

# **LANGUAGE, STRUCTURE, TIME AWARE KNOWLEDGE BASE COMPLETION**

**PRACHI JAIN**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
INDIAN INSTITUTE OF TECHNOLOGY DELHI**

October 2022

©Indian Institute of Technology Delhi - 2022

# **LANGUAGE, STRUCTURE, TIME AWARE KNOWLEDGE BASE COMPLETION**

by

**PRACHI JAIN**

Department of Computer Science and Engineering

Submitted

in fulfillment of the requirements of the degree of Doctor of Philosophy

to the



**INDIAN INSTITUTE OF TECHNOLOGY DELHI**

**October 2022**

This dissertation is dedicated to  
my loving grandparents  
and  
my son Arinjay – may you never stop learning.

# Certificate

This is to certify that the dissertation titled **Language, Structure, Time aware Knowledge Base Completion** being submitted by **Ms Prachi Jain** for the award of **Doctor of Philosophy** in Department of Computer Science and Engineering is a record of bonafide work carried out by her under my guidance and supervision at the **Department of Computer Science and Engineering, Indian Institute of Technology Delhi**. The work presented in this dissertation has not been submitted elsewhere, either in part or full, for the award of any other degree or diploma unless otherwise stated explicitly. In particular, work done in Chapters 4, 5, 6 and 7 were done jointly with undergraduate students. In each case, the part done by the collaborators appeared in their respective bachelor's thesis.

**Mausam**

Professor

Department of Computer Science and Engg.

Indian Institute of Technology Delhi

New Delhi- 110016

**Soumen Chakrabarti**

Professor

Department of Computer Science and Engg.

Indian Institute of Technology Bombay

Powai, Mumbai, Maharashtra- 400076



# Acknowledgements

**“It’s all about the journey, not the outcome.”**

Carl Lewis

As I am about to end my past-due PhD journey. I want to look back and extend my gratitude to all the people whom I met in the course of my PhD, without whom the long journey would have looked very dull.

I would like to express my deepest gratitude to my advisors Mausam and Soumen for their guidance and support during my PhD. Mausam’s drive to do high-quality research and a constant push to go beyond the limits always kept me on my toes. His honest critical feedback always empowered me to be the best I can be. On the other hand, Soumen’s passion for the subject and ability to pay attention to the minutest of details always kept me motivated and pushed me to do my best. I have vastly benefited from this unique mix of advisors. In these last two years, there were times when work life was unavoidably affected by events outside work. I am so grateful to Mausam and Soumen for caring, understanding, and prioritizing their students’ well-being. Thank you Mausam and Soumen, for patiently being there in this period of my life and my career and always showering me with the wisest of advice.

I am thankful to Dr Denial Bernhardt and Apoorv Saxena for hosting me for an internship at Facebook. I would also like to thank Dr Sebastian Riedel, Sara Khodeir, and Gerard Goossen for their constant feedback and support during the internship.

I also want to acknowledge the TCS PhD fellowship for supporting me financially during my PhD. I am also thankful to Microsoft Research, Xerox Research, Google Research, IJCAI, and ACL student travel grants for supporting my travel to numerous conferences at various points in my PhD. I would also like to acknowledge the administrative staff (especially Rekha mam and Hemant bhaiya) at the Computer Science Department in IIT Delhi for their continuous help throughout these years. I was very lucky to have collaborated with some of the brightest undergraduate students on campus — Shikhar Murty, Kabir Chhabra, Pankaj Kumar, Mukund Mundhra, Sushant Rathi. I learnt a lot from them.

I would like to thank the organizing committee of the CIFAR Deep Learning + Reinforcement Learning (DLRL) Summer School and the Lisbon School of Machine Learning Summer

School (LxMLS) for organizing amazing summer schools. I learned a lot and made good friends during the summer schools. I would also like to thank them for supporting my travel. I will always remember the interesting discussions and fun trips with Disha during my time at DLRL summer school.

I have made lifelong friends over the last few years – Ankit, Happy, Himanshu and Yashoteja (aka JHAMP). Thank you for all the laughter, food, and wisdom to keep me moving over these years. I will always look upto them for any professional or personal advise. There are many more wonderful friends I made in my PhD — Yatin, Keshav, Vishal, Dilpreet, Jatin, Chandrika, Deepak, Madhulika, Prajna and Sanchi.

