2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)

An Overview of Utilizing Knowledge Bases in Neural Networks for Question Answering

Sabin Kafle, Nisansa de Silva, Dejing Dou
Department of Computer and Information Science
University of Oregon
Eugene, Oregon, USA
{*skafle, nisansa, dou*}@*cs.uoregon.edu*

***Abstract*—Question Answering (QA) requires understanding queries expressed in natural languages and relevant information content to provide an answer. For closed-world QAs, information access is by means of either context texts, or a Knowledge Base (KB), or both. KBs are human-generated schematic representations of world knowledge. The representational ability of neural networks to generalize world information makes it an important component of current QA research. In this paper, we study the neural networks and QA systems in the context of KBs. Specifically, we focus on surveying methods for KB embedding, how such embeddings are integrated into the neural networks, and the role such embeddings play in improving performance across different question-answering problems.**

***Keywords*-Question-Answering; Knowledge Bases; Neural Networks; Knowledge Base Embedding;**

I. Introduction

Neural Question Answering (NQA) has lead to significant interest in question answering, especially due to the ability of modeling to incorporate multimodal information sources. To serve as a question-answering system, a typical neural network is capable of: leveraging text information via word or character embeddings [1]; image representation [2] via pretrained representations; textual information using unsupervised large-scale language models [3–5]; and/or KBs using embedding methods similar to word embeddings [6]. NQAs systems largely follow a three-stage process, comprised of (a) information retrieval based on the question understanding; (b) answer extraction to generate an answer; and, optionally, (c) a ranking module, to rank the answers [7].

Knowledge Graphs (KGs) are the simpler representational form of Knowledge Bases (KBs), expressed in the form of triples of - *entity, relation, entity* -. Unlike KBs which represent a richer hierarchy and structure symbolic to the real-world model, KGs are much less structured. The simpler representations of KGs have given rise to methods for the representation learning of entities and relations present in a KG. This is in line with advances in embedding methods for multimodal data representation. Most KBs are written in formats (e.g., OWL [8]), which makes them accessible via query languages such as SPARQL [9]. This itself is a significant research area and contributes to reasoner systems such as HermiT [10], which can be used to generate an answer from large knowledge graphs based on SPARQL query formulation. While KBs, which are often represented in structured format, are challenging to integrate into the neural network paradigms, KG embeddings are significantly easier to integrate into the existing systems. This leads to a multitude of applications including factoid question answering, visual question answering, reading comprehension, and open-world question answering, all using KGs as an auxiliary data source for improved performance.

Several KBs (and their triple-based variant KGs) are readily available, with huge amount of information and facts structured within. Some widely used KBs include Freebase [11], DBPedia [12], YAGO [13], Gene Ontology [14], Wordnet [15], ConceptNet [16], and Google Knowledge Graph [17]. Semantic parsing [18] approach to the factoid question answering parse a natural language question into a structured query, which is executed into KBs. A major limitation of a KG is its completeness - no KB exists with all the world's information content incorporated into it. NELL [19, 20] is an example system incorporating semi-automatic KBs, which are reliable in effective context understanding and information-extracting frameworks.

In this survey, we study neural question-answering methods applied to a wide range of question-answering problems including factoid question-answering, visual question-answering, and reading comprehension. We primarily explore the usability and contribution of KGs to neural question-answering. While several methods have been proposed to embed KBs, their usage is rather limited. We hypothesize that proper usage of KGs within a neural QA system should empower neural networks further.

II. Knowledge Base

A Knowledge Base (KB) is structured database with a schema, such as an ontology, describing entities, relations, and attributes, which form the foundation of structural information. Facts are then added to the KB in accordance with the structure, forming the entirety of a KB. A KB can also be represented as a graph of facts, with entities representing the

978-1-7281-1337-1/19/$31.00 ©2019 IEEE
DOI 10.1109/IRI.2019.00058

nodes of the graph, and with the relationships among entities being described by edges. For the rest of the survey, we treat Knowledge Base and Knowledge Graph as the same entity. The reason for such treatment is potential transformation of KB into graphs. Also, the existing neural network literature does not draw any distinction between the two terms and uses them interchangeably.

An ontology is a collection of definitions which model a domain using classes, attributes, and relationships [21]. The collection of facts, their attributes, and the relationships containing the discourse of a particular domain, together with its ontology, constitutes a KB. A KB is also a type of Knowledge Graph (KG); and thus, a KB is richly structured, based on its ontology. A KB can be simplistically interpreted as a database system with the schema analogous to ontology, and its *tuples* can be considered *facts*. A KB, though, is capable of incorporating a much richer set of information, such as logical relationships among facts, and can also be inferred using a formal logic reasoner. KBs are specifically useful for representing a domain that involves a rich set of relationships among different classes [22], e.g., WordNet [15].

