

Relational Learning in Pre-Trained Models: A Theory from Hypergraph Recovery Perspective

Anonymous Authors¹

Abstract

Foundation Models (FMs) have demonstrated remarkable insights into the relational dynamics of the world, leading to the crucial question: *how do these models acquire an understanding of world hybrid relations?* Traditional statistical learning, particularly for prediction problems, may overlook the rich and inherently structured information from the data, especially regarding the relationships between objects. We introduce a mathematical model that formalizes relational learning as hypergraph recovery to study pre-training of FMs. In our framework, the world is represented as a hypergraph, with data abstracted as random samples from hyperedges. We theoretically examine the feasibility of a Pre-Trained Model (PTM) to recover this hypergraph and analyze the data efficiency in a minimax near-optimal style. By integrating rich graph theories into the realm of PTMs, our mathematical framework offers powerful tools for an in-depth understanding of pre-training from a unique perspective and can be used under various scenarios. As an example, we extend the framework to entity alignment in multimodal learning.

1. Introduction

Foundation Models (FMs) (Bommasani et al., 2021; OpenAI, 2023) have emerged as transformative forces in the realm of artificial intelligence, demonstrating impressive performance in various real-world tasks such as knowledge retrieval (Liu et al., 2023), mathematics problem solving (Frieder et al., 2023), coding (Zhang et al., 2022), common-sense reasoning (Rajani et al., 2019; Zhao et al., 2023b), and text-to-image generation (Ramesh et al., 2021; Li et al., 2023b). During interactions with humans, FMs seem to ex-

hibit an understanding of real-world entities to a certain degree, engaging in reasoning based on these entities (Bubeck et al., 2023). For example, FMs can deduce the entity “table” from descriptions of objects placed on it, such as a cup, book, or computer, which raises a fundamental question: *how do FMs learn real-world entities from pre-training?*

To investigate the learning of entities via pre-training, a formidable challenge is to formalize how the relationships between the entities are learned from data. Traditional statistical learning, such as PAC (Valiant, 1984; Mohri et al., 2018), particularly in classification problems, typically treats data as pairs of objects and their corresponding labels, focusing primarily on predicting these absolute labels. However, this approach may overlook the richer, more nuanced information that data inherently carry, especially regarding the relationships between objects. For instance, an image of a camel does not just represent the animal; it may also encapsulate its context, like a desert background, offering deeper relational insights on the camel and the context objects. Similarly, in natural language processing, the meaning of a sentence transcends the mere sum of its words, revealing complex interdependencies between the entities represented by the words. At the same time, PTMs, such as LLMs, often respond to complex relationships between objects. Recognizing this, a new mathematical model is essential to capture these critical, yet often overlooked, facets of relational learning in pre-training, crucial for understanding the capabilities and generalization of the PTMs.

In this work, we propose a novel mathematical framework based on hypergraph recovery to more fully capture the essence of relational learning. Specifically, we abstract the world as a hypergraph: entities are nodes, and relationships between entities are hyperedges. Each hyperedge is assigned a weight, signifying the strength of the corresponding relation. We formulate relational learning from pre-training as hypergraph recovery of the world hypergraph using the information of data. We model data generation as random sampling from the hyperedges. This data generation process mirrors real-world data collection, where a sample represents a perception of a relation between entities, with stronger relations having a higher likelihood of being observed and recorded. Our framework presents two-fold

¹Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

advantages: 1) In contrast to traditional statistical learning, our framework adopts a more nuanced approach. It goes beyond merely capturing individual labels within each data sample, delving into the interrelations between entities. This method yields a richer and more holistic understanding of relational learning in pre-training scenarios. 2) Additionally, the framework integrates rich graph theories into the field of PTMs. This integration invokes powerful analytical tools, providing a novel perspective for relational learning.

Based on the framework, we can answer two important questions about relational learning in PTMs: 1) Identification: Does the data provide sufficient information for relational learning? 2) Data efficiency: If so, what is the essential amount of data required? For the first question, we approach it as an estimation problem within a hypergraph framework and give an affirmative answer by demonstrating that the hypergraph can be identified from sufficient hyperedge samples. To address the second question, we first establish a lower bound $\Omega(\frac{m}{\epsilon^2})$ for ϵ -approximate relational learning of the hypergraph with m hyperedges. We further investigate how a model learns relations via Masked Modeling (MM), a common practical pre-training algorithm (Kenton & Toutanova, 2019; He et al., 2022). In the hypergraph recovery framework, an MM PTM learns a set of relative weight ratios between certain entity relations. We show that MM achieves the near-optimal (in terms of approximation error) sample complexity $\tilde{O}(\frac{m}{\epsilon^2})$, matching the information theoretical lower bound if logarithmic factors are neglected.

Our hypergraph framework is adaptable to scenarios necessitating the capture of entity relations, including multimodal entity alignment (Chen et al., 2020; Zhao et al., 2023a), social network privacy (Korolova et al., 2008), and relational reinforcement learning (Zambaldi et al., 2018a), etc., allowing for an analysis of key relational learning from pre-training data. We focus on multimodal entity alignment, demonstrating feasible alignment across modalities using sufficient unlabeled data, achieved through hypergraph matching. Although aligning without labeled pairs is theoretically possible, practical computational constraints necessitate labeled pairs to reduce complexity.

We conduct experiments to back up the validity of our hypergraph formulation for relational learning in PTMs. In the first experiment of synthetic relational learning, we create synthetic entities whose relations compose weighted graphs, showing the power of MM for learning the synthetic relations. In the second experiment, we examine real-world relational learning of LLMs by evaluating their relational subgraphs and measuring how well the evaluated subgraphs align with the real world. Our results show that the evaluated relations do align with the real world to some degree and more powerful models exhibit better alignment.

We list the contributions of the paper as follows:

- We propose a new mathematical model to formalize relational learning in PTMs, which is grounded in the principles of hypergraph recovery.
- We demonstrate the feasibility of a learning model achieving relational learning and establish a minimax lower bound for the sample complexity involved. Additionally, we show that pre-training using Masked Modeling (MM) approaches near-optimal data efficiency in terms of approximation error within our framework.
- We extend our framework to entity alignment in multimodal learning. We show the feasibility of entity alignment without labeled pairs and demonstrate the role of labeled pairs in reducing the computational complexity.

2. Related Work

Graph Models. Graphs have long been used to characterize structures of data. For instances, parsing graphs use graphs to represent the grammatical dependencies of text, (Chomsky, 2014; Chen & Manning, 2014; Hewitt & Manning, 2019). Semantic networks model the semantic relationships between words and entities by graphical representations (Miller, 1995; Speer et al., 2017). Knowledge graphs represent knowledge as entities and complex relationships within graphs (Suchanek et al., 2007; Lin et al., 2015; Dettmers et al., 2018). Following a similar philosophy, we model the concepts and the relations in the world as a weighted hypergraph and pre-training data as samples of hyperedges from the hypergraph. Our formulation is, instead, a simplified mathematical model to explain how pre-training can learn the complex relations in the world.

Combinatorial Statistics. Combinatorial statistics studies the statistical properties of data with discrete structures. The most related topic in combinatorial statistics to this work is random graph isomorphism. These works model real-world problems, namely, DNA shotgun assembly (Idury & Waterman, 1995), protein matching (Zaslavskiy et al., 2009), social network privacy (Korolova et al., 2008), etc., by random graph problems such as shotgun assembly (Mossel & Ross, 2017; Ding et al., 2023) and random graph matching (Cullina & Kiyavash, 2016; Barak et al., 2019; Ding et al., 2021), exploiting both the combinatorial and statistical properties of the data. Our work takes a step to build the connections between combinatorial statistics and PTM capabilities, harnessing mathematical tools from the former to enhance our understanding of PTMs.

Relational Learning. Relational learning focuses on identifying the relationships among entities (Struyf & Blookeyl, 2010). To understand and exploit the relational structure of data, various relational learning techniques and methods are employed, including inductive logic programming

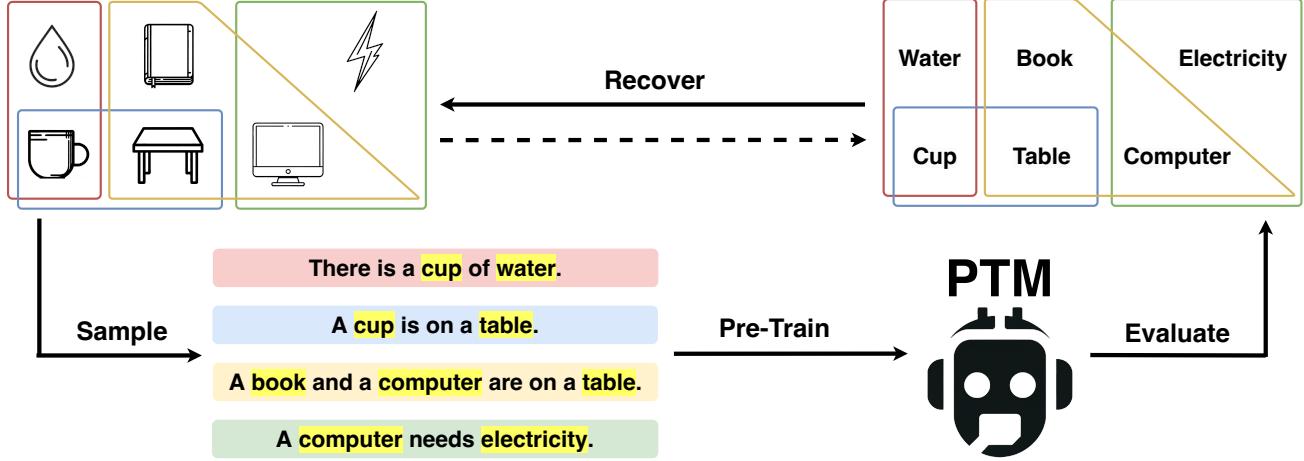


Figure 1. Our hypergraph recovery framework for relational learning in PTMs. The relational model of the world is viewed as a hypergraph. Data are generated by sampling hyperedges from the world relational model and mapping them to perception domains. PTMs learn the entity relations from the data. Recovered relational hypergraphs can be evaluated from the PTMs.

(De Raedt, 2008), probabilistic logic learning (De Raedt & Kersting, 2008), relational reinforcement learning (Džeroski et al., 2001; Zambaldi et al., 2018b), graph neural networks (Chen et al., 2021; Fey et al., 2023), etc. While these works aim to capture entity relations more precisely, our research is dedicated to exploring the emergence of relational learning from pre-training in theory.

Theories of PTMs. Various theoretical frameworks have been proposed to elucidate the mechanisms by which PTMs leverage pre-training data and tasks to achieve generalization. Multi-task learning suggests that PTMs acquire generalizable representations through simultaneous training on diverse tasks (Ando et al., 2005; Xie et al., 2020; Hu et al., 2021; Chen et al., 2022; Yang et al., 2022), under the assumption that these representations are the invariant components across the various tasks. Meta-learning posits that PTMs develop the ability to learn efficiently, postulating that certain meta parameters exist that enable fast adaptation to new tasks, with optimization processes geared towards these meta parameters (Finn et al., 2017; 2018; Tripuraneni et al., 2021). In certain in-context learning scenarios, some in-context learning theories propose that PTMs internalize optimization or learning algorithms, facilitating task and distribution generalization (Akyürek et al., 2022; Li et al., 2023a; Von Oswald et al., 2023). This work diverges by explicitly modeling generalizable knowledge as a relational hypergraph of the world, framing pre-training as a process of hypergraph recovery.

3. Preliminary

Hypergraph. A hypergraph \mathcal{H} is a tuple $(\mathcal{V}, \mathcal{E})$ where \mathcal{V} is a finite set called *nodes* and \mathcal{E} is a family of subsets of \mathcal{V}

called *hyperedges* (Bretto, 2013). A weighted hypergraph \mathcal{H} , denoted by a tuple $(\mathcal{V}, \mathcal{E}, w)$, is a hypergraph equipped with an additional weight function $w : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$. The line graph of the hypergraph \mathcal{H} , denoted by $L(\mathcal{H})$, is the graph whose node set is the set of the hyperedges of \mathcal{H} and edge set is the set of pairs of the hyperedges that intersect. Consider transformations between hypergraphs. Suppose that $\phi : \mathcal{V} \mapsto \mathcal{V}'$ is a bijection from \mathcal{V} to a set of nodes \mathcal{V}' . For a hyperedge $e = \{v_1, \dots, v_k\}$, we use $\phi(e)$ to denote the hyperedge $\{\phi(v_1), \dots, \phi(v_k)\}$. We use $\phi(\mathcal{H})$ to denote the hypergraph $\mathcal{H}' = (\mathcal{V}', \mathcal{E}', w')$ where $\mathcal{E}' = \{\phi(e) \mid e \in \mathcal{E}\}$ and $w'(e') = w(\phi^{-1}(e))$. We write $\mathcal{H}_1 \cong \mathcal{H}_2$ if \mathcal{H}_1 equals to \mathcal{H}_2 up to some bijection, i.e., there exists a bijection ϕ such that $\phi(\mathcal{H}_1) = \mathcal{H}_2$. To measure the differences between two hypergraphs $\mathcal{H}_1 = (\mathcal{V}_1, \mathcal{E}_1, w_1)$ and $\mathcal{H}_2 = (\mathcal{V}_2, \mathcal{E}_2, w_2)$, we consider the following dissimilarity measure

$$d(\mathcal{H}_1, \mathcal{H}_2) = \sum_{e \in \mathcal{E}_1 \cup \mathcal{E}_2} |\bar{w}_1(e) - \bar{w}_2(e)|, \quad (1)$$

where the weight function $\bar{w}_i(e) = w_i(e)$ if $e \in \mathcal{E}_i$ and $\bar{w}_i(e) = 0$ otherwise, $i = 1, 2$. This measure corresponds to the dissimilarity between two graphs constructed from the hypergraphs by the star expansion algorithm (Surana et al., 2021) and captures the hyperedge weight differences between the hypergraphs.

