

UNIVERSIDADE FEDERAL FLUMINENSE

VÍTOR NASCIMENTO LOURENÇO

Learning Attention-enhanced Knowledge
Graphs Representations through Entities'
Context and Semantic Paths

NITERÓI

2021

UNIVERSIDADE FEDERAL FLUMINENSE

VÍTOR NASCIMENTO LOURENÇO

Learning Attention-enhanced Knowledge Graphs Representations through Entities' Context and Semantic Paths

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação.

Orientadora:

Profa. D.Sc. Aline Marins Paes Carvalho

NITERÓI

2021

Ficha catalográfica automática - SDC/BEE
Gerada com informações fornecidas pelo autor

L8921 Lourenço, Vítor Nascimento
Learning Attention-enhanced Knowledge Graphs Representations
through Entities' Context and Semantic Paths / Vítor
Nascimento Lourenço ; Aline Marins Paes Carvalho,
orientadora. Niterói, 2021.
73 f. : il.

Dissertação (mestrado)-Universidade Federal Fluminense,
Niterói, 2021.

DOI: <http://dx.doi.org/10.22409/PGC.2021.m.11439150702>

1. Aprendizado de Máquina. 2. Grafos de Conhecimento. 3.
Produção intelectual. I. Carvalho, Aline Marins Paes,
orientadora. II. Universidade Federal Fluminense. Instituto de
Computação. III. Título.

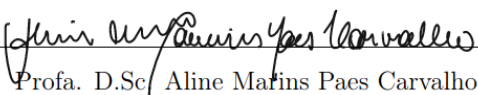
CDD -

VÍTOR NASCIMENTO LOURENÇO


Learning Attention-enhanced Knowledge Graphs Representations through Entities'
Context and Semantic Paths

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação.

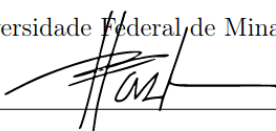
BANCA EXAMINADORA




Prof.ª D.Sc. Aline Marins Paes Carvalho - Orientadora,
Universidade Federal Fluminense



Prof. D.Sc. Adriano Alonso Veloso,
Universidade Federal de Minas Gerais



Prof. D.Sc. Alexandre Plastino de Carvalho,
Universidade Federal Fluminense



D.Sc. Sandro Rama Fiorini,
IBM Research

Niterói
2021

"To boldly go where no one has gone before."

Capt. Jean-Luc Picard

Acknowledgements

To professor Aline Paes, my advisor, who, since the undergraduate studies, believed in my potential, guiding me during the development of this work, helping me with valuable comments and suggestions, allowing me to work freely and independently.

To doctor Sandro Fiorini and professors Ariano Veloso and Alexandre Plastino for accepting to be part of the Examining Committee and for providing insightful contributions.

To my friends and people that I have worked with during my internship to help me grow as a professional and researcher.

To my parents, brother, and girlfriend for their many abdications and all personal support given so that I persevere in my goal.

I am deeply grateful to each of you. So to you, I thank you.

Resumo

O conhecimento humano geralmente é baseado na modelagem de elementos que compõem o mundo. Os elementos, suas propriedades e suas relações com outros elementos dão origem a uma rede de dados. Como parte da coleta e estruturação de dados relacionais, Bases de Conhecimento (BCs) são compilações dessas redes em estruturas processáveis por máquina. Dada a natureza evolutiva da informação, que, essencialmente, continua crescendo, sendo reformulada e transformada, as BCs são naturalmente ruidosas e incompletas. Uma vez que elas são adotadas em aplicações científicas e industriais, há uma alta demanda de soluções para completar suas informações. Originalmente, paradigmas simbólicos foram usados para abordar esse problema. Porém, motivados em parte pelos problemas de escalabilidade das soluções simbólicas, vários trabalhos recentes atacam o desafio de completar GCs aprendendo representações distribucionais, *i.e.*, *embeddings* de entidades e relações, seguido pela aplicação dos *embeddings* na predição de novas relações entre as entidades. Apesar do seu recente sucesso, grande parte desses métodos concentram-se em aprender *embeddings* a partir apenas dos vizinhos locais das relações. Como resultado, eles podem falhar em capturar informações de contexto dos GCs ao negligenciar as dependências de longo prazo e a propagação da semântica das entidades. Nessa dissertação, nós investigamos a completação de BCs, especificamente suas representações em grafo (*grafos de conhecimento*, GCs), através de uma perspectiva de aprendizado de representações, focando na predição de novas relações entre as entidades existentes. Para tanto, nós propomos o \mathcal{A} EMP (**A**ttention-based **E**mbeddings from **M**ultiple **P**atterns – *Embeddings* baseados em Atenção de Múltiplos Padrões), um novo modelo distribucional de inspiração simbólica para aprendizado de representações contextualizadas por meio de: (i) aquisição de informações de contexto das entidades através de um esquema de passagem de mensagens baseado em atenção, que captura a semântica local das entidades enquanto foca em diferentes aspectos da vizinhança; e (ii) captura do contexto semântico, aproveitando os caminhos e suas relações entre as entidades. Nós conduzimos experimentos em grafos de conhecimento referência, comparando os resultados do \mathcal{A} EMP com as abordagens estado-da-arte em predição de relações, mostrando que o \mathcal{A} EMP supera ou compete com esses métodos. Além disso, nós demonstramos que \mathcal{A} EMP tem potencial de escala para GCs grandes, lidando com até milhões de triplas. Nossas descobertas empíricas trazem percepções em como mecanismos de atenção podem melhorar a representação do contexto das entidades e como a combinação de entidades e contextos de caminhos semânticos melhoram a representação geral das entidades e, assim, as capacidades gerais de predição de relações.

Palavras-chave: Grafo de conhecimento; aprendizado de representações; *embedding*; arcabouço de passagem de mensagens; mecanismo de atenção.

Abstract

Human knowledge often establishes itself on modeling elements that compose the world. The elements, their properties, and their relationships with other elements give rise to a network of data. As part of gathering and structuring the relational data, knowledge bases (KBs) compile these networks into machine-readable structures. Due to the evolving nature of information, which essentially keeps growing, rephrasing, and transforming, KBs are naturally noisy and incomplete. Since they are extensively adopted in scientific and industrial applications, there is a high demand for solutions that complete their information. Originally, symbolic paradigms were used to approach this challenge. Nevertheless, motivated by the scalability issues of the symbolic solutions, several recent works tackle KG completion challenge by learning distributed representations, *i.e.*, embeddings for entities and relations, followed by employing them to predict new relations among the entities. Despite their aggrandizement, most of these methods concentrate only on the local neighbors of a relation to learn the embeddings. As a result, they may fail to capture the KGs' context information by neglecting long-term dependencies and the propagation of entities' semantics. In this dissertation, we address the completion of KBs, specifically their graph representations (*knowledge graphs*, KGs), through a representation learning perspective focusing on predicting new relationships among existing entities. For this purpose, we propose \mathcal{A} EMP (**A**ttention-based **E**MBEDdings from **M**ultiple **P**atterns), a novel symbolic-inspired distributional model for learning contextualized representations by: (i) acquiring entities' context information through an attention-enhanced message-passing scheme, which captures the entities' local semantics while focusing on different aspects of their neighborhood; and (ii) capturing the semantic context, by leveraging paths and their relationships between entities. We conduct experiments on knowledge graph benchmarks, comparing \mathcal{A} EMP's results with state-of-the-art approaches on relation prediction, showing that \mathcal{A} EMP either outperforms or competes with these methods. Also, we demonstrate that \mathcal{A} EMP has the potential to scale to larger KGs, handling up to millions of triples. Our empirical findings draw insights into how attention mechanisms can improve entities' context representation and how the combination of entities and semantic paths contexts improves the general representation of entities and, then, the overall relation prediction capabilities.

Keywords: Knowledge graph; representation learning; embedding; message passing framework; attention mechanism.

List of Figures

1.1	Example of a knowledge graph about actors and films.	2
1.2	Example of a knowledge graph under \mathcal{AEMP} perspective. Gray circles indicate the head and tail entities. Dashed-arrows are potentially missing relations. Yellow-shaded arrows indicate the semantic paths between the head and tail. Green, red, and blue-shaded areas represent the different attention mechanisms used in \mathcal{AEMP}	6
2.1	A knowledge graph about films and actors.	11
2.2	Example of the representations of a knowledge graph's entities and relationships in 3-dimensional euclidean space.	13
2.3	Example of the representations of a semantic path between entities entities.	14
4.1	Overview of \mathcal{AEMP} architecture. Boxes represent the vector representations produced in each step. Gray-shaded illustrates the message-passing scheme. Red-shaded points out each hop used on local attention. Green-shaded expresses the iterations used on global attention. Blue-shaded indicates the randomly selected relationships used on random attention. Yellow-shaded indicates the semantic paths.	26
4.2	Example of knowledge graph under the \mathcal{AEMP} highlighting \mathcal{AEMP} 's mechanisms. Similar to the knowledge graph illustrated in Figure 1.2, gray circles indicate the head and tail entities. Dashed-arrows are potentially missing relations. Yellow-shaded arrows indicate the semantic paths between the head and tail. Green, red, and blue-shaded areas represent global, local, and random attention mechanisms, respectively, used in \mathcal{AEMP}	28
4.3	Example of the representation of the semantic paths between head and tail entities, <i>i.e.</i> , <code>PatrickStewart</code> and <code>IanMcKellen</code> entities, respectively.	32

5.1	Confusion matrices of the ground truth relations and predicted relations by each \mathcal{A} EMP's variation. The heatmap indicates the Hit@1 metric varying from 0 to 1, and axes are in descending order (top-bottom for y-axis, and left-right for the x-axis) regarding the number of triples in which the relation is the predicate.	46
5.2	Boxplot of the Hit@1 results from \mathcal{A} EMP and its subset variations using (or not) the semantic paths' representation to predict new relations.	47
5.3	Average Hit@1 results from \mathcal{A} EMP and its subset variations regarding the variation of entities context hops, semantic paths length, and a sample of entities neighbors hyperparameters.	48
5.4	Boxplot of the training time from \mathcal{A} EMP regarding different sized KGs. . .	49

List of Tables

2.1	Notations used in this dissertation.	9
3.1	Comparison between literature approaches, their used mechanisms, and tasks in which they were previously evaluated.	23
4.1	Notations used in this dissertation.	27
5.1	Examples of facts in the format (<i>subject, predicate, object</i>) from datasets WN18 and FB15k.	36
5.2	Datasets statistics summary.	36
5.3	Search space of the ÆMP’s hyperparameters.	37
5.4	Relation prediction results on WN18 and WN18RR datasets. (L), (G), (R) stands for ÆMP local, global, and random attention patterns, respectively. [*]: Results are taken from [67]. The best result value is in bold and second best result value is underlined.	41
5.5	Relation prediction results on FB15k and FB15k-237 datasets. (L), (G), (R) stands for ÆMP local, global, and random attention patterns, respectively. [*]: Results are taken from [67]. The best result value is in bold and second best result value is underlined.	42
5.6	Best (1st) and the second-best (2nd) MRR metric in each dataset.	43
5.7	Ablation studies parametrization settings.	43

List of Abbreviations and Acronyms

$\mathcal{A}EMP$: Attention-based Embeddings from Multiple Patterns;
AI	: Artificial intelligence;
CP	: Canonical Polyadic;
GAT	: Graph attention network;
GCN	: Graph convolution network;
ILP	: Inductive logic programming;
KB	: Knowledge base;
KG	: Knowledge graph;
MR	: Mean rank;
MRR	: Mean reciprocal rank;
MTM	: Machine translation model;
NLP	: Natural language processing;
PRA	: Path-Ranking Algorithm;
RNN	: Recurrent neural network.

Contents

1	Introduction	1
1.1	Motivation	2
1.2	Objectives	4
1.3	Methodology	5
1.3.1	Problem formulation	5
1.3.2	Proposed solution	5
1.4	Contributions	7
1.5	Dissertation organization	7
2	Background	9
2.1	Relational data and knowledge graphs	9
2.2	Learning knowledge graphs' representations	11
2.2.1	Facts embedding	12
2.2.2	Additional information embedding	13
2.2.2.1	Semantic paths	13
2.2.2.2	Entities context	14
2.2.2.3	Message passing framework	15
2.3	Attention mechanisms	15
2.3.1	General attention	16
2.4	Graph attention	16
2.4.1	Knowledge graph attention mechanisms	17
2.5	Discussion	18

3	Related Work	19
3.1	Symbolic statistical relational learning	19
3.2	Knowledge graph representation	20
3.2.1	Entities context	21
3.2.2	Relational paths	21
3.3	Attention mechanisms	22
3.3.1	Attention-based embeddings for general graphs	23
3.3.2	Attention-based embeddings for knowledge graphs	23
3.4	Discussions	23
4	Learning Attention-based Representations from Multiple Patterns	25
4.1	Learning entities context representations	26
4.1.1	Message passing scheme	27
4.1.2	Attention-enhanced message passing	28
4.2	Learning semantic paths representations	31
4.3	Training Objective	32
4.4	Discussions	33
4.4.1	Design Alternatives	33
4.4.1.1	Message passing scheme	33
4.4.1.2	Path representation	34
5	Experimental Results	35
5.1	Experimental Settings	35
5.2	Results	39
5.3	Ablation Studies	43
5.4	Discussions	45
6	Conclusions	50

Contents	xii
6.1 Limitations	51
6.2 Future work	51
References	53

Chapter 1

Introduction

Human knowledge often establishes itself on modeling the elements that compose the world. The elements, their properties, and the existing relations among elements give rise to a network of data. As part of gathering and structuring the relational data, the network of data can be posed as sets of information (*i.e.*, facts, beliefs, and rules) circa elements, often called *entities*. Knowledge bases (KBs) play a fundamental role in compiling this network of knowledge, structuring and storing information, its properties, and its relationships. The KBs are a general formulation of a technology to compile, structure, and store the data networks into machine-readable structures.

One of the first attempts of representing through standards and structuring these data networks (also called *linked data*) is the Semantic Web¹, which is a W3C² standard that extends the World Wide Web aiming to handle linked data. The Semantic Web plays a fundamental role in compiling, structuring, storing, and providing access to linked data through implementing standards aiming to make information machine-readable. As an example of these standards and their technologies, we highlight RDF³, OWL⁴, and SPARQL⁵, which provides an interchangeable structure for linked data, a logic-based structure for the representation of complex semantics among the linked data, and a querying interface for accessing information, respectively.

Existing compiled KBs such as WordNet [46], YAGO [56, 60], Freebase [6], and NELL [47] organize open information over topological and non-topological relations. The KBs might follow the standards proposed in the Semantic Web, provide their own

¹<https://www.w3.org/standards/semanticweb/>

²<https://www.w3.org/>

³<https://www.w3.org/RDF/>

⁴<https://www.w3.org/OWL/>

⁵<https://www.w3.org/TR/rdf-sparql-query/>

standards, or having conversions in between. They have succeeded as core resources on knowledge-related tasks such as question answering [71], query expansion [12], information retrieval [41], recommender systems [55], and even commonsense reasoning [38]. Typically, the information in these bases is organized according to a conceptual structure in the form of triples or emphasizing the relational character of their data in the form of graphs, specifically, *knowledge graphs* (KGs), as illustrated in Figure 1.1.

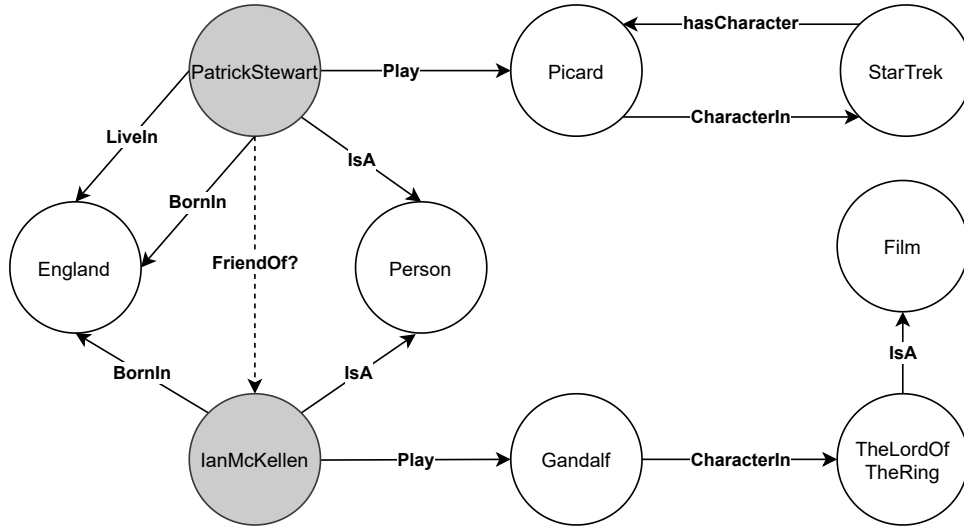


Figure 1.1: Example of a knowledge graph about actors and films.

The KGs successfully leverage industrial scale knowledge learning, inference, and reasoning. Besides, it can be used as data sources to methods that learn patterns from relational data, such as techniques from statistical relational AI [21], and, also, enabling a range of new graph-based methods. Thus, knowledge graphs⁶ are also broadly adopted in industrial applications such as search-engines⁷, social networks⁸, and question-answering systems⁹.

1.1 Motivation

Knowledge bases are usually created by manually providing annotations about facts from a domain or automatically capturing information from the web. In manual annotations, KBs have specialists that manually annotate the entities and their relationships, such as WordNet in the linguistic domain. In automatic capturing, such as done in NELL [47],

⁶Google Knowledge Graph, Pinterest Knowledge Graph, *etc.*

⁷Google, Yahoo, Bing, *etc.*

⁸Twitter, Facebook, Pinterest, LinkedIn, *etc.*

⁹WolframAlpha, Amazon's Alexa, Apple's Siri, *etc.*

the process consists of reading web content and developing a general representation of the world through automatically inferring entities, properties, and relationships.

