DeepEdu - Learning and Representation

Fanli (Christian) Zheng
dyad x machina
thechristianramsey@gmail.com

Haohan Wang
dyad x machina
haohan723@gmail.com

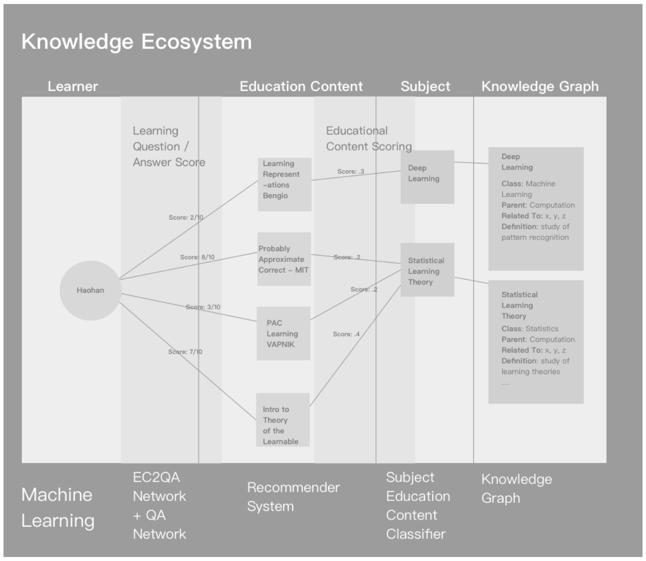


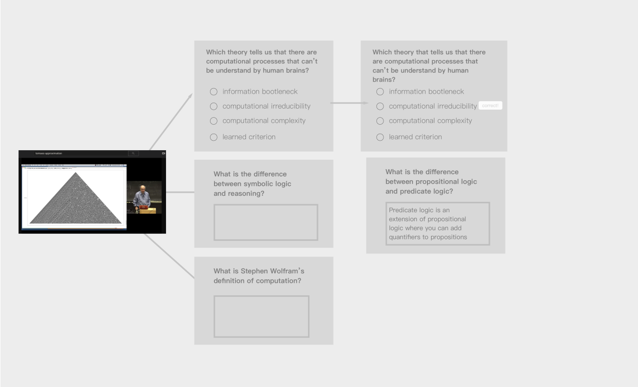
Figure 1: Knowledge Ecosystem

1. INTRODUCTION

1.1 A Future

Let's start with a future view of an individual's education. Many of us have used the internet to educate ourselves with the many media from high-quality videos, papers, articles, podcasts to how-tos being uploaded from numerous individuals, groups, and institutions like never before (60 hours of video are uploaded to youtube.com every minute).

Let us imagine that all of what you have learned online, throughout the entirety of your life, from the hundreds of Youtube videos, Wikipedia articles, Nature papers, and podcasts you've read, watched, or listened to, were all added structurally to your **knowledge journey**, and what if that journey could be consolidated into what we might call a **knowledge footprint** that could be shared with others? Could this replace static degrees? Or augment them to be more inclusive of a learner's true knowledge? How might we test such knowledge? Could we even predict and provide the guidance on what an individual should be learning next to best support their knowledge acquisition?



Which theory tells us that there are computational processes that can't be understand by human brains?

○ information bootleneck
○ computational irreducibility
○ computational complexity
○ learned criterion

Which theory that tells us that there are computational processes that can't be understand by human brains?

○ information bootleneck
○ computational irreducibility correct!
○ computational complexity
○ learned criterion

What is the difference between symbolic logic and reasoning?

What is the difference between propositional logic and predicate logic?

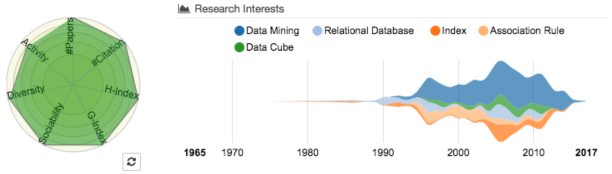
Predicate logic is an extension of propositional logic where you can add quantifiers to propositions

What is Stephen Wolfram's definition of computation?

Figure 2: DeepEdu Network

1.2 A Comparison

Now, let's go back to our current approach to education. Many of us treat knowledge acquisition like a chapter in the individual's life that is limited to one or more formal places i.e. universities. This is misleading since we accrue knowledge from everywhere and most recently the internet has become a primary source of knowledge acquisition but has gone mostly unaccounted for in terms of recognition (i.e. watching a whole series of Youtube lectures on the Information Theory or Discrete Mathematics goes mostly unnoticed when someone views one's resume or by simply looking at his or her degree). The current approach makes it much harder for people to switch to working and exploring the domains or professional fields outside of their degree area. Knowing rigorous mathematics and not having a degree in it, is said to be surprising, therefore the current "thumbnail view" of an individual's knowledge is necessarily inadequate to the new mediums of knowledge acquisition.



The ideas behind this *knowledge ecosystem*, presents only one of many possible solutions to bringing our education system into modernity. The goal of it would be to promote the long-held idea of the **life-long learner**. Moving away from the "education chapter"" of an individual's life to the individual as an evolving learner; learning the necessary skills for what life presents them with today or might tomorrow. It would (combined with traditional education) show us a more accurate depiction of a learner's knowledge and therefore that of a society's collective knowledge.

Visualised over time, we could begin to capture a learner's so-called **knowledge journey**. Composed of every piece of content they've gained knowledge from mapped to the *human knowledge graph*. Showing how an individual has traversed through the world of human knowledge.

This would also serve as a way for others, who may be on a similar **knowledge journey** to connect with "their" cohort which may not need to be bound by geography or demography. This could be the start of meetups, study groups, flexible class models and so on.

For those who are looking for a change, they may find different journeys that help them decide what step to take next. You would also be able to connect someone's occupation to their **knowledge journey**.

On aggregate, we could begin to cluster similar **knowledge journeys** through unsupervised learning, which might lead to completely new journeys that others may be inspired to follow.

1.3 Knowledge Ecosystem

In this essay, we will propose a **knowledge ecosystem**, a new way of approaching education that attempts to build a more accurate depiction of a learner's true knowledge. It will require significant effort to bring to life but we believe the benefits will outweigh the costs. We will talk about how we can use machine learning, deep learning, in particular, to help create and support a **knowledge ecosystem** which is made up of the learner's **knowledge footprint**, **knowledge journeys**, and a **collective human knowledge graph**. We will walk you through some most recent research findings that would enable us to take the space of unstructured educational content on the web and do the following:

- classify content to higher level subjects
- map content unto the human knowledge graph
- test a learner's knowledge of recently viewed educational content through questions and answers, no what matter the subject.

We will also argue that this imagined future is not only **desirable** for society but something similar is required to ensure individual's knowledge to be well represented in a time where the pace of change is rapidly speeding up.

Let us not forget, that even software engineering is currently being recreated with machine learning as a key pillar which wasn't much of a thought 5-10 years ago.

It is going to be tantamount if we have an adaptive system that can represent our current knowledge and also make us predictable to others given the future pushes us to know more than ever and knowing whom to collaborate with to apply such knowledge.

This hypothetical future isn't just conceptual, most of what we will present to you today is currently feasible due to the most recent advances in machine learning, and in particular deep learning

In the last section of this essay, I will review what has been proposed and also call other researchers, educators, and designers to collaborate on such an ecosystem, even if it is just in part.

*Note: For the purpose of this essay we will talk mostly about digital knowledge acquisition and leave the reader to extend the basics to knowledge obtained elsewhere.

2. PRIMARY CONCERNS

There are 3 main concerns that we will attempt to address in this article about online knowledge acquisition that stands in the way of having an adaptive and reliable knowledge ecosystem. We will attempt to present a system that can sufficiently overcome each of the concerns here and in the implementation section.

There are as follows:

- **Passive Consumption** - most of the online content is viewed passively by the learner and the result of passive consumption is that a learner does not grasp the concepts being taught.
- **Untested Knowledge** - even if the learner was engaged while viewing a piece of educational content their knowledge is untested and therefore it isn't clear if they've mastered the content accurately and in some sense holistically.
- **Knowledge Representation** - even if the learner was engaged (1) and their knowledge was tested (2), simply knowing the counts or types of video they watched doesn't make their knowledge predictable and useful to others. In fact, even the learner may be unaware of all of what they've viewed.

2.1 Passive Consumption and Untested Knowledge

How would such an ecosystem ensure us against passive consumption?

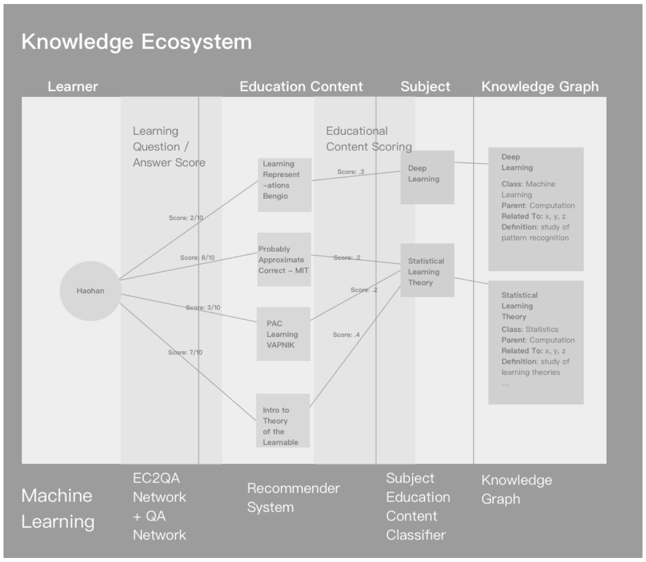


Figure 3: Knowledge Ecosystem

Scenario #1 A learner goes online and begins watching a series on Machine Learning. How do we engage and test a user's knowledge?