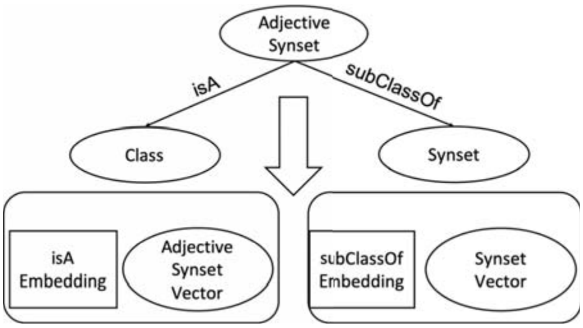

Figure 1. An example of an ontology embedding model.

Knowledge Graphs are typically stored as directed graphs of multi-relational data, whose nodes correspond to entities, and whose edges correspond to relations among them. KBs are represented as a triplet of form (h, l, t) or *(head, label, tail)*, which indicates that there exists a relationship of name $label$ between the entities $head$ and $tail$. The most widely used Knowledge Base is Freebase [11]. It is a structured KB in which entities are connected by predefined predicates or relations. All predicates are directional, connecting from subject to object. A triple *(subject, predicate, object)* denoted by (h, p, t) describes a fact; e.g, *(US Route2, major cities, Kalispell)* refers to the fact that US Route 2 runs through the city of Kalispell. The usage of knowledge graphs is limited by two issues - completeness [23, 24] and compatibility. The issue of completeness arises from the fact that no KBs can ever be exhaustively completed. This inadequacy can lead to error in a query-based system, which completely relies on KBs. Another challenge in usage of KBs lies in its compatibility. Each KB has their own design decisions, and thus, even for the same concepts and relations, different naming conventions are preferred, which presents a challenge in applying more than one KB to a problem. Application of more than one KB could potentially decrease the incompleteness of KBs [6]. A common solution is preferred to both problems: embedding of knowledge bases.

A. Knowledge Base Embedding

The general intuition of KB embedding methods is to learn connections and existing patterns from the KB, which can then either be used to extract further patterns using link predictions [6] or used in downstream tasks as an extremely compact representation of the global knowledge of KB. In general, the relations in KB are of the form - **symmetric**, **antisymmetric**, **inversion** and **composition**. KB embedding methods aim to infer the relations using either implicit or explicit modeling of one or many forms of KB relations [25]. *Symmetric* relations are valid even with the replacement of head with tail entity, while *antisymmetric* relations are not. *Inverse* relations are conjugate of one another, while *composition* refers to a relation defined as a path walk over multiple relations. We summarize some of the more popular approaches and their objectives and relation factorization in Table I.

1) General Embedding Framework: For E entities and R relations where G denotes the knowledge graph consisting of a set of triples (h, r, t) such that $h, t \in E$ and $r \in R$. The embedding model defines a score function $f(h, r, t)$ for each triple, which is the score of its implausibility. The objective of embedding models is to choose f such that score of a plausible triple (h, r, t) is smaller than score of an implausible one (h', r', t'). The model parameters are learned by minimizing a margin-based objective function:

$$\mathcal{L} = \sum_{\substack{(h,r,t)\in\mathcal{G} \\ (h',r,t')\in\mathcal{G}'_{(h,r,t)}}} [\gamma - f(h,r,t) + f(h',r,t')]_+$$

where $[x]_+ = max(0, x)$, and γ is the margin hyper-parameter. $\mathcal{G}'$ is the set of incorrect triples generated by corrupting the correct triple $(h, r, t) \in G$.

2) Embedding paradigms: Let us consider a Knowledge Graph (KG) consisting of components `(subject, predicate, object)`.

RESCAL [31] represents a KG as using a three-way tensor $\mathcal{X}$ with a tensor entry $\mathcal{X}_{ijk} = 1$ when there exists a relation `(i-th entity, k-th predicate, j-th relation)`. For non-existing and unknown relations, the entry is set to `zero`. This method factorizes each relation slice of the tensor as

$$\mathcal{X}_k \approx A R_k A^T, \text{for } k = 1, .., m \quad (1)$$

where A is $n \times r$ matrix containing latent component representation of entities, and R_k is an asymmetric $r \times r$

Embedding Methods	Distance Function		Properties
SE [26]	$\|\boldsymbol{W}_{r,1}\mathbf{h} - \boldsymbol{W}_{r,2}\mathbf{t}\|$	$\mathbf{h}, \mathbf{t} \in \mathbb{R}^k, \boldsymbol{W}_{r,\cdot} \in \mathbb{R}^{k\times k}$	Not Applicable
TransE [6]	$\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$	Antisymmetry, Inversion, Composition
DistMult [27]	$-\langle \mathbf{r}, \mathbf{h}, \mathbf{t}\rangle$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$	Symmetry
ComplEx [28]	$-\Re(\langle \mathbf{r}, \mathbf{h}, \bar{\mathbf{t}}\rangle)$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^k$	Symmetry, Antisymmetry, Inversion
HolE [29]	$-\langle \mathbf{r}, \mathbf{h} \otimes \mathbf{t}\rangle$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$	Symmetry, Antisymmetry, Inversion
ConvE [30]	$-\langle \sigma(\mathrm{vec}(\sigma([\bar{\mathbf{r}}, \bar{\mathbf{h}}] * \boldsymbol{\Omega}))\boldsymbol{W}), \mathbf{t}\rangle$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$	Not Applicable
RotatE [25]	$\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$	$\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^k, \|r_i\| = 1$	Symmetry, Antisymmetry, Inversion, Composition