Despite the success of knowledge graphs (KGs) as core resources on a large plethora of tasks, real-world KGs usually experience incompleteness and noisy information in both information acquisition cases. For instance, situations that lead KGs to be incomplete include not foreseen circumstances and outdated information. Human mistakes in manual annotations and capture errors in automatic acquisition lead KGs to be noisy. Like so, they are strictly dependable on new information acquisition, which is often hard to obtain and noisy prone once the information continuously evolves over time [37]. Thus, an essential aspect regarding KGs arises the existence of automatic methods to complete their information. This capacity involves solving the prediction of missing relations among entities in the KG. For instance, taking Figure 1.1 in consideration, predict the missing relation **FriendOf** between **PatrickStewart** and **IanMcKellen** that relates them as friends.

Several previous works have proposed techniques to predict and infer new relationships from the existing information to address the capabilities of completing knowledge graphs. Such methods can be roughly divided into two categories [16]: distributional [7, 61, 64, 73], which are mostly driven by the recent advances in learning representations from symbolic data; and symbolic methods [18, 68], that leverage logic and domain knowledge to infer new relationships. The distributional methods aim to learn and operate latent representations, *i.e.*, embedding vectors, of entities and relations. These methods rely on encoding the interactions between entities neighborhood, *i.e.*, the entities local relationships, into low-dimensional dense vectors, which allow for new relationships between entities to be predicted from interactions of the entities embeddings [53]. Differently, the symbolic methods [18, 68] aim to predict new relations from the observed examples in the knowledge graph. Generally, these methods extract rules from the examples in order to model logical patterns, *i.e.*, semantic paths between entities [53]. Methods grounded in the symbolic paradigm can reason over complex relational paths due to their rule-based grounding. However, they are well-known to suffer from scalability issues [18]. To learn concepts, these methods need to have a target relation in order to reason [18, 50]. An initial attempt is to try using all possible relations as targets to overcome this issue, which drives the scalability issues. Consequently, these methods have limitations in their rule-inference search space. In contrast, the embedding-based methods, grounded in the distributional paradigm, are capable of only learning local structures, requiring further enhancements to allow for more generality. Nonetheless, they scale, being capable of

operating on knowledge graphs with millions of facts.

As a consequence, arise the need for hybrid methods, such as the ones covered under the neural-symbolic [18, 20] umbrella (which are out-of-scope of this dissertation), and distributional methods capable of handling the semantics of entities relationships, *i.e.*, context (the scope of this dissertation). The recent adoption of attention mechanisms [9] aims to leverage input’s context to learn more general representations. As stated by Professor Yousha Bengio¹⁰ in the 2019 AI Debate [4]:

“Attention is interesting because it changes the very nature of what a standard neural net can do in many ways. It creates dynamic connections that are created on the fly based on context. It is even more context-dependent, but in a way that can favour what Gary (Professor Garry Marcus¹¹) called free generalization that I think is important in language and in conscious processing.”,

the attention mechanisms are examples of leveraging contextual information to the distributional paradigms. Path-based approaches, such as the Path-Ranking Algorithm (PRA) [34], are another example of learning representations inspired in symbolic approaches, which targets learning inference paths in KGs [33, 35].

1.2 Objectives

Motivated by the aforementioned observations, the main objective of this dissertation is guided by the following statement:

A symbolic-inspired distributional solution is able to automatically complete knowledge graphs by predicting new relations among entities leveraging the advantages from both aspects. Thus, the solution: (i) should be scalable by learning dense latent representations from entities and relations; and (ii) the learned representations should be contextualized, encoding entities local structure (entities context) and logical patterns from entities chained relationships (semantic paths).

The secondary objectives are:

¹⁰https://en.wikipedia.org/wiki/Yoshua_Bengio

¹¹https://en.wikipedia.org/wiki/Gary_Marcus

1. to provide a new competing technique for knowledge graph completion;
2. to evaluate if contextualized representations are able to enhance the overall prediction results; and
3. to evaluate if a symbolic-inspired distributional solution has, indeed, the potential to scale to larger knowledge graphs.

1.3 Methodology

Our methodology is divided twofold. First, we formally present the knowledge base completion challenge through a relation prediction perspective, as described in Section 1.3.1. Subsequently, developing the proposed solution for the challenge is the second part of our methodology (Section 1.3.2).

1.3.1 Problem formulation

Given a knowledge base represented by a labeled multi-digraph $\mathcal{KG} = (\mathcal{E}, \mathcal{R}, \mathcal{F})$, *i.e.*, a knowledge graph, where \mathcal{E} is the set of nodes that represents the entities, \mathcal{R} the set of edges' labels that represents the relations, and $\mathcal{F} : \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ the set of edges that represents the existing facts, *i.e.*, existing triples. Our goal is, given a pair of entities (h, t) , where $h \in \mathcal{E}$ and $t \in \mathcal{E}$, to predict the relationship between these two entities. The outcome assembles a new (missing) fact $f' = (h, r, t)$, where $r \in \mathcal{R}$ is the predicted relation, and $\{f' \in \mathcal{F} | P(r|h, t)\}$ is the candidate triple, where $P(r|h, t)$ is the probability of existing a relation r between the pair of entities (h, t) .

A commonly seen variation of this problem is the link prediction task [7, 51, 64]. The main difference between both tasks is that the link prediction task aim to predict an entity e given a pair entity-relation, while the *relation prediction* task aims to predict a relationship between two given entities, as described above. In this work, we focus specifically in the relation prediction task.

1.3.2 Proposed solution

in this dissertation, we address the relation prediction task as a process towards completing knowledge graphs (Section 1.3.1) by designing a novel embedding-based model called \mathcal{AEMP} (**A**ttention-based **E**MBEDdings from **M**ultiple **P**atterns). \mathcal{AEMP} aims at the

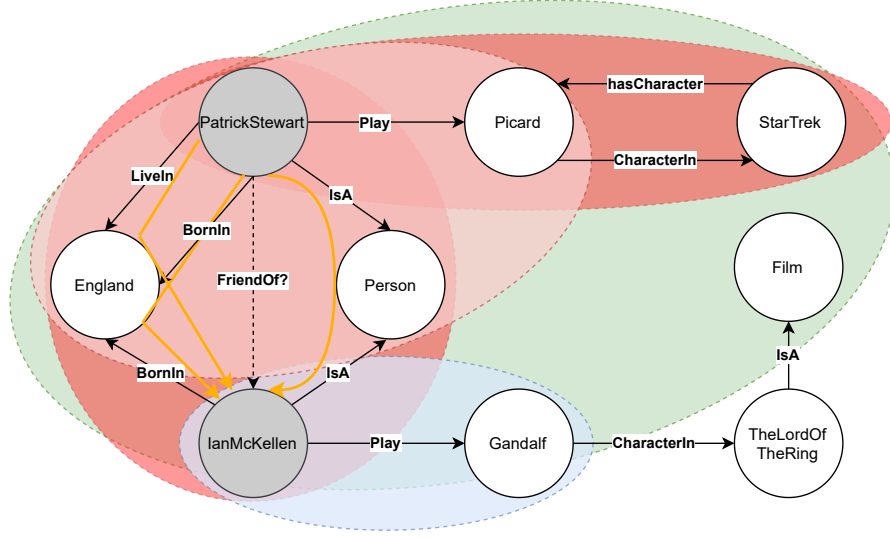


Figure 1.2: Example of a knowledge graph under \mathcal{AEMP} perspective. Gray circles indicate the head and tail entities. Dashed-arrows are potentially missing relations. Yellow-shaded arrows indicate the semantic paths between the head and tail. Green, red, and blue-shaded areas represent the different attention mechanisms used in \mathcal{AEMP} .

major advantages of distributional and symbolic paradigms: local structure learning and scalability from the distributional paradigm and generality with semantic paths from the symbolic paradigm. To achieve such a goal, \mathcal{AEMP} learns contextualized representations from the combination of entities context and semantic paths. The key insights of \mathcal{AEMP} are based on the perception that multiple patterns exist to relate entities. Similar to a word in a sentence, entities local neighborhood evidences their structure and properties, while semantic paths relate long-term dependencies among them. Besides, the unification of the contextual information provides enhanced information to predict new relations among existing entities in knowledge graphs.

Figure 1.2 exemplifies the patterns and mechanisms that \mathcal{AEMP} uses to learn representations. Primarily, \mathcal{AEMP} operates an attention-enhanced message-passing scheme, which iteratively propagates the k -hop local neighbor information of a given entity over the neighborhood edges, paying attention in local (Figure 1.2 red-shaded areas, where light red is the local 1-hop neighbors and the dark red is the local 2-hop neighbors), global (Figure 1.2 green area, results from a complete scheme’s iteration), and random (Figure 1.2 blue area, randomly capture relationships) aspects of the neighborhood to learn a unique contextualized representation of the given entity and its neighborhood. Secondly, \mathcal{AEMP} identifies the semantic paths between a pair of entities and combines them into a single semantic path representation (Figure 1.2 yellow arrows). Finally, \mathcal{AEMP} combines

both contextualized entities and semantic path representations to inform the probability $P(r|h, t)$ of a new relationship between a pair of entities.

We conduct extensive evaluations of $\mathcal{A}EMP$ on four datasets extracted from the two main knowledge graphs benchmarks (WN18 [7], WN18RR [14], FB15k [7], and FB15k-237 [62]). We compare the efficiency of $\mathcal{A}EMP$ against some state-of-the-art models in relation prediction. Our experimental protocol considers the mean reciprocal rank (MRR), mean rank (MR), and hit ratio at k (Hit@ k) metrics. The metrics results show that $\mathcal{A}EMP$ outperforms the compared methods in three (WN18, WN18RR, and FB15k) out of the four datasets and demonstrates competitive results on the fourth benchmark. Furthermore, we conduct three ablation studies over $\mathcal{A}EMP$, in which we evaluate and discuss different aspects of the contextualized representations and how they influence the general performance of the model and the scalability up to millions of triples.

1.4 Contributions

We summarize the main contributions of this dissertation as follows:

- i) A novel attention-enhanced message-passing scheme for learning entities and their context joint representations;
- ii) Identification of semantic paths between pairs of entities and the combination of the paths' representations into a single general semantic path representation; and
- iii) Combination of attention-based entities representations with semantic paths representations for relation prediction.

1.5 Dissertation organization

Given the interdisciplinary nature of the relation prediction task and the associated techniques that inspired $\mathcal{A}EMP$, we recommend reading the text in full. Nonetheless, a guided description of each following chapter follows.

Chapter 2 contains all background knowledge, concepts, and notations that ground the content of this text. The chapter starts by introducing relational data and knowledge graphs, which are the objects of study of this work. Following, we briefly introduce representation learning on graphs and different aspects of learning representations of knowledge

graphs elements. Finally, the concept of attention mechanisms and their application on graphs is presented. We recommend the integral reading for readers that are not acquainted with various aspects of representation learning.

Chapter 3 summarizes the related literature. We, first, introduce symbolic approaches to the knowledge base completion challenges and some of their disadvantages over the distributional methods. The distributional methods are discussed in the following section, where we also approach learning representations from additional information within the knowledge graph. Then, works on attention mechanisms and their variations for graphs and knowledge graphs are examined. Subsequently, we discuss how the related literature relates with \mathcal{AEMP} .

\mathcal{AEMP} is proposed in Chapter 4. We start the chapter by proposing the attention-enhanced message-passing scheme for learning contextualized entities representations. Subsequently, we discuss the representation of semantic paths. Afterward, we introduce \mathcal{AEMP} 's training objective based on the probability distribution of relations regarding a pair of entities representations. Finally, we theorize some design alternatives. This chapter reading is essential for the general understanding of \mathcal{AEMP} , the main contribution of this work.

Chapter 5 holds all experimental evaluation, associated studies, and analysis. In this chapter, implementation details, such as hyperparameters configuration, resources used in the experiments, and compared literature results are also considered. Similar to Chapter 4, this chapter is fundamental for the overall understanding of \mathcal{AEMP} , and it shows the relevance of the proposed approach.

Finally, in Chapter 6 we conclude our work. Besides, in this chapter, we discuss limitations and propose future extensions of the work presented in this dissertation. We divide these extensions into twofold, where in the first part we suggest some immediate exploration, and in the second part, we propose three research questions for long-term studies.

Chapter 2

Background

This chapter presents concepts related to relational data and how they are represented in the form of triples and knowledge graphs (Section 2.1), as the input to the techniques proposed in Chapter 4. The Sections 2.2 and 2.3 cover representation learning and attention mechanisms, respectively, groundings of this dissertation’s central theme.

The notations used in this chapter and the rest of the dissertation and their description are presented in Table 2.1.

Table 2.1: Notations used in this dissertation.

Symbol	Description
h, t	Head and tail entities
r	Relation between a pair of entities
\mathbb{K}	Arbitrary mathematical domain
\mathbf{v}, \mathbf{M}	Vector and matrix
\mathbf{W}	Arbitrary weights matrix
σ	Non-linear transformation
ϕ	Score function
m_{θ}^i	Message of a KG’s element e at iteration i
α	Attention alignment score
\mathcal{N}_{θ}	Neighbor set of a KG’s element
\mathcal{C}_{θ}	Context set of a KG’s element
$\mathcal{P}_{(h,t)}$	Final semantic path representation of entities pair (h, t)

2.1 Relational data and knowledge graphs

Relational data covers the majority of the existing information. Any reference to a datum (also called entity, object, individual, among others) such as properties, types, and

relationships with other data configures a datum to be relational.

As expected, when it comes to structuring relational data, solutions have to be capable of modeling relationships, since relations are intrinsic to the information contained in the data. A simplistic and generic way of organizing relationships between elements is the conceptual structure called facts, *i.e.*, triples in the form $(head, relation, tail)$, in which *head* and *tail* are the entities (also called *subject* and *object*, respectively) and *relation* (also called *predicate*) is the relationship between both entities that qualifies the triple semantics. The following triples exemplify this structure:

$$\begin{aligned} & (BossaNova, typeOf, MusicGenre) \\ & (BossaNova, originallyFrom, Brazil), \end{aligned}$$

where *BossaNova* is the subject in both triples, *typeOf* and *originallyFrom* are relations, and *MusicGenre* and *Brazil* are the objects.

Definition 2.1 (Labeled Multidigraph) A labeled multidigraph is an ordered triplet $\mathcal{G} = (\mathcal{V}, \mathcal{A}, \mathcal{L})$, where \mathcal{V} is a set of vertices, \mathcal{A} is a ordered multiset of edges, *i.e.*, a multiset of ordered pairs of vertices \mathcal{V} , and \mathcal{L} is an ordered set of labels associated to each edge $e \in \mathcal{A}$.

Definition 2.2 (Knowledge Graph) A knowledge graph $\mathcal{KG} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$ is a labeled multidigraph $\mathcal{G} = (\mathcal{V}, \mathcal{A}, \mathcal{L})$, where $\mathcal{KG} \equiv \mathcal{G}$, $\mathcal{E} = \mathcal{V}$ is a set of vertices called entities, $\mathcal{R} \subseteq \mathcal{L}$ is a set of labels called relations, and $\mathcal{T} \equiv \mathcal{A} \circ \mathcal{L}$ is a set derived from the combination element-wise between edges and labels, called triples.

Another common representation of relational data are the *knowledge graphs* (KGs) (Definition 2.2), which are a semantic-based relational representation of data. In this representation, a datum is an *entity*, and the semantic relation between entities is a *fact*, or *triple*. The KGs are usually built upon the labeled multidigraph data structure (Definition 2.1). Graphs are data structures composed of nodes and edges. Edges connect a node to another node. Specifically, labeled multidigraph are graphs which all edges have an associated direction (directed graphs, digraphs) and an associated label (labeled digraphs). Also, these graphs' nodes might have multiple directed edges between them (multidigraph). In this structure, the vertices are the entities, and the edges express relationships between entities. Also, the edges' labels define the semantics of each relationship, thus, composing the facts.

An example of both representation formats (triples and graphs) is depicted in Figure 2.1. Figure 2.1.(a) illustrates the graph, with the entities represented as the graph's nodes and the relationships being the graph's labeled edges. Figure 2.1.(b) encodes the same information in the triple format.

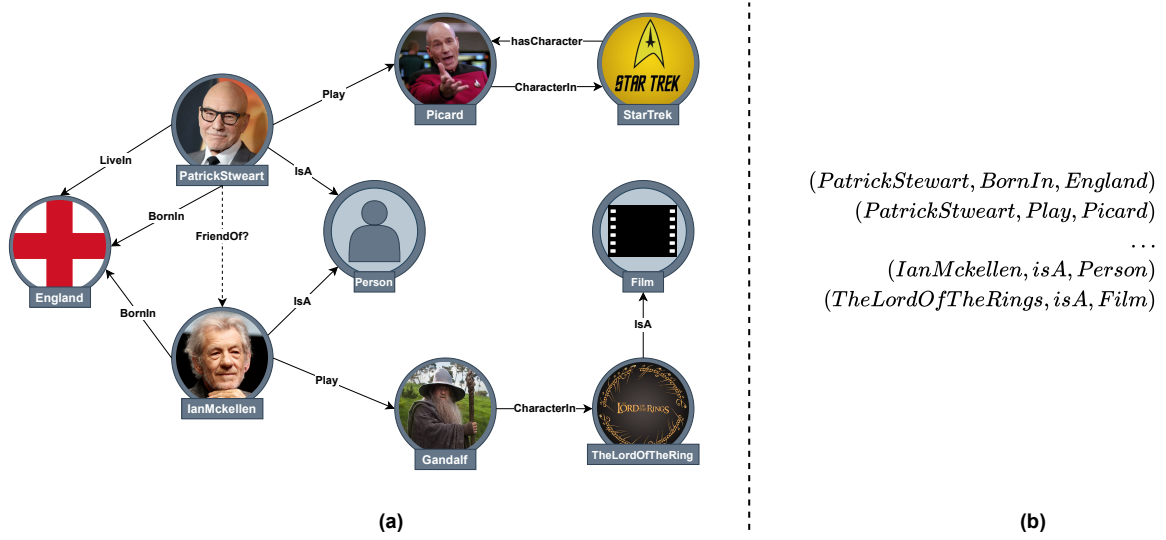


Figure 2.1: A knowledge graph about films and actors.

