Research and Applications

The impact of learning Unified Medical Language System knowledge embeddings in relation extraction from biomedical texts

Maxwell A. Weinzierl , Ramon Maldonado, and Sanda M. Harabagiu

Human Language Technology Research Institute, Department of Computer Science, Erik Jonsson School of Engineering & Computer Science, University of Texas at Dallas, Richardson, Texas, USA

Corresponding Author: Maxwell A. Weinzierl, MS, Erik Jonsson School of Engineering & Computer Science, The University of Texas at Dallas, 800 W. Campbell Road, MS EC31, Richardson, TX 75080, USA; Maxwell.Weinzierl@utdallas.edu

Received 11 February 2020; Revised 23 July 2020; Editorial Decision 1 August 2020; Accepted 7 August 2020

ABSTRACT

Objective: We explored how knowledge embeddings (KEs) learned from the Unified Medical Language System (UMLS) Metathesaurus impact the quality of relation extraction on 2 diverse sets of biomedical texts.

Materials and Methods: Two forms of KEs were learned for concepts and relation types from the UMLS Metathesaurus, namely lexicalized knowledge embeddings (LKEs) and unlexicalized KEs. A knowledge embedding encoder (KEE) enabled learning either LKEs or unlexicalized KEs as well as neural models capable of producing LKEs for mentions of biomedical concepts in texts and relation types that are not encoded in the UMLS Metathesaurus. This allowed us to design the relation extraction with knowledge embeddings (REKE) system, which incorporates either LKEs or unlexicalized KEs produced for relation types of interest and their arguments.

Results: The incorporation of either LKEs or unlexicalized KE in REKE advances the state of the art in relation extraction on 2 relation extraction datasets: the 2010 i2b2/VA dataset and the 2013 Drug-Drug Interaction Extraction Challenge corpus. Moreover, the impact of LKEs is superior, achieving F1 scores of 78.2 and 82.0, respectively.

Discussion: REKE not only highlights the importance of incorporating knowledge encoded in the UMLS Metathesaurus in a novel way, through 2 possible forms of KEs, but it also showcases the subtleties of incorporating KEs in relation extraction systems.

Conclusions: Incorporating LKEs informed by the UMLS Metathesaurus in a relation extraction system operating on biomedical texts shows significant promise. We present the REKE system, which establishes new state-of-the-art results for relation extraction on 2 datasets when using LKEs.

Key words: medical informatics, unified medical language system, information extraction, deep learning

INTRODUCTION

Objective

Our objective was to incorporate knowledge embeddings (KEs) learned from Unified Medical Language System (UMLS) in a state-of-the-art relation extraction system operating on biomedical texts and to evaluate the impact of KEs on the results. KEs are representa-

tions of concepts and relations encoded in knowledge graphs, learned through neural models. Because the UMLS Metathesaurus is a knowledge graph, we considered it as a knowledge source for learning KEs. However, current methods for learning KEs from UMLS¹ ignore the fact that (1) biomedical concepts can be mentioned in various ways in biomedical texts; and (2) new relation

types between mentions of biomedical concepts, not encoded in UMLS, could be of interest. Therefore, we wondered if current KEs provide the best vehicle for incorporating UMLS knowledge in relation extraction. To answer this question, in addition to using KEs learned from the UMLS Metathesaurus with the method described in Maldonado et al,¹ we also learned lexicalized knowledge embeddings (LKEs) for UMLS concepts and relations. LKEs combine lexical knowledge with the semantics available in the UMLS Metathesaurus. When learning LKEs, we relied on (1) the preferred atom associated with each biomedical concept and (2) the name of relation types. We designed a knowledge embedding encoder (KEE) to learn either LKEs or nonlexicalized KEs from UMLS. In addition, the KEE learns neural models capable of generating LKEs for mentions of medical concepts and new relation types expressed in a biomedical text. These models were used in a neural architecture for relation extraction using knowledge embeddings (REKE), which can incorporate either LKEs or unlexicalized KEs into the state-of-the-art relation extraction system available from National Center for Biotechnology Information (NCBI) BlueBERT.² This enabled the primary objective of our study, which was to quantify the impact of LKEs or unlexicalized KEs on the quality of relation extraction results. We found that LKEs are more impactful than unlexicalized KEs, highlighting the role of combining lexical and semantic knowledge in relation extraction. Our approach obtains new state-of-the-art results for relation extraction. We make publicly available the LKEs and the unlexicalized KEs learned from the UMLS Metathesaurus, as well as the KEE and REKE models (<https://github.com/Supermaxman/umls-embeddings>). Acronyms introduced and utilized in this article are listed in Figure 1.

Background and Significance

The extraction of relations between concepts mentioned in biomedical texts has numerous applications that range from advancing basic science to improving clinical practice, as pointed out in Luo et al,³ including clinical trial screening, pharmacogenomics, diagnosis categorization, discovery of adverse drug reactions, biomolecular information extraction, and drug-drug interactions. An excellent review of methods developed for relation extraction in the past decade is presented in.³ The review shows that the relation extraction systems⁴⁻⁷ made use of UMLS knowledge by relying on the MetaMap program⁸ to identify biomedical concepts in texts. The UMLS concepts discovered in this way were used as features for relation classification. But in addition to UMLS concepts, these relation extraction systems used lexical, syntactic and semantic features along with multiple forms of contextual features. Recently, deep learning techniques have revolutionized the field of natural language processing (NLP), producing state-of-the-art results for relation extraction. These methods capture the deep lexical, syntactic, and semantic interactions of words in texts. Current state-of-the-art relation extraction is performed by the NCBI BlueBERT,² which takes advantage of fine-tuning BERT⁹ on biomedical texts. Other relation extraction systems operating on biomedical narratives, such as D'Souza et al¹⁰ and Rink et al,⁴ used relation extraction classification methods leverag-