Proposition: Using advances in deep learning, we propose a dual question and answer generation framework given the educational content.

Result: A learner gets a set of questions and multiple choice answers throughout the video. Keeping the user engaged and sharp to ensure they can answer each of the questions.

As you can see, we've bundled passive consumption and untested knowledge because our proposed ecosystem approaches both of these by always testing the knowledge.

2.2 Knowledge Ecosystem Example

Given a piece of educational content, our knowledge system will generate a set of questions and answers that theoretically capture the major concepts and facts that the learner should know after viewing a part of the content in whole.

You can imagine watching a Youtube video and after a learner views 15 minutes of an hour-long lecture on Computational Complexity a quiz is presented (i.e. conditioned on the past 15 minutes of video), and the score is recorded. In the future, we would also be able to use the knowledge graph to bring in learner's existing knowledge in order to generate more complex questions and answers with these priors and the current educational content.

As the first step, our knowledge system would only consider content that is currently in use for now.

2.3 The Problem of Knowledge Representation

> Given most learner's online knowledge acquisition varies and has been invisible up to now, how we can best represent their knowledge?

Scenario #2 A learner has a degree in Public Health, but since graduating, he has been studying machine learning for the last 3 years. The learner now wants to apply to a job that requires the skills in both Health and Machine Learning. How do we represent their traditional and updated knowledge?

This is a tricky problem that goes beyond any given algorithm. The exact design of a **knowledge footprint** and a **knowledge journey** has been attempted and we will not cover that in depth here. The proposed system presupposes the design of the knowledge footprint.

There is another problem:

> how do we reduce someone's knowledge (in this case a set of educational content and their respective scores) into a symbol that is representative of his/her current knowledge and could be shared across?

Proposition: We introduce **knowledge journeys** and the **knowledge graph** as a way to make sense and structure a learner's knowledge acquisition. The collective **knowledge graph** will tell us about the subject the learner is studying and we can use this to compare to others and create a relative comparison.

Result: Reducing a learner's **knowledge journey** into a common set of dimensions that make up into their **knowledge footprint** which would look similar to those with similar journeys.

As a result, the employer, now familiar with the footprints can check the overlap between the current employee's and a prospective employee's to support their decision making.

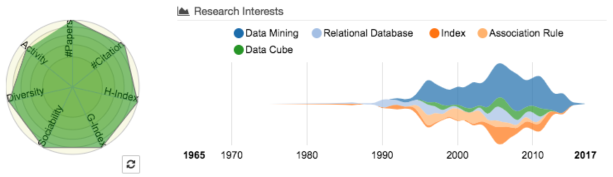
3. CONCEPTS

As mentioned above, we will be introducing a few novel concepts that we believe are the key components of such an educational ecosystem.

3.1 Knowledge Footprint

The concept of a knowledge footprint is one or more custom symbol(s) or badge(s) with a profile that represents one's education relative to that of others. This concept is particularly designed for solving the second concern (knowledge representation) that we proposed earlier.

In turn, this footprint should encapsulate the information of all of one's education (currently focused on digital) while balancing distinction and commonality with others.



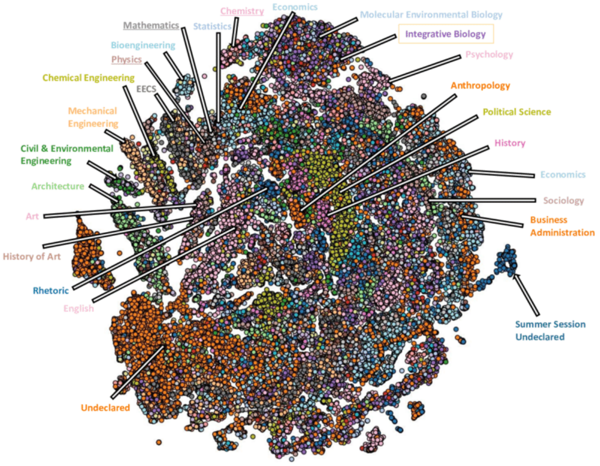


Figure 4: Image title

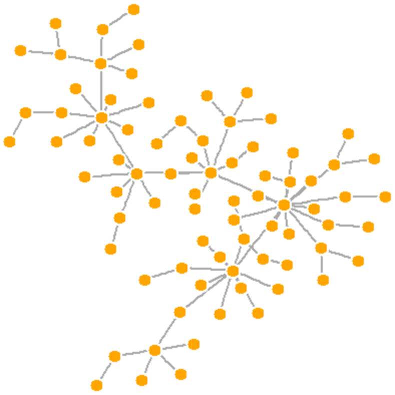


Figure 5: Image title

3.2 Knowledge Journeys

Image src: `https://www.researchgate.net/figure/t-SNE-projection-of-the-embedding-of-all-learners-in-the-dataset-Major-l` `fig1_323391033`

A **knowledge journey** is a somewhat holistic view of all of the educational content a learner has acquired over time. The journey should be a temporal representation of all of the subjects that one has viewed and been tested on. Knowledge journey should be simple enough to comprehend and compare but complex enough that the individual can go back to any particular moment in time and review the educational content they've viewed before. Coupled with the knowledge graph, it can also shed light on the possible next education content that a learner should be or would be interested in learning.

This concept also helps address the knowledge representation concern. This time instead of capturing a consolidated snapshot of one's current knowledge profile, this concept takes a temporal route to capture one's entire learning path.

3.3 Collective Human Knowledge Graph

The collective human knowledge graph can be compared to Google's Search Knowledge Graph which points unstructured information towards the structure. The graph should have all existing subjects that we are currently aware of (i.e. Mathematics, Computer Science, Art, and Sociology). Since each piece of educational content will be classified into one more sub-subject(s), all subjects will coexist within the knowledge graph.

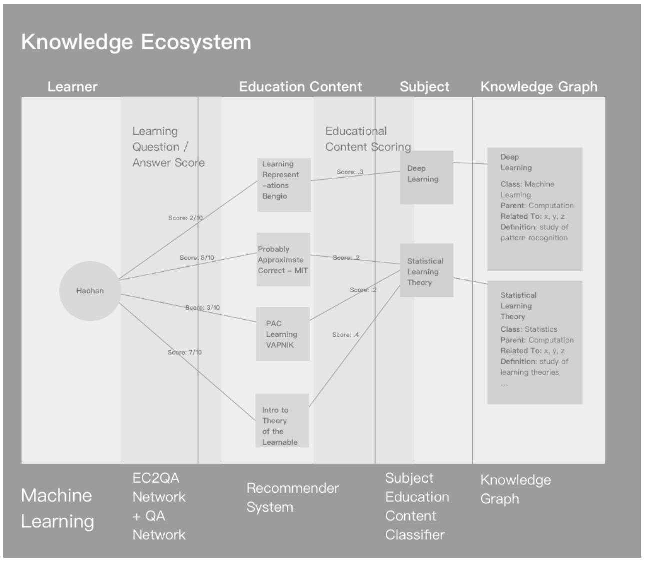


Figure 6: Knowledge Ecosystem

On the other hand, we could also use these subjects that are associated with that piece of content to help create the knowledge footprint for the learner.

3.4 DeepEdu Network

We showed our idea of this novel DeepEdu network earlier because currently, we would have to cobble together multiple networks to make this work. Instead, we can use one network framework, DeepEdu, to solve the problem of generating questions and answer pairs for any given educational content (text, video, image, pdf, etc).

The possible implementation solutions are introduced in the implementation section.

3.5 Knowledge Ecosystem by Example

Now that we are aware of each of the elements, let's talk about how they work in practice.

A learner watches a video titled 'Depression' by Robert Sapolsky.

- The video is classified by a neural network like the following subjects [Neuroscience, Mental Health, Psychology];
- The subjects and the content are then mapped to the **collective knowledge graph** to continuously update it as more information collected about each subject;
- Using **DeepEdu** or similar, a set of questions and answers are generated say every 15* minutes of the video;

- A learner is present with 5 questions to answer and scores 4/5 (80%).
- At the end of the video, the learner records a video summary and is evaluated with a score 7/10 (70%);
- The evaluation network looks at the **semantic and conceptual mutual information** shared between the original content and the learner's video summary to generate a score;
- The system takes up all the scores and outputs a final weighted average score that mapped to the video and also counted at the subject level;
- Eventually, the system will add this content to the learner's **knowledge journey** and update his/her *knowledge footprint* based on the scores;
- A learner now has acquired a new piece of content and updated his/her knowledge footprint and knowledge graph.

e # Implementation

In this section, we set out to answer the following question:

> How might we approach designing such a knowledge ecosystem?

We will take a tour through a relevant set of implementations of the elements within the proposed knowledge ecosystem. We will present the relevant research in machine learning for each of the elements and also propose a new artificial neural network architecture, Educational Content to Question Answer (DeepEdu) network, with the details needed to design such a network.

3.6 Problem Formulation

Building such a knowledge ecosystem is a non-trivial task. As we discussed earlier, here are a few key elements of our ecosystem:

1. **Learning + Feedback [DeepEdu]** – given a learner views a single educational content, reliably evaluate their knowledge and provide feedback for improvement to support learning (credibility, rigour)
2. **Knowledge Graph** – relate any educational content to a concept which belongs to a particular subject, (relatibility, predictability)
3. **Knowledge Journeys** – given educational content, a learner's score on such content, and history of viewing and testing use the knowledge graph to map the learner's journey over time, offer a way for a learner to compare, connect with, and follow another's journey (compare, traverse)
4. **Knowledge Footprint** – given a learner's journey, collapse it into a representative symbol(s) (relatability, stable but evolving system)

Much of element two (#2) and element one (#1) is possible with the recent breakthroughs in the machine learning and deep learning. Elements three (#3) and (#4) may require a different approach to tie the other elements together. We see #3 and #4 as design problems to be approached from the bottom up. In this section, we will mainly focus on the first two (#1 & #2) elements.