Table I

DISTANCE FUNCTIONS $f_r(\mathbf{h}, \mathbf{t})$ OF KNOWLEDGE GRAPH EMBEDDING MODELS, WHERE $\langle\cdot\rangle$ IS THE GENERALIZED DOT PRODUCT, $\circ$ IS THE HADAMARD PRODUCT, $\otimes$ IS CIRCULAR CORRELATION, σ IS ACTIVATION FUNCTION AND $*$ IS 2D CONVOLUTION. $\bar{\cdot}$ IS CONJUGACY RELATION OF COMPLEX VECTORS, AND VECTOR RESHAPING FOR THE CONVE MODEL. COMPLEX AND HOLE METHODS ARE EQUIVALENT. DISTANCE-BASED METHODS ARE A COMPLEMENTARY-BUT-SIMILAR VIEW OF SIMILARITY-BASED METHODS. $\mathbb{C}$ REPRESENTS COMPLEX SPACE AND $\mathbb{R}$ REPRESENTS EUCLIDEAN SPACE.

matrix that models the interactions of the latent components in the k-th predicate. The factor matrices A and R_k can be computed by solving the regularized minimization formulation. The asymmetry of R_k takes into account whether the latent component occurs as a subject or an object. Nickel et al. [31] further explore how RESCAL related to other tensor factorization methods of rank-r DEDICOM and Tucker3.

STransE [32] is comprised of a triples scoring function as

$$f_r(h,t) = \|\mathbf{W}_{r,1}\mathbf{h} + \mathbf{r} - \mathbf{W}_{r,2}\mathbf{t}\|_{l_{1/2}} \tag{2}$$

where $\mathbf{W}$ is the embedding matrix and $\mathbf{r}$ is the relation vector. TransR [33] is comprised of a triples scoring function of the form

$$f_r(h,t) = \|\mathbf{h}\mathbf{M}_r + \mathbf{r} - \mathbf{t}\mathbf{M}_r\|_2^2 \tag{3}$$

Furthermore, Cluster-based TransR (CTransR) is proposed as well, where for each relation, the entity pairs (h,t) are clustered based on their distance $(\mathbf{h} - \mathbf{t})$ where $\mathbf{h}$ and $\mathbf{t}$ are obtained through TransE.

HOLE [29] is a compositional embedding method based on composition operation $\circ$.

$$\mathrm{P}(\phi_r(h,t) = 1|\Theta) = \sigma(\eta_{hrt}) = \sigma(r_r^{\mathrm{T}}(e_h \circ e_t)) \tag{4}$$

where $\phi_r(h,t)$ is the probability of relation between h and h, with η_{hrt}, the full tensor product, represented as composition of head and tail entities $(e_h \circ e_t)$ vector and transformed by the relation matrix r_r^{T}.

The composition operation $\circ$ between two entities can be either a full tensor product, concatenations, or even a circular correlation.

ComplEx [28] is very closely related to HOLE mathematically, where complex embedding is used to solve the problem through latent factorization. The dot product in complex space involves the conjugate transpose of one of the vectors, thus making it non-symmetric and anti-symmetric. Relations can receive different scores, depending on the ordering of the entities involved. The tensor of KG can be learned

$$X = \mathrm{Re}(EW\bar{E}^{\top}) \tag{5}$$

The above factorization shows that the head entity is the complex conjugate of its tail entity in vector space.

ROTATE [25] maps the entities and relations to the complex vector space and defines each relation as a rotation from the source entity to the target entity. Given a triplet (h,r,t), we expect $\mathbf{t} = \mathbf{h} \circ \mathbf{r}$ where $\circ$ is element-wise product and $|r_i| = 1$, and $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathcal{C}^k$. The distance function is ROTATE, which can be defined as

$$d_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\| \tag{6}$$

The objective for optimization is based on negative sampling loss

$$\begin{aligned} L = &- \log \sigma(\gamma - d_r(\mathbf{h}, \mathbf{t})) \\ &- \sum_{i=1}^{n} \frac{1}{k} \log \sigma(d_r(\mathbf{h}'_i, \mathbf{t}'_i) - \gamma) \end{aligned} \tag{7}$$

where γ is the fixed margin, and (h'_i, r, t'_i) are negative samples.

The pioneer work in translation-based embedding models is TransE [6]. It assumes all relations and entities can be represented by vectors of uniform size. One issue with the TransE model lies in its inability to differentiate among different relation mappings, such as *one-to-one*, *many-to-one*, and *one-to-many*, which makes the model unsuitable for representing such relations. TransH [34] treats each relation to be on a different plane. Figure 2 shows the geometrical contrast between TransH and TransE. Other translation methods, TransD [35] and TransX [33], consider diversity of both entity and relation.