Notation. We use A^* to denote the Kleene closure of set A , i.e., $A^* = \bigcup_{i=0}^{\infty} A^i$ where $A^0 = \{\varepsilon\}$ (the set consisting of only the empty sequence) and $A^i = \{(a_1, \dots, a_i) \mid a_j \in A, j = 1, \dots, i\}$. We use $\text{Bij}(A, B)$ to denote the set of all bijections from set A to set B . The notation $O(k)$ (resp., $\Omega(k)$) represents the upper bound (resp., the lower bound) of $C \cdot k$ for some constant C .

165 **4. Hypergraph Recovery Framework**
 166 This section introduces a mathematical framework of hypergraph recovery for relational learning in PTMs and how it
 167 could emerge from pre-training. We first model the entities
 168 and their relations in the world as a weighted hypergraph.
 169
 170 **Abstraction 4.1** (Relational Model of the World). The
 171 relational model of the world is a hypergraph $\mathcal{H}_0 =$
 172 $(\mathcal{V}_0, \mathcal{E}_0, w_0)$, where each node $v \in \mathcal{V}_0$ represents an entity,
 173 each hyperedge $e \in \mathcal{E}_0$ represents a relation between entities,
 174 and the weight function $w_0 : \mathcal{E} \mapsto \mathbb{R}$ represents the strength
 175 of the relations. Without loss of generality, we assume the
 176 weight function is normalized, i.e., $\sum_{e \in \mathcal{E}_0} w_0(e) = 1$. We
 177 further assume that $|\mathcal{V}_0| = n$ and $|\mathcal{E}_0| = m$.
 178
 179 Since data is the perception of the world, we formalize the
 180 data generation as sampling from the relational hypergraph
 181 of the world, as described in Abstraction 4.2.
 182
 183 **Abstraction 4.2** (Data Generation). In the data generation
 184 process, the entities are mapped to a perception domain (e.g.,
 185 language and vision). We denote the perception mapping by
 186 ϕ_0 . In this work, we consider the perception mapping ϕ_0 as
 187 a bijection, which keeps the structure of the relational hypergraph
 188 \mathcal{H}_0 . Each data point e is a perception of the relations
 189 in the domain, corresponding to a hyperedge sampled i.i.d.
 190 from the hypergraph $\phi_0(\mathcal{H}_0)$ according to the weights, i.e.,
 191 $e \sim P_w(e) = w(e) = w_0(\phi_0^{-1}(e))$.
 192
 193 Under this model, we define relational learning as follows.
 194
 195 **Definition 4.3** (Relational Learning). A hypergraph $\mathcal{H} =$
 196 $(\mathcal{V}, \mathcal{E}, w)$ achieves relational learning for the relational
 197 model of the world if $\mathcal{H} \cong \mathcal{H}_0$, i.e., there exists a bijection
 198 $\phi : \mathcal{V} \mapsto \mathcal{V}_0$ such that $\phi(\mathcal{H}) = \mathcal{H}_0$.
 199
 200 In practice, we have only finite samples and it is unrealistic
 201 to expect that the estimated relational hypergraph is
 202 completely the same as the relational model of the world. We
 203 further define ϵ -approximate relational learning to consider
 204 the approximation error of estimation with finite samples.
 205
 206 **Definition 4.4** (ϵ -Approximate Relational Learning). A hypergraph
 207 $\mathcal{H} = (\mathcal{V}, \mathcal{E}, w)$ achieves ϵ -approximate relational
 208 learning for the relational model of the world if there exists
 209 a bijection $\phi : \mathcal{V} \mapsto \mathcal{V}_0$ such that $d(\phi(\mathcal{H}), \mathcal{H}_0) \leq \epsilon$.
 210
 211 We also say that a model \mathcal{M} achieves (ϵ -approximate) relational
 212 learning if we can reconstruct a hypergraph that (ϵ -approximate)
 213 relational learning from the model.
 214
 215 **Definition 4.5** ((ϵ -Approximate) Relational Learning of
 216 Models). A model \mathcal{M} achieves (ϵ -approximate) relational
 217 learning if there exists a testing algorithm $\mathcal{A}_{\text{test}} : \mathcal{M} \mapsto$
 218 \mathcal{H} can estimate hypergraphs from models such that
 219 $\mathcal{A}_{\text{test}}(\mathcal{M}) = \mathcal{H}_{\mathcal{M}}$ achieves (ϵ -approximate) relational learning.
 220 Here, \mathcal{M} and \mathcal{H} denote the sets of all models and all
 221 hypergraphs of interest, respectively.

For PTMs, a typical process of relational learning is as follows: a pre-training algorithm \mathcal{A}_{pre} learns a model \mathcal{M} from a dataset D and a testing algorithm $\mathcal{A}_{\text{test}}$ examines whether the model achieves relational learning, i.e.,

$$\mathcal{H}_0 \xrightarrow{\text{Sample}} D \xrightarrow{\mathcal{A}_{\text{pre}}} M \xrightarrow{\mathcal{A}_{\text{test}}} \mathcal{H}. \quad (2)$$

From the information perspective, whether (ϵ -approximate) relational learning is achievable from a dataset D is equivalent to whether there exists a pre-training algorithm and a testing algorithm that can reconstruct a relational hypergraph equal to the relational hypergraph of the world (up to some bijection). The pre-training algorithm and the testing algorithm are expected to work well for a class of target relational hypergraphs. This goal can be captured by the following minimax formula:

$$\inf_{\mathcal{A}_{\text{pre}}, \mathcal{A}_{\text{test}}} \sup_{\mathcal{H}_0 \in \mathcal{H}_0} d(\mathcal{A}_{\text{test}}(\mathcal{A}_{\text{pre}}(D)), \phi_0(\mathcal{H}_0)) \leq \epsilon, \quad (3)$$

where the \mathcal{H}_0 is the set of target relational hypergraphs.

When we consider whether a model pre-trained by a certain algorithm can achieve relational learning, we need to consider how the pre-training algorithm can utilize the data. In this work, we consider Masked Modeling (MM), a common pre-training method that is widely used in various fields. In MM, a model is pre-trained to predict a sample e based on an input e^- that is generated by masked several tokens in e according to a masking strategy $\pi = \pi(e^- | e)$.

Abstraction 4.6 (Masked Modeling). Given a masked input e^- , a model \mathcal{M} pretrained by MM complements it and outputs e , reflecting the model's belief $\mathcal{M}(e | e^-)$ on

$$P(e | e^-) = \frac{w_0(\phi_0^{-1}(e))\pi(e | e^-)}{\sum_{e'} w_0(\phi_0^{-1}(e'))\pi(e^- | e')}$$

The model predicts a hyperedge $e \sim \mathcal{M}(e | e^-)$. With a slight abuse of notation, we denote the prediction of \mathcal{M} given e^- by $\mathcal{M}(e^-)$.

For two hyperedges e_1, e_2 such that $\pi(e^- | e_1) > 0$ and $\pi(e^- | e_2) > 0$, we can further infer their relative weights from the MM model \mathcal{M} as $\frac{\hat{w}(e_1)}{\hat{w}(e_2)} = \frac{M(e_1 | e^-)\pi(e^- | e_1)}{M(e_2 | e^-)\pi(e^- | e_2)}$. To capture such relations between two hyperedges, we define $e_1 \xrightarrow{\pi} e_2$ if there exists a masked hyperedge e^- such that $\pi(e^- | e_1) > 0$ and $\pi(e^- | e_2) > 0$. For the sake of notational simplicity and in cases where it does not lead to ambiguity, we use $e_1 \leftrightarrow e_2$ without the superscript π . Therefore, under our framework, we can view MM as learning the relative weights between \leftrightarrow related hyperedges.

We also abstract the data generation process of MM.

Abstraction 4.7 (Masked Modeling Data Generation). In the data generation of MM, each hyperedge e_t is first sampled i.i.d. from $P_w(e)$ where $P_w(e) = w_0(\phi_0^{-1}(e))$, for all

220 $t = 1, \dots, N$. For each hyperedge e_t , K masked hyperedges $\{e_{tk}^-\}_{k=1}^K$ are generated i.i.d. by a masking strategy π ,
 221 i.e., $e_{tk}^- \sim \pi(e_{tk}^- | e_{tk})$ where $e_{tk} = e_t$, for all $1 \leq k \leq K$.
 222 The dataset for MM is $D = \{(e_{tk}, e_{tk}^-)\}_{1 \leq t \leq N, 1 \leq k \leq K}$.
 223
 224

225 Under Abstractions 4.6 and 4.7, an MM model \mathcal{M} pre-
 226 trained on D with a loss ℓ is
 227
 228

$$\mathcal{M} = \arg \min_{\mathcal{M}' \in \mathcal{M}} \sum_{t=1}^N \sum_{k=1}^K \ell(\mathcal{M}'(e_{tk}^-), e_{tk}). \quad (4)$$

231 For an MM pre-trained model to achieve relational learning,
 232 it needs to learn relative weights from an MM dataset such
 233 that these relative weights amount to the recovery of the
 234 relational hypergraph \mathcal{H}_0 . Denote the MM pre-training
 235 algorithm in (4) by \mathcal{A}_{MM} under Abstractions 4.6 and 4.7.
 236 Following (2) and (3), this is to consider
 237

$$\inf_{\mathcal{A}_{\text{test}}} \sup_{\mathcal{H}_0} d(\mathcal{A}_{\text{test}}(\mathcal{A}_{\text{MM}}(D)), \phi_0(\mathcal{H}_0)) \leq \epsilon. \quad (5)$$

5. Main Results for Entity Relational Learning

5.1. Identification

244 We first consider whether identifying the relational hyper-
 245 graph \mathcal{H}_0 from a pre-training dataset is possible at the pop-
 246 ulation level. The following theorem affirms the feasibility
 247 of relational learning if sufficient data are available.
 248
 249

250 **Theorem 5.1** (Identifiability). *Under Abstractions 4.1*
 251 and *4.2*, suppose that e_t is a generated data sequence. Let
 252 D_N be the dataset consisting of the first N elements of the
 253 sequences, i.e., $D_N = (e_1, \dots, e_N)$. Then there exist an
 254 pre-training algorithm \mathcal{A}_{pre} and a testing algorithm $\mathcal{A}_{\text{test}}$,
 255 $\mathcal{A} = \mathcal{A}_{\text{test}}(\mathcal{A}_{\text{pre}}(\cdot)) : \mathcal{E}^* \mapsto \mathcal{H}$ such that $\mathcal{A}(D_N)$ con-
 256 verges to a hypergraph \mathcal{H} that achieves relational learning
 257 as $N \rightarrow \infty$ almost surely, i.e., $\mathcal{A}(D_N) \xrightarrow{\text{a.s.}} \mathcal{H} \cong \mathcal{H}_0$.
 258
 259

260 Theorem 5.1 asserts the asymptotic identifiability of the
 261 target hypergraph as the dataset size approaches infinity. The
 262 proof of Theorem 5.1 leverages the law of large numbers
 263 to show that the distance between the estimated hypergraph
 264 and the actual relational hypergraph converges to 0. For
 265 detailed proof, refer to Appendix A.
 266
 267

5.2. Data Efficiency

268 Since relational learning is feasible at the population
 269 level, we then consider the data efficiency to achieve ϵ -
 270 approximate relational learning at the sample level. We first
 271 consider an information theoretical lower bound of the sam-
 272 ple complexity to achieve ϵ -approximate relational learning.
 273
 274

275 **Theorem 5.2** (Information Theoretical Lower Bound). *Under Abstractions 4.1 and 4.2 and assuming that the generated dataset D is of size $|D| = N \geq m$ with m sufficiently*

large, the minimax risk of reconstruction error satisfies

$$\inf_{\mathcal{A}_{\text{pre}}, \mathcal{A}_{\text{test}}} \sup_{\mathcal{H}_0} \mathbb{E}_D [d(\mathcal{A}_{\text{test}}(\mathcal{A}_{\text{pre}}(D)), \phi_0(\mathcal{H}_0))] \geq \frac{1}{16} \sqrt{\frac{m}{N}}.$$

276 Theorem 5.2 presents an information theoretical lower
 277 bound $\Omega(\frac{m}{\epsilon^2})$ of the sample complexity for ϵ -approximate
 278 relational learning. This lower bound is derived from the
 279 sample complexity lower of the discrete distribution esti-
 280 mation problem under ℓ_1 distance, by a reduction from the
 281 estimation problem to an approximate relational learning
 282 problem. The lower bound highlights that the number of
 283 the hyperedges m is an important factor in the difficulty of
 284 relational learning.
 285
 286

287 Now we consider the data efficiency of MM to achieve ϵ -
 288 approximate relational learning. We assume that the model
 289 \mathcal{M} is expressive enough to fit the pre-training data, i.e., for
 290 a MM dataset D , the model \mathcal{M} pre-trained on D satisfies
 291

$$\mathcal{M} = \arg \min \sum_{t=1}^N \sum_{k=1}^K \ell(\mathcal{M}'(e_{tk}^-), e_{tk}). \quad (6)$$

292 To characterize the sample complexity, we introduce the
 293 following additional assumptions.
 294
 295

Assumption 5.3 (Range ratio of the weight function). The range ratio of the weight function is $\kappa = \frac{\max_{e \in \mathcal{E}} w(e)}{\min_{e \in \mathcal{E}} w(e)}$.
 296
 297