2.2 Learning knowledge graphs' representations

Neural networks are relaxed biology-inspired mathematical structures able to produce an output based on input signals inside an artificial neuron, vaguely similar to a brain's neuron, which produces a stimulus based on input pulses. The artificial neuron's output is the result of a non-linear transformation, *i.e.*, activation function [1], over the weighted sum of the input signal features with the addition of a bias, *i.e.*:

$$f(\mathbf{x}, \mathbf{W}) = \sigma\left(\sum_i^n w_{ki}x_i + b_i\right), \quad (2.1)$$

where $\mathbf{x} \in \mathbb{R}^d$ is the vector corresponding a input signal and its d features, $\mathbf{W} \in \mathbb{R}^{n \times d}$ is the weighting matrix, n is the number of input signal, \mathbf{b} is the bias vector, and σ is the activation function.

In this sense, a significant part of our mission towards building intelligent agents is to develop AI systems that are able to *understand* from the identification and generalization of hidden explanatory factors in the observed data. The general objective towards learning representations is to represent symbols in a mathematical space so that statistical models can manipulate them and infer new information based on representations

distribution [53]. Following the previously introduced notion of neural networks and artificial neurons, the *generalized learning* ability of neural networks come from the injection function of Equation 2.1 which maps the input set features to a high-level output representations¹. Learning representation allows models to be aware of a range of latent information, such as priors and context [5].

2.2.1 Facts embedding

Definition 2.3 (Knowledge graph representation) *Given a knowledge graph in the form $KG = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, where $\mathcal{E} \times \mathcal{R} \subset \mathcal{D} \times \mathcal{D}$ and \mathcal{D} is the symbolic domain of knowledge graph's elements, the embeddings of its elements, i.e., entities and relations in a mathematical space \mathbb{K} , are such, given a perfect representation function $\Gamma(\theta) : \mathcal{D} \rightarrow \mathbb{K}^d$ that perfectly maps an element $\theta \in \mathcal{E} \times \mathcal{R}$ from the knowledge graph's symbolic domain \mathcal{D} into a low dimensional embedding vector $e \in \mathbb{K}^d$ of dimension d .*

In the context of knowledge graphs, learning representations aims to provide distributive representations of the knowledge graphs' elements in a latent space. Thus, as stated in Definition 2.3, the elements are condensed into dense vector representations, also called *embeddings*, and relations are transformations within this space.

The Definition 2.3 introduces the concept of a *perfect* representation function, which is able to represent entities and relationships without loss. Such a perfect function is unlikely to be achievable. Like so, knowledge graph embedding models aim to learn a mapping function γ , where γ is an approximated function of Γ . These models' learning process intends to minimize a loss function based on a scoring function ϕ , which applies a transformation in the elements and measures how "close" the transformed element and the embedding are.

For instance, Figure 2.2 illustrates a toy version of TransE [7], a knowledge graph embedding model. In the figure, the knowledge graph is represented in a 3-dimensional Euclidean space, where the representations are obtained by optimizing the score function ϕ .

¹From now on in this dissertation we will use feature and representation as synonyms

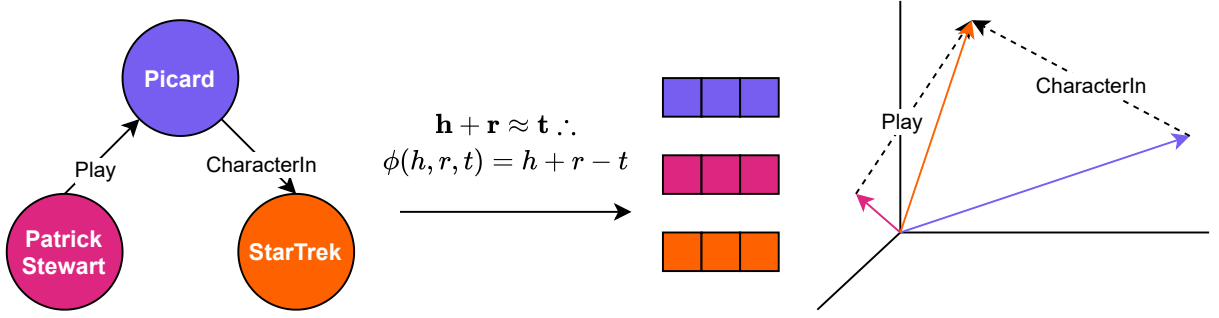


Figure 2.2: Example of the representations of a knowledge graph's entities and relationships in 3-dimensional euclidean space.

2.2.2 Additional information embedding

An alternative to providing more expressive embeddings is to incorporate additional information. This alternative allows models to be exposed not only to the facts information but also to more complex relationships, such as types [48], logical rules [15], long-term semantic relationships [40], local context [54], and text descriptions [70], among others. In the following sections, we discuss how additional information can be integrated. Following our motivation to build a symbolic-inspired distributional method, we will focus on two of them: the long-term semantic relationship, *i.e.*, the semantic paths, and the local context, *i.e.*, the entities information.

2.2.2.1 Semantic paths

Definition 2.4 (Semantic path) *Given a set of facts $\mathcal{F} = \{(h, r_1, n_1), (n_1, r_2, n_2), \dots, (n_k, r_k, t)\}$, a semantic path $h \xrightarrow{r_1, r_2, \dots, r_k} t$ is a sequence of relations through which two entities are connected and bounds a specific semantics.*

Definition 2.5 (Semantic path representation) *Given a semantic path $h \xrightarrow{r_1, r_2, \dots, r_k} t$, the set of relations' embeddings within the path $\mathbb{E}_{\mathcal{R}} = \{e_{r_1}, e_{r_2}, \dots, e_{r_k}\}$, and a composition function \oplus , the semantic path's representation is $\mathcal{P}_{(h,t)} = \bigoplus_{e_r \in \mathbb{E}_{\mathcal{R}}} e_r$.*

A semantic path defines a long-term semantic relationship between entities, as outlined in Definition 2.4. The ability to represent the long-term relationships enables models to be aware of prior semantics [15] and, many times, capture common-sense knowledge [8]. For instance, taking the example illustrated in Figure 2.3, once the relations between PatrickStewart and Picard, and Picard and StarTrek are explicit, is common-sense that PatrickStewart is an actor in StarTrek, similar to inferring over a non-explicit

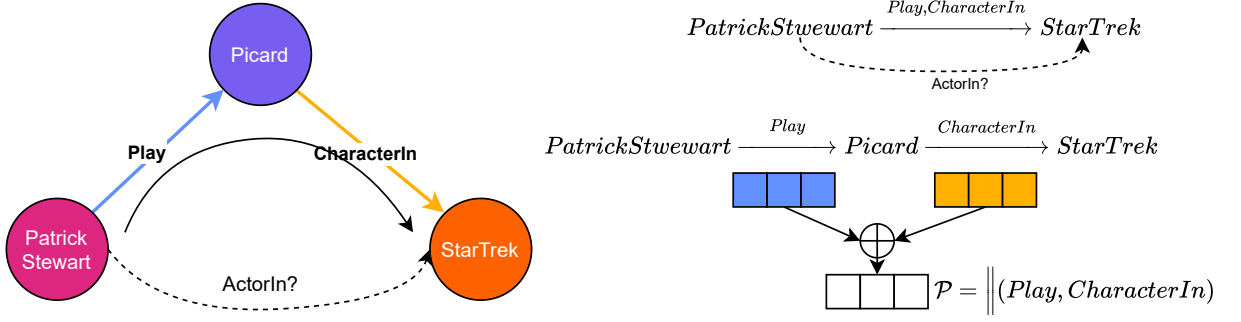


Figure 2.3: Example of the representations of a semantic path between entities entities.

rule. The process of representing a path is most usually straightforward: most existing approaches provide the path representation by composing the relations representations (Definition 2.5) aided by a composition function (*e.g.*, addition, multiplication, recurrent neural networks).

As an example, Figure 2.3 shows the path linking entities `PatrickStewart` and `StarTrek`. The path $PatrickStewart \xrightarrow{Play, CharacterIn} StarTrek$ suggests a semantically rich relationship between entities, in which we might even infer a new relation `ActorIn` in between. The representation of this path consist in applying a composition function over `Play` and `CharacterIn` representations.

2.2.2.2 Entities context

Definition 2.6 (Entity's context) *Given an entity e , the single-hop entity's context is the subgraph formed by all entities directly connected to e , i.e., all facts that involves e . A general k -hop entity's context is composed by the iterative aggregation (k iterations) of entities neighbors.*

Definition 2.7 (Entity's context representation) *Given the subgraph \mathcal{G}'_e of the k -hop entity's context, the embeddings' set of the elements within the subgraph $\mathbb{E}_{\mathcal{G}'_e} = \{e_\theta\} \forall \theta \in \mathcal{G}'_e$, and an aggregation function AGG , the entity's context representation is $AGG_{e_\theta \in \mathbb{E}_{\mathcal{G}'_e}}(e_\theta)$.*

Entities contextual information (Definition 2.6), or entities context for short, plays a fundamental part in describing the entity semantics. Thus, incorporating this information into the entities representation, at some level, leverages models to be cognizant of entities types and domain rules. The entities context representation (Definition 2.7) is characterized by an embeddings' aggregation of the context's elements.

2.2.2.3 Message passing framework

A special case of learning joint representations of an entity and its local context can be defined based in the *Message Passing Neural Networks* framework [22]. Also known as *message passing framework* [27], the framework proposes the use of an iterative framing to learn graphs' nodes representations by aggregating the node's local neighborhood and then updating the node's hidden representation. The iterative process of learning a hidden representation h_u of a node u can be express as:

$$\begin{aligned} h_u^{k+1} &= \text{UPDATE} (h_u^k, m_{\mathcal{C}_u}^k) \\ m_{\mathcal{C}_u}^k &= \text{AGGREGATE} (h_v^k, \forall v \in \mathcal{C}_u), \end{aligned} \quad (2.2)$$

where k is an iteration, UPDATE and AGGREGATE are arbitrary differentiable functions, and $m_{\mathcal{C}_u}^k$ is the message aggregated from the node's neighbors.

The framework, originally developed for an arbitrary graph, can be easily specialized to knowledge graphs. Like so, we define the same aforementioned operations regarding an arbitrary knowledge graph's element θ :

$$\begin{aligned} h_\theta^{k+1} &= \text{UPDATE} (h_\theta^k, m_{\mathcal{C}_\theta}^k) \\ m_{\mathcal{C}_\theta}^k &= \text{AGGREGATE} (h_v^k, \forall v \in \mathcal{C}_\theta). \end{aligned} \quad (2.3)$$

2.3 Attention mechanisms

The attention mechanisms are tools originally develop to aid neural machine translation models (MTMs) to handle long sentences [3, 9]. In short, the mechanisms provide a weighted alignment, also called *context* vector, between the hidden states' representations of an encoder² model. The context vector is then combined with the network's previous state as input to the decoder³. The key success factor of the attention mechanisms is that, through the context vector's aggregation, the decoder has access to some of the input's representation, which allows it to focus on the most relevant pieces of information, even in long sentences.

In the following, we discuss the general attention mechanism and their versioning for graphs and knowledge graphs.

²Encoder: the component that is responsible for learning a dense representation from the input, *i.e.*, the input's encoding.

³Decoder: the component that is responsible for learning a mapping between the encoder's dense representation from the input to a target object.

2.3.1 General attention

Definition 2.8 (Attention mechanism – adapted from [9]) *Given a set of key-value pairs (K, V) , a query q , and p a distribution function (e.g. softmax), where K is the set of keys, i.e., the encoder hidden states, V is the set of values, i.e., the values to which the attention is applied, and the query q is the decoder hidden state, the attention mechanism is such $A(q, K, V) = \sum_i p(a(k_i, q)) * v_i$, where a is an alignment function and the attention weights are $\alpha_{ij} = p(a(k_i, q))$.*

Formally, as stated in Definition 2.8 [9], the attention mechanisms learn a distribution over the input keys regarding an input query. The hidden states are captured and used as *keys* under an alignment function from the encoder. Aside, the query, i.e., the last state of the decoder is also used in the alignment function. Further, a distribution function (usually a softmax function, notwithstanding, alternative functions can be used [19]) is applied over the alignment function resulting in the attention weights. A significant advantage of this formulation is that if both alignment function and probability distribution function are differentiable, the attention weights can be learned in a single feed-forward neural layer.

From the general formulation, the concepts of local and global attention [43] can be derivated. The global attention learns based on the keys set being all hidden states of the encoder. Its advantage is to offer a smooth distribution over a differentiable setting. In contrast, the local attention learns based on a subset of the encoder’s hidden states. A subset allows local attention to be attentive to specific aspects of the keys, but it demands more complex learning techniques since it is no longer differentiable.

2.4 Graph attention

Definition 2.9 (Graph attention mechanism – adapted from [36]) *Given a graph’s element θ (e.g., node, edge, and subgraph), and the set of elements in the neighborhood of θ , \mathcal{N}_θ , the graph attention mechanism $\zeta : \{\theta\} \times \mathcal{N}_\theta \rightarrow \mathbb{R}$ is a mapping function that assigns to a neighbor element a relevance score, in which $\sum_{i \in \mathcal{N}_\theta} \zeta(\theta, i) = 1$.*

Similar to attention mechanisms in Natural Language Processing (NLP) tasks, attention mechanisms in graphs (Definition 2.9) enable methods to focus on different aspects of a graph’s element neighborhood [36]. The main idea of the attention mechanisms in

graphs is to compute the element's hidden representation by attending over its neighborhood.

Formally, following the definition of the graph attention networks (GAT) introduced by Veličković *et al.* [66], the graph attention layer produces an output set of attended representations $h' = \{h'_1, h'_2, \dots, h'_k\}$, $h'_i \in \mathbb{R}^{d'}$, based on its input set $h = \{h_1, h_2, \dots, h_k\}$, $h_i \in \mathbb{R}^d$, where h_i is an element's feature vector, k is the number of elements in the sets, and d and d' are the dimensions of each feature vector.

To that end, the layer defines an attention mechanism $a : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ that is able to compute the *attention coefficient*

$$e_{ij} = a(\mathbf{W}h_i, \mathbf{W}h_j), \quad (2.4)$$

based on a transformation higher-level mapping $\mathbf{W} \in \mathbb{R}^{(d' \times d)}$ of the input features. The coefficient is able to indicate the *relevance* of a neighbor's features h_j regarding the element's representation h_i . To leverage coefficients comparability, they are, then, normalized with a softmax function:

$$\alpha_{ij} = \text{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})}. \quad (2.5)$$

The final attended output representation is a linear combination between the normalized attention coefficients with their input corresponded potentially transformed by a non-linear function σ :

$$h'_i = \sigma \left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W}h_j \right). \quad (2.6)$$

2.4.1 Knowledge graph attention mechanisms

The previous section introduces a general definition of graph attention mechanisms for any of the graph's elements. However, the GAT formulation focus on the graph's nodes, not being able to handle mixed graphs' elements (*e.g.*, nodes and edges, nodes and subgraphs). Moreover, as described in Section 2.1, relations play a fundamental part in the entities role assignment. Thus, the graph attention mechanisms are not capable to properly operate knowledge graphs, which by its nature requires methods efficient on handling mixed graph's elements, *i.e.*, entities, and relations. In contrast, the solution is to extend GATs to an approach that manages entities and relations together.

To formulate the notion of attention mechanisms specialized in knowledge graphs extending the Definition 2.9, we base ourselves in the definition proposed by Nathani,

Chauhan, Sharma *et al.* [51]. The knowledge graph attention layer outputs two sets of attended representations $h' = \{h'_1, h'_2, \dots, h'_{k_1}\}$, $h'_i \in \mathbb{R}^{d'_1}$ for entities, and $h' = \{g'_1, g'_2, \dots, g'_{k_2}\}$, $g'_i \in \mathbb{R}^{d'_2}$ for relations, based on their input set $h = \{h_1, h_2, \dots, h_{k_1}\}$, $h_i \in \mathbb{R}^{d_1}$ and $g = \{g_1, g_2, \dots, g_{k_2}\}$, $g_i \in \mathbb{R}^{d_2}$, respectively, where h_i is an entity's feature vector, g_i is a relation's feature vector, k_1 and k_2 are the number of elements in the entities and relations sets, respectively, and d_1, d_2 and d'_1, d'_2 are the input and output dimensions of each from entity and relation feature vector, respectively.

To compute the attention coefficient on knowledge graphs' elements, the attention mechanism $a : \mathbb{R}^{d'_1} \times \mathbb{R}^{d'_1} \times \mathbb{R}^{d'_2} \rightarrow \mathbb{R}$ is such:

$$e_{ijk} = a(\mathbf{W}_a h_i, \mathbf{W}_a h_j, \mathbf{W}_b g_k), \quad (2.7)$$

where $\mathbf{W}_a \in \mathbb{R}^{d'_1 \times d_1}$ and $\mathbf{W}_b \in \mathbb{R}^{d'_2 \times d_2}$ are higher-level transformation for entities and relation feature vectors, respectively. Similar in GAT, the attention coefficient carries the relevance of an triple $t_{ijk} = (h_i, g_k, h_j)$ among others. The coefficients of a triple t_{ijk} are then normalized regarding the entity i context \mathcal{N}_i and the set of pair relation-entity \mathcal{N}_{in} connected to it:

$$\alpha_{ijk} = softmax_{jk}(e_{ijk}) = \frac{\exp(e_{ijk})}{\sum_{n \in \mathcal{N}_i} \sum_{r \in \mathcal{N}_{in}} \exp(e_{inr})}. \quad (2.8)$$

Finally, the output representation of an entity h_i , is a result of the combination of each triple representation weighted by their attention coefficients under a non-linear transformation σ :

$$h'_i = \sigma \left(\sum_{j \in \mathcal{N}_i} \sum_{k \in \mathcal{N}_{ij}} \alpha_{ijk} e_{ijk} \right). \quad (2.9)$$

2.5 Discussion

This Chapter covers the grounding knowledge, definitions, and notations for the technique proposed in Chapter 4. It addresses relational data representation, specifically knowledge graphs, and machine learning methods such as representation learning and attention mechanisms for knowledge graphs.

Section 2.1 addresses the representation of relational data and their structuring in triples and knowledge graphs. Sections 2.2 and 2.3 covers the machine learning groundings for representation learning and attention mechanisms. The way in which these concepts are applied to our proposed solution is explained in Chapter 4.