In addition, there are several tensor factorization methods for relational learning that generate embeddings for KBs [23, 31, 36–38]. Bayesian Clustering methods have also been successfully applied to embed a KB [39]. Distance-based embedding methods [6, 32, 34, 40, 41] have simpler frameworks, making them preferable for usage in underlying applications.

Additionally, relation paths between entities in Knowledge Graphs provide richer context information, which enables learning more structured embeddings [40–46]. Path queries, to obtain a relational transformation, which is then integrated into a translation model, such as TransE, are used by [40]. The approach in [45] extends the TransE method by the additional objective of learning scoring from a different relation path representation, which is a summation over all relation paths that are termed reliable. The study [46] proposed a dynamic algorithm to enable efficient incorporation of relation paths of bounded length in compositional path models. The authors of [47] propose a KB completion method using RNNs, which are able to infer multi-hop relationships. An external text corpus for correlating KBs with text is used by [48].

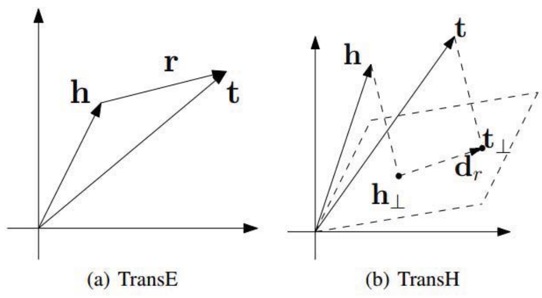

Figure 2. Geometrical modeling of *TransE* [6] and *TransH* [34]. TransE translated head entity to tail entity using relation as a vector, while TransH projects the entity embeddings into a relation plane where the actual translation is performed. Such geometric innovations are often the defining factors in improving KB embedding benchmarks.

The more recent embedding methods focus on variable geometry of embedding space, such as hyperbolic geometry [49], leading to learning multiple models of embedding in hyperbolic space [50–53], which shows much promise for both learning compact representation and using smaller dimensions for learning embeddings. Another research direction is along learning ordered embeddings [54], which are capable of representing hierarchy and order within the geometrical structures [55].

III. Question-Answering Architectures

We briefly review some of the neural networks widely used for question answering. Neural networks [56] enable learning of representation of data with multiple levels of abstraction. These levels of abstractions enable deep learning methods to generalize information, while also being able to narrow down to a specific aspect of information. Different architectures of neural networks, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and more recently, transformer networks [57], are widely used for challenging learning tasks [3], including question answering [58, 59]. Additionally, smaller models are widely used for unsupervised pretraining [1, 60]. Attention mechanisms [61] have proven useful for filtering useful content for retrieval tasks in NQAs.

Memory Networks [62] and related architectures including Neural Turning Machines [63, 64] are neural networks with external memory. They represent an extremely useful paradigm for solving factoid question answering [65] and question answering involving reasoning [62]. Their novelty lies in their ability to manipulate external memory locations, such as a Knowledge Base (KB) or a Universal Schema [66]. Another advantage lies in their different level of guidance applied (i.e, additional information incorporation is easier than in standard neural architectures [62]). We show a simple architecture used for general question-answering systems in Figure 3.

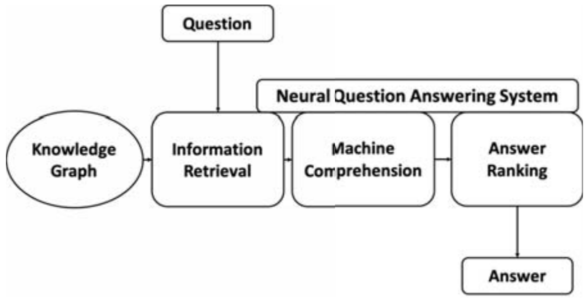

Figure 3. A general architecture for neural question answering, comprised of three components: *Information Retrieval*, which often interacts with Knowledge Graph in embedded form, for generating answer candidates; *Machine Comprehension* and *Answer Ranking*, which are mostly model-dependent. The *Machine Comprehension* component is comprised of attending over multiple layers of information to generate answer candidates. The *Answer Ranking* is based on relevance to the question, while the *machine comprehension* is focused on validating the answer by attending over information sources. Memory networks enable *Machine Comprehension* to interact directly with KB by performing multi-stage retrieval in an iterative manner.

IV. Factoid Question Answering

Factoid Question Answering (FQA) refers to questions which can be answered effectively by a phrase or an entity of a KG. There are mainly two approaches to FQAs - answering questions over a KG or obtaining answer from natural text using open information extraction mechanisms. Few approaches exist which attempt to combine both resources or use multiple KBs.