Assumption 5.4 (Bound on the masking strategy). For each hyperedge $e \in \mathcal{E}$, the support set of masked hyperedges is upper bounded, i.e., $|\text{supp } \pi(\cdot | e)| < C_\pi$ for some constant C_π . For each $e \in \mathcal{E}$ and $e^- \in \text{supp } \pi(\cdot | e)$, the probability $\pi(e^- | e)$ is lower bounded by some constant c_π .
 298
 299

Assumption 5.5 (Bound on the MM path length). For any hyperedges $e, e' \in \mathcal{E}$, there exists a path bounded by L such that $e = e_1 \leftrightarrow e_2 \leftrightarrow \dots \leftrightarrow e_\ell = e'$.
 300
 301

302 Assumption 5.3 bounds the weights of each hyperedge within a certain range. Assumption 5.4 bounds the complexity of the masking strategy by limiting the support set of masked hyperedges and setting a minimum probability threshold for potentially masked hyperedges. Assumption 5.5 bounds the connectivity complexity among the hyperedges under the masking strategy.
 303
 304

305 We analyze the sample complexity for the PTM pre-trained
 306 by MM \mathcal{M} to achieve ϵ -approximate relational learning
 307 with cross-entropy loss in Theorem 5.6.
 308
 309

Theorem 5.6 (Upper Bound by MM). *Suppose that \mathcal{M} is an FM pre-trained by MM on a dataset D with cross-entropy loss. Then \mathcal{M} achieves ϵ -approximate relational*

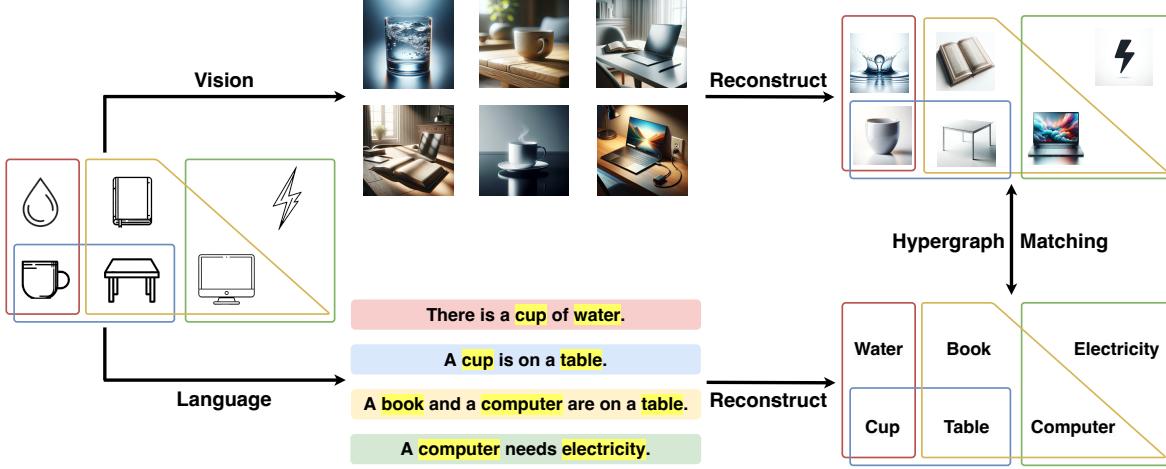


Figure 2. Extension of our hypergraph framework to entity alignment in multimodal learning (taking vision and language for illustration). The relational hypergraphs in different modalities can be reconstructed from data. The entities from different modalities can be aligned by matching the relational hypergraphs.

learning with probability at least $1 - \delta$ if

$$K \geq \frac{2^{14}m^2\kappa^2L^2}{c_\pi^2\epsilon^2} \log \frac{6mC_\pi}{\delta},$$

$$N \geq \max \left\{ \frac{2m\kappa}{c_\pi} \log \frac{3mC_\pi}{\delta}, \frac{8m}{\epsilon^2} \log \frac{6m}{\delta} \right\}. \quad (7)$$

In scenarios defined by specific problems and masking strategies, the term $\tilde{O}\left(\frac{m}{\epsilon^2}\right)$ predominates at low approximation errors, especially when $\epsilon = o\left(\sqrt{\frac{C_\pi}{\kappa}}\right)$. This aligns with the information theoretical lower bound $\Omega\left(\frac{m}{\epsilon^2}\right)$ in Theorem 5.2, disregarding the logarithmic factor. This suggests that MM is near-optimal in data efficiency.

To prove Theorem 5.6, we design an algorithm that computes the relative weights between the pairs of the hyperedges along $e_1 \leftrightarrow \dots \leftrightarrow e_\ell$ paths. By normalization, we obtain an estimation of the hyperedge weights and further a recovered hypergraph from the relative weights. We show that when the dataset D is sufficiently large, the model \mathcal{M} can learn all the relative weights well enough and therefore the reconstructed hypergraph is a good approximation for the relational hypergraph \mathcal{H}_0 (up to some bijection).

Theorem 5.6 reveals that the data efficiency to achieve relational learning is predominantly influenced by three factors: the number of hyperedges m , the range ratio of the weight function κ , and the upper bound of the MM path lengths L . The number of hyperedges m and the range ratio of the weight function κ characterize the complexity of the world relational hypergraph, i.e., the hypergraph with more hyperedges and a larger range ratio requires more samples to be recovered by MM. The MM path length bound L reflects the connectivity under the masking strategy π , influencing how MM learns the relative weights between hyperedges.

Efficient recovery of the relational hypergraph is contingent on a small L , indicating well-connected hyperedges; a large L suggests inefficiency in recovery. This aligns with empirical observations that effective MM performance requires masking a sufficient proportion of each sample (He et al., 2022; Wettig et al., 2023).

6. Main Results for Entity Alignment

We further extend our framework to encompass entity alignment within the realm of multimodal learning. In this context, the relational models associated with different modalities are interpreted as distinct representations or “images” of the relational model of the world, each shaped by its unique perception mapping. Although our focus here is on two modalities for illustrative purposes, the principles and methodologies we discuss are readily generalizable to scenarios involving a greater number of modalities.

Concretely, the relational hypergraph in modality i is mapped from \mathcal{H}_0 by the perception ϕ_i , i.e., $\mathcal{H}_i = \phi_i(\mathcal{H}_0)$ for $i = 1, 2$. Entity alignment is to find a bijection $\phi \in \text{Bij}(\mathcal{V}_1, \mathcal{V}_2)$ such that $\phi(\mathcal{H}_1) = \mathcal{H}_2$. The data supporting entity alignment consists of three parts: D_1 , D_2 , and D_{12} . Here, D_1 and D_2 represent data from the two individual modalities, while D_{12} comprises labeled pairs that denote corresponding relationships across the modalities. For example, in aligning entities between visual and linguistic modalities, the data includes images, text, and labeled pairs that link images with their textual descriptions.

Assuming the data from each modality are sufficient, we can recover the relational hypergraphs \mathcal{H}_1 and \mathcal{H}_2 . Entity alignment is achieved by solving the optimization problem:

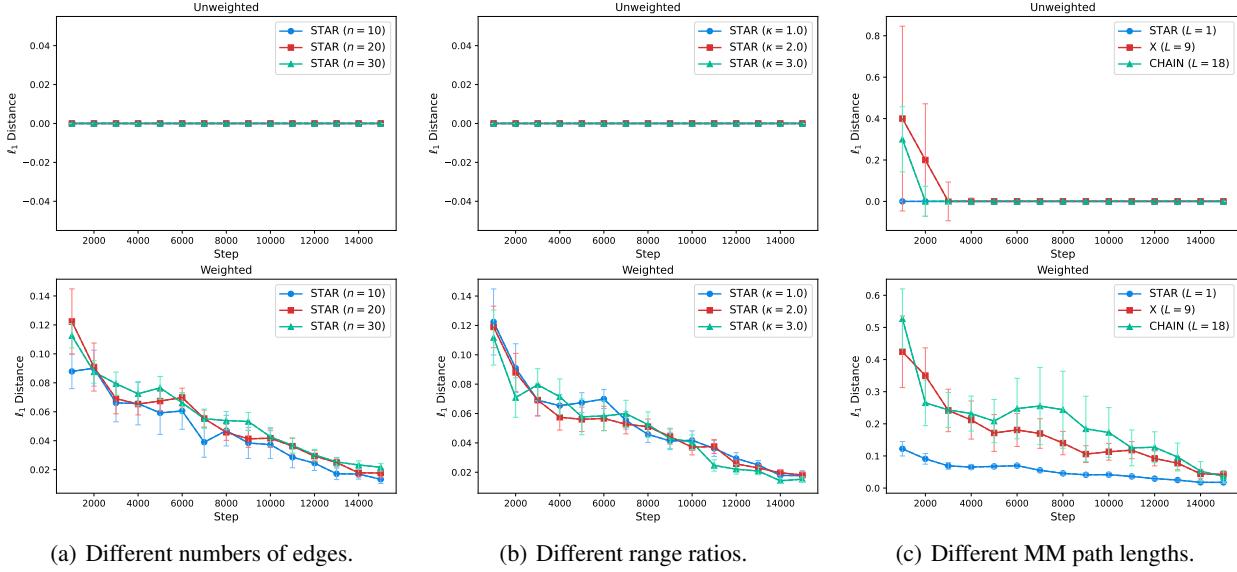


Figure 3. Evaluation results of synthetic relational learning. (a) STAR graphs with different numbers of edges ($m = n - 1$). (b) STAR graphs with different range ratios. (c) Graphs with different MM path lengths. For each, the experiments are repeated for 5 times and the evaluation results are averaged over the 5 trials.

$$\phi^* = \arg \min_{\phi \in \text{Bij}(\mathcal{V}_1, \mathcal{V}_2)} d(\phi(\mathcal{H}_1), \mathcal{H}_2) \quad (8)$$

Practically, labeled pairs are typically necessary to address the computational difficulty of the graph isomorphism problem in (8), as no polynomial-time solution has been found to date (Babai, 2016; Neuen & Schweitzer, 2018). Labeled pairs are external information that pinpoints partial correspondences between the entities of different modalities, potentially reducing the computational complexity. For example, the labeled pairs can reduce dimensions of Weisfeiler-Lehman methods required (Cai et al., 1992) or prune search trees in individualization-refinement algorithms (McKay & Piperno, 2014) (See Appendix B for further illustration).

7. Experiments

We conduct two experiments to show empirically that relational learning in PTMs could be seen as relational hypergraph recovery. We consider two settings: synthetic relational learning and real-world relation evaluation.

7.1. Synthetic Relational Learning

In synthetic relational learning, we train PTMs with text consisting of synthetic entities, whose underlying data distribution corresponds to a graph. We show that PTMs can learn the relations between these synthetic entities.

To generate data for synthetic relational learning, we first construct a graph, whose nodes are entities (represented by tokens) and edges are relations. We attach edges with random weights and normalize the weights. To generate a

training dataset, we sample edges i.i.d. according to the distribution corresponding to the normalized edge weights. We consider masked language modeling (Kenton & Toutanova, 2019). For evaluation, we query the PTM with each synthetic entity to retrieve information about its related entities and the weights of the relations. We reconstruct a graph with the query results and compare the reconstructed graph with the true underlying graph. We conduct experiments for different graphs, with different numbers of edges, range ratios, and MM path lengths, corresponding to the factors that influence the sample complexity of entity relational learning. More details of the synthetic relational learning experiments can be found in Appendix C.1. The evaluation results are shown in Section 5.2. Our results show that the reconstruction errors of both the unweighted sketch graph and the weighted graph decrease as the training goes on. This the PTMs learn the synthetic relations gradually via MM pre-training. Additionally, the results suggest that larger numbers of edges and larger MM path lengths lead to more steps to converge, which coincides with our theoretical analysis in Theorem 5.6. The effect of the range ratios on the convergence of relational learning is not obvious in our experiments. This may suggest a gap between the theoretical upper bound and the actual convergence rate in the experiments in terms of the range ratio.

7.2. Real-World Relation Evaluation

In real-world relation evaluation, we test whether LLMs such as ChatGPT and GPT-4 learn entities and their relations that align with the real world. We use subgraphs extracted from ConceptNet (Speer et al., 2017) as baselines of the

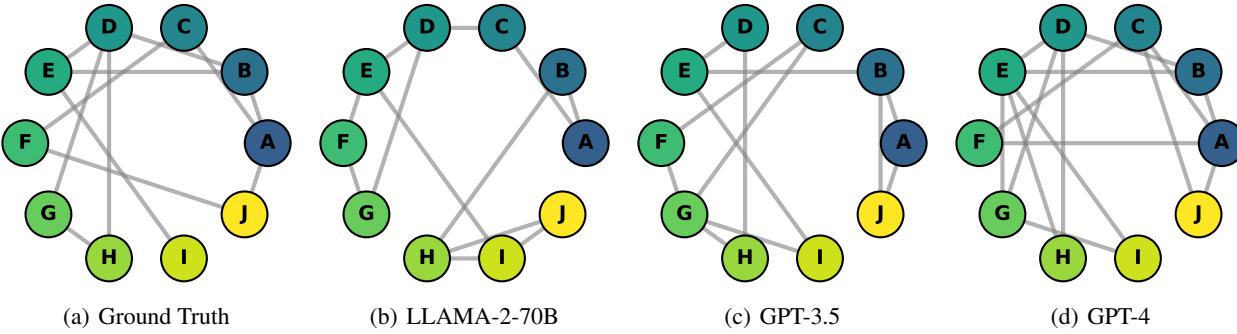


Figure 4. Evaluation results of different LLMs for the real-world relational subgraph generated from the source word “table”. We use different letters to represent different entities (see Appendix C.3 for their correspondences). The graphs (from left to right) are the ground truth (extracted from ConceptNet), evaluation results of LLAMA-2-70B, GPT-3.5, and GPT-4, respectively.

