## Introduction

Automatic question generation is part of Natural Language Processing (NLP). It is an area of research where many researchers have presented their work and is still an area under research to achieve higher accuracy. Many researchers have worked in the area of automatic question generation through NLP, and numerous techniques and models have been developed to generate the different types of question automatically. Work has been done in many languages.

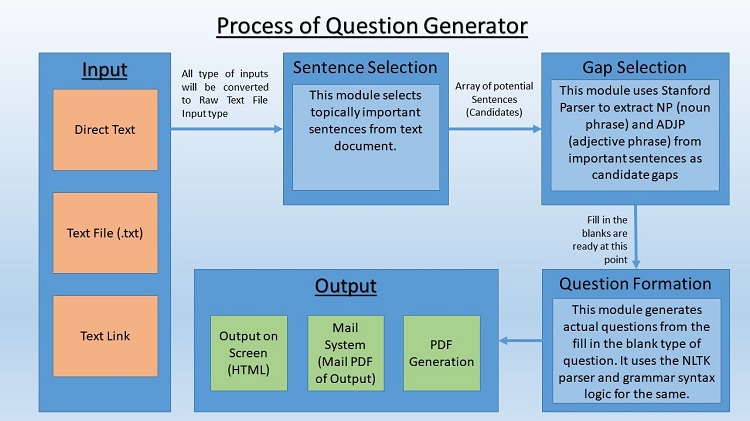
Nowadays, teachers/professors/tutors (academicians) spend a lot of time generating test papers and quizzes manually. Similarly, students spend a lot of time on self-analysis (self-calibration). Moreover, students are dependent on their mentors for the self-analysis. Hence, we are working on this NLP area, which has a huge scope of development at this moment. We want to build a computer application system that can help you in calibrating yourself and remove any dependencies on mentors. Here, students can give the input text of whatever material they referred to, and on this basis they get a set of questions with answers from which they can do a self-analysis (self-calibration). A similar approach is used by mentors for creating test papers and quizzes.

Moreover, online examinations have become very popular, including many major examinations, such as GATE, CAT, and NET. Multiple Choice Questions (MCQ) is very easy for evaluations, and its evaluation is implemented through computerized applications so that results can be declared within a few hours, and the evaluation process is 100% pure.

By making this computerized application, we can reduce the task of an educator. Much time can be saved if we can know what appropriate questions can be asked for the given input of text.

Hence, we want to develop a system which can generate various logical questions from the given text input. Right now, only humans are capable of accomplishing this.

## Implementation



Our system works with the following strategy:

Step 1. Select the best potential set of sentences from the given text input from which we could generate the questions. (Sentence Selection)

Step 2. Find the subject and context of the sentence to find its core agenda. (Gap Selection)

Step 3. Analyze which is the best form of question that can be generated from that sentence. (Question Formation)

### Step-by-Step Implementation

#### Input text

Hinton is a British cognitive psychologist and computer scientist, most noted for his work on artificial neural networks. Hinton was one of the first researchers who demonstrated the use of generalized backpropagation algorithm for training multilayer neural nets. He is a leading figure in the deep learning community. Hinton is called by some as the "Godfather of Deep Learning."

#### Preprocessed text

* Hinton is a British cognitive psychologist and computer scientist most noted for his work on artificial neural networks.
* Hinton was one of the first researchers who demonstrated the use of the generalized backpropagation algorithm for training multilayer neural nets.
* He is a leading figure in the deep learning community.
* Hinton is known by some as the "Godfather of Deep Learning."

#### Step 1 output: Potential set of sentences

* Hinton is a British cognitive psychologist and computer scientist most noted for his work on artificial neural networks.
* Hinton was one of the first researchers who demonstrated the use of the generalized backpropagation algorithm for training multilayer neural nets.
* Hinton is known by some as the "Godfather of Deep Learning."

#### Step 2 output: Subject and context of each sentence

Example sentence: Hinton is a British cognitive psychologist and computer scientist most noted for his work on artificial neural networks.