ing hand-engineered features. D'Souza et al present an ensemble learning approach, while Rink et al, the winner of the original 2010 i2b2/VA challenge,¹¹ used a support vector machine for relation classification. He et al¹² and Luo et al¹³ report deep learning methods using convolutional neural networks to account for local syntactic information in texts. Zheng et al¹⁴ used a hierarchical recurrent neural network model, while Li et al¹⁵ employed graph convolutions, relying on syntactic information available from dependency parsing. However, the superior results of BlueBERT indicate that deep contextual information is more impactful than syntactic knowledge for relation extraction. However, BlueBERT does not use any of the biomedical knowledge encoded UMLS, which we hypothesize could generate improvements to relation extraction performance. A direct way of considering the knowledge encoded in the UMLS is provided by learning KEs from the UMLS Metathesaurus graph. However, incorporating KEs in relation extraction is not trivial, for 2 reasons: (1) KEs provide representations for concepts and relation types encoded in a knowledge graph, not for concept mentions encountered in texts or new relation types of interest (R1); and (2) to our knowledge, no neural relation extraction architecture leveraging KEs for concept mentions and new relation types has yet been designed (R2). The REKE system, presented in this article, is addressing both problems. To tackle problem R1, we have introduced the concept of LKE and made use of lexical knowledge to account for concept mentions as well as concepts encoded in a knowledge graph in the same way. LKEs are a new form of KEs, being one of the novel contributions of this article. LKEs use lexical knowledge not only for representing concepts, but also for relation types either encoded in a knowledge graph or newly defined. Alternatively, when considering unlexicalized KEs for incorporation in REKE, we addressed R1 by performing entity linking from a concept mention to a concept encoded in UMLS and retrieving the KE for the linked concept. Similarly, in order to incorporate the KE of a new relation type, we searched for the most similar relation type encoded in UMLS and retrieved its KE. Moreover, in the article, we present a second novel contribution, by designing the KEE that not only learns to generate LKEs or unlexicalized KEs for concepts and relations, but also learns neural models capable of generating LKEs for any concept mention or new relation type. To address problem R2 we have designed REKE that incorporates either LKEs or unlexicalized KEs in the BlueBERT system by scoring the plausibility of the relation types of interest between pairs of concept mentions. The results of REKE quantify the impact of both types of KEs on relation extraction, which was the focus of our research. By discovering that the LKEs improved relation extraction more than unlexicalized KEs, we pave the way for the need to incorporate LKEs learned from biomedical ontologies into NLP methods operating on biomedical texts. Moreover, learning LKEs as lexicalized representations of concepts and relations encoded in biomedical ontologies is distinct from embedding retrofitting, a graph-based learning technique that relies on lexical relational resources for producing higher-quality word embeddings, as introduced in Faruqui et al.,¹⁶ Alawad et al¹⁷ have applied embedding retrofitting informed by UMLS as inputs for information extraction from pathology reports.

KE: knowledge embedding LKE: lexicalized knowledge embedding KEE: knowledge embedding encoder
REKE: relation extraction using knowledge embeddings KEPE: knowledge embedding projection encoder

Figure 1. Acronyms used in the article.

MATERIALS AND METHODS

Data sources

The UMLS

In our work we used the UMLS Metathesaurus.¹⁸ Each concept from the Metathesaurus has (1) a concept unique identifier (CUI) and (2) a list of typed relations it shares with other concepts. [Supplementary Appendix SA](#) situates the Metathesaurus in the knowledge structure of UMLS and provides examples of concepts and relations encoded in it. Moreover, in the UMLS Metathesaurus lexical knowledge pertaining to concepts is encoded in the form of atoms; strings and terms. Atoms are building blocks of biomedical concept names. Given that a concept may be known through multiple names, a preferred atom is established. To efficiently learn KEs from UMLS, we have filtered out some concepts and relations, as is described in [Supplementary Appendix SA](#). For the research reported in this article, we considered a total of $C = 3\,210\,782$ concepts, spanned by $RT = 336$ relation types instantiated in $R = 12\,833\,112$ relations available from the 2019 UMLS Metathesaurus.

Relations between medical concepts annotated in the 2010 i2b2/VA dataset

Medical discharge summaries and progress notes were annotated by the organizers of the 2010 i2b2/VA relation extraction challenge¹¹ by identifying (1) mentions of 3 kinds of medical concepts: medical problems (PR), tests (TE), and treatments (TR); and (2) 8 types of relations between them, which are defined in [Table 1](#). The dataset is split into (1) a training set, in which the medical concepts and relations between them were annotated; and (2) a testing set, in which relation annotations were withheld, but concept annotations were available. The annotations between medical concepts in the publicly available 2010 i2b2/VA dataset (<https://portal.dbmi.hms.harvard.edu/projects/n2c2-nlp/>) split into (1) a training set, consisting of 170 documents, in which 16 532 concept mentions have been annotated along with 3120 relations between them; and (2) a testing set, consisting of 256 documents, in which 31 161 concept mentions were annotated such that relation extraction systems can be tested to discover 6293 relations. [Table 2](#) provides details of the annotated relation types in each subset we have used.

The DDI corpus: A dataset with annotations of pharmacological substances and drug-drug interactions

The Drug-Drug Interaction (DDI) corpus (<https://www.cs.york.ac.uk/semeval-2013/task9>) has been developed for the DDI Extraction 2013 Challenge,¹⁹ which aimed to automatically identify in texts 4 types of pharmacological substances as well as 4 types of drug-drug interactions, which are relations between pharmacological substances. Pharmacological substances that were annotated are (1) names of drugs (eg, “ergotamine”), (2) names of brands (eg, “ERGOMAR”), (3) groups of drugs (eg, “triacetyloleandomycin”), and (4) drug_n, which represent active substances not approved for human use (eg, toxins, pesticides). The definitions of the 4 types of drug-drug interactions are provided in [Table 3](#). The DDI corpus consists of texts from 2 different sources: (1) documents describing drug-drug interactions from the DrugBank dataset (DDI-DrugBank corpus) and (2) Medline abstracts (DDI-MedLine corpus). DDI-DrugBank contains 792 texts, while DDI-MedLine contains 233 abstracts. There were 15 756 pharmacological substances annotated in DDI-DrugBank and 2746 substances in DDI-MedLine. The DDI corpus was split into a training and a testing corpus, with statistical details of the relation type annotations provided in [Table 4](#).

Relation extraction using KEs learned from the UMLS Metathesaurus

Currently, state-of-the-art relation extraction results are obtained by the NCBI BlueBERT² system, which as illustrated on the left hand side of [Figure 2](#), produces a contextual embedding b for each sentence from a biomedical text where a pair of mentions of medical concepts may participate in a relation of one of the predefined types: $rt_1, rt_2, \dots, rt_K$. The deep contextual embedding b is used by a softmax function, which generates the probability distribution of the relation types that are likely to hold between the concept mentions. The extracted relation type r^i between concept mentions A and B has the highest probability.