Table 1. Summary of the comparison results. The subgraphs are generated from different source entities with $k = 2$ and $d = 3$. The corresponding evaluated graphs are generated from the outputs of different LLMs. The dissimilarity between each pair of the extracted subgraph \mathcal{H} and the estimated graph \mathcal{H}' are measured by their normalized L_1 distance, i.e., $\frac{\|\mathcal{H} - \mathcal{H}'\|_1}{\|\mathcal{H}\|_1}$ where we slightly abuse the notations \mathcal{H} and \mathcal{H}' to denote their adjacent matrices.

	CAKE	DOG	FLY	HUMAN	JACKET	ORANGE	PAPER	SEA	TABLE	ZOO
LLAMA-2-70B	1.00	0.67	1.25	1.00	1.33	1.33	0.75	0.83	1.25	1.67
GPT-3.5	0.67	1.00	1.00	1.25	1.00	1.33	0.75	0.83	1.00	1.00
GPT-4	0.67	0.67	1.00	1.50	1.33	1.00	0.75	0.83	0.75	1.33

real-world relations graphs. For evaluation, we input the chosen entities to LLMs and ask them to choose top-related ones for each entity. We then construct a graph whose nodes are the entities and edges are those top-related pairs. We compare the subgraph extracted from ConceptNet and the graph evaluated from LLMs. If an LLM learns real-world relations, we expect it to produce a similar graph as the one extracted from ConceptNet. Table 1 summarizes some comparison results of the extracted subgraphs generated by different source entities and the corresponding evaluated graphs. In Figure 4, we visualize the result of the source entity “table”. More results are presented in Appendix C.3. We find that GPT-4 achieves the best overall performance among the evaluated LLMs and GPT-3.5 performs slightly better than LLAMA-2-70B. The results suggest different LLMs have different degrees of relational learning and more powerful models seem to understand entity relations better in the sense of relational subgraph reconstruction. Note that we only consider unweighted graphs here because it is difficult to evaluate the relation weights from LLMs accurately. Our results illustrate that the LLMs do organize entities similarly to real-world entities.

8. Conclusion and Outlook

Abstracting the entity relations in the world as a hypergraph, we formalize relational learning in pre-trained models as

recovery of the world relational hypergraph. Under the formulation, we show the relational hypergraph is identifiable provided sufficient data at the population level. We also study the sample efficiency and extend the framework to entity alignment in multimodal learning.

While only extending in multimodal learning in this paper, our framework is a general analysis tool. Understanding the capabilities and generalization potential of the PTM is crucial in our field. We would say that PTMs, such as LLMs, often responding to complex relationships between objects, urgently require new mathematical foundations to have a deeper study. This paper paves a new way to study PTM from a unique perspective by capturing the overlooked data information using a hypergraph. Our framework can be potentially used under various scenarios and impacts on application fields. For example, for data and computational efficiency, it is interesting to design more efficient learning algorithms or architectures, such as for multimodal learning. More broadly, for safety, traditional works about adversarial attack and defense theories often focus on several classes that need to be protected. Our framework is not restricted to classification problems and may impose a potential on the entity concept and even human value level. Further, based on the hypergraph, it is promising to understand the reasoning and causality capabilities of PTMs.

References

- Akyürek, E., Schuurmans, D., Andreas, J., Ma, T., and Zhou, D. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.
- Ando, R. K., Zhang, T., and Bartlett, P. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(11), 2005.
- Babai, L. Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 684–697, 2016.
- Barak, B., Chou, C.-N., Lei, Z., Schramm, T., and Sheng, Y. (nearly) efficient algorithms for the graph matching problem on correlated random graphs. *Advances in Neural Information Processing Systems*, 32, 2019.
- Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselet, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.
- Bretto, A. Hypergraph theory. *An introduction. Mathematical Engineering*. Cham: Springer, 1, 2013.
- Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv preprint arXiv:2303.12712*, 2023.
- Cai, J.-Y., Fürer, M., and Immerman, N. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12(4):389–410, 1992.
- Chen, D. and Manning, C. D. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 740–750, 2014.
- Chen, L., Li, Z., Wang, Y., Xu, T., Wang, Z., and Chen, E. MMEA: entity alignment for multi-modal knowledge graph. In *Knowledge Science, Engineering and Management: 13th International Conference, KSEM 2020, Hangzhou, China, August 28–30, 2020, Proceedings, Part I* 13, pp. 134–147. Springer, 2020.
- Chen, Y., Coskunuzer, B., and Gel, Y. Topological relational learning on graphs. *Advances in neural information processing systems*, 34:27029–27042, 2021.
- Chen, Y., Jamieson, K., and Du, S. Active multi-task representation learning. In *International Conference on Machine Learning*, pp. 3271–3298. PMLR, 2022.
- Chomsky, N. *Aspects of the Theory of Syntax*. Number 11. MIT press, 2014.
- Cullina, D. and Kiyavash, N. Improved achievability and converse bounds for erdos-rényi graph matching. *ACM SIGMETRICS performance evaluation review*, 44(1):63–72, 2016.
- De Raedt, L. *Logical and relational learning*. Springer Science & Business Media, 2008.
- De Raedt, L. and Kersting, K. Probabilistic inductive logic programming. In *Probabilistic inductive logic programming: theory and applications*, pp. 1–27. Springer, 2008.
- Dettmers, T., Minervini, P., Stenetorp, P., and Riedel, S. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.
- Ding, J., Ma, Z., Wu, Y., and Xu, J. Efficient random graph matching via degree profiles. *Probability Theory and Related Fields*, 179:29–115, 2021.
- Ding, J., Jiang, Y., and Ma, H. Shotgun threshold for sparse erdős-rényi graphs. *IEEE Transactions on Information Theory*, 2023.
- Durrett, R. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.
- Džeroski, S., De Raedt, L., and Driessens, K. Relational reinforcement learning. *Machine learning*, 43:7–52, 2001.
- Fey, M., Hu, W., Huang, K., Lenssen, J. E., Ranjan, R., Robinson, J., Ying, R., You, J., and Leskovec, J. Relational deep learning: Graph representation learning on relational databases. *arXiv preprint arXiv:2312.04615*, 2023.
- Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.
- Finn, C., Xu, K., and Levine, S. Probabilistic model-agnostic meta-learning. *Advances in neural information processing systems*, 31, 2018.
- Frieder, S., Pinchetti, L., Griffiths, R.-R., Salvatori, T., Lukasiewicz, T., Petersen, P. C., Chevalier, A., and Berner, J. Mathematical capabilities of ChatGPT. *arXiv preprint arXiv:2301.13867*, 2023.
- Frucht, R. Herstellung von graphen mit vorgegebener abstrakter gruppe. *Compositio Mathematica*, 6:239–250, 1939.

- 495 Han, Y., Jiao, J., and Weissman, T. Minimax estimation
 496 of discrete distributions. In *2015 IEEE International*
 497 *Symposium on Information Theory (ISIT)*, pp. 2291–2295.
 498 IEEE, 2015.
- 500 He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick,
 501 R. Masked autoencoders are scalable vision learners. In
 502 *Proceedings of the IEEE/CVF conference on computer*
 503 *vision and pattern recognition*, pp. 16000–16009, 2022.
- 504 Hewitt, J. and Manning, C. D. A structural probe for finding
 505 syntax in word representations. In *Proceedings of the*
 506 *2019 Conference of the North American Chapter of the*
 507 *Association for Computational Linguistics: Human*
 508 *Language Technologies, Volume 1 (Long and Short Papers)*,
 509 pp. 4129–4138, 2019.
- 510 Hu, J., Chen, X., Jin, C., Li, L., and Wang, L. Near-optimal
 511 representation learning for linear bandits and linear rl.
 512 In *International Conference on Machine Learning*, pp.
 513 4349–4358. PMLR, 2021.
- 514 Idury, R. M. and Waterman, M. S. A new algorithm for DNA
 515 sequence assembly. *Journal of computational biology*, 2
 516 (2):291–306, 1995.
- 517 Kenton, J. D. M.-W. C. and Toutanova, L. K. BERT: Pre-
 518 training of deep bidirectional transformers for language
 519 understanding. In *Proceedings of NAACL-HLT*, pp. 4171–
 520 4186, 2019.
- 521 Korolova, A., Motwani, R., Nabar, S. U., and Xu, Y. Link
 522 privacy in social networks. In *Proceedings of the 17th*
 523 *ACM conference on Information and knowledge management*, pp. 289–298, 2008.
- 524 Li, Y., Ildiz, M. E., Papailiopoulos, D., and Oymak, S.
 525 Transformers as algorithms: Generalization and stability
 526 in in-context learning. In *International Conference on*
 527 *Machine Learning*, pp. 19565–19594. PMLR, 2023a.
- 528 Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C.,
 529 and Lee, Y. J. Gligen: Open-set grounded text-to-image
 530 generation. In *Proceedings of the IEEE/CVF Conference*
 531 *on Computer Vision and Pattern Recognition*, pp. 22511–
 532 22521, 2023b.
- 533 Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. Learning
 534 entity and relation embeddings for knowledge graph
 535 completion. In *Proceedings of the AAAI conference on*
 536 *artificial intelligence*, volume 29, 2015.
- 537 Liu, J., Jin, J., Wang, Z., Cheng, J., Dou, Z., and Wen, J.-
 538 R. RETA-LLM: A retrieval-augmented large language
 539 model toolkit. *arXiv preprint arXiv:2306.05212*, 2023.
- 540 McKay, B. D. and Piperno, A. Practical graph isomorphism,
 541 ii. *Journal of symbolic computation*, 60:94–112, 2014.
- 542 Miller, G. A. Wordnet: a lexical database for english. *Com-*
 543 *munications of the ACM*, 38(11):39–41, 1995.
- 544 Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Founda-*
 545 *tions of machine learning*. MIT press, 2018.
- 546 Mossel, E. and Ross, N. Shotgun assembly of labeled graphs.
 547 *IEEE Transactions on Network Science and Engineering*,
 548 6(2):145–157, 2017.
- 549 Neuen, D. and Schweitzer, P. An exponential lower bound
 550 for individualization-refinement algorithms for graph iso-
 551 morphism. In *Proceedings of the 50th Annual ACM*
 552 *SIGACT Symposium on Theory of Computing*, pp. 138–
 553 150, 2018.
- 554 OpenAI. GPT-4 technical report. *arXiv preprint*
 555 *arXiv:2303.08774*, 2023.
- 556 Rajani, N. F., McCann, B., Xiong, C., and Socher, R. Ex-
 557 plain yourself! leveraging language models for com-
 558 monsense reasoning. In *Proceedings of the 57th Annual*
 559 *Meeting of the Association for Computational Linguistics*,
 560 pp. 4932–4942, 2019.
- 561 Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Rad-
 562 ford, A., Chen, M., and Sutskever, I. Zero-shot text-
 563 to-image generation. *arXiv preprint arXiv:2102.12092*,
 564 2021. URL <https://arxiv.org/abs/2102.12092>.
- 565 Speer, R., Chin, J., and Havasi, C. Conceptnet 5.5: An
 566 open multilingual graph of general knowledge. In *Pro-*
 567 ,
 568 volume 31, 2017.
- 569 Struyf, J. and Blockeel, H. Relational learning., 2010.
- 570 Suchanek, F. M., Kasneci, G., and Weikum, G. Yago: a
 571 core of semantic knowledge. In *Proceedings of the 16th*
 572 *international conference on World Wide Web*, pp. 697–
 573 706, 2007.
- 574 Surana, A., Chen, C., and Rajapakse, I. Hypergraph dis-
 575 similarity measures. *arXiv preprint arXiv:2106.08206*,
 576 2021.
- 577 Tripuraneni, N., Jin, C., and Jordan, M. Provable meta-
 578 learning of linear representations. In *International Con-*
 579 *fERENCE on Machine Learning*, pp. 10434–10443. PMLR,
 580 2021.
- 581 Valiant, L. G. A theory of the learnable. *Communications*
 582 *of the ACM*, 27(11):1134–1142, 1984.
- 583 Von Oswald, J., Niklasson, E., Randazzo, E., Sacramento,
 584 J., Mordvintsev, A., Zhmoginov, A., and Vladymyrov,
 585 M. Transformers learn in-context by gradient descent.
 586 In *International Conference on Machine Learning*, pp.
 587 35151–35174. PMLR, 2023.