Chapter 3

Related Work

This chapter presents previous literature on knowledge base completion and knowledge graph embedding. Some works presented here will be used as grounding for ÆMP in Chapter 4 and baseline in the experimental results presented in Chapter 5. We explore techniques that attempt to complete knowledge bases based in symbolic statistical relational learning (Section 3.1) followed by methods aligned with the scope of this dissertation that learns representations over knowledge graphs (Section 3.2). Then, we show some works on attention mechanisms over graphs (Section 3.3). Finally, in Section 3.4, we discuss the differences between the related literature with the technique proposed in this dissertation (Chapter 4).

3.1 Symbolic statistical relational learning

Initial attempts on prediction new relationship among entities in knowledge bases (KBs) uses Logic Programmings [21] and Inductive Logic Programming (ILP) [50] by mining logical rules from these KBs. For instance, the general-purpose ILP system ALEPH¹ [49] applies search strategies combined with evaluation functions to mine predicate logical rules that describe concepts in a domain. Such rules can be used as a mechanism to infer new relationships among entities by employing logical reasoning. AMIE+ [18] is another relevant method to mine logical rules from KBs that is grounded in the *Open World Assumption* [17]. The main difference between AMIE+ and most of the ILP-based methods is that AMIE+ is able to mine rules despite not having explicit counterexamples, leveraging the principles of Partial Completeness Assumption [17] to automatically infer counterexamples for rules.

¹<https://www.cs.ox.ac.uk/activities/programinduction/Aleph/aleph.html>

The above-mentioned approaches focus on mining logical rules, making them unable to reason over noisy information and uncertainty. To overcome this issue, the combination of graphical models with logic and relational approaches [21] are another set of techniques that generally aims to represent relational data under uncertainty and probabilistic theory. Focusing on the KB completion challenge, methods leverage by this combination, such as the Markov Logic Network [59], represent the joint distribution over the entities and their existing associated facts to predict new relationships through probabilistic inference.

Symbolic statistical relational learning methods aim at reasoning over uncertainty under a hypothesis-logical space. Their limitation goes towards modeling capacities, *i.e.*, they are usually limited to model data based in a predefined probabilistic model, and scalability, given the complexity of hypothesis-logical space used by these approaches, they are unable to leverage differentiable capabilities [11]. In the next section, we discuss differentiable approaches to relational learning. Specifically, we focus on knowledge graph embeddings, which are the scope of this dissertation.

3.2 Knowledge graph representation

Most current methods for learning KG’s entities and relations representations model each fact individually, providing an embedding vector for each KG element in a continuous vector space. Usually, the embeddings of each entity are learned based only on embeddings of their direct neighbors [7, 30, 61, 64, 73]. TransE [7], ComplEx [64], RotatE [61], and QuatE [73] are examples of these methods; they rely mainly on the representation of entities on vector spaces (such as, Euclidean space, and complex spaces) and on algebraically operating this representations to achieve their tasks. These method ignores that facts are part of a much richer structure (the knowledge graph itself).

In short, TransE represents entities and relations in a Euclidean vector, learning the embedding from the translation of and entity regarding a relation. ComplEx uses the complex space to represent entities and relations, aiming to better capture symmetric and antisymmetric relationships. SimpleE proposes enhancing Canonical Polyadic (CP) [28] allowing the two embeddings of each entity to be learned dependently while leading the model to have a simple representation. Similar to ComplEx, RotatE uses the complex space proposing a rotational model, where the relations are modeled as rotations from the entities. QuatE uses quaternion inner product as a compositional operator for the representation of relations and entities in a hyper-complex space.

In the following, we revisit methods that start from the same motivation as ours: entities are likely to be better represented when contextual information is incorporated into the learning process; consequently, one may reach better predictive results to complete the KGs. These methodologies can be divided into two categories: entities context, where local relationship patterns and neighborhood information among entities are observed, and relational paths, where the paths between entities and their semantics are considered.

3.2.1 Entities context

Luo *et al.* [42] pioneered the generation of context-dependent entity embeddings. Their goal is to learn 1-hop local neighborhood representations for entities, called contextual connectivity patterns, and, based on previous learned embeddings, refer to local connectivity patterns, fine-tune them with the connectivity patterns. Oh *et al.* [54] proposes context-aware embeddings by jointly learning from an entity and its multi-hop neighborhood. Both approaches only adopt as context the representations of the entities neighbors, which limits their capacity to generalize the entity’s local structure once the semantics of the relationships between neighbors are neglected by the models.

In contrast, Wang *et al.* [67] urge the necessity to address entities local context regarding their relationships and, consequently, their local structure. Like so, they design a message-passing scheme to learn entities k -hop neighborhood based on the aggregation of graph’s edges, *i.e.*, the relationships. However, the proposed method equally weights the neighbors within the aggregation process.

We argue that the message-passing scheme holds the potential to acquire entities local structures. Thus, we benefit from the message-passing scheme in our model and propose an enhancement to avoid handling equally all neighbors. We adopt an attention mechanism to focus on different aspects of the passed messages, weighting the importance of each fact in the neighborhood for the final entity representation.

3.2.2 Relational paths

Several previous works count with sequences of relations among entities to add contextual information to KG’s embeddings [13, 25, 26, 39, 40]. PTransE [40] extends the use of translation-based embeddings mechanisms [7] to find the relationship between two entities by including multiple-step relation paths between them. Gu *et al.* [26] proposes additive and multiplicative compositions over relations while [52] leverages recurrent neural net-

works to consider relational paths of entities. Both of those strategies model a single path between two entities. Aiming at complex reasoning to populate KBs from texts, [63] and [13] incorporate multiple paths that are built not only upon relations but also with other entities. More recently, Guo *et al.* [25] also targeted at learning from relational paths but using a recurrent neural network with residual connections. By allowing skipping connections, entities in a path can contribute to predicting not only a possible link but can also semantically enhance the objects entities in a path.

In this dissertation, we also build paths motivated by the semantic enrichment that one entity may provide to the others. However, we focus on message-passing schemes leveraged by different patterns of attention to let the model find out during the learning which elements are more relevant to the embeddings and, consequently, the relation prediction task.

3.3 Attention mechanisms

As defined in Chapter 2, attention mechanisms enhance models to be aware of a general context by learning from the encoder’s hidden states [3]. Luong *et al.* [43] proposed the use of local and global attentions on sentences. Both are variations of the original attention aiming to predict an alignment position for the current target and a target-centered window to compute the context vector, respectively.

BigBird [72] is another relevant attention mechanism in NLP that takes inspiration from graph sparsification methods. The BigBird is an attention mechanism that overcomes the Transformers [65], a set of attention mechanisms that are currently considered the state-of-the-art in several language-related tasks by proposing a *sparse attention mechanism*. This new attention mechanism leverages the combined use of random, windowed (local), and global attention to be robust and expressive as the Transformers but using fewer resources.

In the following, we explore attention mechanisms over graphs. We differ from all techniques by leveraging graph-based attention to capture different contextual patterns over elements from the knowledge graph, taking inspiration from local, global, and sparse attention mechanisms.

Table 3.1: Comparison between literature approaches, their used mechanisms, and tasks in which they were previously evaluated.

Model	Mechanisms			Task evaluated	
	Message-passing scheme	Semantic Paths	Attention	Link Prediction	Relation Prediction
TransE [7]				X	X
PTransE [40]		X		X	X
ComplEx [64]				X	X
SimplE [30]				X	X
RotatE [61]				X	X
QuatE [73]				X	X
Nathani <i>et al.</i> [51]	X		X	X	
PathCon [67]	X		X		X

3.3.1 Attention-based embeddings for general graphs

One of the first approaches to employ the concept of attention to learn representations from graph-based data was GAT [66]. However, that work was focused on learning representations using attention weights computed over every other node (in the most general formulation). Moreover, they target the node classification task, neglecting labeled multi-digraphs. Here, besides allowing different views of attention over the nodes and edges, we aim at predicting relations in *knowledge* graphs. GAT, on the other hand, handles graphs that are not designed to take into account the relations, which are not only responsible for connecting entities but also to modulate their roles.

3.3.2 Attention-based embeddings for knowledge graphs

The first attempt to use attention-based mechanisms to learn how to predict relations in KGs was presented in [51]. To solve the issue of capturing the contributions of distant entities, they relied on a relation composition mechanism that introduces auxiliary edges between neighbors in hops. We, on the other hand, allow for local and global attention mechanisms to potentially focus on close and distant neighbors during the message-passing mechanism. Moreover, we include a random attention mechanism that potentially learns when to attend to close and distant neighbors.

3.4 Discussions

This chapter presents works that, at some level, are related to \mathcal{AEMP} , formally proposed in Chapter 4. We start the chapter by introducing some traditional approaches from statistical relation learning to the knowledge base completion task. Later, we explore

techniques of knowledge graphs representation learning focused on learning over facts only. Additionally, how discuss the aggregation of the additional information to the representations. Further, we explore attention mechanisms that inspire $\mathcal{A}EMP$.

Table 3.1 compiles the most related models, their capabilities, and tasks in which they were evaluated. The baseline methods, used in Chapter 5, were those evaluated in the context of the *relation prediction* task. Besides, the content of this chapter is helpful to have an overview of different aspects of the literature and is important for the complete understating of the inspirations considered in Chapter 4.

Chapter 4

Learning Attention-based Representations from Multiple Patterns

In this chapter, we propose and develop¹ **ÆMP** (**A**ttention-based **E**mbeddings from **M**ultiple **P**atterns), illustrated in Figure 4.1, that learns contextualized representations for relation prediction on knowledge graphs. **ÆMP** grounds its architecture in three main components: (i) an attention-enhanced message-passing scheme for learning the joint representation of entities and their contexts; (ii) a method to capture and represent semantic paths as context information in the learning process; and (iii) the combination of both contextual information to the final contextualized representation.

This chapter is composed of Section 4.1, where we describe the attention-enhanced message-passing scheme for learning the joint representation of entities and their contexts. In this section, we present the message-passing scheme (Figure 4.1.(a)), and propose the employment of local (Figure 4.1.(b)), global (Figure 4.1.(c)), and random (Figure 4.1.(d)) attention mechanisms. Section 4.2 describes how we capture the semantic paths (Figure 4.1.(e)) and how they are contemplated as context information into the learning process. The **ÆMP**'s training objective is discussed in Section 4.3, where we also demonstrate how we combined the context information.

The notations used in this chapter and the rest of the dissertation and their description are presented in Table 4.1.

¹<https://github.com/MeLL-UFF/AEMP>

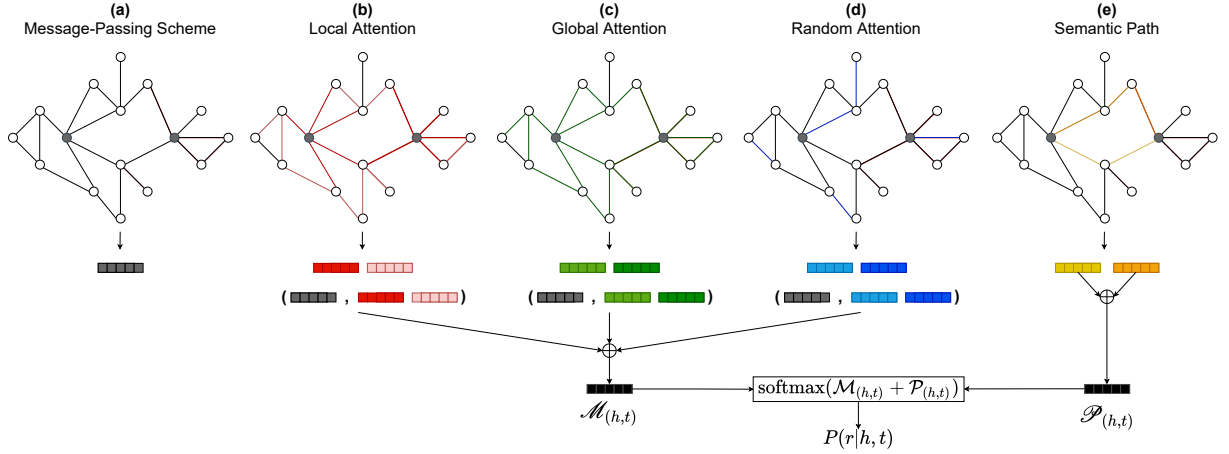


Figure 4.1: Overview of ÆMP architecture. Boxes represent the vector representations produced in each step. Gray-shaded illustrates the message-passing scheme. Red-shaded points out each hop used on local attention. Green-shaded expresses the iterations used on global attention. Blue-shaded indicates the randomly selected relationships used on random attention. Yellow-shaded indicates the semantic paths.

4.1 Learning entities context representations

Inspired by the advances in attention mechanisms over graphs (GATs) [66], as well as their sparse versions in NLP [44, 72], and the message-passing scheme for aggregating entities neighborhood representations [67], we propose a novel mechanism for capturing and learning entities context (Algorithm 1). Our mechanism uses the message-passing scheme to interactively learn entities representations from the propagation of their multi-hop neighbor edges representations. After learning such representations, as a second step in the learning process, we submit them to an attention layer that combines local, global, and random attention mechanisms. Such a combination of different views benefits the model to learn the entities context while focusing on different neighborhood aspects.

Following the example illustrated in Figure 4.2, local attention reinforces local connective patterns, providing a narrow view of the entities and their surroundings (*e.g.*, the red-shaded areas in the figure; the relationship between entities **PatrickStewart** and **England**). Global attention reinforces global connective patterns, broadly focusing on the entities neighborhood (*e.g.*, the green-shaded area in the figure; the relationship between entities **England** and **XMen**). Random attention assists in capturing non-directed patterns (*e.g.*, the blue-shaded area in the figure; the relationship between entities **IanMcKellen** and **Magneton**). After considering different contextual patterns, ÆMP is able to infer new relationships such as the one between **PatrickStewart** and **IanMcKellen**, for example.

Table 4.1: Notations used in this dissertation.

Symbol	Description
m_e^i	Message of entity e at iteration i
s_r^i	Representation of a relationship r at iteration i
$\mathcal{N}(e)$	Incident relationships of an entity e
$\mathcal{N}(r)$	Incident entities to a relationship r
λ	Attention mechanism (local, global, or random)
α_{rk}^λ	Attention alignment score regarding a relation r and an iteration k
$s_r^{\lambda^k}$	Attention-enhanced relationship representation
\mathcal{C}_R	Relationships' context set
\mathcal{C}_I	Iterations' context set
m_e^{ATT}	Final representation of an entity e
$\mathcal{M}_{(h,t)}$	Final entities context representation of entities pair (h, t)
$\mathcal{P}_{(h,t)}$	Final semantic path representation of entities pair (h, t)

4.1.1 Message passing scheme

Equations 4.1 and 4.2 and Figure 4.1.(a) formalizes and describes, respectively, the message-passing scheme for entities representation learning. We define $m_e^i \in \mathbb{R}^d$ as the message, *i.e.*, the contextual representation of an entity e , and s_r^i as the representation of a relationship r between a pair of entities, both computed in an iteration i .

$$m_e^i = \sum_{r \in \mathcal{N}(e)} s_r^i, \quad (4.1)$$

$$s_r^{i+1} = \sigma \left(\text{flatten} \left(m_h^i m_t^{iT} \right) \mathbf{W}_1 + s_r^i \mathbf{W}_2 + b^i \right),$$

$$h, t \in \mathcal{N}(r), m_h^i m_t^i = \begin{bmatrix} m_h^{i(1)} m_t^{i(1)} & \dots & m_h^{i(1)} m_t^{i(d)} \\ & \ddots & \\ m_h^{i(d)} m_t^{i(1)} & \dots & m_h^{i(d)} m_t^{i(d)} \end{bmatrix}, \quad (4.2)$$

The entity's message $m_e^i \in \mathbb{R}^d$ is the sum of all incident relationship representations $\mathcal{N}(e)$ of an entity e (Equation 4.1). The relationships' representations are updated iteratively according to Equation 4.2, where the next relation state s_r^{i+1} is updated based on a cross-neighbor aggregator operation [67]. The aggregator models the entities cross matrix, a pairwise product of the entities representations $m_h^{i(d_k)} m_t^{i(d_k)}$, where m_h^i and m_t^i are the head and tail representations, respectively, (d_k) is a dimension in d . The updated relationship representation s_r^{i+1} is a combination of the flattened entities cross matrix with the last relationship representation s_r^i , both linear transformed with weights \mathbf{W}_1 and \mathbf{W}_2 ,

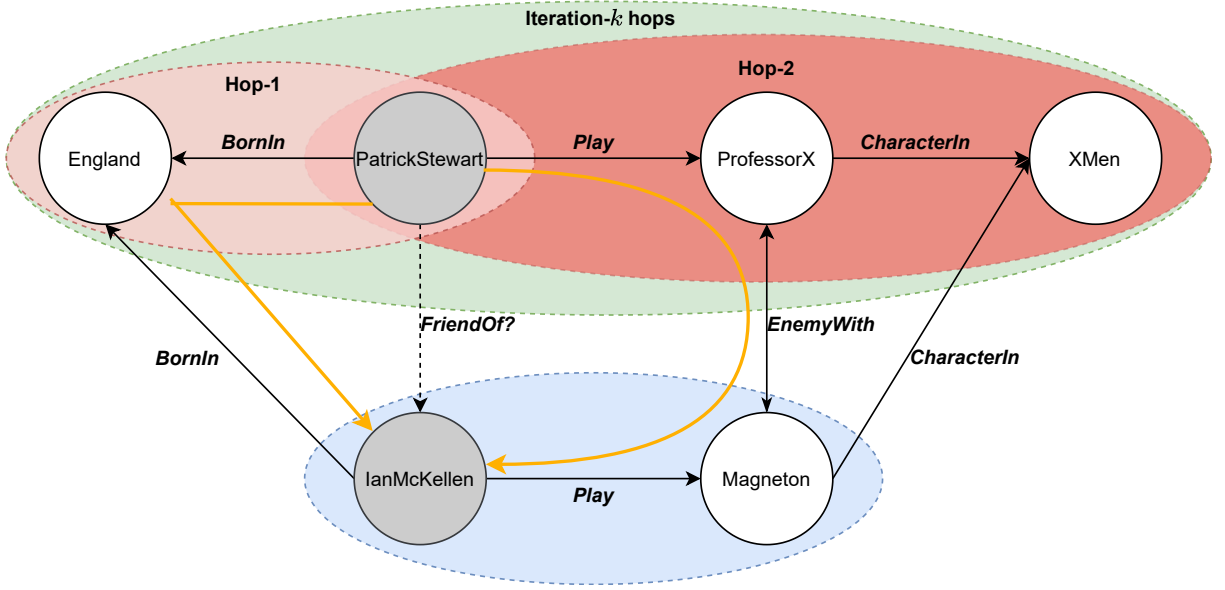


Figure 4.2: Example of knowledge graph under the ÆMP highlighting ÆMP’s mechanisms. Similar to the knowledge graph illustrated in Figure 1.2, gray circles indicate the head and tail entities. Dashed-arrows are potentially missing relations. Yellow-shaded arrows indicate the semantic paths between the head and tail. Green, red, and blue-shaded areas represent global, local, and random attention mechanisms, respectively, used in ÆMP.

followed by a final nonlinear transformation σ .