3.6.1 Before we start…

As discussed above, recent trends in deep learning have produced state-of-art results on many of the tasks that are needed in our system.

To best illustrate the problem and the possible solutions, we will focus solely on a specific type of educational content educational videos (i.e.Youtube or how-to videos). Keep in mind that our ultimate goal is to apply our approaches to any type of educational content including open texts, digital texts, audio or podcasts.

Let us now explore some of the recent research findings that would enable us to bring our knowledge ecosystem to live.

3.7 Learning + Feedback [DeepEdu]

3.7.1 Problem Formulation

In short, in this section we will be providing some insights into how to solve the following puzzle:

> "how can we take a single educational content (ie. video/podcast) and properly test the learner's knowledge and mastery of the content?".

3.7.2 The Approach

We consider learning + feedback as a key component of our ecosystem which would settle our concerns of passive knowledge consumption and untested knowledge.

Based on the above concerns we'll take a novel approach to ensure that our DeepEdu networks can do the following:

1. Generate a set of questions and answers for any educational content (Domain: Question and Answer Generation)
2. Evaluate closed and open-ended answers (Domain: Automatic Answer Evaluation)
3. Provide a score for the content based on the learner's performance (Domain: Aggregate scoring)

The ideal result is to test the learner's knowledge by asking questions and evaluating responses given a set of concepts and the learner's history knowledge.

Now let's start exploring some solutions that are made possible with the most recent deep learning models.

3.7.3 QG and QA Overview

In previous years, deep learning research has taken up a similar problem titled Question Generation (QG) and Question Answering (QA).

Question Generation (QG) was originally part of NLP. The goal of QG is to generate questions according to some given information. It could be used in many different scenarios (i.e. generating questions for reading comprehension, generating data from large-scale question-answering pairs or even generating questions from images). Earlier approaches to QG mainly used human-crafted rules and patterns to transform a descriptive sentence to a related question. Recent neural network-based approaches represent the state-of-art of most of those tasks and these approaches have been successfully used to solve many other NLP tasks (i.e. neural machine translation, summarization, etc.), as the training optimization studies progress, the stability and performance improvements could enhance our ability to generate relevant questions and answers.

As for generating correct answers to questions, Question Answering (QA), this is one of the most popular research domains in NLP just as well. Recently, QA has also been used to develop dialogue systems and chatbots designed to simulate human conversation.

Traditionally, most of the research used a collection of conventional linguistically-based NLP techniques (i.e. parsing, part-of-speech tagging and coreference resolution. However, with recent advances in deep learning, neural network models have shown promising results. Further improvements (i.e.attention mechanism and memory networks) allow the network to focus on the most relevant facts such that they can reach the new state-of-art performance for QA.

Now we have some basic understanding of these 2 tasks for which we will be expanding more in depth later. Consider the next question:

> “what types of questions & answers would be best to test a learner’s knowledge given a piece of educational content (i.e. a lecture video)”

Example

Imagine a learner is watching a video about hypothesis testing and, midway through the video, the educator shows an example and provides the data needed to test the hypothesis. It would be very beneficial for a learner if during this time his/her knowledge is tested with the following example questions:

1. What is the definition of the p-value? (multiple choice)
2. Is this a 1-sided test? (YES or NO)
3. How would you interpret the p-value in the context of this example? (open-ended)
4. What is the difference between the null hypothesis and alternative hypothesis based on the previous comparison? (open-ended)
5. Give me a quick summary about what you have learned through this video or this example. (Open-ended)

As shown above, we would call questions #1 and #2 the close-ended questions; question #3 and #4 the specific open-ended questions; and question #5 a general open-ended question.

Based on the above information, we can update our question formulation into:

1. Generate close-ended question + answers pairs
2. Generate specific open-ended question + answers pairs
3. Evaluate and comment on the general open-ended answers

In terms of the close-ended questions, the answers can be well defined and evaluated. However, the process might be more complex for the open-ended ones. We will approach each of them here from the current research perspective.

Why deep learning?

As we stated above, deep learning has achieved state-of-art performance in both QG and QA tasks, but how?

If you pay close attention to QG and QA type of problems, you can easily reframe the problem into the one in which the model is expected to learn the relationship between the input (educational content) and the output (meaningful question & answer pairs). In other words, if we can provide the needed data for the model, the model will learn a function that captures the relationship between our input and output, or, appropriately map the educational content to the desired question and answer pairs.

To the best of our knowledge, deep learning is one of the most optimal techniques currently developed for learning such complex representations of complex data such as video lectures.

By definition, machine learning is a subfield of Artificial Intelligence that uses statistical learning techniques to give the machine the ability to learn from the data. It explores the algorithms that can be used to parse data, learn from the data, and then apply what they have learned to make an inference. Deep learning is a subset of machine learning that belongs to the family of representation learning. Inside this family, deep learning is particularly good at sampling the features and having additional layers for more abstract feature learning. All of these special properties are crucial for building our DeepEdu networks.

Moreover, deep learning is known as one of the most flexible machine learning algorithms for learning and mapping a **deep representation** of supervised concepts within the data. Deep neural network architecture can be composed into a single differentiable function and trained end-to-end until it converges. As a result, they can help identify the suitable *inductive biases* catered to the training data.

Specifically, deep learning outperforms other techniques when the training data size is large. In our case, we could easily find a large amount of educational content available on the web.

Having a large amount of training data is very troublesome if you plan to do feature engineering manually. When there is a lack of domain understanding for feature introspection, deep learning is preferable.

In the end, deep learning really shines when it comes to many specialized research problems such as NLP, Visual Recognition and Speech recognition. For creating our DeepEdu, all those domains will be involved.

3.7.4 Question Generation (QG)

Let's begin with question generation (QG) task.

The ideal goal of an automatic question generation is to generate a question Q that is syntactically and semantically correct, relevant to the context and match to a set of answers.

In order to achieve this goal, we need to train an algorithm to learn the underlying conditional probability distribution

$$P_\theta(Q|X)$$

parametrized by θ. In other words, we can think of this problem as the one that requires the model to learn a function (with a set of parameters) θ during the training stage using content-question and/or answer sets so that the probability/likelihood $P_\theta(Q|P)$ is maximized over the given training dataset.

We can also think of this problem as a typical seq2seq (sequence-to-sequence) learning problem since both the input and the output are a sequence of text character that the model needs to process and learn from. But in our case, we would like to take any media type and map it to a character output.

Case Studies

1. In this paper QG-Net: A Data-Driven Question Generation Model for Educational Content [19]. They use a bi-directional LSTM network to process the input context words sequence. Encoding the answer into context word vectors.

QG-Net generates questions by iteratively sampling question words from the conditional probability distribution

$$P(Q|C, A, \theta)$$

where θ denotes a set of parameters. In order to construct the probability distribution, they first create a **context reader** that processes each word c_j in the input context and turns it into a fix-sized representation h_j

Meanwhile, they also have a **question generator** that generates the question text word-by-word, given all context word representation and the set of question words in previous time steps.

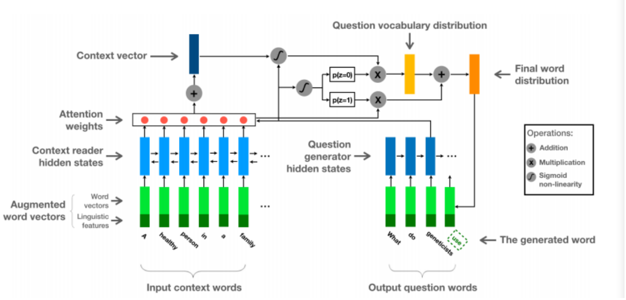


Figure 7: ma

As for quantitative evaluation, they aimed to minimize the difference between the generated question and the true question in the training set during training using the standard back-propagation through time with mini-batch stochastic gradient descent algorithm to learn the model parameters. To ensure the performance, they employ *teacher forcing* procedure for training the LSTM with a standard implementation of beam search, a greedy but effective approximation, to exhaustively search and select the top 25 candidate output question sentences. The final one would be the one with the lowest negative log likelihood.

The high-level QG-Net architecture is as below:

2. In this paper Topic-based Question Generation [**???**], they propose a topic-based question generation algorithm. The algorithm is able to take in an input sentence, a topic and a question type and generate a word sequence related to the topic, question type and the input sentence.

They formulate a conditional likelihood objective function as mentioned before to the model to learn.

Also, they go through a few general frameworks that have been employed for solving a similar problem.

- The first one is the *seq2seq model* that uses a bidirectional LSTM as the encoder to encode a sentence and an LSTM RNN (Recurrent Neural Network) as the decoder to generate the target question.
- The second approach is *question pattern prediction and question topic selection algorithms.* They take in an automatically selected phrase Q and fill this phrase into the pattern that was predicted from pre-mined patterns.
- The last approach is *multi-source seq2seq learning* which aims to integrate information from multiple sources to boost learning.

3. In this paper A Framework for Automatic Question Generation from Text using Deep Reinforcement Learning [10], they implement a reinforcement learning(RF)

framework that consists of a generator and an evaluator for this task.