- 550 Wettig, A., Gao, T., Zhong, Z., and Chen, D. Should you
551 mask 15% in masked language modeling? In *Proceedings*
552 *of the 17th Conference of the European Chapter of the As-*
553 *sociation for Computational Linguistics*, pp. 2977–2992,
554 2023.
- 555 Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C.,
556 Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M.,
557 et al. Transformers: State-of-the-art natural language
558 processing, 2020.
- 560 Xie, S. M., Kumar, A., Jones, R., Khani, F., Ma, T., and
561 Liang, P. In-n-out: Pre-training and self-training using
562 auxiliary information for out-of-distribution robustness.
563 In *International Conference on Learning Representations*,
564 2020.
- 565 Yang, J., Lei, Q., Lee, J. D., and Du, S. S. Nearly minimax
566 algorithms for linear bandits with shared representation.
567 *arXiv preprint arXiv:2203.15664*, 2022.
- 569 Zambaldi, V., Raposo, D., Santoro, A., Bapst, V., Li, Y.,
570 Babuschkin, I., Tuyls, K., Reichert, D., Lillicrap, T.,
571 Lockhart, E., et al. Deep reinforcement learning with
572 relational inductive biases. In *International conference*
573 *on learning representations*, 2018a.
- 575 Zambaldi, V., Raposo, D., Santoro, A., Bapst, V., Li, Y.,
576 Babuschkin, I., Tuyls, K., Reichert, D., Lillicrap, T.,
577 Lockhart, E., et al. Relational deep reinforcement learn-
578 ing. *arXiv preprint arXiv:1806.01830*, 2018b.
- 580 Zaslavskiy, M., Bach, F., and Vert, J.-P. Global alignment of
581 protein–protein interaction networks by graph matching
582 methods. *Bioinformatics*, 25(12):i259–1267, 2009.
- 583 Zhang, S., Chen, Z., Shen, Y., Ding, M., Tenenbaum, J. B.,
584 and Gan, C. Planning with large language models for code
585 generation. In *The Eleventh International Conference on*
586 *Learning Representations*, 2022.
- 588 Zhao, X., Zeng, W., and Tang, J. Multimodal entity align-
589 ment. In *Entity Alignment: Concepts, Recent Advances*
590 *and Novel Approaches*, pp. 229–247. Springer, 2023a.
- 592 Zhao, Z., Lee, W. S., and Hsu, D. Large language models as
593 commonsense knowledge for large-scale task planning.
594 In *RSS 2023 Workshop on Learning for Task and Motion*
595 *Planning*, 2023b.
- 596
- 597
- 598
- 599
- 600
- 601
- 602
- 603
- 604

A. Proof

A.1. Proof of Theorem 5.1

We can consider the combined algorithm $\mathcal{A} = \mathcal{A}_{\text{test}} \circ \mathcal{A}_{\text{pre}}$ directly. We design an algorithm (Algorithm 1) that recovers hypergraphs from dataset and show the reconstructed hypergraph converges to \mathcal{H}_0 up to some bijection almost surely by the law of large numbers. Denote the hypergraph recovered from \mathcal{D}_N by \mathcal{H}_N . Define random variables $X_N = d(\phi_0^{-1}(\mathcal{H}_N), \mathcal{H}_0)$ for $N = 1, 2, \dots$. It remains to show $X_N \xrightarrow{a.s.} 0$.

For any $\epsilon > 0$, define

$$E_N := \{\omega \in \Omega : X_N(\omega) > \epsilon\}, \quad (9)$$

where Ω is the sample space.

Let

$$Y_{e,t} = \begin{cases} 1 & x_t = \phi_0(e), \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

Then we have

$$\begin{aligned} P(E_N) &= P\left(\sum_{e \in \mathcal{E}_0} \left| \frac{1}{N} \sum_{t=1}^N Y_{e,t} - w_0(e) \right| > \epsilon\right) \\ &\leq P\left(\bigcup_{e \in \mathcal{E}_0} \left| \frac{1}{N} \sum_{t=1}^N Y_{e,t} - w_0(e) \right| > \frac{\epsilon}{m}\right) \\ &\stackrel{(a)}{\leq} \sum_{e \in \mathcal{E}_0} P\left(\left| \frac{1}{N} \sum_{t=1}^N Y_{e,t} - w_0(e) \right| > \frac{\epsilon}{m}\right) \\ &\stackrel{(b)}{\leq} 2m \exp\left(-\frac{2N\epsilon^2}{m^2}\right), \end{aligned} \quad (11)$$

where the inequality (a) is due to union bound and the inequality (b) is due to Hoeffding's Inequality.

Notice that

$$\sum_{N=1}^{\infty} P(E_N) \leq \frac{2m \exp(-2\epsilon^2/m^2)}{1 - \exp(-2\epsilon^2/m^2)} < \infty. \quad (12)$$

By the first Borel-Cantelli lemma (Durrett, 2019, Chapter 2), we have

$$P\left(\limsup_{N \rightarrow \infty} E_N\right) = 0. \quad (13)$$

Equivalently, we have

$$P\left(\lim_{N \rightarrow \infty} X_N > \epsilon\right) = 0. \quad (14)$$

Since (14) holds for any $\epsilon > 0$, we have $P(\lim_{n \rightarrow \infty} X_N = 0) = 1$, i.e., $X_N \xrightarrow{a.s.} 0$.

A.2. Proof of Theorem 5.2

We prove the information theoretical lower bound by constructing a reduction from finite distribution estimation under ℓ_1 distance to concept understanding.

For any unknown finite distribution $P = (p_1, \dots, p_m)$ on $\{1, \dots, m\}$, we construct a world model $\mathcal{H}_0 = (\mathcal{V}_0, \mathcal{E}_0, w_0)$ as follows:

1. $\mathcal{V}_0 = \{v_1, \dots, v_{m+1}\}$;
2. $\mathcal{E}_0 = \{\{v_1, v_2\}, \dots, \{v_m, v_{m+1}\}\}$;
3. $w_0(\{v_i, v_{i+1}\}) = p_i$.

Algorithm 1 Hypergraph Estimation from Datasets

Input: a dataset D , a candidate hyperedge set \mathcal{E}_0 , and a masking strategy π .

Initialize $\mathcal{E} = \{\}$, $\mathcal{V} = \emptyset$, and $\tilde{w} = 0$.

for $x \in D$ **do**

$$\mathcal{E} = \mathcal{E} \cup \{x\}$$

$$\mathcal{V} = \mathcal{V} \cup x$$

$$\tilde{w}(x) = \tilde{w}(x) + 1$$

end for

Compute $W = \sum_{e \in \mathcal{E}} \tilde{w}(e)$.

$$w = \tilde{w}/W.$$

Return $\mathcal{H} = (\mathcal{V}, \mathcal{E}, w)$.

For a dataset $D' = x_{k=1}^N$ sampled from P , convert it to a dataset $D = \{\{v_{x_k}, v_{x_{k+1}}\}\}_{k=1}^N$ for hypergraph recovery. For an algorithm \mathcal{A} , apply it to the dataset D and we obtain an estimation $\mathcal{H} = \mathcal{A}(D) = (\mathcal{V}, \mathcal{E}, w)$ for the world model \mathcal{H}_0 . We then compute an estimation P' for the finite distribution P , where $P' = (p'_1, \dots, p'_m)$ and

$$p'_i = w(\{v_i, v_{i+1}\}). \quad (15)$$

Denote the minimax risk of estimating a finite distribution on $\{1, \dots, m\}$ with a dataset of size N as $R(m, N)$. Denote the minimax risk of estimating a hypergraph \mathcal{H}_0 of m hyperedges with a dataset of size N as $R_{\mathcal{H}}(m, N)$. Then we have

$$\begin{aligned}
R(m, N) &\leq \inf_{\mathcal{A}} \sup_{P \in \mathcal{P}_m} \sum_{i=1}^m \|p'_i - p_i\| \\
&= \inf_{\mathcal{A}} \sup_{\mathcal{H}_0 \in \mathcal{H}_m} \sum_{e \in \mathcal{E}_0} \|w(e) - w_0(e)\| \\
&= \inf_{\mathcal{A}} \sup_{\mathcal{H}_0 \in \mathcal{H}_m} d(\mathcal{H}, \mathcal{H}_0) \\
&= R_{\mathcal{H}}(m, N),
\end{aligned} \tag{16}$$

where the first inequality is due to the definition of the minimax risk $R(m, N)$.

According to Theorem 2 in Han et al. (2015), we have

$$R(m, N) \geq \max_{0 < \zeta < 1} F(\zeta), \quad (17)$$

where

$$\begin{aligned} F(\zeta) &= \frac{1}{8} \sqrt{\frac{em}{((1+\zeta)N)}} \mathbb{1}\left(\frac{(1+\zeta)N}{m} > \frac{e}{16}\right) \\ &+ \exp\left(-\frac{2(1+\zeta)N}{m}\right) \mathbb{1}\left(\frac{(1+\zeta)N}{m} \leq \frac{e}{16}\right) \\ &- \exp\left(-\frac{\zeta^2 N}{24}\right) - 12 \exp\left(-\frac{\zeta^2 m}{32 \ln^2 m}\right). \end{aligned} \quad (18)$$

Combining (16) and (17) and letting $\zeta = 1$, we have

$$R_{\mathcal{H}}(m, N) \geq F(1) \geq \frac{1}{8} \sqrt{\frac{em}{2N}} - \exp\left(-\frac{N}{24}\right) - 12 \exp\left(-\frac{m}{32 \ln^2 m}\right) \geq \frac{1}{16} \sqrt{\frac{m}{N}}. \quad (19)$$

715 **A.3. Proof of Theorem 5.6**

 716 **Lemma A.1.** Suppose that P_0 is a finite distribution on $[m_0] = \{1, \dots, m_0\}$ whose range ratio is κ_0 . Then

$$\begin{aligned} 718 \quad \min_{i \in [m_0]} P_0(i) &\geq \frac{1}{m_0 \kappa_0} \\ 719 \quad \max_{i \in [m_0]} P_0(i) &\leq \frac{\kappa_0}{m_0 + \kappa_0 - 1} \end{aligned} \tag{20}$$

 723 *Proof of Lemma A.1.* Let $B_1 := \min_{i \in [m_0]} P_0(i)$ and $B_2 := \max_{i \in [m_0]} P_0(i)$. By the definitions, we have

$$\begin{aligned} 724 \quad B_1 + (m-1)B_2 &\geq 1 \\ 725 \quad B_2 + (m-1)B_1 &\leq 1. \end{aligned}$$

 727 By the definition of range ratio, i.e. $\kappa_0 \frac{B_2}{B_1}$, we further have

$$\begin{aligned} 729 \quad B_1 + (m_0-1)\kappa_0 B_1 &\geq 1 \\ 730 \quad B_2 + \frac{m_0-1}{\kappa_0} B_2 &\leq 1. \end{aligned}$$

733 This implies

$$\begin{aligned} 734 \quad B_1 &\geq \frac{1}{m_0 \kappa_0 + 1 - \kappa_0} \geq \frac{1}{m_0 \kappa_0} \\ 735 \quad B_2 &\leq \frac{\kappa_0}{m_0 + \kappa_0 - 1}. \end{aligned}$$

□

 739 **Lemma A.2.** Suppose that $\{X_t\}$ is a sequence of random variables sampled i.i.d. from a categorical distribution $\text{Cat}(K, \mathbf{p})$
 740 where $\mathbf{p} = (p_1, \dots, p_K)$. Then we have

$$741 \quad P \left(\sum_{k=1}^K \left| \frac{1}{T} \sum_{t=1}^T \mathbb{1}(X_t = k) - p_k \right| \leq \epsilon \right) \geq 1 - \delta \tag{21}$$

745 if

$$746 \quad T \geq \frac{2K}{\epsilon^2} \log \frac{2K}{\delta}. \tag{22}$$

 748 *Proof of Lemma A.2.* Let $S := \sum_{k=1}^K \sqrt{p_k(1-p_k)}$ and $\epsilon_k := \frac{\sqrt{p_k(1-p_k)}}{S} \epsilon$ for $k = 1, \dots, K$. Then we have

$$\begin{aligned} 750 \quad &P \left(\sum_{k=1}^K \left| \frac{1}{T} \sum_{t=1}^T \mathbb{1}(X_t = k) - p_k \right| \geq \epsilon \right) \\ 751 \quad &\stackrel{(a)}{\leq} \sum_{k=1}^K P \left(\left| \frac{1}{T} \sum_{t=1}^T \mathbb{1}(X_t = k) - p_k \right| \geq \epsilon_k \right) \\ 752 \quad &\stackrel{(b)}{\leq} \sum_{k=1}^K 2 \exp \left(-\frac{T \epsilon_k^2}{2p_k(1-p_k)} \right) \\ 753 \quad &\leq 2K \exp \left(-\frac{T \epsilon^2}{2S^2} \right), \end{aligned} \tag{23}$$

761 where the inequality (a) is due to union bound and the inequality (b) is due to Chernoff bound.

 763 According to the concavity of the function $f(x) = \sqrt{x(1-x)}$, we have

$$765 \quad S = K \cdot \frac{1}{K} \sum_{k=1}^K f(p_k) \leq K f \left(\frac{1}{K} \sum_{k=1}^K p_k \right) = K f \left(\frac{1}{K} \right) = \sqrt{K-1} < \sqrt{K}. \tag{24}$$

768 Combining (23) and (24), we obtain the desired result.

□

770 We provide a constructive proof of Theorem 5.6 by designing an algorithm that recover hypergraphs from MM pre-trained
 771 models. The algorithm includes two Phases: underlying hypergraph estimation and weight estimation. In Phase 1, we
 772 estimate the underlying hypergraph by evaluating the probability of the MM pre-trained model output and selecting all
 773 hyperedges of positive probabilities. In Phase 2, we evaluate a sequence of relative weights between the hypergraphs. We
 774 estimate the weight function by those relative weights and a normalization. The algorithm is presented in Algorithm 2.
 775 Specially, we implement the weight estimation algorithm in a breadth-first style (Algorithm 3). We utilize the data structure
 776 queue to implement the algorithm. A queue Q supports two operations: $Q.\text{push_back}(x)$ that pushes the element x to the
 777 back of the queue Q and $Q.\text{pop_front}(x)$ that removes and returns the front of the queue Q .
 778

Algorithm 2 Hypergraph Estimation from MM Pre-Trained Models

Input: a MM pre-trained model \mathcal{M} , a candidate hyperedge set \mathcal{E}_0 , and a masking strategy π .