To incorporate UMLS knowledge in the BlueBERT system we have first used a KEE, which learns to represent (1) the biomedical concept mentions A and B as UMLS-informed KEs e_A^m and e_B^m , respectively, and (2) each of the relation types of interest $rt_1, rt_2, \dots, rt_K$ as UMLS-informed relation type embeddings $re^1, re^2 \dots re^K$. These KEs were used to compute a vector of relation plausibility

Table 1. Relation types defined in the 2010 i2b2/VA dataset

Relation Type	Definition	Example
TrIP	A certain treatment has improved or cured a medical problem	“On April 2, 1994, she had several episodes of [<i>bradycardia</i>] _[PR] and asystole, which responded only to [<i>manual ventilation</i>] _[TR] .”
TrWP	A patient’s medical problem has deteriorated or worsened because of or in spite of a treatment being administered	“He had been noting night sweats, increasing fatigue, anorexia, and [<i>dyspnea</i>] _[PR] , which were not particularly improved by [<i>increased transfusions</i>] _[TR] or alterations of hydroxy urea.”
TrCP	A treatment caused a medical problem	“He has gained [<i>30 lb</i>] _[PR] since [<i>the tube feed</i>] _[TR] was initiated (70 lb to 100 lb) during this admission.”
TrAP	A treatment administered for a medical problem	“Pt was [<i>Levo</i>] _[TR] 250mg x 3 days for UTI ([<i>pyuria</i>] _[PR] on admission).”
TrNAP	The administration of a treatment was avoided because of a medical problem	“However, due to the episode of [<i>bleeding</i>] _[PR] on postoperative day #2, [<i>the Coumadin</i>] _[TR] was held for 2 consecutive days, and then restarted on postoperative day #4.”
TeRP	A test has revealed some medical problem	“[<i>Reflexes</i>] _[TE] were trace in upper extremities and [<i>absent in the knees</i>] _[PR] with downgoing toes bilaterally.”
TeCP	A test was performed to investigate a medical problem	“In the ICU the pt remained [<i>stuporous</i>] _[PR] , responsive to [<i>painful stimuli</i>] _[TE] .”
PIP	Two problems are related to each other	“She was initially noted to have [<i>a slow ventricular response</i>] _[PR] in [<i>atrial fibrillation</i>] _[PR] , but this gradually improved over time.”

scores $s_{A,B}^- = [s_{A,B}^1, s_{A,B}^2, \dots, s_{A,B}^K]$, which, as illustrated in Figure 2, is concatenated to the contextual embedding h , before it is used by the softmax function. The plausibility of relations between concepts is considered by KE models when learning to generate relation type embeddings based on a knowledge graph. Knowledge embedding models, including those used in the KEE, evaluate the relation plausibility using a scoring function. In our previous work,¹ we experimented with several KE models and found that the best UMLS KEs were obtained when using the scoring function of the TransD model.²⁰ This motivates the usage of the TransD scoring function for estimating the plausibility of each relation type of interest rt_i through a score $s_{A,B}^i$ computed as

$$s_{A,B}^i = -\|(\mathbf{I} + \mathbf{r}e^i \times e_A^{mp\top}) \times e_A^m + \mathbf{r}e^i - (\mathbf{I} + \mathbf{r}e^i \times e_B^{mp\top}) \times e_B^m\|^2 \quad (1)$$

where $\mathbf{I}$ is an identity matrix and $e_A^{mp\top}$ and $e_B^{mp\top}$ are the transposed projection embeddings of the concept mention A and of concept mention B , respectively, learned in the KEE.

The KE encoder informed by the UMLS knowledge graph

The KEE learns either LKEs or nonlexicalized KEs identical to those reported in Maldonado et al.¹ In the past years, several models for learning KEs were proposed, such as RESCAL,²¹ which learns KEs using matrix factorization; TransE,²² which produces translation-based KEs; TransD,²⁰ which extended TransE by dynamically projecting the concept and relation embeddings in various spaces; DistMult,²³ which simplifies RESCAL by using a diagonal matrix; and more recently KBGAN,²⁴ which uses adversarial learning to gener-

ate KEs. In Maldonado et al,¹ we showed how our extension of the KBGAN method outperformed all other methods when learning UMLS KEs, which motivates us to reuse the same adversarial framework in the KEE.

Learning to generate LKEs or unlexicalized KEs in the KEE relies on a generative adversarial network (GAN), which formulates the problem of learning as 1 of 2 competing agents: a generator and a discriminator, as introduced in.²⁵ Metaphorically, the generator can be thought as acting like a team of counterfeiters, trying to produce fake currency and using it without detection. The discriminator can be thought of acting like the police, trying to detect the counterfeit currency. Competition in this game enabled by the GAN drives both teams to improve their methods until the counterfeiters are indistinguishable from the genuine articles. In the KEE, the discriminator learns to score the relation plausibility between a pair of UMLS concepts while the generator tries to fool the discriminator by generating plausible, yet incorrect relations. To accomplish this goal, given any UMLS concept c^i , the generator selects a set of negative examples of relations connecting concept c^i to other concepts from UMLS, computes a probability distribution over this set and samples a single relation to be proposed to the discriminator, as in Maldonado et al.¹ It operates similarly when considering each relation of type rt_{UMLS} between UMLS concepts. The discriminator sends a reward to the generator, G -reward, in response to comparing the negative example to a true relation encoded in the UMLS Metathesaurus, providing an excellent framework for learning UMLS KEs that benefits from good negative examples in addition to the abundance of positive examples encoded in UMLS. Eventually, the discriminator learns not to be fooled by the generator anymore.

The KEE architecture, illustrated in Figure 3 combines the modules for learning KEs, shown at the top of the figure, with the GAN framework, shown at the bottom. LKEs are learned in the encoders shown on the left side whereas unlexicalized KEs are learned by the encoders from the right side of the figure. The LKE of each concept c^i encoded in the UMLS Metathesaurus is learned in the LKE, whereas the LKE of each relation type rt_{UMLS} is learned in the relation type knowledge encoder. Both encoders have the same neural architecture. We used the insight that names of biomedical concepts or relation types have meanings that are informed by lexical knowledge (eg, the various tokens or pieces of the name). The name meaning is derived when combining affix/suffix information with etymological information (typically of Latin or Greek origin), eg,

Table 2. Relation counts in the 2010 i2b2/VA dataset

Relation type	Train	Dev	Test	All
TrIP	49	2	152	203
TrWP	24	0	109	133
TrCP	171	13	342	526
TrAP	800	85	1732	2617
TrNAP	52	10	112	174
TeRP	903	90	2060	3053
TeCP	149	17	338	504
PIP	659	96	1448	2203
Total	2807	313	6293	9413