They refer to *the generator* part of the model as the *agent* and the *action* of the agent is to generate the next work in the question. The probability of decoding a word is

$$P_\theta(word)$$

with a stochastic policy.

The evaluator part of the model then assigns a *reward* to the output sequence predicted using the current *policy* given by the generator. Based on the reward assigned by the evaluator, the generator updates and improves its current policy. In short, the goal in RL-based question generation is to find a policy that can maximize the sum of the *expected return* at the end of the sequence generation.

** Summary **

In this QG section, we have discussed 3 different algorithms. Based on our learning, we conclude that a generative seq2seq model might be a suitable option for this task. As for our objective function, we should be formulating a conditional probability distribution that is conditioned on the provided content (i.e. the video) and answers. As suggested, we can use a bi-directional LSTM RNN as the encoder to encode the content and an LSTM RNN as the decoder to generate the question.

3.7.5 Question Answering (QA)

Now, let's move on to our question answering (QA) task. The general goal of QA is to predict the answer to a question based on the information found in the passage, given a passage and a question. By solving this task, our DeepEdu network should be able to easily evaluate the answer provided by learners and achieve full automation of learning + feedback cycle.

Here is an overview of a basic QA model's implementation [9]:

1. Build representation for the passage and the question separately;
2. Incorporate the question information into the passage;
3. Get the final representation of the passage by directly matching it against itself;
4. Generate the answer.

And the typical mechanisms applied for solving such a problem include:

- Embedding
- Encoder Decoder
- Attention Mechanism

Close-ended Questions

Visual Question Answering (VQA)

VQA is a challenging research problem that focuses on providing a natural language answer given any image and any free-form natural language question. As we are focused on video educational content first, our problem will include both NLP and visual recognition tasks. We figure VQA should be a great approach to start with.

Case Studies

1. In this paper Deep Attention Neural Tensor Network for Visual Question Answering [**???**], they propose a novel *deep attention neural tensor network* that can discover the joint correlation over images, questions and answers with tensor-based representation.

As for their workflow, the model one of the pairwise interaction (i.e. between image and question) by **bilinear features**, which is further encoded with the third dimension (i.e. answer) to be a triplet using a bilinear tensor product. During this step, the model takes in a question + a corresponding image + candidate answers as the input. A CNN (convolutional neural network) and a GRU RNN are used for extracting feature vectors and question respectively. Then the representation is passed on as a *multi-modal feature* and integrated by a **bilinear pooling** module. Moreover, they decompose the correlation of triplets by their question and answer types with a **slice-wise attention module** on the tensor to select the most discriminative reasoning process inference.

In the end, they optimize the proposed network by learning a label regression with KL-divergence losses. They claimed that these techniques enable them to do scalable training and fast convergence over a large number of answer set. During the inference stage, they feed the embeddings of all candidate answer into the network and then select the answer which has the biggest triplet relevance score as the final answer.

The high-level network architecture is as follows:

2. In this paper Question Type Guided Attention in Visual Question Answering, they propose a model called **Question Type-guided Attention (QTA)**. This model utilizes the information of question type to dynamically balance visual features from both top-down and bottom-up orders.

Also, they propose a *multi-task extension* that is trained to predict question types from the lexical inputs during training which generalizes into applications that lack question type, with a minimal performance loss.

As for their main contribution, they focus on developing an attention mechanism that can exploit high-level semantic information on the question type to guide the visual encoding process.

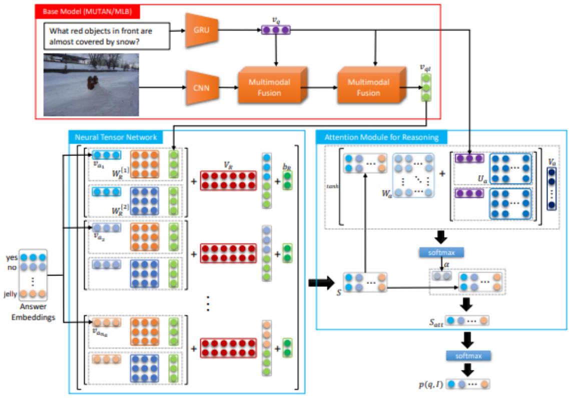


Figure 8: Deep Attention Neural Tensor Network

Specifically, they introduce a novel VQA architecture that can dynamically gate the contribution of ResNet and Faster R-CNN features based on the question type. In turn, it allows them to integrate the information from multiple visual sources and obtain gains across all question types.

3. In this paper Multi-Turn Video Question Answering via Multi-Stream Hierarchical Attention Context Network [15], they propose a hierarchical attention context network for context-aware question understanding by modelling the hierarchically sequential conversation context structure. They incorporate the multi-step reasoning process into **the multi-stream hierarchical attention context network** to enable the progressive joint representation learning of the multi-stream attentional video and context-aware question embedding.

To construct their dataset, they collected conversational video question answering datasets from YouTubeClips and TACoS-MultiLevel. The first dataset has 1987 videos and the second dataset has 1303 videos. They invited 5 pairs of crowd-sourcing workers to construct 5 different conversational dialogues. In total, they collected 37228 video question answering pairs for TACoS-MultiLevel data and 66806 for YouTube-Clips data.

4. In this paper MovieQA: Understanding Stories in Movies through Question-Answering [17], they construct a new dataset ** MovieQA** dataset that can be used to evaluate automatic story comprehension from both video and text.

They collect 408 subtitled movies and obtained their extended summaries in the form of **plot synopses**(movie summaries that fans write after watching the movie) from Wikipedia. They used plot synopses as a proxy for the movie. They have annotators create both quizzes and answers pairs by referring to the story plot. Time-stamp is also attached to each question and answer pair.

In the second step of data collection, they used the multiple-choice answers and question collected as the input to show to a different group of annotators. By doing so, annotators could re-formulate the question and answers for a sanity check.

** Summary **

By going through the previous examples, we can see that VQA and a few similar algorithms are designed to efficiently process image and text input while making the inference based on it.

First, the key components for creating and training a VQA model include feature selection, feature pooling and a specially designed attention mechanism. The input of the model is typically a video clip + question + answer pairs. A CNN and sometimes an RNN is needed for such a task.

Another important learning is that we can follow the steps described in MovieQA example to collect and annotate our training data by asking crowd-sourcing workers to construct the question and answer pairs. Also, more advanced algorithms like multi-stream hierarchical attention context network might be necessary for a better performance.

Dual Question-Answering Model

Both Question Generation(QG) and Question Answering(QA) are two well-defined sets of problems that aim to either infer a question or an answer given the counterpart based on the context. However, they are usually explored separately despite their intrinsic complementary relationship. However, our DeepEdu network needs to take on both roles simultaneously.

How might we solve this issue?

Let's look into a few algorithms are designed for this.

Case Studies

1. In this paper Dual Ask-Answer Network for Machine Reading Comprehension [20], they present a model that can learn question answering and question generation simultaneously. They tie the network components that playing similar roles into 2 tasks to transfer cross-task knowledge during training.

Then *the cross-modal interaction* of a question, context and answer are captured with a pair of **symmetric hierarchical attention** processes.

The high-level architecture of the model is illustrated as below:

In short, the model is composed of the following components: embedding layer, encoding layer, attention layer and the output layer. The model is fed with a question-context-answer triplet (Q, C, A) and the decoded Q and A from the output layer. Their loss function consists of two parts:

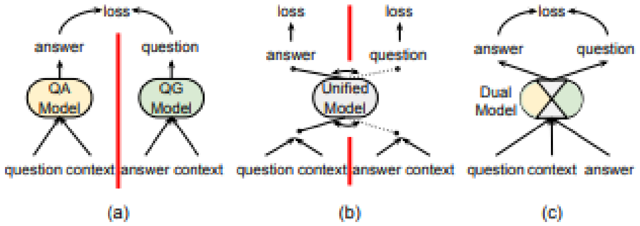


Figure 9: Dual Ask-Answer Network 1

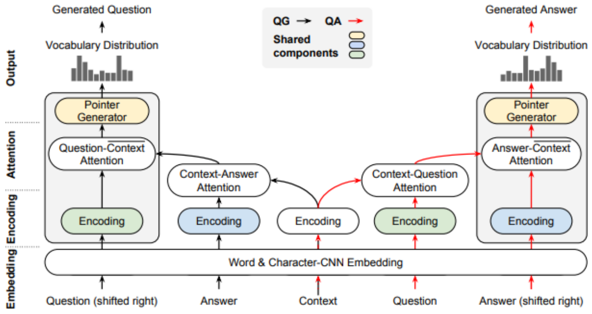


Figure 10: Dual Ask-Answer Network 2

- negative log-likelihood loss
- a coverage loss (penalize repetition of the generated text)

2. In this paper Harvesting Paragraph-Level Question-Answer Pairs from Wikipedia [6], they apply their question-answer pair generation system to 10000 top-ranking Wikipedia articles and create over a million question-answer pairs.

In their task formulation part, they clarify that they break this task into 2 sub-tasks:

- candidate answer extraction
- answer-specific question generation

To complete the tasks, they first identify a set of question-worthy candidate answer set $ans = (A1, A2, ...Ai)$. For each candidate answer A_i, they then aim to generate a question Q -a sequence of tokens $y1, y2, ...yn$ - based on the sentence S that contains candidate A_i such that - Q asks about an aspect of A_i (of potential interest to a human) - Q might rely on information from sentences that precedes S in the paragraph.

Mathematically, they compose a function

$$Q = argmax_Q P(Q|S, C)$$

.

3. In this paper Visual Question Generation as Dual Task of Visual Question Answering [11], they propose an end-to-end unified model, **Invertible Question Answering (iQAN)** to introduce question generation as a dual task of question answering to improve VQA performance.