// Phase 1: underlying hypergraph estimation

Initialize $\mathcal{E} = \{\}$.

for $e \in \mathcal{E}_0$ **do**

 Apply π to e and get a masked hyperedge e^- .

if $M(e | e^-) > 0$ **then**

$\mathcal{E} = \mathcal{E} \cup \{e\}$.

end if

end for

$\mathcal{V} = \bigcup_{e \in \mathcal{E}} e$.

// Phase 2: weight estimation

Initialize $\tilde{w}(e) = 0$ for all $e \in \mathcal{E}$.

Select e_0 from \mathcal{E} and let $\tilde{w}(e_0) = 1$.

$\tilde{w} = \text{BFWEIGHTESTIMATION}(e_0, \mathcal{E}, \mathcal{M}, \pi, \tilde{w})$ (Algorithm 3).

Compute $W = \sum_{e \in \mathcal{E}} \tilde{w}(e)$.

$w = \tilde{w}/W$.

Return $\mathcal{H} = (\mathcal{V}, \mathcal{E}, w)$.

Algorithm 3 BFWEIGHTESTIMATION(e_{init} , \mathcal{E} , \mathcal{M} , π , \tilde{w})

Input: a selected hyperedge e_{init} , a hyperedge set \mathcal{E} , a MM pre-trained model \mathcal{M} , a masking strategy π , and a weight function \tilde{w} .

Initialize an empty queue Q .

$Q.\text{push_back}(e_{\text{init}})$.

while Q is not empty **do**

$e = Q.\text{pop_front}()$.

for $e' \in \mathcal{E}$ such that $e \xrightarrow{\pi} e'$ **do**

if $\tilde{w}(e') > 0$ **then**

 Continue.

end if

$\tilde{w}(e') = \frac{\pi(e^- | e)\mathcal{M}(e' | e^-)}{\pi(e^- | e')\mathcal{M}(e | e^-)} \tilde{w}(e)$.

$Q.\text{push_back}(e')$.

end for

end while

Return \tilde{w} .

We first show that the underlying hypergraph can be recovered with high probability in Phase 1. We denote $\min_{e \in \mathcal{E}_0} w_0(e)$ and $\max_{e \in \mathcal{E}_0} w_0(e)$ by c_w and C_w , respectively. By the definition of the model \mathcal{M} , it suffices to show that each hyperedge e and possible masked hypergraphs e^- (i.e., $\pi(e^- | e) > 0$) are covered by the training dataset \mathcal{D} . According to the

825 data generation process, each sample in the dataset \mathcal{D} corresponds to a pair of (e, e^-) sampled from the distribution
 826 $P((e, e^-)) = P_w(e)\pi(e^- | e)$. With slight abuse of notation, we write $(e, e^-) \in \mathcal{D}$ if \mathcal{D} contains the corresponding sample
 827 of the pair (e, e^-) . Denote the support set of $P((e, e^-))$ by S_π . By Assumptions 5.3 and 5.4, we have $|S_\pi| \leq mC_\pi$ and
 828 $P(e, e^-) \geq c_w c_\pi$ for all $(e, e^-) \in S_\pi$. Denote the event that the underlying hypergraph \mathcal{H}_1 recovered in Phase 1 satisfies
 829 $\mathcal{H} \sim \mathcal{H}_0$ by E_1 . Then we can obtain
 830

$$\begin{aligned} P(E_1^c) &= P(\exists (e, e^-) \in S_\pi, (e, e^-) \notin \mathcal{D}) \leq \sum_{(e, e^-) \in S_\pi} P((e, e^-) \notin \mathcal{D}) \\ &\leq |S_\pi| \min_{(e, e^-) \in S_\pi} P((e, e^-) \notin \mathcal{D}) \\ &\leq mC_\pi(1 - c_w c_\pi)^N. \end{aligned} \quad (25)$$

837 We then consider the weight estimation process in Phase 2, supposing that the underlying hypergraph \mathcal{H}_1 recovered in Phase
 838 1 satisfies $\mathcal{H} \sim \mathcal{H}_0$ and the isomorphism mapping from \mathcal{H} to \mathcal{H}_0 as ϕ . Notice that if we replace \mathcal{M} with \mathcal{M}_0 in Algorithm 3,
 839 the estimated weight function w satisfies $w(e) = w_0(\phi(e))$ for all $e \in \mathcal{E}$. Since we train by MM with cross-entropy loss, we
 840 have

$$\mathcal{M}(e | e^-) = \frac{\sum_{t=1}^N \sum_{k=1}^K \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-)}{\sum_{e \in \mathcal{E}} \sum_{t=1}^N \sum_{k=1}^K \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-)}. \quad (26)$$

844 We first consider only randomness over sampling masked hyperedges for given hyperedges. Denote the number of e in
 845 $\{e_t\}_{t=1}^N$ by $f_N(e)$. For any $e \in \mathcal{E}$, $e^- \sim \pi(\cdot | e)$ and $\epsilon_1 > 0$, we have
 846

$$\begin{aligned} &P\left(\left|\frac{1}{NK} \sum_{t=1}^N \sum_{k=1}^K \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-) - \frac{f_N(e)}{N} \pi(e^- | e)\right| \geq \frac{f_N(e)}{N} \pi(e^- | e) \epsilon_1\right) \\ &= P\left(\left|\frac{1}{K} \sum_{k=1}^K \left[\frac{1}{N} \sum_{t=1}^N \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-)\right] - \frac{f_N(e)}{N} \pi(e^- | e)\right| \geq \frac{f_N(e)}{N} \pi(e^- | e) \epsilon_1\right) \\ &\stackrel{(a)}{\leq} 2 \exp\left[-2K \left(\frac{f_N(e)}{N} \pi(e^- | e) \epsilon_1\right)^2\right], \end{aligned} \quad (27)$$

856 where the inequality (a) is due to Hoeffding's inequality. By union bound, we have
 857

$$\begin{aligned} &P\left(\exists (e, e^-), \left|\frac{1}{NK} \sum_{t=1}^N \sum_{k=1}^K \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-) - \frac{f_N(e)}{N} \pi(e^- | e)\right| \geq \frac{f_N(e)}{N} \pi(e^- | e) \epsilon_1\right) \\ &\leq \sum_{(e, e^-)} 2 \exp\left[-2K \left(\frac{f_N(e)}{N} \pi(e^- | e) \epsilon_1\right)^2\right]. \end{aligned} \quad (28)$$

864 When $\left|\frac{1}{NK} \sum_{t=1}^N \sum_{k=1}^K \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-) - \frac{f_N(e)}{N} \pi(e^- | e)\right| \geq \frac{f_N(e)}{N} \pi(e^- | e) \epsilon_1$ holds for all pairs of (e, e^-) , for
 865 any e, e' such that $e \leftrightarrow e'$ with e^- being the common masked hyperedge, we have
 866

$$\begin{aligned} \left|\frac{\tilde{w}(e)}{\tilde{w}(e')} - \frac{f_N(e)}{f_N(e')}\right| &= \left|\frac{\mathcal{M}(e | e^-) \pi(e^- | e') - f_N(e)}{\mathcal{M}(e' | e^-) \pi(e^- | e)} - \frac{f_N(e)}{f_N(e')}\right| \\ &\leq \left(\frac{1 + \epsilon_1}{1 - \epsilon_1} - 1\right) \frac{f_N(e)}{f_N(e')} \\ &= \epsilon_2 \frac{f_N(e)}{f_N(e')}, \end{aligned} \quad (29)$$

875 where $\epsilon_2 := \frac{1 + \epsilon_1}{1 - \epsilon_1} - 1 = \frac{2\epsilon_1}{1 - \epsilon_1}$. This implies
 876

$$(1 - \epsilon_2) \frac{f_N(e)}{f_N(e')} \leq \frac{\tilde{w}(e)}{\tilde{w}(e')} \leq (1 + \epsilon_2) \frac{f_N(e)}{f_N(e')}. \quad (30)$$

880 By Assumption 5.5, for any $e \in \mathcal{E}$, there exists a path $e_{\text{init}} = e^{(1)} \leftrightarrow \dots \leftrightarrow e^{(\ell)} = e$, $\ell \leq L$ and we have
 881
 882
 883

$$(1 - \epsilon_2)^L \frac{f_N(e)}{f_N(e_{\text{init}})} \leq \frac{\tilde{w}(e)}{\tilde{w}(e_{\text{init}})} = \tilde{w}(e) \leq (1 + \epsilon_2)^L \frac{f_N(e)}{f_N(e_{\text{init}})}. \quad (31)$$

884
 885
 886 Notice that
 887

$$\begin{aligned} w(e) &= \frac{\tilde{w}(e)}{\sum_{e' \in \mathcal{E}} \tilde{w}(e')} \\ &= \frac{\tilde{w}(e)/\tilde{w}(e_{\text{init}})}{\sum_{e' \in \mathcal{E}} \tilde{w}(e')/\tilde{w}(e_{\text{init}})} \\ &\in \left[\frac{(1 - \epsilon_2)^L}{(1 + \epsilon_2)^L} \cdot \frac{f_N(e)}{N}, \frac{(1 + \epsilon_2)^L}{(1 - \epsilon_2)^L} \cdot \frac{f_N(e)}{N} \right] \end{aligned} \quad (32)$$

896 We then obtain
 897

$$\begin{aligned} \|w - w_0 \circ \phi\|_1 &= \sum_{e \in \mathcal{E}} |w(e) - w_0(\phi(e))| \\ &= \sum_{e \in \mathcal{E}} \left| w(e) - \frac{f_N(e)}{N} + \frac{f_N(e)}{N} - w_0(\phi(e)) \right| \\ &\leq \sum_{e \in \mathcal{E}} \left| w(e) - \frac{f_N(e)}{N} \right| + \sum_{e \in \mathcal{E}} \left| \frac{f_N(e)}{N} - w_0(\phi(e)) \right| \\ &\stackrel{(a)}{\leq} \left[\frac{(1 + \epsilon_2)^L}{(1 - \epsilon_2)^L} - 1 \right] \sum_{e \in \mathcal{E}} \frac{f_N(e)}{N} + \sum_{e \in \mathcal{E}} \left| \frac{f_N(e)}{N} - w_0(\phi(e)) \right| \\ &\stackrel{(b)}{=} \left[\frac{(1 + \epsilon_2)^L}{(1 - \epsilon_2)^L} - 1 \right] + \sum_{e \in \mathcal{E}} \left| \frac{f_N(e)}{N} - w_0(\phi(e)) \right|, \end{aligned} \quad (33)$$

912 where the inequality (a) is due to (32) and the equality (b) is due to $\sum_{e \in \mathcal{E}} f_N(e) = N$. Note that $\frac{(1 + \epsilon_2)^L}{(1 - \epsilon_2)^L} - 1 \leq \frac{\epsilon}{2}$ if
 913 $\epsilon_1 \leq \frac{\epsilon}{64L}$ for ϵ sufficiently small. By (33) and Lemma A.2, with $\epsilon_1 = \frac{\epsilon}{64L}$, we have
 914

$$\begin{aligned} P(E_1 \wedge \|w - w_0 \circ \phi\|_1 \geq \epsilon) &\leq P \left(\sum_{e \in \mathcal{E}} \left| \frac{f_N(e)}{N} - w_0(\phi(e)) \right| \leq \frac{\epsilon}{2} \wedge \exists (e, e^-), \left| \frac{1}{NK} \sum_{t=1}^N \sum_{k=1}^K \mathbb{1}(e_{tk} = e, e_{tk}^- = e^-) - \frac{f_N(e)}{N} \pi(e^- \mid e) \right| \geq \frac{f_N(e)}{N} \pi(e^- \mid e) \epsilon_1 \right) \\ &\quad + P \left(\sum_{e \in \mathcal{E}} \left| \frac{f_N(e)}{N} - w_0(\phi(e)) \right| \geq \frac{\epsilon}{2} \right) \\ &\leq \sum_{(e, e^-)} 2 \exp \left[-2K \left(\frac{f_N(e)}{N} \pi(e^- \mid e) \epsilon_1 \right)^2 \right] + 2m \exp \left(-\frac{N\epsilon^2}{8m} \right) \\ &\stackrel{(a)}{\leq} \sum_{(e, e^-)} 2 \exp \left[-2K \left(\frac{c_w}{2} \pi(e^- \mid e) \epsilon_1 \right)^2 \right] + 2m \exp \left(-\frac{N\epsilon^2}{8m} \right) \\ &\leq 2mC_\pi \exp \left[-2K \left(\frac{c_w c_\pi}{128L} \epsilon \right)^2 \right] + 2m \exp \left(-\frac{N\epsilon^2}{8m} \right), \end{aligned} \quad (34)$$

932 where the inequality (a) is due to $\frac{f_N(e)}{N} \geq c_w - \frac{\epsilon}{2} \geq \frac{c_w}{2}$ when $\sum_{e \in \mathcal{E}} \left| \frac{f_N(e)}{N} - w_0(\phi(e)) \right| \leq \frac{\epsilon}{2}$ holds and ϵ is sufficiently
 933 small.
 934

935 Combining (25) and (34), we have

$$\begin{aligned}
 & P(\|w - w_0 \circ \phi\|_1 \leq \epsilon) \\
 & \geq 1 - P(E_1^c) - P(E_1 \wedge \|w - w_0 \circ \phi\|_1 \geq \epsilon) \\
 & \geq 1 - mC_\pi(1 - c_w c_\pi)^N - 2mC_\pi \exp \left[-2K \left(\frac{c_w c_\pi}{128L} \epsilon \right)^2 \right] - 2m \exp \left(-\frac{N\epsilon^2}{8m} \right) \\
 & \geq 1 - \delta,
 \end{aligned} \tag{35}$$

