retrieval Group

Thanks & Question