In achieving their goal, they leverage the **dual learning** framework that is proposed in machine translation area initially, which uses $A-to-B$ and $B-to-A$ translation models to form two closed translation loops and let them teach each other through a *reinforcement learning process*.

In their VQA component, given a question q, an RNN is used for obtaining the embedded feature $\mathbf{q}$, and CNN is used to transform the input image v into a feature map. A *MUTAN-based attention module* is then used to generate a question-aware visual feature v_q from the image and the question. Later, another *MUTAN fusion module* is used for obtaining the answer feature $\hat{a}$

4. In this paper A Unified Query-based Generative Model for Question Generation and Question Answering [16], they propose a query-based generative model for solving both tasks. The model follows the classic *encoder-decoder* framework. The **multi-perspective matching encoder** that they are implementing is a bi-directional LSTM RNN model that takes a passage and a query as input and perform query understanding by matching it with the passage from multiple perspectives;

The decoder is a**attention-based LSTM RNN** model with *copy and coverage* mechanism. In the QG task, a question will be generated from the model given the passage and the target answer; whereas in the QA task, the answer will be generated given the question and the passage.

They also leverage a *policy-gradient reinforcement learning* algorithm to overcome *exposure bias* (a major problem resulted from sequence learning with cross-entropy loss function).

The case both QG and QA tasks into one process by firstly matching the input passage against the query, then generating the output based on the matching results.

As for the training, they first pre-train the model with cross-entropy loss and then fine-tune the model parameters with policy-gradient reinforcement learning to alleviate the exposure bias problem. They end up adopting a similar sampling strategy as the scheduled sampling strategy for generating the sampled output during the reinforcement learning process.

** Summary **

As mentioned earlier, QG and QA tasks are intrinsically bounded and one cannot find the solution for either of them without taking the other party into account.

In this section, we have discussed some approaches that many groups of people have taken to help the machine operate on both tasks simultaneously.

For our problem, it is very motivating to see the current progress and how much we can learn from their approaches. In sum, our general setup is most similar to the dual learning framework. We need to tie QG and QA together and also map it to concepts. In the first diagram of the section, we can see that they connect the loss function from both sides of the model and it is very similar to the strategy adopted by GAN (Generative adversarial network). Some advanced mechanisms are proposed as well for effectively solving these tasks (i.e. symmetric hierarchical attention and policy-gradient reinforcement learning algorithm.

Open-ended Question

Problem Formulation

> Open-ended questions bring clarity.

As we mentioned above, the open-ended question could be roughly split into two categories. A general open-ended question or a specific open-ended question.

Technically speaking, these two categories are not that particular distinct since both problems require the system to draw a conclusion based on the context and question provided. As for the answer, it is allowed to have a pretty high degree of freedom. Therefore, our system should be able to evaluate the answer with relatively flexible rules or standards including flexibility in media type.

Based on our assumptions, we will combine these two problems into one for our investigation.

It may appear unapproachable at first glance to teach a system to generate correct answers for and evaluate open-ended answers. Again, we need to reframe our problem and then break it apart.

Based on our research, we believe it is helpful to think of this type of issue as a particular type of QA problem; the difference is that after the QA procedure, we need to match and evaluate the answers generated by the machine and the learner such that we can provide an adequate evaluation.

Let's start by looking at an existing knowledge evaluation system that has been used for grading the essays automatically - **Automated essay scoring (AES)**. AES focuses on automatically analyzing the quality and assigning a score to a given writing. AES systems rely not only on grammars, but also on more complex features such as semantics, discourse and pragmatics.

It has four general types:

- Essay Grade: it is known as the first AES system.
- Intelligent Essay Assessor: it is using Latent Semantic Analysis features
- E-rater: it has been used by the ETS to score essay portion of GMAT
- IntelliMetric: it is developed and used by the College Board for placement purposes.

Below are some research findings we consider as useful for our unified goal.

Case Studies

1. In this paper Neural Automated Essay Scoring and Coherence Modeling for Adversarially Crafted Input [7], they develop a network that can effectively learn connectedness features between sentences and propose a framework for integrating and jointly training the local coherence model with a state-of-art AES.

They examine the robustness of the AES model on adversarially crafted input and specifically focus on input related to local coherence; A local coherence model can evaluate the writing based on its ability to rank coherently ordered sequence of sentences higher than their counterparts.

The models they used are *Local Coherence (LC)* model and LSTM AES model. The first model has 2 main parts: sentence representation and clique representation; and the second model is a combined model that does vector concatenation and joint learning.

2. In this paper Open-Ended Long-form Video Question Answering via Adaptive Hierarchical Reinforced Networks [21], they study the problem of open-ended video question answering from the viewpoint of the *adaptive hierarchical reinforced encoder-decoder network* learning.

They present the adaptive hierarchical encoder network to learn the joint representation of the long-form video contents according to the question with adaptive video segmentation. They also develop the reinforced decoder network to generate the neural language answer for open-ended video question answering. Meanwhile, they construct a large-scale dataset for open-ended long-form video QA and validate the effectiveness of the proposed method.

The framework of **Adaptive Hierarchical Reinforced Networks** is are below:

The first part of the model is the hierarchical encoder networks that learn the joint representation of multimodal attentional video and textual question with adaptive video segmentation.

The second part is the reinforced decoder networks that generate the natural language answers for open-ended video question answering.

3. In this paper, Multi-turn Dialogue Response Generation in an Adversarial Learning Framework [13], they propose an adversarial learning approach that can generate multi-turn dialogue responses. The network framework that they introduce is called *hredGAN* that is

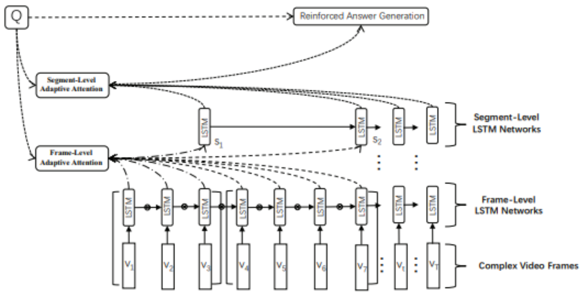

Q
Reinforced Answer Generation
Segment-Level Adaptive Attention
Frame-Level Adaptive Attention
LSTM
s_1
s_2
Segment-Level LSTM Networks
Frame-Level LSTM Networks
v_1
v_2
v_3
v_4
v_5
v_6
v_7
v_t
v_T
Complex Video Frames

Figure 11: Open-Ended Long-form Video QA Network

based on *conditional GANs*. The generator part of the model is a modified **hierarchical recurrent encoder-decoder network (HRED)** and the discriminator is a word-level bi-directional LSTM RNN that shares context and word embedding with the generator.

During the inference step, *noise sampling* is conditioned on the dialogue history and is used to perturb the generator's latent space for generating possible responses. The final response is the one ranked the best by the discriminator.

In sum, their hredGAN combines both generative and retrieval-based multi-turn dialogue systems to improve the model's performance. One of the special design of the model is that the generator and the discriminator share the context and word embedding and this allows for joint end-to-end training using back-propagation.

** Summary **

Based on our limited research, we find that it is achievable to generate the answers for open-ended questions based on the educational video and provide appropriate feedback/rating based on the current techniques. The first paper presents a newly developed AES model that rates learner's writing by taking into account a specified metrics. It also demonstrates a possible approach to enhance any given AES model by training it with the adversarially crafted input.

In the second paper, we investigate a network that is able to answer the open-ended questions based on the video and a given question. The Adaptive Hierarchical Reinforced Networks they proposed is composed of hierarchical encoder networks and the reinforced decoder networks. It is possible for us to adopt their general framework trained with our educational video data specifically.

Similar to the second paper, the last paper indicates that we can generate responses conditioned on the context. By leveraging conditional GAN framework, their model performs very well on this task.

3.7.6 Summary of Learning and Feedback Networks

Based on our previous discussion, we find that both QG and QA (including VQA) tasks have been well-studied. A number of specifically designed algorithms were presented and proved effective for solving these problems.

There is also plenty of research has also been done in open-ended question answering realm. Though the performance may not be guaranteed, some techniques presented above are greatly relevant and thought-provoking such as AES model and open-ended question answering networks.

Current research is promising but we need more research and innovation in this area.

However, we are confident that by combining some techniques introduced before to create such a coherent DeepEdu network is not that far-fetched.

3.7.7 Datasets and Annotation Suggested

In order to approach this problem from scratch, we need to create our own dataset for which we will provide some related resources to start with:

1. YouTube-8M Dataset [2]. This is a large-scale labelled video dataset that consists of millions of YouTube video IDs, with high-quality generated annotations from a diverse vocabulary of 3800+ visual entities. As you can see from its introduction, it comes with precomputed audio-visual features from billions of frames and audio segments. In short, we can expect the following content from this dataset:

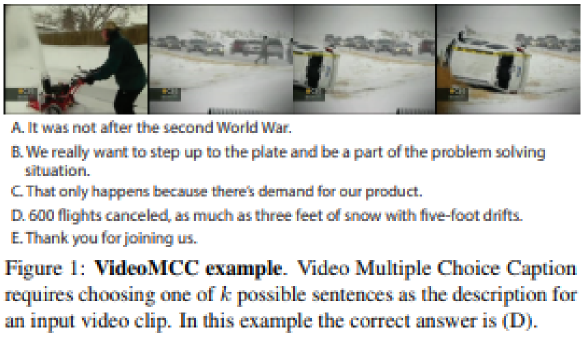
- the dataset consists of 6.1M videos URLs, labelled with a vocabulary of 3863 visual entities
- the video-level dataset comes out to be 18 GB in size, while the frame-level features are approximately 1.3 TB
- it comes with pre-extracted audio & visual features from every second of video.