Finally, I would like to express my heartfelt gratitude to my parents for the numerous sacrifices they have made to support me and my education. Their unwavering faith always helped me sail through all the ups and downs of my PhD journey. I would also like to thank my grandparents, husband, sister, brother and parent-in-laws, without their love, support and encouragement I would not have achieved this feat.

# Abstract

Large Knowledge Bases (KBs) have been built to access a comprehensive collection of facts in a machine readable automatic format. Although these KBs are large, their coverage is far from complete. Most relations between entities are found to be missing in many widely-used KBs. Inference can be used to improve the coverage of such KBs and hence make them more suitable for practical applications like search, dialogue and question answering. This inference process is called Knowledge Base Completion (KBC).

In this dissertation, we analyze existing KBC systems and propose various new KBC methods and models. In particular, we exploit various attributes of KBs — language, structure and temporal attributes to improve KBC performance. We also extensively study the evaluation of KBC models and propose fair evaluation policies.

First, we studied different aspects of modeling KB structure and their impact on KBC performance:

- KBC models can be categorized on the basis of the way they represent entities. Matrix factorization (MF) models have a vector defined for each entity-*pair*, while tensor factorization (TF) models maintain a vector for each *entity*. We compared the effectiveness of the MF and TF paradigms for the general task of KBC. We recognized that special care is needed to handle out-of-vocabulary entity-pairs when evaluating MF against TF. We also propose the first fair unified KBC evaluation protocol to compare MF and TF approaches for KBC.
- Our analysis of KBC models reveals that they often make entity predictions that are incompatible with the type required by the relation. For example, DistMult incorrectly predicts ‘*Akira Isida*’ (type-person) for the query ‘*Chief Phillips (type-film), released\_in\_region, ?*’. We propose an unsupervised typing gadget, which enhances KBC models (like DistMult and Complex) with type-compatibility checkers. The enhanced models (TypeDM and TypeComplex) showed improved KBC performance over the base models. Further analysis revealed that our models better represent the latent types of entities and their embeddings also predict supervised types better than the embeddings learned by baseline models.

While implementing the above models, we found that the norm (L1, L2, or L3) used for regularization or measuring distances, and the rate of negative sampling used to train the models, can have significant consequences for KBC accuracy, sometimes overturning conventional wisdom about how various models compare with each other. Through our investigation of these issues, we implemented a very competitive version of ComplEx KB embedding, better than some follow-up systems.

Next, we study temporal KBs, which associate a relational fact  $(s, r, o)$  with a valid set of times (often an instant or interval). We propose TIMEPLEX a Temporal Knowledge Base Completion (TKBC) model, primarily targeted to the link-prediction and time-interval prediction tasks. To the best of our knowledge, this is the first work that predicts the time interval in which the given fact is valid in a general model-independent manner. Also, this is the first work that proposes a time-aware evaluation strategy for TKBC.

Finally, we study Open KBs where entities and relations are represented via textual schema-free strings. Open KBC is generally performed using an inference rule corpus. Using linguistic insights, we develop an algorithm — Knowledge Guided Linguistic Rewrites (KGLR) — which provides independent verification for statistically-generated Open KB inference rules. The generated high precision rule corpus eventually helps in improving the KBC task performance.

## सार

तथ्यों के व्यापक संग्रह को एक मशीन पठनीय स्वचालित प्रारूप में लाने के लिए बड़े ज्ञानकोष (नॉलेज बेस - के.बी.) का निर्माण किया गया है। हालांकि ये के.बी. बड़े हैं, लेकिन इनका कवरेज अधूरा है। कई व्यापक रूप से उपयोग किए जाने वाले के.बी. में इकाइयों (एंटीटी) के बीच अधिकांश संबंध (रिलेशन) गायब पाए जाते हैं। ऐसे के.बी. के कवरेज में सुधार के लिए अनुमान (इनफ्रेंस) का उपयोग किया जा सकता है और इससे उन्हें व्यावहारिक अनुप्रयोगों (जैसे खोज, संवाद और प्रश्न उत्तर) के लिए अधिक उपयुक्त बनाया जा सकता है। यह इनफ्रेंस प्रक्रिया को नॉलेज बेस कंप्लीशन (के.बी.सी.) कहा जाता है।