Table 3. Relation types defined in the Drug-Drug Interaction corpus

Relation type	Definition	Example
Mechanism	Indicates drug-drug interactions that are described by their pharmacokinetic mechanism	"The bioavailability of SKELID is decreased 80% by calcium, when calcium and SKELID are administered at the same time, and 60% by some aluminum –or [magnesium] _[DRUG] – containing antacids, when administered 1 hour before [SKELID] _[BRAND] ."
Effect	Indicates drug-drug interactions describing an effect or a pharmacodynamic mechanism	"The induction dose requirements of [DIPRIVAN] _[BRAND] injectable emulsion may be reduced in patients with intramuscular or intravenous premedication, particularly with narcotics (eg, morphine, meperidine, and [fentanyl] _[DRUG] , etc.)"
Advise	This relation type is used when a recommendation or advice regarding the drug interaction is given	"Patients receiving [antibiotics] _[GROUP] and sulfonamides generally should not be treated with [ganglion blockers] _[GROUP] ."
Int	This relation type is used where a drug-drug interaction appears in the text without providing any additional information	"Use with anticholinergics: because of their mechanism of action, [cholinesterase inhibitors] _[GROUP] have the potential to interfere with the activity of [anticholinergic medications] _[GROUP] ."

We used the development (Dev) portion from Peng et al,² sampled from the training data, to estimate results of learning to extract relations.

Figure 2. Architecture for relation extraction from biomedical texts that incorporates knowledge embeddings learned from the Unified Medical Language System (UMLS) Metathesaurus. NCBI: National Center for Biotechnology Information.

the name “atherosclerosis” contains the prefix “ather,” which means fatty deposit in Greek, which is combined with “sclerosis,” derived also from Greek to indicate the process of hardening of arterial walls. To capture the meaning of names (of concepts or relation types), we performed word-piece tokenization (WPT)⁹ on the preferred atom p^i of each biomedical concept c^i encoded in UMLS, or rt^i , the name of the relation type r_{UMLS}^i . WPT is completely data-driven and guaranteed to generate a deterministic segmentation for any possible sequence of characters of each atom or name. This is especially important for biomedical text in which rare words, which are not available in common word embedding collections, are prevalent. WPT splits p^i or rt^i , respectively, into tokens, t_k^i , which are word pieces often shared across concept names or relation type names. [Supplementary Appendix SE](#) provides an example that highlights the advantage of tokenizing concept preferred atoms. In each encoder, the word-piece tokens were further processed by the language model available from the Bidirectional Encoder Representation for Transformers (BERT),⁹ to capture deep lexicalized context. For each concept c^i , all the tokens t_k^i from its preferred atom p^i were

transformed into the contextualized token embeddings a_k^i . The embeddings a_k^i informed the generation of a single encoding s^i for the name of the concept c^i , produced through a span encoder, implemented with bidirectional long short-term memory.²⁶ For each relation type r_{UMLS}^i , a single encoding rs^i for the name of the relation type was produced in the same way. Finally, the LKEs are learned from s^i and rs^i using a pair of KEPEs, implemented as a fully connected layer.

Because in the GAN framework, the generator and the discriminator use the UMLS semantic knowledge differently in their adversarial game, a different KEPE had to be used by the generator, denoted as G-KEPE, than the one used by the discriminator, denoted as D-KEPE. The G-KEPE projects s^i into the LKE le_g^i whereas D-KEPE projects s^i into the LKE le_d^i , both corresponding to concept c^i . In the same way, G-KEPE projects rs^i into the LKE rle_g^i whereas D-KEPE projects rs^i into the LKE rle_d^i , both corresponding to relation type r_{UMLS}^i . All LKEs are real-valued vectors of dimension dim .

For every UMLS concept c^i and any relation type r_{UMLS}^i , the generator selects a set of relations of type r_{UMLS}^i connecting concept c^i

Figure 3. Knowledge embedding encoder that relies on an adversarial learning framework. UMLS: Unified Medical Language System.

to other concepts from UMLS, denoted as R_G^i . Figure 3 illustrates several examples of relations from R_G^i , connecting the concept $c^i = \text{Anxiety}$ to other UMLS concepts, all of the type $r_{UMLS}^i = \text{May_Treat}$. The generator computes a probability distribution over R_G^i using the scoring function of the DistMult model,²³ known to be effective in estimating relation likelihood distributions. Given the concept c^i represented by the LKE le_g^i , the relations of type r_{UMLS}^i , represented by the LKE $rl_e_g^i$ and a concept c^j represented by the LKE le_g^j , the DistMult scoring function uses a point-wise multiplications between le_g^i , $rl_e_g^i$ and le_g^j :

$$f_g(c^i, r_{UMLS}^i, c^j) = \sum_{l=1}^{dim} le_g^{il} \times rl_e_g^{il} \times le_g^{jl} \quad (2)$$

This scoring function is used to compute the generator's probability distribution, P_g , over R_G^i as:

$$P_g(c^i, r_{UMLS}^i, c^k) = \frac{\exp(\sigma(f_g(c^i, r_{UMLS}^i, c^k)))}{\sum_{m=1}^n \exp(\sigma(f_g(c^i, r_{UMLS}^i, c^m)))} \quad (3)$$

where σ is the sigmoid function and n is the cardinality of R_G^i .

The generator samples from R_G^i based on the distribution P_g , a relation and proposes it to the discriminator. For example, given the distribution of *May_Treat* relations involving concept $c^i = \text{Anxiety}$ illustrated in Figure 3, the relation instance $r_{sample} = \text{May_Treat}(\text{Caffeine}, \text{Anxiety})$ is sampled by the generator and presented to the discriminator. The discriminator compares the sampled relation to a true relation encoded in UMLS, eg, $r_{encode} = \text{May_Treat}(\text{Zolazepam}, \text{Anxiety})$ by scoring both r_{sample} and r_{encode} . As reported in,¹ when experimenting with several Knowledge Embedding Models for learning UMLS KEs in the GAN framework, the best results were obtained when using DistMult for the generator and TransD for the discriminator. This motivated the same selection of scoring functions in KEE. The scoring function employed by the TransD model²⁰ learns 2 KEs for each concept and relation. Specifically, for any concept c^i or relation type r_{UMLS}^i , We provided to TransD 2 pairs of LKEs: $[le_d^{im}, le_d^{ip}]$ and $[rl_e_d^{im}, rl_e_d^{ip}]$. To obtain these pairs of LKEs, we have split in 2 equal-sized vectors the LKEs le_d^i and $rl_e_d^i$. This enables the discriminator to use the TransD scoring function:

$$f_d(c^i, r_{UMLS}^i, c^j) = -\| (I + rl_e_d^{ip} \times le_d^{ip\top}) \times le_d^{im} + rl_e_d^{im} - (I + rl_e_d^{ip} \times le_d^{ip\top}) \times le_d^{im} \|^2 \quad (4)$$

where I is an identity matrix. It is to be noted that equations 1 and 4 use the same scoring function, but for different purposes, and have different arguments of the function. The scoring function f_d is used by the discriminator to learn to minimize the *marginal loss* between the plausibility of relation r_{sample} and relation r_{encode} , which is defined as:

$$L_d = \sum_{\mathcal{R}} \|\lambda - f_d(c^i, r_{UMLS}^i, c^j) + f_d(c^i, r_{UMLS}^i, c^k)\|_+ \quad (5)$$

where λ is a margin parameter, while $\mathcal{R}$ is the set of all relations encoded in the UMLS Metathesaurus. The only way in which the discriminator can minimize its loss is by updating the LKEs it uses. The gradients of L_d are backpropagated to the Lexicalized Knowledge Encoder and to the Relation Type Knowledge Encoder, to update the parameters of the span encoder and of D-KEPE, which leads to updating the LKEs le_d^i and $rl_e_d^i$ respectively. The discriminator also sends a reward to the generator, *G-reward*, quantified by $f_d(c^i, r_{UMLS}^i, c^j)$. The generator uses its reward to learn to propose

more plausible relations of type r_{UMLS}^i , by minimizing its loss function, as in:²⁷

$$L_g = \sum_{(c^i, r_{UMLS}^i, c^k) \sim P_g} -\log(P_g(c^i, r_{UMLS}^i, c^k)) \times f_d(c^i, r_{UMLS}^i, c^k) \quad (6)$$

where (c^i, r_{UMLS}^i, c^k) represents r_{sample} selected by the generator from its distribution, P_g . To minimize its loss, the generator updates the LKEs it uses by backpropagating the gradients of L_g to the LKE and to the Relation Type Knowledge Encoder, updating the parameters of the span encoder and of G-KEPE, which leads to updating the LKEs le_g^i and $rl_e_g^i$ respectively. This explains why we need separate G-KEPE and D-KEPE.

By minimizing their respective losses, the generator and discriminator can eventually reach a point of equilibrium, as detailed in.²⁵ However, in practice, training ceases when the discriminator is able to identify the most plausible relation proposed by the generator as a relation encoded in the UMLS Metathesaurus with 99% accuracy. This convergence produces the final LKEs, updated for the final usage of the discriminator, namely le_d^i is the LKE for concept c^i , and $rl_e_d^i$ is the LKE for relation type r_{UMLS}^i .

When lexical knowledge is omitted, the KEE operates by using the concept knowledge encoders for each concept c^i and the relation knowledge encoder for each relation type r_{UMLS}^i , which as seen in Figure 3, have identical architectures. In this case, the KEs for concepts and relation types are initialized randomly, and then a simple index lookup provides the KEs that are updated by the generator and the discriminator. The resulting KEs, after the discriminator and the generator converge, are those updated for the final usage of the discriminator, namely e_d^i as the KE for concept c^i , and re_d^i as the KE learned for relation type r_{UMLS}^i . It is important to note that in addition to LKEs and unlexicalized KEs, the KEE architecture also learns (1) a neural model for lexicalized knowledge encoders of concepts and (2) a neural model for lexicalized knowledge encoders of relation types, as a consequence of the updating of the LKEs by the GAN framework.

Exemplifying relation extraction informed by UMLS KEs

The KEE provides LKEs or unlexicalized KEs only for concepts or relation types encoded in UMLS, not for the various mentions of concepts in texts or for new relation types. However, the KEE also provides neural models that are learned, capable to generate LKE for concept mentions or new relation types of interest. To showcase how this can be accomplished, we exemplify the operation of REKE, providing details which were omitted in Figure 2. In Figure 4, we illustrate a short text that has 2 concept mentions, denoted as m_A and m_B . For each concept mention we generate either (1) the LKE (eg, le_A^m and le_B^m) or (2) the unlexicalized KE (e_A^m and e_B^m). To generate le_A^m or le_B^m , we used the neural model of the lexicalized knowledge encoder, learned in KEE, operating on m_A or m_B in the same way as it operates on names of concepts. However, generating e_A^m and e_B^m is more difficult because we need to identify the UMLS concepts referred by the mentions m_A and m_B . This is accomplished by the application of the ScispaCy entity linker,²⁸ which identifies the UMLS concept c^A mentioned in text by m_A and the UMLS concept c^B mentioned in text by m_B . Figure 4 illustrates these concepts (through their CUIs) retrieved for the 2 concept mentions. Then, the unlexicalized KE e^A attributed to m_A is the KE of concept c^A , already generated by the KEE. In the same way, e^B is attributed to m_B . In Figure 2, the KEs e_A^m and e_B^m

Figure 4. Example of relation extraction using the relation extraction incorporating knowledge embeddings system on a biomedical text. NCBI: National Center for Biotechnology Information.

represent either le_A^m and le_B^m or e^A and e^B , whereas in Figure 4, we showcase both possible forms of KEs used for representing concept mentions.

Relation types of interest rt_i are represented in REKE either through LKEs or unlexicalized KEs. Examples of relation types of interest are provided in Table 1. When LKEs were considered, the relation type embeddings are generated by the neural model of the relation type knowledge encoder from the KEE, using the definition of the relation type of interest as input. Table 1 provides examples of definitions for relation types of interest on 2 datasets. When unlexicalized KEs were considered, we first needed to find rt_{UMLS}^i , the relation type encoded in UMLS most similar to each new relation type rt_i . This was performed by hand. Given rt_{UMLS}^i , its unlexicalized KE available from the KEE was retrieved. Supplementary Appendix SC provides additional details. Either LKEs or unlexicalized KEs for each relation types of interest rt_i are illustrated as re^i in Figures 2 and 4.