943 if

$$\begin{aligned}
 & mC_\pi(1 - c_w c_\pi)^N \leq \frac{\delta}{3}, \\
 & 2mC_\pi \exp \left[-2K \left(\frac{c_w c_\pi}{128L} \epsilon \right)^2 \right] \leq \frac{\delta}{3}, \\
 & 2m \exp \left(-\frac{N\epsilon^2}{8m} \right) \leq \frac{\delta}{3}.
 \end{aligned} \tag{36}$$

951 After simplification, we have

$$\begin{aligned}
 K & \geq \frac{2^{14}m^2\kappa^2L^2}{c_\pi^2\epsilon^2} \log \frac{6mC_\pi}{\delta}, \\
 N & \geq \max \left\{ \frac{2m\kappa}{c_\pi} \log \frac{3mC_\pi}{\delta}, \frac{8m}{\epsilon^2} \log \frac{6m}{\delta} \right\}.
 \end{aligned} \tag{37}$$

952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989

990 **B. Entity Alignment**

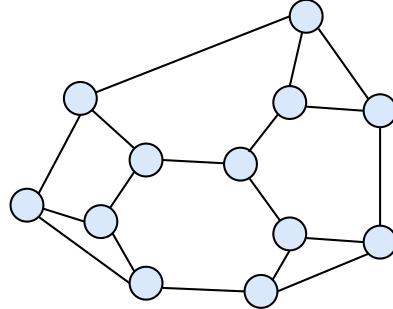
991 While we show that entity alignment is feasible without labeled pairs in theory, labeled pairs are important in practice. A
 992 possible reason is that solving the entity alignment problem is computational challenging, no known polynomial algorithms
 993 addressing the problem. The role of the labeled pairs might be reducing the inherent complexity required to solve the
 994 computational problem. Here are two examples of how the labeled pairs can help to solve the alignment problem more
 995 efficiently.
 996

997 **Example B.1.** When all m labeled pairs for the hyperedges are available, we can efficiently determine the alignment
 998 mapping between entities by leveraging hyperedges as identifiers. More concretely, we assign a unique number as the
 999 identifier to each hyperedge. Subsequently, each node is labeled with a tuple containing the identifiers of the hyperedges
 1000 it belongs to, arranged in descending order. The nodes within each hypergraph are then organized into sequences based
 1001 on their lexicographic order. Correspondence between entities is established through the alignment of nodes at identical
 1002 positions within these sequences. The entire alignment process is of computational complexity $\tilde{O}(mn)$.
 1003

1004 Example B.1 shows that we can align entities efficiently given all m labeled pairs for the hyperedges. This also means that
 1005 as long as we can find the graph matching between the line graphs of the hypergraphs, we can also align the hypergraphs
 1006 with only polynomial extra computational overhead. Therefore, we can focus on the graph matching problem of the line
 1007 graphs of the hypergraphs.

1008 WL test serves as a potent heuristic for graph matching, demonstrating efficacy across a wide range of graphs. Nonetheless,
 1009 certain graphs challenge the capabilities of low-dimensional WL tests, leading to their failure (Cai et al., 1992). Although
 1010 higher-dimensional WL tests may achieve accurate graph matching, they impose significantly greater computational demands.
 1011 Labeled pairs could help to overcome this dilemma.

1012 **Example B.2.** Frucht graph (Figure 5) is a regular graph without non-trivial automorphism (Frucht, 1939). 1-WL does not
 1013 work for Frucht graph because of its regularity. While higher-dimensional WL tests are applicable, they are significantly
 1014 less efficient. However, if a labeled pair is identified, one can exclude the nodes in the label pair from both graphs and apply
 1015 the 1-WL test to the resulting subgraphs, leading to efficient graph matching.
 1016



1017
 1018 *Figure 5.* Frucht graph.
 1019
 1020
 1021
 1022
 1023
 1024
 1025
 1026
 1027
 1028
 1029
 1030
 1031
 1032
 1033
 1034
 1035
 1036
 1037
 1038
 1039
 1040
 1041
 1042
 1043
 1044

1045 **C. Experiments**1046 **C.1. Synthetic Relational Learning**1047 **C.2. Data**1049 **C.2.1. GRAPH STRUCTURES**1051 When the number of nodes is n , the different graph structures (Figure 6) are

1053 • STAR:

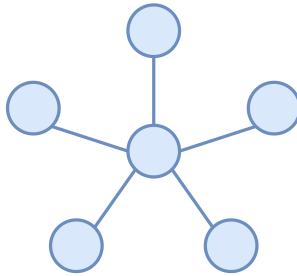
- $\mathcal{V} = \{0, 1, \dots, n - 1\}$;
- $\mathcal{E} = \{\{0, i\} \mid i = 1, \dots, n - 1\}$;

1058 • X:

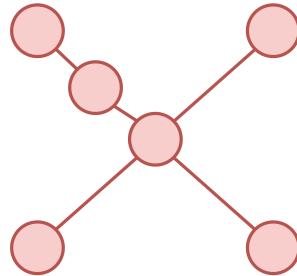
- $\mathcal{V} = \{0, 1, \dots, n - 1\}$;
- $\mathcal{E} = \{\{0, k\} \mid k = 1, 2, 3, 4\} \cup \{\{4i + k, 4i + k + 4\} \mid 4i + k + 4 \leq n - 1\}$;

1063 • CHAIN:

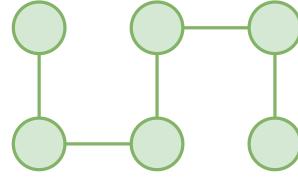
- $\mathcal{V} = \{0, 1, \dots, n - 1\}$;
- $\mathcal{E} = \{\{i, i + 1\} \mid i = 0, \dots, n - 2\}$.



(a) STAR.



(b) X.



(c) CHAIN

1084 *Figure 6. Different graph structures ($n = 6$).*1086 **C.2.2. DATA GENERATION**

1088 Each node of the graph is attached with a token, starting from "a" and following the order of tokens of BERT's tokenizer.
 1089 Each edge is assigned a weight, sampled from $\{w_{\min}, w_{\max}\}$. Specifically, we use $w_{\min} = 1.0, w_{\max} = 1.0$ for $\kappa = 1.0$,
 1090 $w_{\min} = 1.0, w_{\max} = 10.0$ for $\kappa = 10.0$, and $w_{\min} = 1.0, w_{\max} = 100.0$ for $\kappa = 100.0$ in our experiments. Then the
 1091 weights of the graph are normalized. When generating data, we first sample an edge from the graph, with probability
 1092 proportional to the the weights. We then concatenate the tokens of the edges with a random order. Tokens are separated by
 1093 spaces to avoid that they are combined by the tokenizer. For each graph, we generate 100000 samples for each graph, with
 1094 80000 samples for training, 10000 samples for validation, and 10000 samples for testing.

1095 **C.2.3. MODEL**

1097 We choose BERT as our underlying PTM. We use the implementation of HuggingFace (Wolf et al., 2020) with the default
 1098 tokenizer and the default configuration of BERT.
 1099

1100 C.2.4. PRE-TRAINING

1101 We pre-train our model by MLM from scratch. For the masking strategy, we mask one of the tokens in a sample uniformly
 1102 at random. We train the model by AdamW, with the initial learning rate 2×10^{-5} , weight decay 0.01, the cosine scheduler.
 1103 The other hyperparameters of AdamW are the same as the default of HuggingFace TrainerArguments. We pre-train the
 1104 model for 100 epochs. Per-device training batch size is 256. The experiments are run on a server with Ubuntu. All the
 1105 models are trained on two NVIDIA GeForce RTX 3090 GPUs.

1107 C.3. Real-World Relation Evaluation
1108

1109 To extract a subgraph from ConceptNet, we first choose a source entity, query for the k most related entities, and then repeat
 1110 such a process for the returned entities. We adopt a breadth-first-search-like generation process to choose a subset of entities
 1111 and construct a subgraph by considering k most related entities within these chosen entities and the generation process is
 1112 limited to some depth d (the source entity are of depth 0).

1113 In the real-world relation evaluation, we assess the LLMs' understanding of entity relationships by querying the k most
 1114 related entities within a specified set. We employ the prompt: "Consider the following concepts: [ENTITIES]. Suppose that
 1115 these concepts are nodes of an undirected graph. For each concept, consider $[k]$ most related concepts. According to the
 1116 relations between these concepts, which edges should be included? Please answer with an edgelist.", where "[ENTITIES]"
 1117 and "[k]" are placeholders for the actual entity set and the number of top-related entities, respectively. The LLMs will
 1118 generate responses comprising edgelists, potentially accompanied by additional text, which are then utilized to construct
 1119 relational graphs. See Figure 7 for an illustration. These graphs are compared with the corresponding subgraphs extracted
 1120 from ConceptNet. See Figure 7 for an illustration. These graphs are compared with the corresponding subgraphs extracted
 1121 from ConceptNet.



1122
1123 Consider the following concepts: table, furniture, on, house,
1124 room, off, building, home, space, plate. Suppose that these
1125 concepts are nodes of an undirected graph. For each
1126 concept, consider 2 most related concepts. According to the
1127 relations between these concepts, which edges should be
1128 included? Please answer with an edgelist.



1129
1130
1131
1132
1133
1134 (table, furniture), (table, plate), (furniture, house), (on, table),
1135 (on, plate), (house, room), (room, furniture), (off, on), (off, table),
1136 (building, house), (building, room), (home, house), (home,
1137 room), (space, room), (space, building), (plate, on).

1141 *Figure 7.* An example of real-world relation evaluation using GPT-4. Here, "table" is the source entity, with $k = 2$ indicating the two
 1142 most closely related entities considered for generating the relational graph.

1143 Figures 8-17 are the evaluation results for all the source entities listed in Table 1.

1144 The correspondences between the entities and the letters used in the above figures are summarized in Tables 2 and 3.

	A	B	C	D	E	F
CAKE	CAKE	BIRTHDAY	DESSERT	CELEBRATION	LIZARD	PARTY
DOG	DOG	BARK	HOUSE	TREE	BUILDING	HOME
FLY	FLY	INSECT	BUG	FLEA	MEADOW	WIRETAP
HUMAN	HUMAN	SCHOOL	HOME	LEARN	PLACE	HOUSE
JACKET	JACKET	COAT	SHELL	CLOSET	MATERIAL	HUSK
ORANGE	ORANGE	FRUIT	PEEL	EAT	YOU	SKIN
PAPER	PAPER	WRITE	SHEET	PEN	BED	CLOSET
SEA	SEA	OCEAN	WATER	SAIL	LAKE	DRINK
TABLE	TABLE	FURNITURE	ON	HOUSE	ROOM	OFF
ZOO	ZOO	ANIMAL	ELEPHANT	SQUIRREL	CIRCUS	TRUNK

Table 2. The correspondences between the entities and the letters (Part 1).

	G	H	I	J	K	L
CAKE	GARDEN	ROCK	-	-	-	-
DOG	PLANT	GROW	TOWN	BANK	PLACE	-
FLY	DOG	WOOD	HAYFIELD	INVESTIGATION	TAP	-
HUMAN	STUDY	KNOWLEDGE	LOCATION	BED	BUILDING	-
JACKET	BEDROOM	CLOTHES	WOOD	WOOL	CHAFF	-
ORANGE	FOOD	HUNGER	ME	BODY	MOLE	-
PAPER	OFFICE	POCKET	SLEEP	FURNITURE	BEDROOM	CLOTHES
SEA	BOAT	WIND	POND	LIQUID	BEVERAGE	-
TABLE	BUILDING	HOME	SPACE	PLATE	-	-
ZOO	RODENT	BALLOON	ATTIC	CAR	-	-

Table 3. The correspondences between the entities and the letters (Part 2).

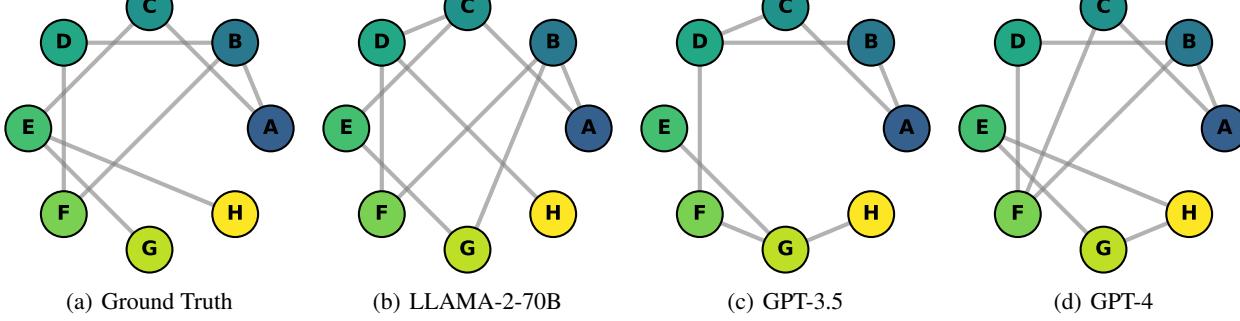


Figure 8. Cake.