4.1.2 Attention-enhanced message passing

Built on top of the message-passing scheme, we define a multi-context attention layer motivated by both GATs [66] and the sparse attention mechanism [44, 72]. We extend the GATs’ attention mechanism by adopting local and global attention, inspired by [43]. Moreover, we include random attention within the message-passing scheme iterations to leverage sparsely arranged possible relationships.

Illustrated in Figure 4.1.(b) and described in Equations 4.3 and 4.4, the local attention apparatus employs the relationships’ messages acquired in *each hop* \mathcal{C}_R from a particular iteration k as the context of the resultant message s^k in the iteration. It allows for the querying message to be informed of its neighborhood structural information, *i.e.*, its local context structure.

$$\alpha_{rk}^{local} = \text{align}_r(s^k, s_r^k) = \frac{\exp(s^{k\top} \mathbf{W} s_r^k)}{\sum_{r' \in \mathcal{C}_R} \exp(s^{k\top} \mathbf{W} s_{r'}^k)} \quad (4.3)$$

$$s_r^{local^k} = \sigma \left(\sum_{r \in \mathcal{C}_R} \alpha_{rk}^{local} \mathbf{W} s_r^k \right) \quad (4.4)$$

The global attention (Figure 4.1.(c), Equations 4.5 and 4.6) operates over the final entities messages of *each iteration* \mathcal{C}_I . By doing so, the global semantic of each iteration is captured and informed to the querying message.

$$\alpha_{rk}^{global} = \text{align}_k(s_r, s_r^k) = \frac{\exp(s_r^\top \mathbf{W} s_r^k)}{\sum_{i \in \mathcal{C}_I} \exp(s_r^\top \mathbf{W} s_r^i)} \quad (4.5)$$

$$s_r^{global^k} = \sigma \left(\sum_{i \in \mathcal{C}_I} \alpha_{ri}^{global} \mathbf{W} s_r^i \right) \quad (4.6)$$

We highlight the difference between contexts from equations 4.4 and 4.6, where \mathcal{C}_R is intrinsic related to the messages acquired within a hop and \mathcal{C}_I is composed of all messages from an iteration.

Finally, the random attention illustrated in Figure 4.1.(d) and described in Equations 4.7 and 4.8 randomly captures, according to predefined probability called *context selection criteria*, further aspects of the entities neighborhood. Such aspects might get neglected by the local attention based on each hop or the global attention based on a whole iteration update. In this way, the random attention weights are built upon both the hops and iterations, *i.e.*, using each randomly capture specific relationship representation $s_{r'}^i$ of a iteration i within a hop r' . The random mechanism allows the querying message to pay attention to diverse aspects of its neighborhood, introducing an indiscriminate bias.

$$\alpha_{rk}^{random} = \text{align}_{rk}(s, s_r^k) = \frac{\exp(s^\top \mathbf{W} s_r^k)}{\sum_{\{i,r'\} \in \mathcal{C}_R} \exp(s^\top \mathbf{W} s_{r'}^i)} \quad (4.7)$$

$$s_r^{random^k} = \sigma \left(\sum_{\{i,r'\} \in \mathcal{C}_R} \alpha_{ri}^{random} \mathbf{W} s_{r'}^i \right) \quad (4.8)$$

The final representation of an entity and its local, global, and random contexts regarding Equation 4.1 are the last aggregation operation of the message-passing scheme, where the final message of an arbitrary entity e is m_e^K , where K is the last iteration. Here, we use m_e^{local} , m_e^{global} , and m_e^{random} being the resulting representation of the attention-enhanced message-passing scheme on each attention pattern, and m_e^{ATT} being the concatenation of

Algorithm 1: Attention-enhanced Message-Passing Scheme

```

1  $i \leftarrow 1$ 
2  $S_i \leftarrow S_0$ 
3  $S^{random} \leftarrow \{\}$ 
4  $S^{global} \leftarrow S_0$ 
5  $S^{local} \leftarrow S_0$ 
6 while  $i \neq K$  do
7    $S_{i+1} \leftarrow \{\}$ 
8    $hop \leftarrow 1$ 
9   while  $hop \neq H$  do
10     $M_i \leftarrow \text{Equation 4.1 } (S_i)$ 
11     $s_r^{i+1} \leftarrow \text{Equation 4.2 } (M_i)$ 
12     $S_{i+1} \leftarrow S_{i+1} \cup s_r^{i+1}$ 
13     $S^{local} \leftarrow S^{local} \cup s_r^{i+1}$ 
14    if with a probability  $p$  then
15       $S^{random} \leftarrow S^{random} \cup s_r^{i+1}$ 
16     $hop \leftarrow hop + 1$ 
17  end
18   $S^{local} \leftarrow \text{Equation 4.4 } (S_{i+1}, S^{local})$ 
19   $S^{global} \leftarrow S^{global} \cup S_{i+1}$ 
20   $i \leftarrow i + 1$ 
21 end
22  $M^{local} \leftarrow S^{local}$ 
23  $M^{global} \leftarrow \text{Equation 4.6 } (S_K, S^{global})$ 
24  $M^{random} \leftarrow \text{Equation 4.8 } (S_K, S^{random})$ 
25  $M^{ATT} \leftarrow M^{local} \oplus M^{global} \oplus M^{random}$ 
26 return  $M^{ATT}$ 

```

the outcome representation of each attention mechanism:

$$m_e^{ATT} = m_e^{local} \oplus m_e^{global} \oplus m_e^{random}, \quad (4.9)$$

where m_e^{local} , m_e^{global} , and m_e^{random} are the representations of the entity and its local context, global, and random contexts, respectively.

Finally, to provide the final embedding of the head and tail pair $\mathcal{M}_{(h,t)}$, we combine both head and tail final messages, as shown in the following equation:

$$\mathcal{M}_{(h,t)} = m_h^{ATT} \oplus m_t^{ATT}, \quad (4.10)$$

where m_h^{ATT} and m_t^{ATT} are the final representation messages from entities head and tail, respectively.

The Algorithm 1 details the process followed by the attention-enhanced message pass-

ing scheme. The algorithm is initialized in the first iteration with an initial state (*e.g.*, random features, Xavier initialization [23], bag-based features [45]) (Algorithm 1 – Lines 1-5). The message passing scheme (Algorithm 1 – Lines 6-20) iteratively updates the hidden state S_i by aggregating entities neighbors (Algorithm 1 – Lines 9-17). During the algorithm’s loop phase, the random (Algorithm 1 – Line 15), the local (Algorithm 1 – Line 18), and the global (Algorithm 1 – Line 19) contextual information are gathered. Further, the attention coefficients are computed based on the previously gathered contexts (Algorithm 1 – Lines 22-24). Finally, the entity’s final representation is provided by the contexts’ concatenation (Algorithm 1 – Line 25).

4.2 Learning semantic paths representations

Our aforementioned attention-based approach provides essential information for learning entities local-expanded context. However, it is incapable, given the limit in the number of hops, of capturing long semantic paths, *i.e.*, long sequences of relations between entities. As an example, the semantic path (Figure 4.2 yellow-shaded arrows) between **PatrickStewart** and **IanMcKellen** entities is overlooked, since it is longer than the entities context hops. To address this issue, we propose considering the semantic paths between entities within the learning process of \mathcal{AEMP} .

To do so, as described in Algorithm 2 and illustrated in Figure 4.3, we, first, identify $P(h, t) = \{\mathbf{p}_1, \dots, \mathbf{p}_m\}$, which are all the semantic paths between two entities (Algorithm 2 – Line 1), where h and t are the head and tail entities and $\mathbf{p}_i = (r_1, \dots, r_n)$ is the semantic path $h \xrightarrow{r_1, \dots, r_n} t$ between them, and r_i is a relation within the path. In the algorithm we use the breadth-first search algorithm [10] with maximum path length as constrain to find all paths between the entities. After, we provide one-hot representations to the identified paths (Algorithm 2 – Lines 2-4) and, similar to [40], in order to learn a single representation of the semantic paths, we perform a linear transformation (Algorithm 2 – Line 6) over the concatenation of the paths’ representations (Algorithm 2 – Line 4), mapping the concatenated representation to the same dimensional space than $\mathcal{M}_{(h,t)}$. Equation 4.11 shows this process:

$$\mathcal{P}_{(h,t)} = \mathbf{W} \left\|_{\mathbf{p} \in P(h,t)} \mathbf{p}, \quad (4.11)$$

where \mathbf{W} denotes the linear transformation matrix, and $\|$ denotes concatenation operation.

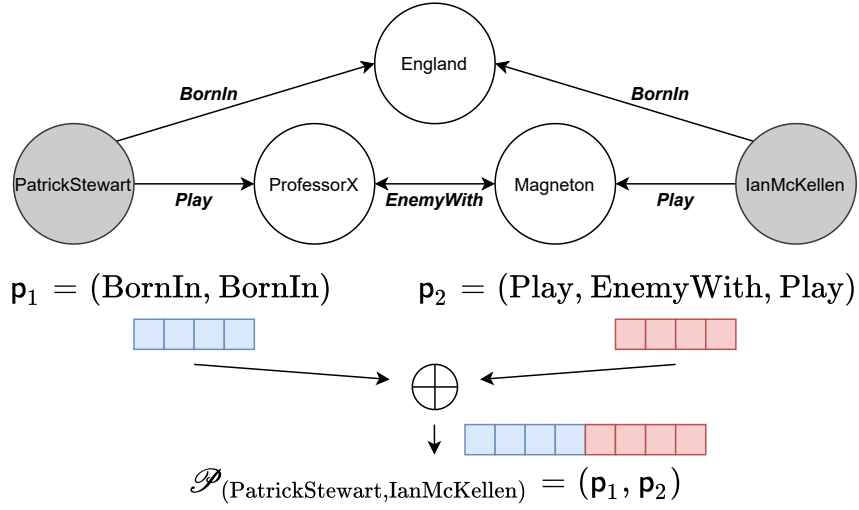


Figure 4.3: Example of the representation of the semantic paths between head and tail entities, *i.e.*, `PatrickStewart` and `IanMcKellen` entities, respectively.

Algorithm 2: Learning semantic paths representation

```

1  $\mathcal{P}(h, t) \leftarrow \text{BreadthFirstSearch}(h, t, \text{max\_length})$ 
2  $\mathcal{P}_{(h, t)} \leftarrow \text{OneHotEncoding}(\mathbf{p}')$ ;  $\mathbf{p}' \in \mathcal{P}(h, t)$ 
3 foreach  $\mathbf{p} \in \mathcal{P}(h, t) - \{\mathbf{p}'\}$  do
4    $\mathcal{P}_{(h, t)} \leftarrow \mathcal{P}_{(h, t)} \oplus \text{OneHotEncoding}(\mathbf{p})$ 
5 end
6  $\mathcal{P}_{(h, t)} \leftarrow \mathbf{W} \mathcal{P}_{(h, t)}$ 
7 return  $\mathcal{P}_{(h, t)}$ 

```

4.3 Training Objective

To achieve the main objective of predicting relations over KGs, we borrow the idea of a loss function based on the distribution probability of a relation over a head and tail pair from [67]. Like so, we take the probability distribution computed by a softmax function from the addition of both entities context $\mathcal{M}_{(h, t)}$ and semantic path context $\mathcal{P}_{(h, t)}$ representations:

$$P(r|h, t) = \text{softmax}(\mathcal{M}_{(h, t)} + \mathcal{P}_{(h, t)}). \quad (4.12)$$

The training loss is represented in Equation 4.13, where our objective is to minimize the cross-entropy loss between the predicted probability of a training fact and its ground-truth:

$$\min \mathbf{L}(\Omega) = - \sum_{(h,r,t) \in \mathcal{T}} r \log(P(r|h,t)). \quad (4.13)$$

As a result of the training phase, the model learns a distribution function over a training set that maps the probability of a relation regarding two entities. To infer new relationships after training given a query, *i.e.*, a pair of entities in the form of head and tail, and a relation, we retrieve the learned representations of each query’s elements and apply the learned distribution function over the representations aiming to measure the probability of the relationship to be true.

4.4 Discussions

In this chapter, we propose and develop a new method for learning representations in knowledge graphs. Our approach, named \mathcal{A} EMP, proposes: *(i)* a new attention-enhanced message passing scheme to learn entities context representations (Section 4.1); *(ii)* semantic paths’ representations based on relations’ embeddings (Section 4.2); and *(iii)* the combined use of both representations to the final attention-enhanced contextualized representations (Section 4.3).

Although \mathcal{A} EMP was introduced in this chapter in the form of a specialized framework, it can easily be generalized and extended to support design alternatives in its three main components. In the following, we develop some discussions towards design alternatives possible in a general formulation of \mathcal{A} EMP.

4.4.1 Design Alternatives

4.4.1.1 Message passing scheme

The message passing scheme is generalizable through an *update* and an *aggregate* functions [27]. Thus, Equations 4.1 and 4.2 can be posed as the sum of the incoming neighbors’ messages and a non-linear transformation over a linear combination of the previous embedding with the neighborhood information.

Given the general form of the message passing framework, new entities neighborhood aggregators can be applied. We show some examples in the following:

Average neighbor aggregator [67]. Applies a non-linear transformation over the input

vectors' element-wise average:

$$s_e^{i+1} = \sigma \left(\frac{1}{3}(m_h^i + m_t^i + s_r^i)\mathbf{W} + b \right), h, t \in \mathcal{N}(r). \quad (4.14)$$

Graph convolutional networks (GCNs) [32]. Adopt a self-loop update strategy aligned with a symmetric-normalized aggregation function:

$$s_e^{i+1} = \sigma \left(\mathbf{W} \sum_{v \in \mathcal{N}(h) \cup \{h\}} \frac{s_e^i}{\sqrt{|\mathcal{N}(h)| |\mathcal{N}(v)|}} \right). \quad (4.15)$$

4.4.1.2 Path representation

Different composition operators can be used to provide the semantic paths' representations. As described in Section 4.2, we use the concatenation operator in the \mathcal{A} EMP framework over the relations' representations that composes the target semantic path. Thus, we can generalize the Equation 4.11 to a generic composition function. In the following, we show the use of Recurrent Neural Networks (RNNs) as a composition function. One of the advantages of using RNNs to represent the semantic paths is that they can potentially capture patterns among different semantic paths.

Recurrent neural networks (RNNs) [40]. Applies a recurrent neural network over the relations to compose the semantic path representation.

$$\mathcal{P}_{(h,t)} = f(\mathbf{W}[c_{i-1}; r_i]), \quad \forall r_i \in \mathbf{P}(h, t), \quad i = |\mathbf{P}_{\mathbf{p}_0}^{\mathbf{p}_k}|, \quad (4.16)$$

where $\mathbf{P}(h, t)$ is the set of relations within the target semantic path.

This chapter's content will base all empirical evaluation and further studies presented in Chapter 5.

Chapter 5

Experimental Results

This chapter presents the experimental methodology and $\mathcal{A}EMP$'s evaluation in the relation prediction task targeting knowledge base completion. We conduct experiments on real-world datasets largely used in the literature and report the obtained results compared with several previous methods that have also focused on the relation prediction task. Furthermore, we conduct three ablation studies to draw insights over $\mathcal{A}EMP$ and its variations. Thus, Section 5.1 approaches the experimental protocol; Section 5.2 covers the results and comparisons on the relation prediction task; Section 5.3 discuss over the ablation studies; and Section 5.4 debates over the experimental results brought in this chapter.

5.1 Experimental Settings

Datasets. To evaluate $\mathcal{A}EMP$'s specialized framework proposed in the Chapter 4 in the relation prediction task for knowledge base completion (Section 1.3.1), we conduct the experiments based on four datasets, namely WN18, WN18RR, FB15k, and FB15k-237, extracted from two widely used knowledge graphs.

The datasets FB15k and WN18 were first introduced by Borders *et al.* [7] aiming to provide a benchmark to evaluate knowledge base completion techniques. Specifically, the FB15k is a subset from Freebase [6], a knowledge graph containing human knowledge facts, such as exemplified in Table 5.1. This dataset contains a total of 592213 facts that links 14951 entities with 1345 relations. The WN18 is a subset from WordNet [46], a knowledge graph containing lexical relations among English words, similarly as before exemplified in Table 5.1. This dataset contains a total of 40943 entities and 18 relations

organized into 151442 facts.

WN18RR [14] and FB15k-237 [62] are subsets of WN18 and FB15k, respectively. Both datasets were proposed after the identification [14, 62] of major data leakage in test sets, where a large number of test triples can be obtained simply by inverting triples in the training set. In this sense, FB15k-237 is composed of 14541 entities and 237 relations, summing 310116 facts, where the original FB15k’ inverse relations were removed. In a similar way, WN18RR was introduced to overcome the data leakage problem on WN18 by featuring 11 relations from the original 18 and the same 40943 entities, summing 93003 facts.

Table 5.1 displays examples of facts from WN18 and FB15k datasets, and Table 5.2 shows some statistics regarding the four above-mentioned knowledge graphs.

Table 5.1: Examples of facts in the format (*subject, predicate, object*) from datasets WN18 and FB15k.