A Knowledge Graph-based factoid question answering involves mapping the question in natural language into triples of Knowledge Graphs. The distinction is made between FQA systems mapping to just one triples and mapping to multiple triples. The system which maps to a single triple is called Simple Question Answering (SimpleQA). Simple QA is a relatively easy problem compared to other factoid and non-factoid QAs. They are also the most frequent type of questions asked [67]. A SimpleQA task involves answering a question such as *"What is the hometown of Obama?"* which asks for a direct topic of an entity *"Obama"* which is *"hometown"*. The challenges to SimpleQA systems lie in possibility to formulate a question in multiple ways, making the mapping process hard to generalize. Another highly successful paradigm to factoid question is semantic parsing [18, 68, 69]. The semantic parser transforms natural language into logical form. It is capable of solving tricky questions involving multiple relations and questions involving ordering.

A. Simple Question Answering (SimpleQA)

A common approach to solving a SimpleQA problem is to extract a set of candidate answers from Knowledge Base using relation extraction [68–71] or distributed representation [72–74]. **WikiAnswers** [67] is introduced as a paraphrasing dataset which helps generalize for unseen words and question patterns. Another dataset, **SimpleQuestions**, is introduced by [65]. SimpleQA involves embedding of a knowledge base to find the entity of the knowledge base which is closest to the question's representation as the answer. The general framework for factoid question answering is: Given an input question sentence $S = \{w_1, w_2, .., w_Q\}$ and a sentence representation $s \in R^k$, we find the entity e in KB E such that $f(s, e) > f(s, e'), e' \cup e = E$.

A CNN-based approach can be applied to factoid QAs [75] with a two-step pipeline: entity linking, and fact selection. Memory networks are applied in [65] to simple question answering. The memory network consists of a memory, and of a neural network which is trained to query that memory, given some inputs. It consists of four components: Input map (I); Output map (O); Generalization (G); and Response (R). The workflow is to store *Freebase* into memory and then train the model to answer questions. A KB triplet is represented by a bag-of-words model, with subject and relationship having value 1 and object entries set to $1/k$, where k is the number of objects. The answer ranking is based on cosine similarity. Lukovnikov et al. [76] encode questions using GRUs, and a word is represented as a concatenation of Glove vectors [60] with character level encoding. Golub and He [77] propose a character-level approach based on the attention-enhanced encoder-decoder architecture [61]. The model of [77] consists of a character-level RNN-based question encoder and an attention-enhanced RNN decoder, coupled with two separate character-level CNN-based entity label and predicate URI encoders.

A word-level RNN-based approach with emphasis on possible paraphrases of questions is proposed by [78]. The task of predicting subject and relation is factorized into two sub-tasks: prediction of relation first, followed by entity given the relation and question. Both [78] and [75] improve the performance of their approaches using a BiLSTM-CRF tagging model which is separately trained to label parts of the question as entity mention or context (relation pattern).

B. Multi-Relation Question Answering

The formulation of multi-relation question answering is driven by the necessity to map questions in natural text to more than one triple in a knowledge base. For challenging questions, such as "*What mountain is the highest in North America?*," which requires learning a representation for mathematical function "highest", Xu et al. [74] use textual data to filter out wrong answers. A dependency parser-based query node expansion is devised in [69] where ClueWeb text is used to learn correlation between KB relations and words using co-occurrence statistics with the alignment model. Dong et al. [73] uses multi-column CNNs to understand questions from three different aspects: answer path, answer context, and answer type. Then it learns their distributed representations. Yang et al. [79] maps natural language to knowledge base by semi-automatically generating mappings between knowledge base triples and natural text, using information extraction methods.

[80] propose an encoder-decoder framework model for factoid question answering, with ability to query a KB. Yin et al.[81] pre-process Freebase to remove dummy entities and to obtain more direct triples. An L-hop factual memory network is constructed for computational layers, where each layer accesses candidate facts and question embedding.

A major constraint on factoid question answering models is the data limitation. While there are multiple ways to phrase a single question, the dataset size suffers from sparseness and is unable to work with methods that require a larger training datasize. SimpleQA have made substantial progress recently, due to the introduction of the SimpleQuestions [65] dataset, making larger neural network models trainable until convergence without overfitting. While the focus on the SimpleQA task is to generalize mapping of questions to facts, non-simple QA tasks and multi-resource open domain QA tasks require learning the mathematical and functional dependencies required to answer the question. This makes the problem considerably more complex, while at the same time, limited training data constrains the model to use lesser parameters. There are also very few methods which attempt to leverage multiple knowledge sources.

V. Attention-based Question Answering

Attention-based QA are extremely popular approaches for multi-modal data problems such as Visual Question Answering (VQA) and problems requiring deeper understanding of input data, such as Reading Comprehension (RC) (also called Machine Comprehension). A common approach to VQA concatenates visual and textual representations obtained from CNN and RNN respectively, to perform joint inference. This approach can be improved upon by introduction of attention maps for input image, each with embedding for a certain section of image, which are then attended over using attention mechanism for learning a joint embedding which then performs the final classification or sequence generation task. Multimodal bilinear compact pooling [82] proposes an efficient but highly optimized bilinear pooling over two data sources, enabling a robust embedding for visual question answering.