When considering to extract a relation between a pair of concepts mentioned in text, BlueBERT replaces the concept mentions with their type, eg, *gentamycin* is replaced by TR, as it is a treatment, and sepsis risk factor by PR, as it is a medical problem, as detailed in Figure 4. By performing this substitution, BlueBERT simplifies the learning of the contextual information characterizing relations. Because a sentence may have multiple pairs of concept

mentions, each of them potentially participating in a relation, BlueBERT casts the problem of relation extraction as a multiclass classification problem. In this process, BlueBERT generates an embedding h for each pair of concepts mentioned in a sentence, encoding the deep semantic context in which the concept mentions and the relations between them are expressed. By concatenating h with a vector $s_{A,B}$, which assembles the plausibility scores of relations of type rt_i , connecting concept mentions as m_A and m_B , REKE takes advantage of knowledge encoded in UMLS. Each element of $s_{A,B}$ is computed as in equation 1, which amounts to $s_{A,B}^i = f_d(m_A, rt_i, m_B)$, where f_d is the scoring function of TransD, defined in equation 4. Because the function f_d operates on a triple of KEs, it allows us to use either the LKEs or the unlexicalized KEs for concept mentions and relation types to compute the plausibility scores, as illustrated in Figure 4.

To learn how to extract relations, REKE uses training data available from the corpora described in the Data Source section. The training data from each corpus annotated with relations between concept mentions allowed us to learn the embeddings re^i along with the parameters of REKE to minimize a joint loss function: $\ell_R = \ell_C + \gamma \ell_P$; where ℓ_C is the relation classification loss, ℓ_P is the relation type plausibility loss, and γ is a weight parameter. The relation classification loss ℓ_C is defined as the categorical cross-entropy loss:

Table 4. Relation counts in the Drug-Drug Interaction corpus

Relation Type	Train	Dev	Test	All
Mechanism	950	373	302	1625
Effect	1313	396	360	2069
Advise	636	193	221	1050
Int	146	42	96	284
Total	3045	1004	979	5028

We used the development (Dev) portion from Peng et al.,² sampled from the training data, to estimate results of learning to extract relations.

Table 5. Relation extraction performance on the 2010 i2b2/VA dataset

System	Precision	Recall	F1
D'Souza et al. ¹⁰	72.9	66.7	69.6
He et al. ¹²	73.1	66.7	69.7
Luo et al. ¹³	68.7	73.7	71.1
Rink et al. ⁴	72.0	75.3	73.7
NCBI BlueBERT ²	78.4	74.4	76.4
REKE incorporating unlexicalized KEs	79.3	75.2	77.2
REKE incorporating LKEs	79.5	76.8	78.2

KE: knowledge embedding; LKE: lexicalized knowledge embedding; NCBI: National Center for Biotechnology Information; REKE: relation extraction incorporating knowledge embeddings.

Table 6. Relation extraction performance on the Drug-Drug Interaction corpus

System	Precision	Recall	F1
Zhang et al. ¹⁴	74.1	71.8	72.9
Li et al. ¹⁵	77.6	75.7	76.6
NCBI BlueBERT ²	79.3	80.5	79.9
REKE incorporating unlexicalized KEs	80.3	79.8	80.0
REKE incorporating LKEs	83.2	80.7	82.0

KE: knowledge embedding; LKE: lexicalized knowledge embedding; NCBI: National Center for Biotechnology Information; REKE: relation extraction incorporating knowledge embeddings.

$$\mathcal{Q}_C = - \sum_{r^i \in D} \sum_{k=1}^K y_k^{ij} \log(P(r_k^{ij})) \quad (7)$$

where $y_k^{ij} \in \{0, 1\}$ indicates whether relation type k has been annotated in the training corpus between any concept mentions m^i and m^j while a potential relation of type k between m^i and m^j is represented as r_k^{ij} and D represents all the possible relations between pairs of concept mentions in the same sentence. The relation type plausibility loss ℓ_P is a marginal loss that mirrors equation 4:

$$\ell_P = \sum_{(m^i, r^i, m^j) \in D} \sum_{r^k \neq r^i} \|\lambda - f_d(m^i, r^i, m^j) + f_d(m^i, r^k, m^j)\|_+ \quad (8)$$

where the margin loss is computed between the plausibility of the relation type r^i of a relation instance $r^i = (m^i, r^i, m^j)$ available from the training corpus and the plausibility of every other false relation type $r^k \neq r^i$. The relation type plausibility loss informs the learning of relation type embeddings re^k through backpropagation

Table 7. Fine-grained relation extraction performance on the 2010 i2b2/VA dataset

Relation Type	Precision	Recall	F1
Total TrP	76.3	75.9	76.1
TrWP	68.6	32.1	43.8
TrIP	74.5	53.9	62.6
TrNAP	66.0	62.5	64.2
TrCP	64.6	69.3	66.9
TrAP	79.6	82.8	81.2
Total TeP	85.9	83.0	84.4
TeCP	72.2	59.2	65.0
TeRP	87.7	86.9	87.3
PIP	74.5	68.1	71.1
Total	79.5	76.8	78.2

Table 8. Fine-grained relation extraction performance on the Drug-Drug Interaction corpus

Relation Type	Precision	Recall	F1
Int	76.4	43.8	55.6
Advise	87.4	80.7	87.6
Mechanism	87.2	83.4	85.3
Effect	78.9	83.9	81.3
Total	83.2	80.7	82.0

by enforcing the constraint that annotated relation instances are more plausible than invalid relation instances while being consistent with knowledge encoded in the UMLS Metathesaurus.

RESULTS

REKE was evaluated in 2 domains: (1) clinical notes from the 2010 i2b2/VA dataset and (2) scientific articles from the DDI corpus. The results of relation extraction on both datasets are presented in Tables 5 and 6. Following the evaluation protocol used by the creators of each of the datasets,^{11,19} the standard metrics of micro precision, recall, and F1 scores were utilized to compare the performance of REKE against prior work. To compare the results of REKE when it incorporates LKEs or unlexicalized KEs with current state-of-the-art relation extraction results, obtained by the NCBI BlueBERT,² we trained REKE with the same hyper-parameters and training schedule set by BlueBERT, detailed in the Supplementary Appendix SD.

The results listed in Table 5 indicate that our intuition is correct, as incorporating in REKE either LKEs or unlexicalized KEs learned from the UMLS produces superior results, increasing both precision and recall on the 2010 i2b2/VA dataset. When REKE uses LKEs, it produces new state-of-the-art results. Table 6 shows that REKE using LKEs improved the state of the art in both precision and recall, while REKE using unlexicalized KEs generates superior results only for precision and F1-scores on the DDI dataset. This indicates that the impact of LKEs on relation extraction can be quantified into an increase of 2.4% of the F1 score on the 2010 i2b2/VA dataset and an increase of 2.6% F1 score on DDI corpus over current state-of-the-art results. We also evaluated the performance of REKE on fine-grained relation extraction when incorporating LKEs, as presented in Tables 7 and 8. The analysis of these results indicates that the REKE system performs best on relation types which have more specific meanings, such as TrP, TeP, advise, and mechanism types, and perform worse on more general relation types, such as PIP and Int, suggesting that knowledge

encoded in UMLS leads to superior improvement for the extraction of relation types that have more specific semantics.