इस थीसिस में, हम मौजूदा के.बी.सी. सिस्टम का विश्लेषण करते हैं और विभिन्न नई के.बी.सी. विधियों का प्रस्ताव करते हैं। विशेष रूप से, हम के.बी.सी. प्रदर्शन में सुधार करने के लिए के.बी. की विभिन्न विशेषताओं का फायदा उठाते हैं - भाषा, संरचना और समय। हम के.बी.सी. मॉडल के मूल्यांकन का भी बड़े पैमाने पर अध्ययन करते हैं और निष्पक्ष मूल्यांकन नीतियों का प्रस्ताव रखते हैं।

सबसे पहले, हमने के.बी. संरचना के मॉडलिंग के विभिन्न पहलुओं और के.बी.सी. पर उनके प्रभाव का अध्ययन किया।

- के.बी.सी. मॉडल को उनके द्वारा एंटीटीज का प्रतिनिधित्व करने के तरीके के आधार पर वर्गीकृत किया जा सकता है। मैट्रिक्स फैक्टराइजेशन (एम.एफ.) मॉडल में प्रत्येक इकाई-जोड़ी (एंटीटी पेयर) के लिए एक वेक्टर परिभाषित होता है, जबकि टेंसर फैक्टराइजेशन (टी.एफ.) मॉडल प्रत्येक इकाई (एंटीटी) के लिए एक वेक्टर बनाए रखते हैं। हमने एम.एफ. और टी.एफ. की प्रभावशीलता की तुलना के.बी.सी. में प्रदर्शन से की। हमने माना कि एम.एफ. और टी.एफ. की प्रभावशीलता की तुलना करते समय आउट-ऑफ-वोकैबुलरी इकाई-जोड़े को विशेष देखभाल की आवश्यकता होती है। और हम एम.एफ. और टी.एफ. दृष्टिकोणों की तुलना करने के लिए पहले निष्पक्ष एकीकृत के.बी.सी. मूल्यांकन प्रोटोकॉल का प्रस्ताव करते हैं।
- के.बी.सी. मॉडल के हमारे विश्लेषण से पता चलता है कि वे अक्सर ऐसी भविष्यवाणी करते हैं जो असंगत होती हैं संबंध द्वारा आवश्यक प्रकार के साथ। उदाहरण के लिए, DistMult गलत भविष्यवाणी करता है - "अकीरा इसिदा (टाइप-पर्सन)" की, प्रश्न - "चीफ फिलिप्स (टाइप-फिल्म)", "क्षेत्र में रिलीज", ? के पूछे जाने पर।

- हम एक गैर-पर्यवेक्षित टाइपिंग गैजेट का प्रस्ताव करते हैं, जो के.बी.सी. मॉडल (जैसे DistMult और ComplEx) को बेहतर करता है टाइप-संगतता चेकर्स के साथ। एन्हांसड मॉडल (TypeDM और TypeComplEx) ने बेस मॉडल की तुलना में बेहतर के.बी.सी. प्रदर्शन दिखाया। आगे विश्लेषण से पता चला कि हमारे मॉडल अव्यक्त प्रकार की एंटीटीज का बेहतर प्रतिनिधित्व करते हैं और उनके एम्बेडिंग भी पर्यवेक्षित प्रकारों की भविष्यवाणी बेसलाइन मॉडल द्वारा सीखी गई एम्बेडिंग से बेहतर है।

उपरोक्त मॉडलों को लागू करते समय, हमने पाया कि नॉर्म के लिए प्रयुक्त मानदंड (L1, L2, या L3) या दूरियों को मापने, और मॉडलों को प्रशिक्षित करने के लिए उपयोग किए जाने वाले नकारात्मक नमूने की दर, केबीसी सटीकता के लिए महत्वपूर्ण परिणाम हो सकते हैं, यह कभी-कभी विभिन्न मॉडलों की एक दूसरे के साथ तुलना के पारंपरिक ज्ञान को उलट देते हैं। इनकी जांच के माध्यम से, हमने ComplEx के.बी. एम्बेडिंग का एक बहुत ही प्रतिस्पर्धी संस्करण लागू किया है।

इसके बाद, हम टेंपोरल के.बी. का अध्ययन करते हैं, जो एक तथ्य (s, r, o) को एक वैध समय के सेट (अक्सर एक पल या अंतराल) के साथ जोड़ते हैं। हम टी.के.बी.सी. के लिए TimePlex मॉडल प्रस्तावित करते हैं, यह मॉडल मुख्य रूप से लिंक-भविष्यवाणी और समय-अंतराल भविष्यवाणी के लिए लक्षित है। हमारी सर्वोत्तम जानकारी के अनुसार, यह पहला काम है जो समय अंतराल की भविष्यवाणी करता है जिसमें दिया गया तथ्य सामान्य मॉडल-स्वतंत्र तरीके से मान्य है। साथ ही, यह पहला काम है जो टी.के.बी.सी.के लिए समय के प्रति जागरूक मूल्यांकन रणनीति का प्रस्ताव करता है।