R-Net [83] obtain significant performance gains on RC dataset, SQuAD [84]. The difference between VQA and RC lies in decoding stage of inference, where VQA decoding is done based upon preset vocabulary. RC datasets require sampling of input text to generate answer phrases or sequences. This requires probabilistic decoding, using a combination of language decoding and pointer networks [85] to obtain answer effectively. R-Net uses GRUs to learn embeddings for the input question and sentence, which are then passed to gated attention-based recurrent networks to determine importance of information in the passage regarding a question. Each passage representation incorporates aggregating matching information from the whole question. Another gate is added to determine the importance of passage parts relevant to the question. Another attention to match over itself is used to incorporate context into question-aware embeddings. A Pointer Network is used to predict the start and end position of the answer. The success of R-Net has given rise to Reasonet [86], Fusionnet [87], Qanet [88], Macnet [89], and S2-Net [90].

While there are many different variants of visual question answering and reading comprehension methods in literature (see [59] for more details), the underlying mechanism entails learning the fixed vector representation for both question and input data (either image or text), then using the attention or bilinear pooling to learn joint embeddings. The learned vectors are used for making predictions. We do not attempt to cover the entire attention-based question-answering methods, due to space and time constraints. Recently, it was found that using transfer-learning approaches [3] often significantly improves the performance of the model. This was utilized in multiple novel works [91–94].

VI. Conclusion

In this paper, we surveyed multiple areas of neural question answering, including Knowledge Base embeddings, neural networks architecture, and various advances in factoid and attention-based question answering. While Knowledge Base (KB) embeddings methods are advanced enough to be relied upon as information resources, we observe that multitudes of works on question answering still rely on older approaches. This leads to suboptimal performance from KBs, making a proper evaluation difficult. We believe this paper serves as an important milestone in syncing up the progress across different fields, in order to leverage strong, connected components for building richer sets of question answering models. The advancements in research in KB embeddings toward different geometrical spaces, including hyperbolic spaces, suggests that neural networks with representational capacity in such spaces with curvature may be the next application for building question-answering models.

References

[1] T. Mikolov *et al.*, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013, pp. 3111–3119.

[2] Q. Wu *et al.*, "Visual question answering: A survey of methods and datasets," *Computer Vision and Image Understanding*, 2017.

[3] J. Devlin *et al.*, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019, pp. 4171–4186.

[4] A. Radford *et al.*, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, 2019.

[5] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *ACL*, 2018, pp. 328–339.

[6] A. Bordes *et al.*, "Translating embeddings for modeling multi-relational data," in *NIPS*, 2013, pp. 2787–2795.

[7] B. Kratzwald *et al.*, "Rankqa: Neural question answering with answer re-ranking," *arXiv preprint arXiv:1906.03008*, 2019.

[8] G. Antoniou and F. Van Harmelen, "Web ontology language: Owl," in *Handbook on ontologies*. Springer, 2004, pp. 67–92.

[9] A. Seaborne and E. Prudhommeaux, "Sparql query language for rdf," *W3C recommendation*, 2006.

[10] R. Shearer *et al.*, "Hermit: A highly-efficient owl reasoner." in *OWLED*, vol. 432, 2008, p. 91.

[11] K. Bollacker *et al.*, "Freebase: a collaboratively created graph database for structuring human knowledge," in *ACM-SIGMOD*. ACM, 2008, pp. 1247–1250.

[12] S. Auer *et al.*, "Dbpedia: A nucleus for a web of open data," *The semantic web*, pp. 722–735, 2007.

[13] F. M. Suchanek *et al.*, "Yago: a core of semantic knowledge," in *WWW*. ACM, 2007, pp. 697–706.

[14] M. Ashburner *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[15] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[16] H. Liu and P. Singh, "Conceptnet-a practical commonsense reasoning tool-kit," *BT technology journal*, vol. 22, no. 4, pp. 211–226, 2004.

[17] A. Singhal, "Introducing the knowledge graph: things, not strings," *Official google blog*, 2012.