DISCUSSION

In order to further assess the effect of incorporating KEs in relation extraction, we conducted an ablation analysis considering several con-

Table 9. Ablation study of relation extraction performed on the 2010 i2b2/VA dataset

System	Precision	Recall	F1
BlueBERT	78.4	74.4	76.4
REKE-KE	79.3	75.2	77.2
REKE-LKE-Link	79.4	75.7	77.5
REKE-LKE-Classif-Loss	78.8	76.5	77.6
REKE-LKE	79.5	76.8	78.2

KE: knowledge embedding; LKE: lexicalized knowledge embedding; REKE: relation extraction incorporating knowledge embeddings.

figurations of REKE. First, we considered the incorporation of the unlexicalized KEs in REKE, performed as illustrated in Figure 4, denoted that configuration as REKE-KE, while REKE-LKE denotes the incorporation of LKEs as in Figure 4. When considering unlexicalized KEs, we used an additional option, such that after performing the entity linking of a concept mention m_A to UMLS concept c^A , produced by the ScispaCy entity linker,²⁸ instead of performing a lookup for the unlexicalized KE of c^A , we provided c^A to the neural model of the lexicalized knowledge encoder learned by KEE, and thus generated a new LKE for m_A . We did the same for m_B . We denoted this configuration of REKE as REKE-LKE-Link. Finally, we created a version of REKE in which instead of minimizing a joint loss function, we ignored the relation type plausibility loss, and denoted this configuration as REKE-LKE-Classif-Loss, indicating the only loss function it uses. The ablation study was performed on the 2010 i2b2/VA dataset. Table 3 lists the results of each of these configurations of REKE, as well as the results of the BlueBERT system.

The results listed in Table 9 indicate that regardless whether LKE or unlexicalized KEs were incorporated in the BlueBERT

Figure 5. Examples of relations annotated or identified by the relation extraction incorporating knowledge embeddings (REKE) or the National Center for Biotechnology Information BlueBERT systems in the test set of (A) the 2010 i2b2/VA dataset and (B) the Drug-Drug Interaction (DDI) corpus. ICU: intensive care unit; KE: knowledge embedding.

Error Analysis Class	Text Example	Relation Identified by REKE	Relation Annotated	Discussion
A1	Given that [the stroke] _[PR] occurred while taking [aspirin] _[TR] and plavix, and given the severe akinesis of the left ventricle, we decided to initiate anticoagulation with warfarin to help prevent future strokes.	TrCP	TrWP	It is difficult to infer from the text whether aspirin caused the stroke or whether it only worsened the medical problems of the patient. Additional knowledge, acquired from scientific articles may contribute to the identification of the correct relation type in this example.
A2	[Chronic renal insufficiency] _[PR] with [baseline creatinine] _[TE] 1.8-2.	TeRP	TeCP	The scarcity of the sentential context leads to the inference of multiple possible relation types. Only the values of the baseline creatine would indicate the correct relation type. However, LKEs do not capture the meaning of numerical values.
B1	[Thalidomide] _[DRUG] has been reported to enhance the sedative activity of barbiturates, [alcohol] _[DRUG] , chlorpromazine, and reserpine.	effect	no relation	The context of the sentence indicates the effect of Thalidomide on the sedative activity of alcohol . Hence we believe that the annotation is incorrect.
B2	Patients treated with [proton pump inhibitors] _[GROUP] and [warfarin] _[DRUG] concomitantly should be monitored for increases in inr and prothrombin time.	effect	advise	REKE confused the effect and advise relation types because it likely believed that "increases in inr and prothrombin time" was an effect of the Interaction marked by the coordinating "and". REKE did not recognize from context that this was an advice, because it ignored the phrase "should be monitored for". This is explained by the relatively small Training set, which did not allow REKE to learn such contextual cues.

Figure 6. Errors produced by the relation extraction incorporating knowledge embeddings (REKE) system when incorporating lexicalized knowledge embeddings. The error classes A1 or A2 correspond to the results on the 2010 i2b2/VA dataset, whereas B1 or B2 pertain to results on the Drug-Drug Interaction (DDI) corpus.

system, performance was improved, while REKE-LKE obtained the best results. The improvement of REKE-LKE and REKE-LKE-Classif-Loss over REKE-KE and REKE-LKE-Link indicates that bypassing the difficult task of entity linking in the UMLS Metathesaurus and generating LKEs for concept mentions directly through the usage of the neural model of the lexicalized knowledge encoder learned in KEE leads to the best results. Entity linking is a difficult task which often produces incorrect results.²⁸ To exemplify how entity linking can hurt the incorporation of KEs in relation extraction systems, we consider the example provided in Figure 4. For this example, the entity linker incorrectly identifies the concept referred by “sepsis risk factor” as the having CUI=C0035648, corresponding to the concept “risk factor.” By considering the UMLS KE for “risk factor” instead of “sepsis risk factor,” the REKE system loses valuable information. Furthermore, the ablation study shows that when using only the relation classification loss, as in the REKE-LKE-Classif-Loss configuration, the quality of relation extraction decreases, compared with the results of REKE-LKE. This observa-

tion indicates that the usage of the relation type plausibility loss improves the quality of relation extraction. The plausibility loss can be interpreted as a means of learning to determine if a potential relation type is consistent with the knowledge encoded in the UMLS Metathesaurus, and thus improves performance when such consistency is accounted for.

A qualitative analysis of the results obtained by the REKE-LKE explains why it generates fewer errors than the NCBI BlueBERT system. Figure 5 showcases the operation of REKE-LKE on examples from the test sets of (1) the 2010 i2b2/VA dataset and (2) the DDI corpus. The results of both relation extraction systems are contrasted with the gold-standard annotations and are analyzed qualitatively. However, REKE-LKE also incorrectly identifies some relations. In Figure 6 we present an error analysis of the operation of REKE-LKE in both datasets by illustrating the 2 main classes of errors that we have found. We also perform a qualitative analysis of REKE using T-SNE dimensionality reduction,²⁹ presented in Supplementary Appendix SE.