Though the video content is not limited to the education category, we can still use it to get a strong baseline model.

Naturally, the next step would be to constrain our model to train particularly on educational content. The data needed for training may include the raw video clip, annotation/caption of the whole video content, and question + answer pairs (including the timestamps).

2. In this study Video Captions for Online Courses: Do YouTube's Auto-generated Captions Meet Deaf Students' Needs? [14], they studied auto-generated captions generated on YouTube online courses. They find that, on average, there were 7.7 phrase errors per minute of a total 68 minutes video caption. It implies that we cannot rely on the automated caption of the video, a lot more manual efforts are needed to fix this problem.

Some other resources that might help:



A. It was not after the second World War.

B. We really want to step up to the plate and be a part of the problem solving situation.

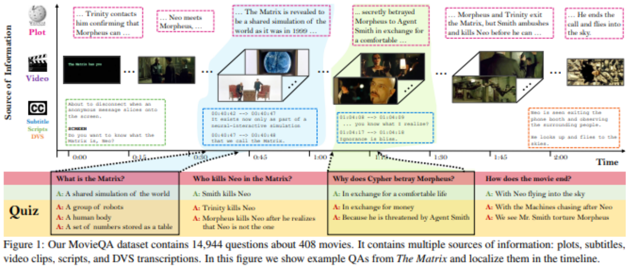
C. That only happens because there's demand for our product.

D. 600 flights canceled, as much as three feet of snow with five-foot drifts.

E. Thank you for joining us.

Figure 1: **VideoMCC example.** Video Multiple Choice Caption requires choosing one of k possible sentences as the description for an input video clip. In this example the correct answer is (D).

Figure 12: VideoMCC


Source of Information
Plot
Video
Subtitle
Scripts
DVS
... Trinity contacts him confirming that Morpheus can ...
... Neo meets Morpheus, ...
... The Matrix is revealed to be a shared simulation of the world as it was in 1999 ...
... secretly betrayed Morpheus to Agent Smith in exchange for a comfortable ...
... Morpheus and Trinity exit the Matrix, but Smith ambushes and kills Neo before he can ...
... He ends the call and flies into the sky.
About to disconnect when an anonymous message slices onto the screen.
SCREEN
Do you want to know what the Matrix is, Neo?
00:40:42 --> 00:40:47
It exists now only as part of a neural-interactive simulation
00:40:47 --> 00:40:48
that we call the Matrix.
01:04:08 --> 01:04:09
... you know what I realize?
01:04:17 --> 01:04:18
Ignorance is bliss.
Neo is seen exiting the phone booth and observing the surrounding people.
He looks up and flies to the skies.
0:00
0:15
0:30
0:45
1:00
1:15
1:30
1:45
2:00
Time
Quiz

What is the Matrix?	Who kills Neo in the Matrix?	Why does Cypher betray Morpheus?	How does the movie end?
A: A shared simulation of the world	A: Smith kills Neo	A: In exchange for a comfortable life	A: With Neo flying into the sky
A: A group of robots A: A human body A: A set of numbers stored as a table	A: Trinity kills Neo A: Morpheus kills Neo after he realizes that Neo is not the one	A: In exchange for money A: Because he is threatened by Agent Smith	A: With the Machines chasing after Neo A: We see Mr. Smith torture Morpheus

Figure 1: Our MovieQA dataset contains 14,944 questions about 408 movies. It contains multiple sources of information: plots, subtitles, video clips, scripts, and DVS transcriptions. In this figure we show example QAs from *The Matrix* and localize them in the timeline.

Figure 13: MovieQA_1

3. VideoMCC. They formulate *Video Multiple Choice Caption (VideoMCC)* as a way to assess video comprehension through an easy-to-interpret performance measure. In their paper VideoMCC: a New Benchmark for Video Comprehension [18], they propose to cast video understanding in the form of multiple choice tests that assess the ability of the algorithm to comprehend the semantics of the video. An example is as below:

4. As what we have covered earlier, in this paper MovieQA: Understanding Stories in Movies through Question-Answering, they introduce a new dataset called MovieQA that can evaluate automatic story comprehension from both video and text.

The 2 figures below can offer a better illustration:

5. In this paper Video Description: A Survey of Methods, Datasets and Evaluation Metrics [1], they present multiple methods, datasets and evaluation metrics for video description task in a comprehensive survey.

Q: How does E.T. show his happiness that he is finally returning home?
A: His heart lights up

Q: Why do Joy and Jack get married that first night they meet in Las Vegas?
A: They are both vulnerable and totally drunk

Q: Why does Forrest undertake a three-year marathon?
A: Because he is upset that Jenny left him

Q: How does Patrick start winning Kat over?
A: By getting personal information about her likes and dislikes

Figure 2: Examples from the MovieQA dataset. For illustration we show a single frame, however, all these questions/answers are time-stamped to a much longer clip in the movie. Notice that while some questions can be answered using vision or dialogs alone, most require both. Vision can be used to locate the scene set by the question, and semantics extracted from dialogs can be used to answer it.

Figure 14: MovieQA_2

6. Lastly, in this paper QuAC : Question Answering in Context [4], they present a *QuAC* dataset for QA task in Context that contains 14K information-seeking QA dialogs such as a student who poses a sequence of freeform question to learn as much as possible about a hidden Wikipedia text or a teacher who answers the questions by providing short excerpts from the text. Inspired by their idea and effort, we imagine that it is might be possible to develop a system that can also allow the learners to pause the video and to ask an information-seeking question. As a result, our system will find the answer based on the current content.

3.8 Knowledge Graph

Learning is a journey of knowledge accumulation and experience To create an adequate knowledge ecosystem requires us to first figure out what 'knowledge' really is and how to best represent it. *Knowledge Graph* is used as one of the tools to help make sense of what type or category acquired 'knowledge' is and what it is related to. You can think of knowledge graphs as a collection of relational facts, where each fact states that a certain relation holds between 2 entities.

In this section, we will be discussing how we can construct our education specific knowledge graph from scratch.

3.8.1 What is a graph?

Graphs are networks of dots and lines - Graph Theory (Dover Books)

Mathematically speaking, graphs are mathematical structures used to model pairwise relations between objects. A graph in this context is made of vertices, nodes, or points which are connected by edges, arcs or lines. Typically a graph consists of two sets. A set of vertexes and a set of edges as below:

$$GRAPH_{v,e} = \begin{pmatrix} v_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ e_{2,1} & e_{2,2} & \cdots & e_{2,n} \end{pmatrix}$$

3.8.2 What is a Knowledge Graph(KG)?

As for the knowledge graph, it is a graph representation of a knowledge base. One of the most popular knowledge graphs is the multi-relational graph used by Google and its services to enhance the search engine's results with information gathered from a variety of sources. Per Wikipedia, Google's Knowledge Graph [8] uses a graph database to provide structured and detailed information about the topic in addition to a list of links to other sites.

In general, a knowledge graph represents a knowledge domain. It connects the objects and facts of different types in a systematic way. Knowledge graphs encode knowledge arranged in a network of nodes and links rather than tables and columns. With knowledge graphs, people and machines can easily capture and utilize a dynamically growing semantic network of facts about things. In other words, we can use it to connect the facts related to people, processes, applications, data and many other custom objects as well as their relationships among them with a structured knowledge graph.

Also, plenty of successful applications show that people have applied knowledge graph successfully and efficiently to a variety of domains (i.e. to support semantic search(i.e. Google's Knowledge Graph), personal assistant(i.e. Apple's Siri) and deep question answering (i.e. Wolfram Alpha and IBM's Watson).

3.8.3 Problem Formulation

Given we've implemented the learning and feedback module, knowing where a given piece of education content fits into the knowledge space is a vital task if we want the knowledge footprint to make a learner predictable to others as well as being able to recommend new educational content that the learner can take on successfully. We will need the following things to connect our educational content:

- A classifier to take a piece of educational content
- A hierarchical algorithm that can map the relationship of the related content based on their hierarchy in the knowledge graph
- (optional) A probabilistic algorithm (i.e. Bayesian network to map the strength/significance of their relationship with each other

In our case, a graph dedicated to education should do the following:

- Provide flexibility to add new subjects
- Connect related subjects to each
- Map concepts with the subject
- Connect concepts related to the content

As shown before, here is how our graph will be used to support our ecosystem as a whole:

3.8.4 Automatic Knowledge Graph Construction

Classic knowledge representation techniques allow a knowledge engineer to create rules that can be interpreted by a reasoner to infer new or missing triples(subject, predicate, object). These rules are usually expressed through an ontology which allows for the propagation of properties from top classes to the lower classes.

However, we are looking for solutions that can allow us to complete our educational knowledge graph construction process automatically. Based on our research, generic knowledge graphs usually cannot sufficiently support many domain-specific applications (i.e. education) and finding the representation of the graph to feed the triples into a machine learning algorithm is still an open area of research.