WN18	FB15k
(02174461, _HYPERNYM, 02176268)	(/M/07PD_J, /FILM/FILM/GENRE, /M/02L7C8)
(05074057, _DERIVATIONALLY_RELATED_FORM, 02310895)	(/M/06WXW, /LOCATION/LOCATION/TIME_ZONES, /M/02FQWT)
(08390511, _SYNSET_DOMAIN_TOPIC_OF, 08199025)	(/M/05ZR0XL, /TV/TV_PROGRAM/LANGUAGES, /M/02H40LC)
(02045024, _MEMBER_MERONYM, 02046321)	(/M/0GK4G, /PEOPLE/CAUSE_OF_DEATH/PEOPLE, /M/0L9K1)

Table 5.2: Datasets statistics summary.

Dataset	Facts (triples)				Entities		
	Training	Validation	Test	Total	Total	Unique	Average degree
WN18	141442	5000	5000	151442	40943	5	7.39 ± 16.46
WN18RR	86835	3134	3034	93003	40943	5754	4.54 ± 8.57
FB15k	483142	59071	50000	592213	14951	21	79.22 ± 220.72
FB15k-237	272115	20466	17535	310116	14541	314	42.65 ± 127.70

Baselines. We compare $\mathcal{A}EMP$, and its variations, with six state-of-the-art models, namely, TransE [7], ComplEx [64], SimpleE [30], RotatE [61], QuatE [73], and PathCon [67]. TransE, ComplEx, SimpleE, RotatE, and QuatE are the models representing the state-of-the-art in embedding-based models, while PathCon is the state-of-the-art closer to our model, using entities context and relational path as features in the learning process. We reused the results reported in [67].

Implementation details. $\mathcal{A}EMP$ is publicly available at <https://github.com/MeLL-UFF/AEMP>.

We implemented it using Python and PyTorch¹ and trained using a single Nvidia V100 GPU². During our experiments, we vary the hyperparameters accordingly to Table 5.3. In our best experimentation settings, we employed a learning rate of 10^{-3} with Adam [31] optimizer. Also, to avoid overfitting, we employ L2 regularization using L2 weight loss of 10^{-7} . Besides, we adopt a batch size of 128 (the number of training examples adopt in one iteration), 25 training epochs (the number of training iterations), hidden states of 64 dimensions, and 0.2 as the random attention context selection criteria. Finally, on WN18 and WN18RR benchmarks, we employed 3-hops entities context and semantic path length up to 3; on FB15k and FB15k-237 benchmarks, we employed 2-hops entities context and semantic paths' length up to 2. We adopt different values for entities context hops and semantic paths' length due to hardware limitations considering each benchmark's size.

In order to provide negative examples, we utilize the negative sampling strategy to corrupt the relation r of each true fact (h, r, t) [67]. The strategy simply corrupt a triple $(h, r, t) \in \mathcal{F}$ by providing negative samples $(h, r', t) \notin \mathcal{F}$. The corrupted triples are used as negative examples in the training phase.

Table 5.3: Search space of the \mathcal{A} EMP's hyperparameters.

Hyperparamter	Search space
Batch size	{64, 128}
Epoch	{25, 50}
Hidden state dimension	{64, 128}
L2 regularization weight	$\{10^{-6}, 10^{-7}, 10^{-8}\}$
Learning rate	$\{10^{-1}, 10^{-2}, 10^{-3}\}$
Maximum entities context hops	{1, 2, 3}
Maximum semantic path length	{1, 2, 3}
Random attention context selection criteria	{0.2, 0.25, 0.5, 0.8}

Evaluation protocol. We evaluate \mathcal{A} EMP and the state-of-the-art models under the *relation prediction* task. As described in Section 1.3.1, the task aims to infer a new fact $f = (h, r, t)$ by predicting a relation r given a pair of entities (h, t) . We selected and reported results from Mean Reciprocal Rank (MRR), which is the average of the reciprocal ranks of a query's results

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}, \quad (5.1)$$

¹<https://pytorch.org/>

²<https://www.nvidia.com/en-us/data-center/v100/>

where Q is the query element, and $rank_i$ refers to the first relevant element's rank position for the i -th query; the Mean Rank (MR), which is the average of the predicted ranks of a query

$$MR = \sum_{i=1}^{|Q|} \frac{rank_i}{|Q|}, \quad (5.2)$$

where Q is the query element; and correctly predicted relations (Hit ratio) in the top 1 and 3 ranks evaluation metrics

$$Hit@k = \frac{1}{|Q|} \sum_{i=1}^{|Q|} 1 \text{ if } rank_i \leq k, \quad (5.3)$$

where Q is the query element and $k \in \{1, 3\}$ is the rank. A lower value of MR points out better results, while the other metrics, *i.e.*, MRR and Hit@k target higher values. The reported results from PathCon [67] and AEMP in each dataset are the averages and standard deviation values from five independent executions.

In the following, we show some examples of how to calculate the aforementioned metrics. In the example, we show two queries (*PatrickStewart, Picard*) and (*IanMcKellen, PatrickStewart*), and the top four predicted relations among the entities by a hypothetical model. In the first query, the model ranked the correct relation in the third position, while, in the second query, the model ranked the correct relation in the first position.

head	relation	tail	score	rank	
PatrickStewart	BornIn	Picard	0.823	1	
PatrickStewart	EnemyWith	Picard	0.751	2	
PatrickStewart	Play	Picard	0.718	3	*
PatrickStewart	CharacterIn	Picard	0.423	4	
IanMcKellen	FriendOf	PatrickStewart	0.961	1	*
IanMcKellen	EnemyWith	PatrickStewart	0.930	2	
IanMcKellen	Play	PatrickStewart	0.718	3	
IanMcKellen	CharacterIn	PatrickStewart	0.423	4	
$MR = \frac{1}{2} * (3 + 1) = 0.5$ $MRR = \frac{1}{2} * (\frac{1}{3} + \frac{1}{1}) = 0.66$ $Hit@1 = \frac{1}{2} * ((3 \leq 1) + (1 \leq 1)) = \frac{1}{2} * (0 + 1) = 0.5$ $Hit@3 = \frac{1}{2} * ((3 \leq 3) + (1 \leq 3)) = \frac{1}{2} * (1 + 1) = 1.0$					

5.2 Results

To evaluate $\mathcal{A}EMP$, we compare it to several state-of-the-art solutions, and the empirical results are reported in Table 5.4 and Table 5.5. Our approach demonstrates competitive results, outperforming the state-of-the-art methods on all metrics for WN18 and WN18RR datasets and two out of four metrics for FB15k. For FB15k-237, PathCon achieves the higher values on three metrics, yet $\mathcal{A}EMP$ surpassed the other state-of-the-art models. These findings go towards our initial premise that contextual information does enhance the learned representations.

In general, $\mathcal{A}EMP$ results using the global attention for learning the entities context achieves the best overall results reaching the best or the second-best cases seven times. Following, we see that the random attention mechanism is in six times in the first two positions (either it is the best or the second-best result), followed by the combination of local and global attention mechanisms and the random and global attention mechanisms (they are both fives times in the first two places). While the global mechanism may have pushed forward those results, we can see that there are several cases where global attention is neither the best nor the second-best choice. At the same time, some other method still wins – for example, Hit@3 of WN18 and MRR of FB15k-237 of the random mechanism is better than global alone or some of its combinations. We highlight in Table 5.6 the best and second MRR metric in each dataset.

The more significant absolute gains occur in WN18 and WN18RR, which are sparse KGs, *i.e.*, KGs of low average entities degree. This observation indicates that the attention mechanisms are able to reinforce connective patterns between entities, leading to better representations. Further investigating the learning capabilities enabled by each attention mechanisms and their combinations on WN18RR, Figure 5.1 depicts the confusion matrices of each $\mathcal{A}EMP$'s variation, where the axes are ordered in descending order, *i.e.*, top-bottom and left-right for the y-axis and the x-axis, respectively, and the heatmap indicates the Hit@1 metric. The local attention mechanism ($\mathcal{A}EMP$ (L) – Figure 5.1.(a)) demonstrates better performance on predicting commonly seen relations almost perfect score on the top two most common relations, however for rarely seen relations the method tends to present poor results. In contrast, the global and random attention mechanisms ($\mathcal{A}EMP$ (G) – Figure 5.1.(b) and $\mathcal{A}EMP$ (R) – Figure 5.1.(c), respectively) presents smoother results on rarely seen relations, but both perform worst than the $\mathcal{A}EMP$ (L) on the most common relations. The combination of local attention mechanism with global or local attention mechanisms ($\mathcal{A}EMP$ (L+G) – Figure 5.1.(d) and $\mathcal{A}EMP$ (L+R) – Figure 5.1.(e),

respectively) presents improvement on rarely seen relations, while keeping the good performance on commonly seen relations. Finally, the three combined attentions mechanisms ($\mathcal{A}EMP$ (L+G+R) – Figure 5.1.(g)) suffers on the least seen relation. However, the model presents overall better results, achieving good higher hit ratios on most relations.

Learning entities context assisted by the random attention mechanisms demonstrates better results over denser (higher average entities degree) knowledge graphs, *i.e.*, FB15k, and FB15k-237 datasets, as seen in Table 5.2. PathCon presents itself as an excellent choice for FB15k-237. Regarding all the datasets and metrics, it reaches the best results in five of the cases and the second-best results in three cases. However, $\mathcal{A}EMP$ ties with PathCon in two of the five winner situations (MRR of FB15k-237 and Hit@3 of FB15k) and is close in the other two (Hit@1 and Hit@3 of FB15k-237). These results further indicate that adding different patterns of attention favors finding missing relations in KGs.

Table 5.4: Relation prediction results on WN18 and WN18RR datasets. (L), (G), (R) stands for \mathcal{A} EMP local, global, and random attention patterns, respectively. [*]: Results are taken from [67]. The best result value is in bold and second best result value is underlined.

Model	WN18				WN18RR			
	MRR	MR	Hit@1	Hit@3	MRR	MR	Hit@1	Hit@3
TransE*	0.971	1.160	0.955	0.984	0.784	2.079	0.669	0.870
ComplEx*	0.985	1.098	0.979	0.991	0.840	2.053	0.777	0.880
Simple*	0.972	1.256	0.964	0.976	0.730	3.259	0.659	0.755
RotatE*	0.984	1.139	0.979	0.986	0.799	2.284	0.735	0.823
QuatE*	0.981	1.170	0.975	0.983	0.823	2.404	0.767	0.852
PathCon	0.9915 \pm 0.0007	1.0275 \pm 0.0039	0.9859 \pm 0.0010	0.9970 \pm 0.0007	0.9689 \pm 0.0025	1.0839 \pm 0.0088	0.9447 \pm 0.0038	0.9935 \pm 0.0013
\mathcal{A} EMP (L)	0.9450 \pm 0.0526	1.1503 \pm 0.1144	0.8991 \pm 0.0983	0.9958 \pm 0.0011	0.9632 \pm 0.0134	1.1257 \pm 0.0312	0.9366 \pm 0.0283	0.9876 \pm 0.0054
\mathcal{A} EMP (G)	0.9917 \pm 0.0006	1.0274 \pm 0.0018	0.9863 \pm 0.0008	0.9969 \pm 0.0004	0.9764 \pm 0.0014	1.0645 \pm 0.0047	0.9580 \pm 0.0024	0.9942 \pm 0.0009
\mathcal{A} EMP (R)	0.9908 \pm 0.0008	1.0282 \pm 0.0021	0.9845 \pm 0.0014	0.9971 \pm 0.0004	0.9710 \pm 0.0002	1.0773 \pm 0.0034	0.9480 \pm 0.0008	0.9948 \pm 0.0009
\mathcal{A} EMP (L+G)	0.9942 \pm 0.0004	1.0288 \pm 0.0037	0.9920 \pm 0.0005	0.9952 \pm 0.0003	0.9792 \pm 0.0019	1.0717 \pm 0.0079	0.9659 \pm 0.0027	0.9916 \pm 0.0023
\mathcal{A} EMP (L+R)	0.9717 \pm 0.0101	1.1020 \pm 0.0343	0.9532 \pm 0.0164	0.9904 \pm 0.0056	0.9807 \pm 0.0035	1.0736 \pm 0.0100	0.9677 \pm 0.0074	0.9908 \pm 0.0021
\mathcal{A} EMP (G+R)	0.9907 \pm 0.0007	1.0297 \pm 0.0037	0.9846 \pm 0.0010	0.9965 \pm 0.0005	0.9758 \pm 0.0029	1.0652 \pm 0.0074	0.9568 \pm 0.0049	0.9949 \pm 0.0011
\mathcal{A} EMP (L+G+R)	0.9940 \pm 0.0005	1.0336 \pm 0.0075	0.9917 \pm 0.0007	0.9949 \pm 0.0006	0.9786 \pm 0.0010	1.0679 \pm 0.0047	0.9635 \pm 0.0018	0.9937 \pm 0.0011

Table 5.5: Relation prediction results on FB15k and FB15k-237 datasets. (L), (G), (R) stands for \mathcal{A} EMP local, global, and random attention patterns, respectively. [*]: Results are taken from [67]. The best result value is in bold and second best result value is underlined.

Model	FB15k			FB15k-237				
	MRR	MR	Hit@1	Hit@3	MRR	MR	Hit@1	Hit@3
TransE*	0.962	1.684	0.940	0.982	0.966	1.352	0.946	0.984
CompLex*	0.901	1.553	0.844	0.952	0.924	1.494	0.879	0.970
Simple*	0.983	1.308	0.972	0.991	0.971	1.407	0.955	0.987
RotatE*	0.979	1.206	0.967	0.986	0.970	1.315	0.951	0.980
QuatE*	0.984	1.207	0.972	0.991	0.974	1.283	0.958	0.988
PathCon	0.9821 \pm 0.0002	1.5115 \pm 0.0585	0.9699 \pm 0.0003	0.9940 \pm 0.0002	0.9797 \pm 0.0005	1.1588 \pm 0.0214	0.9653 \pm 0.0009	0.9944 \pm 0.0004
\mathcal{A} EMP (L)	0.9689 \pm 0.0017	1.0823 \pm 0.0060	0.9445 \pm 0.0028	0.9876 \pm 0.0022	0.7900 \pm 0.0546	2.8834 \pm 0.4814	0.7188 \pm 0.0664	0.8296 \pm 0.0649
\mathcal{A} EMP (G)	0.9824 \pm 0.0004	1.4948 \pm 0.0637	0.9704 \pm 0.0005	0.9940 \pm 0.0003	0.9790 \pm 0.0007	1.1950 \pm 0.0326	0.9640 \pm 0.0012	0.9943 \pm 0.0004
\mathcal{A} EMP (R)	0.9815 \pm 0.0004	1.4466 \pm 0.0806	0.9688 \pm 0.0006	0.9938 \pm 0.0002	0.9797 \pm 0.0004	1.1922 \pm 0.0255	0.9652 \pm 0.0006	0.9941 \pm 0.0004
\mathcal{A} EMP (L+G)	0.9798 \pm 0.0015	1.0666 \pm 0.0095	0.9664 \pm 0.0021	0.9917 \pm 0.0029	0.9765 \pm 0.0011	1.3631 \pm 0.0941	0.9643 \pm 0.0017	0.9877 \pm 0.0003
\mathcal{A} EMP (L+R)	0.9032 \pm 0.0136	3.2039 \pm 0.2885	0.8714 \pm 0.0161	0.9202 \pm 0.0144	0.8724 \pm 0.0436	2.7244 \pm 0.5824	0.8369 \pm 0.0545	0.8885 \pm 0.0446
\mathcal{A} EMP (G+R)	0.9823 \pm 0.0002	1.4866 \pm 0.0478	0.9704 \pm 0.0004	0.9938 \pm 0.0003	0.9790 \pm 0.0004	1.2103 \pm 0.0442	0.9643 \pm 0.0007	0.9940 \pm 0.0004
\mathcal{A} EMP (L+G+R)	0.9786 \pm 0.0020	1.0695 \pm 0.0075	0.9641 \pm 0.0039	0.9924 \pm 0.0027	0.9754 \pm 0.0011	1.4346 \pm 0.0467	0.9622 \pm 0.0018	0.9877 \pm 0.0006

Table 5.6: Best (1st) and the second-best (2nd) MRR metric in each dataset.

	WN18	WN18RR	FB15k	FB15k-237
TransE				
ComplEx				
SimplE				
RotatE				
QuatE			1st	
PathCon				1st
ÆMP (L)				
ÆMP (G)			2nd	2nd
ÆMP (R)				1st
ÆMP (L+G)	1st	2nd		
ÆMP (L+R)		1st		
ÆMP (G+R)				2nd
ÆMP (L+G+R)	2nd			

5.3 Ablation Studies

To better assess the capabilities of ÆMP we conduct three ablation studies. As such, we *(i)* evaluate the benefits of using semantic paths’ representations towards the capacity of ÆMP to predict relations, and *(ii)* investigate the influence of the number of hops and the number of context neighbors, towards the predictions’ results. Further, we *(iii)* analyze the scalability capacity of ÆMP. Results are illustrated in Figures 5.2, 5.3, and 5.4 and details of used parametrization are in Table 5.7.

Table 5.7: Ablation studies parametrization settings.

Hyperparamter	Ablation Studies		
	<i>(i)</i>	<i>(ii)</i>	<i>(iii)</i>
Batch size	128	128	128
Epoch	20	10	5
Hidden state dimension	128	128	128
L2 regularization weight	10^{-7}	10^{-7}	10^{-7}
Learning rate	10^{-3}	10^{-3}	10^{-3}
Entities context hops	3	{1,2,3}	3
Sample of entities neighbors	4	{1,4,16}	16
Semantic paths length	3	{1,2,4}	3
Random attention context selection criteria	0.2	0.2	0.2

In the first study (Figure 5.2), we assess the top 1 hit ratio (Hit@1) on WN18RR, varying each attention mechanism and their combination alongside either enabling or disabling the representation of semantic paths. Considering the variations that implement only a subset of the attention mechanisms, $\mathcal{A}EMP$ achieves its best performance when it is able to combine head and tail entities context representations with the representation of the semantic path. However, when all three local, global, and random attention mechanisms are enabled, and their patterns are observable, $\mathcal{A}EMP$ reaches equivalent results when either using the representation of the semantic paths or not. This observation empirically indicates that the attention-enhanced message-passing scheme might be able to not only represents entities local neighborhood but also to learn longer sequences of relationships, *i.e.*, the semantic paths between the head and tail entities.