[18] J. Berant *et al.*, "Semantic parsing on freebase from question-answer pairs." in *EMNLP*, vol. 2, 2013, p. 6.
[19] T. Mitchell *et al.*, "Never-ending learning," *Communications of the ACM*, vol. 61, no. 5, pp. 103–115, 2018.
[20] A. Carlson *et al.*, "Toward an architecture for never-ending language learning." in *AAAI*, vol. 5, 2010, p. 3.
[21] T. Gruber, "Ontology," *Encyclopedia of database systems*, pp. 1963–1965, 2009.
[22] B. Chandrasekaran *et al.*, "What are ontologies, and why do we need them?" *IEEE Intelligent Systems and their applications*, vol. 14, no. 1, pp. 20–26, 1999.
[23] R. Socher *et al.*, "Reasoning with neural tensor networks for knowledge base completion," in *NIPS*, 2013, pp. 926–934.
[24] R. West *et al.*, "Knowledge base completion via search-based question answering," in *WWW*. ACM, 2014, pp. 515–526.
[25] Z. Sun *et al.*, "Rotate: Knowledge graph embedding by relational rotation in complex space," *arXiv preprint arXiv:1902.10197*, 2019.
[26] A. Bordes *et al.*, "Learning structured embeddings of knowledge bases," in *AAAI*, 2011.
[27] B. Yang *et al.*, "Embedding entities and relations for learning and inference in knowledge bases," *arXiv preprint arXiv:1412.6575*, 2014.
[28] T. Trouillon *et al.*, "Complex embeddings for simple link prediction," in *ICML*, 2016, pp. 2071–2080.
[29] M. Nickel *et al.*, "Holographic embeddings of knowledge graphs," in *AAAI*, 2016.
[30] T. Dettmers *et al.*, "Convolutional 2d knowledge graph embeddings," in *AAAI*, 2018.
[31] M. Nickel *et al.*, "A three-way model for collective learning on multi-relational data," in *ICML*, 2011, pp. 809–816.
[32] D. Q. Nguyen *et al.*, "Stranse: a novel embedding model of entities and relationships in knowledge bases," in *NAACL-HLT*, 2016, pp. 460–466.
[33] Y. Lin *et al.*, "Learning entity and relation embeddings for knowledge graph completion." in *AAAI*, 2015, pp. 2181–2187.
[34] Z. Wang *et al.*, "Knowledge graph embedding by translating on hyperplanes." in *AAAI*. Citeseer, 2014, pp. 1112–1119.
[35] G. Ji *et al.*, "Knowledge graph embedding via dynamic mapping matrix." in *ACL (1)*, 2015, pp. 687–696.
[36] M. Nickel and V. Tresp, "Logistic tensor factorization for multi-relational data," *arXiv preprint arXiv:1306.2084*, 2013.
[37] M. Nickel *et al.*, "Factorizing yago: scalable machine learning for linked data," in *WWW*. ACM, 2012, pp. 271–280.
[38] D. Krompaß *et al.*, "Type-constrained representation learning in knowledge graphs," in *International Semantic Web Conference*. Springer, 2015, pp. 640–655.
[39] I. Sutskever *et al.*, "Modelling relational data using bayesian clustered tensor factorization," in *NIPS*, 2009, pp. 1821–1828.
[40] K. Guu *et al.*, "Traversing knowledge graphs in vector space," in *EMNLP*, 2015, pp. 318–327.
[41] D. Q. Nguyen *et al.*, "Neighborhood mixture model for knowledge base completion," in *CoNLL*, 2016, pp. 40–50.
[42] Y. Luo *et al.*, "Context-dependent knowledge graph embedding." in *EMNLP*, 2015, pp. 1656–1661.
[43] A. Garcia-Duran *et al.*, "Composing relationships with translations," in *EMNLP*, 2015, pp. 286–290.
[44] C. Liang and K. D. Forbus, "Learning plausible inferences from semantic web knowledge by combining analogical generalization with structured logistic regression." in *AAAI*, 2015, pp. 551–557.
[45] Y. Lin *et al.*, "Modeling relation paths for representation learning of knowledge bases," in *EMNLP*, 2015, pp. 705–714.
[46] K. Toutanova *et al.*, "Compositional learning of embeddings for relation paths in knowledge bases and text," in *ACL*, vol. 1, 2016, pp. 1434–1444.
[47] A. Neelakantan *et al.*, "Compositional vector space models for knowledge base completion," in *ACL*, 2015, pp. 156–166.
[48] Z. Wang and J. Li, "Text-enhanced representation learning for knowledge graph," in *IJCAI*. AAAI Press, 2016, pp. 1293–1299.
[49] W. P. Thurston, "Three dimensional manifolds, kleinian groups and hyperbolic geometry," *Bulletin of the American Mathematical Society*, vol. 6, no. 3, pp. 357–381, 1982.
[50] M. Nickel and D. Kiela, "Poincaré embeddings for learning hierarchical representations," in *NIPS*, 2017, pp. 6338–6347.
[51] O.-E. Ganea *et al.*, "Hyperbolic entailment cones for learning hierarchical embeddings," in *ICML*, 2018, pp. 1632–1641.
[52] M. Nickel and D. Kiela, "Learning continuous hierarchies in the lorentz model of hyperbolic geometry," in *ICML*, 2018, pp. 3776–3785.
[53] C. De Sa *et al.*, "Representation tradeoffs for hyperbolic embeddings," *Proceedings of machine learning research*, vol. 80, p. 4460, 2018.
[54] I. Vendrov *et al.*, "Order-embeddings of images and language," *arXiv preprint arXiv:1511.06361*, 2015.
[55] L. Vilnis *et al.*, "Probabilistic embedding of knowledge graphs with box lattice measures," in *ACL*, 2018, pp. 263–272.
[56] Y. LeCun *et al.*, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[57] A. Vaswani *et al.*, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.
[58] M. Iyyer *et al.*, "A neural network for factoid question