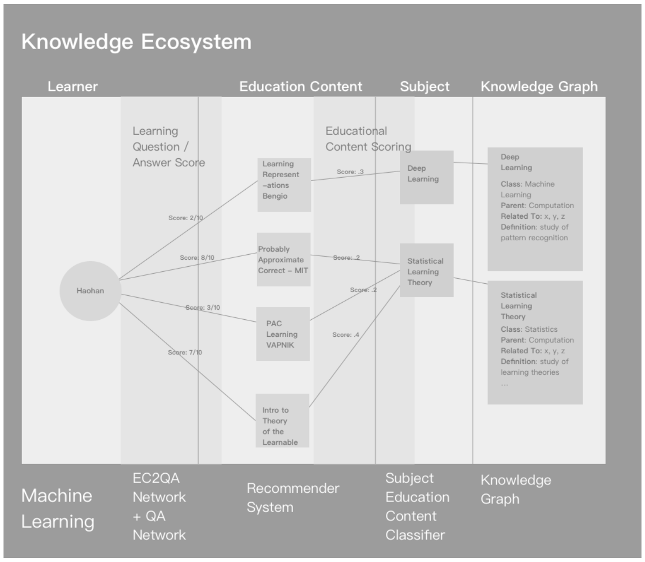


Figure 15: Knowledge Ecosystem

As a start, let's focus on how we can automate our knowledge graph construction process.

Key Components

Here are some key components that worth highlighting before we dive deep into the possible solutions:

1. **Entity recognition** that aims to extract the concept of interest from structured or unstructured data;
2. **Relation identification** that leverages on the semantic meaning of data.

3.8.5 Case Studies

1. In this paper KnowEdu: A System to Construct Knowledge Graph for Education [3], the authors propose a system, titled **KnowEdu**, that can automatically construct a knowledge graph for education. In short, the system is able to extract concepts of subjects or courses and then identifies the educational relations between the concepts.

More importantly, it adopts the *neural sequence labelling algorithm* on pedagogical data to extract instruction concepts. They then employ *probabilistic association rule mining* on learning assessment data to identify the significance of each of the relations.

In sum, their system consists of the following modules:

- Instructional Concept Extraction Module to extract instructional concepts for a given subject or course.
- Educational Relation Identification Module to identify the educational relations that interlink instructional concepts to assist the learning and teaching process directly.

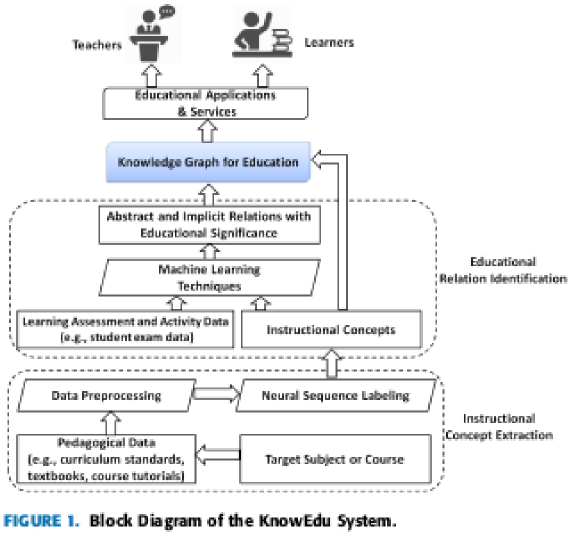

Teachers
Learners
Educational Applications & Services
Knowledge Graph for Education
Abstract and Implicit Relations with Educational Significance
Machine Learning Techniques
Educational Relation Identification
Learning Assessment and Activity Data (e.g., student exam data)
Instructional Concepts
Data Preprocessing
Neural Sequence Labeling
Instructional Concept Extraction
Pedagogical Data (e.g., curriculum standards, textbooks, course tutorials)
Target Subject or Course
FIGURE 1. Block Diagram of the KnowEdu System.

Figure 16: knowedu

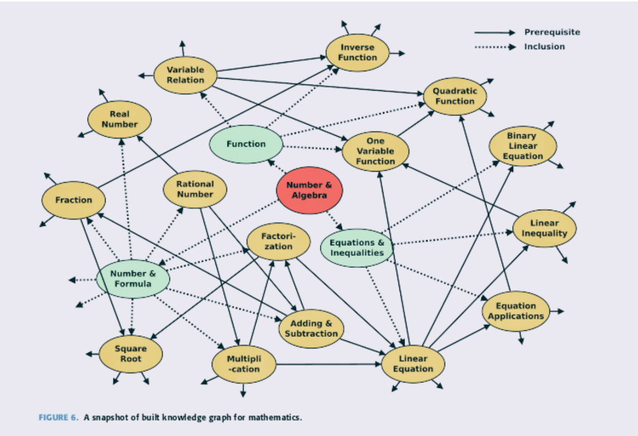


Figure 17: knowledge_graph

Below is a block diagram of the KnowEdu System.

Here are some algorithms they employ, they use a *conditional random field (CRF)* model for entity or terminology recognition task. They adopt a neural network, or more particularly Gated recurrent unit network (GRU) architecture for neural sequence labelling on educational entity extraction task.

In terms of relation identification, they implement probabilistic association data mining techniques on learning assessment data and accomplish the task of educational relation identification.

A snapshot of the knowledge graph for mathematics generated by knowedu system.

Below are some papers related to Knowledge Graph embedding which is used to embed components of a KG including entities and relations into continuous vector space so as to simplify the manipulation while preserving the inherent structure of a KG.

As for its benefits, it can help with numerous downstream tasks (i.e. KG completion and relation extraction, and hence be used to drastically improve the knowledge acquisition and mapping speed for our KG.

2. In this paper Generalized Embedding Model for Knowledge Graph Mining [12], they present a model for learning neural presentation of generalized knowledge graphs using a novel *multi-shot unsupervised neural network* model, called the **Graph Embedding Network (GEN)**. This model is able to learn different types of knowledge graphs from a universal perspective and it provides flexibility in learning representations that work on graphs conforming to different domains.

In developing their model, they extend the traditional one-shot supervised learning mechanism by introducing a *multi-shot unsupervised learning* framework where a 2-layer MLP network for every shot. This framework can, in turn, be used to accommodate both homogeneous and heterogeneous networks.

3. In this paper Probabilistic Knowledge Graph Embeddings [**???**], they explored a new type of embedding model that can link prediction in relational knowledge graph. They are set out to solve a question with their approach that even large knowledge graphs typically contain only a few facts per entity, leading effectively to a small data problem where parameter uncertainty matters. As for the solution, they suggest that the knowledge graphs should be treated within a *Bayesian* framework.

In short, they present a *probabilistic interpretation* of existing knowledge graph embedding models. By reformulating the models like ComplEx and DistMult, they construct the generative models for relational facts.

They also apply *stochastic variational inference* to estimate an approximate posterior for each entity and relation embedding in the knowledge graph. By doing so, they can estimate the uncertainty, but more importantly, it allows them to use gradient-based hyperparameter optimization with stochastic gradient descent on the optimized variational bound.

As a result, their model shows experimentally new state-of-art results in link prediction task.

3.8.6 Summary

A significant amount progress has been made in automating knowledge graph using deep learning and other machine learning techniques based on our research.

The first paper introduces a system that almost exactly matches our goal. They also demonstrate the current progress and possible solutions for solving each of the obstacles to developing an education based knowledge graph. In sum, for the instructional concept extraction task, they use a CRF model and a neural sequence labelling algorithm and they have proved effective for the task. They adopt a probabilistic data mining technique in learning assessment data to approximate the relations with educational significance.

The last 2 papers demonstrate the progress that has been made in KG embedding learning domain. As mentioned above, it is one of the most effective methods in representing knowledge graphs. It is an indication that we should implement this approach to represent our knowledge graph for future knowledge acquisition, maintenance and continuous manipulation purposes.

Key Takeaways

1. In terms of entity recognition task, We need to first get the data from the reliable open semantic sources (i.e. Wikipedia or Freebase. Or we can crawl the data on our own to collect more high-quality training data.

2. Next, we need to apply the models to extract and map the entities. Besides the models mentioned above, there are plenty of great GitHub repositories that we can refer to help us with this task. Or we can look for the tools developed for this task (i.e. *node.js* and *Wolfram Mathematica embedded symbolic functions* or just simply build your own with *TensorFlow*. Some useful techniques that we should keep in mind are NLP and semantic data tagging/ labelling techniques.

3. After the entity extraction and mapping, we need one or more algorithms to learn and capture the relationship among all entities (i.e. a probabilistic model.

4. The last but the not the least, visualize our map.

3.8.7 Datasets and Annotation Suggested

1. Knowledge Vault: A web-scale approach to probabilistic knowledge fusion. In this paper Knowledge Vault: A Web-Scale Approach to Probabilistic Knowledge Fusion [5], they introduce the **Knowledge Vault** that combines extraction from the Web content (obtained through analysis of text, tabular data, page structure, and human annotation) with prior knowledge derived from existing knowledge repositories.

They employ a supervised machine learning models for fusing these distinct information sources. As a result, their system can automatically construct a web-scale probabilistic knowledge base.

2. Google Knowledge Graph Search API.

3. Wikipedia

4. Freebase

3.9 Knowledge Journeys

A collective knowledge graph is our ecosystem's backbone. It can be applied universally to some extent, but a learner's learning journey is highly personal and unique at the individual's level. Every learner enters the system with their own set of problems that they are curious about and their specific missions towards the mastery. As a result, their knowledge journeys should vary significantly. As for our role, we need to ensure we have a system that can handle these high degrees of freedom. Or more ideally, to adaptively represent and learn from a learner's history, interests, and whom they are related to best support their knowledge acquisition adventure.

3.9.1 Problem Formulation

Below are few key makeups of our knowledge journeys:

1. First, we need to have a system that can match all the content that a learner has acquired back to the collective educational knowledge graph;

2. Generate their personal knowledge graph with the timestamps on each content and the subject(s) it belongs to;

3. Unfold the timestamps and map the previous generated static knowledge graph onto a timeline and display a learner's knowledge as a personal knowledge acquisition storyline/gallery;

4. Adaptively update the journey along with a learner's learning progress.

As the outcome, a learner's knowledge journey will be available anytime for the learner to look through and it makes it possible for them to easily go back in time to review a particular piece of content they have learned and for others to unfold their learning journey along the timeline for more details.