The second study (Figure 5.3), similar to the first study, assesses the top 1 hit ratio on WN18RR. The study aims to evaluate the influence of the entities context hops, a sample of entities neighbors, and semantic paths' length hyperparameters in $\mathcal{A}EMP$'s (and a subset of its variations) performance. $\mathcal{A}EMP$ (circle symbol) achieves the overall best result using samples of up to 16 neighbors, two entities context hops, and semantic path length of four relations (Figure 5.3.(d)). In comparison, the subset variations achieve overall better results using the maximum numbers of entities context hops and semantic path length. Those results reinforce the previous study, suggesting that $\mathcal{A}EMP$ with its attention-enhanced message-passing scheme have generalization capabilities, capturing contextual semantics with less context information.

Also, we draw some insights concerning the interaction between hyperparameters. The use of semantic paths of length up to four achieves better results aligned with three hops from the entities contexts. This combination indicates that the use of contextual information surrounding the path contributes to better representations (Figure 5.3.(d)). A second insight taken from the study analysis is the performance of local $\mathcal{A}EMP$'s attention-only variation (square symbol) regarding the correlation between sample entities neighbors and entities context hops hyperparameters. $\mathcal{A}EMP$ (L) performs better when few neighbors are sampled, but it is allowed to look further in the neighborhood through the context hops (Figure 5.3.(b)). However, when it is provided to $\mathcal{A}EMP$ (L) more contextual information it performs better using a balanced hyperparameters settings, *i.e.*, four samples of entities neighbors, two entities context hops, and semantic paths of length two (Figure 5.3.(c)).

The third study (Figure 5.4) analyzes the scalability capacity of $\mathcal{A}EMP$. We measure its training time regarding the number of triples on each previously introduced dataset

and on a new dataset, ogbl-biokg [29], a KG composed of a large number of biomedical facts (5088434 triples). Results show that the training time curve of $\mathcal{A}EMP$ grows slower than linear time over the number of triples, indicating that it is suitable to learn from larger KGs. Also, our results indicate that the scalability of $\mathcal{A}EMP$ is not only ruled by the number of triples, but other factors might interfere on training time (*e.g.*, the sparsity of the knowledge graph).

5.4 Discussions

This chapter evaluates the performance of $\mathcal{A}EMP$ as presented in Chapter 4 and variations of it. First, we compare $\mathcal{A}EMP$ and its variations with six state-of-the-art approaches in the task they were designed for, *i.e.*, completing knowledge bases by predicting new relationships among entities using four of the most widespread benchmarks in the literature. Further, we conduct three ablation studies, where we closely investigate the influence of semantic paths and hyperparameters settings in the overall technique’s performance and analyze the technique’s scalability capabilities.

Overall, $\mathcal{A}EMP$ prove to be a better or, at least, a competing technique for predicting new facts in knowledge bases. Its multi-context attention layer innovates on gathering entities context information, even, at some level, generalizing semantic paths. Also, it is shown that $\mathcal{A}EMP$ can scale to larger knowledge graphs.

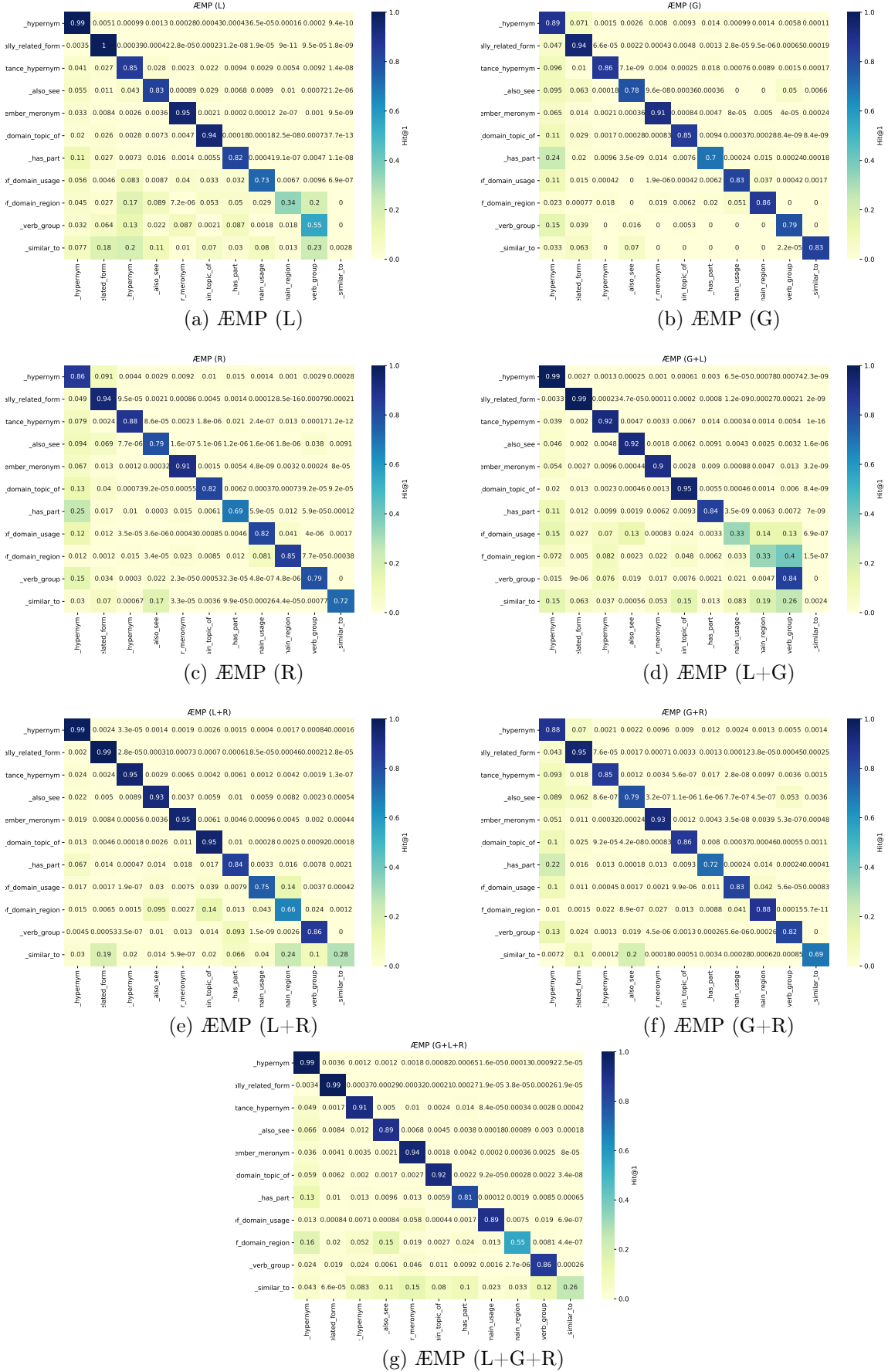


Figure 5.1: Confusion matrices of the ground truth relations and predicted relations by each \mathcal{AEMP} 's variation. The heatmap indicates the Hit@1 metric varying from 0 to 1, and axes are in descending order (top-bottom for y-axis, and left-right for the x-axis) regarding the number of triples in which the relation is the predicate.

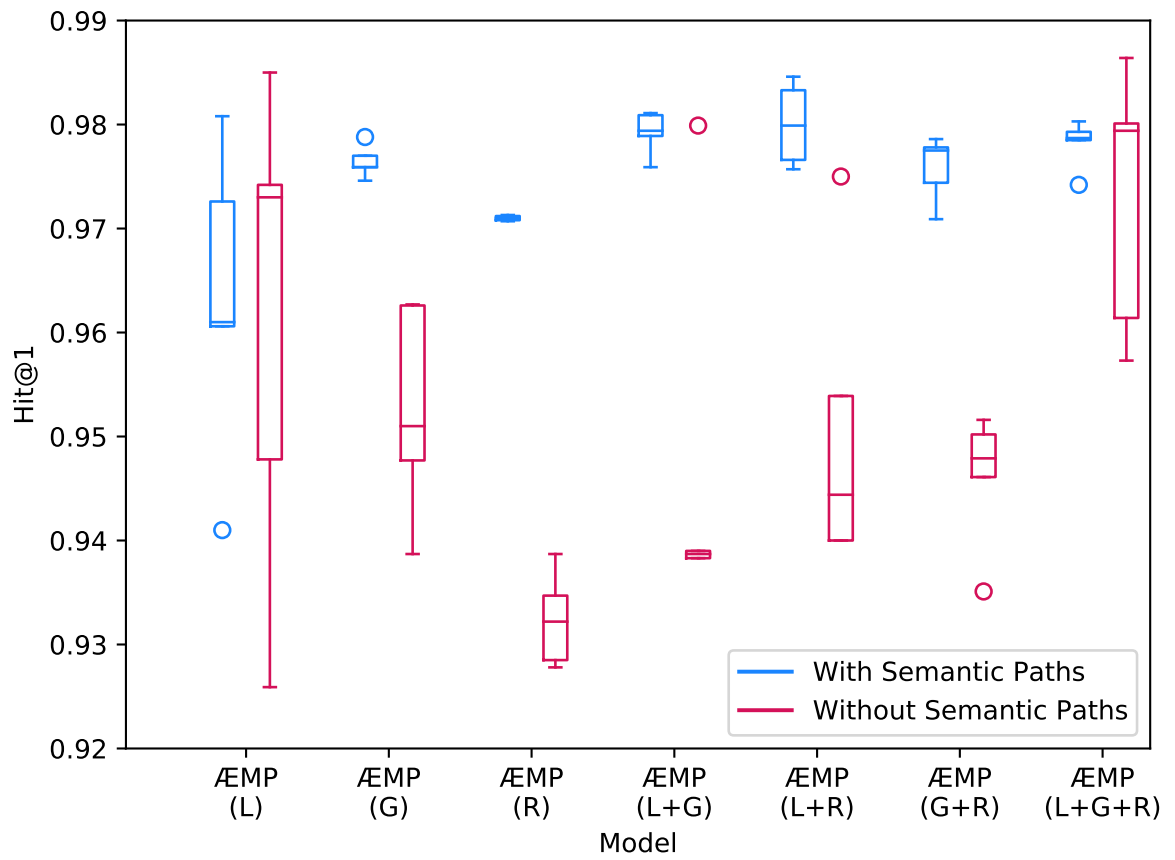


Figure 5.2: Boxplot of the Hit@1 results from $\mathcal{A}EMP$ and its subset variations using (or not) the semantic paths' representation to predict new relations.

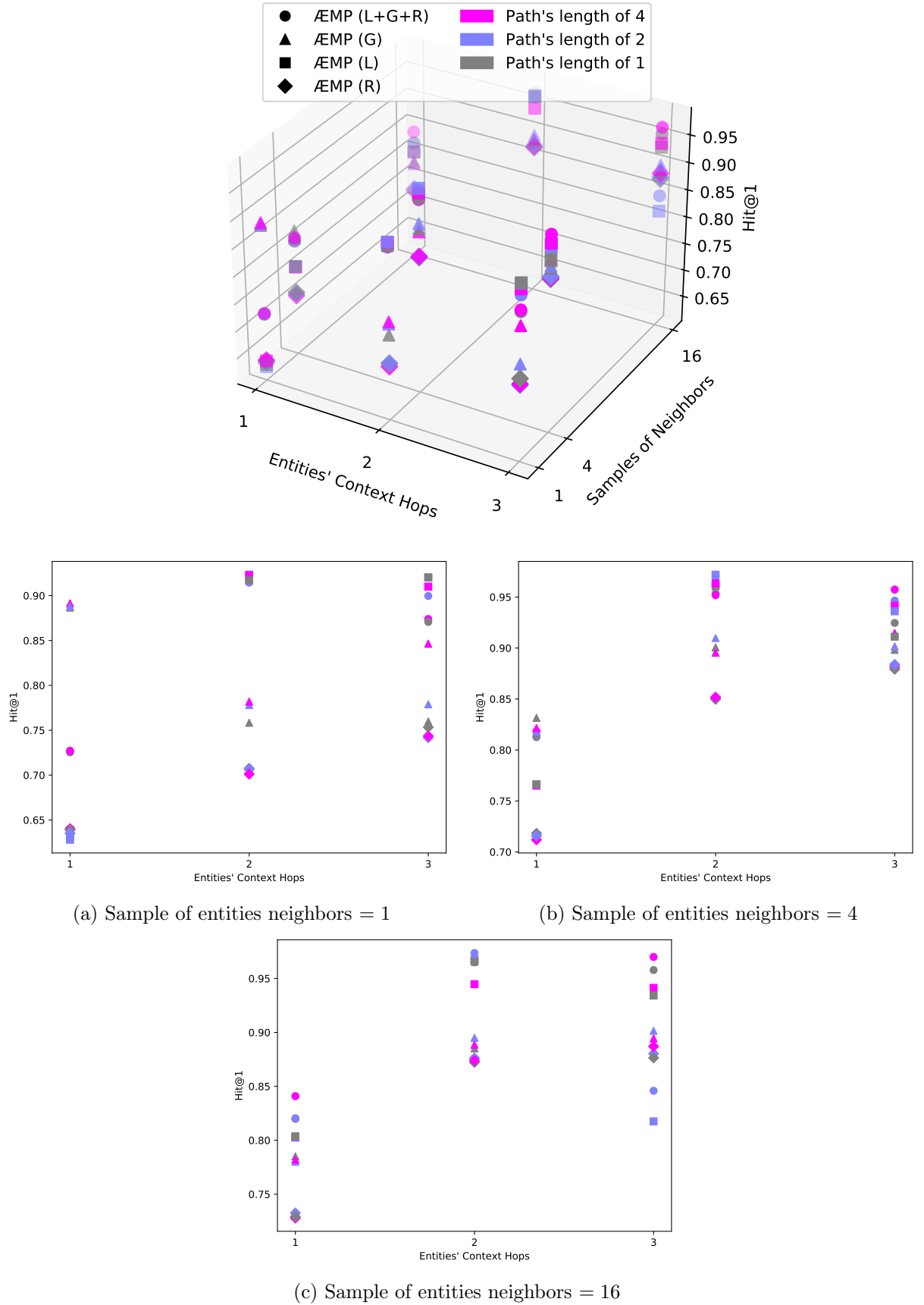


Figure 5.3: Average Hit@1 results from ÆMP and its subset variations regarding the variation of entities context hops, semantic paths length, and a sample of entities neighbors hyperparameters.

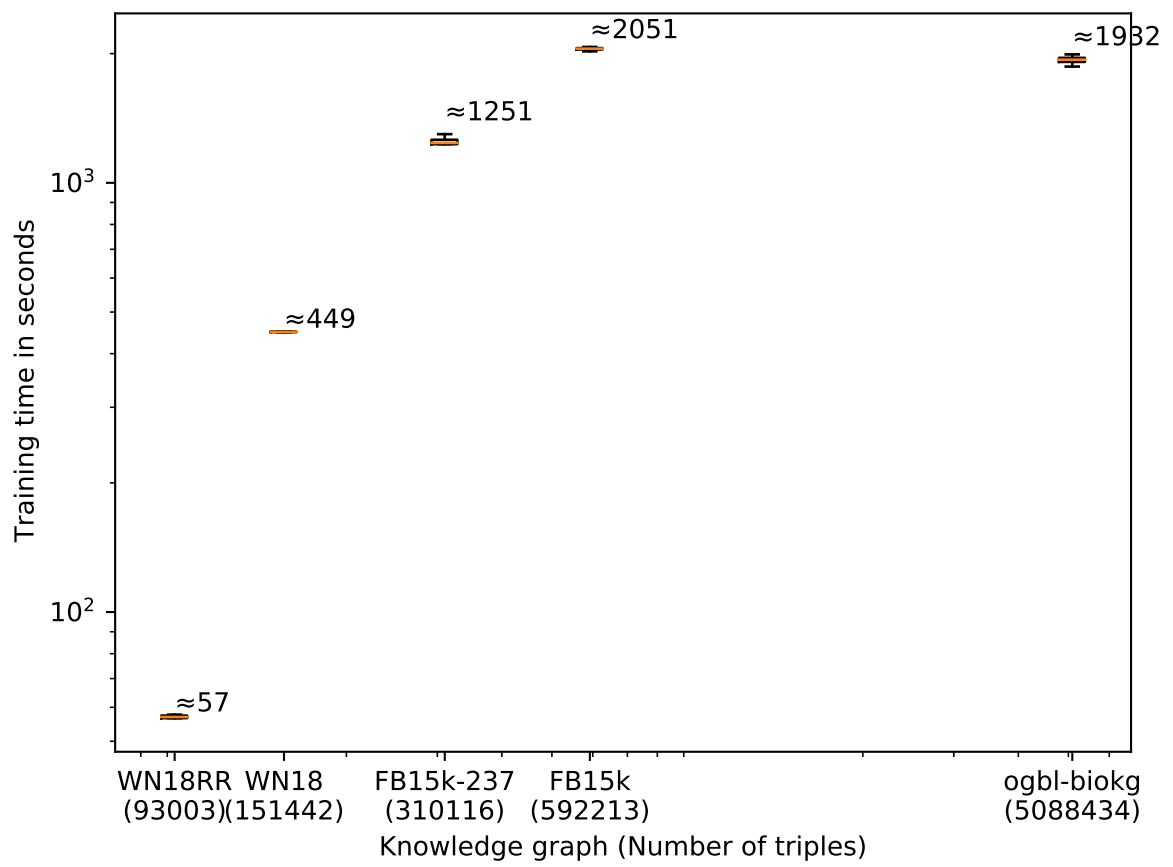


Figure 5.4: Boxplot of the training time from $\mathcal{A}EMP$ regarding different sized KGs.

Chapter 6

Conclusions

Relational data and their graph representation, particularly knowledge and knowledge graphs (KGs), are powerful resources to model elements, properties, relationships, rules, among others. They are broadly adopted in the academy and industry as core resources for many downstream tasks, including, but not limited to, large-scale search, question-answering, and social network modeling. Intending to provide more robust and reliable knowledge graphs, we propose a new solution to complete their information based on exclusive existing information.