अंत में, हम ओपन के.बी. का अध्ययन करते हैं जहां स्कीमा-मुक्त टेक्स्ट के माध्यम से एंटीटीज और रिलेशंस का प्रतिनिधित्व किया जाता है। ओपन के.बी.सी. आमतौर पर एक अनुमान (इनफ्रेंस) नियम कोष का उपयोग करके किया जाता है।

हम एक एल्गोरिथम विकसित करते हैं – ज्ञान निर्देशित भाषाई पुनर्लेखन (KGLR) – जो सांख्यिकीय रूप से उत्पन्न ओपन केबी अनुमान नियमों के लिए स्वतंत्र सत्यापन प्रदान करता है। भाषा की अंतर्दृष्टि के प्रयोग से हम एक एल्गोरिथम विकसित करते हैं - ज्ञान निर्देशित भाषाई पुनर्लेखन (के.जी.एल.आर.) - जो सांख्यिकीय रूप से उत्पन्न ओपन के.बी. अनुमान (इनफ्रेंस) नियमों के लिए स्वतंत्र सत्यापन प्रदान करता है। उत्पन्न उच्च परिशुद्धता नियम कॉर्पस अंततः के.बी.सी. कार्य को बेहतर बनाने में मदद करता है।

# Contents

<b>Certificate</b>	<b>i</b>
<b>Acknowledgements</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>I Prologue</b>	<b>1</b>
<b>1 Introduction</b>	<b>3</b>
1.1 KB inference tasks . . . . .	5
1.1.1 Macro inference task . . . . .	5
1.1.2 Micro inference task . . . . .	8
1.2 Contributions . . . . .	9
1.3 Outline of the dissertation . . . . .	9
<b>2 Related Work</b>	<b>11</b>
2.1 Knowledge Bases . . . . .	11
2.2 Open vs. Closed KBs . . . . .	13
2.3 Incompleteness and inference . . . . .	14
2.3.1 Macro inference . . . . .	14
2.3.2 Micro inference . . . . .	17
<b>II Macro Inference</b>	<b>23</b>
<b>3 Macro inference preliminaries</b>	<b>25</b>
3.1 Problem formulation . . . . .	25
3.2 Evaluation protocol . . . . .	26
3.3 Popular datasets . . . . .	26
3.4 Tuple belief score and loss objectives . . . . .	28

3.4.1	Negative sampling . . . . .	29
3.4.2	Loss Objective . . . . .	29
3.5	Model Regularization . . . . .	31
<b>4</b>	<b>Matrix Factorization and Tensor Factorization</b>	<b>33</b>
4.1	Introduction . . . . .	33
4.2	Standard Evaluation Protocol . . . . .	34
4.3	Comparison Under Standard and Sanitized KBC Evaluation Protocol . . . . .	35
4.4	Toward a Robust MF Model . . . . .	39
4.4.1	MF with a Trained OOV Vector ( $F_{vecOOV}$ ) . . . . .	39
4.4.2	Generate OOV Entity Pair Vectors ( $F_{genEP}$ ) . . . . .	41
4.4.3	Using TF to Guide MF . . . . .	41
4.5	Discussion and Future Work . . . . .	44
4.6	Conclusion . . . . .	47
<b>5</b>	<b>Enhanced Knowledge Base Inference using unsupervised Type Induction</b>	<b>49</b>
5.1	Introduction . . . . .	49
5.2	TypeDM and TypeComplex . . . . .	51
5.3	Reflexivity aware model . . . . .	52
5.4	Experiments . . . . .	52
5.5	Analysis of Typed Embeddings . . . . .	55
5.6	Conclusion and Future Work . . . . .	57
<b>6</b>	<b>Knowledge Base Completion: Engineering Issues For Training</b>	<b>59</b>
6.1	Review of KBC score and loss functions . . . . .	59
6.1.1	TransE . . . . .	59
6.1.2	RotatE . . . . .	60
6.1.3	DistMult and Complex . . . . .	60
6.2	Additive vs. multiplicative loss with all negative triples . . . . .	61
6.2.1	Inner product . . . . .	61
6.2.2	$L_1$ distance . . . . .	62
6.2.3	The special case of $L_2$ distance . . . . .	63
6.3	Experimental study . . . . .	64
6.4	Discussion . . . . .	66
<b>7</b>	<b>Temporal Knowledge Base Completion</b>	<b>67</b>
7.1	Introduction . . . . .	67
7.2	Preliminaries and Prior Work . . . . .	68