3.10 Knowledge Footprint

Just a quick review of the concept, a learner's knowledge footprint is going to be represented as one or more badges or symbols that act as a thumbnail view of a learner's knowledge profile at a specific time point. In a sense, it will continuously evolve and be modified when a learner proceeds his/her learning journey. By constructing such symbol(s), a learner can easily understand their current knowledge profile relative to their past's and to other's and to timely make an adjustment to alter their future learning plan.

As for the implementation, we would consider employing the unsupervised learning and dimensionality reduction techniques to help uncover and represent the visible and invisible dimensions of a learner's knowledge journey. In turn, the model can consolidate and collapse the journey into one or more symbols to accurately represent a learner's current knowledge profile. Still, it is not a trivial task and we believe it is going to be a teamwork that requires people like machine learning and deep learning researchers, educators, and designers to work collaboratively for the best possible outcomes.

Note that we will not discuss the detailed research findings or implementation steps of Knowledge Journeys and Knowledge Footprint elements in this essay. We will leave these areas for our future research.

3.10.1 Possible Next Steps

As a reasonable next step, we would also like our system to be as personalized as possible so as to provide guidance for a learner along their knowledge journey, given that our system has already encoded and possessed the information of a learner's knowledge footprint and journey.

In this case, a recommender system might be a desirable choice for handling such a job since it is an intuitive line of defence against consumer over-choice given the ever-growing educational content available on the web.

4. NEXT STEPS

4.1 Overview

We have now proposed one approach to better represent a modern individual's knowledge by taking the world of unstructured educational content (Youtube videos, Medium articles, etc), classifying it by its concepts which belong to one or more subjects from an education-based knowledge graph.

We surveyed some of the relevant machine and deep learning research, proposed the Educational Content to Questions and Answers (DeepEdu) network, a novel neural network for taking in any type of educational content and generating a set of questions, answers, and the evaluation of a learner's knowledge understanding without the need for the educator be involved.

We then proposed to tie these components together as an adaptive knowledge ecosystem consisting of an individual's knowledge footprint, mapped to a central knowledge graph, visualised over time as a learner's knowledge journey. This would serve to support the independent (aspiring or current) learner to pursue their life-long education. It would take into consideration the education they acquired from unstructured sources, thereby formalising their informal knowledge for themselves and others.

4.2 Challenges

There will be many challenges in coming up with solutions to enable knowledge acquisition and representation at the learner and group level. We will address some of the primary challenges to designing the components of the proposed knowledge ecosystem:

- Datasets
- DeepEdu dataset
- Knowledge graph dataset
- Deep Learning Architecture
- DeepEdu network

For our purposes, we will focus on the DeepEdu network and dataset as the main focus from here.

4.3 DeepEdu dataset

The success of deep learning techniques is predicated on the assumption that one needs a significant amount of data to train a large neural network to learn the representation that can capture a set of admissible functions needed to learn the given concepts.

The DeepEdu network will also face the same challenges. It requires a large dataset of educational content (i.e. Youtube videos, Wikipedia pages, Medium articles) along with a set of questions that have a set of correct and incorrect answers for each question.

$$\{content1 : \{question1 : \{answer1, answer2, ...\}, ...$$

Title	Concepts
Robert Sapolsky Human Behavioral Biology Lecture 1	neurobiology, n

However, there are many approaches we can decrease the amount of data needed while still learning a good representation of the concepts. We will present an approach that we believe can best utilise the intrinsic structure of educational content and use the minimal amount of data.

1. The network can be trained on unstructured educational content without questions or answers to create an *education embedding* by employing unsupervised techniques like variational auto-encoder (VAE). This requires a dataset of educational content, possibly many media types(audio/video/text). This could also be done by using a semi-supervised technique by supervising the media type and learning representations for the types of educational content.

2. The network can then be trained on **existing** educational content that *already* has preexisting questions and answers for specific parts of the content (i.e. Khan Academy, Udacity, Coursera). The domain of structured educational content is surely very different from the unstructured content and may hinder the network but we believe it might be the case that it helps more than it hinders so it is worth experimenting.

3. The final approach is annotating a set of unstructured educational content with questions and answers.

4.4 Education Partners

The second (#2) approach requires a large amount of existing educational content paired with existing questions and answers. This data is rich and informative; it could help us learn an early representation and create a strong baseline model. Most of these datasets have been created by experts and educators and their efforts could be valuable for our model to learn from. For this purpose, we would need to partner with large (in terms of a library of content) educational institutions (i.e. Udacity, Coursera, MOOCs, Edx, Khan Academy) to work with their datasets for pretraining and collaborate on how we annotate the space of unstructured educational content.

4.5 Educator Enrichment

The third (#3) dataset that would be used as our primary data source to train and test the model's performance, remains the most important component as well as the most time and labour intensive one. Partnering with educators and subject matter experts to source the subspace of unstructured educational content and to create questions and answers, we would eventually arrive at a curated dataset that our model would be primarily trained on. The end result would be a new benchmark that other researchers can begin to build newer architectures to make progress in the problem domain.

4.6 Minimum Viable Dataset (Benchmark)

We propose to narrow the problem space and jumpstart research in this area by focusing on only one subject (i.e. psychology, mathematics, design) rather than trying to take on mapping the whole space of unstructured educational content. This drastically reduces the data that is needed and the number of experts and partners needed to kickstart the initiative.

4.7 Call for collaborators

We hope to bring together collaborators that would include machine learning researchers, educators, technologists, and designers to begin our first foray into a modern knowledge ecosystem. Specifically starting with the DeepEdu dataset and network, one education subject, and at least one educational partner.

Please email us at `two@dyadxmachina.com` with the subject *Knowledge Ecosystem Initiative* or leave your email below for collaboration.

We are looking for educators, researchers, and designers to collaborate. Send us your email to stay updated

Email Address *

4.8 Conclusion

In conclusion, we presented a new perspective on knowledge acquisition and representation, proposed the concept of a modern and adoptive *knowledge ecosystem* that a learner can rely on upon through their entire educational lifetime.

Our main task was to take an individual and begin to get a true depiction of their knowledge beyond their traditional degree, which is only a small percentage of one's education.

We focused on taking the world of unstructured educational content online, and providing structure in the form of testing and mapping it to a *knowledge graph.* We introduced the ideas of **knowledge journeys** and **knowledge graph** as means to make sense and structure a learner's knowledge acquisition from 2 distinct perspectives - *temporal representation* vs *thumbnail profile.*

There is still much more research to be done in bringing to life the *Educational Content to Questions and Answers (DeepEdu)* neural networks as well our ultimate knowledge ecosystem and collaboration required amongst machine learning researchers, educators, and designers,

As deep learning researchers, we are looking forward to designing or seeing others design the minimum viable dataset, benchmark, and architecture motivated by this work.

5. ABOUT THE AUTHORS

Independent deep learning researchers focused on using machine learning for the good of humanity and beyond.

5.1 Haohan Wang

- LinkedIn
- Tumblr
- Github
- Email: `haohan723@gmail.com`

5.2 Fanli Zheng (Christian Ramsey)

- LinkedIn
- Tumblr
- Github
- Email: `thechristianramsey@gmail.com`

5.3 Contact Us

Feel Free to contact us if you have any questions!

- Our Website dyad x machina
- Our Hub dyad x machina projects
- Our Github dyadxmachina
- Our Book Mathematics for Deep Learning and Artificial Intelligence

References

[1] Aafaq, N. 2018. Video description: A survey of methods, datasets and evaluation metrics. (2018).

[2] Abu-El-Haija, S. 2017. YouTube-8M dataset. (2017).

[3] Chen, P. 2018. KnowEdu: A system to construct knowledge graph for education. (2018).

[4] Choi, E. 2018. QuAC : Question answering in context. (2018).

[5] Dong, X.L. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. (2014).

[6] Du, X. 2018. Harvesting paragraph-level question-answer pairs from wikipedia. (2018).

[7] Farag, Y. 2018. Neural automated essay scoring and coherence modeling for adversarially crafted input. (2018).

[8] Google 2015. Google knowledge graph. (2015).

[9] Kostadinov, S. 2017. How does the [current] best question answering model work? (2017).

[10] Kumar, V. 2018. A framework for automatic question generation from text using deep reinforcement learning. (2018).

[11] Li, Y. 2018. Visual question generation as dual task of visual question answering. (2018).

[12] Liu, Q. 2018. Generalized embedding model for knowledge graph mining. (2018).

[13] Olabiyi, O.O. 2018. Multi-turn dialogue response generation in an adversarial learning framework. (2018).

[14] Parton, B.S. 2016. Video captions for online courses: Do youtube's auto-generated captions meet deaf students' needs? (2016).

[15] Shi, Y. 2018. Question type guided attention in visual question answering. (2018).

[16] Song, L. 2018. A unified query-based generative model for question generation and question answering. (2018).

[17] Tapaswi, M. 2016. MovieQA: Understanding stories in movies through question-answering. (2016).

[18] Tran, D. et al. 2016. VideoMCC: A new benchmark for video comprehension. *CoRR*. abs/1606.07373, (2016).

[19] Wang, Z. 2018. QG-net: A data-driven question generation model for educational content. (2018).

[20] Xiao, H. 2018. Dual ask-answer network for machine reading comprehension. (2018).

[21] Zhao, Z. 2018. Multi-turn video question answering via multi-stream hierarchical attention context network. (2018).