* Subject 1: Hinton
* Subject 2: a British cognitive psychologist and computer scientist
* Subject 3: work on artificial neural networks

The same is done for all other sentences that were selected in step 1.

#### Step 3 output: Question formation

We support two types of questions: fill-in-the-blank statements and answer in brief type of questions.

Example sentence: Hinton is a British cognitive psychologist and computer scientist most noted for his work on artificial neural networks.

Output of fill-in-the-blank statements:

* \_\_\_\_\_\_ is a British cognitive psychologist and computer scientist most noted for his work on artificial neural networks.
* Hinton is a \_\_\_\_\_\_.
* Hinton is a British cognitive psychologist and computer scientist, most noted for his work on \_\_\_\_\_\_.

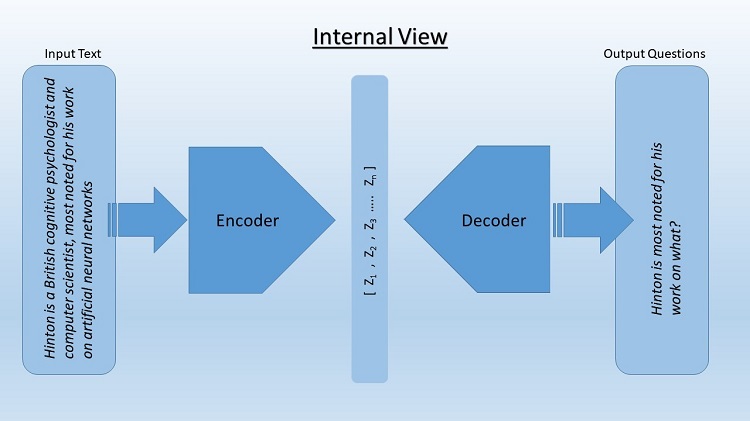
Output of fully stated questions (generated from the fill-in-the-blank statements):

* Who is a British cognitive psychologist and computer scientist most noted for his work on artificial neural networks?
* Who is Hinton?
* Hinton is most noted for his work on what?

## Ongoing Work

Until now, we have succeeded in forming two types of questions: fill-in-the-blank statements and fully stated questions (which are generated from the fill-in-the-blank statements). The second part (question generation from blanks) is mostly hardcoded right now.

Next, we want to implement it using encoder-decoder nets, which will increase the quality drastically. Encoder-decoder nets have been used by Google for its neural machine translation (language translation) and recurrent neural networks. By keeping encoder-decoder at the core, we also take help from Stanford Parser and NLTK for grammar analysis and more basic natural language analysis.



This image shows how encoder-decoder network works internally.

### Encoder

The encoder takes a preprocessed sentence from the input text and converts it according to the weights of the hidden layer. This hidden layer creates an intermediate representation of the input text and passes it to the decoder.

### Decoder

The decoder converts the hidden-layer information into question form. Machine translation uses the same concept. Here, we treat questions, essentially, as another language.

The programming is done with the help of Intel® Distribution for Python\*, which makes the working very fast and efficient. The speed boost comes from the Intel® Math Kernel Library (Intel® MKL), a collection of routines that use the capabilities of recent Intel processors to provide better performance for common data-science-related tasks, such as linear algebra or fast Fourier transforms. We use [Intel® AI DevCloud](https://software.intel.com/en-us/ai-academy/tools/devcloud) for testing our data models. The Intel AI DevCloud is a free cloud compute available for [Intel® AI Developer Program members](https://software.intel.com/en-us/ai/sign-up) powered by Intel® Xeon® Scalable processors for machine learning and deep learning training and inference compute needs.

## Conclusion

Our system can be used in multiple self-analysis scenarios. For example, students can use it to make learning easier as well as more interactive and interesting. Teachers and professors can use this system to quickly create a quiz. A central examination board can use this system to generate a unique test that is not known to any professor, eliminating the possibility of cheating and thereby securing the privacy and integrity of the examination.

There are very few competitors in the field of NLP. The major competitor is IBM Watson\*, which can answer any question but cannot (so far) generate questions themselves.