7.2.1	Time-Agnostic KBC	68
7.2.2	Temporal KBC Problem Setup	69
7.2.3	TKBC Systems	69
7.2.4	Standard Evaluation Schemes	70
7.3	Evaluation Metrics and Filtering	70
7.3.1	Time Interval Prediction	70
7.3.2	Link Prediction	72
7.4	The Proposed TIMEPLEX Framework	73
7.4.1	TIMEPLEX Base Model	73
7.4.2	Relation Recurrence and Pair Scores	73
7.4.3	Training	77
7.4.4	Inference	77
7.5	Experiments	78
7.5.1	Datasets & Experimental Setup	78
7.5.2	Results and Observations	80
7.5.3	Diagnostics	82
7.6	Discussion	86
7.7	Conclusion	86
<b>III Micro-Inference</b>		<b>87</b>
<b>8</b>	<b>Improving the quality of inference rule-base for micro-inference</b>	<b>89</b>
8.1	Problem formulation	89
8.2	Micro inference	90
8.3	Knowledge-based Linguistic Rewrites	91
8.4	Experiments	93
8.5	Conclusions	97
<b>IV Epilogue</b>		<b>99</b>
<b>9</b>	<b>Conclusions and Future Directions</b>	<b>101</b>
<b>Bibliography</b>		<b>107</b>
<b>List of Publications</b>		<b>129</b>
<b>Biography</b>		<b>131</b>



# List of Figures

2.1	Knowledge graph fragments — (a) Regular KG (b) Temporal KG . . . . .	12
2.2	Macro Inference Task classified on the basis of features they use for inference: (a) Path over the graph structure, (b) Neighborhood of the query entity, (c) Atomic triple/fact. Note that the node highlighted in blue is query entity. . . . .	14
4.1	Synthetic dataset where, entities are represented as circles, color of the circle is its type. Relations are color coded links between entities. . . . .	45
5.1	TypeDM and TypeCompLex. . . . .	51
5.2	Projection of vectors representing entities belonging to frequent KB types- { <b>people</b> , <b>location</b> , <b>organisation</b> , <b>film</b> , <b>sports</b> }: a: TypeDM, $u_e$ ; b: TypeDM, $e_e$ ; c: TypeCompLex, $u_e$ ; d: DM, $e_e$ . . . . .	56
6.1	Influence on test MRR of number of negative samples used per positive training example. The peak is very mild and could be due to statistical variation. . . . .	65
7.1	This module attempts to capture fact recurrence. For tuple - ‘ <i>Presidential Elec- tion, heldIn, USA, 2016</i> ’, we first check if “ <i>heldIn</i> ” is a recurrent relation. If yes, we look for other facts of ‘ <i>Presidential Election, heldIn, USA</i> ’ occurring at different time points. We identify the min time-difference and compute its validity w.r.t. typical relation recurrence time. For non-recurrent relations only a bias term is learnt. . . . .	75
7.2	The data statistics are pre-computed for model training. We model time gaps as drawn from gaussian distributions. The image shows pre-training data statis- tics collection strategy for a specific relation pair ‘ <i>bornIn, graduatedFrom</i> ’. ‘ <i>Joe Biden</i> ’ was born in ‘ <i>1942</i> ’, while he graduated in 1965. While, ‘ <i>Michelle Obama</i> ’ was born in ‘ <i>1964</i> ’, while she graduated in 1985. Similar facts can be used to compute the average age of a person when he graduates (here it is 22). Such statistics are computed for all relation pairs. . . . .	76