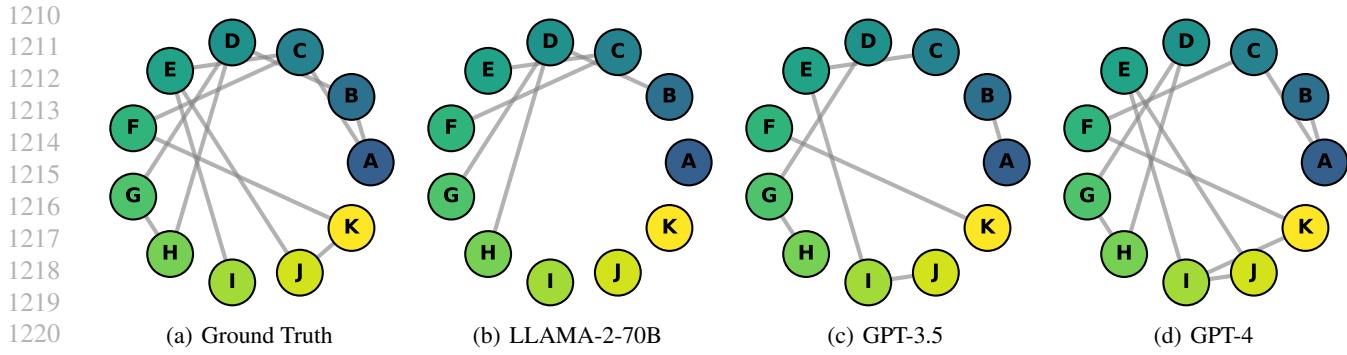


Figure 9. Dog.

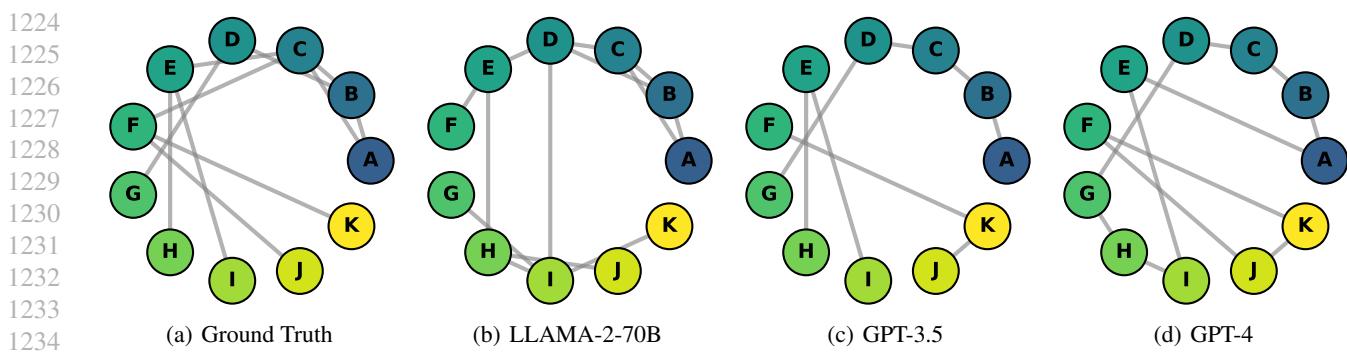


Figure 10. Fly.

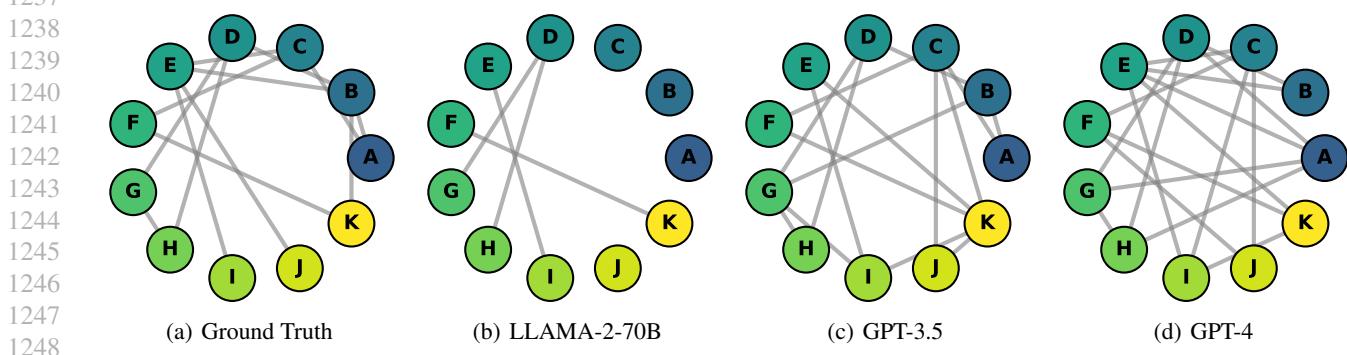


Figure 11. Human.

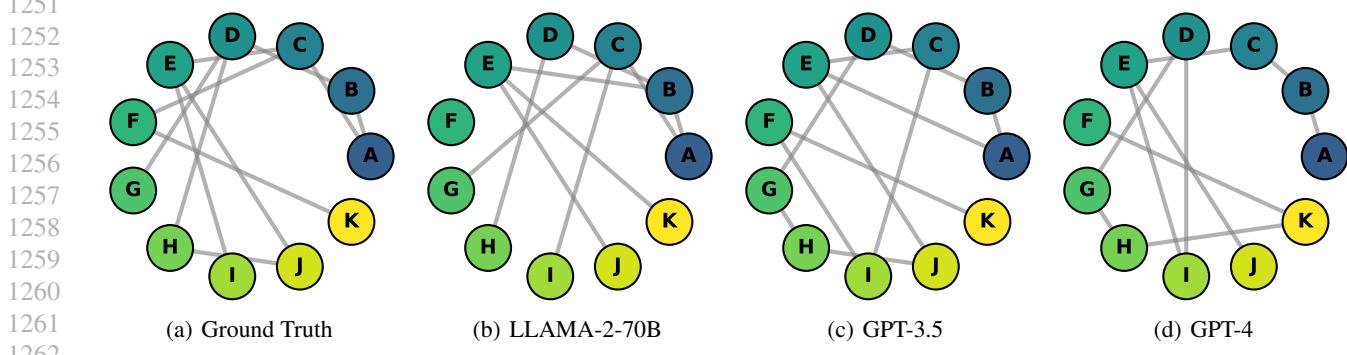


Figure 12. Jacket.

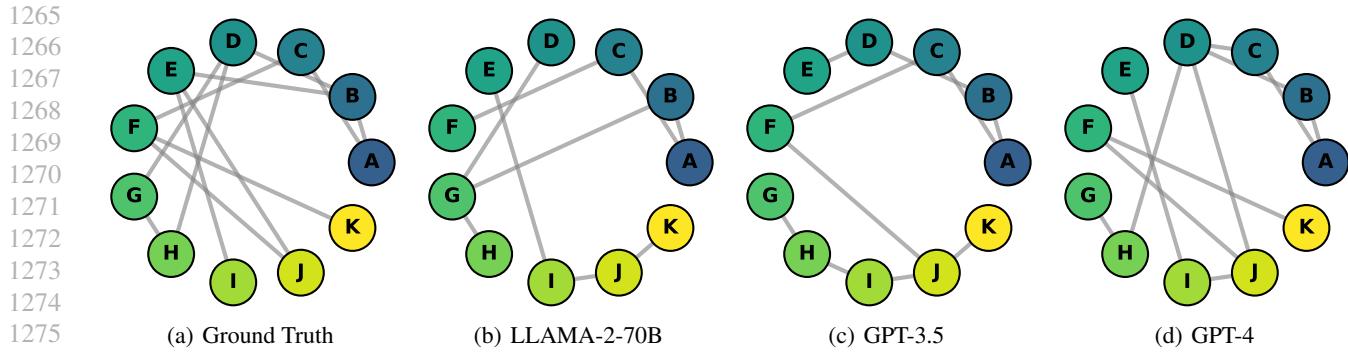


Figure 13. Orange.

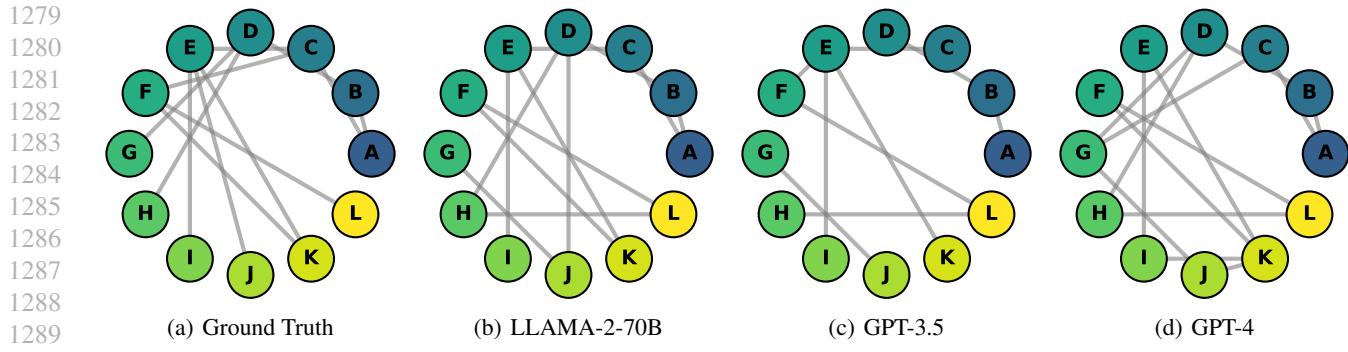


Figure 14. Paper.

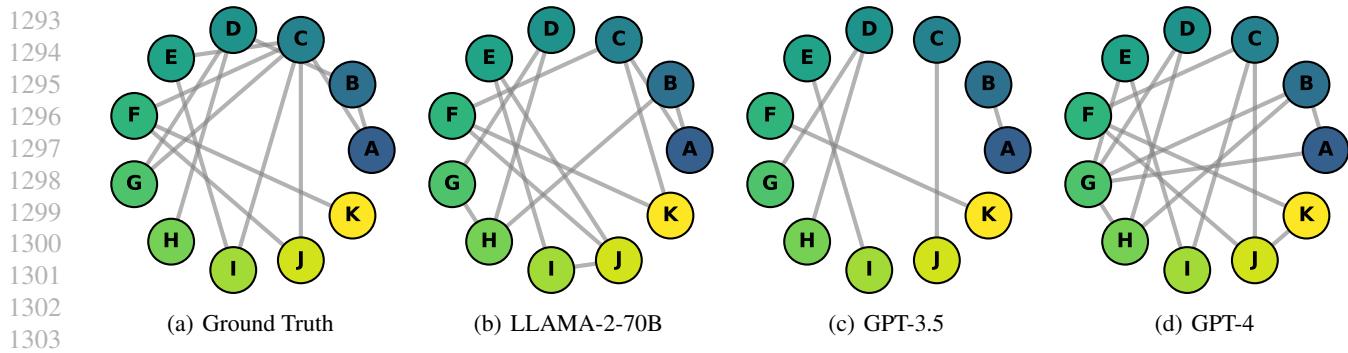


Figure 15. Sea.

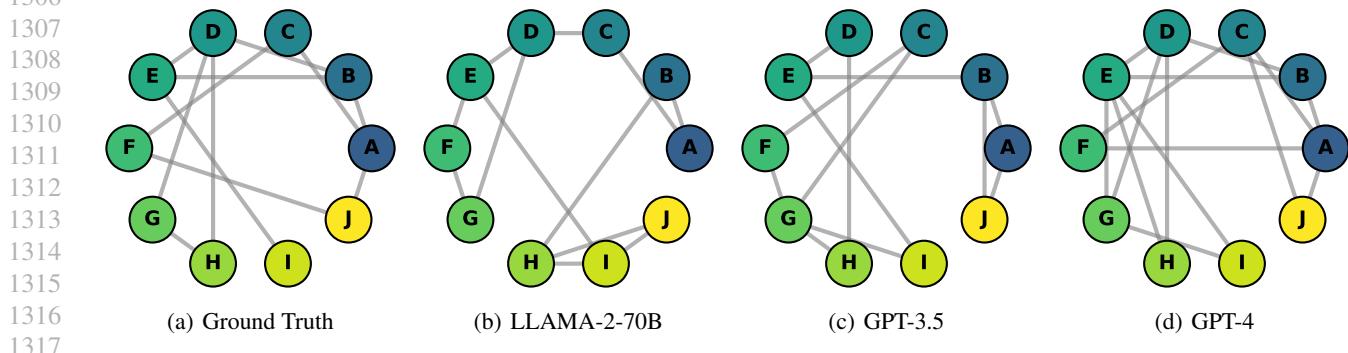


Figure 16. Table.

1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340

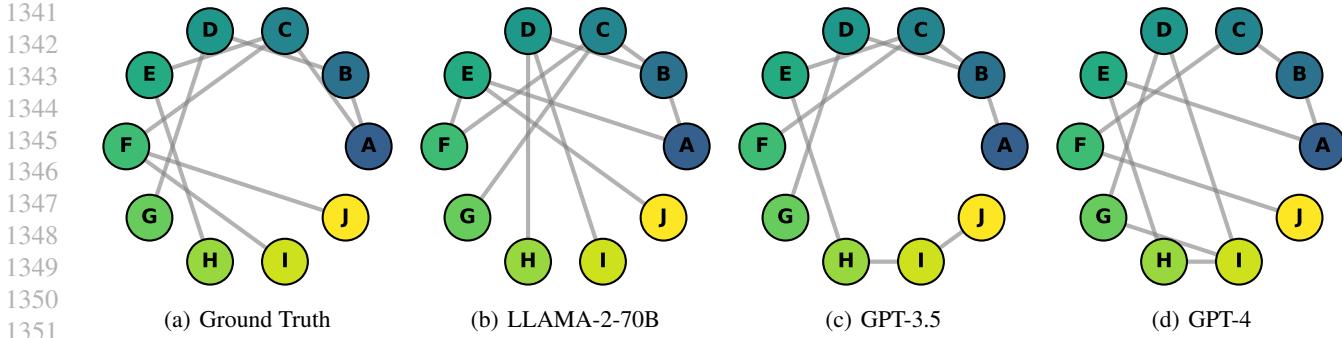


Figure 17. Zoo.