CONCLUSION

Incorporating either unlexicalized KEs or LKEs informed by the UMLS Metathesaurus in a relation extraction system operating on biomedical texts shows significant promise. We present the REKE system, which established new state-of-the-art results for relation extraction when using LKEs. We also present a KEE system, which generates either LKEs or unlexicalized KEs for concepts and relation types encoded in the UMLS Metathesaurus. The KEE also delivers 2 neural models: (1) a neural model of the lexicalized knowledge encoder and (2) a neural model of the relation type knowledge encoder. The first model is used in REKE to generate LKEs for mentions of biomedical concepts encountered in texts, while the second model is used in REKE to generate LKEs for new relation types, not available in the UMLS Metathesaurus. As the focus of our research was to study the impact of KEs learned from UMLS on relation extraction, we found that both unlexicalized KEs and LKEs improve the quality of relation extraction, but LKEs produce the best results when incorporated in neural relation extraction.

AUTHOR CONTRIBUTIONS

MAW, RM, and SMH drafted the manuscript. MAW designed the neural systems in the work with significant discussions with and contributions from RM and SMH. MAW carried out the primary experiments. MAW and RM carried out secondary experiments. MAW contributed to data collection and analysis. MAW, RM, and SMH edited and provided feedback on all drafts.

SUPPLEMENTARY MATERIAL

[Supplementary material](#) is available at *Journal of the American Medical Informatics Association* online.

CONFLICT OF INTEREST STATEMENT

None declared.

REFERENCES

- Maldonado R, Yetisgen M, Harabagiu S. Adversarial learning of knowledge embeddings for the unified medical language system. *AMIA Jt Summits Transl Sci Proc* 2019; 2019: 543.
- Peng Y, Yan S, Lu Z. Transfer learning in biomedical natural language processing: an evaluation of BERT and ELMo on ten benchmarking datasets. In: proceedings of the 18th BioNLP Workshop and Shared Task; 2019: 58–65.
- Luo Y, Uzuner Ö, Szolovits P. Bridging semantics and syntax with graph algorithms-state-of-the-art of extracting biomedical relations. *Brief Bioinform* 2017; 18 (1): 160–78.
- Rink B, Harabagiu S, Roberts K. Automatic extraction of relations between medical concepts in clinical texts. *J Am Med Inform Assoc* 2011; 18 (5): 594–600.
- Xu H, Stenner SP, Doan S, *et al.* MedEx: a medication information extraction system for clinical narratives. *J Am Med Inform Assoc* 2010; 17 (1): 19–24.
- de Bruijn B, Cherry C, Kiritchenko S, *et al.* Machine-learned solutions for three stages of clinical information extraction: the state of the art at i2b2 2010. *J Am Med Inform Assoc* 2011; 18 (5): 557–62.
- Luo Y, Uzuner O. Semi-supervised learning to identify UMLS semantic relations. *AMIA Jt Summits Transl Sci Proc* 2014; 2014: 67–75.
- Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. *Proc AMIA Symp* 2001; 17–21.
- Devlin J, Chang M-W, Lee K, *et al.* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: proceedings of the 2019 Conference on the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, volume 1 (Long and Short Papers) 2019; 4171–86.
- D'Souza J, Ng V. Ensemble-based medical relation classification. In: proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. Dublin, Ireland: Dublin City University and Association for Computational Linguistics; 2014: 1682–93.
- Uzuner Ö, South BR, Shen S, *et al.* 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *J Am Med Inform Assoc* 2011; 18 (5): 552–6.
- He B, Guan Y, Dai R. Classifying medical relations in clinical text via convolutional neural networks. *Artif Intell Med* 2019; 93: 43–9.
- Luo Y, Cheng Y, Uzuner Ö, *et al.* Segment convolutional neural networks (Seg-CNNs) for classifying relations in clinical notes. *J Am Med Inform Assoc* 2018; 25 (1): 93–8.
- Zhang Y, Zheng W, Lin H, *et al.* Drug–drug interaction extraction via hierarchical RNNs on sequence and shortest dependency paths. *Bioinformatics* 2018; 34 (5): 828–35.
- Li D, Ji H. Syntax-aware multi-task graph convolutional networks for biomedical relation extraction. In: proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019); 2019: 28–33.
- Faruqi M, Dodge J, Jauhar SK, Dyer C, Hovy E, Smith NA. Retrofitting word vectors to semantic lexicons. In: proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; 2015: 1606–15.
- Alawad M, Hasan SS, Christian JB, Tourassi G. Retrofitting word embeddings with the UMLS metathesaurus for clinical information extraction. In: 2018 IEEE International Conference on Big Data; 2018: 2838–46.
- Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res* 2004; 32: D267–70.
- Herrero-Zazo M, Segura-Bedmar I, Martínez P, *et al.* The DDI corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *J Biomed Inform* 2013; 46 (5): 914–20.
- Ji G, He S, Xu L, Liu K, Zhao J. Knowledge graph embedding via dynamic mapping matrix. In: proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing; 2015: 687–96.
- Nickel M, Rosasco L, Poggio T. Holographic embeddings of knowledge graphs. In: proceedings of the 30th AAAI Conference on Artificial Intelligence; 2016: 1955–61.
- Bordes A, Usunier N, Garcia-Duran A, Weston J, Akhnenko O. Translating embeddings for modeling multirelational data. In: proceedings of the 27th Conference on Neural Information Processing Systems; 2013: 2787–95.
- Yang B, Yih W, He X, *et al.* Embedding entities and relations for learning and inference in knowledge bases. In: proceedings of the 3rd International Conference on Learning Representations (ICLR); 2015.
- Cai L, William YW. Kbgan: adversarial learning for knowledge graph embeddings. In: proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics; 2018: 1470–80.
- Goodfellow I, Pouget-Abadie J, Mirza M, *et al.* Generative adversarial nets. In: proceedings of the 27th Conference on Neural Information Processing Systems, volume 2; 2014: 2672–80.
- Schuster M, Paliwal KK. Bidirectional recurrent neural networks. *IEEE Trans Signal Process* 1997; 45 (11): 2673–81.
- Williams RJ. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach Learn* 1992; 8 (3–4): 229–56.
- Neumann M, King D, Beldagy I, *et al.* ScispaCy: fast and robust models for biomedical natural language processing. In: proceedings of the 18th BioNLP Workshop and Shared Task; 2019: 319–27.
- van der Maaten L, Hinton G. Visualizing data using t-SNE. *J Mach Learn Res* 2008; 9: 2579–605.