7.3	TIMEPLEX learns soft time constraints between pairs of relations. All training tuple with time intervals are reduced to facts with start time. For a dummy fact: ‘ <i>Barack Obama, bornIn, USA, 1961</i> ’. We first look at other facts related to the subject ‘ <i>Barak Obama</i> ’ in dummy KB, like he graduated in ‘1983’ and became president in year ‘2009’. We compute contribution of each relation-pair score in the validity of candidate tuple ‘ <i>Barack Obama, bornIn, USA, 1961</i> ’, based on their signed time difference and typical time differences observed between two relations. Here Barack graduated at the age of 22, which is very close to the mean age of graduation we computed from dummy data (see Figure 7.2). So, this is a positive signal. Here, we describe this mechanism for subjects, objects are handled analogously. . . . .	76
7.4	$L_2$ distances (y-axis) between TIMEPLEX time embeddings increase with time gap (x-axis). . . . .	83
8.1	Sample rules verified (Y) and filtered (N) by our method. Rules 4, 5 were correctly and 6 wrongly filtered. . . . .	91
8.2	The precision and yield of inference rules after KGLR validation, and that of KB generated by inference using these rule-sets. Comparison with PPDB <sub>e</sub> is yield-controlled. . . . .	95
8.3	Ablation study of rule verification using KGLR rewrites on our devset of 600 CLEAN rules . . . . .	96
9.1	This figure demonstrates the benefits of additional temporal information for entity alignment and the multi-KBC task using dummy data. Following inferences would have been difficult if we ignored time information. (1) <i>Sachin Bansal</i> of KB1 and <i>Sachin Bansal</i> of KB2 are different entities as the two graduated at different times. (2) <i>Sachin Bansal</i> of KB1 and <i>Sachin Bansal</i> of KB3 cannot be the same as a person cannot graduate before he is born. (3) <i>Sachin Bansal</i> of KB2 and <i>Sachin Bansal</i> of KB3 are likely to be the same as his graduation age is 24 years which is close to the average age of graduation (as computed in Section 7.4.2). . . . .	103
9.2	Images of entities of similar type look alike. Also, they look different from pictures of other entity types. In this figure, we show Wikipedia title images of entities of type (a) person (b) countries (c) water bodies. . . . .	105

# List of Tables

1.1	Incompleteness of Freebase: The table lists the fraction of missing relations (commonly used) of entities (type PERSON) [Min et al., 2013, West et al., 2014]	4
2.1	Scoring functions of several KBC models. First 1-2 rows list translation models, row 3-8 lists bilinear models . Note that row 8 is a matrix factorization model, while row 1-7 are tensor factorization models. Larger value of $\phi^M$ implies more confidence in the validity of the triple. $\bullet$ denotes rotation . $\star$ denotes the complex conjugate. $\Re$ refers to the real part of the complex valued score returned by the ComplEx model. . . . .	16
2.2	Sample paraphrases extracted from a manually aligned parallel corpus. Italicized words are best paraphrases according to paraphrase probability estimates [Bannard and Callison-Burch, 2005]. . . . .	19
2.3	Sample textual patterns/path features (shown in italics), for rule extraction. . . .	20
3.1	Number of distinct entities, number of relation types, and size of train/test/valid folds of popular KBC benchmarks. . . . .	28
4.1	The first five rows compare 5 models on 4 datasets using the standard evaluation protocol - MRR, HITS@k (H@k) on test queries ( $s^*$ , $r^*$ , ?). The 6 <sup>th</sup> row shows F’s performance using our proposed KBC evaluation protocol. The last 2 rows report results of 2 most-frequent sanity-check baselines. . . . .	36
4.2	Original F with old evaluation protocol vs. F (trained OOV vector) with KBC evaluation protocol. Gold tuple in bold, and italics means that entity-pair was not seen during training. (a) Bill Gates is seen with one $o$ in training — not the gold answer; (b) Tina Fey is seen with two $os$ including the gold answer. . . . .	37
4.3	No. of distinct entities, no. of relations and entity pair OOV rate, i.e., percentage of tuples in test set, whose entity pairs (ep) weren’t seen while training. . . . .	38