In this sense, this dissertation devises Attention-based Embeddings from Multiple Patterns (\mathcal{AEMP}), a novel symbolic-inspired distributional method that learns contextualized representations from the combination of distinct views of entities context and semantic path context to complete knowledge graphs. Firstly, \mathcal{AEMP} learns the joint representation of the entities and their context through a novel attention-enhanced message-passing scheme that features a multi-context attention layer built on top of the message-passing scheme. The attention layer extends the graph attention mechanism by adopting local, global, and random attention mechanisms. Secondly, the model learns the representation of the semantic path context by identifying semantic paths between a pair of entities, providing to each of them a unique representation, and then fusing these representations into a single semantic path context representation. From the combination of both context representations, the model can successfully infer a relationship’s probability within two entities.

We conduct an experimentation with four datasets based in two real-world knowledge bases to evaluate our model and compare our results with six state-of-the-art approaches in knowledge graph embedding. Our results show that \mathcal{AEMP} has the potential to outperform the state-of-the-art models in the relation prediction task. Likewise, we dissect \mathcal{AEMP}

exploring aspects regarding the influence of the parametrization, aside from the contextual information considered and its scalability capacities. The empirical analysis indicates that the proposed attention-enhanced message-passing scheme can represent the entities and their context and the semantic path context. Also, it shows that \mathcal{AEMP} has the potential to scale to larger KGs.

6.1 Limitations

We identify two possible limitations of this work regarding implementing the model and further aspects to be analyzed in the ablation studies. The first limitation was diagnosed during the experiments and shows that the model is a hard memory-bound approach. Thus, the model might require a large amount of hardware memory to explore four or more hops in the message-passing phase since, within each hop, the number of neighbors to be considered in the aggregation process grows exponentially. The second limitation aims at a twofold ablation analysis, where the individual contribution for the overall result of the semantic paths representation and design alternatives as discussed in Chapter 4 should both be explored.

6.2 Future work

As an immediate and natural extension of this work, we highlight the evaluation of the design alternatives suggested in Section 4.4.1 and their influence in \mathcal{AEMP} 's overall performance. We believe, for instance, that the employment of networks that deal with sequential data (*e.g.*, recurrent networks or transformers) might provide better representation for longer semantic paths. We also point out as an important next step the assessment of \mathcal{AEMP} 's reasoning capabilities. For that, we expect to evaluate \mathcal{AEMP} under other knowledge bases, such as YAGO [56,60] and NELL [47], and they provide substantial comparisons with traditional symbolic approaches.

Besides the natural extensions aforementioned, we drive future enquires by proposing three research questions:

1. *Can the proposed attention-enhanced message-passing scheme be generalized to jointly learn in the aggregation loop the representation of the semantic paths?*
2. *Can \mathcal{AEMP} be used as framework in order to explore even further contexts (*e.g.*,*

temporal semantics of facts, domain rules) to encompass a broader range of tasks within the knowledge base completion challenge?; and

3. *To deal with multimodal knowledge graphs, how can we extend \mathcal{AEMP} to jointly learn over the relational information and the multimedia content?*

The first research question is inspired by the effectiveness of sparse attention mechanism over transformers [72]. The goal is to analyze the expressiveness of attention-enhanced message-passing towards learning semantic paths in the aggregation loop. As a possible first outcome by reaching this goal, a more general framework that in-loop represents entities context and semantic paths are expected. Another possible outcome is the scalability of the solution, once it will optimize a learning phase of the current \mathcal{AEMP} framework.

The second research question aim to address further additional information within knowledge graphs under the scope of \mathcal{AEMP} . In this sense, we suggest learning additional representations (*e.g.*, ontologies, temporal aspects [24, 69], logic rules [2, 58]) and encompass with the learned entities context and semantic paths representations.

The third research question is aligned with the second. It aims to explore extensions of the knowledge graph. Here, our goal is to approach knowledge graphs that encode not only entities and relations but, also, multimodal information [57] using \mathcal{AEMP} . We suggest guiding the study towards proposing modifications over the attention-enhanced message-passing scheme to jointly compute the multimodal content representations.

In conclusion, as a first major contribution, this dissertation advances the state-of-the-art in the knowledge base completion challenge by contributing with a new symbolic-inspired distributional solution to automatically complete knowledge graphs. Our second major contribution extends to the representation learning set of techniques by proposing a novel attention-enhanced message-passing scheme to learn contextualized entities' representations and the combination of the attention-based representations with the semantic paths' representation. Finally, we contribute by evaluating and conducting ablation studies over the two major contributions showing that the proposed solution is, at least, competitive with state-of-the-art approaches in the relation prediction task.

References

- [1] APICELLA, A., DONNARUMMA, F., ISGRÒ, F., PREVETE, R. A survey on modern trainable activation functions. *Neural Networks* 138 (2021), 14–32.
- [2] BADREDDINE, S., D’AVILA GARCEZ, A., SERAFINI, L., SPRANGER, M. Logic tensor networks, 2021.
- [3] BAHDANAU, D., CHO, K., BENGIO, Y. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015* (2015).
- [4] BENGIO, Y. AI Debate, 2019. Remarks by Professor Yoshua Bengio at the 2019 AI Debate hosted at Mila, Québec, Canada. Transcript available at <https://medium.com/@Montreal.AI/transcript-of-the-ai-debate-1e098eeb8465>. Accessed: 13/03/2019.
- [5] BENGIO, Y., COURVILLE, A., VINCENT, P. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 8 (2013), 1798–1828.
- [6] BOLLACKER, K., EVANS, C., PARITOSH, P., STURGE, T., TAYLOR, J. Freebase: a collaboratively created graph database for structuring human knowledge. In *In SIGMOD Conference* (2008), p. 1247–1250.
- [7] BORDES, A., USUNIER, N., GARCIA-DURAN, A., WESTON, J., YAKHNENKO, O. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26, NeurIPS 2016*. 2013, p. 2787–2795.
- [8] BOSSELU, A., RASHKIN, H., SAP, M., MALAVIYA, C., CELIKYILMAZ, A., CHOI, Y. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (Florence, Italy, 2019), Association for Computational Linguistics, p. 4762–4779.
- [9] CHAUDHARI, S., MITHAL, V., POLATKAN, G., RAMANATH, R. An attentive survey of attention models, 2020.
- [10] CORMEN, T. H., LEISERSON, C. E., RIVEST, R. L., STEIN, C. *Introduction to Algorithms, Third Edition*, 3rd ed. The MIT Press, 2009.
- [11] COSTABELLO, L., PAI, S., MCCARTHY, N., JANIK, A. Knowledge graph embeddings tutorial: From theory to practice, 2020. <https://kge-tutorial-ecai2020.github.io/>.

- [12] DALTON, J., DIETZ, L., ALLAN, J. Entity Query Feature Expansion Using Knowledge Base Links. In *Proceedings of the 37th international ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2014* (2014), ACM Press, p. 365–374.
- [13] DAS, R., NEELAKANTAN, A., BELANGER, D., MCCALLUM, A. Chains of reasoning over entities, relations, and text using recurrent neural networks. *arXiv preprint arXiv:1607.01426* (2016).
- [14] DETTMERS, T., MINERVINI, P., STENETORP, P., RIEDEL, S. Convolutional 2D Knowledge Graph Embeddings. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence, AAAI 2017* (2017), p. 1811–1818.
- [15] DING, B., WANG, Q., WANG, B., GUO, L. Improving knowledge graph embedding using simple constraints. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Melbourne, Australia, 2018), Association for Computational Linguistics, p. 110–121.
- [16] DUMANCIC, S., GARCIA-DURAN, A., NIEPERT, M. A comparative study of distributional and symbolic paradigms for relational learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19* (2019), p. 6088–6094.
- [17] GALÁRRAGA, L., RAZNIEWSKI, S., AMARILLI, A., SUCHANEK, F. M. Predicting completeness in knowledge bases. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining* (2017), WSDM '17, Association for Computing Machinery, p. 375–383.
- [18] GALÁRRAGA, L., TEFLIOUDI, C., HOSE, K., SUCHANEK, F. Fast Rule Mining in Ontological Knowledge Bases with AMIE+. *The VLDB Journal* (2015).
- [19] GALASSI, A., LIPPI, M., TORRONI, P. Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems* (2020), 1–18.
- [20] GARCEZ, A. S. D., LAMB, L. C., GABBAY, D. M. *Neural-Symbolic Cognitive Reasoning*, 1 ed. Springer Publishing Company, Incorporated, 2008.
- [21] GETOOR, L., TASKAR, B. *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2007.
- [22] GILMER, J., SCHOENHOLZ, S. S., RILEY, P. F., VINYALS, O., DAHL, G. E. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (2017), ICML'17, JMLR.org, p. 1263–1272.
- [23] GLOROT, X., BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics* (2010), Y. W. Teh and M. Titterton, Eds., vol. 9 of *Proceedings of Machine Learning Research*, PMLR, p. 249–256.
- [24] GOEL, R., KAZEMI, S. M., BRUBAKER, M., POUPART, P. Diachronic Embedding for Temporal Knowledge Graph Completion. *Proceedings of the AAAI Conference on Artificial Intelligence 34*, 04 (2020), 3988–3995.

- [25] GUO, L., SUN, Z., HU, W. Learning to exploit long-term relational dependencies in knowledge graphs. In *International Conference on Machine Learning, ICML 2019* (2019), p. 2505–2514.
- [26] GUU, K., MILLER, J., LIANG, P. Traversing knowledge graphs in vector space. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (2015), p. 318–327.
- [27] HAMILTON, W. L. Graph representation learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 14, 3 (2020), 1–159.
- [28] HITCHCOCK, F. L. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics* 6, 1-4 (1927), 164–189.
- [29] HU, W., FEY, M., ZITNIK, M., DONG, Y., REN, H., LIU, B., CATASTA, M., LESKOVEC, J. Open graph benchmark: Datasets for machine learning on graphs. *arXiv preprint arXiv:2005.00687* (2020).
- [30] KAZEMI, S. M., POOLE, D. Simple Embedding for Link Prediction in Knowledge Graphs. In *Advances in Neural Information Processing Systems 31, NeurIPS 2018* (2018).
- [31] KINGMA, D. P., BA, J. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015* (2015).
- [32] KIPF, T. N., WELLING, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
- [33] LAO, N., COHEN, W. W. Fast query execution for retrieval models based on path-constrained random walks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2010), KDD '10, Association for Computing Machinery, p. 881–888.
- [34] LAO, N., COHEN, W. W. Relational retrieval using a combination of path-constrained random walks. *Machine Learning* 81, 1 (2010), 53–67.
- [35] LAO, N., MITCHELL, T., COHEN, W. W. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (2011), Association for Computational Linguistics, p. 529–539.
- [36] LEE, J. B., ROSSI, R. A., KIM, S., AHMED, N. K., KOH, E. Attention models in graphs: A survey. *ACM Trans. Knowl. Discov. Data* 13, 6 (2019).
- [37] LEVESQUE, H. J. The Logic of Incomplete Knowledge Bases. In *On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases, and Programming Languages*, M. L. Brodie, J. Mylopoulos, and J. W. Schmidt, Eds. New York, NY, 1984, p. 165–189.
- [38] LIN, B. Y., CHEN, X., CHEN, J., REN, X. Kagnet: Knowledge-aware graph networks for commonsense reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint*

- Conference on Natural Language Processing, EMNLP-IJCNLP 2019* (2019), p. 2822–2832.
- [39] LIN, X., LIANG, Y., GIUNCHIGLIA, F., FENG, X., GUAN, R. Relation path embedding in knowledge graphs. *Neural Computing and Applications* 31, 9 (2019), 5629–5639.
- [40] LIN, Y., LIU, Z., LUAN, H., SUN, M., RAO, S., LIU, S. Modeling Relation Paths for Representation Learning of Knowledge Bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015* (2015), p. 705–714.
- [41] LIU, Z., XIONG, C., SUN, M., LIU, Z. Entity-duet neural ranking: Understanding the role of knowledge graph semantics in neural information retrieval. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2018* (2018), p. 2395–2405.
- [42] LUO, Y., WANG, Q., WANG, B., GUO, L. Context-Dependent Knowledge Graph Embedding. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015* (Stroudsburg, PA, USA, 2015), p. 1656–1661.
- [43] LUONG, M. T., PHAM, H., MANNING, C. D. Effective approaches to attention-based neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2015* (2015), p. 1412–1421.
- [44] MARTINS, A., ASTUDILLO, R. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *International Conference on Machine Learning, ICML 2016* (2016), p. 1614–1623.
- [45] MIKOLOV, T., CHEN, K., CORRADO, G., DEAN, J. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013* (2013), Y. Bengio and Y. LeCun, Eds.
- [46] MILLER, G. A. Wordnet: A lexical database for english. *Commun. ACM* 38, 11 (1995), 39–41.
- [47] MITCHELL, T., COHEN, W., HRUSCHKA, E., TALUKDAR, P., YANG, B., BETTERIDGE, J., CARLSON, A., DALVI, B., GARDNER, M., KISIEL, B., KRISHNAMURTHY, J., LAO, N., MAZAITIS, K., MOHAMED, T., NAKASHOLE, N., PLATANIOS, E., RITTER, A., SAMADI, M., SETTLES, B., WANG, R., WIJAYA, D., GUPTA, A., CHEN, X., SAPAROV, A., GREAVES, M., WELLING, J. Never-ending learning. *Commun. ACM* 61, 5 (2018), 103–115.
- [48] MOON, C., JONES, P., SAMATOVA, N. F. Learning entity type embeddings for knowledge graph completion. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (2017), CIKM '17, Association for Computing Machinery, p. 2215–2218.
- [49] MUGGLETON, S. Inverse entailment and prolog. *New Generation Computing* 13, 3 (1995), 245–286.

- [50] MUGGLETON, S., DE RAEDT, L. Inductive logic programming: Theory and methods. *The Journal of Logic Programming 19-20* (1994), 629–679. Special Issue: Ten Years of Logic Programming.
- [51] NATHANI, D., CHAUHAN, J., SHARMA, C., KAUL, M. Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, ACL 2019* (2019), p. 4710–4723.
- [52] NEELAKANTAN, A., ROTH, B., MCCALLUM, A. Compositional vector space models for knowledge base completion. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), ACL 2015* (2015), p. 156–166.
- [53] NICKEL, M., MURPHY, K., TRESP, V., GABRILOVICH, E. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE* 104, 1 (2016), 11–33.
- [54] OH, B., SEO, S., LEE, K.-H. Knowledge Graph Completion by Context-Aware Convolutional Learning with Multi-Hop Neighborhoods. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018* (New York, New York, USA, 2018), p. 257–266.
- [55] PALUMBO, E., RIZZO, G., TRONCY, R. Entity2rec: Learning user-item relatedness from knowledge graphs for top-n item recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017* (2017), p. 32–36.
- [56] PELLISSIER TANON, T., WEIKUM, G., SUCHANEK, F. YAGO 4: A Reasonable Knowledge Base. In *The Semantic Web* (2020), Springer International Publishing, p. 583–596.
- [57] PEZESHKPOUR, P., CHEN, L., SINGH, S. Embedding Multimodal Relational Data for Knowledge Base Completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018), Association for Computational Linguistics, p. 3208–3218.
- [58] QU, M., CHEN, J., XHONNEUX, L.-P., BENGIO, Y., TANG, J. Rnnlogic: Learning logic rules for reasoning on knowledge graphs, 2020.
- [59] RICHARDSON, M., DOMINGOS, P. Markov logic networks. *Mach. Learn.* 62, 1–2 (2006), 107–136.
- [60] SUCHANEK, F. M., KASNECI, G., WEIKUM, G. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web* (2007), WWW '07, Association for Computing Machinery, p. 697–706.
- [61] SUN, Z., DENG, Z.-H., NIE, J.-Y., TANG, J. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations, ICLR 2019* (2019).
- [62] TOUTANOVA, K., CHEN, D. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality* (Beijing, China, 2015), p. 57–66.

- [63] TOUTANOVA, K., CHEN, D., PANTEL, P., POON, H., CHOUDHURY, P., GAMON, M. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015* (2015), p. 1499–1509.
- [64] TROUILLON, T., WELBL, J., RIEDEL, S., GAUSSIER, E., BOUCHARD, G. Complex embeddings for simple link prediction. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning, ICML 2016* (2016), p. 2071–2080.
- [65] VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, L. U., POLOSUKHIN, I. Attention is all you need. In *Advances in Neural Information Processing Systems* (2017), I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30, Curran Associates, Inc.
- [66] VELIČKOVIĆ, P., CUCURULL, G., CASANOVA, A., ROMERO, A., LIÒ, P., BENGIO, Y. Graph attention networks. *International Conference on Learning Representations, ICLR 2018* (2018).
- [67] WANG, H., REN, H., LESKOVEC, J. Entity context and relational paths for knowledge graph completion, 2020.
- [68] WANG, Q., LIU, J., LUO, Y., WANG, B., LIN, C.-Y. Knowledge base completion via coupled path ranking. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2016* (2016), p. 1308–1318.
- [69] WU, J., CAO, M., CHEUNG, J. C. K., HAMILTON, W. L. TeMP: Temporal Message Passing for Temporal Knowledge Graph Completion. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2020), Association for Computational Linguistics, p. 5730–5746.
- [70] YAO, L., MAO, C., LUO, Y. Kg-bert: Bert for knowledge graph completion, 2019.
- [71] YIH, W.-T., CHANG, M.-W., HE, X., GAO, J. Semantic Parsing via Staged Query Graph Generation: Question Answering with Knowledge Base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, ACL 2015* (2015), Association for Computational Linguistics, p. 1321–1331.
- [72] ZAHEER, M., GURUGANESH, G., DUBEY, K. A., AINSLIE, J., ALBERTI, C., ONTANON, S., PHAM, P., RAVULA, A., WANG, Q., YANG, L., AHMED, A. Big bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems* (2020), H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., p. 17283–17297.
- [73] ZHANG, S., TAY, Y., YAO, L., LIU, Q. Quaternion Knowledge Graph Embeddings. In *Advances in Neural Information Processing Systems 32, NeurIPS 2019*. 2019, p. 2735–2745.