answering over paragraphs." in *EMNLP*, 2014, pp. 633–644.
[59] S. Antol *et al.*, "Vqa: Visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2425–2433.
[60] J. Pennington *et al.*, "Glove: Global vectors for word representation." in *EMNLP*, vol. 14, 2014, pp. 1532–1543.
[61] D. Bahdanau *et al.*, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.
[62] J. Weston *et al.*, "Memory networks," *arXiv preprint arXiv:1410.3916*, 2014.
[63] A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," *arXiv preprint arXiv:1410.5401*, 2014.
[64] S. Sukhbaatar *et al.*, "End-to-end memory networks," in *NIPS*, 2015, pp. 2440–2448.
[65] A. Bordes *et al.*, "Large-scale simple question answering with memory networks," *arXiv preprint arXiv:1506.02075*, 2015.
[66] S. Riedel, L. Yao, A. McCallum, and B. M. Marlin, "Relation extraction with matrix factorization and universal schemas," in *NAACL-HLT*, 2013, pp. 74–84.
[67] A. Fader *et al.*, "Paraphrase-driven learning for open question answering." in *ACL (1)*. Citeseer, 2013, pp. 1608–1618.
[68] W.-t. Yih *et al.*, "Semantic parsing for single-relation question answering." in *ACL (2)*. Citeseer, 2014, pp. 643–648.
[69] X. Yao and B. Van Durme, "Information extraction over structured data: Question answering with freebase." in *ACL (1)*. Citeseer, 2014, pp. 956–966.
[70] X. Yao, "Lean question answering over freebase from scratch." in *HLT-NAACL*, 2015, pp. 66–70.
[71] H. Bast and E. Haussmann, "More accurate question answering on freebase," in *ACM-CIKM*. ACM, 2015, pp. 1431–1440.
[72] A. Bordes *et al.*, "Question answering with subgraph embeddings," in *EMNLP*, 2014, pp. 615–620.
[73] L. Dong *et al.*, "Question answering over freebase with multi-column convolutional neural networks." in *ACL (1)*, 2015, pp. 260–269.
[74] K. Xu *et al.*, "Question answering on freebase via relation extraction and textual evidence," in *ACL*, 2016, pp. 2326–2336.
[75] W. Yin *et al.*, "Simple question answering by attentive convolutional neural network," in *COLING*, 2016, pp. 1746–1756.
[76] D. Lukovnikov *et al.*, "Neural network-based question answering over knowledge graphs on word and character level," in *WWW*, 2017, pp. 1211–1220.
[77] D. Golub and X. He, "Character-level question answering with attention," in *EMNLP*, 2016, pp. 1598–1607.
[78] Z. Dai *et al.*, "Cfo: Conditional focused neural question answering with large-scale knowledge bases," in *ACL*, 2016, pp. 800–810.
[79] M.-C. Yang *et al.*, "Joint relational embeddings for knowledge-based question answering." in *EMNLP*, vol. 14, 2014, pp. 645–650.
[80] J. Yin *et al.*, "Neural generative question answering," in *Proceedings of the Workshop on Human-Computer Question Answering*, 2016, pp. 36–42.
[81] S. Jain, "Question answering over knowledge base using factual memory networks," in *NAACL-HLT*, 2016, pp. 109–115.
[82] A. Fukui *et al.*, "Multimodal compact bilinear pooling for visual question answering and visual grounding," in *EMNLP*, 2016, pp. 457–468.
[83] W. Wang *et al.*, "Gated self-matching networks for reading comprehension and question answering," in *ACL*, 2017.
[84] P. Rajpurkar *et al.*, "Squad: 100,000+ questions for machine comprehension of text," in *EMNLP*, 2016, pp. 2383–2392.
[85] O. Vinyals *et al.*, "Pointer networks," in *NIPS*, 2015, pp. 2692–2700.
[86] Y. Shen *et al.*, "Reasonet: Learning to stop reading in machine comprehension," in *ACM-SIGKDD*. ACM, 2017, pp. 1047–1055.
[87] H.-Y. Huang *et al.*, "Fusionnet: Fusing via fully-aware attention with application to machine comprehension," *arXiv preprint arXiv:1711.07341*, 2017.
[88] A. W. Yu *et al.*, "Qanet: Combining local convolution with global self-attention for reading comprehension," *arXiv preprint arXiv:1804.09541*, 2018.
[89] B. Pan *et al.*, "Macnet: Transferring knowledge from machine comprehension to sequence-to-sequence models," in *NIPS*, 2018, pp. 6092–6102.
[90] C. Park *et al.*, "S2-net: Machine reading comprehension with sru-based self-matching networks," *ETRI Journal*, 2019.
[91] S. Reddy *et al.*, "Coqa: A conversational question answering challenge," *TACL*, vol. 7, pp. 249–266, 2019.
[92] S. Min *et al.*, "Multi-hop reading comprehension through question decomposition and rescoring," *arXiv preprint arXiv:1906.02916*, 2019.
[93] H. Li *et al.*, "Visual question answering as reading comprehension," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6319–6328.
[94] A. Zadeh *et al.*, "Social-iq: A question answering benchmark for artificial social intelligence," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8807–8817.