4.4	Results on F model after explicitly modeling OOV vectors. OOV training outperforms other baselines, especially for NYT+FB. Results on FB15k-237 not reported, due to 100% OOV rate. . . . .	40
4.5	Performance segregated by OOV and non-OOV test queries on FB15k. F+C (RAL) performs best on both OOV and non-OOV. . . . .	40
4.6	Change in performance of ComplEx initialized with corresponding embeddings extracted from ComplEx+F (AS). . . . .	42
4.7	Performance (MRR, HITS@k/H@k) of joint models. AL = additive loss. AS = additive score. F+ComplEx combined with regularized additive loss (RAL) is the highest scorer as well as most robust across all datasets. . . . .	43
4.8	The fraction of entity-pairs occurring exactly once and exactly twice. NYT+FB has an unusual distribution. . . . .	44
4.9	Performance of TF model (CX) and MF model (F) and F+ComplEx (RAL) across various relations in Synthetic dataset. . . . .	46
5.1	Samples of top two DM predictions (having inconsistent types) on FB15K. TypeDM predicts entities of the correct type in top positions in the corresponding examples. . . . .	50
5.2	Sizes were approximately balanced between base and typed models (FB15K). . . . .	53
5.3	KBC performance for base, typed, and related formulations. Typed models outperform their base models across all datasets. Reflexive version of typed model further improves the performance on most datasets. Note that the mean performance reported in the table is comparable to the model scores reported in later chapters. . . . .	54
5.4	Interpretation of embeddings w.r.t. supervised types: cluster homogeneity H, completeness C, and type prediction F1 score. . . . .	57
6.1	Table (a) and (b) reports performance of popular KBC models along with their counterparts trained such that all entities contrast a positive fact while computing the loss: COMPLEX-V2 and SIMPLE-V2. Models are evaluated on five commonly used benchmark datasets for KBC. ComplEx-N3 is same as ComplEx except it adds new facts with inverse predicates in train set, use L3 regularization and 1-N training. All model parameters are of same range. COMPLEX-V2 shows near SOTA performance on all datasets. V2 models are indistinguishable or slightly worse. . . . .	64
6.2	Performance (MRR) improvement for RotatE (100-dim) by scoring against all entities while training (instead of negative sampling) . . . . .	65

7.1	<i>Jean</i> is the gold answer ( $o^*$ ) to the query $(s, r, ?, T)$ shown above. Rows are ranked system predictions, which may be seen with same $s$ and $r$ for different intervals (Column 2). Columns 3–4 show the filtering of existing methods (1:unfiltered, 0:filtered). Columns 5–8 (Method 3, our proposal) show the filtering for each time instant. The bottom row shows ranks of <i>Jean</i> as computed by different methods. Existing methods over- or under-estimate performance. Method 3 assigns <i>Jean</i> a rank of 3.25, which is the average of the filtered ranks $\{4, 4, 3, 2\}$ for each time instant in $[2000, 2003]$ . . . . .	72
7.2	Details of datasets used. . . . .	78
7.3	Number of parameters for each model. For HyTE we assume bucket size = 1 here. $\delta$ is the fraction of dimension to represent time in DA-Simple model. . . . .	79
7.4	Hyperparameters for training TIMEPLEX(base) model embeddings on various datasets, tuned on MRR for validation set. Temporal smoothing was found to help on ICEWS datasets, however it gave no improvement for interval datasets. We tuned the parameters in a staged manner - first we tune learning rate ( $lr$ ), regularization weight ( $r$ ), batch size( $b$ ), and temporal smoothing weight ( $ts$ ). We performed a random search in the following ranges: $lr \in [0.0001, 1.0]$ , $r \in [0.0001, 1.0]$ , $b \in [100, 5000]$ , and $ts \in [0.0001, 10.0]$ . The models were most sensitive to regularization weight and learning rate. After finding best values for these parameters, we tuned $\alpha$ , $\beta$ and $\gamma$ weights for each dataset, doing a grid search over the set $\{0.0, 2.0, 5.0, 7.0, 10.0\}$ . . . . .	80
7.5	Link prediction performance across four datasets. The last row reports results for TIMEPLEX(base) augmented with pair/recurrent features. . . . .	81
7.6	Link prediction performance of the best models using a filtering strategy that does not enumerate time points in an interval, and filters on an exact match instead. We find that while TIMEPLEX convincingly outperforms the previous SOTA TNT-Complex using this filtering strategy as well. . . . .	81
7.7	Time prediction performance using - TAC, gIOU, IOU and aeIOU . . . . .	82
7.8	High confidence (99%) relation orderings extracted from YAGO11k. . . . .	83
7.9	High confidence (99%) relation orderings extracted from WIKIDATA12k . . . . .	83
7.10	Ordering constraint violations among top predictions of various models (% of facts in test set). . . . .	84
7.11	Comparing time prediction performance of TIMEPLEX, TIMEPLEX(base) and TNTComplex. . . . .	85
7.12	Ablation study on Yago11k. Recurrence feature significantly help in link prediction while relation pair feature helps time-interval prediction. . . . .	86

8.1 Micro inference task problem formulation. . . . . 90